

2012

# New Game Physics - Added Value for Transdisciplinary Teams

Schiffler, Andreas

<http://hdl.handle.net/10026.1/923>

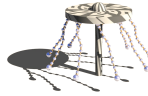
---

<http://dx.doi.org/10.24382/3878>

University of Plymouth

---

*All content in PEARL is protected by copyright law. Author manuscripts are made available in accordance with publisher policies. Please cite only the published version using the details provided on the item record or document. In the absence of an open licence (e.g. Creative Commons), permissions for further reuse of content should be sought from the publisher or author.*



---

**New Game Physics**  
**Added Value for Transdisciplinary Teams**

ANDREAS SCHIFFLER

---

# Appendix

Committee in Charge:

*Supervisor:*

Prof. Jill SCOTT

Zurich University of the Arts (ICS)  
University of Plymouth

*2nd Supervisor:*

Dr. Daniel BISIG

Zurich University of the Arts (ICST)  
University of Zurich (AIL)

March 11, 2012

# Contents

<b>A Personal Motivation</b>	<b>1</b>
<b>B Physics in Context</b>	<b>5</b>
B.1 Terminology and Scope . . . . .	5
B.2 Fields of Physics . . . . .	6
B.3 Characteristics of Physics . . . . .	9
B.4 Issues with Physics . . . . .	14
<b>C Examples of Game Physics</b>	<b>19</b>
C.1 Space Invaders . . . . .	19
C.2 Asteroids . . . . .	21
C.3 Battlezone . . . . .	23
C.4 Arkanoid . . . . .	25
C.5 Mario Bros. . . . .	27
C.6 Crazy Machines . . . . .	28
C.7 Battlefield 2142 . . . . .	31
C.8 FlatOut . . . . .	32

C.9	Half-Life 2 . . . . .	34
<b>D</b>	<b>Game Categories</b>	<b>37</b>
D.1	Action Games . . . . .	37
D.2	Strategy Games . . . . .	40
D.3	Adventure Games . . . . .	42
D.4	Simulation Games . . . . .	43
D.5	Puzzle Games . . . . .	44
D.6	Educational Games . . . . .	46
<b>E</b>	<b>Interviews</b>	<b>47</b>
E.1	Alen Ladavac . . . . .	47
E.2	Chris Crawford . . . . .	56
E.3	Danny Kodicek . . . . .	61
E.4	David Bourg . . . . .	72
E.5	Ed Rotberg . . . . .	78
E.6	Kevin Ryan . . . . .	87
E.7	Liemandt Foundation . . . . .	95
E.8	Matthew Wegner . . . . .	99
E.9	Max Behensky . . . . .	112
E.10	Megan Fox . . . . .	119
<b>F</b>	<b>Surveys</b>	<b>131</b>
F.1	Game Physics and Video Game Players . . . . .	131
F.2	Game Physics and Physicists . . . . .	188
<b>G</b>	<b>Science Art Reviews</b>	<b>207</b>
G.1	Artworks using known game physics design principles . . . . .	207

G.2 Artworks not applicable to game physics design . . . . .	209
<b>H Source Code</b>	<b>212</b>
H.1 Pendulum Game . . . . .	212
H.2 Gamebase Database . . . . .	213
H.3 Gamespot Database . . . . .	225
H.4 MAME Database . . . . .	236
H.5 Database Merge . . . . .	244
H.6 Graph Generation . . . . .	247
H.7 playtrulyrandom.com . . . . .	253
<b>I Video Documentation</b>	<b>272</b>
I.1 Script . . . . .	272
<b>Bibliography</b>	<b>282</b>
<b>Glossary</b>	<b>295</b>

# Abstract

**Andreas Schiffler (2011), ‘*New Game Physics: Added Value for Transdisciplinary Teams*’, Ph.D. University of Plymouth, UK.**

This study focused on game physics, an area of computer game design where physics is applied in interactive computer software. The purpose of the research was a fresh analysis of game physics in order to prove that its current usage is limited and requires advancement. The investigations presented in this dissertation establish constructive principles to advance game physics design. The main premise was that transdisciplinary approaches provide significant value. The resulting designs reflected combined goals of game developers, artists and physicists and provide novel ways to incorporate physics into games. The applicability and user impact of such *new game physics* across several target audiences was thoroughly examined.

In order to explore the transdisciplinary nature of the premise, valid evidence was gathered using a broad range of theoretical and practical methodologies. The research established a clear definition of game physics within the context of historical, technological, practical, scientific, and artistic considerations. Game analysis, literature reviews and seminal surveys of game players, game developers and scientists were conducted. A heuristic categorization of game types was defined to create an extensive database of computer games and carry out a statistical analysis of game physics usage. Results were then combined to define core principles for the design of unconventional *new game physics* elements. Software implementations of several elements were developed to examine the practical feasibility of the proposed principles. This research prototype was exposed to practitioners (artists, game developers and scientists) in field studies, documented on video and subsequently analyzed to evaluate the effectiveness of the elements on the audiences.

The findings from this research demonstrated that standard game physics is a common but limited design element in computer games. It was discovered that the entertainment driven design goals of game developers interfere with the needs of educators and scientists. Game reviews exemplified the exaggerated and incorrect physics present in many commercial computer games. This “pseudo physics” was shown to have potentially undesired effects on game players. Art reviews also indicated that game physics technology remains largely inaccessible to artists. The principal conclusion drawn from this study was that the proposed *new game physics* advances game design and creates value by expanding the choices available to game developers and designers, enabling artists to create more scientifically robust artworks, and encouraging scientists to consider games as a viable tool for education and research. The practical portion generated tangible evidence that the isolated “silos” of engineering, art and science can be bridged when game physics is designed in a transdisciplinary way.

This dissertation recommends that scientific and artistic perspectives should always be considered when game physics is used in computer-based media, because significant value for a broad range of practitioners in succinctly different fields can be achieved. The study has thereby established a state of the art research into game physics, which not only offers other researchers constructive principles for future investigations, but also provides much-needed new material to address the observed discrepancies in game theory and digital media design.

Keywords: *game physics, computer game design, transdisciplinary studies, digital art*

# Appendix A

## Personal Motivation

Throughout my childhood and pre-secondary education, play and more serious subjects have always been mixed together seamlessly and naturally. As 7 year old, my first science project was a working “crystal diode radio” circuit, which was quickly incorporated into a play situation as “hyperspace transmitter” of my cardboard-box spaceship. Only a few years later as my knowledge of electronics grew, driven by a fascination of pinball machines and arcade consoles, I built simple electronic games using digital CMOS chips and attempted to design magnetic pinball controls. Another hobby of mine which was rooted in the sciences is chemistry. It may have been initially the general fascination with gunpowder and solid fuel rockets, but I started and completed a vocational training program as Chemistry Laboratory Technician (CTA) during the last 3 years at the private high-school Odenwaldschule (OSO). And when other youth had a magazine subscription for Superman comics, I preferred one for *Scientific American*, which was my way of bringing a literary genre I liked, science fiction novels, come to life in reality.



Also throughout these years, the digital revolution entered into my field of view and I was determined to participate. So I started to self-educate myself in programming as soon as affordable personal computers such as the Apple II or the Commodore 64 became available. Of course the design of video games were part of the first steps I took in the world of computer science. But already during these early years, I had a fascination with the way science and digital technology intersected and programs I wrote in these early years revolved around algorithmic graphics, simple simulations, and fractal generation - a playful interaction with serious subject matter in mathematics and other natural sciences.

Thus it may have been these positive experiences that put me on a continuing search for new areas of the “serious, but ludic”. Having covered electronics and chemistry, and without an affinity to biology, I entered university as a physics major. My student work at the Institute of Space and Atmospheric Studies (ISAS) at the University of Saskatchewan, Saskatoon exposed me to structure world of scientific research. Because ISAS is an institute which is primarily focused on experimental physics, I had great fun “playing” with the large amounts of data generated by the experiments. Initially my work on auroral imagers exposed me for days on end to video material of the fascinating phenomenon of the *Aurora Borealis* - definitely an aesthetic experience. Later in my career I joined the SuperDARN radar group for my M.Sc. studies. During my work in this research group, I had yet another experience that solidified my belief in the merits a playful interaction with scientific data. While trying to understand the science behind the radar measurements, I produced a data visualization program which generated a dense grid of plots on a single screen allowing for rapid interactive exploration of the data. While scanning through tens of thousands of graphs, I spotted an artifact that had been previously overlooked in the analysis of the data using more conventional means such as printouts of fitted-lines. These newly found “double-peaked spectra” became later my masters theses topic,

the results of which are to this day under investigation by the researchers in the plasma physics field. Schiffler (1997)

The 6 year career period that followed the completion of my physics degree, included work that was much less scientifically oriented. Working as engineer, programmer and programmer-in-residence at the media arts center ZKM, Karlsruhe exposed me to artists and their methods of representation and interpretation. I programmed several interactive media installations during this time such as Jill Scott's *Interskin* (1996), Jeffrey Shaw's *Distributed Legible City* and Knowbotic Research's *IO\_Dencies*, and in retrospect, they all had an air of simulation with distinct ludic elements in them; not traditional "game physics", but artistic visions and combinations between the physical and the simulated through interactive virtual spaces. The years at the ZKM were also a time of relative freedom to explore some personal work and expand on ideas and concepts such as deconstructing the rapidly growing Internet with the game *FontAsteroids*, later shown at *net\_condition* in Karlsruhe, 1999 - a game based on the distinctly physical game control of acceleration and inertia copied from the classic arcade game *Asteroids* made by Atari. Schiffler (1999) During all these years, I was a knowledgeable but relatively restraint computer gamer: knowing about computer games and their technology seemed more interesting than actually playing these games excessively.

My general interest in computer games continued after I left the art space to join the "DotCom" boom as programmer and software engineer around 2000 and software development remains my primary profession to this day. And as the Internet and computer technology exploded and became exceedingly mainstream, the wave after wave of "new" computer games and game consoles left me relatively uninterested as I became increasingly critical towards this entertainment genre. The presented study is an attempt to not only combine my obvious interests and talents, but to also pro-actively engage

with some of the shortcoming present in the computer game entertainment industry today: to attempt to demonstrate that ludic or artistic methods and scientific principles do not need to be mutually exclusive in this medium and transdisciplinary approaches actually create a previously unseen quality and ultimately value to computer games and interactive media.

# Appendix **B**

## Physics in Context

The reader of this dissertation may benefit from some basic guidance regarding how the term *physics* is related to this study. Physics is popularly used as a relatively unspecific category for a field of science. For example, an Internet search for the word “physics” produces millions of links to a wide variety of websites. As this simple experiment indicates, the term is used in a much broader range of contexts and scopes than ever before, since it has evolved from its original meaning as a fundamental academic discipline into a multi-branched field of study with a wide range of applications.

### **B.1 Terminology and Scope**

The classical organization of physics into the fields of mechanics, acoustics, thermodynamics, optics and electromagnetism has been superseded by discoveries of the twentieth century – relativity theory, quantum theory, and others – so the field is now redefined to cover an extremely wide range of individual specializations and sub-fields of investigation into natural phe-

nomena. Within each field, the scientific inquiries revolve around a number of central theories, which are believed to be correct for a particular domain of validity (Newton 2007). It is important to note that this segmentation is a recent development, since the word *physics* has its roots in the Greek word for “natural” or “nature,” and stands for a collective science that attempts to describe the constituencies of the natural world and the forces they exert on each other. As a result of such wide-ranging applicability, the study and applications of physics are profoundly affecting all of us because the newfound understanding of the laws that govern the universe creates a deep relevance to humanity, for example to our security, livelihoods and economic abilities. It is not my intent, however, to pursue a broad science-theoretical discussion of physics or to perform an investigation of the various historical changes in our conception of the physical world, but to use physics as a source of scientific ideas that have a specific and applied cultural impact. This research thus treats physics as an important and rich collection of concepts where each individual theory and its many subtopics are believed to be relevant to individuals, our culture, and the human condition.

## B.2 Fields of Physics

In order to organize the various branches of physics as “resources” for this study, I have compiled a comprehensive list of fields of physics. The history of physics has its beginnings with Greek philosophers such as Leucippus<sup>1</sup>, Aristotle,<sup>2</sup> Archimedes,<sup>3</sup> and others. Their writings discussed areas such as mechanics, light, sound, electricity, magnetism and even atomic theory (Cajori 1917). One can clearly recognize these as fields of physics, thus making them possible candidates for the list. As Mill (1851, p. 367) shows, how-

---

<sup>1</sup>Leucippus wrote a first theory of atomism, 5<sup>th</sup> c. B.C.

<sup>2</sup>Aristotle described elements and dynamics in *Physicae Auscultationes*, 4<sup>th</sup> c. B.C.

<sup>3</sup>Archimedes laid the foundations of statics and hydrostatics, 3<sup>rd</sup> c. B.C.

ever, in his seminal work on inductive reasoning, an unreflected inclusion of these areas is problematic, since Greek physical inquiry was ultimately a failure due to its distinctly speculative character. Despite these shortcomings, Greek natural philosophy did develop mathematics and the concept of science as a rational, coherent system of knowledge. This development helped launch a seventeenth-century scientific revolution which included works by Copernicus,<sup>4</sup> Kepler,<sup>5</sup> and Galileo<sup>6</sup> amongst others. The key achievement of this era was Sir Isaac Newton's *Philosophiae Naturalis Principia Mathematica* (Newton 1687), which described the three laws of motion and the law of gravity. But even with these advances and eighteenth- and nineteenth-century contributions from Huygens<sup>7</sup>, Celsius,<sup>8</sup> Coulomb,<sup>9</sup> Faraday<sup>10</sup> and many others, the coverage of physical inquiry remained essentially confined to the same areas already known by the Greeks.

Only in the early twentieth century did it become clear that known descriptions of phenomena were insufficient to describe new observations and experiments involving extremely small sizes or very high speeds. As a result, new theories of quantum mechanics and relativity were described by Planck,<sup>11</sup> Heisenberg,<sup>12</sup> Einstein,<sup>13</sup> Feynman<sup>14</sup> and others which revolutionized the description of the natural world as significantly as the aforementioned initial Greek philosophies. The resulting paradigm shift was so abrupt and deep that it provided the basis for a clear split into two general groups, widely recognized today as *classical physics* and *modern physics*.

---

<sup>4</sup>N. Copernicus formulated the first heliocentric cosmology, 15<sup>th</sup> c. A.D.

<sup>5</sup>J. Kepler described planetary motion in *Mysterium Cosmographicum*, 17<sup>th</sup> c. A.D.

<sup>6</sup>G. Galilei experimented in the areas of mechanics, sound and light, 16<sup>th</sup> c. A.D.

<sup>7</sup>C. Huygens (1678) defined a wave theory of light.

<sup>8</sup>A. Celsius created a scientifically based temperature scale.

<sup>9</sup>C.-A. deCoulomb discovered electric charges and properties of electrostatics.

<sup>10</sup>M. Faraday theorized on electromagnetism and electrodynamics.

<sup>11</sup>M. Planck is considered the founder of quantum theory.

<sup>12</sup>W. Heisenberg contributed to nuclear physics and quantum field theory.

<sup>13</sup>A. Einstein (1916) created the special and general theories of relativity.

<sup>14</sup>R. Feynman worked on a theory of quantum electrodynamics and superfluidity.

The subset of *classical physics* comprises those areas that do not rely on theories containing elements from quantum mechanics or Einsteinian relativity, or both. These fields and sub-fields include:

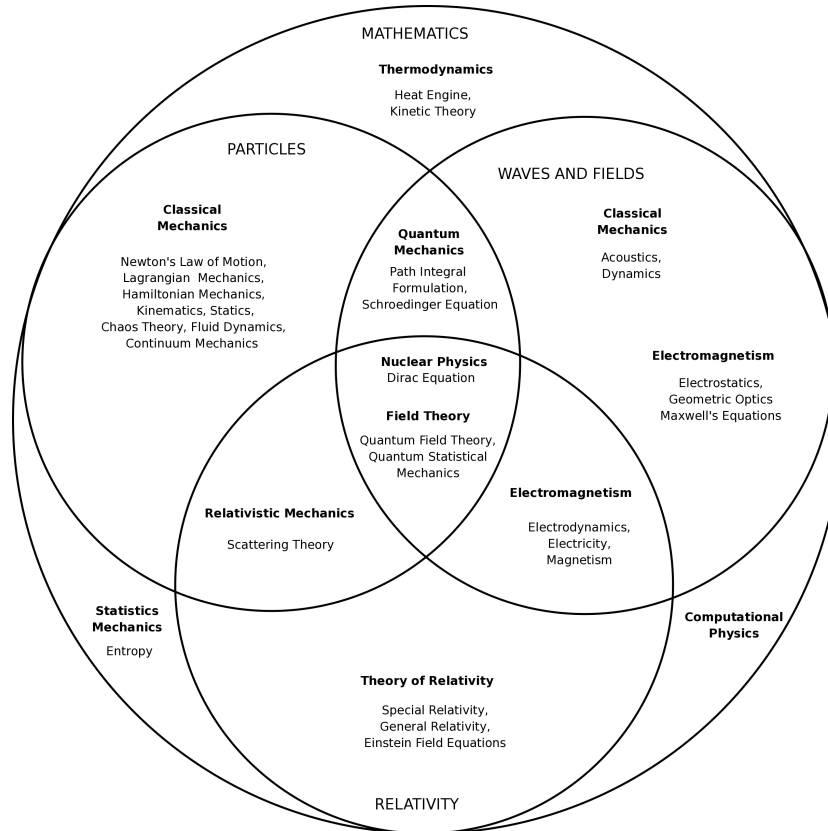
- Classical mechanics (Newton's Law of Motion, Lagrangian Mechanics, Hamiltonian Mechanics, Kinematics, Statics, Chaos Theory, Fluid Dynamics, Continuum Mechanics, Acoustics, Dynamics)
- Thermodynamics (Heat Engines, Kinetic Theory) and statistical mechanics (Entropy)
- Classical electrostatics and dynamics (Geometric Optics, Maxwell's Equations, Electricity, Magnetism)

The remaining fields form the subset of *modern physics*, characterized by a need to incorporate the new theories of the twentieth century. Modern physics covers an extremely wide range of individual specializations and sub-fields including Information Technology (IT)-based investigations. The following fields primarily comprise this group:

- Quantum mechanics (Path Integral Formulation, Schrödinger Equation) and quantum thermodynamics
- Special and general relativity
- Nuclear physics (Dirac Equation)
- Quantum field theory (Einstein Field Equations) and quantum statistical mechanics (Scattering Theory)

In summary, physics consists of a collection of effective theories split into the two general subsets of classical and modern physics. These may be orga-

nized in a hierarchical as well as overlapping manner using a *Venn diagram*<sup>15</sup> as shown in figure B.1.



**Figure B.1:** Overview of fields and sub-fields of physics organized in a *Venn diagram*

## B.3 Characteristics of Physics

Each physics field listed in section B.2 contains theories with possible relevance for computer game simulations. It is necessary, however, to identify

<sup>15</sup>Venn diagrams are illustrations used to show the possible mathematical or logical relationships between sets - commonly 3 sets are used - and were introduced by philosopher J. Venn in the 19<sup>th</sup> century (Venn 1880).



some of the key characteristics of physics, in order to qualitatively evaluate such applications and to identify shared characteristics that span all fields of physics.

One of the most important characteristics of physics is a focus on the rigorous application of the scientific method. The basis for the scientific revolution over the last millennium was rooted in the recognition of the benefits of generalized methods for seeking “the truth.” The definition of identifiable features that distinguish scientific inquiry from other methodologies of knowledge production was (and still is) not straightforward, as the following quote (believed to be one of the first citations of the scientific method) illustrates:

”Truth is sought for its own sake. And those who are engaged upon the quest for anything for its own sake are not interested in other things. Finding the truth is difficult, and the road to it is rough.” (Sambursky 1974, from Ibn al-Haytham, 965–1039 AD)

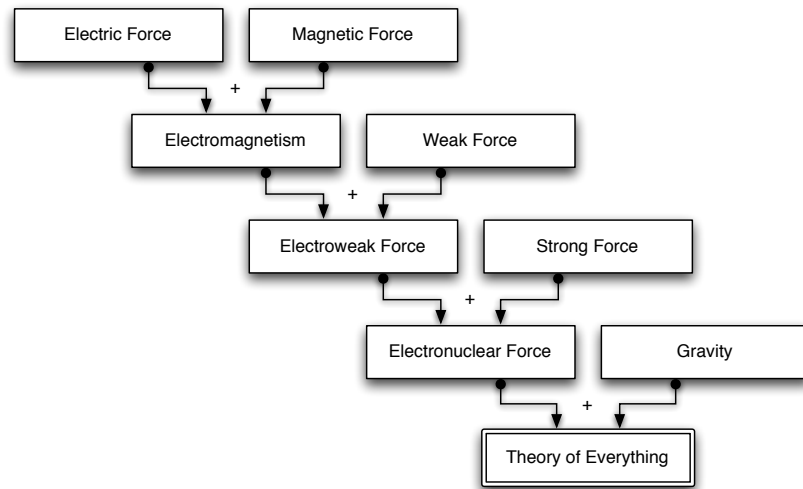
Today, most sciences attempt to acquire knowledge and to formulate laws based on physical evidence and experiments. This multi-stage technique of gathering empirical evidence and subjecting it to specific principles of reasoning is called the *scientific method*. The stages of this process were summarized by philosopher Russell (1931) as follows: “The first [stage] consists in observing the significant facts; the second in arriving at a hypothesis, which, if it is true, would account for these facts; the third in deducing from this hypothesis consequences which can be tested by observation.” Thus it seems that physics derives its scientific successes through a general agreement by participants in the research to use the common *hypothetico-deductive* thought model. (Godfrey-Smith 2003, p. 236) This model iterates over four core elements:

1. **Characterizations** (observations, definitions): Start with existing experiences, consider the problem and try to make sense of it. Look for previous explanations; if this is a new problem, then continue with (2).
2. **Hypotheses** (theoretical, hypothetical explanations): Conjecture an explanation for the problem. When nothing else is yet known, try to state an explanation.
3. **Predictions** (reasoning, logical deduction): Deduce a prediction from that explanation. If (2) were true, then state a consequence of that explanation.
4. **Experiments**: Test the conjecture by looking for the opposite of that consequence in order to disprove (2). It is a logical error (affirming the consequent) to seek (3) directly as proof of (2).

Historically, advances in physics have been strongly connected to the application of the scientific method. For example, the physicist Ibn al-Haytham quoted earlier is credited to have applied the scientific method for the first time in his “Book of Optics” published in 1021 to successfully advance the understanding of the physics of light (Gorini 2003). Similarly, Galileo launched the scientific revolution of the seventeenth century by breaking with the tradition of Aristotelian science and documenting his results of physics experiments purely as mathematical constructs (Feldhay 1998). And even though contemporary science philosophers such as Kuhn (1962) debate what constitutes the scientific method, it still remains the de facto standard for all current physics publications. Its uniformity allows for an independent peer review at any stage of the process and reduces the influence of an experimenter’s bias, such as personal or cultural beliefs. I believe that it is this property of the scientific method that makes it a good tool to transfer information across different disciplines and thus makes it relevant for this research.

Another shared characteristic, which is at the core of physics, is a belief that an underlying simplicity and unity in nature does exist. Ever since the theories of the scientific revolution of the seventeenth century explained the complicated motions of the celestial bodies through a small set of laws, physics has succeeded repeatedly in explaining increasingly diverse phenomena with only a small set of underlying principles (Taylor 2001). The emerging study of complex systems tries to explain chaotic physical phenomena in terms of simple behaviors. Crutchfield (2008), the director of the Complexity Sciences Center at the University of California, writes about this new sub-branch of physics called *complexity theory*: “In looking back at this history, it becomes clear that randomness from simplicity and order from complication are two sides of the same coin and that coin was the concept of pattern and pattern emergence.” Even entire theories have been labeled with the “unity” moniker such as the Grand Unified Theory (GUT) which describes a model in particle physics. Similarly the ongoing struggle to define a fundamental framework of physical laws, called the Theory of Everything (TOE), is based on patterns observed in existing theories (see figure B.2). These historical and modern processes operating within physics research are clear examples that physicists have shared goals about the unification of theories that describe nature. The significance of this characteristic for new game physics will become apparent in section ??, which describes how a similar vision is often shared by artists who use science in their works.

Physicists have always relied on mathematics to describe nature, indicating that the usage of mathematics as an underlying logical language is yet another characteristic. Galileo used techniques derived from Euclid’s *Elements* to investigate concepts of velocity and acceleration. Science historian Boyer (1949, p. 4) even claims that “calculus had its origins in the logical difficulties encountered by the ancient Greek mathematicians in their attempt to express their intuitive ideas on the ratio and proportionality of lines, which they vaguely recognized as continuous, in terms of numbers, which they re-



**Figure B.2:** Pattern of physics theories leading to the *Theory of Everything*; adapted from Wikipedia (2010f)

garded as discrete.” Knowledge of calculus is indeed fundamental in applied physics and thus treated as “a rite of passage for the modern MIT student into the world of science and engineering.” (Pritchard 2010) The American Institute of Physics in its dedicated journal defines *mathematical physics* simply as “the application of mathematics to problems in physics and the development of mathematical methods suitable for such applications and for the formulation of physical theories.” (AIG 2010) Indeed, many recent advances in physics<sup>16</sup> were only made due to the key role mathematics played to *mechanize* thinking (Kline 1959, p. 68). In general, therefore, it seems that mathematics is an integral part of the language of physics and thus needs to be considered even when physics is brought outside a scientific context.

The last characteristic discussed in this section, is the importance that

---

<sup>16</sup>For example, the relativistic model of the electron created by Dirac (1928) was used to predict the electron’s magnetic moment as well as its antiparticle, the positron, and was purely based on applied algebra.

physics places on accuracy and the quantification of precision. The study of physics always includes lessons in error analysis, an acknowledgement of the inevitability of uncertainty associated with measurements and that “the best you can hope to do, is to ensure that errors are as small as reasonably possible and to have reliable estimates of how large they are.” (Taylor 1997, p. 3) In physics, uncertainties have an equal importance to the actual measurements, in part due to requirements implied by the scientific method, such as to design repeatable experiments. More recently, physics has recognized another reason to strive for accuracy. Ye et al. (2009) posits that an “improvement of measuring precision in physics by an order of magnitude often implies a new or unknown effect to be explored, consequently even a new physical law to be established.” To point out that additional research is needed to make measurements more precise may be a self-serving statement by physicists to secure government funding, but scientific results are often used as input for policy decisions, which may lead to extensive debates around the *precision* of a theory or experimental datasets.<sup>17</sup> Thus precision provides a form of security for physicists in these debates. Even theoretical research that focuses on principles of abstraction and generalization, must ultimately lead to experimentally verifiable results and is thus indirectly linked to criteria of accuracy. It follows from all these examples, that precision is an inherent quality of the information and data used in physics as well as a key motivator for research activities.

## B.4 Issues with Physics

The preceding section listed some characteristic elements of physics in order to inform the analysis of the transformation of physics theories into the realm of computer games. This section has a similar goal, but will rather attempt

---

<sup>17</sup>A good example of physics data surrounded by controversy and public debate are the geophysical measurements supporting the existence and severity of global warming.

to identify problems and issues that relate to the perception of physics by lay people and physicists themselves. How does the general public or those not involved in scientific study feel about physics? Since most players confronted with physics in a computer game are not scientists, the answers to this question will be relevant for this research into game physics.

Generally, society has a positive attitude towards most mainstream fields of physics, because investments into these areas appear to be producing tangible benefits in the form of knowledge and wealth. The problem is that many physics fields are being categorized as *fringes of physics*<sup>18</sup> as soon as their language, methods, benefits or the scale of the experiment<sup>19</sup> becomes incomprehensible. Such fringes are perceived more negatively than well-known fields and they tend to be dismissed as nothing more than the elaborate intellectual exercise of a few elitists. A contributing perception is that these physicists can't find a lingua franca with laymen outside their *ivory towers* (James E. Cote 2007). The problem is aggravated when the scientists themselves accept that "even educated people can't understand them anymore" and so retreat into intellectual isolation. Furthermore, legitimate research conducted in fringe fields is further discredited when lumped together with "pseudo physics," a practice of physics that often contradicts well-established theoretical or experimental results and is characterized by a lack of falsifiable predictions and peer review. This misunderstanding may cause the public to perceive unspecified threats from physics, creating a fear which tends to dominate the conversation.<sup>20</sup> In summary, fundamental physics research is sometimes perceived negatively by the public due to *fringe* labeling, lack of

---

<sup>18</sup>Examples of current research considered *fringe physics* include cold fusion (Fleischmann & Pons 1989), anti-gravity (Tajmar et al. 2006), zero-point energy, polarizable vacuum or the Heim theory (Heim 1977, Prager 2009).

<sup>19</sup>An example of such *big science* would be the Large Hadron Collider (LHC) at CERN, Switzerland with its multi-Billion US\$ budget.

<sup>20</sup>For example the science-fiction movie *Contact* (Zemeckis, 1997) uses a popularized version of such debates as main plot element. It also occurred when the LHC went into operation in 2009 (CERN 2008).

Classical Physics	Modern Physics
Substance only (materialism)	Both substance & event descriptions
Continuity only	Both continuity & quantization
Symmetry only	Both symmetry & asymmetry
Space only; spatialized time	Both space & time (coupled)
Determinism only	Both determination & indetermina- tion
Particles only	Both particles & waves
External relations only	Both external & internal relations
External sources of order only	Both external & internal (self- organizing) order

**Table B.1:** *Various Dualities in the Transition from Classical to Modern Physics*, adapted from (Eastman 2004, p. 23, table 2.2)

communication or a perception of fear when media fictionalizes and sensationalizes research. Negative attitudes towards physics are likely to affect the perception of physics in games.

Another problem originated with the introduction of modern quantum physics (see section B.2) in the twentieth century when a problematic dichotomy in the perception of physics was created. Our worldview of perceptual objects is dominated by classical physics, which has no counterpart in quantum physics. During the transition from classical to modern physics, Newton’s discrete and self-identical *classical* objects were replaced with interacting and stochastic wave-particle entities. The resulting dichotomy was resolved within the science of physics by introducing dualities of complementary pairs of entities as summarized in table B.1.

And how does this aspect affect public perception? These newly introduced dualities solve the “*puzzles* of physics ... which depend on the presupposition of perceptual objects rooted in classical physics.” (Eastman 2004, p. 22) However, they are also much more difficult to understand and commu-

nicate. This difficulty maintains the dichotomy in the public sphere where physicists on the one hand and the untrained public on the other, operate with mutually exclusive perceptual views of the world. There is ample evidence that this is a broad issue lacking sufficient discourse. Physicist and novelist Snow (1959) describes in the influential lecture *The Two Cultures and the Scientific Revolution* the rift between the sciences and both humanities and arts, and points out that such dichotomies are common in Western thought.<sup>21</sup> Many physics educators also struggle to help students to *unlearn* the misconceptions and “common sense” ideas about physics that have been found to be very widespread<sup>22</sup> and are a direct result of the “classical” concepts acquired while growing up (Freedman 1996). Taken together, these examples suggest that there are significant challenges when topics in modern physics are used or communicated. Thus, modern physics warrants additional attention in the context of this research, in particular for transdisciplinary game design teams.

While physics has been undoubtedly successful over its long history, some physics practitioners today experience a lack of clarity in how to make further progress. One cause may be that philosophers of science suggest various critiques of physics. Babin (1989) posits that there are “no experiments which can lay claim to *ultimately* or *absolutely* proving anything,” an argument that would seem to undermine the previously mentioned search for a unified theory of nature. In his book *A Critique of Pure Physics*,<sup>23</sup> the poet Neubert

---

<sup>21</sup>Snow’s theory has since been disputed by Brockman (1991, 1995), who argues that technology is a bridge which has created a “Third Culture” in which leading scientists and thinkers contribute their thoughts in plain language. This discourse has been related by German author Gábor Paál to Hegel’s idea of *Realphilosophie* (Hegel 1805) which emphasizes thinking on an empirical basis (Paal 2009).

<sup>22</sup>Illustrative examples of such erroneous ideas which are robust and difficult to dislodge from students’ minds, have been presented in studies done by Halloun & Hestenes (1985) for mechanics, by McDermott & Shaffer (1992) for electricity, and by Goldberg & McDermott (1987) for optics.

<sup>23</sup>A title chosen as analogy to Kant’s famous *Critique of Pure Reason* (Kant 1781).



(2009, p. 9) finds “significant cracks in the bedrock of pure physics” and describes many common issues in physics, including “conceptual disagreements, alternative interpretations of data, theory that goes far beyond experiment, unsolvable problems, subspecialty bias and extraordinary assumptions that paper over deep problems” – all of which undermine the credibility of physics.

Many in the research community choose to ignore such critiques, but some scientists do call for radical change in the way physics is conducted. The key to make progress in physics today as string-theorist Greene puts it, is for physics to “shift from a small element of perception all the way to a monumental rethinking.” (Scientific American 2003) Another group of physicists are at odds with the status quo of the academic process surrounding physics. The journal *Progress in Physics* describes in the preamble of its “Declaration of Academic Freedom” the record of scientific discovery as “replete with instances of suppression and ridicule by establishment” and “blighted and besmirched by plagiarism and deliberate misrepresentation, perpetrated by the unscrupulous, motivated by envy and cupidity.” (Rabounski 2005) These observations demonstrate some prominent examples of the discrepancy that exists between a perception of physics as *hard science* that has a clearly defined path for making progress and a reality which is much more opaque. I believe that computer games may actually help to deconstruct such limited perceptions and could even provide novel tools to physicists, facilitating the implementation of the proposed “radical change.”

# Appendix C

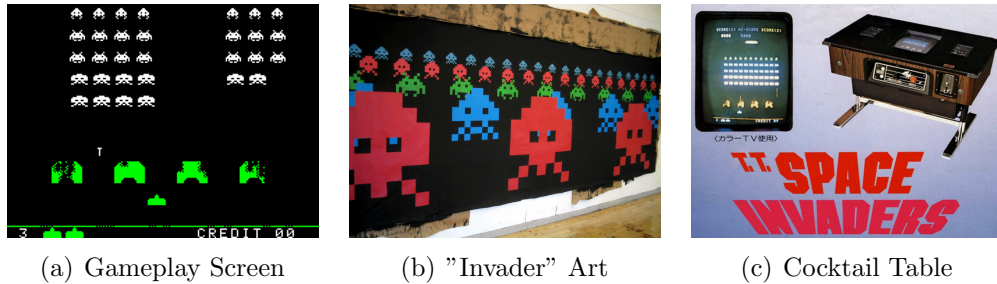
## Examples of Game Physics

This chapter provides some illustrative examples of game physics with a focus on either object dynamics or narrative forms. Because all three-dimensional (3D) games tend to employ the game physics element of virtual-space simulation, this element does not have to be demonstrated separately. The analysis of incorrect physics or other simplifications supports the section on *Pseudo Game Physics* of this dissertation.

### C.1 Space Invaders

The game *Space Invaders* was originally manufactured by Taito and released in 1978. It ranks as one of the most influential computer games (Wikipedia 2010*d*). The gameplay is simple: a horizontally movable laser defends itself against attacking aliens which approach from the top of the screen and drop bombs on shelters and the player's avatar. The game ends if an alien reaches the bottom of the screen or all player avatars have been destroyed. The pixelated alien graphics shown in figure C.1(b) is a good example of a game

element that has become a powerful cultural icon.



**Figure C.1:** The Arcade Game *Space Invaders* (Taito Corp., 1998)

From a game physics point of view, this game is important because the use of *gravity*, this simple but extremely common game physics element was established with this game. Gravity is simulated by the constant motion of the “bomb” object from the top to the bottom of the screen as if moving in a gravitational field (i.e., falling towards Earth). This element was immediately recognizable by the game players, because the predominant orientation of screens in arcade consoles (and later TV sets used for home entertainment) is vertical. This game physics element is still used in almost all 2D games today – in particular, in all variations of *Space Invaders* such as *Galaxian* (Namco, 1979) or *Galaga* (Namco, 1981), as well as in the popular genre of Platformer games such as *Mario Bros.* (Nintendo, 1983).

It is important to note that the game physics in *Space Invaders* is not physically correct, due to its extremely simplified implementation. Physical gravity would constantly accelerate the bombs dropped by the aliens and therefore would result in a squared displacement over time as  $\Delta y \propto \Delta t^2$ . The observed motion of the bombs in the game, however, is linear and follows  $\Delta y \propto \Delta t$ . The only way a linear motion could be achieved is in the presence of an atmosphere, where any dropped object would accelerate until it reaches a terminal velocity. The observation therefore contradicts the setting of the

game in *outer space*, where no atmosphere exists.<sup>1</sup>

Another relevant fact is that the perception of the game physics by the player was already instrumentalized in a game design modification. After the initial excitement about the game had passed and sales dropped, the manufacturer released a “Cocktail Table” version, shown in figure C.1(c), which simply rotates the screen into the horizontal position to skew the player’s physical frame of reference. This provided a slightly more difficult game experience for the player, although exactly the same software was used.

## C.2 Asteroids

In 1979, Atari Inc. released the game *Asteroids*. It was one of the most popular and memorable games of the arcade era (The Atari Library 2010). In this game, the player controls a spaceship in an asteroid field which is periodically traversed by flying saucers. The object of the game is to shoot and destroy asteroids while avoiding being hit by them or alien saucers’ counter-fire. As the player shoots asteroids, they break into smaller asteroids or disappear until the playing field is cleared (Wikipedia 2008b).

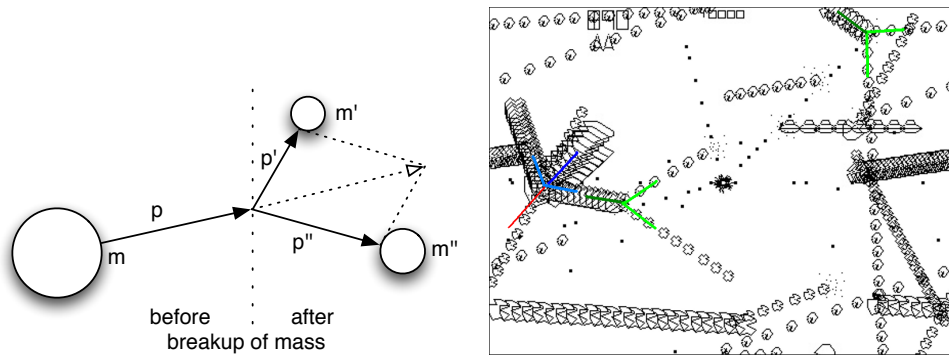
In a modern review by Juul, he dismisses the game as “forgotten futurism” which is “almost good ... oddly empty” and provides “no progress” (von Borries et al. 2007, pp. 34–35). However, the game was a great commercial success due to its innovative use of game physics in the form of *inertial control* and *object dynamics* which it had borrowed from an earlier game entitled “Spacewar!”<sup>2</sup> The spaceship is represented as a simple triangle that

---

<sup>1</sup>In fluid dynamics, *terminal velocity* is the velocity at which the fluid resistance force (drag force) of a falling object equals the weight of the object minus the acting force due to the fluid, which halts acceleration and causes the speed to remain constant. At high speeds, an atmosphere acts like a fluid on objects.

<sup>2</sup>The basic gameplay in *Spacewar!* (Steve Russell et al., 1962) involves two armed spaceships called “the needle” and “the wedge” attempting to shoot one another while

can be rotated and to which thrust can be applied using a button controller. This means the player's avatar is maneuvered using a two-dimensional (2D) zero-gravity inertial simulation. If no thrust is applied, the ship would still drift according to its acquired momentum, making navigating around the screen a fun challenge for the player. The design of the asteroids – the main game objects – also attempts to simulate physically accurate dynamics: when hit, they break apart into smaller pieces which fly off into separate directions, seemingly conserving linear momentum as shown in figure C.2(a), to create a realistic display of space debris.



(a) 2D Conservation of Linear Momentum (b) 24 gameplay frames combined; green: correct, blue: incorrect, red: error

**Figure C.2:** Error in momentum conservation simulation for  $\vec{p} = \vec{p}' + \vec{p}''$  in the game *Asteroids* (Atari, 1979)

However, two violations of physics laws can be observed here. First, as the ship moves, momentum is not conserved and the ship eventually comes to a stop when the player is not thrusting. This friction does not seem to apply to the asteroids moving in the same spatial medium. Second, there are occasions where the simulation of an asteroid's destruction does not correctly conserve linear momentum laws, as illustrated in figure C.2(b) by lines.<sup>3</sup>

maneuvering in the gravity well of a star. Player controls in *Spacewar!* included clockwise and counterclockwise rotation, thrust, fire, and hyperspace.

<sup>3</sup>Green and blue lines are vector representations of the observed asteroid momentum

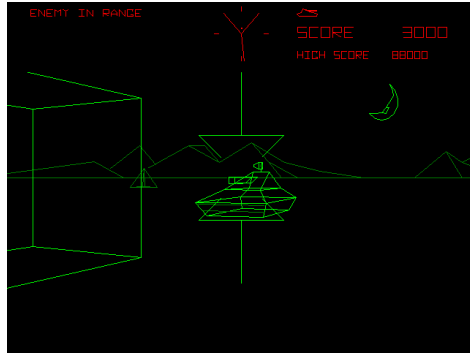
## C.3 Battlezone

By 1980, Atari Inc. was the recognized leader in computer games because they had established the arcade industry with the production of several popular games (Cohen 1987). In 1980, they released the single-player simulator game *Battlezone* (BZ). The game was a seminal, 3D wireframe “Shoot-em-Up” in which the player views the action from inside a tank (The Atari Library 2006). The principal programmer Ed Rotberg commented on the difficulty of dealing with representations in the game world: “Given the technology that we had, the real challenge was how to make the game appear as if we had more technology than we did.” (Bousiges & Butler 2007) The game is set on a plain surrounded by a mountainous horizon, active volcanoes, a crescent moon, and various objects drawn with vector outlines, as shown in figure C.3(a). *Battlezone* was the first environmental 3D landscape game because of these visual features. Players were immersed into a “surrealistic landscape,” as each level was littered with simple geometric objects like boxes and pyramids representing buildings (Wikipedia 2006a).

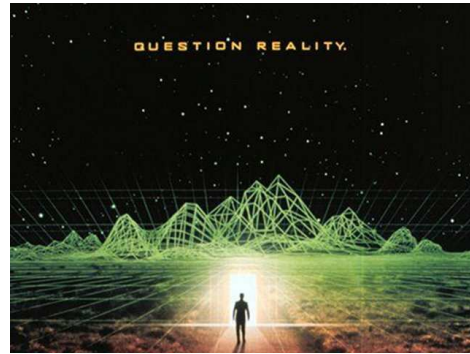
*Battlezone* is also considered the first commercial VR game because it combined actual goggles and a realistic two-joystick steering system with the innovative implementation of first-person 3D graphics to create a convincing effect. Comments from players of the game proved that the spatial experience, even in its crude wireframe mode, attracted them to the game (Bousiges & Butler 2007). The games’ visuals had significant impact in popular media and design as well, as exemplified by the appearance of a console in the genre-defining movie *Tron* (Lisberger, 1982) or the use of its CRT-green lines to identify cyberspace in *The Matrix* (Wachowski, 1999) or *The 13th Floor* (Rusnak, 1999, shown in figure C.3(b)). Even in a more artistic statement, a

---

in the game. The red line indicates an instance of “missing” momentum which would be needed to satisfy physical conservation laws.



(a) Gameplay screen (building & tank)



(b) *The 13<sup>th</sup> floor* (1999) film poster

**Figure C.3:** *Battlezone* (Atari Inc., 1980) creates convincing VR using just wireframe rendering

wireframe world was used as the backdrop in Beck Hansen’s video *E-Pro* to parody our current technology-centric culture (Hansen n.d.). These examples illustrate an artistic strategy to simplify the graphics, in order to shift the perception of the viewer to behaviors deemed more interesting by the artist; the strategy will be discussed again in section ?? in relation to game physics design principles.

In *Battlezone*, game physics is associated with moving objects, which are rendered more accurately than static ones. These objects move around freely in the game space, only restricted by other objects and *distance culling*.<sup>4</sup> When an enemy object is hit, it breaks up into fragments, which are visualized as “eye-candy”: the pieces fly in an explosion pattern and rain down in what appears to be a physically correct parabolic motion. But the effect is not real physics, as programmer Rotberg attested: “The physics pretty much stopped once each piece hit the ground ... the rumbling rotations of the pieces in flight were pre-canned and about the arbitrary graphical center of

---

<sup>4</sup>*Distance culling* is the process of skipping over objects during scene rendering based on an algorithmic visibility test.

the piece. No attempt to establish the true center of gravity was made.” (Rotberg, pers. comm. 2006, see append.) Perhaps, if the game designers had better Central Processing Units (CPUs) in these arcade consoles, they would have modeled the simulation more accurately, especially given the game developers’ academic background (see interviews by author in section ??).

## C.4 Arkanoid

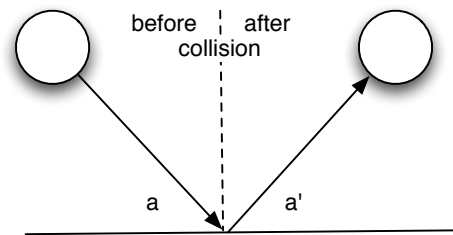
The gameplay principle of the popular game *Arkanoid* (Taito, 1986) is simply to move a “paddle” to prevent a ball from exiting the playing field. The player attempts to bounce the ball against a number of bricks positioned in the upper area of the screen. Whenever the ball hits a brick, it causes that brick to disappear, yields points and sometimes provides “power-up” items that briefly change the game rules. When all the bricks have vanished, the player progresses on to the next level, which contains a new pattern of bricks. (Wikipedia 2008a)

This game was a modern version of Atari’s *Breakout* (1976) that was itself an advancement of the classic *Pong* (Atari, 1972), a seminal game that started the commercialization of computer games. All these games made use of a simple physics principle, the *Perfectly Elastic Collision*, which is illustrated in figure C.4(a). This physical law describes the angle of an object, which is bounced off an obstacle under certain conditions (no friction loss, flat obstacle, no momentum transfer, no rotation) and provides the complete gameplay dynamics of these types of games. None of these games consider any conservation of energy laws, since the balls do not slow down during normal play nor are they visibly accelerated when entering the screen.

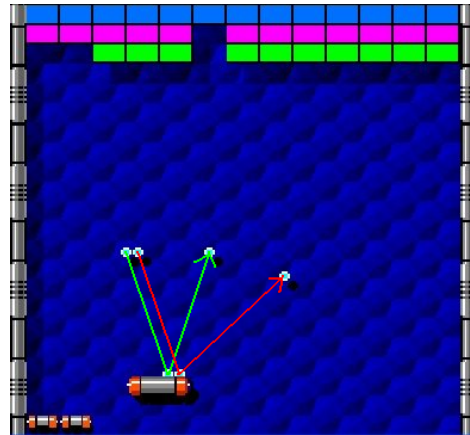
One of the “power-up” capsules that can be collected by the player in-



stantaneously doubles the number of balls. These balls then travel side by side with the same direction and speed. A mistake in the simulation of the physics is revealed when both balls hit the paddle. By overlaying multiple gameplay images with the motion vectors of both balls (see arrows in figure C.4(b)), one can observe similar incoming but different outgoing angles. This is inconsistent with the physical law applicable to the shown geometry. The algorithm simply adds a constant or arbitrary angle component when the ball is close to the edge of the paddle, probably to simulate deflections on a rounded corner. This implementation was likely a design decision to introduce randomness and thereby make the game harder to play. This could be perceived as “sloppy” simulation, in particular in light of the evidence presented in this text.



(a) Law of Perfectly Elastic Collisions



(b) 2 frames combined; ball splits

**Figure C.4:** In *Arkanoid* (Taito, 1986) the law of perfectly elastic collisions ( $a = a'$ ) is broken during “ball split”

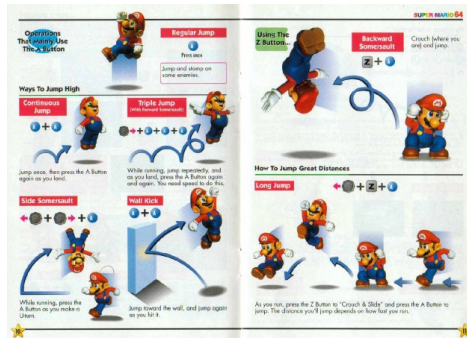
## C.5 Mario Bros.

An archetypical “platformer” type of game<sup>5</sup> is the Nintendo classic *Mario Bros.*, released in 1983 (Wikipedia 2010*c*). A follow-up version was *Super Mario Bros.*, which was released in 1985. This version of the game was bundled with each Nintendo Entertainment System (NES) console, a strategy that generated a very wide distribution of the game. It featured gameplay that changed the way computer games were created, because it used an industry-first technology of smooth-scrolling levels (Wikipedia 2010*e*). Incidentally, this feature of game physics was based on motion dynamics (see figure C.5(a)) and made it a landmark in home entertainment. The avatars’ (the characters “Mario” and “Luigi”) method of primary attack is to jump or “stomp” on top of their enemies. Mario can also kick shells into other enemies and the kicked shells can bounce back off walls or other vertical obstructions. It is this physics-based dynamic motion of the game objects, such as the parabolic trajectory of the avatar jump or the frictionally decelerated bounce of a shell that brings life to the game and makes it such a genre-defining classic.

On close examination, an incorrect implementation of game physics can be found when the algorithm reverts back to the linear gravity of the *Space Invaders* era (see section C.1) once the game character *Mario* “dies.” As the overlay of several game sequence images reveals (see figure C.5(b)), pseudo physics was used when the avatar icon is moved off-screen to indicate game-death. A linear and non-physical position update of  $y \leftarrow y + \delta y$  was used for Mario in the game loop, whereas other game objects (i.e., the turtle in the example) are still animated in a physically correct way.

---

<sup>5</sup>Platform games originated in the early 1980s, and 3D successors were popularized in the mid-1990s and were at one point the most popular genre of computer games. The common unifying interface element of these type of games is a jump button.



(a) Manual of *Super Mario 64*



(b) 12 frames combined; Mario “dies”

**Figure C.5:** Gravity is simulated incorrectly (linear) when Mario “dies” in *Mario Bros.* (Nintendo, 1983)

## C.6 Crazy Machines

Another relevant game type that uses game physics is the puzzle game. The game *Crazy Machines* (FAKT Software GmbH, 2005) was a modern version of the classic game *The Incredible Machine* (TIM) by K. Ryan (see also section ??), which was released and rereleased in various versions in the 1980s. The main objective in the game is for the player to solve a puzzle by creating a series of Rube Goldberg devices, which consist of a collection of game objects arranged in a needlessly complex fashion to perform a simple task (i.e., place a ball into a bucket). The gameplay revolves around building imaginative machines that can, for example, turn cranks, rotate gears or pull levers. These elements are then animated using the physics engine. It even features a controllable physical environment where the player may change gravity or friction. The physics engine also covers quite a wide range of physics fields through its simulation of air pressure, electricity, gravity, light and particle effects. *Crazy Machines* had a second relevant feature: since the game was constructed as a virtual laboratory, the player followed the

narrative of an “experiment.”

These types of games are essentially game physics simulators for the physical interactions between all game objects. They allow the player to create contraptions which could be called “virtual kinetic art” such as the example shown in figure C.6(b). The physics engines of *TIM* and *Crazy Machines* are also capable of simulating complex constructions correctly. For example, the motion analysis of a projectile trajectory in *Crazy Machines* shows the expected parabolas as illustrated in figures C.7(a) and C.7(b). However, these simulation engines remain confined to classical physics, since they are modeled after Newton’s *clockwork universe*, which guarantees repeatable and predictable outcomes. The *TIM* engine, for example, does not use a random number generator for its physics simulations to assure that the results for any given “device” representing a game level are reproducible (Wikipedia 2006b). In the authors view, if this design attribute were made more apparent, the player would have an opportunity to learn about the history of physics and the modern vs. Newtonian worldview.



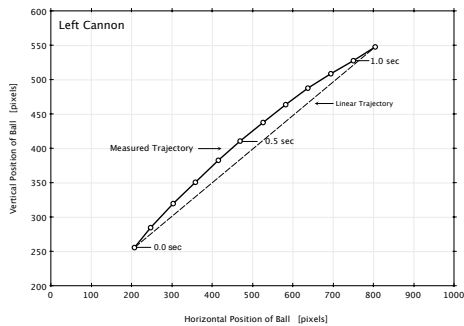
(a) Gameplay and “Crazy Professor”



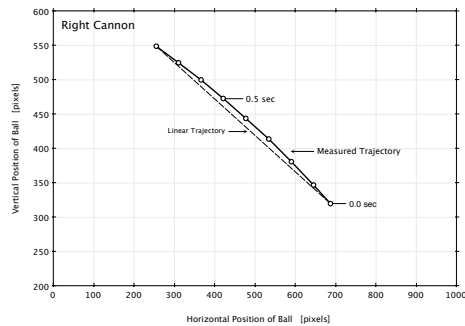
(b) *Crazy Machines 2* (2007)

**Figure C.6:** *Crazy Machines* (FACT, 2005) is based on experimental gameplay with a “Scientist” giving instructions

*Crazy Machines* still shows many instances where a departure from real



(a) 12 frames combined, left cannon



(b) 9 frames combined, right cannon

**Figure C.7:** Analysis of projectile in *Crazy Machines* shows correct parabolic trajectory

physics can be observed. First, physical coordinate systems may be mixed up when a side-view gravity is blended with a top-view billiard table on the same screen. Second, balls can have “unlimited” energy and keep bouncing forever. And third, the transfer functions of many energy-converting elements (generators, solar-panels, etc.) operate without losing energy. Interestingly, the game also uses a portrayal of a physicist as a “Crazy Professor.” This cartoonish depiction visible in figure C.6(a) as a male with fuzzy hair and a white lab coat, is a very common stereotype based on Einstein. This meme (see section ??) is also popular in other types of media and film. But such a stereotypical image of the scientist is particularly problematic because it turns kids (especially girls) off of scientific careers (TED 1998) and thereby reduces the game’s potential to serve as a science education tool. This is unfortunate, because the fundamental topic of such games does predestinate their use in an educational setting (Kendall 2007).

## C.7 Battlefield 2142

*Battlefield 2142* (DICE, 2006) was designed as a modern First-Person Shooter (FPS) computer game for multiplayer<sup>6</sup> use (Wikipedia 2010a). It is a good example of the popular genre of FPS games where the player sees the game world and other players from a realistic first-person perspective (see figure C.8(a)). Multiplayer games are often based on 3D game-engines and traditionally were aimed at endowing the games with a distinct “movie feel.” Apart from the use of game physics for character and object dynamics, this aim was achieved through the introduction of traditional story narratives, extensive use of pre-rendered video clips, cut-scenes using pre-scripted game sequences, and the inclusion of real place names and historical characters.



(a) First-person perspective



(b) Bullets “sparking” on impact

**Figure C.8:** Modern FPS games like *Battlefield2142* (DICE, 2006) make extensive use of “movie physics”

While the dynamic simulation in *Battlefield 2142* is probably quite accu-

---

<sup>6</sup>A *multiplayer game* is game which is played by several players simultaneously. The players might be independent opponents, formed into teams or be just a single team pitted against the game. Since the availability of Local Area Network (LAN) and online broadband access, many computer games include multiplayer modes or are specifically designed as multiplayer games.

rate because a high-quality custom physics engine is used (Häggström 2009), actual gameplay constitutes a departure from real-world physics, because of the extensive use of movie metaphors in the game. For example, the player has super-hero strength, sparking bullets (see figure C.8(b)) are everywhere, weapons have an unrealistic level of impact damage, and the speed of sound after explosions seems infinite. The long list of physics law violations has not only become accepted by the players, but is treated by developers as required game design. In other words, the players' expectations of similar visuals found in action, sci-fi and fantasy movies are replicated in these games (Intuitor 2006).<sup>7</sup> Perhaps one should think about divorcing such experiences from these games to advance game design; however, this change of approach would require player education and a different management of player expectations.

## C.8 FlatOut

The game *FlatOut* is a member of a game class called “driving simulators,” which are generally considered state of the art in their implementation of game physics. Here, simulated physics may include a 6 degree-of-freedom motion for the car chassis and sophisticated models for suspension, springs, dampers, anti-roll bar, roll-center, anti-pitch, camber, wheel hop, toe, tire models, and tire relaxation. Other notable driving simulation games are *Grand Prix Legends* (Papyrus Design, 1998) – because it was considered by many people to be one of the most realistic racing games ever released (Wikipedia 2008c) – and *Racer* (van Gaal, 2000), with its fully open source implementation conducted by a community of simulation enthusiasts (van Gaal & Dolphinity B.V. 2007).

*FlatOut* (Bugbear Entertainment, 2004) is a vehicle simulation implemen-

---

<sup>7</sup>The dependency of games on “movie physics” will be further explored in section ??.

tation which places emphasis on demolition derby-style races. The game is known for its extensive use of physics in relation to vehicle damage and collisions. For example, dents on vehicles may vary based on the type of accident, object and angle of impact; or falling objects may damage mostly the upper areas of a car. During play, the car will inevitably collide with many roadside items, due to the design of the racetrack and object placements in the simulated environment. Thus, the game physics simulations of the collisions create one of the main sources of fun in the game for the user.



**Figure C.9:** A vehicle hitting a tree in *FlatOut* (Bugbear, 2004) results in an unrealistic simulation of the ejected driver

Although the game physics of the driving simulation feels accurate when used by the player, inaccurate physics is immediately apparent during any head-on collision that may occur. For example, when the car collides with a tree, the accurate vehicle simulation stops and a “gory” animation of the driver being ejected from the car in a highly unrealistic way is shown (see figure C.9). A cartoonish “ragdoll” model of the driver replaces the accurate driving simulation, and movie physics elements provide “eye candy” for the players. In this case, the depiction is mostly removed from reality, creating a spectacle for the player where the expectation of “pseudo” trumps realism.



I believe that extended exposure to such simulations may potentially lead to psychological problems – a topic that should be further investigated.

## C.9 Half-Life 2

*Half-Life 2* (Valve Corp, 2004) is a good example of a FPS action game with a science fiction theme. This highly popular game has won industry awards for advances in computer animation, sound, narration, computer graphics, artificial intelligence and game physics (Wikipedia 2010*b*). It was also found that it is the game most often cited by players in connection with game physics (see section F.1).

Here, the game engine is based on a heavily modified version of the *Havok* physics engine (Havoc Inc. 2008) in order to allow extensive interactions between game objects and characters in real time. To highlight this feature, the game designers even included a special “Playground” area right at the beginning of the game, as shown in figure C.10(a). The playground allows the players to experiment with the rich possibilities created by the advanced physics engine. The player can:

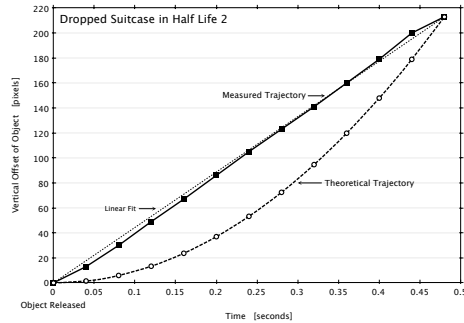
- Arrange and spin triangular objects arranged as a Tic-Tac-Toe game
- Load a teeter-totter (seesaw) with objects and see it move in a physically correct manner<sup>8</sup>
- Mount a merry-go-round and accelerate it into a frictionally slowing rotation with the click of a button
- Push a swing by running into it, resulting in a realistic-looking pendulum motion of the seat

---

<sup>8</sup>Bricks are conveniently placed beside the seesaw, so the player has immediate access to a virtual weight scale.



(a) Teeter-totter in “Playground” section of first game level



(b) Analysis of 13 combined frames; object acceleration is linear

**Figure C.10:** Advanced and simplified game physics in *Half-Life 2* (Valve, 2004)

Comments from players on the game physics were positive. Noe (2007), for example, writes, “I mess around with the cinder blocks and teeter-totter for a while. The physics in this game are great.” Some of the game physics elements in this game, however, are problematic. *Half-Life 2* is another example that uses the negative image of the “crazy scientist” (see also section C.6), and the basic premise of the narrative is about a physics experiment that has gone wrong. Although the main character (“Gordon”) in the game’s story is assisted by a scientist (simply referred to as the “Professor”), the physicist is depicted as a caricature of an actual scientist. The pre-scripted dialog sequences in the game include revealing quotes such as:

“The worst *she* might do is *couple* with your head. ... Here my pet, hop up.”<sup>9</sup>

and

---

<sup>9</sup>Quote while the professor is explaining the presence of a dog sized “pet” alien to Gordon, while pointing to his head. The contextual meaning of *couple* is “to have sexual intercourse” and thus supports the crazy scientist cliché for the player.

“Good job, Gordon, throwing that switch and all. I can see your MIT education really pays for itself.”<sup>10</sup>

A detailed analysis of object dynamics in *Half-Life 2* also reveals that some incorrect physics simulations are present, a fact which is surprising because the advanced *Havoc* physics engine is used by the game. Falling objects – in the analysis a suitcase is dropped – are animated using the physically incorrect linear acceleration algorithm discussed previously. The analysis results in a graph, shown in figure C.10(b), which shows that a linear displacement was used and not the physically correct parabolic curve Newton’s 2<sup>nd</sup> law would dictate. Experiments with the swing indicate that its hinges use a no-friction model: any object placed into pendulum motion, such as the swing’s seat, will keep moving indefinitely. These are likely deliberate simplifications of the game physics introduced by the game engine programmers as performance optimization techniques.

---

<sup>10</sup>Quote of a non-player character after a power switch was enabled by Gordon.

# Appendix **D**

## Game Categories

These are the proposed game categories for this research:

### **D.1 Action Games**

Action games are the largest class of computer games and are typically recognized as defining the genre since most original arcade games were of this type. These games place fast reflexes and coordination ability (hand-eye-coordination skill) as criteria for the player's success in playing the game. Current action games may also place high demands on the tactical reasoning skills of the player. They often feature the depiction of violent physical force, especially shooting, as their main interactive feature.

Action games can be sub-categorized into a number of distinct genres based on their theme or the technology used:

**Early Action** games frequently used a combat or space theme due to the

abstract nature of the graphics that were capable with the hardware of the time and the preference of the player in the 80s (*Asteroids* - Atari/1979, *Defender* - Williams/1980, *Galaga* - Midway/1981, *Star Wars* - Atari/1983).

**Maze** games involve some form of maze as the playing field (*The Amazing Maze Game* - Midway/1976, *Pac-Man* - Bally/Midway/1981, *Bomberman* - Nintendo/1985). Today this genre is completely absorbed as a standard component in most gameworld layouts of the “Platform” or “Shooter” genres.

**2D Platform** games (or *Platformers*) are characterized by the players character having to move around platforms and ledges which are visualized on a static or scrolling 2D gameworld (*Donkey Kong* - Nintendo/1981, *Pitfall!* - Activision/1982, *Super Mario Bros* - Nintendo/1985).

**Fixed Shooters** are a very influential early genre where the player controls a weapon with varying degrees of freedom (an adaptation of a shooting gallery) to destroy enemies on a 2D gameworld shown on a single screen. (*Space Invaders* - Midway/1978, *Centipede* - Atari/1980, *Galaga* - Midway/1981, *Area 51* - Atari/1995).

**Slide Shooters** (or *Scrolling Shooters*) are a variation of the Fixed Shooter genre which use a 2D gameworld that is larger than the screen shifting horizontally, vertically or multi-directional (*Vanguard* - Centuri/1981, *1942* - Capcom/1984, *Darius* - Taito/1986, *Raiden* - Fabtek/1990)

**Fighting** games come in two flavors: **One-on-One** fighting games use two player-chosen character which fight each other using interactively controlled moves (*Yi-Ar-Kung-Fu* - Konami/1985, *Street Fighter II* - Capcom/1991). **Beat 'em Up** fighting games (or side-scrolling fighting games) is a game genre where the player fights through a horde of

computer-controlled enemies in a series of side-scrolling stages, typically with a powerful “boss” enemy at the end (*Kung Fu Master* - DataEast/1984, *Double Dragon* - Technos/1987).

**2.5D Isometric** games are a form of 2D Platformers which present a three dimensional scene by compositing two dimensional graphics that display the world with a fixed camera orientation and without perspective. (*Congo Bongo* - Sega/1993)

**3D Platform** games are the extension of the 2D Platformers using more advanced graphics but keeping the general gameplay similar to their 2D origins (*SuperMario 64* - Nintendo/1996).

**First-Person shooters** (FPS) are characterized by an on-screen rendering that simulates the character’s point of view and are almost always centered around the act of aiming and shooting weapons (*Wolfenstein 3D* - ID Soft/1992, *Doom* - ID Soft/1994, *Quake II* - ID Soft/1996, *Unreal* - Epic/1998, *Half-Life* - Sierra/1999).

**Third-Person shooters** (TPS) employ a third person camera perspective while in many cases retaining the FPS-style character control and game characteristics (*Max Payne* - Rockstar Entertainment/2001, *Resident Evil: 4* - Capcom/Ubisoft/2005).

**Survival Horror games** are a particular sub-genre of a TPS which is defined primarily by theme rather than visual style, making it a difficult genre to classify. Typically the player has to battle opponents in claustrophobic environments in a third-person perspective, with a game-world that is using liberal amounts of horror elements and isolation themes (*Silent Hill* series - Konami/1999-2006).

**Rhythm games** are a relatively new form of music-themed coordination games. The player must match the timing of a sequence presented

by the game on input devices (buttons, mats, microphones, or musical instruments). Gameplay revolves around dancing (*Dance Dance Revolution* series - Konami/1998), instrument play (*Guitar Hero* - Harmonix/2005, *Rock Band* - Harmonix/2007) or singing (*Lips* - Microsoft/2008).

## D.2 Strategy Games

The focus in strategy games is the combination of analytical skill and tactics as the player must balance the relation between resources and various elements in the game, emphasizing cogitation rather than manipulation. Games exist on a continuum from pure skill to pure chance, and strategic games are usually found towards the skill end of the spectrum. The turn-based games defined the genre up through the 1980's due to their modest demands on processing power and evolved to feature action sequences and more character-oriented narratives. Types of strategy-games are for example labeled "god-games" or "wargames". Many simulator games are considered strategy games when their general theme is centered around the simulation of complex socio-economic systems rather than a physical gameworld (i.e. *Tycoon* or *Age of Empire* series of games).

The genre is typically divided in subtypes based on the pacing of the gameplay:

**Turn Based strategy games** proceed in phases or turns with breaks in between player moves much like traditional board games such as chess (*Defender of the Crown* - Cinemaware/1984, *Pirates* - Microprose/1987, *Civilization* - MicroProse/1991) or traditional D&D-style games<sup>1</sup> (*The*

---

<sup>1</sup>D&D refers to the *Dungeons and Dragons* fantasy role-playing game (RPG) originally designed and first published in 1974 by Tactical Studies Rules, Inc.

*Elder Scrolls* series - Bethesda Softworks/1994-2006, *Final Fantasy II* - Square Co Ltd./1988).

**Real-Time strategy games** are characterized by gameplay which proceeds continuously or in “real-time.” This leads to qualitatively different dynamics and faster gameplay making it a popular genre (*SimCity* - Maxis/1990, *Populous* - Bullfrog/1989, *Dune II* - Westwood Studios/1993, *Warcraft* - Blizzard/1994, *Age of Mythology* - Ensemble Studio/2002).

**Massively Multiplayer Online Role-playing Games** (MMORPG) have their roots in D&D style fantasy role-playing games as well, but extend it to implement a large online gameworld that is shared by many players simultaneously. (*MUD* - R.Bartle/R.Trubenshaw/1978, *Neverwinter Nights* - AOL/1991-1997, *Ultima Online* - Origin Systems/1997, *EVE* - CCP Games/2003).

The following categories, while simulators in their own right, are considered more strategy games than simulator games:

**Artillery** games are based on the simulation of projectile trajectories, where players take turns to aim and shoot their weapons at each other. Games in this category are amongst the earliest computer games developed (*Artillery-3* - M. Forman/1976, *Scorched Earth* - W. Hicken/1991, *Scorched3D* - G. Camp/2001)

**Building Simulators** involve the creation of a virtual city or building on the computer via the gameplay and might be specialized economic simulators (*SimCity* - Brotherbund/1989, *City Life* - Monte Christo/2006).

**Fictional Life Simulators** are intended to simulate characters inhabiting a fictional world. Players require a mixture of skill, chance, and strategy



to control usually narrow aspects of reality (*The Sims* - Electronics Arts/2000, *FarmVille* - Zynga/2009).

**Economic Simulators** focus on simulating an economy or business. (*Capitalism* - Interactive Magic/1995) In current games, economic simulations are seldom encountered as stand-alone genre, but are typically one of many game elements used within strategy games.

### D.3 Adventure Games

The genre of adventure games focuses on presenting the player with an interactive system for storytelling and narrative to explore. Its game principle typically imposes a high demand of logical thinking and persistence from the player. The game presents a loosely structured sequence that can be compared with parts of a movie and stops at intervals demanding the solution of tasks or riddles in order for the narrative to progress. Historically the genre evolved through several stages as the implementation of the storytelling uses increasingly complex technology:

**Text Adventure** games (1976–1984) where the gameworld is constructed from textual descriptions and the player interfaces using text entry. (*Adventure* - W. Crowther/D. Woods, 1976, *Zork* - Stanford Univ./1979).

**2D Adventure** games (1984–1993) where 2D graphics are added to the gameworld and menu or mouse based interactions are used to improve playability (*King's Quest* - Sierra/1984, *Maniac Mansion* - Lucas Arts/1987). An extension of this genre are **Media Adventure** games (1993-1997) where the graphics and narrative is further enhanced by incorporating film elements (*Gabriel Knight II-The Beast Within* - Sierra/1996))

**3D Adventure** games (1997–2010) use highly interactive 3D environments to visualize the gameworld and allow complex interactions with the game objects (*Ultima - Origin/1997*, *Everquest - Verant Interactive/1999*, *Asheron's Call - Microsoft/1999*, *Lineage: The Blood Pledge - NC Interactive/2000*).

## D.4 Simulation Games

Most simulation games attempt to convey a concrete experience and place realism as an important if not the most important goal for its game design. The player needs to master complex principles that have no direct relation to external reality to succeed in these types of games. These games can be free-form simulations, with no plot or mission system as found in other genres. Categories from the beginnings of computer games are:

**Early Space Simulation** games include *Lunar Lander* (Atari/1979) which a space-physics theme and *Battlezone* (Atari/1980) which was later called the first commercial Virtual-Reality game.

**Early Sports Simulation** games modeled existing sports as simple graphics and interactions (*Pong* - Bushnell/1972).

**Early Race Simulation** games were simple driving simulations and their key feature was a crude implementation of the graphics for first-person or third-person perspectives (*Night Driver* - Atari/1976, *Pole Position* - Namco/1982).

The more current and “traditional” categories of simulation games are:

**Flight/Space Simulators** are extensively used in the aviation industry to train pilots, but made their mark early in computer game history and

continue to be a very popular genre that provides accurate and interactive simulation of flying crafts (*Jet* - Sublogic Corporation/1985, *Microsoft Flight Simulator 2000* - Microsoft/2000). Space simulations are an extension of flight simulators into space, but should not be mixed up with space-themed strategy or action games (*Microsoft Space Simulator* - Microsoft Game Studios/1994, *Space Combat* - Laminar Research/2004).

**Vehicle Simulators** attempt to more or less accurately (ie. less in arcade-style simulators) simulate vehicle or race driving involving sometimes some of the most accurate physics simulations of all games genres (*REVS* - G. Crammond/1986, *NASCAR* - Papyrus/1994, *GT Legends* - Atari/2005).

**Boat/Submarine Simulators** are games where players command a submarine or controll a sail-boat (*Submarine Commander* - Thorn EMI/1982, *GATO* - Spectrum Holobyte/1985, *Silent Hunter* - SSI/1996).

**Sports Simulators** typically emphasize playing the sport (such as the *Madden NFL* series - Electronic Arts/1984-today), while others simulate the strategy behind the sport (such as *Championship Manager* - Dornmark/1992) or satirize the sport for comic effects (such as *Arch Rivals* - Midway/1989). Almost all sport categories are covered by computer simulations including individual and team sports (*Track & Field* - Konami/1983, *Sensible Soccer* - Renegade Software/1992, *Tiger Woods PGA Golf* series - Electronics Arts/1998-2006).

## D.5 Puzzle Games

The traditional game of solving puzzles can be found as well in many computer games. Computer driven puzzle games can be highly unique but also

very frustrating to the player, since the machine can usually solve the puzzle a player might work on for hours in mere milliseconds. Many real puzzle games such as jigsaw puzzles and the Rubik's Cube can be presented digitally, but are not considered typical computer game genres.

One can identify the following computer-specific puzzle games genres based on the challenges created by the game design:

**Visual Matching Puzzles** use player controlled blocks or elements to create patterns which score points or advance the game. An example for this genre is “the greatest video games of all time” (*Tetris* - A. Pazhitnov/1985) and its many variations.

**Hidden Object Puzzles** involve the interactions of the player with a play-field to deduce locations of otherwise invisible objects. An example for this genre is “the most time wasting game of all time” (*Minesweeper* - R. Donner/1989) which is included in most versions of the Windows operating system.

**Character Control Puzzles** involve controlling game characters using a set of commands and executing them in an efficient way to achieve the game goal (*Lemmings* - Psygnosis/1991, *Oddworld* series - GT Interactive/Microsoft/EA/1997-2005).

**Construction Puzzles** involve typically the creation of a series of Rube Goldberg devices - arrangements of a collection of objects in a needlessly complex fashion so as to perform some simple task (*The Incredible Machine* series - Dynamix/EA/1993-2001, *Crazy Machines* series - PepperGames/2002-2005).

## D.6 Educational Games

When a game is designed to teach or train during gameplay, the game is called an *edutainment* game due to the combination of education and entertainment “in one package”. In edutainment games, the primary design focus is on the teaching part and game content is usually well-researched, designed around teaching principles or based on an actual curriculum.

Educational games make use of the whole variety of general game genres and teaching subjects making it hard to categorize specifically:

**Child Education** games use memory, drill, puzzle and logic elements in the gameplay and target elementary and secondary level children through a variety of game styles and typically ample amounts of multimedia content (*Carmen Sandiego* series - Broderbund/TLC/1983-2004, *Zoombinis* - The Learning Company/1997, *Urban Jungle* - Gov. of Croatia/2005).

**Serious Games** (SG) is a subcategory of educational games which focus on an audience outside of primary or secondary education. The SG genre might include games with a marketing or advertising goal (*America’s Army* - US Army/2002-2005, *ReMission* - HopeLab/2005).

**Programming** games revolve around the task of writing a program in a domain-specific programming language in order to control the actions of the game elements or characters (*Core War* - D.G. Jones/A.K. Dewdney/1984, *Robot Battle* - GarageGames/2002).

# Appendix **E**

## Interviews

### E.1 Alen Ladavac

Alen Ladavac is the lead programmer at Croteam ([www.croteam.com](http://www.croteam.com)), developers of Serious Sam franchise (see [www.seriously.com](http://www.seriously.com)) - a very popular First-Person Shooter style game-series first released in 2001. <sup>1</sup>

The Interview with Alen Ladavac (AL) was conducted via e-mail between 21 July - 24 Aug 2006 by Andreas Schiffler (AS).

**AS:** Before I start to get into the more specific details, I'd like to know a bit more about your background as programmer and game developer. What did you study and where did you get your initial experiences as game developer?

**AL:** I started programming early in the primary school (which is 8 year school here, 7-14yr), on ZX Spectrum and some other now long forgotten machines. Then I went to a mathematics-informatics secondary school (which is 4 years, something like a gymnasium, 15-18yr), and at that time I programmed on the Amiga and involved in the demo-making community. During that time, I had several part-time jobs making custom small-business software. Finally, I studied CS and the Zagreb University. During the time at the univer-

---

<sup>1</sup>References: croteam (2003) Boker (2005) ?

sity, I joined my current company, Croteam, and worked on Football Glory, our first game which was published during my second year. I continued working on the first Serious Sam game while still at the university, and the game was published about a year after I got my BA. In general I always considered my formal education in CS as more of a “form” than of actual practical use, as I always had to be much ahead of the formal program, due to what I was doing in my spare time. So most of my programmer knowledge is self-taught and originates from experience, books and Internet resources. Not to say that the formal CS training was all useless, but it was generally sub-par. I contrast this with e.g. mathematics and physics classes, which were really useful, as I would have a hard time learning all that myself.

**AS:** Since my questions revolve around physics in general - not just game physics - I'd like to dig a bit in that area. What was the most interesting aspect of your physics education that you remember?

**AL:** I got most of my physics education in the secondary school. We learned the basics of all necessary fields: Newtonian dynamics/kinematics, optics, thermodynamics, electrics, etc. But most important, the teachers were very concentrated on getting us to “think as a physicist”. I really draw a lot of experience and “the feel” from there.

**AS:** What about today's physics research - anything on the radar that you are really interested in outside of your field of work?

**AL:** As programmers, I guess we're all interested in “the geek stuff” and any physics news always sound fun, but I can't say that there's anything I'm particularly following right now. The problems we deal with in games (i.e. simulations) are not investigated much in the academia circles, and where they are, that's usually mostly disconnected from “our reality”, which is making it plausible and fast, whilst most other branches strive for solutions that are correct and just generally implementable (with less concern on speed).

**AS:** You have been involved with the gaming industry for many years now. In very general terms, what do you think of today's games and game platforms?

**AL:** They are amazing. The hardware is beyond what was even imaginable 10-15 years ago, and we are able to do some really rewarding implementations. Yet, there is so much to do, and every step we make in getting towards more realistic rendering, sound, physics, or AI, just uncovers ten more things left to do. It gives a feeling that there will always be something new to work with, and it is just great. The only disappointment is with the so called “next-gen” platforms which apparently are turning up as “previous-gen”, much like most developers feared when the early specs were out. It is a bad feeling to have the whole public (and especially your publisher) expecting you to deliver something which will not

be possible, simply because the hardware vendors are doing something that's borderline with false advertising.

**AS:** How do you think mathematics, engineering and physics is perceived today by the general public? Is there a need to educate more in this area? If so, why?

**AL:** Mostly like "rocket science", and it's a shame. In general, I feel that people should strive more to understand the things that they are dealing with, especially in this age of technology, not to expect everything to be served to them and to be "automatic". I think that people would be better off understanding at least the underlying basics of the "products" that they use daily. Sadly, the current system(s) in our world seem to turn in the opposite direction. But I'm not sure how much could be done to improve this, because the problem is not in education, but in changing the overall stance about the "products", from "things that do stuff for me" into "things that I use to do stuff". But perhaps, I'm quite off the track here, so I'll just move on to the other topics...

**AS:** Now, I'd like to get a bit more technical and also ask some questions related to the "Serious Sam" (SS) games series and engine where the second version features extensive use of physics as a game design element.

**AS:** How did the initial idea for SS evolve? What did you like about existing FPS games and engine implementations at the time you started SS1 and what was missing?

**AL:** The idea itself really was a product of an evolution in sick ideas and a lot of crazy turns. We just wanted to make a game that is crazy and that's not holding back. What was a bit against the trends in seriousness at that time. (Hence the crazy name as well.) We drew the idea from the Doom playability, which was missing at that time. Other than that, we wanted to make it have large open spaces and to make it bright and sunny. Again, both in contrast with the trends. For those we definitely needed an engine.

**AS:** What about the heavy development effort of SS2 - why didn't you just use Havok and save some time?

**AL:** Perhaps hard to believe... first of all, we considered it shamelessly expensive. Secondly, and it was confirmed in discussions I later had with some other people in the industry, the sheer effort of supporting Havok or similar is comparable to what we did with "rolling our own". It maybe was tight, but I believe firmly that now a) the amount of experience we now have in dealing with physics problems is priceless, b) the integration is very tight and we don't have any overheads, c) as mentioned above, the effort seems to be comparable. I believe that most of the effect of (c) is actually because of (a) - if you haven't wrote one, you don't know what the problems are. Many graphics programmers today believe that, even though we are using HW for everything, you must have written



a SW renderer at least once, otherwise you're not aware of what the HW is doing. Heck, I believe that my eyes opened a lot when I wrote a C compiler. (Basic one though, but still an eye opener.) So - it takes to write one to know one.

**AS:** As you mentioned in your email, "in (SS) V2, we've added a full physics simulator". Could you point out a few of the highlights and comment on how easy or difficult it was to implement? What is fake (or left out for performance reasons)?

**AL:** Highlights, well... Getting there was not trivial. But looking at the currently functional system, it really is astoundingly simple. Once we broke through with the actual iterative solver that we use now the world just started to look much simpler. The most complicated thing still remains collision, especially with meshes. Now fakes, there's numerous. Perhaps the worst is that the collision is not continuous, but step-wise, and that there are special anti-tunneling checks happening when needed.

We are always concentrated on plausibility instead of reality, so that sometimes I'm amazed that some of the things we do even work at all. The way that we do it with time steps of 0.1 sec and sub-stepping of 10x being one of the most amazing. We run collisions at the display frame rate, limited to at least 10 fps. This is however not enough for the solver so we run the solver at at least 100fps, but update the positions in the contact constraints. It sounds crazy, but it works.

**AS:** Do you have any stories, examples or even some source code that illustrates how difficult it can be to implement a proper physics simulation?

**AL:** Again, the physics seems rather straightforward, once the iterative solver is done. Its only the collisions that are the problem. Doing a cylinder-cylinder separating axis collider and then generating contact points is a real brain-breaker. Not because its such a high math, but because it requires so many little details in just the right spots. Collisions between primitives and meshes are similarly problematic. It is not a lot of code, but you have to know exactly what you are doing.

**AS:** You also mentioned, that game physics is not as complicated as some people make it sound like, and that the biggest problem in making a physics engine is the plain lack of available information. So how did you get started and how much was trial-and-error? What information would you like to see in the future for your development staff?

**AL:** We did get some info from available books and papers, and were able to ask around for some pointers, but we had to discover a lot of it. The worst problems are with the academic papers of physical simulation. There are some that really contain some very quality info, but (a) usually the authors are not concerned with interactivity, or at least not at the desired scale and (b) there's a lot of "math-speak" there. Not that I don't

know how to read the math speak, or that I wouldn't know what a differential equation is, but the fact is that when someone writes a useful principle in a formal paper with all the fancy theorems and axioms, it is just so hard to wrap your brain around it. It is a kind of ridiculous to apply math writing as a "compression algorithm" to make the paper fit in less pages. I much more enjoy reading papers written by industry programmers, because they always accompany their formulas with real text explaining what they meant, and why is that important.

I would like to have some time to do a write up on how to implement a physics simulator in a simple way, but I don't know when I'll have the time.

**AS:** In an interview you said about the SS2 engine that "Designers and artists can get familiar with it very fast and start producing game content in a matter of days." Do you think an engine that is that flexible and easy to use could be used outside of traditional game development circles (i.e by media artists might make interactive projects with it, scientists and educators use it to visualize data or concepts)?

**AL:** Yes, we believe so in our company, and we have been investing into making it open in that direction. But we are still not done with wrapping it up so that we can actually market it to people outside of the games industry. Though we are open towards people that ask around for it, and the editors are available publicly.

**AS:** Let me point your attention briefly to a recent development in the gaming industry to incorporate physics acceleration into games (i.e. AGEIA, shader programs).

**AS:** What do you think of the current crop of "physics-accelerated" games? Do you think these devices or drivers help the game designers in creating better games? Why (or Why not)?

**AL:** Physics acceleration should not be compared directly to rendering acceleration. That is apples to oranges. This is where AGEIA is wrong (IMO), and they are pulling people's leg, against the realistic facts. One interesting info is that, on a decent CPU, their poster-child (CellFactor) seems to run faster without an accelerator than with it (there was a cheat on some popular site, explaining how to disable acceleration while still retaining all same functionality.)

Anyway, why I believe it is so is: 1) Physics is a full-loop feedback process. Result of this frame of physics is needed for the next frame of AI and physics to be processed. Current gfx cards exploit heavily the fact that they can buffer and pipeline freely. I know, because I do a lot of gfx programming and system design, that the gfx card vendors are very sensitive every time we try to ask for any kind of feedback from the card. Even if PCIE was available for this and even if PCIE had the same downstream speed as it has upstream

(what is not the case ATM, AFAIK), still the hardware on the card cannot be compared to the gfx hardware, because the gfx hardware pipelines a lot internally. 2) Phasing-in is going to be difficult, much more difficult than rendering. Physics are different because, unlike in rendering, you have to yield the exact same result with and without acceleration. So you can't make anything useful in the gameplay rely on the accelerator. And the place where more detailed physics is needed is the gameplay, not decorative effects like explosions (for such effects, simple fakes are enough). So, people will not buy cards unless there are games that make a difference (in gameplay) with and without accelerator. Still, there is no sense in making your gameplay rely on it unless 99% customers already has the card. A vicious circle. 3) **AS:** There's a hot concept called "stealth education" that says it's possible for people to play a game because they enjoy it and "accidentally" learn stuff along the way (i.e. Teaching not as obvious or targeted as in "edutainment"). Unfortunately, the concept has yet to prove its worth with the pre-teen and teen community. So, on this concept there are a few questions:

**AS:** Certainly SS had no educational agenda - it was purely entertainment (player quote: "I loved the mindlessness of SS")? Are there any commercial game titles available today that you are aware of, that implement this concept - especially related to the sciences?

**AL:** None that I'm aware of.

**AS:** How would you judge the commercial potential for "stealth educational" games for the gaming industry of today? Would such a game sell compared to other less "educating" games such as SS?

**AL:** People want to relax and have fun, not to be educated (stealthily). At least that's how I feel. Perhaps I may be wrong, I didn't ponder on this deeply. Anyway, if there's commercial interest by some companies or governments for having such games, then there's commercial interest in developers doing them. You probably know of "the serious games" concept (nothing to do with Serious Sam). That is similar, though it's not stealth.

**AS:** As an historical example of a game with some teaching potential, the early Atari game "Lunar Lander" comes to mind. Would you consider such a game "educational"? If yes, in which way? If no, why not?

**AL:** I'm not familiar with that game. Sorry.

**AS:** Do you have any examples of games you have seen, that try, but fail in educating? What are the reasons for the failure?

**AL:** I've seen i.e. a game about parking a car (can be found on the Peugeot site, IIRC). It is quite fun, and I had loads of good time playing that (a simple flash game). But it's not connectible with actual parking of a car, because in this game you park from the top

perspective. This particular one was primarily an advertising game, but I would guess that this would be the easiest way to get caught in a trap trying to make an educational game - most games have to abstract some concepts and translate them into their own worlds, in order to be fun. In the process, it is easy to lose the actual most-important moment from the real world. Another example - any war game or shooter. In the real life I guess you'd get so scared of being killed that most of the tactics that people employ when they know they're gonna respawn soon - just don't quite cut as reasonable. Sorry I can't be of more help here, but I don't know of much educational games.

**AS:** Now, I'd like to broaden the topic to some more philosophical ideas in game design.

Game developers today recognize the importance of proper physics simulations for interactivity when creating an immersive virtual world. How would you rate the importance of game physics versus graphics and sound, social constructs and story line, or other gaming elements for creating this "immersion"?

**AL:** First thing to always have in mind about game design is that you're not immersed in a simulation of a real world, but in the game's world. So, as long as it is consistent - everything is fine. Humans are quick to adapt to arbitrary "physics laws", as long as they are consistent. So, each game chooses which of those concepts are more important. There are games where social interaction is most important (for example [World of Warcraft] WoW), those where graphics are most important (mostly leading-edge [First Person Shooter] FPS games like Doom3, Unreal series, ...), or where physics is most important (usually simulations, driving games, etc.). Usually, one has to give away some of those to be able to push forward in the feature that's most important. For us, it was graphics, then physics and lastly social interactions. Personally, I'd like to work on a game that had just exactly the opposite order, because interactions is where most juice hasn't been squeezed yet, so to say. But that's not likely to happen, probably. :)

**AS:** Assuming we had a game that implements some basic laws of physics for its gameplay - say a submarine shooter. What would you think about a "reality" slider for the game that would allow smooth control from "arcade physics" to "real physics" similar to adjusting the sound volume? Would players use it and would it teach them something about the world?

**AL:** It is usually available in some simulators, where it makes sense for beginners to disable some hard to handle effects. In shooters, it would risk to make them not fun. People would definitely use it - they use whatever dials you can provide. But the question is would it be possible to tune it so that its playable on all the positions of the slider, and would it be too much of an effort to implement, to justify the results.

**AS:** Spatiality is the main characteristics of most computer games in that they are mainly concerned with the representation and negotiation of space. Do you agree with this assertion? Could this preoccupation of spatiality in games (especially 3D games) be an important reason, why only a limited number of physics-fields (mostly dynamics and optics) are used in modern computer games?

**AL:** Most of them are not, IMO. Space is most usually used as something that provides the use of time, because of limited velocity. I'd say that time is more important than space in most games. But that's very philosophical. :) And, gamers are ordinary people and as such, dynamics, optics and acoustics are the only fields they are interested in. Most people don't even understand the very basics of thermodynamics besides just cooking a coffee. :) Which other fields did you have in mind? Electro-magnetism?

**AS:** One of the weakness of computers that is specifically apparent for games in their limited I/O capability - usually relegated to keyboard and mouse or a gamepad for input. Do you think that the design of the physical interfaces might be an interesting advance for computer games from an educational point of view?

**AL:** Yes. If people were able to move in the game by interfaces that were not so abstracted, then the experience would be much more realistic, and the immersion would have less of the "suspension of disbelief".

**AS:** To make games "better" and more innovative, do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**AL:** Game artists need more education because of the games being more realistic, which requires them to understand what is going on. I'm having problems with artists for example believing that if they increase a mass on an object, it's apparent bounciness is going to decrease. Explaining to them that we cannot simulate the floor crushing below a large rock directly, and that they have to decrease the bounciness manually to emulate that, leads to a lot of confused looks. This also applies to the way light interacts with surfaces, but that is not a part of this topic, so I'll skip on that.

**AS:** It is fair to say that computer games ARE very influential in our society (i.e. in terms of money, the game industry is bigger than film and music together). Is there an effect on society? Is that a positive or negative effect? Are games in their effect on society similar to - say - movies and can they distort the users view on reality?

**AL:** I believe that the society is much too resistant to have something like games change it in a significant way. It is more like the water - it seems like it gives in, but it always retains the same volume. People that now spend hours daily on [World of Warcraft] WoW

would otherwise do it on IRC, or in absence of computers would go to a pub and get together. It's just different means to fulfill the same human needs.

**AS:** To finish off, I'd like to ask you a few questions that came up during my first session of our PhD work group - I just wanted to hear how you would answer them (if you have anything to comment on these at all).

How might games be used in scientific research? Do you have any examples?

**AL:** They can be used in simulations of crowd behavior and research of social interactions. I believe that I heard somewhere that Sims were used for social simulation already. (Don't have any solid reference on this.)

**AS:** Folk Physics is best suited to cope with the kinds of objects and events that we encounter on a day-to-day basis and is not really designed to provide accurate descriptions or explanations of the universe. Do you think it is possible that game physics will "teach" some form of folk physics to the players? Could this have any negative side-effects?

**AL:** Not sure if this is what you wanted to hear here, but... for example in Sam series, the gravity is  $30\text{m/s}^2$ . This is because the character is required to be able to jump 2m up, and if the  $g$  was 9.81, it would be plain boring to wait for a jump to finish. People adapt quickly, and never notice the difference. I didn't see anyone having problems with jumping in real life, after playing Sam. It over-exaggerated, but I guess that just proves the point.

**AS:** What is the advantage/difference of a physic games compared to the real physic experience (for example in an experiment)?

**AL:** Perhaps the overhead is lower in a physics game, if you want to setup a complicated experiment. Also, you don't want to throw people down stairs in real life, to see how they bounce.

**AS:** Would it be more interesting for a game developer to use physics laws to create new games in which specific laws would have the main influence on the game? Or is it more interesting to improve existing games by incorporating exact physics phenomena into the game logic (example: a modified Tetris game)?

**AL:** The first choice. Because the games that were already made were tuned to that exact physics that they used, and would probably not be fun otherwise. Even in new games, we always still bend the laws to create better playability.

**AS:** Thank you for this interview.

## E.2 Chris Crawford

Chris Crawford earned a Master of Science degree in Physics from the University of Missouri in 1975 but soon entered the world of game development. He is a longtime proponent of thoughtful and experimental game design and left his mark on the game industry with several classic game titles written for Atari in the 1980s. More importantly and long lived though, were his publications “The Art of Computer Game Design” (1982) and the series “The Journal of Computer Game Design” (1987-1993). His books, now recognized as classics in the field, discuss what one can learn from the history of game play, the necessity of challenge in game play, applying dimensions of conflict, understanding low and high interactivity designs, watching for the inclusion of creativity, and understanding the importance of storytelling. It is this last point which currently occupies Chris most of his time while working on his latest project “Storytron” in a effort to create a truly interactive storytelling experience. <sup>2</sup>

The interview with Chris Crawford (CC) was conducted by e-mail from 2 April - 22 April 2006 by Andreas Schiffler (AS).

**AS:** Since you are a few years older than myself, let me try to get some insight into your background with physics, gaming and game physics from as a “historical” perspective.

You have a Masters degree in Physics. In what field did you work in during these years and what relevance had computers and programming for your studies if any?

**CC:** My thesis was a catalog of dynamical parallaxes of visual binary star systems. A dynamical parallax requires an interactive solution to set of equations, so it had to be done on a computer. But I went further and attempted a statistical analysis of the data in an attempt to find a component of galactic angular momentum in the angular momenta of the visual binaries. This was really hairy statistical computing. My analysis showed no such component. Although this work had no direct relevance to my later work with games, it did provide me with important analytical skills that proved to be valuable in my work with games.

---

<sup>2</sup>References: Hague (2002*a*) Chris (n.d.) Crawford (n.d.)

**AS:** The hardware during the beginnings of game design when you worked for Atari was very, very limited compared to today's standards. Did any of the early games you made for the Atari 800 have an algorithm that could be remotely called "physics" in it?

**CC:** Sure. My first game, Wizard, which was never published, had some algorithms that were derived from my experience in physics. My next two games, Energy Czar and Scram, were both physical simulations of real-world situations. My next game, Eastern Front (1941), had some algorithms in the AI that were derived from physics concepts. Indeed, many of my games had something in them that was informed by my knowledge of physics.

**AS:** Did you like studying physics or was it a rather dull subject and not what you expected? What was the most interesting aspect of your physics years that you remember? What about today's research - anything on your radar there?

**CC:** No, I enjoyed physics immensely. I didn't like all the math getting in the way, but that's the way physics is. I have not kept up with the current state of physics.

**AS:** Some of the earliest computer games used simulations as one of the main game principles - for example "Lunar Lander". What did you think of LL? Since you might be more familiar with some of the games from that era, can you point to one or more similar games?

**CC:** These were very simple games, but they were our starting point. Three other classics from that era were "Hunt the Wumpus", a kind of simple guessing game, "Hammurabi, a simple simulation with three linked differential equations, and "Star Trek", a turn-based game that grew quite complicated over the years.

**AS:** The previously mentioned game Lunar Lander brought NASA style simulations into the mainstream of the arcades. Do you think there was any creative connection between the "hard-sciences" of academia or the military labs and early game design?

**CC:** Not much. The stuff they were doing was immensely more complicated than that stuff being done on microcomputers. During the 70s, most people using big computers regarded microcomputers as toys.

**AS:** Since the Atari times, computer hardware has progressed rapidly and each generation of games has more CPU power available. Has any of your games since these early years had some physics simulation component that took advantage of these readily available clock-cycles? If so, how good was it and what did it do? If not, why not?

**CC:** Over the years I have adjusted my software designs to take advantage of the more powerful hardware. This has taken primarily the form of using more memory to get more advanced data structures. I also do much more complicated algorithms.

**AS:** Now, I'd like to ask a few more general questions related to comments you made on



game design and the game industry in your past publications.

You made the assertion in “The Art of Computer Game Design” that one of the fundamental motivations for all game-play is to learn. Do you think this is still a valid claim?

**CC:** Yes, I do. Deep down, the human being is driven to learn. That's why we take on new hobbies, visit new places, reach out to new people. We're always trying to satisfy an insatiable curiosity. Playing games is only part of this drive.

**AS:** In an interview posted on the net, you mentioned two games that were “impressive” to you: Hidden Agenda and SimCity. Is it coincidence that your favorite games seem to be ones with strong leaning elements? What do you think about today's games as learning tools? Or should we just keep to GoogleEarth and Wikipedia and forget about mainstream games?

**CC:** What some people call “bias” I call “point of view”. A good game should present a point of view. The best teachers are almost always passionate in their beliefs, and that passion comes across.

**AS:** Imagine an entertaining “Asteroids” clone that uses real physics - designed with a NASA scientist as part of the development team to ensure scientific accuracy. Could this work as a game and actually teach something? From a game design standpoint, what might be the biggest obstacle to overcome for this imaginary game during its development?

**CC:** Good teachers never teach reality; they teach a carefully chosen subset of reality in order to make it clear to the student. In our designs, we should seek not realism but clarity - and what is clear to one level student will be confusing to a lower level student.

**AS:** One of the key elements for computer games is their “responsiveness” in a sense that a computer game can be parametrized ad-infinitum according to the player's wishes - a feature that makes a computer game quite distinct from board- or card-games. Has this “advantage” of computer games been used well over the years?

**CC:** Yes, I think so. I don't see any need to press this feature any harder. Of course, in terms of the nature of the challenges we offer players, we haven't even scratched the surface.

**AS:** Assuming we had a game that implements some basic laws of physics for its game play - say a submarine shooter. What would you think about a “reality” slider for the game that would allow smooth control from “arcade physics” to “real physics” similar to adjusting the sound volume? Would players use it and would it teach them something?

**CC:** Interesting concept, but I don't think that increasing reality is a one-dimensional concept. For example, you might want an adjustable coefficient for the elasticity or inelasticity of collisions - but then what about another adjustable coefficient for friction? You'd

probably need one adjustment for every major physical effect at work.

**AS:** Spatiality is the main characteristics of most computer games in that they are mainly concerned with the representation and negotiation of space. Do you agree with this assertion? Could this be the reason for the limited number of physics areas (mostly dynamics and optics) used in modern computer games?

**CC:** Yes, I agree that spatiality is an obsession with game designers, but I don't see any conflict with physical simulation arising from this.

**AS:** You mention a weakness of computers that is specifically apparent for games in their limited I/O capability. Do you think that the design of the physical interfaces might be an interesting advance for computer games?

**CC:** I don't think that we need grand new input devices. The most important new device we need is speech recognition.

**AS:** You said that "artistic maturation will be the dynamo that drives the computer games industry". What do you think has become of this dynamo - thumbs up or down? Do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**CC:** Thumbs down. Game designers are artistically no further along than they were in 1986. I would rather see the computer science people learn more art stuff than the other way around, but I think that the arts people are doing a fairly good job of learning the technology of the computer.

**AS:** You predicted in 1982 that computer games will be very influential on society as a whole. It is fair to say that computer games ARE very influential (i.e. in terms of money, the game industry is bigger than film and music together), but what do you think is their effect on society? Are games in their effect on society similar to - say - movies?

**CC:** Nowhere near as much. Games are still much too limited in their expressive power. You can't really say anything interesting with games, so there's little reason for them to have much redeeming social value.

**AS:** Let me point your attention quickly to the current (game) events front and ask you a few questions about what you think of recent developments in the gaming industry.

Since 3Dfx came with their first SGI-for-the-Masses Voodoo1 card in 1996, the graphics card makers have been in a hard-pitched battle for supremacy on the pixel front. What's your take on this and how has it helped (or not helped) games and game design?

**CC:** Overall, it has been detrimental, because game developers have concentrated all their efforts on better graphics, and ignored the more fundamental design problems facing

the industry.

**AS:** I'm not sure if you see or heard anything about the new Ageia physics accelerator cards yet - if not, head over to this link <http://physx.ageia.com/> and have a look. What do you think of these "accelerated" games? Is there any worthwhile physics in it - any learning potential? Do you think these devices help the game designers?

**CC:** Sorry, I don't have time to study this.

**AS:** Have you heard of ARGs - [http://en.wikipedia.org/wiki/Alternate\\_Reality\\_Game](http://en.wikipedia.org/wiki/Alternate_Reality_Game)? What do you think of these as a computer game designer? What would you say, if such a game were to be extended by real physics elements (i.e. some simple astronomy)?

**CC:** Again, I am uncomfortable with the idea of impressing reality too much into the fantasy world of games. All drama distorts physical reality to foster dramatic reality. There's nothing wrong with this; in order to make a point clearly, you must distort reality. Reality is far too complex for any human expression to capture in its entirety.

**AS:** To finish off, I'd like to ask you a few questions that came up during my first session of our PhD workgroup - I just wanted to hear how you would answer them (if you have anything to comment on them).

How might games be used in scientific research? Do you have any examples?

**CC:** Games are primarily educational devices. However, they could be used in scientific research for a number of purposes: human sensory-motor performance, for example cognitive function experimentation. There have already been some fairly good experiments with exploring the difference between Homo Economicus and Homo Sapiens. There are also some linguistics simulations on language transformation that might better be served by games.

**AS:** Folk Physics is best suited to cope with the kinds of objects and events that we encounter on a day-to-day basis and is not really designed to provide accurate descriptions or explanations of the universe. Do you think game physics can "teach" some form of folk physics to the player? Could this have any negative side-effects?

**CC:** Sure, they already do. And yes, all learning has negative side effects, because all expression presents a subset of reality, and that subset is necessarily misleading in what it leaves out. We must accept that learning is a process of convergent approximation that never even approaches the asymptote of universal understanding.

**AS:** What is the advantage/difference of a physics game instead of real physics experience?

**CC:** The painting of Mona Lisa does not communicate the reality of the woman. It presents a single artist's point of view for a single snapshot of time. The real woman is

much more complex. But the painting communicates truths that would not be apparent in a simple photograph. So which is more accurate: the painting or the photograph?

**AS:** Is it more interesting to you to use physics laws to create new games in which specific laws would have the main influence on the game, or is it more interesting to improve the existing games with the exact physics phenomena?

**CC:** I would prefer to build a game in which the laws of physics are communicated through their violation. For example, how about a game in which  $h = 1e-3?$  or  $c = 3e2?$  or  $G = 1e4?$  Now THOSE would be interesting games! People would be fuzzy blobs. A single photon could hit as hard as a bullet. And if two people got too close together, they'd smash each other to death! That's FUN!

**AS:** Thank you for the interview!

## E.3 Danny Kodicek

Danny Kodicek is a graduate of Cambridge University, now working as a freelance programmer and multimedia developer specializing in Web sites and educational software for making science simulations. He wrote the book "Mathematics and Physics for Programmers" an foundation book that explains mathematics and physics needed for game development. <sup>3</sup>

The interview with Danny Kodicek (DK) was conducted by e-mail from 17 Sept - 22 Sept 2006 by Andreas Schiffler (AS).

**AS:** Before I start to get into the more specific details, I'd like to know a bit more about your background as programmer and game developer leading up to your writing of the book Mathematics and Physics for Programmers. What did you study and where did you get your work experiences?

**DK:** I studied Maths at Cambridge, but I never thought I'd use it - at the time I was hoping to get into films, but after leaving I started trying to get somewhere as a writer. I wrote a childrens book but didnt manage to sell it. Then my mother, who had been teaching on a screen writing course, brought me in on a project she was doing with someone she met there that worked in multimedia, which was a project about music for young children. My mother knew lots about music but nothing about computers; the

---

<sup>3</sup>References: Charles River Media (2005) Powell's Books (2005) Director Online (2007)

other person knew about computers but nothing about maths. As someone that knew something of both, I was there to bridge the gap! I began on the project as a writer and interactive designer, but started to get interested in Director as I saw it in action. Pretty soon afterwards I was working as a programmer. Interestingly, I discovered that I really missed doing maths, and I fairly quickly established myself as a maths-head on various forums, which helped get me known while I was still fairly inexperienced as a programmer.

**AS:** What was the most interesting aspect of physics that you remember? What about today's physics research - anything on your radar that you are really interested in outside of your field of work?

**DK:** I was always more into the abstract than the practical, both at school and university. Physics was always just a way to keep score. But I do remember getting very interested in thermodynamics. Right now I'm particularly interested in the work of Ian Stewart, a mathematician who studies biology, particularly the constraints of physics on biological development, and the way that evolution takes advantage of readily available physical phenomena, particularly symmetry-breaking.

**AS:** Are you involved in the computer gaming industry in some form today? In very general terms, what do you think of today's games and game platforms?

**DK:** I'm not involved in gaming right now, no. I'm making educational science simulations for secondary school students - recently I made a Forces and Motion simulation and another on Motors and Generators (although right now I'm tied up in a text-based project without much of a science aspect to it). I love games of all kinds, although as a puzzle fan I tend to go for the intellectual challenge rather than the twitch. What I look for most in a game is character and story, so I like games like Grim Fandango and one I'm currently revisiting with my kids, Abes Odyssee. However, I'm not particularly up to date - I mostly prefer to play games that are a few years old, that have withstood the test of time. And as a married man I don't get as much game time as I used to! What I'm generally seeing in the game world is an increased emphasis on realism, and a lot of stratification into standard genres (although this has always been the case, really). Very few games have really caught my eye for a while; only a few titles really stand out as mould-breaking (Grand Theft Auto sprung into my mind just now for its large free-roaming world and mission structure)

**AS:** You mention in an interview - referring to elastic simulation in Director - "From my point of view, though, it's just a toy, really, as with many things that I do in my spare time. Director's great for making toys." Would you see the incorporation of better physics as a "innovation" or more as a "refinement" for games? When do you think game physics had an impact in the gaming industry and with what title?

**DK:** Im not a big fan of realism in games. I can get realism outside. What matters is gameplay, and whether this is enhanced by realistic physics is generally doubtful. In my book Im pretty disparaging of the trend towards ragdoll-physics (great, you can make bodies get blown up realistically), and I havent seen much to change my mind. Some smaller games do use physics in an interesting way - for example, theres the little Flash game Spaced Penguin (and a number of sequels and imitators) where you have to use gravitational fields to guide a penguin through space to a waiting ship; and another one doing much the same with magnets. And of course theres probably the best all-out physics game, Worms Armageddon. But in the mainstream game world its not particularly important except in the simulation genre (sports and flight sims, and to some extent driving games) Another very interesting example of a toy game is Neal Stephensons quantum football game, where you have to manoeuvre a particle into a goal by manipulating its probability function. Now thats really abstract physics

**AS:** How do you think mathematics, engineering and physics is perceived today by the general public? Is there a need to educate more in this area? If so, why?

**DK:** Naturally, I think theres always a need to get more people interested in maths. Maths underlies absolutely everything thats interesting about the world. But I think its always taught very badly. The over-emphasis on numbers (which are almost irrelevant in real mathematics) rather than geometry I think is a great pity. Numbers are hard to understand and of limited value, whereas geometry is all around us and very easy to grasp. Get the kids doing topology, thats my opinion. Physics and engineering are more specialized fields, and Im not quite so convinced they are vital for every educated person. Of course, everyone should know how electricity works, how its generated and why this leads to global warming. Everyone should know about light waves. Everyone should know about the structure of the solar system. And again, I think these things are taught poorly (why is Energy not the very first and most basic thing taught in physics?). But theyre not as fundamental to life as the understanding of general patterns, which is what maths is. As for engineering - does the person in the street need to know why a bridge stays up? Im not sure.

**AS:** Now, I'd like to get a bit more technical and also ask some questions related to your book Mathematics and Physics for Programmers.

Your book is obviously targeted towards game developers but in a way that opens it up to other programmers such as media artists. Are you aware of people using the book outside the traditional gaming community like scientists or artists? What was their work and their feedback?

**DK:** Im pleased to say the book seems to have been most enthusiastically taken up by

the people I was most writing for, the reasonably experienced programmers who came to it from a non-mathematical background and found themselves lacking some basic understanding. In general, though, I have no idea whos using it. Ive had a surprisingly small amount of feedback, although its been uniformly positive when its come.

The physics sections in your book focus mostly on problems that arise in simulating mechanics. If you were to extend the book with additional chapters on other fields of physics, what topics would you add?

Obviously mechanics is the most important bit of physics for your everyday programmer. If I were extending the book, Id probably do more mechanics rather than other areas of physics, with more of an emphasis on approximate methods, particularly Verlet integration. But if I were looking at other areas, Id say optics is the most obvious, because of 3d geometry, and more on waves. Itd be nice to write a 3d water tank simulation with refraction. Magnetism would probably be quite useful, with maybe a little on vector field mathematics. But I wouldnt say any of these are really core topics in the same way as the rest of the book!

**AS:** Do you have any stories, examples or even some source code that illustrates how difficult it can be to implement a proper physics simulations?

**DK:** Whoo, have I ever? :) Magnetic field lines. As a part of the Motors and Generators simulation I did, we wanted to include field line diagrams to show how the wire in a motor is pushed up by the field density beneath it. But it turned out that calculating these fields was really hard. I learned more calculus in the three months I spent on that problem (on and off) than my entire university career. However, I was gratified to discover that an MIT professor specializing in educational visualizations of field lines had exactly the same errors on his site - when I queried him about it he said that he remembered noticing the problem at the time and never managed to get rid of it (if youre interested, the problem was that my field lines kept crossing and in particular, spiraling in to the current-carrying wires) - and it seemed to be a fundamental error in the whole simulation. I only got rid of it by cheating and assuming that the field due to the magnets was uniform rather than being made up of a number of dipoles. (You can see the result at [www.sunflowerlearning.com](http://www.sunflowerlearning.com): go to Evaluation on the top, log in and navigate using the left-hand menu once the software has loaded). I remain convinced Ive found a hitherto unknown theorem about magnetic fields proving that monopoles can exist after all

**AS:** Game developers recognize the importance of proper physics simulations in the interactivity when creating an immersive virtual world. How would you rate the importance of game physics versus graphics & sound, social constructs & story line, or other gaming elements for creating “immersion”?

**DK:** As Ive already indicated, I dont rate physics particularly highly for gameplay, but I agree that its helpful in creating the kinds of highly realistic simulated worlds we expect these days. Theres something satisfying about coming into a simulated space and realizing that you can pick the chairs up and throw them around. I think this particularly comes into its own in multiplayer games, where the ingenuity of the players in discovering new ways to manipulate their environment can create really interesting results. I heard of a particularly wonderful example which I believe was in Everquest. In this game, players can choose whether or not they are able to harm or be harmed by other players. Some genius discovered a way around this by learning the skill of carpentry. A gang of rogue carpenters would surround a supposedly immune character and build a wardrobe around them, trapping them inside, and would only release them when paid a ransom. Because the wardrobe was their property, the captured person couldnt destroy it. Now thats a wonderful example of the consequences of using realistic physics.

**AS:** Historically, the hardware during the beginnings of game design (“the Atari years”) was very, very limited compared to todays standards. Do you think there was any “worthy” game physics in these early games?

**DK:** Im amazed you even ask the question! Of course there was. **AS:** Asteroids. As fine a demonstration of the principles of inertia and momentum as Ive ever seen. Whats more, crucially this was absolutely fundamental to the gameplay. Mastering these principles was the key to success in the game.

**AS:** Can any portion of the game physics described in the book be used to perform real science?

**DK:** It depends what you mean by real science. In and of itself, clearly not. Because computers do what we tell them to do, there is no way to investigate reality by looking at a simulation. We can only look at the simulation and see how closely it mirrors what we see in the real world. However, approached from the other direction, we can certainly do real science by taking a phenomenon observed in nature, hypothesizing about its cause, simulating our hypothesis and seeing if it matches our expectations. This doesnt prove anything, but you can never actually prove anything in science, so thats okay. On the other hand, its possible that you mean other things by real science. For example, you can certainly use these techniques in educational software, its what I do every day.

**AS:** Let me point your attention briefly to a recent development in the gaming industry. You might have heard about the new Ageia physics accelerator cards (if not, head over to this link <http://physx.ageia.com/> and have a look). What do you think of the current crop of “physics-accelerated” games as shown in these demos? Do you think these accelerator



devices help the game designers to make better games?

**DK:** Well, its very telling that mostly these engines are used for blowing stuff up. Creating more realistic explosions, debris and general destruction is all very well, but does it make a fundamentally more interesting or absorbing game than Doom? Im not convinced. Theres a certain hey wow factor in all of this, and naturally the graphics are impressive and help to sell a game, but after a while of playing any game you stop watching the pictures and get involved in the action (or if you dont, its not much of a game!). Thats not to say that the occasional money shot cant be exciting. I remember the first time I came down into the hidden valley with the dinosaurs in Tomb Raider and I was blown away. But really, playability is the key. The other problem with ultra-realistic physics (others might see it as a selling point, but I dont) is that it makes the game much less predictable. For me, an important part of gameplay is the ability to retry a section when something goes wrong. The more realistic the physics becomes, the more chaos starts to come into the picture, with the result that a tiny difference in your timing can result in a huge difference to the game. In some cases (such as the multiplayer games discussed before) this may be a benefit, because the game is not designed to be replayed, so flexibility is good. But for a more difficult single-player game, that can be really frustrating. In the aforementioned Spaced Penguin, for example, there are a number of levels involving gravitational bodies that move around during the level in some simple pattern. The result is that the whole level is guesswork: there really is no way to predict where your penguin will end up, because a split-second difference in the timing completely changes its path.

**AS:** Theres a hot concept called “stealth education” that says its possible for people to play a game because they enjoy it and “accidentally” learn stuff along the way. Unfortunately, the concept has yet to prove its worth with the pre-teen and teen community. So, on this concept there are a few questions:

Are there any commercial game titles available that you are aware of, that implement some form of “stealth education”?

**DK:** Modern-day, no. Asteroids and Worms are the only ones that spring to mind, and I dont think their originators were aiming for that! There are obviously a number of educational titles in game form which are more or less successful as games, whatever their educational merits. The best of these is The Logical World of the Zoombinis (there are a couple of more recent sequels, but theyre not as good), although its mostly about maths rather than physics. On an abstract level, though, it does provide a very good demonstration of the scientific method. But I dont think these examples are quite the same as what youre describing.

**AS:** How would you judge the commercial potential for “stealth educational” games for

the gaming industry of today?

**DK:** Im probably not the best person to ask about whats commercial, given my business record, but it sounds unlikely to me. Education by stealth happens all the time, its called life. I cant see any huge value in building it intentionally into a game. Having said that, theres nothing wrong with taking an educational concept and designing a game around it, but whether youll achieve something that succeeds in both camps is a very difficult proposition.

**AS:** As an historical example of a game with some teaching potential, the early Atari game “Lunar Lander” comes to mind. Would you consider such a game “educational”? If yes, in which way? If no, why not?

**DK:** I would, as I would with Asteroids. This is because they implement one simple idea in a very focused way. Whats more, inertia and momentum are conceptually hard concepts to understand theoretically, so a game like this which gives them a practical basis is an excellent way to learn them. More recently weve also seen Marble Madness, Super Monkey Ball and the mercury blob one (whatever it was called). And I can imagine that you could make games on other science topics with some thought - a puzzle game based on electric circuits, for example.

**AS:** Did you work ever on educational games? (Here, I’d be especially interested in game that incorporate some form of physics.) Do you have any examples of games that try, but fail in educating? What are the reasons for the failure?

**DK:** Ive done various educational games but more focused on Maths than physics. Im also pleased to say that mostly theyve worked quite well from an educational perspective. As for games that try and fail in education, theyre always the ones that try to actually teach something directly. Especially if they say matched to the such-and-such curriculum anywhere on the cover. Its never a sign that youre going to have fun. The real problem with teaching science (as opposed specifically to maths) through games is that beyond a certain elementary level, most of science is about the quantitative rather than the qualitative. We dont just need to know that something goes faster when we push it, we want to know how much faster. Discovering that the planets orbit around the sun is a big step, but discovering why they move around the sun is very much tied up with discovering how they move around the sun, which means measurement. And that process is a long way from anything that fits into todays game models. Interestingly, as part of the TimeHunt project we did create some science-based puzzles, in particular one developed with a Czech group of physical chemists. They devised an alternative chemistry, and a series of experimental devices, and the puzzle was to work out what the devices do and to use them to decode the chemistry. I also spent a while developing some initial ideas for a multi-player adventure

based on alchemy (I cant go into details as its not my project and it is still active to some extent). A key part of the game was developing your alchemical laboratory, and this too would have involved a complex set of rules waiting to be discovered. But as with my example of the Zoombinis earlier, these are examples of games about Science as a tool, rather than about discovering the physics or chemistry of the real world.

**AS:** Now, I'd like to broaden the topic to some more philosophical ideas in game design.

Assuming we had a game that implements some basic laws of physics for its gameplay - say a submarine shooter. What would you think about a "reality" slider for the game that would allow smooth control from "arcade physics" to "real physics" similar to adjusting the sound volume? Would players use it and would it teach them something about the world?

**DK:** Players dont use anything unless it either helps them progress in the game / get a higher score, or makes the game itself more fun or challenging. If this could be achieved, then naturally they would use it. (There are already examples: you can have a pool simulation and choose to enable or disable spin, or a golf game where you choose to ignore the wind). I can certainly imagine it working well in the context of a driving game, which is the best example of a game genre where physical laws are deliberately contravened in order to make the game easier to control. And now I think about it theres another good example: Prince of Persia, where you can choose to stop time. As to whether it would teach you anything, I dont know. As I said earlier, were constantly surrounded by real physics. Everything we do relies on it, and weve evolved over millions of years to take advantage of the regularities of physical laws. So making a game more generally realistic wont teach us anything we cant learn by looking out of the window. (This is as opposed to Asteroids, which is an example of a controlled simulation, which ignores all real-world physics except one small field, the laws of inertia, force and momentum) What makes your particular suggestion interesting, ironically, is the ability to go the other way. Heres reality (or the best simulation of it we can do: lets not forget that its always going to have to be a bit simplified). Now we can choose to abstract one element of it: look, we can decide that Newtons First Law is no longer active, so we can now enable our characters to stop and turn in mid air, as in most platform games. Seeing whats different between the two simulations tells us something. Thats science: isolating variables and performing controlled experiments. Nevertheless, you would have to work hard to make a game that works equally well in both modes, or that makes use of the shift of modes. Perhaps the best context is like Prince of Persia, a fantasy game where we shift from real physics to magic.

**AS:** Spatiality is the main characteristics of most computer games in that they are

mainly concerned with the representation and negotiation of space. Do you agree with this assertion? Could this preoccupation of spatiality in games be an important reason, why only a limited number of physics-fields (mostly dynamics and optics) are used in modern computer games?

**DK:** Possibly. Is it a preoccupation? Or is it just that were physical creatures at a particular scale and so physical spaces are what were used to? The computer, after all, is a visual medium. Its pretty hard to design a non-spatial game when youre representing it on a screen (I know of two games using sound-only, one of which I made, and both of them also revolved around navigation through a space based on aural clues). But you missed one other key element of scientific gameplay: logic. Many, many puzzle games involve logic, and as I said before, problem-solving by logic pretty much is the scientific method. Lets think of the key areas of physics other than mechanics. We have thermodynamics. We have field theory, electromagnetism, gravitation etc. We have fluid dynamics. We have optics and waves (which is really much the same as field theory). We have relativity. We have quantum mechanics. Except insofar as these impinge on mechanics (waves move water, magnetism can affect motion), how would we even make a realistic simulation of these things, let alone turn them into games?

**AS:** One of the weakness of computers that is specifically apparent for games in their limited I/O capability - usually relegated to keyboard and mouse or a gamepad for input. Do you think that the design of the physical interfaces might be an interesting advance for computer games from an educational point of view?

**DK:** Absolutely. It has to be said, though, that mostly its been hard to sell these innovations to the general public, and its not for want of ideas. There are hundreds of controllers out there; hundreds of cameras, VR goggles, laser guns, dance pads. Mostly these have found it hard to prove themselves as more than just one-off gizmos. However, there are other things out there. I had the pleasure of trying a 3d mouse once, which was a ball with a kind of pen stuck in it. It had force feedback, so you could use it to sculpt a ball of virtual clay in a 3d space, and it was a truly wonderful experience. Im sure something like that could make for a fantastic game controller - but once again, would it create anything new from an educational perspective?

**AS:** To make games “better” and more innovative, do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**DK:** I think everyone, naturally, should know more about everything. We should all be much more open to learning in all fields. And its true that both sides have a tendency to disparage the other (although I know many talented artist-programmers). On the other

hand there is an awful lot of knowledge out there and we have to specialize to some extent! But theres another side to game development. Theres programming, theres art, and theres gameplay, and its in the gameplay side that were going to have to concentrate if were going to make genuine innovations in the gaming experience (of any kind, not just education). Gameplay is not about art or science, but about psychology, and both the scientists and the artists working in programming could do well to learn more about that.

**AS:** It is fair to say that computer games ARE very influential in our society (i.e. in terms of money, the game industry is bigger than film and music together). Is there an effect on society? Is that a positive or negative effect? Are games in their effect on society similar to - say - movies and can they distort the users view on reality?

**DK:** Well, looking at my three-year-old sons absolute obsession with games, I cant doubt it to some extent. I know that when Im engrossed in a game I find the same thing he does, that suddenly the whole world is cast in the context of the game. He relates to things in terms of Crash Bandicoot, and its a bit concerning at times! (I keep trying to tell myself its better to let him get over it than to overreact and throw the Playstation in the bin, but its hard at times). Games have a number of features that make them better than films - interactivity, obviously, but they also have a social value that people often miss: games are a lot more fun if done in a group. When watching a film, you have to shut up and pay attention; the same is not true of most games. On the other hand, games do present you with a very simplified view of the world - and Im not talking about physics here, but psychology again, whereas good stories can teach you much more about survival in the most complex environment, human society.

**AS:** To finish off, I'd like to ask you a few questions that came up during my first session of our PhD workgroup - I just wanted to hear how you would answer them (if you have anything to comment on these at all).

How might games be used in scientific research? Do you have any examples?

**DK:** It depends. Naturally, there are plenty of ways to use games to study people! We can look at game strategies (I always thought it would be an interesting experiment to track peoples eye movements while playing a board game to try to work out their thought processes); we can make games that are deliberately unfair to see how people react to them; we can try to use games to induce hypnotic states or particular emotions; we can study the social consequences of games, as you mentioned above, and so on. As to whether games themselves can be useful for science, Im not sure. For physics, Id come out with an all-out no, but for biology, there are certainly examples. The Sims springs to mind immediately as something we really could study to look at patterns of emerging behaviour. Artificial Life generally is an area I expect to see coming in much more in future games: we already

have games that include flocking simulations, stealth games where the guards have simple behaviours, as well as the various Sim games, and I definitely think there is potential for a lot more exploration in this area.

**AS:** Folk Physics is best suited to cope with the kinds of objects and events that we encounter on a day-to-day basis and is not really designed to provide accurate descriptions or explanations of the universe. Do you think game physics “teach” some form of folk physics to the player? Could this have any negative side-effects?

**DK:** I couldn't agree less with this. Folk physics is an absolutely excellent description of the universe at every scale we need to know about as normal human beings. And ironically, the more accurate your game simulations become, the more closely they will confirm the intuitive folk physics of players. No one really thinks that they can run off a cliff, spin their legs and run back before they fall, no matter how many Road Runner cartoons they've seen. The humour of the cartoon, in fact, is exactly how it goes against our intuitive physics. And as we saw before, the differences between the cartoon physics and reality can in themselves make you think about the science: I recall once watching a Bugs Bunny cartoon in which, falling down in a plummeting lift, he saved himself by opening the door and stepping out onto the ground just as it crashed. As a child, it did make me wonder why it wouldn't work. (I always hate superhero films where the falling person is saved by being caught just before they land, as if they wouldn't break every bone in their body just as much by that change of momentum as by hitting the floor!)

**AS:** What is the advantage/difference of a physics games compared to the real physics experience (i.e. in an experiment)?

**DK:** The difference is that one is a game, and one is not. It's a facetious answer, but it's pretty much all there is to say. You can have a game that involves scientific principles, as we've seen, but that's different from a game that just takes place in a physics environment.

**AS:** Would it be more interesting for a game developer to use physics laws to create new games in which specific laws would have the main influence on the game? Or is it more interesting to improve existing games by incorporating exact physics phenomena into the game logic?

**DK:** I think the former, definitely. The great thing about computer games is their ability to let us do things we can't do in the real world, not the things we can! And discovering the difference: that's your stealth education right there.

**AS:** Thanks for your time in answering all these questions. If you happen to have a friend or colleague who might be interested in doing an interview on this topic of game design and game physics, please make an introduction for me.

**DK:** I know hundreds. Ill put the word out on the gd-algorithms mailing list and see if anyones interested.

**AS:** Thank you for the interview.

## E.4 David Bourg

David Bourg is a naval architect and marine engineer who also teaches at the college level in the areas of ship design, construction, and analysis, as well as at high schools on topics such as naval architecture and software development. He is professionally involved in computer game development and consulting through his company, Crescent Vision Interactive (David M. Bourg & Associates, LLC). David wrote the popular book *Physics For Game Developers*, which is an introductory level book on the subject of rigid body dynamics in the context of real-time games. <sup>4</sup>

The interview with David Bourg (DB) was conducted by e-mail from 16 April - 30 April 2006 by Andreas Schiffler (AS).

**AS:** Before I start to get into the more specific details, I'd like to know a bit more about your background as engineer, programmer and game developer leading up to your writing of the book *Physics For Game Developers*. What did you study and where did you get your work experiences?

**DB:** I started programming computers at the age of 12 when my friends and I would fiddle around with graphics and animation with the aspiration of writing a game (this was back in the days of the TRS 80). I studied Naval Architecture & Marine Engineering in college where I found my greatest interests in programming physics-based simulations to solve real world engineering problems. I worked for several ship building/design companies on such marine vessels as hovercraft, patrol boats, work boats, and offshore structures. I started collaborating with a friend of mine on Mac and PC games. This collaboration grew into a business where we successfully developed and marketed many diversionary type games and consumer applications. With the rapid advancement of 3D graphics, I found myself growing increasingly interested in apply my physics/engineering background to make realistic games behave in a likewise realistic manner. My investigation into this

---

<sup>4</sup>References: Bourg (2006) Stewart (2001) amazon.com (2006) O'Reilly Press (2001)

area of game development revealed this was a new avenue for games and there was not much literature on the subject. This is what inspired me to write the book.

**AS:** What was the most interesting aspect of physics that you remember? What about today's physics research - anything on your radar that you are really interested in outside of your field of work?

**DB:** My interests in physics have evolved as I've grown and gained more knowledge and experience. My earliest interests were in electricity and magnetism. While practicing engineering and game development my interests lay mostly in mechanics. My current interests lie in computational fluid mechanics, which is the area of concentration for my PhD research. As a hobby I'm interested in theoretical physics and cosmology.

**AS:** Are you still involved in the computer gaming industry in some form today? In very general terms, what do you think of today's games and game platforms?

**DB:** My involvement in the gaming industry these days is somewhat limited. Currently I teach an online course for the Game Institute. The subject of the course is game physics and I use my book as a text book.

As for today's games my favorite platform is the PC and I'm always impressed with the A-list games. The combination of graphics, physics, AI, art, and storytelling go a long way to create immersive experiences. I enjoy playing when I have time. As for the future I'd like to see more emergent AI.

**AS:** You mention in an interview that "physics-based realism will be crucially important for programmers ... in developing ever more sophisticated games". Would you see the incorporation of better physics as an "innovation" or more as a "refinement" for games? When do you think game physics had an impact in the gaming industry and with what title?

**DB:** I see better physics more as a refinement these days. A few years ago I would say real physics on the level of integrated rigid/soft body simulation was an innovation. The title that sticks out most in my mind in this regard is Half Life II. But keep in mind, even Pong (arguably the first computer game) used real physics to compute the trajectory of the ball after striking the paddle. So, one could argue that physics has been in games since day one and has been evolving ever since.

**AS:** How do you think mathematics, engineering and physics is perceived today by the general public? Is there a need to educate more in this area? If so, why?

**DB:** I think most people view mathematics and physics as impractical and something they'll never use in the real world. Engineering is the practical application of science so I don't think the perception is as bad. I do think more education is required in the areas of



math and science in general. Not so much on teaching these subjects but rather teaching how these subjects are so important and applicable in modern work and life.

**AS:** Now, I'd like to get a bit more technical and also ask some questions related to your book *Physics For Game Developers*.

Your book is obviously targeted towards game developers and was well received by that audience. Are you aware of anyone else was using the book outside the gaming community like scientists or artists? What was their feedback?

**DB:** Actually, yes. I've received a lot of feedback from science teachers who say they use my book to help generate interest among young students in classical sciences like physics. Put in the context of modern video games, physics does not seem as boring and irrelevant to kids.

**AS:** Your book focuses mostly on problems that arise in simulating mechanics. If you were to extend the book with additional chapters on other fields of physics, what topics would you add?

**DB:** I'd add chapters on soft-body mechanics, fluid mechanics, and perhaps light and optics.

**AS:** Do you have any stories, examples or even some source code that illustrates how difficult it can be to implement a proper physics simulations?

**DB:** Nothing specific except to say that the most challenging part of writing physics simulations is tuning them. Very often your first go at yields unrealistic results at best and just crashes at worst. Tuning is the iterative process of tweaking and refining the simulation to get things just right and stable. It can be extremely frustrating and time consuming.

**AS:** The book's introduction discusses one of the driving forces for game physics: the game developers' recognition of the importance of proper physics simulations in the interactivity when creating an immersive virtual world. How would you rate the importance of game physics versus graphics & sound, social constructs & story line, or other gaming elements for creating "immersion"?

**DB:** It really depends on the game. For a race car game where crashing is part of the entertainment I'd say realistic crash simulation is just as important as graphics. In fact, realistic physics makes graphics programming easier and better in many cases. Even RPG games where the story dominates can use real physics to good advantage while simulating realistic special effects such as fire burning or spell effects.

**AS:** Historically, the hardware during the beginnings of game design ("the Atari years") was very, very limited compared to today's standards. Do you think there was any physics

as discussed in your book worth mentioning in these early games?

**DB:** Yes. Like I mentioned early even Pong used physics.

**AS:** Can any portion of the game physics described in the book be used to perform real science?

**DB:** Yes. In fact all the material discussed in my book comes from my experience in performing real science and engineering. The science and math is the same its just the problems and purposes are different between science and engineering applications versus game applications

**AS:** Let me point your attention briefly to a recent development in the gaming industry. You have probably heard about the new Ageia physics accelerator cards (if not, head over to this link <http://physx.ageia.com/> and have a look). What do you think of the current crop of “physics-accelerated” games? Do you think these devices help the game designers in creating better games?

**DB:** I have been following these developments but I have not yet tried the games that take advantage of the physics cards. Therefore, its tough for me to comment. I can say that I expect these cards to do for physics what graphics cards did for graphics. I predicted this in my OReilly interview article years ago when my book was released.

**AS:** Theres a hot concept called “stealth education” that says its possible for people to play a game because they enjoy it and “accidentally” learn stuff along the way. Unfortunately, the concept has yet to prove its worth with the pre-teen and teen community. So, on this concept there are a few questions:

Are there any commercial game titles available that you are aware of, that implement some form of “stealth education”?

**DB:** None that Im aware of.

**AS:** How would you judge the commercial potential for “stealth educational” games for the gaming industry of today?

**DB:** Im not sure. Ive been involved in some education products and its very difficult to balance the education content relative to the entertainment content. Too much education content makes it obvious that its an edutainment product and not enough is useless. I think its tricky.

That said, I think that some games can be used as sort of virtual labs to teach some topics even if the game was never intended to be educational. For example, students under proper instruction could conduct basic projectile motion demonstrations in a game like Half Life II.

**AS:** As an historical example of a game with some teaching potential, the early Atari game “Lunar Lander” comes to mind. Would you consider such a game “educational”? If yes, in which way? If no, why not?

**DB:** Off the top of my head I suppose you could get some educational benefit from that game. The learning object would be to learn how forces (as vectors) act on objects to affect their motion.

**AS:** Did you work ever on educational games? (Here, I’d be especially interested in game that incorporate some form of physics.) Do you have any examples of games that try, but fail in educating? What are the reasons for the failure?

**DB:** I have not worked on any educational games that use physics.

**AS:** Now, I’d like to broaden the topic to some more philosophical ideas in game design.

Assuming we had a game that implements some basic laws of physics for its gameplay - say a submarine shooter. What would you think about a “reality” slider for the game that would allow smooth control from “arcade physics” to “real physics” similar to adjusting the sound volume? Would players use it and would it teach them something about the world?

**DB:** From practical standpoint I think a reality slider would be hard to implement. The level of realism would have to be quantized so-to-speak. Moreover, arcade style games often just fake the physics, but this puts more burden on developers to actually think of all of, or enough, scenarios with which to produce graphics for. Real physics, often is more beneficial to the graphics programmers/artists than the player; it relieves programmers and artists of the burden of preparing too much time consuming scripted graphics.

**AS:** Spatiality is the main characteristics of most computer games in that they are mainly concerned with the representation and negotiation of space. Do you agree with this assertion? Could this preoccupation of spatiality in games be an important reason, why only a limited number of physics-fields (mostly dynamics and optics) are used in modern computer games?

**DB:** Well, the very nature of video games is a visual form of entertainment which has evolved into games with rich virtual environments. Making these rich environments immersive means making them look realistic and behave realistic, which is where the physics come in. Mechanics and optics fit this role quite well. I think other physics fields see limited use because they are probably not as interesting in a game, that is, they don’t lend themselves to creating better entertainment. Just because a physics field can be implemented in a game doesn’t mean that it’s fun or makes the game anymore fun. The bottom line after all is making the games fun.

**AS:** One of the weakness of computers that is specifically apparent for games in their limited I/O capability - usually relegated to keyboard and mouse or a gamepad for input. Do you think that the design of the physical interfaces might be an interesting advance for computer games from an educational point of view?

**DB:** I think so from an entertainment and immersive point of view. Some of our most fundamental and sensitive senses are touch and smell. Providing tactile feedback or olfactory stimulus are areas where I/O can be enhanced. As youre probably aware, there are devices on the market that attempt to take advantage of these senses to create even more immersive experiences.

**AS:** To make games “better” and more innovative, do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**DB:** I think the teams that create games must use all of the above.

**AS:** It is fair to say that computer games ARE very influential in our society (i.e. in terms of money, the game industry is bigger than film and music together). Is there an effect on society? Is that a positive or negative effect? Are games in their effect on society similar to - say - movies and can they distort the users view on reality?

**DB:** I agree that games are influential on society, certainly in terms of economics, education, careers, etc. The second part of your question seems more to with games influencing behavior. This is a very debatable topics. Im inclined to say their influence on behavior is somewhat sensationalized, especially in the media.

**AS:** To finish off, I'd like to ask you a few questions that came up during my first session of our PhD workgroup - I just wanted to hear how you would answer them (if you have anything to comment on these at all).

How might games be used in scientific research? Do you have any examples?

**DB:** One example is to study social interactions in multiplayer games.

**AS:** Folk Physics is best suited to cope with the kinds of objects and events that we encounter on a day-to-day basis and is not really designed to provide accurate descriptions or explanations of the universe. Do you think game physics “teach” some form of folk physics to the player? Could this have any negative side-effects?

**DB:** All games have some elements of unrealistic physics either by design or by necessity due to other development restrictions. I would that most game players realize games are make believe and I cant envision at the moment specific examples that would create negative side effects.

**AS:** What is the advantage/difference of a physic games compared to the real physic experience (i.e. in an experiment)?

**DB:** Repeatability, cost effectiveness, safety.

**AS:** Would it be more interesting for a game developer to use physics laws to create new games in which specific laws would have the main influence on the game? Or is it more interesting to improve existing games by incorporating exact physics phenomena into the game logic?

**DB:** Neither. I think its more important to use real physics when it makes the programmers job easier or makes the game more entertaining.

**AS:** Thank you for the interview.

## E.5 Ed Rotberg

Ed Rotberg worked with Atari during in the late 1970s and early 1980s when several classic arcade games were introduced by Atari such as "Missile Command", "Asteroids", "Centipede", "Tempest" and "Battlezone". Ed was the primary force behind the 3-D tank game "Battlezone" which is considered the first "Virtual Reality" game. He later worked on an update to "Asteroids" called "Blasteroids" which is an early game using physics principles. Later in 1981-1986, Ed was Vice President of Software at Sente, a new coin-op company founded by a former Atari executive. After that he worked also for Apple Computer and 3DO.<sup>5</sup>

The interview with Ed Rotberg (ER) was conducted by e-mail from 15 April - 22 April 2006 by Andreas Schiffler (AS).

**AS:** Before I start to get into the more specific details, I'd like to know a bit more about your background as a programmer and game developer leading up to your work at Atari around 1980. What did you work on and what did you study? Did you have any connections to the sciences (i.e. Mathematics or Physics)?

**ER:** Well, my first job out of college was working at Texas Instruments on their Advanced Scientific Computer system (ASC). These computers were used for advanced mathematics

---

<sup>5</sup>References: Wikipedia (2006*a*) Hague (2002*b*) Rotberg (1982)

modeling at such places as the Geophysics Fluid Dynamics Laboratory. That said, I was on OS programmer so I did not dabble in any of the science aspect of its use.

I had a brief stop working on a minicomputer system for doing photoelectric typesetting. This is in 1976, and was pretty much state of the art at the time.

From there I went to G.D. Searle Pharmaceuticals where I worked interfacing micro computers to lab equipment. So while this was definitely tied into the sciences, again, I was pretty much just doing interface work.

**AS:** What was the most interesting aspect of school or university physics that you remember? What about today's physics research - anything on your radar that you are interested in?

**ER:** My personal favorite part of physics was optics. It turns out that I've used dynamics a lot more than optics, so I wish I had paid better attention.

**AS:** Are you still involved in the computer gaming industry in some form today? What do you think of today's games and game platforms?

**ER:** I am still involved in the gaming industry. I am CTO of a very small game development company where I live in the Sierra foothills. As for today's game platforms, they are quite simply incredible. The processing power, especially for graphics is simply awesome. As for the games being developed today, while they are very well done and visually stunning, they are pretty much evolutionary in nature. In fact, I would guess that 99 percent of the games commercially available are either licensed properties, or sequels to other established game IP. There is very little creativity going on in the actual design of gameplay itself at the major publishing houses.

**AS:** You had some input into the movie TRON, a flop at the box-office at the time, which has evolved to become a very influential icon of the "arcade-generation". Do you think the arcade era, including some of your work, has influenced society beyond the screen in some way? If so, how?

**ER:** I think that there has been a definite influence in some of the slang used today, but the social aspect of the arcade versus the home game has pretty much gone away. Now the "social" side of video gaming is represented by the MMORGs. Beyond that, video games today are a more and more accepted form of entertainment. So much so that it has become a convenient political avenue for aspiring politicians looking for a "cause".

**AS:** You mentioned in an interview, that you left the game industry partially because of "indifference to innovation in game design". Is this still true, and how could one introduce more creativity into today's game design?

**ER:** As you can tell from some of my comments above, I still believe that this is true.

The cost of doing a “triple-A” title is enormous today. Video game budgets get into the tens of millions of dollars. Naturally, with this much money at stake, the publishers have become almost entirely “risk averse”. I say “almost” because I still hope that there is someone out there that still values creativity. Most of that creativity seems to be coming from the colleges, and small developers who can afford to fund their own work. At the GDC there was a wonderful exhibit of games from small, independent developers. Some of the titles displayed there were very unique and highly creative. In my opinion, its a shame that probably none of these will ever get shelf space in a commercial environment.

**AS:** Now, I'd like to get a bit more technical. Some of the earliest computer games used simulations as one of the main game principles - for example “Battlezone”, for which you are famous for.

Since you might be more familiar with some of the games from that era, can you point me to one or more similar “simulation” games of the era?

**ER:** Im glad that you are using the word simulation in quotes. These games were really not any kind of simulation, with the possible exception of Lunar Lander. Even Battlezone was not a simulation by any stretch of the imagination. It was a game that made it easy to control a tank-like vehicle, which obeyed practically no laws of physics. There was no inertia, there was no gravity applied to projectiles. It did make use of 3D perspective math, but thats not what I think of when I think of simulations. So, I basically dont think any simulations came to the arcade until some of the later driving games.

**AS:** The hardware during the beginnings of game design when you worked for Atari was very, very limited compared to todays standards. How did that impact your creativity? Did you have any choices in your algorithms?

**ER:** There was always choice in algorithms, and even more so in general code design. The limitations were extensive and included the amount of RAM and ROM (all of the code was in ROM in those days), processor speed and display and audio technology. This necessarily limited the scope of the games we were able to develop. That said, it also spurred creativity, not only in game design, but in code design as well. When you know that every machine cycle is precious, every byte of RAM is at a premium, you are pretty much forced to employ non-traditional techniques to take the maximum advantage of your platform.

**AS:** Do you have any stories, examples or even some source code that illustrates how difficult it was to implement proper simulations at the time?

**ER:** I have no source code from those days. But what I can tell you is that in order to do the math required for Battlezone and keep the frame rate reasonable, we employed

a second processor to handle the matrix math. This was a custom designed bit-slice processor. Even with this second processor, we ended up simplifying the transforms to 2x2 matrices instead of 3x3 matrices, doing the perspective divide in a separate step. Yes, Battlezone was actually more 2.5D as far as math is concerned.

**AS:** Did any of the early games you made, use an algorithms that one could call “physics simulation”? Is there real physics in the game “Battlezone” for example? If so, what was it?

**ER:** Battlezone pretty much had very, very little of what anyone would call “real” physics. About the only thing that term would apply to was the flight motion of the pieces of the tank or missile when it was shot. Even then the “physics pretty much stopped once each piece hit the ground. And I do mean only the flight motion. The rumbling rotations of the pieces in flight was pre-canned and about the arbitrary graphical center of the piece. No attempt to establish the true center of gravity.

**AS:** There was a special version of Battlezone, “The Bradley Trainer” (also known as Army Battlezone or Military Battlezone). Did the depth of the simulation and especially the physics in this version differ from the arcade version significantly???

**ER:** Yes, there was such a version. It was programmed in 3 months time, under a lot of stress. That said, the game did have a bit more “real physics” than Battlezone did. The trajectory, and if I remember correctly, the muzzle velocity of the ordinance was done using real physics. This required additional math support from the bit-slice processor, though not to process the motion, as that is pretty simple. The additional support was to perform 3D collision checks as we a) had many shells active at any one time, and b) we were now doing collision testing in 3D instead of Battlezones 2D.

**AS:** Since the Atari times, computer hardware has progressed rapidly and each generation of games has more CPU power available. Has any of your games since these early years had some physics simulation component that took advantage of these readily available clock-cycles? If so, how good was it and what did it do? If now, why not?

**ER:** First of all, there have been a number of games that I have worked on since the “Atari times”, and even during the Atari times that used a lot of real physics. Is some of them I was not the programmer responsible for the physics, and in others I was. For example I was involved in an Atari product called “Hard Drivin II” which featured a lot of real physics for the vehicle simulation. The vehicle model was quite complex at the time, and ran at a very high update rate relative to the display refresh, which was unusual at that time. That said, I was only responsible for writing some music playing code for this game. I worked on two other driving games that used advanced “real physics” modeling for the vehicle dynamics. These simulations modeled everything from torque and



horsepower curves for the engine plant to the shock absorber system, to the friction from the individual tire patches and road type surface. One of these “games” (NASCAR Silicon Motor Speedway), featured 6 Pentium III processors, 3 projection screen monitors, full-motion sled, and force-feedback steering. One of the CPUs was dedicated to the physics modeling alone, and ran at a 1-millisecond update rate. We even went so far as to try to model the physics of a real engine in order to synthesize the engine sound. These days, this is all handled by a single dual-core Athlon based processor. Time do keep changing.

Right now Im working on a sports title (I cant be more specific than that) that features a lot of “real physics”.

In some cases these products were extremely good. The NASCAR simulator is a marvelous experience - once you learn how to control the vehicle. It is not quite like driving a real car, although it is very close. Driving is always difficult as people are very much familiar with driving real vehicles, but not these very high powered racing vehicles. These simulators are still in operation today, and it would probably be appropriate for you to interview users of this system both regular and first-time users.

**AS:** Let me point your attention briefly to a recent development in the gaming industry.

I’m not sure if you see or heard anything about the new Ageia physics accelerator cards yet - if not, head over to this link <http://physx.ageia.com/> and have a look. What do you think of these “accelerated” games? Is there any worthwhile physics in it - any learning potential? Do you think these devices help the game designers?

**ER:** I have mixed feelings about this. I personally feel that a power PC user would be better off for more than just games by purchasing a dual core processor. Developers can certain take advantage of a dual core system and probably get even better performance than provided by this additional hardware which, I would imagine, imposes a specific API that would not be present when programming to take advantage of dual core CPUs. Perhaps such a processor and its associated API will relieve the burden placed on the software designer for designing and implementing efficient physics code, but I dont see that as necessarily being a win for either the developer or the consumer. I would imagine (though I have no data on this) that the cost to the gamer of buying this hardware would exceed that of equipping their PC with a dual core processor.

**AS:** Theres a hot concept called “stealth education” that says its possible for people to play a game because they enjoy it and “accidentally” learn stuff along the way. Unfortunately, the concept has yet to prove its worth with the pre-teen and teen community. So, on this concept there are a few questions:

Are there any commercial game titles available that you are aware of, that implement

some form of “stealth education”?

**ER:** A case could be made that any number of games implement this stealth education whether the designers intended this or not. The Bradley Trainer you referred to earlier was a decided attempt to do this by the Army. They found that their troops were avid game players and wanted to parlay this into additional training by helping them to readily identify silhouettes of friendly versus enemy vehicles/aircraft. This is in addition to training them on the basic operation of the gunner station in a Bradley IFV.

Beyond this, “edutainment” has long been a viable, if not major, market for games. There are very, very many of these that have been created, some of which succeed better than others at their intent.

**AS:** How would you judge the commercial potential for “stealth educational” games for the gaming industry today?

**ER:** Same as it ever was, with the exception that modern technology allows A) vastly more content than was previously possible, and B) a more immersive environment.

**AS:** As an historical example of a game with some teaching potential, the early Atari game “Lunar Lander” comes to mind. Would you consider such a game “educational”? If yes, in which way? If no, why not?

**ER:** It is absolutely educational for any number of reasons. It demonstrates principles of gravity, inertia, and action-reaction.

**AS:** Did you work ever on educational games? (Here, I’d be especially interested in game that incorporate some form of physics.) Do you have any examples of games that try, but fail in educating? What are the reasons for the failure?

**ER:** I worked on an edutainment product for the 3DO company, called Station Invasion, however, there was no physics in the intended “curriculum”, so I cant help you there. I am perhaps not the best individual to ask about examples of educational games that succeed or fail. I know that to the extent that Station Invasion was not as big a success as we would have liked, the problem was definitely that there was insufficient content before it started repeating. There are a number of reasons for this, but needless to say that todays technology would allow more content, though there were also time and budget restrictions at the time that also affected the product.

**AS:** Now, I’d like to broaden the topic to some more philosophical ideas in game design.

Assuming we had a game that implements some basic laws of physics for its gameplay - say a submarine shooter. What would you think about a “reality” slider for the game that would allow smooth control from “arcade physics” to “real physics” similar to adjusting the sound volume? Would players use it and would it teach them something about the

world?

**ER:** This is a very, very broad question with lots of implications. The answer would be highly dependent upon the game involved. Certainly for a sports simulation, you don't want to provide such a slider for a number of reasons. The learning curve of a game makes it such that players who have played on one setting would be at a disadvantage when competing against others on a different setting. The same holds true really for any multiplayer game.

The broader question is exactly what is the worth of such "learning" from a video game. In my opinion, at least for the general case, it is not very much. Have very accurate physics in some part of a simulation, and not absolutely every part takes that physics out of context and greatly diminishes its value and probably its "learnability". For example, using very, very accurate physics for the vehicle dynamics in a driving game teaches the player very little when their body and inner ear cannot feel the effect of the forces that a real vehicle would present to a driver. This is not to say that there can be nothing learned, but such learning out of context probably has a very reduced value to the player.

It is a point of fact that many games will intentionally, knowingly, go out of their way to deviate from "real physics" in order to improve the playability of a game. The edutainment developers learned long ago that know matter how well a product can teach, if the player is not engaged by the game, it will do no good. The primary goal and responsibility of a game developer is to develop a fun product. When that is at odds with real physics, the reality of the physics must go.

This is not to say that there are not some games that can work very well by being built around real physics. For example the more accurately a golf game can simulate the physics of a real golf ball flight and response to wind, grass, etc., the more authentic the experience for the game player. Of course in this case, the game playability need not be sacrificed as the developer can tune the skill level of the player to a great extent.

I will leave you one very interesting example of some interesting physics in a game. Perhaps you are already familiar with it, but it is an indie game that you can download from the Internet. The name of the game is Strange Attractors, and it is a game based upon being able to turn gravity on or off as the players sole control. Certainly the physics is not completely accurate. According to the developers, the amount of gravity that a body exerts in this game is linearly dependent upon the radius of the object. Further, gravity only affects the players piece; the other bodies in the game do not affect each other. Nevertheless, I found it a very interesting and creative concept. Here is the link: <http://www.ominousdev.com/>

**AS:** Spatiality is the main characteristics of most computer games in that they are mainly concerned with the representation and negotiation of space. Do you agree with this assertion? Could this preoccupation of spatiality in games be an important reason, why only a limited number of physics-fields (mostly dynamics and optics) are used in modern computer games?

**ER:** Actually, I am somewhat at odds with your assertion, primarily because of the word “mainly”. Certainly a large number of games can be characterized this way. But in fighting games, for instance, the main concern is the representation of the physical combat. In many sports games, while negotiating space is a part of the game, the game skills, batting, golfing, running football plays, etc, is at least as important, if not overwhelmingly so, than any negotiation of space.

Given that, I will fall back again and state that it is my opinion, based upon my experience that the two primary reasons for games eschewing the implementation of real physics are Playability and Performance. In some cases, difficulty in implementing real physics is a factor as well. Even if a designer intends to implement “real physics” it is unlikely, except in the most trivial cases, that there will not be compromises or outright failings in the implementation.

**AS:** One of the weakness of computers that is specifically apparent for games in their limited I/O capability - usually relegated to keyboard and mouse or a gamepad for input. Do you think that the design of the physical interfaces might be an interesting advance for computer games from an educational point of view?

**ER:** Only in so far as such games are used at a center, or certainly so in the case of arcade games where the control design is part of the game design and the specialized controller is shipped with every game. If you are counting on a player purchasing a special control in order to play a specific game, then you are setting yourself up for failure, unless you are willing to target only a very small, fanatical market. I do not see any change to the general interface for computers to change any time soon, and certainly not in a way that is primarily intended for game play.

**AS:** To make games “better” and more innovative, do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**ER:** Both! There is never much downside in more training. To follow your artistic paradigm, knowledge expands the palette with which the artist can create their work. I use Photoshop extensively for my hobby of photography. It has helped me very often to have some techniques that even our artists are not familiar with in order to achieve certain

graphics effects. A trivial example to be sure. J Nevertheless, having cross training allows each developer to understand more of the gestalt of the development process. That is rarely a bad thing.

**AS:** It is fair to say that computer games ARE very influential in our society (i.e. in terms of money, the game industry is bigger than film and music together). Is there an effect on society? Is that a positive or negative effect? Are games in their effect on society similar to - say - movies and can they distort the users view on reality?

**ER:** In most cases, games are no better or worse than other forms of entertainment. A person who spends 18 hours a day playing an MMORG is just as depraved as someone who sits on the sofa and watches TV for the entire day. As with most things, moderation is the key. I dont think games really distort reality more than TV or movies, but some individuals will always be susceptible to such distortion. At least in the case of children playing games, they hopefully have their parents to help them maintain a clearer view of reality or be there to limit their childs exposure to any of these entertainment media.

**AS:** To finish off, I'd like to ask you a few questions that came up during my first session of our PhD workgroup - I just wanted to hear how you would answer them (if you have anything to comment on these at all).

How might games be used in scientific research? Do you have any examples?

**ER:** About the only scientific research I can think of where games could be used would be for physiological research such as perception and reaction, or psychological research. Beyond that, I am dubious of their potential efficacy in scientific research.

**AS:** Folk Physics is best suited to cope with the kinds of objects and events that we encounter on a day-to-day basis and is not really designed to provide accurate descriptions or explanations of the universe. Do you think game physics "teach" some form of folk physics to the player? Could this have any negative side-effects?

**ER:** Certainly games teach the physics that they implement. In most first person games, a player learns the way to control a characters movement in a game in order to negotiate the game space (as you pointed out). This involves learning the way that physics were implemented in the game - usually having only limited association with reality.

I dont believe that a player who manipulates a Spider Man character to climb up a wall or cling to a ceiling believes that they are actually capable of such things. Early arcade "driving" games feature top-down views of a car on a course, and having steering wheels that could turn infinitely in either direction and often required the player to spin the steering wheel in order to maneuver the car at high speed did not impinge upon a drivers ability to control a real car. Any negative side effects from video games are not likely to

be related to folk physics, or “game physics” as we refer to it.

**AS:** What is the advantage/difference of a physic games compared to the real physic experience (i.e. in an experiment)?

**ER:** Its safer, cheaper, and allows for an experiment that might not be practical in real life. That said, this also applies to any computer simulation regardless of whether it is in a game or not.

**AS:** Would it be more interesting for a game developer to use physics laws to create new games in which specific laws would have the main influence on the game? Or is it more interesting to improve existing games by incorporating exact physics phenomena into the game logic?

**ER:** I dont think that this is an either/or situation. There are so many possibilities out there that it behooves us to explore them. The sad fact is that such innovation is unlikely to come from the industry leaders at this time in our business for reasons already mentioned.

## E.6 Kevin Ryan

Kevin Ryan graduated from the University of Oregon with a BS in Computer and Information Science in 1983 and become one of the original owners/-partners of the game development company Dynamix. He has been creating games for over 20 years, including titles for Apple II, Commodore 64, Amiga, and MS-DOS/Windows computers. Kevin programmed a well known and prize winning game title called “The Incredible Machine” (TIM) which was first released in 1993 (and re-released several times until 2001). TIM simulates not only the physical interactions between objects, but also ambient effects like varying air pressure and gravity - a game design innovation which created a large fan-following around the game and earned the producers a patent in 1996.<sup>6</sup>

The interview with Kevin Ryan (KR) was conducted by e-mail from 6 May - 7 June 2006 by Andreas Schiffler (AS).

---

<sup>6</sup>References: Garage Games (2002) Garage Games (2005) Wikipedia (2006b) Gorp (2004)

**AS:** Before I start to get into the more specific details, I'd like to know a bit more about your background as programmer and game developer. What did you study and where did you get your initial work experiences as game developer?

**KR:** I graduated from the University of Oregon in 1983 with a major in Computer and Information Science and a minor in Mathematics. I wrote Zoo Master for the Apple II (in 6502 machine code) my senior year and it was published by Earthware (a very small company). Soon after that I hooked up with three other guys and became a part-owner of the new game development company Dynamix.

**AS:** Many of the your game titles involve some form of physics simulation. Did you study any physics and what was the most interesting aspect of your physics education that you remember? What about today's physics research - anything on the radar that you are really interested in outside of your field of work?

**KR:** The only physics that I had at college was the first year course sequence that all physics majors took at the University of Oregon. It was a little more rigorous than the other sequence that was for non-physics majors. I also took an interesting course called Physics of Music that was fun, but was also, I found out on the first day, one of those courses that non-science majors take to get their required science credits. The first class started with the Professor stating "We will have some math in this class addition/subtract and (gasp) also division."

They only thing that I have currently looked at is research papers relating to friction and how it effects the rolling and bouncing of balls off various surfaces - very specific to a game I am currently close to finishing. In working on my current Minigolf game I've noticed how much easier it is to look up almost anything on the Internet. For TIM I had to almost exclusively rely on my college math and physics books.

**AS:** You have been involved with the gaming industry for many years now. In very general terms, what do you think of todays games and game platforms?

**KR:** From a development standpoint, life is great. Much better tools versus in many cases non existent tools. Video cards that do most of the work for you versus having to write low level graphics routines in assemble language. Lots of memory versus for TIM having everything including graphics have to fit within less than 400 Kb. A new delivery platform, the Internet, eliminating the need for a retail box. Open source tools a click away. Research on any subject a click away.

As far as playing games goes they look prettier, sound better, and from my perspective give that same amount of entertainment.

**AS:** Now, I'd like to get a bit more technical and also ask some questions related to the well known game title "The Incredible Machine" (TIM) which features some innovative use of physics (or its simulation) in a game.

How did the initial idea for TIM evolve? What did you contribute to the concept and the actual implementation that made it the game we know?

**KR:** The Incredible Machine was originally a couple of screens on Commodore 64 with mock ups of blocky car machine things - nothing like the final product. It was shown to Electronic Arts, but instead they wanted us to build Arctic Fox for the soon to be released Amiga computer. This was in 1985.

At the start of 1993 I started work on TIM with the only connection to the earlier one being the name. I spent January and February of 1993 working on the design of TIM at home. Ended up with a 60-80 page design document. I then spent March getting the technical coding complete - the basic building blocks that I would need to implement all the parts.

The parts' design, the interactions, and puzzles were pretty much all my design with feedback from Jeff Tunnell. I did all the coding. The way the GUI works was pretty much designed by the group of people at the JTP offices lead by Jeff. I would come into the office once a week to show what I had done and get feedback from everyone. One day they had a very elegant user interface to replace my clunky one.

One artist did the bulk of the work and a second artist did a little too. There was a music guy who did the music and all the sound effects. The total cost was \$37,000 which was very low in comparison to all the other games being worked on within the company.

**AS:** TIM involves some pretty nifty physics elements. Could you point out a few of the highlights and comment on how easy or difficult it was to implement? What is fake (or left out)?

**KR:** TIM was written in 1993 and so I had big concerns about performance on the computers of the time. Everything was done in integer math. I treated 1024 as 1 and everything below that as the fractional part. I could do a quick shift of 10 bits to the right to get the integer part. My machine code background came in handy here.

All parts are defined by a polygon outline. A lot of time was spent getting a very quick collision system in place. To get it working took a little thought - to get it fast was much tougher. Each part had a set of physics values assigned to it (mass, center of gravity location, density, etc.) and each line segment along its polygon border could have its own special characteristics. Once this underlying system was in place new parts could be added in just a day or two.



Almost all physical interactions are just one nice mathematical formula worked out by Sir Issac. Add in gravity and then fake the way that air density works and it is very surprising how rich a set of interactions you can get.

Ropes are completely fake - splines. There are no rotational forces anywhere in the game. Parts don't rotate, but the bitmap animation of the ball type objects makes it look like they are. The balls were all just 8 sided polygons. Lots of parts are defined as static - i.e. infinite mass and not effected by gravity. I can them put them in a different linked list within the game and cut down on my main loop processing time..

**AS:** Do you have any stories, examples or even some source code that illustrates how difficult it can be to implement a proper physics simulation?

**KR:** Everything in TIM is done in discrete time steps one part at a time. Imagine two balls both moving in the same direction very quickly with not much distance between them with the ball in front moving slightly faster. Now what happens if the ball in back is processed before the ball in front and is moving fast enough to hit the front ball?

The relative velocities of the two balls is such that they are moving away from each other, but they have collided. What happens is they end up bouncing into each other and get stuck. Ooops. My simplifying has resulted in feeding garbage values into impact routine. Interestingly this exact same bug showed up in the 3D Minigolf game that I am currently working on: [www.minigolfmania.com](http://www.minigolfmania.com)

TIM is 2D not 3D. So everything happens within a (x,y) coordinate system - less complex and faster. But the resolution of the screen that you see things on though is discrete pixels. Figuring out the best way to do the mapping between the polygon outline and the actual screen pixels took some thought.

**AS:** Wikipedia authors write "the engine does not use a random number generator in its simulation of physics, assuring that the results for any given 'machine' are reproducible". Is it true, and if so, why was it designed that way? Did you experient with randomness (maybe for some game elements)? If you were to add a random element, what would it be?

**KR:** One of the first design decisions I made was that every machine that was built would be deterministic. We had built in solutions with the game and it was important that they always run the same. One interesting bug was that the order of the parts in a linked-list was reversed every time I saved and reloaded a level. This would result in the same machine running differently solely because the parts were processed in a different order. I ended up fixing my loading/saving routines.

The only random elements in the game are things that do not affect the physics. For

example ambient animations that some of the parts do like the little guy in the game who would occasionally look at his watch or tap his foot.

**AS:** Could TIM be used to perform real science? If so, do you have any suggestions or examples? Was there any feedback regarding TIM from outside the gaming community (i.e. schools, universities or artists)?

**KR:** Yes I think it could be used. Take a beach ball and a cannon ball and turn up the air density and see how long it takes them to fall. Now turn down the air density to a vacuum - hey look, they fall at the exact same speed. Try a balloon and see that it no longer rises, but falls also. Hey look, the balloon falls at the exact same speed as a cannon ball in a vacuum.

One thing I wanted to do, but didn't have the time, was to create a level that was a series of AND and OR gates using perfectly elastic spheres and some switches that would end up adding the number of balls that were dropped through the machine. Some low level computer hardware circuits act this way. I was going use the pool balls that were not only perfectly elastic, but also not affected by gravity. Would have been neat, but didn't have the time.

I had to finish by mid-September to get into the stores for Christmas so there is a real time constraint in development. This flows back to the initial design where things are not included not because they wouldn't be good, but also because the realities of shipping time frames make them impossible.

TIM made its way into many schools. I know that Sierra had a teacher's manual made up although I never saw it myself. I think it was used in the 6th through 9th grades.

**AS:** Today's PCs can perform more complicated simulations. If you were to extend TIM with additional component and simulations from other fields of physics such as nuclear-physics or electrodynamics, what elements would you add?

**KR:** Real optics and electricity were both things that would be interesting. The original TIM just had switches with the simple states of electricity on or off. The lasers were just colors and color mixing.

**AS:** What improvements were made in the TIM follow up "Chain Reaction"? What's your take on the competition "Crazy Machines"? What new stuff will we see in your next "Rube Goldberg"-style game?

**KR:** The major changes was that Chain Reaction was 3D. We restricted all parts to a 2D plane for reasons of user interface. We went back on forth on Chain Reaction with having gravity always on as you place parts and making it more like TIM where you hit a button to turn on gravity. I don't remember what we finally went with, but I think it

was like TIM because there were game play problems with gravity always on.

The new games like “Crazy Machines” are really cool. Beautiful graphics. Great physics.

Hey, how did you know that I may be doing another “Rube Goldberg” type game? That’s supposed to be a secret. :)

**AS:** Let me point your attention briefly to a recent development in the gaming industry.

You have probably heard about the new Ageia physics accelerator cards (if not, head over to this link <http://physx.ageia.com/> and have a look) or similar developments using the shader hardware on the graphics card. What do you think of the current crop of “physics-accelerated” games? Do you think these devices help the game designers in creating better games?

**KR:** I have the sense that the current games are only just now starting to scratch the surface of what can be done. We are still in the learning phase of how to really use this to enhance game play and fun. I have heard of the Ageia card, but don’t know the specifics.

**AS:** There’s a hot concept called “stealth education” that says it’s possible for people to play a game because they enjoy it and “accidentally” learn stuff along the way. Unfortunately, the concept has yet to prove its worth with the pre-teen and teen community. So, on this concept there are a few questions:

Did TIM have a “stealth education” agenda? Are there any commercial game titles available today that you are aware of, that implement this concept - especially related to physics?

**KR:** Thinking about it, yes it did have a “stealth” element within the limits of the processing power available at the time. I put in the gravity and air density sliders specifically so people could see how they affected part physics. I didn’t put them in for the game play they added and if I remember correctly there are only a few built-in puzzles that aren’t default Earth-like values.

**AS:** How would you judge the commercial potential for “stealth educational” games for the gaming industry of today? How did, for example, “Bridge Construction Set” sell?

**KR:** I loved “Bridge Construction Set”, but I don’t know how it sold. I know that Jeff Tunnell was very impressed with it when it first came out (he emailed me a link to its download site) and he now has it for sale off of the Garage Games site. If I had more free time I’d do a game very much like this with better graphics and a more global story of some sort to tie things together. I suspect a lot of people had fun watching their bridges collapse.

**AS:** As an historical example of a game with some teaching potential, the early Atari

game “Lunar Lander” comes to mind. Would you consider such a game “educational”? If yes, in which way? If no, why not?

**KR:** Yes. Gravity. Acceleration. Momentum. Many quarters from me when I was young. I think Gravitar had the same elements. Even more quarters from me.

**AS:** Do you have any examples of games that try, but fail in educating? What are the reasons for the failure?

**KR:** I can’t think of any specific examples, but I remember their being a large amount of “eduware” from the 90s that fail. Probably because they were not entertaining. You don’t have to sacrifice reality to be entertaining just like you don’t have to sacrifice entertaining to be realistic.

**AS:** Now, I’d like to broaden the topic to some more philosophical ideas in game design.

Game developers today recognize the importance of proper physics simulations for interactivity when creating an immersive virtual world. How would you rate the importance of game physics versus graphics & sound, social constructs & story line, or other gaming elements for creating this “immersion”?

**KR:** I think that it really depends upon the genre of the game. I have been trying lots of different casual games recently on Real Arcade. These games seems to be mostly about pretty graphics and sound with a story line tacked on. Game play is good, but very simple. People get hooked on these things so they must be immersive in some way.

On the other hand when I first had TIM running it was just polygon vectors moving and interacting on the screen. And underneath it that is still all it really is. In TIM the physics really is the game play - aside from the fish tank, cat, etc. type parts I put in there to give it a little more personality.

**AS:** Assuming we had a game that implements some basic laws of physics for its game play - say a submarine shooter. What would you think about a “reality” slider for the game that would allow smooth control from “arcade physics” to “real physics” similar to adjusting the sound volume? Would players use it and would it teach them something about the world?

**KR:** Interestingly when I read your question above the first thing I thought of was “How would I implement that?” When I first start out on a project I am balancing out how long I can spend creating a game versus how long it would take to implement different ideas. When working on Chain Reaction I told Jeff that “Fun and Realistic” is best, but that “Fun and Not Realistic” is better than “Not Fun and Realistic”. This was in regards to always having gravity on in Chain Reaction, which was causing game play problems.

The slider you mention sounds like being able to slide between “Fun and Realistic” and

“Fun and Not Realistic”. I think it would only be used if it’s end result was allowing the players to select a quantitatively different type of entertainment; or if it ended being in practice a difficulty slider. We actually had a slider similar to this in “David Wolf: Secret Agent” that did things like adjust the flight model on the jet to make it easier to fly. But this game (1989) did not have anything like real physics in it so it was more a slider between “arcade” and “let the computer play the game for you”.

**AS:** Spatiality is the main characteristics of most computer games in that they are mainly concerned with the representation and negotiation of space. Do you agree with this assertion? Could this preoccupation of spatiality in games be an important reason, why only a limited number of physics-fields (mostly dynamics and optics) are used in modern computer games?

**KR:** I think I would mostly agree with that. Thinking back to the original text Adventure game that I use to play - I’d draw maps on paper to keep track of where I was. Even many of the old text games were spatial although some like Hammurabi weren’t.

I think that that is probably one of the reasons. Another reason would be that innovation is hard and also more risky. My experience with the bigger game publishers is that they are very risk adverse.

**AS:** One of the weakness of computers that is specifically apparent for games in their limited I/O capability - usually relegated to keyboard and mouse or a gamepad for input. Do you think that the design of the physical interfaces might be an interesting advance for computer games from an educational point of view?

**KR:** It would be, but from a practical point there has to be enough market penetration that game publishers develop games that use them. That has always been the rub in that people won’t buy novel hardware because nothing runs on it and developers won’t develop for it because no one has bought them.

**AS:** To make games “better” and more innovative, do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**KR:** Probably both, but I wonder how much either group is interested? And looking at history it always seems to be one dynamic individual who moves things forward or changes things in a significant way. Alexander bringing about the final decline of the long lasting Egyptian Dynasties or Julius Caesar bringing about the end of the Republic. Perhaps the game industry needs a Shakespeare of art and science all in one to make a big step forward type breakthrough for others to follow?

To finish off, I’d like to ask you a few questions that came up during my first session

of our PhD workgroup - I just wanted to hear how you would answer them (if you have anything to comment on these at all).

What is the advantage/difference of a physic games compared to the real physic experience (i.e. in an experiment)?

**KR:** Back in 1986 Damon Slye told me about a friend of his who if he crashed his airplane in Microsoft Flight Simulator was not going to fly it ever again. One of the nice things about the flight sims (especially the more recent ones) is that you can get a good sense of what is involved in flying a plane and you don't really die if you crash. Not dying is a pretty good advantage.

**AS:** Would it be more interesting for a game developer to use physics laws to create new games in which specific laws would have the main influence on the game? Or is it more interesting to improve existing games by incorporating exact physics phenomena into the game logic?

**KR:** There was a game that I was thinking of doing at one time and actually may still end up doing eventual whose working title was "Gravity Master". All interactions would be gravity. The screen would have little planets on it that were all attracting the others. Every so often (5 seconds or so) a new planet would sent into the screen. The player's only interaction would be moving a star (or more massive object) around on the screen with his mouse. The player would be affecting all the other objects only by the gravity of the object that he was moving around. This would be interesting and fun to create, but I'm not sure if it would be fun to play or not.

## E.7 Liemandt Foundation

The Liemandt Foundation, a nonprofit group devoted to furthering education through technology, is running contests since 2003 to encourage college students and teams to develop educational games in the areas of middle school math and science. The contest focuses on the notion of "stealth education" in gaming, pushing students to create primarily entertaining games that also teach science and math topics such as forces, statistics, or the solar system. Students have complete freedom in their game designs. Advising the contest are experts such as Ultima creator and gaming legend Richard Garriott and educational game visionary Marc Prensky, and the project is affiliated with

the Digital Media Collaboratory at the University of Texas.<sup>7</sup>

The interview with Liemandt.org (LO) was conducted by e-mail from 2 April - 31 May 2006 by Andreas Schiffler (AS).

**AS:** I'd like to know a bit more how the Liemandt Foundation came into being. Who founded it and when did that occur? What was the primary focus for the foundation at the onset? Has this changed over the years? Does the foundation plan to continue the contests and what other future plans does it embark on?

**LO:** The Foundation started as a small private family nonprofit about 25 years ago. Greg Liemandt had made money in software and wanted to teach his kids about philanthropy. Every year, Greg, his wife Diane, and their 3 kids would sit down and read grants and decide where to give donations.

Greg passed about 10 years ago. His wife, Diane, runs the foundation and has remarried to Ron Reimann, who is also very involved. One son, Joe Liemandt, has also made considerable money in his own software company and has guided the foundation more/donated more in the past 5 years.

About 4 years ago, Joe convinced his mother, Ron & siblings that the foundation should stop dividing up funds into little donations here and there - and instead sink a large investment into one project that could have a bigger impact. They wanted to focus on educational video games, as theyd all been interested in technology-enabled education for some time. They hired me to help them figure out how to move forward the educational video game space and I came up with the Hidden Agenda project.

With Hidden Agenda, the foundation plans to continue investing in the space. By the fall, we will launch a free website to give kids access to the educational games that have been built through our contest.

**AS:** How successful was the foundation so far in creating educational games? Can you give a rough breakdown of the science fields the various contest winners touched? How important is physics as a field?

**LO:** Weve pulled together great games for Physics, Chemistry, Physical Health, and Math so far. See [www.hiddenagenda.com](http://www.hiddenagenda.com) to check the teaching subjects that students have focused on up to now.

**AS:** Once of the core ideas of the foundation is "stealth education". How did this term

---

<sup>7</sup>References: Liemandt Foundation (2006) Collegiate Presswire (2003) Jacobs (2004) Chen (2005)

evolve and is there any scientific background (papers, books, websites) that you can point me to regarding this type of education?

**LO:** Im trying to remember - I think I may have made this term up but I may have gotten it from somewhere. I spent quite a bit of time talking with Marc Prensky and reading his book, "Digital Game-Based Learning" while coming up with the project.

**AS:** The contest rule states, that games will then be judged based on the following weighted criteria - 70% entertainment, 30% education. What is the reason for this ratio? Would a game still be considered "stealth education" if it were only 10% educational?

**LO:** Games have to be fun if kids are going to play them, and ultimately learn from them. The most educational game out there is useless if kids dont think its fun.

**AS:** Are there any commercial game titles available that implement "stealth education" well? What about some historical example - the early Atari game "Lunar Lander" comes to mind? Do you have any examples of games that fail in educating? What are the reasons for the failure? (Here I'd be especially interested in game that incorporate physics.)

**LO:** Someone like Marc Prensky, Ben Sawyer, Jim Gee or others who have studied this field extensively would be able to answer this question better. Im not an expert in the field - just someone trying to help a foundation make a big positive impact.

**AS:** How do you judge the commercial potential for "stealth educational" games for the gaming industry? While the foundation has created and published several "winning" game designs, what else could be done to further the mission on improving commercial gaming design with "stealth education" in mind?

**LO:** One thing Im excited about is that many of the students who have competed in the contest are now graduating and joining big commercial game companies. Most of them have expressed an interest in instilling the "stealth education" notion into the commercial games theyll make in the future. Im also proud to have Richard "Lord British" Garriott as an advisor and contest judge. I hope and believe that the project has helped him think even more proactively about adding education to his games at NCSoft.

**AS:** The following set of questions might best be answered by a "judge" of content entries with direct experience in gaming and game design.

Assuming we had a game that implements some basic laws of physics for its gameplay - say a submarine shooter. What would you think about a "reality" slider for the game that would allow smooth control from "arcade physics" to "real physics" similar to adjusting the sound volume? Would players use it and would it teach them something?

**LO:** We are currently building a great arcade-like game with a real physics engine. I see no reason to muddy up the learning process by adding "arcade physics." The game,



called “Waste of Space” will be available in September. I can send you a link to it if you promise not to distribute it without my permission.

**AS:** Spatiality is the main characteristics of most computer games in that they are mainly concerned with the representation and negotiation of space. Do you agree with this assertion? Could this be the reason for the limited number of physics areas (mostly dynamics and optics) used in modern computer games?

**LO:** Perhaps. Not an expert in games or physics so am not the best person to answer this.

**AS:** One of the weaknesses of computers that is specifically apparent for games is their limited I/O capability - usually relegated to keyboard and mouse or a gamepad for input. Do you think that the design of the physical interfaces might be an interesting advance for computer games from an educational point of view?

**LO:** I don't think they are necessary. Some of the best games ever have had the simplest controls.

**AS:** To make games “better” and more educational, do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**LO:** Perhaps some of both. As far as the contest has gone, some of the best games have come from collaboration among students of both. Then again, some one-person teams with terrible graphical talent have made wonderful games that we've been able to clean up with the help of graphic artists. In the end, I see the artistic element as something we can add later.

**AS:** It is fair to say that computer games ARE very influential in our society (i.e. in terms of money, the game industry is bigger than film and music together). Is there an effect on society? Is that a positive or negative effect? Are games in their effect on society similar to - say - movies and can they distort the user's view on reality?

**LO:** Plenty of books have been written about this, claiming both. I do think that games and movies have some similarities - with the main difference being that people can actually participate in games.

**AS:** To finish off, I'd like to ask you a few questions that came up during my first session of our PhD workgroup - I just wanted to hear how you would answer them (if you have anything to comment on them).

Folk Physics is best suited to cope with the kinds of objects and events that we encounter on a day-to-day basis and is not really designed to provide accurate descriptions or explanations of the universe. Do you think game physics “teach” some form of folk physics to

the player? Could this have any negative side-effects?

**LO:** I dont think there will be any worse side effects than there are on the “folk physics” that are taught to kids in school at a younger age.

**AS:** What is the advantage/difference of a physic games compared to the real physic experience (i.e. in an experiment)?

**LO:** I think both are necessary, helpful and fun. But if kids dont have access to real physics experiments, games could be a good replacement.

**AS:** Would it be more interesting for a game developer to use physics laws to create new games in which specific laws would have the main influence on the game? Or is it more interesting to improve existing games by incorporating exact physics phenomena into the game logic?

**LO:** Im sure there is room for both, but Id say that people will be more interested in new games than on rework of old stuff. And interest, at this point, is the most important thing!

## E.8 Matthew Wegner

Matthew Wegner is a development director for Flashbang Studios and does casual computer game development. He started the website fun-motion.com, a site about physics games where he reviews games and collects information about Physics in games. The current review list sits at 35+ games (as of May 2006) and the site also contains many links and comments on the topic of physics in games.<sup>footnote</sup>References: Wegner (2006)

The interview with Matthew Wegner (MW) was conducted by e-mail from 28 May - 22 July 2006 by Andreas Schiffler (AS).

**AS:** Before I start to get into the specific details, I’d like to know a bit more about your background as programmer and game developer. What did you study and where did you get your initial work experiences as game developer?

**MW:** Despite the fact that I basically do programming today, I actually went to art school (Cogswell College in Sunnyvale, CA). I ended up leaving before I finished my degree, but I did maintain close relationships with a lot of my college friends and started up my company with them 3 and a half years ago. I dont have any formal programming

or computer science training.

My initial work experience as a game developer, in the professional sense, was actually when I co-founded Flashbang Studios. We currently develop for the “casual games” market, although our goal is to eventually fund games more along the lines of the “indie” aesthetic. You know, titles that are innovative for innovations sake, viable markets be damned.

**AS:** You mention, that you are driven by a “passion for physics-games” on the website. Did you study any physics and what was the most interesting aspect of your physics education that you remember? What about today’s physics research - anything on the radar that you are really interested in?

**MW:** My last formal physics education was in high school (AP Physics in my junior year, if I recall correctly). What I remember most is the unavailable answers for such seemingly simple questions as the root cause of inertia or gravity. Complicated formulas for describing motion are all well and good, but I took issue with the notion of terming such things as “laws” of physics. Theyre merely descriptions, not the underlying mechanisms. There was a lot of falling back to, “because thats the way things are”.

To be honest I dont keep a very close ear to the ground for modern physics research. Im dimly aware of the directions that things like quantum mechanics are taking, but the psychological ramifications of an observer-centric world are more interesting to me than the underlying mathematical ramifications.

**AS:** You have been involved in the gaming industry for a while, I presume. In very general terms, what do you think of todays games and game platforms? Where are we headed?

**MW:** Im actually fairly young (26), so I dont have the depth of experience of a lot of my peers in the industry. I have played games all my life, though, and tried to become more aware of the processes behind their development when I was younger.

Professionally Ive been focused on casual games for the past three years, so that gives me a much different perspective from a retail game developer. Its kind of depressing to see how retail development is still focusing on the same market segments and genres of games as it did ten years ago. But then again the casual market has emerged with the same fixation on standard genres, so perhaps thats the inevitability of the dollar.

Platform-wise I think games will diversify from 50-hour, high-production experiences to a wider range of experiences and time investments. Its already happening now, but I think the trend will continue. Its the difference between eating once a day and snacking continuously.

Culturally the notion of “gaming” and “gamers” will dilute to the point where games are viewed in the same way that mainstream entertainment is today. Particularly in the casual market, there is constant discourse identifying the “casual gamer” (typically the 35 year-old soccer mom). Other media dont distinguish like this-there isnt such a thing as “causal books” or “casual movies”. Everyone participates in those kinds of entertainment.

**AS:** How do you think mathematics, engineering and physics is perceived today by the general public? Is there a need to educate more in this area? If so, why?

**MW:** I think mathematics, engineering, physics, and other scientific disciplines are viewed by the general public as Black Magic. Seriously. I doubt the average American has a clue as to how their TV, microwave, or car works. The complexity of objects in our day-to-day lives dictates that, though. I would consider myself a fairly advanced computer user, but even then I dont know anything about designing silicon or the process involved in fabricating integrated circuits. Theres no way I can understand the operation of every device that allowed you to use a computer to email me this interview.

But I dont think the issue is one of knowledge. Theres simply too much information in the world today for any one personal to comprehend even a fraction of it. The issue is how well the general public understands the principles behind that knowledge (the scientific method, I suppose). Society is certainly lacking in that regard.

Is there a need to improve it, though? I guess it depends how you define need. It isnt required for society to function. A general improvement in critical thinking could result in increased efficiency across the board, I suppose, but I dont think its a requirement. Thats the nature of technology. We dont necessarily need to understand something to benefit from it.

**AS:** Now, I’d like to ask some questions related to your website fun-motion.com which features the extensive analysis of physics (or its simulation) in games as part of a selection and review process.

With what game did you get started on the website (i.e. your first review)?

**MW:** Well, the first game I actually posted on the site was one of my own projects: <http://www.fun-motion.com/physics-games/amoeball/>

The game that inspired me to create the site in the first place, though, was Ski Stunt Simulator (<http://www.fun-motion.com/physics-games/ski-stunt-simulator/> and <http://www.fun-motion.com/physics-games/ski-stunt-simulator-extreme/> ). I found the game immensely fascinating and wanted to share it with others (and discover more like it).

**AS:** How many games did you review so far and how many games are in the “review-queue” at the moment? What’s the best game you’ve seen so far?

**MW:** Currently the site has 42 game reviews. I have another 22 games to cover in my notes, although I'm discovering new ones all the time. I'm planning a "classic physics game" series, too, which will add to the total. I expect the site to easily hit 100 reviews.

**AS:** How do you approach a review and what do you look for in terms of the game-physics or simulation aspect? Do you ever "measure" the accuracy of the physics (i.e. by analyzing screenshots) to verify the implementation?

**MW:** My reviews have actually evolved over the months, at least in terms of length and formatting. The later reviews are all pretty standardized. In terms of content I try to avoid the trap of merely describing the game. It's difficult to cogently analyze something to provide an insight. For instance, in the recent "The Blob" game review I focused on the disconnect between the underlying physics simulation and the player's mental model of that simulation: <http://www.fun-motion.com/physics-games/de-blob/>

I do talk about the fidelity of the physics simulation, although not necessarily to "verify" the real-world accuracy of its implementation. Real-world simulation isn't desirable in many games. We are talking about entertainment, after all. Something like FlatOut 2 is deliberately tuned to be "arcade" like in its controls (<http://www.fun-motion.com/physics-games/flatout-2/>). It's impossible to control a car at 140mph through narrow dirt tracks, but FlatOut allows that experience, and they do it in such a way as to only just barely enable the player to pull it off. The deliberately inaccurate physics make the player feel like they're a much better driver than they really would be in real-world circumstances.

Other games have less obvious parallels to real-world physics. Consider Gish (<http://www.fun-motion.com/physics-games/gish/>). It behaves in a physical way, but without any direct real-world counterpart. Scale is completely ambiguous, as are the materials of his world. He's roughly gelatinous, sure, but without real-world reference points it's hard to judge the accuracy of its physics. I think games like Gish that utilize our understanding of how real physical objects behave, but through fantastic settings, have the most promise in terms of physics-based game design. It allows more suspension of disbelief, too. You can't say that Gish is behaving improperly, but you could say that a car is behaving improperly.

**AS:** People can suggest games for you to review. Do you have many rejects from these suggestions? Assuming you had some "rejects", can you give me an example title and explain why you didn't regard it as "physics game"?

**MW:** My definition of a physics game is fairly discretionary. Most video games have spatial and physical elements to them. For instance, the original Super Mario Brothers has highly abstract implementations of collisions, inertia, and gravity. Very few games are solely about raw numerical relationships without some element of spatial relationships.

So my definition of a physics game is, “a game where the player primarily interacts with the mechanics of a complex physics system”. What I typically focus on is how complex the physics system is. I’ve had a lot of suggestions for games that are too simplistic in their physics implementation, or simply too uninteresting (i.e. a standard racing title).

And to be honest I let my personal bias dictate a lot of my decisions. I would rather focus on an independently-produced unknown title than a big-budget flight simulator. I respect risky innovation that isn’t necessarily out there to make a buck.

Sometimes I just don’t understand the appeal, though. *Elasto Mania* is a hugely popular game. I get a constant stream of emails asking me if I’ve heard of it or why I haven’t reviewed it. I just don’t understand what’s so fun about it. The physics are very abstracted in what I feel is a very undesirable way. I need to review it at some point to better understand why the damn thing is so popular, but I can’t bring myself to play it for more than a few minutes.

**AS:** You categorize physics-games into games that focus on “Real-World Activities” and ones that implement “Abstracted Activities”. Could you provide an approximate ratio for these two categories based on the games you reviewed.

**MW:** It’s kind of a sliding scale between real-world and totally abstracted, but I would say my reviews feature both about equally. I guess it depends how you define realistic, real-world activities. There are games that have stricter adherence to physical properties but in an abstract environment (take *Switchball*: <http://www.fun-motion.com/physics-games/switchball/>). It’s abstracted, because you can’t really control a ball like that in real life. If you could, though, that’s how it would behave. There is a definite sense of scale and material.

**AS:** Do you have any stories, examples or even some source code that illustrates how difficult it can be to implement a proper physics simulation?

**MW:** I guess it depends if you’re talking about implementing the physics simulation itself or if you’re talking about implementing a game on top of that simulation. There are dozens of papers on physics simulation available, as well as some very robust pre-built engines. Personally I have no interest in the mathematics of physics simulation, though.

As for implementing a game, it’s always a tricky balance between enabling the player to manifest their intentions in the game and adhering to the guidelines or reality. Nobody wants a game that mimics reality 100%. That’s what reality is for.

I have a good example. I wanted to implement a jetpack physics test with a standard two thumb stick game pad. In my mind, the left thumb stick controlled the left thruster, and the right thumb stick controlled the right thruster. This combination would allow any

degree of movement. It took about an hour to rig up, and I was excited to try it out. In my mind I was performing amazing stunts with the greatest of ease.

What happened, though, was that as soon as I took off the jetpack invariably spun out of control. I could hover for a few seconds, maybe, if I was very careful to maintain my thrusters direction. It was just impossible to fly around with any degree of control.

And that makes sense, really. If you were to build a real jetpack with two rotating nozzles as thrusters it would be absolutely un-flyable. You'd kill yourself. The solution in my physics test was to apply some arbitrary impulses and torques to my jetpack to artificially force it upright. I could tweak these numbers to gradually remove stability from the point where it felt "cheap" to the point where it was fun and player skill was required.

Modern aviation does the same thing, really. Most jets, especially aerodynamically-unsound stealth bombers, have fly-by-wire systems. The computer does most of the real adjustment; the pilot loses 1-to-1 control of the system. In games its just a lot easier to apply arbitrary forces than it is to calculate what the stabilizing forces should be (and the players mental model of both systems will be identical, so why waste time).

**AS:** The game "The Incredible Machine" (TIM) had an engine that "does not use a random number generator in its simulation of physics, assuring that the results for any given 'machine' are reproducible". Why do you think it was designed that way? What about other games that you have reviewed - what percentage would you say, can simulate reproducibly?

**MW:** I would venture to say that it was designed that way. A deterministic physics system is required for some genres. Bridge Builder and Armadillo Run are good examples. The physics for any particular level or solution will play out the same way every time. Its a requirement of the games puzzle-oriented design. This allows players to compete fairly against each other-your solution will perform the same on someone elses computer. Optimization is no longer about chance but about changing your structural design.

Other games use deterministic physics in order to synchronize multiplayer experiences.

Im not sure what the percentages would be. Any single-player game that uses physics predominately for visual effects is probably not deterministic. Its impossible to tell with highly-interactive systems anyway (particularly when time is involved, since the player hitting a button 3ms later will have a different outcome).

**AS:** Could any of the games you have reviewed be used to perform real science? If so, do you have any suggestions or examples of how this might be done? Was there any feedback regarding your website from outside the gaming community (i.e. schools, universities or

artists for example)?

**MW:** I don't think many of the titles I review have much capacity outside the scope of entertainment. Some do, though. A game like Bridge Builder could certainly teach someone about the basics of structural integrity, and it's certainly less messy than gluing a bunch of toothpicks together.

As for "real" science in the sense of cutting-edge progress, I really doubt it. I would imagine the computation required for truly meaningful simulation is outside the scope of real-time environments. It's my understanding that using computational models for experiments is becoming very predominant in science today. I just don't think the level of simulation in games is useful for anything except psychological experiments.

**AS:** In your FAQ, you state: "My personal definition of a physics game is a game where the player primarily interacts with the mechanics of a complex physics system." This focus on physics as dynamics is also reflected in your site's domain name ("motion"). This is understandable, since most games' notion of physics IS dynamics. Can you give any examples of games which are not centered around dynamics/motion?

**MW:** Some of the games I've reviewed are more about structural engineering. Armadillo Run and Bridge Builder are still simulated using mass-spring systems, but the player is internally digesting the tension of the springs as structural integrity of the material.

<http://www.fun-motion.com/physics-games/bridge-builder/> <http://www.fun-motion.com/physics-games/armadillo-run/>

Operation Cleaner 2 is also more focused on the properties of materials at rest:

<http://www.fun-motion.com/physics-games/operation-cleaner-2/>

That's probably not quite what you're looking for, though, since all three of those games still make heavy use of motion to communicate their physical status. I can't think of any games that primarily focus on something else.

**AS:** If you were given the task to implement elements into a game from other fields of physics such as nuclear-physics or electrodynamics, what elements would you add? Which field of physics do you think is the hardest to use as a game element?

**MW:** Any element of physics that operates on a significantly different scale or mechanism than physical reality as experienced by humans is probably difficult. I think astrophysics would be easy to translate into game form, as an exception. Anything microscopic is probably much more abstract. Particle physics would probably be the most difficult to utilize in a meaningful way.

**AS:** Let me point your attention briefly to the recent development in the gaming industry



to incorporate physics acceleration into games (i.e. AGEIA).

**MW:** Have you seen any of the current crop of “physics-accelerated” games? What do you think of them? Do you think these devices help the game designers in creating better games? If you think they do, how do they do that and or why are the games better?

I have seen the games, although I havent had the chance to actually sit down with PhysX hardware and play the current titles (Im on a list with AGEIA to get review hardware, although Im not sure they take me seriously). I think hardware-accelerated physics is a promising development. Years ago I had thought there was an opportunity for such a thing (hell, I remember when adding dedicated hardware like a SoundBlaster was a weird thing).

Better games? Thats a very broad statement. Its like asking if a new paintbrush will help artists paint better pictures. I think theres an opportunity there for designers to make their worlds more physically coherent. Rather than worlds where some objects are physically interactive, and some arent, theres the possibility of creating worlds where everything is physical. Thats an exciting jump.

Creating more realistic worlds isnt a viable goal in itself, though. Lets say we get the point where were simulating reality to an indistinguishable degree. What then, though? Wed still need to build a game on top of it.

Manipulating perception is already more important than simulation at this point. If the goal of a game designer is to create the experience of racing a car, you dont necessarily need to calculate wind shear (although there are games that attempt to do this). Shaking the camera as a car whips past is good enough. The player will assume the camera shake was wind blast; you dont need to simulate the fluid dynamics of the surrounding air. And, really, if you did simulate it-and the effect was identical to a cheap hack-then its pointless. It doesnt matter if the numbers inside a computer are correct or not. It matters if the experience inside a players head is correct.

In terms of practicality, I dont think the current crop of hardware-accelerated physics will mean much for actual game design. Developers cant require the hardware yet, so current usage is visual embellishment. Until we reach a point where designers can safely target consumers that must have hardware accelerated physics to play the game it wont help design one bit.

**AS:** Theres a hot concept called “stealth education” that says its possible for people to play a game because they enjoy it and “accidentally” learn stuff along the way (i.e. Teaching not as obvious or targeted as in “edutainment”). Unfortunately, the concept has yet to prove its worth with the pre-teen and teen community. So, on this concept there

are a few questions:

Do any of the games you reviewed have a “stealth education” agenda and are there any commercial game titles available today that you are aware of, that implement this concept for physics?

**MW:** Some of the games Ive reviewed will enhance someones understanding of physics, sure. But in terms of the developer intending to do so with an agenda of education in mind? None that Im aware of. There are some websites that have little Flash games to illustrate physics concepts, but I dont know of any commercial, retail games with an educational agenda.

**AS:** How would you judge the commercial potential for “stealth educational” games for the gaming industry of today?

**MW:** I definitely think the potential is there. Games today are remarkably complex. Even something as seemingly trite as an action shooter requires the player learn and keep track a lot of very complicated concepts. The player is certainly capable of digesting advanced material.

I guess it depends what the educational payload for a stealth game is. Teaching world history would be easy (Civilization 4 with more accurate information and fixed time lines would work). If your payload was differential calculus, though, you might have a more difficult time. There are some small Flash games that have very specific political messages as payload that could be considered successful (<http://www.newsgaming.com/games/index12.htm> for an example)

**AS:** As an historical example of a game with some teaching potential, the early Atari game “Lunar Lander” comes to mind. Would you consider such a game “educational”? If yes, in which way? If no, why not?

**MW:** Sure, Id consider it educational. Games help players develop mentally, and sometimes teach them more about the world they live in. I guess it depends which denotation of “education” youre relying on.

**AS:** Do you have any examples of games that try, but fail in educating? What are the reasons for the failure?

**MW:** I dont have any examples offhand, but Ive seen some games that are too overt in their payloads and end up sounding preachy (particularly religious titles). I think stealth education games need to keep in mind that they should be entertaining games first and foremost.

**AS:** Now, Id like to broaden the topic to some more philosophical ideas in game design.

Game developers today recognize the importance of proper physics simulations for interactivity when creating an immersive virtual world. How would you rate the importance of game physics versus graphics & sound, social constructs & story line, or other gaming elements for creating this “immersion”?

**MW:** Physics are extremely important. Or, more specifically, the sensation of virtual kinesthesia is. A lot of time is spent tuning a control mechanic so that it simply “feels” better. Look at a lot of the old 8-bit Nintendo games. Mario doesn't come to an instant halt when the player lets off the button. He glides to a stop. Old racing games have similar simplified abstractions of concepts like inertia, gravity, and friction.

What's interesting is that two racing games (or two platform games, whatever) can have different qualities of control. One game could simply feel better than the second. And the difference isn't necessarily one of realism - adding more accurate simulation doesn't make a control feel better to the player. It's a subjective thing. In fact, sometimes dedication to realism isn't the right solution. It's a lot like traditional animation with squash and stretch and the like.

I think physics has made a stronger push towards more realism, though. Gamers today expect interactive avatar-based environments to have reasonable physics. They want crates to push around, explosions to impact nearby objects, and so on. I don't think you need flawless underlying simulation, though. To draw parallels with the push towards realism in graphics, you simply need to create the illusion and perception of something. A texture of grass substitutes each individual blade, and with physics simplified rigid bodies substitute complex destructible objects. In the player's head, though, they see real grass and real objects, though.

**AS:** Assuming we had a game that implements some basic laws of physics for its gameplay - say a submarine shooter. What would you think about a “reality” slider for the game that would allow smooth control from “arcade physics” to “real physics” similar to adjusting the sound volume? Would players use it and would it teach them something about the world?

**MW:** That's a good question, and I'm not sure how many players would use it. I guess it depends on the game and the incentives for using it. Is it entertaining to use? Are they rewarded appropriately with the increase in realism?

Some racing games do give the player the choice in fidelity of physics simulation. Usually it turns on/off forces that artificially help the car stay upright, turn easier without as much friction loss, and so on.

I think more realistic physics would teach them something about the world, but I also

think that simplified abstract physics teach the player something about the world. Players could compare the realistic physics in the game to the simplified physics in the game, but they're already doing those sorts of comparisons. Even simple physics are still viewed in the context of reality's physics. There's always that point of comparison; motion is something we're substantially attuned to.

**AS:** Spatiality is the main characteristic of most computer games in that they are mainly concerned with the representation and negotiation of space. Do you agree with this assertion? Could this preoccupation of spatiality in games (especially 3D games) be an important reason, why only a limited number of physics-fields (mostly dynamics and optics) are used in modern computer games?

**MW:** I think spatial relationships and dynamics are predominant in games because they are predominant in life. We are physical beings, after all. The nature of motion is remarkably intuitive to all animals, so I think it makes sense that it's a very easy thing to build inside simulations (as a random aside, we had chickens growing up and they would predictively track the parabolic paths of rocks or insects you threw to them even chickens understand dynamics).

Very few games make zero use of spatial relationships or dynamics. Even games that are largely about numbers have some spatial significance (a game like Civilization, devoid of dynamics, still makes heavy use of distances). As humans we like turning raw numbers into spatial relationships: pie charts, bar graphs, whatever.

So it is a preoccupation, I guess, but a natural one. If we were subatomic creatures I would suppose our creations would reflect that.

**AS:** One of the weaknesses of computers that is specifically apparent for games is their limited I/O capability - usually relegated to keyboard and mouse or a gamepad for input. Do you think that the design of the physical interfaces might be an interesting advance for computer games from an educational point of view?

**MW:** I think the Wii's controller is a good step towards more intuitively enabling physical controls. I'd love to fiddle with high-end haptic devices sometime, although I doubt the market will ever be able to support the cost of them for entertainment purposes.

**AS:** To make games "better" and more innovative, do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**MW:** Both? I do think that game developers need more education outside of learning trade skills. This isn't just related to scientific knowledge; I've seen many developers limit their creative influences to other games. It's a terrible way to get inspiration.

One way to make games more innovative is simply to have more people that haven't been making games or been thinking about making games their whole life - give it a try. I think technology will start to really enable this over the next few decades. Even now tools like Flash are enabling average people to crudely communicate in interactive media. Tools will be able to magnify their efforts so that the ability to express yourself interactively will be as commonplace as the ability to use your cell phone or start a blog is today.

**AS:** It is fair to say that computer games ARE very influential in our society (i.e. in terms of money, the game industry is bigger than film and music together). Is there an effect on society? Is that a positive or negative effect? Are games in their effect on society similar to - say - movies and can they distort the user's view on reality?

**MW:** Actually, game sales only surpass box office movie ticket sales. Movies still win if you consider rentals, DVD sales, and so on.

I think video games are as influential as any other media: books, movies, television, and so on.

Real-world human relationships don't operate the same in media as they do in the real world. But, I think the exposure to a wide variety of relationships (realistic, contrived, or otherwise) broadens our understanding of emotions in general. An interesting question is this - does reading/watching media about human relationships better equip me emotionally to deal with real relationships? Can a poorly-written movie still provide insight into my own relationships? I think games act in a similar way with regards to physics and the nature of reality.

As for positive or negative, I think the argument that video games are a complex form of entertainment, and therefore aid cognitive development, is a very valid one.

**AS:** To finish off, I'd like to ask you a few questions that came up during my first session of our PhD workgroup - I just wanted to hear how you would answer them (if you have anything to comment on these at all).

**MW:** Folk Physics is best suited to cope with the kinds of objects and events that we encounter on a day-to-day basis and is not really designed to provide accurate descriptions or explanations of the universe. Do you think it is possible that game physics will "teach" some form of folk physics to the players? Could this have any negative side-effects?

I think some games will actually help players unlearn folk physics. Take the Battlefield franchise (7 million units sold). The game has several physics models for various craft: airplanes, tanks, boats, etc. Players need to do things like take the speed of their vehicles into account when they fire missiles or drop bombs. Things don't fall straight down, a classic example of naive physics.

I guess it depends on the game, though. Some games will have a very simplified physics model, while others could teach such interesting concepts as planetary motion. Players are more likely to experiment inside the bounds of a games physics engine, too. They very quickly explore the possibility space. That kind of experimentation is beneficial to learning. I think humans tend to do the same kind of exploration as babies (chew on things, throw toys, generally figure out how physical reality operates), but as teenagers or adults we kind of give up on that. Games are a good outlet for continued experimentation.

**AS:** What is the advantage/difference of a physic games compared to the real physic experience (i.e. in an experiment)?

**MW:** The question implicitly addresses one major advantage. The artificial physics experience is malleable. That affords a lot of control in designing experiments, particularly those aimed at player behavior.

Physics games are probably just as well suited as reality for any experiments regarding human behavior and psychology. You could get someone drunk and have them play a racing game to learn about reaction time and drunk driving (without needing to put them on a real road with a real car).

You use the word experience, too. I dont think its that difficult to get the experience of reality and the experience of a physics game to match up. Users play a game in their head. They arent privy to the raw numbers inside the silicon. Rather, they take cues from the game to build their own mental model of how that world is operating. This model drives the experience, not the game. Oftentimes our mental models of a world are much more complex than the underlying simulation, so the experience is more identical to a real-world experience than the hard-and-fast simulation.

Its interesting that the concept of folk physics is the reverse of this phenomenon.

**AS:** Would it be more interesting for a game developer to use physics laws to create new games in which specific laws would have the main influence on the game (example: TIM)? Or is it more interesting to improve existing games by incorporating exact physics phenomena into the game logic (example: a modified Tetris game)?

**MW:** Triptych is a good example of a “physics-y” version of a classic genre. It plays a bit like a physical version of Tetris: <http://www.fun-motion.com/physics-games/triptych/>

I have thought that it would be an interesting experiment to craft new physical laws and then apply them to a game world. Define a basic concept like gravity in a new way, or mutate the rules of inertia (how about “an object in motion tends to accelerate?”). Honestly, though, I think it would be a tremendous failure. Players just wouldnt be able to relate. The success of dynamics in games is wholly due to the fact that we experience

those dynamics on a persistent, continuous basis.

## E.9 Max Behensky

Max Behensky worked as a programmer at Atari's arcade game division. Together with Rick Moncrief (creator of titles including Asteroids, and Lunar Lander) he was responsible for the programming of one of the earliest sophisticated driving simulators called Hard Drivin'. This world's first driving simulation game featured a realistic car-dynamics simulation. Currently he is still involved with designing driving simulators - the latest design being the best commercial driving simulation called the SMS Nascar Simulator which involves extensive physics modeling.<sup>8</sup>

The interview with Max Behensky (MB) was conducted by e-mail from 1 May 7 August 2006 by Andreas Schiffler (AS).

**AS:** Before I start to get into the more specific details, I'd like to know a bit more about your background as engineer, programmer and game developer. What did you study and where did you get your initial work experiences?

**MB:** I got my Bachelors degree at the Massachusetts Institute of Technology. I started in 1975 and graduated in 1981. After a stint repairing computers while at school, I started at Logo Computer Systems in 1981, working on developing a Logo interpreter for the Apple II. With many other LCSi folks I moved to Ataris Cambridge Research lab in 1982.

**AS:** In your past, what was the most interesting aspect of physics that you remember? What about todays physics research anything on your radar that you are really interested in outside of your field of work?

**MB:** When I started really studying physics as a senior in high school and in college, I started to get excited by the ability to use mathematics to figure out what really happens in a given situation.

It doesnt relate to my work in video game simulation, but Ive been interested in Quantum Mechanics and its philosophical implications for a long time.

**AS:** Since you are still involved with the computer gaming industry today, in very general

---

<sup>8</sup>References: The International Arcade Museum (1995) Minsky et al. (1990) Polson Enterprises (1996) MachineDesign (2006) Twin Lanes Technologies (2000)

terms, what do you think of today's commercial games and game platforms?

**MB:** I can't stand playing most driving games because their vehicle models are so bad. I don't get the time to play games too much these days. I've spent thousands of hours playing Diablo over the Internet with a friend who moved 200 miles away, but that is just as much a form of virtual hanging out as game playing. I like playing the Zelda series with my daughter, and I've been playing Prince of Persia, The Sands of Time on the PC lately. Game designers are starting to apply at least an approximation of physics to human movement, and I think that is pretty interesting.

**AS:** How do you think mathematics, engineering and physics is perceived today by the general public? Is there a need to educate more in this area? If so, why?

**MB:** I think mathematics (specifically calculus) and physics are perceived by most people as way too hard to understand, and basically useless in real life. I think this is wrong on both counts. Newtonian mechanics evolved together with calculus, and a general appreciation of it is useful any time you want to get your car unstuck from the snow. Mathematics is a screwdriver you can use to take the universe apart, and is both exciting and powerful. We need to figure out how to teach these tools to both kids and adults in a way that makes them interesting and exciting.

On the homepage of Twin Lanes Technology Inc., the slogan reads We make Physics Fun!. I believe it for cars, but where is there still work to do? How about a Shuttle simulator, now that you can soon buy a scale-size model from NASA. ;-)

I have had much more luck making fun games by simulating things that people actually have experience with. People can't tell if you got a Shuttle simulator right, but anyone who drives can appreciate a good car model. Even kids have enough experience with wagons and bicycles and just the way things move in real life to appreciate a good vehicle model.

Getting human body physics right in games would be a major improvement. I did some work on applying conservation of angular momentum to a figure rigged onto a skeleton with mass for each bone. Using this figure, I made a trampoline bounce simulation that was startling in its fidelity. Using the game controls you could make the figure bend at the waist, move its arms, etc. while bouncing and rotating and it looked perfect.

**AS:** Now, I'd like to get a bit more technical and also ask some questions related to your previous work on the Atari project Hard Drivin' (HD) as well as your latest involvement with the SMS NASCAR simulator (NS).

**MB:** Doug Milliken was listed as a Test Driver in the HD credits, supposedly because Atari didn't want anyone to know what he really did and Hard Drivin' had to be as accurate as possible. Why this aura of secrecy and how did it happen, that the physics



got such an important role in the game?

Physics was the game. Rick Moncrief had the vision of creating a driving simulator for the training market with accurate physics. Doug Milliken and I had invented the force feedback steering wheel at Atari Cambridge Research in 1983 and showed it to Rick. When I moved out to California in 1984, Rick hired me to work on the project.

The vehicle model took me more than a year of full time work to develop. Dougs dad had done the pioneering work in the 1950s on applying the sort of physical analysis used in aircraft to car handling. Their company, Milliken Research, Inc. sold us a simple car model written in basic by the late Hugo Radt. Using their model as a reference, I wrote another model in C extended from 2 to 3 dimensions. I added an engine and drivetrain model, and pounded on it until I got it to run in 30 milliseconds on a dedicated physics processor.

The graphics hardware designed by Jed Margolin was state of the art for the day, but it had pretty low performance; if you were lucky you got 250 flat shaded polygons at a 20 hz rate. Even with extremely limited graphics the simulation was a lot of fun because the physics behaved more or less like a real car. The graphics were good enough to show you what the physics were doing.

In early 1988 it became clear that Atari didnt have the marketing resources to sell a driver training simulator, so we put very simple rules into the simulation (start with a limited time, reach the next checkpoint for more time) and added a track with some physically interesting objects (jumps, a loop, and some traffic). Stephanie Mott invented the championship lap, which allowed you to race with a pre-recorded copy of the best previous driver. Erik Durfey helped come up with the dashboard shift that modeled the way your head gets thrown around by the centripetal acceleration in a corner.

There was no real attempt at secrecy. Doug Milliken was, in fact, our test driver. At the time, while working for his father, he drove cars on skidpads and test tracks for most of the major automakers. He drove our simulated car for us on skidpads we created for the purpose, and helped verify the accuracy of the model.

**AS:** Car-physics means doing an accurate car model to mathematically describe the physics of how the parts of the car (engine, transmission, springs, shock absorbers, tires, etc.) react to each other, to the road and to the driver's inputs. From your experience with these kind of models, how difficult is it to get it right and how accurate is the final outcome when its compared to a real car?

**MB:** Accurate car physics is extremely difficult to do. The force versus position and velocity relationship between a tire and the road is much more complicated than (for

example) the forces on an airplane wing. The basic slip angle/friction circle calculation is not nearly enough. You have to handle all the weird special cases (like braking to a full stop), slipping the clutch or spinning the tires, etc. I always aspired to creating a real time model good enough to use in the car design process. The Hard Drivin model was not good enough, but the Twin Lanes Technologies model is getting there.

**AS:** Many books about game-physics emphasize the need to manually tune the simulation, so it feels right. In HD or NS, how was it tuned and was there any experimental data used in the process? What are the algorithmic-shortcuts and non-realistic tweaked elements in these games?

**MB:** I don't agree with this philosophy at all. If you model the system accurately enough you don't need to tune it. Put in the real parameters for a real car and if your model is good, and the car was fun to drive, then your simulation will be fun too.

The kinds of tweaking we did on all these games were the exact same sort of tweaks that Detroit uses to make their cars easier to drive (making them understeer so they are less likely to spin out, for example). This sort of thing is necessary because real race cars are so hard to drive at the limits and we couldn't give our players enough visual cues with our primitive graphics.

As for shortcuts, in Hard Drivin we didn't have enough processing power for 4 wheels, so we used a two wheel model with mathematical training wheels to prevent it from tipping over. In Race Drivin we added a floating point DSP co-processor and had enough power to do 4 wheels accurately at a 4 millisecond loop rate. The TLTI model has no shortcuts and runs with a 1 millisecond loop rate.

**AS:** NS is the best commercial driving simulator (as far as I know). Can you give me an idea how sophisticated the physics underlying the NS really is? (No need to reveal any trade-secrets, just some general idea lines of code or CPU% maybe.) How does it technically (i.e. 68K vs. P4) compare as to the games developed in the Atari years?

**MB:** Some of this is discussed above. All free body simulations just integrate  $F=ma$ , which is fairly simple. The hard part is, of course, figuring out what  $F$  is. The TLTI model used at NSMS has a proprietary tire model that takes real tire data and interpolates it in a physically based way to deal with all the possible operating points. Up until recently, the NSMS system had a dedicated PC just for the physical simulation of the car and control of the acceleration sled.

**AS:** A review of the NS states, that one of the design principle was to keep the driving experience as real as possible. To what extent does this realism come from the car-physics simulation as compared to the visuals, physics construction and the sound?

**MB:** I believe that the physics is, by far, the most important factor in the realism of the experience. From Hard Drivin we learned that good physics with bad graphics beats bad physics with good graphics every time. Force feedback game controls that give you physically based forces in the steering wheel and a force (not position) sensitive brake make a big difference as well. Engine sounds are difficult to do well (an attempt I made at simulating the acoustics of an engine didnt work out), but as long as you have the frequency/rpm relationship correct it gives you the cues you need. Having the correct doppler shift on external sounds is important as well.

**AS:** I'd like to know, how players of NS perceive the experience. Does it matter to them that it is a realistic, physics-based simulation, or could one have replaced it with a fake simulation? Do you know of any car simulators that get it really wrong?

**MB:** As Ive said above, I believe that realistic physics is essential, at least for something like driving that most people have some experience with. Most other game type driving simulations get it wrong, and I dont like driving them. One exception was GP Legends from Papyrus, which reminded me of the Hard Drivin model.

**AS:** Let me point your attention briefly to a recent development in the gaming industry. You have probably heard about the new Ageia physics accelerator cards (if not, head over to this link <http://physx.ageia.com/> and have a look). What do you think of the current crop of physics-accelerated games? And do you think these devices help the game designers in creating better games?

**MB:** I dont know much about these physics co-processors. I have in the past used dedicated processors for my physics, but machines are getting fast enough, with good vector floating point that this is unnecessary for car simulation, anyway. These processors seem to be good for handling large systems of colliding objects, which isnt a problem Ive cared much about in the things that Ive done.

**AS:** Theres a hot concept called stealth education that says its possible for people to play a game because they enjoy it and accidentally learn stuff along the way. Unfortunately, the concept has yet to prove its worth with the pre-teen and teen community. So, on this concept there are a few questions:

Are there any commercial game titles available that you are aware of, that implement some form of stealth education?

**MB:** I think that many types of computer games teach you valuable skills of one form or another. The Sim City genre of games is one example.

**AS:** How would you judge the commercial potential for stealth educational games for the gaming industry of today? Could a driving simulator be used to illustrate for example

the speed and effects of a car accident for new drivers?

**MB:** I know that our driving games have helped me and others to handle poor traction situations. Rick Moncrief tells the story of how, during the development of Hard Drivin he headed home on a slick road. He got on the gas too hard, got sideways, caught the skid, and was a few blocks down the street before he realized that anything unusual or disturbing had happened.

**AS:** As an historical example of a game with some teaching potential, the early Atari game Lunar Lander comes to mind. Would you consider such a game educational? If yes, in which way? If no, why not?

**MB:** Of course lunar lander was educational. I cant think of any better way to give someone an intuitive grasp of the relationship between force, acceleration, velocity, and position.

**AS:** Now, I'd like to broaden the topic to some more philosophical ideas in game design. Assuming we had a game that implements some basic laws of physics for its gameplay say a submarine shooter - what would you think about a reality slider for the game that would allow smooth control from arcade physics to real physics similar to adjusting the sound volume? Would players use it and would it teach them something about the world?

**MB:** I dont believe in such sliders. Make the game real, simulate something fun, and people will like it. Deal with difficulty the way engineers in the real world do; understand the basic engineering of whatever you are simulating well enough to make an easy version.

**AS:** Spatiality is the main characteristics of most computer games in that they are mainly concerned with the representation an negotiation of space. Do you agree with this assertion? Could this preoccupation of spatiality in games be an important reason, why only a limited number of physics-fields (mostly dynamics and optics) are used in modern computer games?

**MB:** Im not sure I agree with this assertion. Some games (such as Pac Man or Quake) are primarily about moving through space. Others have more to do with resource optimization (Sim City, god games). Others are just general problem solving (Myst, many role playing games). Dynamics is vital for most simulation games (Spacewar, Lunar Lander, Hard Drivin). You could probably make a fun game with electric and magnetic fields, but Im not sure enough people would understand what is going on to make it popular.

**AS:** One of the weakness of computers that is specifically apparent for home-entertainment games in their limited I/O capability usually relegated to keyboard and mouse or a gamepad for input. Having worked with physical interfaces, might they offer interesting advance for computer games? Could haptic-interfaces be used to convey physics? If so,

how?

**MB:** Doug Milliken, Peter Milliken and I developed the first (as far as I know) force feedback game joystick at Atari Cambridge Research in 1983 based on an idea by Marvin Minsky that it would be fun to feel the maze in PacMan. I have always felt that it is a shame that almost all haptic game controls are just used to shake the user a little while firing a gun. The physically based force in the steering wheel for all the simulations I have worked on is vital to the driving experience.

Ive always thought that it would be cool to make a physics world game with springs, masses, friction, dampers, etc. that you could feel with a force feedback joystick and connect together. This requires real time closed loop control that is difficult to do with the limited interfaces available under Windows, however.

**AS:** To make games better and more innovative, do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**MB:** It depends on the game. There are many highly creative and successful game designers that can do games that have nothing to do with science. Game design studios often employ dedicated physics engineers to support their game designers visions. There is starting to be a complete divide many game studios between design and engineering that wasnt there at all in the old days. That being said, I think it is good for game designers to have an appreciation for what is possible, both in the real world and with the constraints that game hardware and software places on them.

**AS:** It is fair to say that computer games ARE very influential in our society (i.e. in terms of money, the game industry is bigger than film and music together). Is there an effect on society? Is that a positive or negative effect? Are games in their effect on society similar to say movies and can they distort the users view on reality?

**MB:** I believe that computer games have a large effect on those who play them. The interactive aspects of these games, as well as the many hours that people spend playing them make a greater impact than film or music. I am convinced that playing violent games can distort childrens and adults view of the consequences of violence. I am also convinced that well designed games can be a positive force in the world.

**AS:** To finish off, I'd like to ask you a few questions that came up during my first session of our PhD workgroup I just wanted to hear how you would answer them (if you have anything to comment on these at all).

Usually game developer used results from scientific research such as algorithms in their works. How might games be used in scientific research? Do you have any examples?

**MB:** Dr. Cox, of the University of North Carolina, has been using custom versions of Hard Drivin and more modern simulations to explore the effects of impairment (such as low blood sugar in diabetics) on driving for many years.

**AS:** Folk Physics is best suited to cope with the kinds of objects and events that we encounter on a day-to-day basis and is not really designed to provide accurate descriptions or explanations of the universe. Do you think game physics might teach us some form of folk physics (i.e. the player learns game-folk-physics by playing)? Could this have any negative side-effects?

**MB:** Im not sure what you mean by Folk Physics. I certainly believe that good simulations of driving can make you a better driver, and bad simulations can make you worse.

**AS:** What is the advantage/difference of a physic game compared to the real physic experience (i.e. in an experiment)?

**MB:** It is much easier and quicker to make physical experiments in a virtual world than the real one.

**AS:** Would it be more interesting for a game developer to use physics laws to create new games in which specific laws would have the main influence on the game? Or is it more interesting to improve existing games by incorporating exact physics phenomena into the game logic?

**MB:** Both seem interesting to me. The day for games such as Lunar Lander or Spacewar may be past, but they were certainly cases where physical laws were the basis of the game. My own approach to game design is more the latter; Ive tried to find something that is fun in the real world and simulate it as accurately as possible.

**AS:** Thanks for your time in answering all these questions. If you happen to have a friend or colleague who might be interested in doing an interview on this topic of game design and game physics, please make an introduction for me.

## **E.10 Megan Fox**

Megan Fox is a games programmer/designer and musician and presently a Senior at The University of Colorado at Boulder, majoring in Mathematics with a minor in Japanese. Her last published project was Kasei, an action combat demo, designed to test the Elium 3D RPG game engine software in a controlled, completed environment. The game engine includes extensive physics integration (rag-doll, physically accurate player movement and force

application, fast-moving projectiles, etc) and environmental simulations.<sup>9</sup>

The interview with Megan Fox (MF) was conducted via e-mail from 20 June - 21 June 2006 by Andreas Schiffler (AS).

**AS:** Before I start to get into the more specific details, I'd like to know a bit more about your background as programmer and game developer. How did you get started and where did you get your initial work experiences as a game developer?

**MF:** The obsession started when I was 6, when I bet my grandfather I'd be a games developer someday. I got started by self-teaching, and began when I was er somewhere around 8-11, I think. Standard hello world and guess-the-number stupidity. Graduated from there to basic DOS graphical applications around 13, and began writing simple RPG engines / top-down scrolling space shooters / etc. I was particularly proud of my tile-based smooth scrolling engine. Dabbled a bit in sound, went from there on to 3D, wrote a few (terrible) 3D engines, and then started doing what you saw on that resume page.

I got my first professional experience recently. I was hired on near the beginning of the year with Idyllon, purely on the merit of my work on Elium/Kasei. I am "the" developer (were a small team) working on a prototype come next year, if we can pitch it and get more funding, well expand and do a full release.

**AS:** You are studying Mathematics. What is the driving force behind that and is there any relevance or connection between your studies and to your software developments? Do you want to continue working in game development?

**MF:** Couple of reasons:

1.) Mathematics is the underlying force in programming. Everything (creative) you do requires it, and even when you're not leveraging specific formulas or approaches, the method of thought and approach you learn as a mathematician remains constantly useful.

2.) I found I was learning nothing in Computer Science. That was my original major, before dropping out, and I simply wasn't learning anything new. The only courses that taught me anything were the math courses (of which there were many), so upon my return, I saw little reason to go back into CS.

I am a game developer first, programmer second, and mathematician third, so a better question is whether I would want to continue working in mathematics. Simple answer, no; The work in this field doesn't suit me, I greatly despise theory for the sake of theory with

---

<sup>9</sup>References: Pepper Games (2005) Fox (2005)

no immediate applicability.

“But” - the major is general, and will serve me well regardless of where I end up in life. Its a pure science, and pure sciences are always useful, even outside of professional existence. Most importantly, I find it interesting, fun and stimulating, qualities the CS program failed to inspire in me.

**AS:** I am sure you have also done some physics in the past - so, what was the most interesting aspect of your physics education that you remember? What about today's physics research - anything on the radar that you are really interested in outside of your field of work?

**MF:** I assume you mean real physics here, not game physics.

I greatly enjoyed the more physical side of physics (that is, as opposed to EM theory), though I enjoyed fiddling with EM. That said, what I enjoyed was more the engineering bits than the theoretical bits; I built myself a little gauss gun (coil gun / mass driver) for a project that impressed everyone, and it was my first foray into circuit design.

Beyond that, it turned into gibberish. I dont enjoy trying to understand field theory, I hate Gaussian approximations, etc. None of it clicked, and none of it struck me as immediately useful, and so I found myself incapable of understanding it beyond what was necessary to pass the exams.

**AS:** How do you think mathematics, engineering and physics is perceived today by the general public? Is there a need to educate more in this area? If so, why?

**MF:** Well, its a bit of a dual issue.

On the one hand, I would say that greater education is necessary. Not simply more books, but encouraging the students to understand the content of the books. The average Americans understanding of electrical power, for instance, is terrible, and I cant even begin to express how irritated I am that few grasp universal gravitation. I find these topics to be far more useful than knowing who was the third president or what the capital of Nevada is, yet all too commonly the priority is reversed.

On the other hand, far too often are people likely to utterly disregard anything not immediately proven by science. As a spiritual person (mind you, spiritual, not religious), I find this distressing, and see the greater majority of “pure scientists” as living hollow existences wherein they try very hard not to think about anything beyond their linear existence by way of diving into ever-more-involving hobbies and careers.

On the magically-created third hand of poor paragraph structure, however, far too many are both deeply religious AND deeply scientific. They disregard anything not immediately allowed by either their religion or their science. So I think my biggest problem is the basic



idiocy of the average human mind, and the common inability to think outside the box.

So perhaps I just wish that our school systems created thinkers rather than memorizers.

**AS:** You are learning Japanese and call yourself a Designer and Musician on your home-page. How do you balance the programming and math work with these artistic fields?

**MF:** I dont. Im an artist masquerading as a scientist. If I open my mouth too much near my technical associates, Im branded a flake. I find questions of philosophy and the basic purpose of sentient life to be far more useful pursuits than the quantification and qualification of the physical world, but trying to explain why I believe that to a scientist is more or less impossible. “That which is outside what they can measure” doesnt exist to them, in general.

Its as though I find myself in a foreign country, and find it appropriate to master the native tongue. Mastering science makes me a proper scientist no more than learning the language of Japan makes me Japanese, but it does make it that much easier to express myself coherently to those who speak the language.

Besides, its fun. If I werent actively learning, the next 80 years would be boring indeed.

**AS:** Now I'd like to ask some questions related to your career and work in the field of game programming and maybe get a bit more technical.

Do you experience any issues being a woman working in a field that is primarily populated by “male-geeks”? Do you think you bring a different perspective into game development? If so, what is that?

**MF:** Differing perspective, hmm. Yes and no.

The basic problem I have with the game industry has little or nothing to do with sexism, but is instead rooted entirely in “good ol boy”-ism. The mentality of the enthroned chief developer is to look down upon all those he deems as lesser, which is to say anyone that doesnt know exactly what he does, and to lord that difference over them. That many male developers immediately assume that a female developer is such a lesser individual is true, but it isnt the basic problem; A novice in the industry faces precisely the same difficulty.

I suppose I bring a somewhat different perspective into the field, a less hormonally driven perspective at least, but it really isnt so different from anyone elses. Im just less likely to throw a strumpet into a lead role, and would instead favor a strong female with an athletic build (Jade from “Beyond Good and Evil” or Alyx from “Half Life 2”, as opposed to whatever the females name was in the recent leather-themed Prince of Persia 2). I doubt my perspectives as a gamer female are significantly different from those of a male in a similar position; A male RPG enthusiast is likely to see eye-to-eye with me, and not understand an all-out-action FPS enthusiast in precisely the same fashion as myself.

**AS:** You seem to be trying to get ahead in the gaming industry. In very general terms, what do you think of today's games and game platforms? Where are we headed in your opinion?

**MF:** As usual, they're teetering. Companies like EA throw out visually impressive games that lack content, but as games are a visual medium they sell. Then the games fail to impress, but having seen those graphics raises the bar for the rest of the developers. And so the fight continues to make ever-impressive graphics while still retaining time for basic game design.

I find that RPGs like *Knights Of The Old Republic 2*, *Neverwinter Nights 2* and so on give hope for a bright future. *KOTOR2*, for instance, sold incredibly well, and was a solid game. There are plenty of other examples that don't immediately come to mind. Then there was *SWAT4*, a unique and well designed FPS, along with *F.E.A.R.*, *Sin* (released in indie fashion no less), and so on.

I wouldn't say games are going up or down, they're remaining right where they have been for the past few years, but with increasingly less of a focus on graphics power and more on other immersion techniques; Gameplay isn't the only focus, but it's at least become a concern again. That is to say, they remain somewhat in a slump, but are finally starting to pull themselves out of the "polygon OMG!" hole they dug for themselves.

**AS:** Your last published project *Kasei* involved the ODE physics engine. Did you work with some of the physics part of the engine? If so, what did you implement. Did you find anything ODE related that was impressive from a technical perspective? Any shortcomings?

**MF:** I didn't touch ODE internals, but I did work enough with it to identify shortcomings.

The primary and critical shortcoming is the lack of island disabling. This is where, when objects are "connected" via frictional forces, you may disable ALL connected bodies if even one body manages to auto-disable itself. The idea is that if they're all touching, and even one body manages to come to rest, then it's likely that the rest should rest as well.

A developer implemented this into ODE for *Bloodrayne 2*, and it brought performance up to look more like a competitive commercial engine; At the very least, it enabled flawless stacking behavior. I intend to add this myself, as it's required by *Idyllon*.

The other shortcoming is the speed of their collision routines and the lack of a convex hull collider, but neither of those are easy to solve issues.

**AS:** Do you think a game could be used to perform real science? If so, do you have any suggestions or examples (which could be related to your studies in mathematics)?

**MF:** Not right now, no. A game (simulation) can be used to test initial hypothesis, in

certain basic cases, but accuracy issues will always be present in a way that prevents any sort of conclusive result.

That said, this is for the physical sciences - a game could be useful for behavioral and psychological experiments, just not for simulations of the physical world.

**AS:** Today's PCs can perform more complicated simulations. If you were to extend a game like Kasei with additional components and simulations from other fields of the sciences (i.e. biology, electrodynamics, etc.) what elements would you add?

**MF:** Artificial intelligence. It's the next most lacking area that needs to be focused on. Any research relating to the dynamic synthesis of dialog based on certain information immediately gets my attention, as does the subsequent conversion of the assumed textual output to convincing speech.

Beyond that, I'm interested primarily in convincing fakes of physical effects, not realistic representations. My growth system for Idyllon, for instance, is meant to resemble actual plant growth in output, but that which drives it is designed for fun factor and usability over accurate detail. I am concerned only with the convincing illusion of a plant growing from seed to full grown, not with the minute simulation of mineral transfer that determines leaf structure and health levels.

**AS:** Have you ever played games like "The Incredible Machine" , "Chain Reaction" or "Crazy Machines"? I'd like to get your feedback on these so called "Rube Goldberg Machine" based games which involve a fairly complex simulation of physics.

**MF:** They're extremely entertaining for the same reasons that recent physically-active games are entertaining. Games are finally becoming less about a linear song-and-dance, and more about a little mini world in which you can play god. It's incredibly fun to throw things around in a game just to see what happens, and that list of games tries to harness that fun into a focused objective.

I'm uncertain if this is more because it's a novelty, or because the basic activity is fun; Those games will remain entertaining for their puzzle aspect, but it's unknown if game physics will remain singularly entertaining or simply become common place.

Were I to speculate, physics allow for a world that is vastly more believable. Even after the initial blush wears off, they should remain a critical element for buy-in that helps you immerse more into the game environment.

So - the puzzle element will remain entertaining, but the physics appeal will likely die down as it becomes common-place for items in puzzle games of that nature to behave as you would expect vs as the game says they do.

**AS:** Let me point your attention briefly to a recent development in the gaming industry.

You have probably heard about the new Ageia physics accelerator cards (if not, head over to this link <http://physx.ageia.com/> and have a look) or similar developments using the shader hardware on the graphics cards. What do you think of the current crop of “physics-accelerated” games? Do you think these devices help the game designers in creating better games?

**MF:** They allow the designer to create more reactive worlds. Whether that ability is used for good or evil is speculation best left to Batman.

Theyre a tool, nothing more, much as 3D accelerators finally allowed the designer to put increasing complexity and believability into the presented scenes. They are a necessary creation, and the specific PhysX cards will die as the physics acceleration becomes a common feature on graphics accelerators. I speculate that the new term will become “game accelerator.”

**AS:** Theres a hot concept called “stealth education” that says its possible for people to play a game because they enjoy it and “accidentally” learn stuff along the way. Unfortunately, the concept has yet to prove its worth with the pre-teen and teen community. So, on this concept there are a few questions:

Did any of the games you worked on have any “stealth education” agenda? Are there commercial game titles available today that you are aware of, that implement this concept - especially related to physics?

**MF:** Not a one Ive worked on has had such a thing, nor am I likely to design for it. I am concerned only with the entertainment value of a thing, not its usefulness as a learning aid. In fact, you could say my obsession with creating believable and reactive worlds is destructive to learning; If I succeed in creating a believable world, not only does it train a mind to play by my simulated world logic, but it allows that mind to enact increasingly destructive acts simply to see how they play out. I dont believe this to be damaging to the mind, but many do.

For physics, however, I would suggest that any learning gained would be chiefly at the toddler age, possibly up to pre-teen. Its likely that a sufficiently realistic environment could assist with spatial reasoning and logic.

An unintentional example would be Big Rig Truckers (I believe thats the name), a game wherein one of the goals is to park your truck through increasingly complex courses within increasingly limited regions, using a perspective and interface meant to mimic sitting in the trucks cab. Ive heard more than a few people say it actually helped them to reverse-drive into parking spaces more quickly and effectively.

**AS:** How would you judge the commercial potential for “stealth educational” games for

the gaming industry of today? Would it be a viable game concept?

**MF:** Debatable. I learned a great deal from Oregon Trail as a kid, which I would classify as a proper game rather than edutainment. Its likely that, with increasingly accurate simulations, more and more practical knowledge can be imparted automatically.

That said, NPC simulation is still vastly simplified, and far too basic to teach proper social skills or interaction. So, your potential realms of education are somewhat limited.

Who knows though. Perhaps a great deal about compassion could be taught with a game that puts you in the position of one of the rescuers running into the building during 9/11. Not my bailiwick either way.

**AS:** As an historical example of a game with some teaching potential, the early Atari game “Lunar Lander” comes to mind. Would you consider such a game “educational”? If yes, in which way? If no, why not?

**MF:** I wouldnt consider it educational, assuming were both thinking of the same game. Wasnt that a game chiefly about jumping a car over bumps in the road?

**AS:** Do you have any examples of games that try, but fail in educating? What are the reasons for the failure?

No specific examples, because I dont play them or pay attention to them. Any game that fails does so because it attempts to put education before fun; If a game ceases to be fun, the player will simply find another game.

Most edutainment is this way. It attempts to educate by forcing certain fact recall or ability before the game can progress. Stealth education must be something that can be completely ignored in theory, but that remains very hard to completely tune out.

**AS:** Now, I’d like to broaden the topic to some more philosophical ideas in game design.

Game developers today recognize the importance of proper physics simulations for interactivity when creating an immersive virtual world. How would you rate the importance of game physics versus graphics & sound, social constructs & story line, or other gaming elements for creating this “immersion”?

**MF:** Convincing immersive virtual worlds must start with accurate and consistent rule sets. This is first and foremost graphics, then physics, then sound. Humans are primarily visual creatures, but remain physical beings used to interacting in a world constantly in motion. Sound is vital, but less so than the other two, and in sound the most vital aspect is consistency rather than realism; A world can be alien, uses entirely unknown sounds, and will still be believable so long as the sounds are generated consistent to the expected rules. In this regard, convincing sound is chiefly a matter of convincing physical simulation

tied to the sound engine.

Now that's just the world. You can have a convincing world with unconvincing NPCs. The two are separate, and the one doesn't force buy-in to the other; In fact, if the one is too realistic, then it throws into contrast how awful the other is, which is where we've gotten with physics showing up the stupid nature of AI.

Storyline is irrelevant to buy-in, so long as the drives of the characters are clear and sensible. In this, the buy-in for a story is less about the story, and more about the believability of the characters, throwing it back into the realm of social constructs. The story can be anything, so long as the actions of the NPCs aren't stilted or idiotic.

**AS:** Assuming we had a game that implements some basic laws of physics for its gameplay - say a submarine shooter. What would you think about a "reality" slider for the game that would allow smooth control from "arcade physics" to "real physics" similar to adjusting the sound volume? Would players use it and would it teach them something about the world?

**MF:** You'd be designing two games in one, a simulation and a shooter, which is useless. Most gamers enjoy one or the other, having both in one makes no sense.

Any education you'd gain from the game would be one-shot, when the player tested to see what "real" was. When they found they didn't enjoy it, they would slide it back, and that would be the end of it. Only simulation gamers would play with the realism slider to the max, and they don't need to be taught what realistic submarine movement is like.

If the goal is to teach someone what real submarine movement is like, then the game must be designed to be realistic AND FUN - and if the sim mode is fun, then the arcade mode need never be created. If real submarine movement can not be fun when compared to arcade, then any hope of stealth entertainment was sunk before it left the port. A slider that effectively pans between fun and boring won't teach anything, who's going to slider it over to boring?

**AS:** Spatiality is the main characteristic of most computer games in that they are mainly concerned with the representation and negotiation of space. Do you agree with this assertion? Could this preoccupation of spatiality in games be an important reason, why only a very limited number of physics-fields (mostly dynamics and optics) are used in modern computer games?

**MF:** Games are spatially concerned because it's one of the few things that can be done effectively in a game. Adventure games failed not because they were boring, but more because gamers weren't given the ability to solve a puzzle their way, only the pre-scripted way. Interaction games fail because AI remains far too stupid to be an interesting conver-

sational partner.

Movement and environment are solved problems, and the rest can be faked well enough to work within that context. Take it beyond that, and technology is still too far behind. Expect changes in this as AI technology advances; Consider what the simple application of physical world has done to games already.

**AS:** One of the weakness of computers that is specifically apparent for games in their limited I/O capability - usually relegated to keyboard and mouse or a gamepad for input. Do you think that the design of the physical interfaces might be an interesting advance for computer games from an educational point of view?

**MF:** Short of complete integration into the virtual environment via suit / “wet wire” / etc, I dont believe there are any useful advances in game input to be made. Changes can be made that make certain games more interesting or interactive, but nothing that works in a general sense.

Feel free to make me eat those words when Nintendo Revolution debuts, though.

**AS:** To make games “better” and more innovative, do you think artistic game developers should have more scientific knowledge? Or do computer science graduates working on games need more artistic training and exposure?

**MF:** CS graduates need more artistic exposure. No one I talk to has any understanding of what appropriate music or sound can do to make an otherwise uninteresting scene seem epic, for instance, and very few understand that a world can be believable without being “realistic” (thus the obsession with realistic graphics rather than hand drawn in appearance or mixtures of styles).

More so, designers simply need to be more artist and less developer. Too often, the guy or gal in charge is an engineer first and artist second, and so they design the numbers without thinking about how theyll be presented to the user.

**AS:** It is fair to say that computer games ARE very influential in our society (i.e. in terms of money, the game industry is bigger than film and music together). Is there an effect on society? Is that a positive or negative effect? Are games in their effect on society similar to - say - movies and can they distort the users view on reality?

**MF:** You loaded that question, careful.

Movies distort only when they present the user with unknown material thats been modified. As a very simple example, everyone refers to a pistols magazine as a “clip” thanks entirely to Hollywood, when the two things are entirely different. People also tend to expect explosions to be more impressive than they really are, and less noisy, because theyve never seen one outside a movie.

The same, however, can be said of any medium whatsoever. A romance novel that presents gay men as particularly creative would influence the views of someone that has never met a gay person, just as Steinbecks obsession with absinthe made quite a few think of it as a cool / artistic beverage (because he never mentioned its, ahem, “other” effects).

So, games are like movies, and movies are like books, and books are like magazines and everything else. People are easily influenced by anything and anyone that appears to know more about a subject than they themselves do. Never blame the media, always blame the idiotic that didnt have the mental capacity to learn for himself.

**AS:** To finish off, I'd like to ask you a few questions that came up during my first session of our PhD workgroup - I just wanted to hear how you would answer them (if you have anything to comment on these at all).

Folk Physics is best suited to cope with the kinds of objects and events that we encounter on a day-to-day basis and is not really designed to provide accurate descriptions or explanations of the universe. Do you think game physics “teach” some form of folk physics to the player? Could this have any negative side-effects?

**MF:** This ties into my last point. It can have negative effects only when the individual has never experienced the physical effect in question outside of the simulation.

So, just as you temper a youths experience in violent movies with real-world social interaction, you temper a youths video game box throwing by taking them outside and letting them throw rocks around the yard.

That said, they do teach a sort of folk physics that becomes expected. If friction in games is consistently different, then people will come to expect all games to have similar friction, even if its technically incorrect. Theres already a problem where people want AI to be stupider than it need be, in terms of its ability to hunt you down, because it stops being fun at a point; Such could happen with physics.

**AS:** What is the advantage/difference of a physic games compared to the real physics experience (i.e. in an experiment)?

**MF:** Numerical accuracy issues, and the ability to say you tested your results with more than a game. A simulations results are only as good as the simulation; Short of a direct, completely consistent and in-depth comparison of how a simulations results differ from reality, the simulations results are useless in a real-world context.

**AS:** Would it be more interesting for a game developer to use physics laws to create new games in which specific laws would have the main influence on the game? Or is it more interesting to improve existing games by incorporating exact physics phenomena into the game logic?



**MF:** Its more interesting to create new rules, and make the world play by them consistently. Humans interact in physical reality every day, and games are an escape from that; They dont necessarily want realistic physics any more than they want their virtual hero to be reduced to a screaming girly man by a single bullet impact anywhere in their body. They should only be as realistic as is fun.

**AS:** Thank you for the interview.

# Appendix **F**

## Surveys

### F.1 Game Physics and Video Game Players

#### Introduction

This survey is part of my PhD theses about "Game Physics in Video Games"; the research is briefly summarized on this page of my website: <http://www.ferzkopp.net/joomla/content/view/55/15/> - feel free to review it before starting the survey.

The survey questions are designed to get an insight into how game physics - the representation of physics and physical realities in video games - is perceived by YOU the video game player.

You will encounter 33 questions about video games and game physics. Some questions will be about your educational background in physics, your views about the sciences in general and your level of involvement with video game programming. To work through all the questions in earnest will probably take you about 20-30min - so if you are in a hurry right now, maybe come back later.



Everyone is encouraged to take this short survey, but the focus is on people who actively play video games. So if you are a gamer - even if it is just occasionally or if you played video games in your past - please consider answering these questions. If you never played a computer game in your life, some questions might not be relevant to you, but I encourage you to contribute your ideas and comments as well.

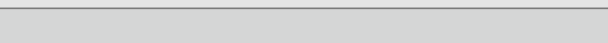




If some of the questions you encounter seem to be not applicable to you or if you don't

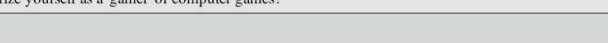



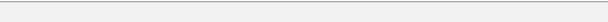
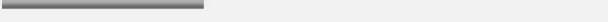
understand them, just skip the question and continue with the next one. At the end of the survey you can provide feedback.

Before you start, I want to assure you that this survey is purely for research and remains anonymous (i.e. I don't ask for your email or phone number). Please note, that some of your written comments might be quoted in a publication such as my thesis however. If you have any questions about this survey, the research, have suggestions or are interested in the results when they are published, feel free to write me an email: [aschiffler at ferzkopp.net].

Thank you very much for your time and assistance in my research! Andreas Schiffler

1. what's your gender?			
		Response Percent	Response Total
Male		96.3%	387
Female		3.7%	15
<b>Total Respondents</b>			<b>402</b>
(skipped this question)			0

2. which age group are you in?			
		Response Percent	Response Total
20 years or younger		45.3%	182
21 to 30 years old		36.1%	145
31 to 40 years old		14.7%	59
41 to 50 years old		2.7%	11
51 years or older		1.2%	5
<b>Total Respondents</b>			<b>402</b>
(skipped this question)			0

3. How would you categorize yourself as a gamer of computer games?			
		Response Percent	Response Total
I am/was an active, 'hardcore' gamer who is/was involved with and very knowledgeable about the gaming scene.		56.8%	189
I am/was a regular game player, but don't consider myself a dedicated 'gamer'.		32.7%	109
I am/was an occasional game player who plays for mostly recreational purposes when there is time.		9.3%	31
I am/was a game player that plays video games rarely; and mostly by chance.		0.9%	3
I am/was never a video game player myself, but have seen a few video games.		0%	0
Other (please specify)		0.3%	1
<b>Total Respondents</b>			<b>333</b>
(skipped this question)			69

4. How much video game playing do you do? (consider any interaction with a video game, even if it is on a cell phone)

		Response Percent	Response Total
Almost every day		71.6%	237
A few times a week		21.5%	71
Once a week		3.6%	12
A few times a month		2.7%	9
A few times a year		0.3%	1
Basically never		0.3%	1
		<b>Total Respondents</b>	<b>331</b>
		(skipped this question)	71

27. What kind of game do you (or did you) play? (select all that apply)		Response Percent	Respos Total
Early action games (i.e. arcade emulators)		49.7%	165
Maze games (i.e. PacMan, Bomberman)		40.4%	134
2D Platform games (i.e. Super Mario Bros.)		65.4%	217
Fixed Shooters (Galaga, Area 51)		33.7%	112
Slide shooters (i.e. Raiden)		37%	123
Competitive fighting games (i.e. Street Fighter II)		38.6%	128
Beat 'em Up fighting games (i.e. Double Dragon)		38%	126
3D Platform games (i.e. SuperMario 64)		60.8%	202
<b>First-Person shooters (i.e. Quake, Half-Life)</b>		<b>90.4%</b>	<b>300</b>
Third-Person shooters (i.e. Resident Evil)		67.2%	223
Survival Horror (i.e. Silent Hill)		28.3%	94
Turn based RPG and strategy games (i.e. The Elder Scrolls)		58.7%	195
Real-time RPG and strategy games (i.e. Age of Mythology)		67.2%	223
Massively Multiplayer Online Role-playing Games (i.e. Ultima Online, EVE)		44.3%	147
Text Adventures (i.e. Zork)		23.2%	77
2D Graphics Adventures (i.e. King's Quest)		38.9%	129
3D Interactive Adventures (i.e. Everquest)		42.2%	140
Flight Simulators (i.e. MS Flight Simulator)		34.3%	114
Space Simulators (i.e. Space Combat)		35.8%	119
Driving/Racing Simulators (i.e. GT Legends)		51.8%	172
Boat/Submarine Simulators (i.e. Silent Hunter)		13.6%	45
Sports Simulators (i.e. EA Sport s series)		28%	93
Building Simulators (i.e. Sim City)		52.1%	173
Fictional Life Simulators (i.e. The Sims)		36.4%	121
Business or Economic Simulation Games (i.e. Capitalism)		21.1%	70
Visual Matching Puzzle (i.e. Tetris)		45.8%	152
Hidden Object Puzzles (i.e. Minesweeper)		34.3%	114
Character Control Puzzles (i.e. Lemmings)		45.8%	152
Construction Puzzles (i.e. TIM, Crazy Machine)		37.7%	125
Child Education Games (i.e. Carmen Sandiego Series)		9.9%	33
'Serious Games' for Training/Advertising (i.e. America's Army)		19.6%	65
Programming Games (i.e. Core War, Robot Battle)		16.3%	54
Other (please specify)		8.4%	28
		<b>Total Respondents</b>	<b>332</b>
		(skipped this question)	70

9. Using the same list of game categories, pick the genre that you like best right now.		
	Response Percent	Respos Total
Early action games (i.e. arcade emulators)	0.3%	1
Maze games (i.e. PacMan, Bomberman)	0%	0
2D Platform games (i.e. Super Mario Bros.)	2.2%	7
Fixed Shooters (Galaga, Area 51)	0.9%	3
Slide shooters (i.e. Raiden)	0.6%	2
Competitive fighting games (i.e. Street Fighter II)	1.3%	4
Beat 'em Up fighting games (i.e. Double Dragon)	0.9%	3
3D Platform games (i.e. SuperMario 64)	2.5%	8
<b>First-Person shooters (i.e. Quake, Half-Life)</b>	<b>42.1%</b>	<b>133</b>
Third-Person shooters (i.e. Resident Evil)	3.2%	10
Survival Horror (i.e. Silent Hill)	0.9%	3
Turn based RPG and strategy games (i.e. The Elder Scrolls)	7.6%	24
Real-time RPG and strategy games (i.e. Age of Mythology)	13.9%	44
Massively Multiplayer Online Role-playing Games (i.e. Ultima Online, EVE)	8.5%	27
Text Adventures (i.e. Zork)	0%	0
2D Graphics Adventures (i.e. King's Quest)	1.9%	6
3D Interactive Adventures (i.e. Everquest)	3.5%	11
Flight Simulators (i.e. MS Flight Simulator)	0.3%	1
Space Simulators (i.e. Space Combat)	0%	0
Driving/Racing Simulators (i.e. GT Legends)	2.8%	9
Boat/Submarine Simulators (i.e. Silent Hunter)	0%	0
Sports Simulators (i.e. EA Sport s series)	0.9%	3
Building Simulators (i.e. Sim City)	0.9%	3
Fictional Life Simulators (i.e. The Sims)	0.6%	2
Business or Economic Simulation Games (i.e. Capitalism)	0.6%	2
Visual Matching Puzzle (i.e. Tetris)	0.3%	1
Hidden Object Puzzles (i.e. Minesweeper)	0%	0
Character Control Puzzles (i.e. Lemmings)	0%	0
Construction Puzzles (i.e. TIM, Crazy Machine)	1.6%	5
Child Education Games (i.e. Carmen Sandiego Series)	0%	0
'Serious Games' for Training/Advertising (i.e. America's Army)	0.6%	2
Programming Games (i.e. Core War, Robot Battle)	0.6%	2
<b>Total Respondents</b>		<b>316</b>
(skipped this question)		86

7. What gaming platform do/did you use to play your games? (check all that apply)			
		Response Percent	Response Total
Game Console (i.e. Xbox, PS2)		72.7%	240
Personal Computer (i.e. PC, Mac)		96.7%	319
Handheld Console (i.e. Gameboy, PSP)		51.2%	169
Other Portable Device (i.e. Cell Phone, iPod)		23%	76
Commercial Arcade Console		11.5%	38
Other (please specify)		3.6%	12
<b>Total Respondents</b>			<b>330</b>
(skipped this question)			72

8. Have you ever programmed a computer game (or part of a game) yourself?			
		Response Percent	Response Total
Yes		55.7%	186
No		44.3%	148
<b>Total Respondents</b>			<b>334</b>
(skipped this question)			68

9. Is there a gender bias towards men in video games?			
		Response Percent	Response Total
Yes		76.6%	255
No		23.4%	78
<b>Total Respondents</b>			<b>333</b>
(skipped this question)			69

10. Without surfing over to wikipedia.org, consulting a book or asking an expert, please answer in a one or two sentences: what do you think "game physics" is?			
<b>Total Respondents</b>			<b>235</b>
(skipped this question)			167

1. The simulation of objects interacting with each other. Games that use a physics engine will have more interactive environments because objects will react to being hit.
2. Game physics is the way in which any game uses gravity, collision detection, or friction to create a (sometimes) realistic effect.
3. The technology of the interaction of object elements in an environment.
4. Using real-world physics or demonstrations of real-world physics in game format for enjoyment.
5. game elements react in believable ways to the application of force. this is evaluated as force is applied, and not preprogrammed.
6. Game physics is the simulation and/or approximation (be it realistic or intentionally unrealistic) of real world actions and interactions inside a virtual setting such as a video game.
7. Game physics is the use of concepts like friction, velocity, or mass to influence the movement of objects.
8. a game wich gameplay is essentially based on physics phenomens
9. Is about all things that affects other objects and these objects react in some specified (by physics) way

10. Various game mechanics which are trying to simulate real life physics and give the game a more 'realistic' feeling.
11. To me, game physics is the interaction between any 2 elements in a game. From bullets ripping into someone and knocking the backwards, to hitting a barrel with a car and the barrel flying apart/away. The physics of a game is how objects interact.
12. i think it is the way things interact with other things an example is a ragdoll smashing against a wall or some rocks tumbling down a hill ( in a video game such as half-life 2)
13. A simulation of real world objects
14. The system controlling all motions within a game. This would include object/object interactions, player motions, player/object interactions, and player/player interactions (including online synchronizing and lag prediction)
15. Any virtual physics that models real world physics.
16. a realistic reaction of momentum on all objects in the semi-releastic world and reaction too the player or to its enviroment
17. Real life physics coded into the game script, such as gravity, constraints and ragdoll effects.
18. Computer animation physics or game physics involves the introduction of the laws of physics into a simulation or game engine, particularly in 3D computer graphics, for the purpose of making the effects appear more real to the observer. Typically simulation physics is only a close approximation to real physics, and computation is performed using discrete values.
19. Game physics is the ability to manipulate objects. Think Half - Life 1 physics compared to Half - Life 2 physics. item a game environment that uses real-world newton physics for objects and people so that a user can feel and interact more with the game environment or "person" such as Toribash.
20. The possibility for players to solve make their own solution for a problem using the known laws of physics.
21. Game Physics are, in short, the physics of a game.
22. game physics are all the codeing and design of a game that help it simulate real-life physics. This makes the game seem more life like if the game physics are well designed.
23. The equations and programming used when figuring out how objects and things should react in a realistic way in a digital environment.
24. Game physics is like ragdolls in FPS. Like when u bump into a table the plates fall off. Kinda like that in Oblivion.
25. Anything that allows objects to be dynamically moved whilst in-game.
26. Game physics is the physics that are utilized in games when the user interacts with the environment, and as a result of this interaction, an action occurs in the gaming world, such as a stack of boxes falling over when shot with a gun.
27. How well the game is displayed and how it runs... Ragdoll physics for an example.
28. physics made fun. objects reacting how they would in real-world situations.
29. The way in-game objects interact with a virtual environment.
30. Uh games that extensively use basic/advanced laws of physics in order to produce a realistic sort of feel to the game.
31. Using physics inside a game to determine how realistic movements are without using any pre-done animations.
32. A realistic, or sometimes exaggerated, depiction of real-world physics in a simulated environment.



Sometimes put in the game for added realism, other times to enhance the gameplay, and occasionally as the basis for gameplay itself.

33. Games that have physics as their main game function. They are usually realistic and rely heavily on the physics.
34. When a video game tries to simulate real life physics. Such as water, gravity, vehicle movement etc.
35. Simulation of real life physics in video games.
36. Game physics is modeling objects in a game to act like they do in real life. To some extent all games have physics. However, colloquially people generally refer to physics as Newtonian physics between objects and the play such as collision detection, gravity, inertia, etc.
37. game physics is in games to simulate real life more, i mean every body loves seeing a car realistically fly 20-feet after hitting it with a rocket and hitting your foe then have your foe drop like a ragdoll down the stairs. physics games is also a main base to games, like say....toribash.
38. real-time physics simulation used to enhance gameplay
39. It's when you have physics in a game.
40. a mathematical "engine" that creates the basic rules for all the interaction between virtual objects in real or non-real fashion, but maintaining coherence in the game flow.
41. Integration of simple laws into code to allow for more realistic and unexpected interaction between objects.
42. Anything that has to do with particle effects, object movements, collisions and no collision, ragdoll effects, advanced liquid physics, cloth physics.
43. I would tend to believe game physics would be the physics implemented into the game such as being able to cut a table in half or even just ragdoll physics.
44. The representation of Scientific proven real world dynamics in the game.
45. Game physics are simulations of real-life physics applied to a video game.
46. how objects react when they come in contact with other objects.
47. that is, all things that make the game look real on the physics side, like gravity friction and buoyancy..
48. I think that game physics are physics laws that directly influence the game experience, or that are a major part of the gameplay of the game.
49. Is a physics model of a game. It could go from simple model like in the walls of "Pong" to really complex model like in flight/space simulator.
50. The set of rules for the movement of objects, in such a way that mimicks real-world physics (to the point where a user can intuit what will happen) and allows for emergent behaviour when objects interact.
51. gravity, inertia, how a character interacts with their environment.
52. The ability to move around objects, smash objects or even just something falling down from the sky is using some sort of physics.
53. how do you make physics?
54. Having a set of rules which mimics real-world physics (well, mechanics mainly) controlling the behaviour of in-game objects instead of using a predefined set of animations.
55. games trying to adjust to reality, or simulating something =P (sorry bad English, i talk spanish)
56. game physics is when an environment represented in a game is given realistic object interactions, for example doors shattering, gravity, balls bouncing off walls. this provides extremely interactive

gameplay when implemented properly.

57. physx
58. object interaktion in a fantasy (or real) world between objects in this world. mostly to try to be as good as reality ;)
59. The use of newtonian physics in games, usually most used for things like collision and rag doll skeletons. It also crosses over with gameplay as it is directly related to the physics world.
60. Game Physics is dynamic processing of a games sprites, polygons, or environments, to elicit a realistic reaction and outcome.
61. Game physics are the physics simulated in a game to match the real life physics
62. implementation of real-world physics in a game, sometimes extrapolated to have more fun
63. I think that game physics exist when a game has physics close to real life. Games that are physics games have Newton's laws implemented in them.
64. Being able to move objects freely around a map in a game.
65. I think that game physics is collision detection and similar stuff in games.
66. I believe that "game physics" is a term meant to be used for physical integrations between an object, another object or its environment; however, I believe that it excludes simple interactions like walking left or right, and only pertains to complex interactions like buoyancy or momentum.
67. The physical interaction between in-game objects and the in-game world, aswell as other in-game objects.
68. Games of which the gameplay is based on intelligent/original physics programming. Could be realistic (Ragdoll, fighting) or surreal physics (Asteroids, Geometry Wars, anything vector).
69. Game Physics are the basis of how the certain game is played.
70. It is like the ragdoll effect. When a charcter or thing in game is getin hit, it flies away and lands like in the real world almost.
71. Game physics are the physics in a game which govern how objects and characters in the game interact with other objects.
72. The engine a game uses to decide how to treat objects in game.
73. The interaction between objects in a 'level'/world, like they do in 'real-life' eg, if you kick a barrel it would roll of the stairs.
74. When you can hit or bump something and expect it to fall over or otherwise react like an equivalent object would in real life.
75. "real" interaction with the game environment and the items within it
76. It's about realistic movement in the game, and the game would have realistic aspects of real world. For example : if you shoot a box with a gun, it will move / break / crack
77. Simluating real world physics in a virtual enviroment.
78. It's all about making the game react as if it were a real-life situation. It doesn't necessarily have to be realistic, but it reacts in a way that you would instinctively recognise.
79. game physics are programs in games that simulate even at the most basic level physical interaction with the game world like gravity for example
80. It's a (usually real-time) physical simulation in a game.
81. Game Physics define the main physical laws that are used in the game, like gravity, velocity, drift, hitback.
82. It is the (semi)realtime simulated physical interaction of objects in the game world, which are normally related to player action in the environment. It may take on the form of a challenge (eg.

simulated elasticity structural deformity in Triptych) or a means to help the player (the gravgun in HL2).

83. It's the thing that makes stuff in games move almost correctly according to the law of physics. The code for the physics are extremely complicated and sometimes require a lot of juice from the computer.
84. It's how the physics is simulated in a game. How objects fall, how resistant objects are, etc.
85. The best game features right after the graphics ;)
86. A game with a realistic physics
87. It's basically a simulation of what happens in real life. Drop an apple and it falls to the floor, and maybe bounces or rolls a little. It doesn't just fall straight down.
88. A game that revolves heavily around physics. Ex Armadillo Run
89. Game physics is the way things act in a 3D environment in a game.
90. It is when an object moves realistically without being a static object.
91. The manner in which objects react and behave with each other.
92. The inclusion of realistic response to events in a game using some form of physical theory as basis.
93. Game physics is how objects work in a game. Example, a table floating in the air (with exception if it's possessed). It is physically impossible for a table to float in the air due to gravity and physics.
94. Unique game developer term with precise definition. It's severely misused however
95. The application of real world physics to game movements of objects, such as rag doll to simulate reality in terms of contingency. Interactive environments and objects that respond.
96. Game physics allows to have coherent "laws" in the game. Although not necessarily real, it still provides feedback to the user that he might understand (at the price of a bit of adaptation if rules are not real-world-like). It also is a decisive element in the immersion of the player.
97. The modeling of physical behaviors.
98. Simulating an interactive physical environment in real-time for entertainment purposes.
99. simulation of the physical properties of objects in a game, and the forces acting on them.
100. The way in which a game's properties simulate motion and interaction among objects.
101. the simulation of the physical behaviour of a fictional world. it can be more or less realistic, also depending on if it wants to be (a copy of the world as we know it)
102. The ability of a game engine to render real world qualities to virtual objects. Such as rag-doll, trajectories, impact consequences, and representing object weight, inertia and gravity to the virtual environment.
103. Game physics are the scripts or limitations applied to a game that make things react in different ways, for example if you pick up an object and throw it, it will have different virtual forces applied to it (the force of being thrown, gravity, etc.)
104. Interaction with the in game world. Objects, particles, ammunition, gravity
105. Game Physics is how accurately the physics is presented. It is partially represented through the engine, and partially through art. Any representation of AI (artificial intelligence).
106. A part of a game devoted to controlling movement within a game environment.
107. game physics is the part of the game engine (whether created in house by the development team, or bought in as middleware) that gives objects in game certain attributes. This is often in accordance with real world Physics, and usually focusses on Gravity, applied forces, bullet trajectories and water motion.
108. physics that objects display when force is applied to them.

109. The necessary data calculations for virtualizing movement.
110. It is representing a force, usually gravity or force of contact, 'dynamically' rather than hardcoding the movement or animation.
111. Gravity. Such as jumping, running, bullet speed.
112. How the objects react in the game to player stimuli (sp?). How monsters fall down death, how buildings collapse etc.
113. realism in a game
114. The way in which physical objects in the world interact with each other and the world.
115. The simulation of "physical objects", in a game. These objects have certain physical properties like real world objects and it's physics engine's job to simulate as closely as possible the objects as if they were real world objects, but developers may trade accuracy for performance, especially in games.
116. Game engine functionality which replicates real world behaviour of objects in an attempt to provide a convincing portrayal of the physical universe.
117. Simulation of Physics (ie, physical laws) in computer games.
118. The ability to move one or more objects in a style, when a force is applied to it, to that found in the real world.
119. The physics that are applied to a game world. So influence of gravity on objects, interactions like pushing objects, momentum/friction etc.
120. To me, games physics are the way the ingame characters and object react to each other, in terms of movement and momentum
121. an engine within a game that creates an artificial sense of movement and gravity
122. It is the algorithms used in a (usually 3D) game to simulate interaction between objects: contacts, movement calculated from forces applied, breaking of objects depending on stress, realistic movement of fluids, vehicle movement, ragdoll physics. The aim is to simulate reality in order to enhance gameplay.
123. modelling of effects such as gravity etc in a virtual environment
124. The manner in which objects/buildings/persons, etc. react physically in their environment. How realistic body is at running or falling or buildings exploding, etc. How those particles and elements interact with each other and with the "world's" gravity, wind, water and other factors.
125. how objects act/interact with the player/game
126. "Game physics" is a set of guidelines laid down in code determining how objects in the game will react to certain physical stimuli.
127. The physical aspect of a game be it characters running, dying, jumping, being blown up ect.
128. Game physics engines causes things in computer games to behave in a realistic manner regarding their physics. For example, buildings fall realistically, barrels tumble and bounce off walls, etc.
129. The ability to do stuff to the background in the game?
130. The physical representation of real world objects and how they behave when a physical force is applied to them.
131. The way people, monsters, and other objects jump and collide - anything that's calculated while the game is occurring. In some games like The Incredible Machine the player can adjust gravity and air density and things. Spore is supposed to use a detailed system of physics to generate animations for its player-built creatures. If a game invented a natural resources like magic that behaved according to laws, that might be considered speculative physics.

132. Game physics are the elements of a game that control the way that you and other things/people interact with the environment, and how they react to your actions.
133. I think "game physics" refers to a game's attempt to simulate physical interactions between characters and their environment in a way that mimics real-life behavior.
134. nice feature
135. the rules, boundaries and freedoms a gaming environment provides in relation to gameplay.
136. The part of a 3D (or 2D) game applications' "engine" that creates the real-world physics behind the game world; e.g., objects falling, force, gravity, friction etc.
137. The world environment of the game...lighting, gravity, action/reaction, etc. Basically the reality (or unreality if simulating a fantasy world) of the game.
138. How a game creates the illusion of reality, or how a game attempts to mimic physical reality in a convincing fashion.
139. The rules that govern how a player interacts with the game environment, such as movement, gravity and the effects of collisions.
140. The actual physics of a game itself, like the Havok engine.
141. Game physics is how the engine simulates interactions between objects in the game.
142. Game physics is simulating some aspect of physical mechanics or other real-world behaviour in a game environment to increase immersion or interactive possibility.
143. Simulation of the physical world (or some possible physical world) as part of a game.
144. A whole world of simulated 3-D objects interacting naturally (or at least consistently).
145. Game physics is the representation of real world physics within a game, i.e. no dummy movements or pre-processed movement schemes. Everything is calculated on the spot, so that each movement is relatively individual and independent from each other. They also affect how objects interact with each other on a realistic level, i. e. the transfer of kinetic energy from one to another. Game physics can have a very deep impact on a game, making it a more immersive experience, especially in the likes of the next-generation first-person shooters.
146. The method of handling the physical interactions between characters, objects and other elements of the environment, be it simple "blocked tile" detection or complex collision detection. Also, one can include game controls in the field of game physics: acceleration, movement, etc. can be handled via the physics engine.
147. The physical interactions between objects/players/items in a gaming level and/or environment. Not only the physical collisions and their calculations but also such things as flexes, muscle, mass, weight and buoyancy.
148. The real time calculation of approximations to real world classical mechanics and fluid mechanics, for 'objects' within a game world.
149. I would say: The simulation of representations of real world objects through rigid body simplifications. Calculation of gravity and collisions in order to give an approximation of the real world behavior of such objects.
150. Game physics is the programming of code designed to represent the rules of physics (gravity, momentum etc) in a virtual context.
151. How an object moves or reacts in a game
152. Any attempt to emulate/simulate real-world physics in a game, whether it's as simple as falling crates or as complicated as accurate liquids. I guess.
153. Game physics is the set of rules that govern how objects and the player move in a game.

154. The implementation of the basic laws of physics in relation to objects in game.
155. Physics that is incorporated in the game, part of the game execution pipeline, which is making physical objects inside a game look moving and getting interacted reality like.
156. Simulation or realistic physics into games - eg - effect of gravity, and density on mass, energy etc... effect of friction , movement of particles all to give a more believable experience to the gamer
157. Accurate or near-accurate modelling of physical phenomenon/effects (momentum, velocity, gravity etc.)on in-game objects calculated and projected in real time.
158. The physical laws that objects within a game obey.
159. How the protagonist physically affects the game world
160. Making the games world react in the same way as the real world. Every object has weight, mass, friction etc. Things such as smoke or water react in a realistic fashion. Games physics is a way of trying to make the game as realistic as possible.
161. Interaction of two or more discreet games objects, be they items, players, game scenery,
162. The simulation of physics within a videogame. The physics can either be used to replicate reality, or to have impossible physics (often occurring in some fantasy games).
163. A simulation of the actual laws of physics within a game. This can include objects moving, gravity simulations, water simulations, and probably many more.
164. The way objects and items react in games.
165. Object that obey many of the laws of physics
166. Realistic physics in games.
167. The internal rules of the virtual gaming world, like for example how a box will fall if dropped, how a blade of grass will move if disturbed.
168. It's the part of the game core that deals with the laws of the virtual world. It is the part that controls the way entities interact and behave.
169. The consistent behavior of physics in a simulated world.
170. The reactions of things and how realistic they seem. whether things react how they should.
171. The approximation of physically based motion in games.
172. Game physics is the implementation of a model of a subset of the physical laws of the universe used to enhance the realism or fun in a game.
173. Game physics is the simulation of the dynamic systems in a computer game. The most well-known form of game physics is rigid body dynamics, the simulation of the motion and collision-based interaction of non-deformable rigid objects. Another common form of game physics is particle simulation, the simulation of the motion and interaction of point masses. Other types of game physics (currently less common) include deformable body simulation, fluid simulation, fracture simulation (using finite element methods), and others. (No, I didn't look this up. I'm a game physics researcher / programmer.)
174. vehicle and character motion, collision detection, rigid body dynamics and special effect physics
175. Game physics is the interactive synthesis of the motion of game objects according to the laws of physics. Examples of such objects are rigid bodies, jointed characters, vehicles, fluids and cloth.
176. The reproduction of simple physical rules to recreate loosely plausible physical phenomena. usually in the form of rigid body dynamics or simple deformable objects like cloth.
177. How 'motion' is incorporated into games - ala when moving, throwing a grenade, firing a weapon, how objects fall, fly, move, etc.
178. Approximated simulation of real world physic's lows

179. Game lib, usually middleware, that simulates real life physics using basic newtonian equations. Usually rigid body movements sometimes constrained, and real time collisions.
180. Game Physics is the creation of a physics simulator in a virtual environment designed to create a more realistic (as in realistic within the confines of the game, not necessarily in real life) area in which to play the game.
181. Physical mechanisms (ie. gravity, friction etc.) used in games.
182. Game Physics refer to simulation of any physical process (but most frequently basic newtonian mechanics) in a context which attempts to balance the competing 'game forces' of control, realism and real-time performance.
183. the implementation of a realistic physics engine, so character models, and world models have ragdoll elements and will react with the game world realistically
184. The simulation of real-world dynamic interactions between objects (solid or fluid) or between objects and players.
185. The rules by which game objects interact through touching or other forces ( ie magnetism, which is rare in games ).
186. The simulation of interactive game objects based (perhaps loosely) on real world physical laws.
187. Currently simple collision detection, advanced lighting, and simulated gravity. it's massive calculations suffering from the damn decimal point i.e. compound rounding errors. Current Physics calculations SUFFER because of the "fuzzyness" inherent in using the base10, decimal, and sequential time systems. Game Physics today is BARELY physics and in my mind... brute force and waste heat that can be expressed as lengthy run time(distance from optimal). It's sequential physics, not relative physics.\*grin\* a barely get that one myself, but I'm working on it.
188. The simulation step of the game loop - that is, rendering code draws the scene but provides no interaction or collision. Game Physics is the part of the game that makes the world 'solid' and allows you to interact with it.
189. How objects move in a game
190. using the laws of physics, simulating motion of anything relevant to the game allowing the player to interact with objects and entities through the same laws.
191. The faking of real world physics.
192. Simulate real world physics in video games
193. The realistic effect of real life physics on game objects.
194. Game Physics is the implementation of real world physical laws (or an approximation) into a virtual environment. This allows for greater immersion as virtual objects respond to manipulation by the player in a way that is intuitive and realistic.
195. Making in-game objects interact with each other in the way they are expected to in the real world. This also includes the movement of the player's avatar or vehicle.
196. Game physics is the programming to allow objects to move in a realistic and more dynamic way. (No wiki, i just speak like that :P)
197. It's the way the player and objects in the world interact with each other. Rigid-body dynamics (I think that's the term, anyway) is becoming quite common.
198. the realtime simulation of realistically looking (mainly rigid) body dynamics
199. the accurate representation (or simulation/approximation) of physical laws within a game world (not necessarily those that we encounter in the real world, but physical laws none-the-less)
200. the manifestation of real world physics (i.e. gravity and how objects react to one another when

forces are applied to them, as well as kinetic reactions such as bullets passing through objects) in the Game.

201. Game physics is the implementation within a the confines of a games "engine" an attempt at approximating as far a possible the physical forces that affect the real world.
202. Game physics, I believe, is an attempt at simulating real life interactions between two or more objects
203. Game physics is how various objects in the game move and interact with each other. This can includes some of the laws of physics, such as bouyancy, friction and gravity. A good example of game physics and how they can be manipulated is Halflife 2 where everything interacts pretty realistically with everything else. Physics aims to make games more realistic and also enable amusement in most games, such as hurling objects at people and whatnot.
204. Game physics is the way items within a game world interact with each other; such as a falling box landing on water.
205. A recreation of natural laws present in the world used to make a game more interseting or lifelike.
206. The interaction between objects in the virtual world.
207. The use of numerical methods to simulate the expected evolution of a physical system in a game (such as rolling barrels, fluid mechanics, General Relativistic corrections in space-based MMOs, etc).
208. The simulation of the laws of dynamics in a video game.
209. Game physics is adding rules of physics to games in order to make them seem more believable, engrossing, entertaining etc.
210. The simulation of realistic physics in videogames.
211. Defines the behaviour of non-living objects in the game. eg: How a ball bounces, how a crate falls and tumbles, how a corpse falls down a staircase
212. A realistic representation of "real world" physics.
213. It's a simulation of real physics - gravity, joints etc...
214. using a physics simulation for game mechanics.
215. Somewhat accurate simulations of real life observable natural phenomena such as water, fire, explosions and rigid body structures.
216. A set of programming tricks for interactively animating game objects in a physically convincing manner.
217. physics for games
218. The runtime simulation of moving and colliding game entities.
219. mathematical simulation of game objects which provides the player with a greater degree of interaction with their environment, and supports emergent behaviour.
220. collision detection and response to that. using "real world" physics as a basis for the visuals, sound and game play.
221. Objects behave in game as the would in real life.
222. Trying to put things in games such as the effects of gravity, water effects, gas effects etc.
223. Game objects dynamically (i.e. non-scripted) interacting with each other in a realistic fashion to create dynamic game situations which have the possibility of being different every time depending on player or AI action.
224. Simulation of real-life physics (falling, colliding, explosions, water flow, elasticity, magnetism etc) in a game context, more or less realistically.



225. The part of the game engine that simulate the physical properties of objects and effects in the game
226. Game physics is the correct manipulation of an object within the gaming world. The object will react accurately to any form of interaction.
227. The dynamic in wich in game objects are effected byt the game world or player.
228. Basically, objects being either attracted or repelled by other objects, and what happens when they collide. I think it also covers things like fire effects and lighting, but I'm less certain on that.
229. Placing realistic physics upon objects in games. e.g. Objects react to forces in game as they would in real life.
230. The implimentation of a simulation of phsyics within a gaming enviroment
231. I believe its mainly number crunching, but I'd define it as 'the realistic movement/reaction of objects in-game'.
232. And I thought this survey was gonna be easy... Well, I guess game physics are the same as real physics. You know, interaction with stuff, that can cause over reactions, etc.
233. Simulating the physical properties of every physical entity in a game, so that they will respond to forces in the game as they would do to those forces in real life.
234. The mathematical system in the game's code which determines the physical properties of different objects in the game environment and how these objects and their properties interact with one another.

11. Now that you had a chance to describe what you think "game physics" is, can you give one or two examples of what it is NOT in your view?	
Total Respondents	217
(skipped this question)	185

1. A gimmick
2. In my opinion, game physics is not anything that has a massively predetermined behavior such as an animation in a game that always occurs one way (i.e. the death animation in the orignal Halo game).
3. nope
4. Something demonstrating a half-hearted attempt at physics or games using only a standard liquid physics engine.
5. jumping in a mario game, old tetris, sonic games, etc.
6. Game physics does not include simple 2-dimensional interactions such as sprite collisions. HOWEVER, other more complicated 2-d interactions (projectiles particle physics, etc.) do apply.
7. Most platformers have "built-in" falling speeds and things like icy surfaces don't have anything to do with friction. This is pseudo-physics.
8. half life 2 has a great physics system but isn't principally based on
9. ?
10. They're not necessary for a game to be fun (but they add more value to it).
11. It is not bullet time, or any other silly effects people consider to be part of the games "physics".
12. um? i dont know...
13. ???
14. Graphics, rewards, user interfaces, story, and AI would not be included (although those would all interact with game physics, of course).

15. It is not textures, dialogue, characters, models, ect...
16. calculations
17. Things that let you draw objects, but not interact with them in any way.
18. Nothing really.
19. Non physics games rely more on animation than on the actual manipulation of objects.
20. It is not "Need For Speed" - Apparently theses guys think that its all physics, but if you play it, it's defiantly not. If you go outside right now and jump in your car, drive up to 180km/hr and hit a brick wall, the newtonian physics and forces will damage you and your car beyond recognition, BUT in NFS, you can do exactly that and get a scratch. It's not "CS 1.6" I mean "fake" physics where if you jump into the air, you come back down. That's not real physics... real physics has momentum, forces, these type of things.
21. a game with just ragdoll physics is not really game physics, its just visual.
22. Gravity in certain games, such as Super Mario Bros., or the first Halo.
23. Game physics is not jumping in mario or anyother type of basic animation that mimics physics.
24. Advanced graphics, like reflective glass particles. How your character is animated.
25. Half-Life 2: Ragdolls Walaber Trampoline: The Cloth Physics
26. Shooting a non player character and it reacts in the same way every time, ie falling over the same way using animations.
27. The basic ability of movement of a character,
28. Might and magic: Dark Messiah. Unreal tournament series.
29. minesweeper, age of empires
30. I can't, actually. :)
31. Games that dont give that sense of realism, and/or games that dont use physics throughout the entire game (Not just using some physics for one or two objects, but more of environmental physics).
32. Pre-made animations that occur over and over.
33. I suppose gravity in a game would fit into this category. You expect things to fall, and its a staple of games. I suppose "game physics" more specifically are the result of two indiviual components acting on each other.
34. A game that teaches physics and a game thats engine is not centered around physics.
35. Well lets see... they aren't perfect. Yet.
36. Tetris, basically.
37. While you can have physics that are based on a system not found naturally, such as certain space flight models, if the model is not sophisticated, it is generally not called game physics. The model must be sufficiently sophisticated and close to the actual system, whatever that system is and however sufficient is defined. I would not call games that allow characters to jump physics, however I would call a game where the characters mass and acceleration has an effect on game objects.
38. well heres one thing thats not... on call of duty2 i got shot right of a building but just becuse a little tip of my toe was still touching the end of the buliding i didnt fall of and floated in the air... i mean come on! would that happen in real life?
39. animations are not physics.
40. Any games with space.
41. street fighter alpha

42. Mario coming back down after jumping up.
43. Anything that has to do with A.I or graphics.
44. I do not believe pre animated or scripted actions are game physics
45. Anything not dealing with how something physically moves through the playing field.
46. Game physics are not monkeys.
47. graphics, controls, models etc.
48. graphics..
49. Game physics are not physics that are just for looks and that do not affect the gaming experience.
50. there is no game physics were the game doesn't try to mimic a real situation or a fantasy one. Card videogames Got no physics model
51. Anything pre-programmed, or which might as well be pre-programmed. A side-scroller target falling through a gravity curve that explodes when it hits the ground isn't "game physics", even if it's following a properly described curve.
52. The graphics obviously, and the sounds created.
53. Physics are not sounds or graphics.
54. fdfsf
55. A goal in and of itself. A good game == game physics. Game physics != a good game.
56. uhh.. i think.. Minesweeper =P
57. game physics is not a non-interactive pre-rendered environment with no random gameplay element to it. for example, when killing an enemy, a game not using a physics engine will most likely have a set of three or four character animations, with one picked seemingly at random. When you dispatch an enemy in a game that is using a physics engine, the enemy will likely switch to 'ragdoll mode', so that the dead enemy falls realistically, bouncing off the ground and objects in a realistic manner.
58. hard question, everything that has nothing to do with motion of objects
59. Game physics isn't graphics or artificial intelligence in games.
60. Game Physics are not large flashy explosions or carnage.
61. Hmmmmmmm i cant think of anything sorry
62. well, everything that is predictable and scripted
63. Super Mario Brothers and Sonic are not physics games.
64. In half-life, only being able to slide boxes around the level.
65. preset deaths, no rigid body physics, and very glitchy
66. To me, game physics will never be Mario's jump. It's hard to explain why, but in simplest terms: its not hard to program the jump, but this is not to say that the momentum he carries into it isn't physics. I also don't think the movement in any side-scrolling shooter I've ever played implemented game physics. The interactions are always too simple and trite to be considered game physics.
67. Animation to look like physics.
68. Frame by frame (sprites) animation based games with hard-coded rules/gravity/throws/...
69. The first CS game where the characters land straight at any place And space invaders
70. It is NOT purely just how a basketball bounces in a game, it is how it bounces, and how other things react to its bouncing.
71. gravity, simple trajectory

72. a standard 'lay' animation on top of the stairs if the barrel h
73. Ragdoll corpses (ragdoll player characters \*are\* "game physics"), basic necessities like gravity, friction and momentum.
74. moving of "static" items which do not move without "real" movement
75. A game which doesn't really use any realistic physics. For example, using the same example as above : If a box would be shot, it would just stand there, and no mark is done from the shot. Or another example : If you're driving a car in game, and you stop, and you would immediatly stop without any speed slowement. It would just stop.
76. I can't think of any examples.
77. not really i dont know enough about physics to give a true account so i dont even try
78. It's not a scripted or animated sequence, but something where the player has control over the elements of the simulation.
79. Even Tetris has a falling speed - a gravity - or Pong has a certain ball and paddle speed. Every game is a simulation, so every game needs physics.
80. This could be an addition to the above... what it is not is the deflection of a "pong" ball from the paddle if it is predefined 45 degrees. If the responses of the physical system are non-dynamic and predefined, it would not count.
81. As said before, the physics in games aren't exactly real-life looking yet. It's starting to, but we're not there yet.
82. The gameplay, I guess...
83. It is NOT a silly feature that makes your computer crash or lame!
84. "Scripted" physics, or something like gibbs bouncing around. To me, those don't count as physics.
85. Super Mario is not a physics game, although the whole gameplay do revolve around physics.
86. game physics is not a wall that dosnt move game physics is not somthing that moves but you do not have any influence over such as a fire that no matter what you do it burns in the same way.
87. Master Chief in Halo, running and jumping like crazy and Galaga, the arcade shooter are both good examples of "fake physics"
88. Game physics is not hardwired animations as response to events.
89. Physics is NOT hard to figure out. If it's physically impossible, then that throws off the gamer. If it is physically possible, then it's great physics. A gamer does not need to be knowledgable in physics for it to work.
90. The same as normal physics. Continuous.
91. It doesn't perfectly emulate the real world.
92. Huh, hard one... :) It is not something that has to restrict the creativity level of a game. Let's take the classic example of fire in a spacegame, there's no fun without eye candy effects ;) (let's hope it was the kind of answer you wanted :/)
93. Control systems. Force feedback controls.
94. Movie physics Physical simulation for industrial robotics
95. scripted/hardcoded sequences of movement
96. Menus Pause buttons
97. ...
98. Game physics is not game character interaction, nor is it user interface (UI) related.
99. Game physics is not like a scripted path along which an object moves, or a scripted reaction to what a player does.

100. weapons power, sounds, image related.
101. Anything having to do with the UI (user interface) or gameplay aspects of the game. Most of the art has little to do with how physics is represented.
102. a bullet that does NOT fly through a wall is not physics.
103. game physics is NOT how fast the player can move, or how much damage a falling object, which is affected by the game physics, does to your character(s).
104. not graphics, not game play elements
105. It is not solely the virtualization of the "intelligence" of the program's response to human input nor is it part of a program's design, but it will still be a factor in determining those parts.
106. It is not about making the gameplay, technology or storylines realistic.
107. The storyline of the game.
108. Lore and npc interaction?
109. not sure
110. An attempt at a rigorous model of actual physics beyond what is required not to excessively challenge the player's suspension of disbelief.
111. Artificial Intelligence.
112. enemy processing in 2D games non-destructable objects
113. It is not what things look like or how things sound? Done really understand this question.
114. Characters/objects which move through set animations rather than using a physics engine (particularly when being knocked back, etc)
115. when a person deis they move around
116. Localized weapon impact.
117. model design, lighting deign, graphics
118. computer/AI/NPCs actions
119. Sorry, can't think of anything.
120. Character faces and design
121. Object color, and how it fits in with the goals of the character.
122. Umm usually there's no subatomic physics in games. Game physics don't have to be realistic - there can be objects that are more than 100% elastic (i.e. they keep bouncing higher and higher with no more input of energy) and two visually identical objects can behave differently (like some walls you can walk through because there are secret passages). People also joke about female characters' boobs bouncing as being game physics - it might be if it was calculated on the fly, but isn't if it's part of an FMV or other human-created animation.
123. the way the game looks, or controls.
124. I don't think that game physics are sophisticated enough to accurately convey to the game player what might actually happen in real life in similar situations. In other words, I don't believe that game physics are realistic enough to be totally convincing.
125. its not important to make a game fun to play
126. This is an irrelevant question since the one before this rules out what game physics are in my opinion.
127. Not the method of how movement is controlled.
128. Sure.
129. Game physics is not particle effects, not magic or skill systems, not a matter of purely visual accuracy.

130. Rules that govern storyline. It is ok for a science fiction game to include science fiction, such as methods of faster than light travel.
131. Types of impact.
132. Anything not having to do with interaction between objects. Character interaction for example.
133. It doesn't include AI, or other "character behaviour", since it's more to do with the way that the environment reacts. It's also not necessarily the player character's movements, though there can be interplay there.
134. Stuff that is so far beyond our ability to simulate on a physical level that there is no point trying. Examples would include human character behaviour, dialogue, and so forth. It also doesn't include things that are "set in stone", like the plot in a game that tells a story.
135. Not necessarily things like water or clouds, those are dynamic animations; neither are particle effects, which are a very local kind of physics have no bearing on the "game physics" of the larger whole.
136. As previously mentioned, any pre-calculated dummy movements are not game physics since they have already been determined.
137. It is certainly the source of realism, but not the source of a good game. Game physics make up a little part of the game. (Same as in the case of graphics). Unrealistic (or completely absent) game physics can still mean a good game. (And ragdoll physics are blatant eye candy attempts abusing game physics.)
138. Scripted events. Graphics. Gravity.
139. pre-generated/scripted sequences of cars moving/objects falling etc etc
140. Hmmmm... Character's movement? At least in current games. The newer crop of lucasarts games seems to be fixing that.
141. I would consider calculations with bullets and other things to do with ballistics to be separate, although obviously they are codependent.
142. It isn't when an game object is static or cannot be interacted with.
143. It is not showing a sprite of a destroyed building (early Age of Empires)
144. Shooting a barrel and it flying across the room bouncing wildly off walls.
145. it is not the motion of the game player it is not part of rendering pipeline it is not as integral for a game (puzzle games like mahjong dont have much physics involved)
146. rag doll effect is NOT physics animation of water on a sea shore or river, which is not affected by the presence of a ship , person or falling debris
147. It is not necessarily equivalent to the physical laws of the real world.
148. Scripted sequences
149. It is not the be all and end all. A game with the most advanced physics yet seen can still be of poor quality, and a game without physics can be more enjoyable to play.
150. Changing visual properties of the world (day/night cycle), scripted movement of NPC/game objects.
151. An excuse to detract fun from a game (unless it's a simulator).
152. In old FPS games such as Quake, I do not consider there to be physics. Of course, objects are attracted to the floor by "gravity", but this is such a basic simulation that I don't think it counts.
153. I don't really understand your question.
154. Scripted events made to look realistic
155. Just boxes that fall over when pushed. Just things that break when shot.

156. It's not the level design, the interface or anything else that doesn't apply to the "rules" of the gaming world.
157. It's not a simulation of real world physics. The purpose of a game physics engine is to be fun, NOT to be real.
158. Breaking the consistency due to programming errors.
159. static animations that are drawn to look like realistic events. If it isn't governed by mathematical laws.
160. - Traditional Character Animation - Prerendered Scenes with physics calculations in them
161. Game physics is not simply the encoding of physical equations. In order to be adequate, a game's model of physics must in some way enhance gameplay.
162. Game physics is NOT scripted game kinematics – i.e. hand-animated motion of characters and other objects. It is NOT other types of classical physics such as the simulation of radiometric transfer of electromagnetic energy (i.e. rendering).
163. pre-baked animation, inverse kinematics, scripted motion
164. Marmalade, jam.
165. It is not an accurate reproduction of the physical world we live in.
166. This is not real world simulation, game physics let us feel what it is, but modern computers aren't capable to create worlds with absolutely correct physics laws
167. Animations that are generated off-line, or scripted that don't simulate use Newtonian physics.
168. An attempt to recreate real world physics
169. Lighting. Textures.
170. An accurate / realistic simulation.
171. the physics should be separate from sound, graphics and artificial intelligence.
172. I'm not sure whether I understand the question, but game physics should not be just a gimmick to stay current with the latest fashion.
173. Game rules that don't appear to come from the 'physics laws' of the game. For instance, a double jump would fall under mario's physics, but picking up a powerup for more lives or to run faster would not.
174. Graphical effects, artificial intelligence, game flow etc.
175. Game physics is NOT ? It's not the story line, the artwork, the animations, or the music.
176. Game Physics is not necessarily the use of physics for gameplay (although that's a good use of game physics) - rather it's the underlying simulation structure of a game.
177. It's not AI
178. it's not: sloppy physics for the sake of games, it's not: 'real' physics for the sake of physics
179. True physics.
180. It's not an exact simulation of the real world because a game must be fun
181. Calculation of the transfer of momentum and resultant velocity of gamepad flung in frustration at screen.
182. I don't understand this question. 'Not in my view'? Do you mean the inner workings of the game itself which are performing the physics calculations, or examples of what I don't consider 'game physics'?
183. Bullet Physics (since i think they are point to point). Grenades on their first arc (i.e. before they hit the floor)
184. That's hard; anything's based on physics if you model it well enough. Graphics and sound aren't

at the moment. Contrived/abstracted situations, eg. Tetris, minesweeper, aren't, but Tetris's falling blocks work according to Tetris's "physics".

185. it is not an accurate simulation of real physics.
186. it is not pre-rendered animation, no matter how realistic the animation looks. It is also not necessarily the accurate representation of the real world in a computer game.
187. I have difficulty understanding this question, but I will answer it as best I can. Some elements of real world physical interactions are employed in games in more simplistic terms and are not "physics" within the game. For example if I was to reach down and pick up a gun in real life, I would have to use my hand to pick it up physically. In games this is simplified meaning if you run over it it "appears" in your hand. This is not game physics. The same applies to many things such as health packs, etc. A game like Quake 2 had little I would now define as game "physics" as all no objects truly "react" to exterior forces other than in a simple binary way. I.e I shoot a bad guy he dies, but that body is them immovable.
188. Game physics is not, and never can be, 'real'
189. Lighting within a game Voice interaction by characters
190. it is not a necessity nor is it always appropriate
191. Its not making explosions look pretty.
192. - Graphics; while being 'physics' in the real world, it is defined differently in most games. - An attempt to be an accurate physical simulation; the purpose of a game is to be believable and semi-predictable, so accuracy is only a concern if the user notices the discrepancy from an exact solution. (I was joking about GR in space MMOs).
193. A gimmick to sell games.
194. Scripted encounters where perhaps only one or two objects move to give the impression of physics.
195. Grass swaying in a game (Oblivion for example) has no correlation with wind or anything else, it is merely a movement to make the game appear better.
196. behaviour of living things. i.e, how a living being walks, runs, jumps etc.
197. It is NOT an opportunity for the developer to create a "pseudo physics" system that only works when the player chooses the one solution to a problem that the developer has decided is the correct one.
198. not graphics, not sound
199. simulation on special effects. i.e something that doesn't influence the game
200. Intelligent behavioural pattern simulations and accurate real world lighting fall outside my concept of game physics.
201. Scripted walk cycles.
202. rendering audio bowl of walnuts
203. Anything static or animated/pre-canned.
204. events such as Mario's jump in Mario 64, even when the player has analogue control over the direction and duration don't have enough basis in real physics to count. Scripted physics events such as a wall which crumbles the same way every time.
205. escapism and modernism. (in other words: I don't know what you mean)
206. The way a game looks and the way a game plays.
207. Scripted animations. I'd also say "effects" physics were separate from "game physics" - e.g. making realistic explosions using physical modelling of flying particles which don't interact with the player isn't the same as actually having those particles interact with the rest of the game environment.



208. (Don't like this question: game physics is not lots of things, including a gothic cathedral) What you're getting at, I suppose, is that it's not real physics but a simulation. Can't be more specific than that.
209. very simple, unrealistic pre-programmed movements of objects.
210. Pre-scripted animations AI
211. Uh, straight trigger stuff. Pre-scripted collapses. Static fire animations. I think the physics essentially needs to be fluid, it can't be an on/off thing. Apologies for the vagueness here.
212. Unbelievable physics on objects, such as you shoot a barrel and it flies 200 feet into the air. Like in the JKF game, you shoot him and he jumps forwards about 50 metres.
213. Unfortunately not from the top of my head
214. Tables fixed to the floor immovably. Enemies dying without ragdoll physics and their corpses sticking out of stairs at odd angles etc.
215. Erm...
216. The "ragdoll physics" in Deus Ex 2. Having bodies flop around and their arms flail ridiculously when shot.
217. Anything which is designed to act in a pre-determined, scripted way that would not happen normally as part of a natural, physical process.

12. If you know of physics in games that have some physics in it, please list one or two of the best ones here. Maybe include the type of physics they use and why you think they are good. (If you don't know any, that's fine - just skip this question.)

<b>Total Respondents</b>	<b>221</b>
(skipped this question)	181

1. Faces of War: Destructible environments, simulates vehicles well, and bullets extremely well. Half Life 2: The game that proved that there was a reason for physics in games
2. Two games that definitely show off very profound physics include Half life 2 (though I've only played it once) and a small independently developed game called Rag Doll Masters.
3. World of Warcraft, takes into account all the different interactions, such as riding a mount, holding a weapon.
4. Robo Blitz - Abstract physics. Marrying various theoretical items to real-world physics is FUN. Cortex Command - Excellent particle physics. Completely customizable physics engine.
5. Armadillo Run Toribash Half Life 2 Gish
6. Half-Life 2 - some of the best rigid body physics simulation on the market today Far Cry - great "rag-doll" simulations and rigid body physics Splinter Cell - great rag-doll and cloth simulations Trespasser - not great physics by today's standards, but it was one of the first games to attempt realistic physics simulation
7. Cell-Factor has a very solid feeling physics engine, objects felt like they had a tangible weight and they weren't slippery or rubbery. On top of that it played an important role in gameplay: when I played against my friend I don't think I shot him a single time, we both just threw junk at each other.
8. -toribash : the ragdoll system is just perfect for this kind of game -armadillo run : typically the kind of game wich MUST have physics reactions
9. Toribash Armadillo Run Bridge Builder Gish
10. Half Life 2 Flat Out 2 Armadillo Run

11. I haven't really played any "great" physics games. I say this because most games today focus more on graphics than realistic environments. And most "ragdoll" or "physics" games are 3-10 minutes long.
12. half life 2 is the best game ever and it uses the havok physics engine. and half life: source is pretty awesome to.
13. cortex command half life 2
14. Gish, Braid (No, it's not out yet; I was a beta-tester), FlatOut, and maybe Psychonauts
15. Half-life 2 Ski stunt simulator
16. Flatout 2 (Havok Engine) Half-Life 2 (Havok Engine) Ghost Recon Advanced Warfighter (Meqon-PhysX)
17. Half life 2 - Comes with a level editor, where you can spawn objects and create a rube goldberg - ish type machine. Armadillo run - Guide an armadillo through courses using materials given such as cloth, metal sheets, rubber, metal bars ect.
18. Armadillo Run, Simply the best physics game over. Uses a very complex system of physics to accomplish what it does.
19. 1. Falling sand. It may seem 2-d, but it has physics, and I really enjoy sandbox games. 2. Half Life 2. The game is simply awesome, and it has sick physics (well, in its time, atleast.)
20. Toribash - Humanoid figures that are controlled by the user (turn-based). The bigger momentum+force allows more damage to the other figure and sometimes a limb will fall off (the physics in this game are not exactly newtonian physics - I've never seen anyone rip of someones arm before). The most awesome thing about Toribash is you can basically customize how you want to do it. Like Tekken was an awesome game, but I got fed up with it because I wanted to do my own moves and so forth, and in Toribash I can. I control all movement of my figure, and you can actually think about a move before you do it in your head - it's practically the same as anything you do in real life. You jump by using mainly you knees and hips right? So you try that in Toribash, and your figure can jump. So simple, yet so fun to play. No missions, no finishing the game so you can play with the awesome things. It's just there, ready to do whatever you want, whatever you can imagine.
21. Armadillo run, physics puzzle game where you build constructions. Toribash, physics-turn-based fighting game.
22. Robot Arena 2 has to be my favorite, you build a robot from parts and then use it to fight other robots, but the robots apply very realistically to the laws of physics.
23. Halo 2 with 3d rag doll physics and good collisions, Gears of war, line rider, any game from www.teagames.com, lots of others
24. Garry's Mod, a game for Half-life 2. (<http://www.garrysmo.com/>) In it, you would build things. You could attach objects together, freeze them in the air, weld them together, attach ropes, nearly anything could be built. The physics reacted like the would in real life, or very close, and you could do many things with the physics. IL-2 Sturmovik, a flight simulator. I would say the best ever. It not only had good graphics, but if a wing was hit, or one of your aeroloin (the wing flaps) was hit, your plane might start to slowly tilt to one side, and if a wing was torn off from a flak shell or going too fast and having your flaps down causing immense drag, it would tear off and you would not die, but still be able to control your planes flaps and such, just not very well. It was always interesting to watch planes spiral in, or crash and get wings torn off, or whole planes get blown in half.
25. Half-Life 2: Ragdolls Walabers Trampoline: Ragdoll

26. Toribash - Very good ragdoll physics. Half-Life 2 - Exelent use of physics overall
27. Age Of empires 3 : Uses The Havok physics engine, i think that these physics are good because it adds a 'freedom' to the game where buildings get knocked over when hit with cannons, objects fall and interact with the environment in a believable way. I think good physics adds a sense of 'complete control' to the game, where you are not limited to just one or two set ways to achieve a task, you can use multiple ways and each one works in the way you expect.
28. The Newton physics engine is... rad.
29. Half Life 2! Really great physics simulation there. Toribash Trackmania Nations
30. I've played quite a few physics games, but I think two of my favorites would be N, Nball, and Trackmania Nations (Any of the Trackmania series is good, but Trackmania nations is free). I guess they use some good player physics, but I wouldn't be able to say the type of physics they use (Possibly basic gravity/force physics...). The reason they are my favorite is simply because they're free (Not Nball), you can make custom levels in them, and they're fun.
31. Ragdoll Masters, Trackmania Nations, Ski Stunt Simulator.
32. Ski Stunt Simulator. A skiing simulator that measures friction, speed, rotation based on torque, impact, and stresses. The game is always different and exciting, it is fun and challenging.
33. Toribash, it uses good ragdoll physics and is very in depth with the fighting style. Stair Dismount, It also has good ragdoll physics.
34. Half-Life 2. I'm certain almost everyone who took this test put in Half-Life 2. The game is the closest thing to real life yet. It has the common "ragdoll" and water physics but the way it is played out can be almost anyway you want. For example, if you put in a cheat code for all the guns and unlimited ammo early on in the game you could change the outcome of the rest of the game.
35. Half-Life 2 - modified Havok Engine - it simulates all objects, big and small, customizable buoyancy and stuff Cortex Command - good flying and destruction
36. Trackmania Nations: racing game with semi-realistic simulation Half Life 2: sophisticated interactions between the player and objects, and objects and objects, such as crates.
37. Toribash has very realistic physics, and flatout has VERY VERY REALISTIC PHYSICS.
38. half life 2 well, basically anything that uses physics.. i am blanking out right now
39. Armadillo Run Gish
40. Gish. Half Life (for having fun with ragdoll(ish) behavior). UT2 had fun ragdoll effects. sure there were other really good ones. can't think of any at the moment.
41. Cell Factor- Ageia Physx (crazy physics including cloth and fluid simulation) Stoked Rider- Ageia Physx (fluid physics and avalanches)
42. Half life 2 - Physics involved with puzzles and almost everything in the game portal - portal make the physics set up by half life 2 more in depth (donno if that counts or not)
43. Toribash would have to be one of my favorite physics based games. As for a non-Physics based game (as in its not solely about physics) i would have to say F.E.A.R.
44. Half-Life 2
45. Armadillo run.
46. Half Life 2, they have good physics overall and also nice ragdoll physics i think
47. Half Life 2 is the best I've seen so far. It uses interactive physics and physics puzzles, and even uses physics as a weapon (Gravity Gun). BreakQuest is another one, and it uses physics and mixes it with the classic Breakout game to make a Breakout game with physics that directly influence

the gameplay.

48. Orbiter - Gravity and flight dynamics LineRider - Gravity and motion
49. Armadillo Run, FlatOut.
50. Halo 2: ragdoll physics, Half life 2: gravity (can you say gravity gun).
51. SwitchBall (a very high amount of interactivity including cloths, fluids and is supported to use PhysX Accelerator Card), Gears of War (the best amount of physics, you can even move around the dead locust bodies just by kicking them) and Cell Factor (fully destructible environments and up to 10K objects moving at once).
52. dadadaw
53. Battlefield 2 Hitman - Blood Money
54. Max Payne 2 Half Life 2 Toribash Flatout 2 Cortex Command
55. armadillo run - uses basic (but realistic) 2d physics presented in a 3d manner. allows the user a nearly infinite number of solutions for each puzzle. ski stunt simulator - again uses 2d physics, but using the mouse as a controller requires some degree of physical skill, as opposed to pressing a couple of buttons.
56. breakquest - rigid-2d-physics with very believable physics bridgebuilder - also 2d, but springs. very simple "objects", but deep fun
57. Gish - You're a tarball and you have to get around, they used physics for most of the puzzles, pulling levels and swinging around on ropes.
58. Toribash is probably my favorite because almost nothing is predetermined and each match is unique. I also enjoy the simplicity of Line Rider.
59. Half life 2, Oblivion, Dark Messiah and a couple more games
60. all, Havoc games HL2 ragdollmasters toribash soldier
61. Gish and Cortex Command are physics games.
62. Half-life 2, and Dark Messiah of Might and Magic
63. Half life 2 had what I felt was a very fine execution of physics. Objects had momentum, weight and resistance and you're given ample ability to play with the physics and even use them to dispatch your foes. Many FPSs try to do this type of thing, but most just end up having one guard in the entire game that can be killed by shooting a stack of barrels, but they never use the physics again and it's a waste. Half Life 2 may have been the first game to innovate the use of physics successfully. Gish was a unique game that puts you as a blob of tar. It had simple controls, a good learning curve, innovating ideas and level design and the game innovates and re-innovates the uses of its physics engine over and over again. One innovation I can truly appreciate is that to jump, you make yourself sticky which slows you down and flattens you on the ground and then you press space and shoot yourself upwards. N is one of the few platformers that deserve worthy mention anywhere. It doesn't use a complicated physics engine (actually it's collision detection that is its real prize), but it uses what it has so well that it's worth a mention here.
64. Half-life 2.
65. -Toribash (Ragdoll, unexpectedly entertaining after some digging...) -Mu-cade (Friction, gravity, originally used)
66. Toribash- ragdoll combat fighting Cortex Command- land destruction future war
67. Armadillo Run. It is fun and you can create almost anything And the bridge builder games
68. Half Life 2- It has many physics oriented puzzles. Armadillo Run- an independently developed game that is based around physics and building different contraptions, like TIM.

69. Elder Scrolls IV: Oblivion : General Object physics, rag doll dead people
70. There's a mod called "Bouncy" for Quake 3 Arena that disabled all damage except for 10hp loss maximum from hitting the ground too hard. All weapons had their impact momentum increased to match the old damage rating, and an off-hand grappling hook with limited energy was added. Increasing the hook shot velocity and pull tension up high enough to be useful made playing "space" maps extremely fun, by encouraging really impressive aerial acrobatics without giving players the power of flight.
71. Half life 2,
72. Gran Turismo 4 - I just love the driving mechanism and how good effort it is. I know the full game isn't physically made, but the driving is made really well.
73. Unreal Tournament 2, Unreal2 engine
74. Half-life 2 used the gravity gun, which was a real break-through for physics. It was semi-realistic. The other good one is line-rider, which is cartoony and fun, and the interactivity is outstanding. It's so much better than just watching boxes fly.
75. well ichor for liquid physics and toribash for fight physics off the top of my head
76. Ufo the Unknown Enemy (the destroyable terrain and fluid simulation), Silent Storm (for the same reason), Tower of Goo (a very fun game that uses physics as a game play element), The Incredible Machine (also uses the physics as a real game play element instead of just being a decoration)
77. Trackmania Nations and especially TM.Sunrise - many tricks and stunts can be done with the cars. Armadillo Run, because of the endless number of building possibilities.
78. Half-life 2, even if it's old, the physics engine is very flexible and look more real than most new games. I think Crysis will have very good physics.
79. Max Payne 2 has got great physics in it : the dead corpse were ragdolls, and every objects (like chair, tables...) can be move and can fall.
80. Rainbow six vegas: nice bodyphysics :) Dark messiah of might and magic: body and objectphysics
81. Ski stunt simulator is one of the best 'Toribash is very cool too
82. Half Life 2 - it has some great physics, and even applies what is probably the best gun in any game to date- the Gravity Gun. Not to mention the ragdoll physics (my favorite!)
83. Armadillo Run Garrys mod And many games on Fun-motion that i cant remember the name on.
84. Company of heros for overall realistic physics. Red orcherstra so mostrelistic ragdoll and ballistics physics
85. Prey, where boxes move when you walk into them and how the gravity changes make the whole room's objects freak out.
86. HalfLife2 - cool puzzles using physics, throwing of stuff using gravity-gun
87. SSX - Has SOME physics, however some of the tricks are impossible to make in real life. Tony Hawk's Underground - the physics of skateboarding is good, and when you flip or turn too sharp, you fall. That's how it is in real life. (These are the top of my head, there's more, but this is what I can think of)
88. Half Life 2 (rag doll, gravity gun, interaction with objects in path), World of Warcraft (cloth, water), Jedi (gravity)
89. \* The Incredible Machine: real world physics, sufficiently accurate to provide good clues to the player without being too much accurate to the point of a physic simulation which would be unplayable (especially back at the time of its release). \* Superbike World Championship: the one from EA (1999). You were feeling every thing in the bike from torque to weight through

acceleration. (Or at least, it was good enough to make me imagine a part of it ;))

90. Burnout 2 and 3. Half-life 2
91. Gish - it has a pretty cool 2D spring-mass system Half-Life 2 - it has pretty cool rigid body dynamics
92. half-life 2 (gravity gun)
93. Battlefield, Battlefield 2 (Most games require at least a basic form of physics; those two Battlefield games happen to be some of the more advanced ones, which included such things as bullet drop.)
94. prey - nice gravity tricks dark messiah - physics used as a weapon i-war - inertia
95. Half life two has great physics. For example, it was the first game I played where bodies of defeated enemies can fall, or be thrown across the room realistically, with their arms and legs flopping round and stuff, due to the ragdoll physics.
96. Marathon, Myth ammo flight path, low gravity walking.
97. Easier to point out when something is not good...but...I guess in games like Oblivion where jumping is dependent on the angle and height you are jumping from, as to how you will land or whether you will incur damage (the later of which I know is gameplay0. Or in Zelda T.P when you slide down the hill in the snow peaks and your frozen leaf responds the way a snow board does in real life.
98. Unity engine...
99. Half Life 2, where the physics appear to affect almost every aspect of the game.
100. Half Life 2- Havok good display of realistic movement Oblivion- Havok
101. Currently, AOE 3.
102. Half-Life 2: -simulating how objects are thrown was well done, both in how it relates to combat with the gravity gun and how it relates to puzzles with crossing hazards. -simulating weight and buoyancy was not well done since it has no effect on most of the game and was only used in very boring seesaw puzzles - 'rag doll' physics were okay. Less repetitive than scripted death animations but not as consistently interesting to look at.
103. Prey
104. Oblivion uses some awesome physics. Mobs fall and land in interesting positions, on a slope they will keep sliding down until stopped by another object as in RL. Also Titan Quest for almost exactly the same reason.
105. Half-Life 2: making the world feel more 'alive'. Armadillo Run: using physics to create very inventive puzzles and situations.
106. Gish, Half-Life 2
107. Half-Life 2 (gravity, projectile dynamics, simple optics). Good because it introduces fun play mechanics, but is not very naturalistic since the effects are too prominent and "centre-staged", particularly in puzzles. Max Payne 2 (gravity, projectile dynamics). Quite naturalistic, improved immersion in the game world. Severance: Blade of Darkness (gravity, projectile dynamics, fragmentation effects, real-time lighting and shadow). The most convincing broken object fragment behaviour I have ever seen - added hugely to the believability of the game world.
108. Halo - physical objects move realistically.
109. Age Of Empires 3 - its good because the physics are fairly accurate.
110. virtual pool - friction/momentum/gravity etc all modelled in this. Seems like a very accurate simulation, power on cue ball affects angles as much as your aim. shooters - things falling on the floor, bullets etc

111. Loads of games use the who ragdoll physics schtick nowadays (especially FPSs). I did enjoy Second Sight, mainly to TK enemies into stupid places and positions. The physics were a bit dodgy (getting boxes glued to your feet), but that made it more entertaining.
112. Gears Of War: a third person shooter. i think the physics in this were great...why? because i didnt notice them. they were extremely realistic.
113. Doom3 Halo : good ragdoll physics and vehicle steering.
114. Unreal Tournament + Halo both have consistent, convincing physics modelling
115. Age of Empires III - love the way all the wind blows, buildings explode, people walk, etc. especially enjoy the ship battles and deeply elements interact with their environment.
116. half life 2 - Source engine
117. Age of Empires III had some spectacular physics such as dynamic building destruction and ragdoll effects. Another favorite would be Gears of War, which has some truly astounding physics (a tower collapsing brick by brick would be an example here).
118. Ghost Recon Advanced Warfighter, uses dedicated Ageia PhysiX engine processor.
119. Half-Life 2 (Gravity Gun)
120. The Longest Journey, it lets you do things to different puzzles, you can move pieces of the environment to move on to new places and make things happen.
121. Half-Life 2, Psy-Ops, Halo 2.
122. Hmm lots of games have pretty good physics - the Mario games, most sidescrollers... Good game physics are those which are consistent throughout the game and which make the game easy to control, things move where you want and expect them to, and creatures move in ways that make them seem alive. Realistic physics such as humanoid characters that can't jump very high are usually bad game physics because they feel sluggish and don't look exciting.
123. Half Life 2-it uses the Havoc Physics engine. i think the physics are good because they bring a new sense of realism into the game. unfortunately, Havoc will never be licensed to the Mac do the licensing costs.
124. I feel that X-Plane has good physics. Being a flight simulator, physics are of course more integral to the design than in a platformer.
125. half life 2. I think they used a 3rd party library. I believe they used it for ragdoll effects and for various game objects.
126. In Soldier of Fortune 2 the weapons recoil and accuracy are made to mimic "realistic" gun behaviour wich makes the game more intense because it adds randomness to the gameplay.
127. Half Life 2 using Havock. Just seemed very real. Anything created using Unity game engine and Aegia.
128. Many first person shooters have good "realistic" physics, although there are some that don't. Call of Duty...lighting is great, gravity seems realistic (although you can fall from pretty high without taking damage).
129. Vendetta Online attempted to include inertia in the flight model, and tried to make collisions look believable.
130. DMC3, Half-Life 2.
131. Gish, which based a simple but fun game on mechanics. Half Life 2, which incorporated physics for both an immersive environment and engaging puzzles. Oblivion, which used it well to give a visceral feeling to melee combat.
132. There is some quite nice rigid body mechanics in the bowling in Wii Sports. Myth II: Soulblighter

had some very fun, if rather unrealistic, ballistics.

133. UT2 had amusing rag-doll physics, though not perfect. Halo's Warthog had a completely believable exaggeration of vehicular mass and movement.
134. Doom 3, Quake 4, Prey, Gooball
135. Quake III Arena features rather unrealistic physics, e.g. you run faster when you jump, you can do rocket jump, etc. Of course, it doesn't mean it has an incomplete or bad physics, only a unique one. Half-Life 2 and Prey are also good examples where physics make up a larger portion of the game; the gravity gun in the former and the controllable gravity (and portals) in the latter gave new opportunities at creating puzzles in the game.
136. Half-Life 2, Deus Ex: Invisible War, Unreal Tournament 2, Armadillo Run, Rainbow Six: Vegas
137. Good: Max Payne 2 Nice to kill people 'realistically' (I'm evil I know) Half Life 2 Gravity Gun... you should know the rest... Crap: Doom 3 it just wanted to have the same clothes as its friends... it doesnt make you cool mate...
138. Armadillo Run, awesome physics. I'm sure you know of this game. Far cry was the first game that really made me stop and start playing with barrels thinking "this looks real..."
139. Dark Messiah, obviously - which not only had ragdolls but the ability to activate them before death - kicking, etc. Freezing someone and pushing them made them topple over. Barrels and the like could be thrown and smashed realistically. Starshatter - The Gathering Storm. A space game, with realistic Newtownian physics (turn using small thrusters, cutting engines and turning still meant you travelled at your original speed in the same direction etc.
140. Half life 2 FEAR
141. Half Life 2, using the Havoc engine - one of the few games that actually tries to use physics properly, like the gravity gun and physics puzzles. FlatOut - not sure what physics engine it used, but the destructible objects and scenery, and of course the ragdolls flying out of the windscreen, help the enjoyment no end.
142. Half Life 2 - Highly interactive objects which can almost always be manipulated with the gravity gun. This is good as it gives a lot of player options in a quite linear game, N! - The player movement physics. These are not particularly lifelike but they do allow for wonderfully flowing game play.
143. Half Life 2
144. Half life - 2: Rigid Body, Cloth, fluid etc
145. Half Life 2- the gravity gun, the use of the magnetic crane - dropping a train compartment on the enemy was innovative, as most shooters expect me to aim with my mouse and shoot...the fact that I used the magnet, picked up a container and dropped it on the enemy means I did something unconventional and even more rewarding than shoot hide reload shoot! Battlefield 2: hard to keep the planes flying until u get it right - good fun after that.. the extreme mods also account for the trajectory of the bullets due to the effectof gravity etc - so u are suddenly thinking on your feet - a lot more than you used to!
146. Half-Life 2, Company of Heroes, Max Payne 2, Dark Messiah of Might + Magic. Max Payne was one of the first games to popularise Ragdoll physics in characters, while CoH uses physics modelling to great visual effect in battles.
147. Source engine games (Counter Strike Source, Half Life 2) - the physics of CSS increases the enjoyment of the game e.g. blocking, stacking, moving objects. Also causes problems for me when throwing grenades :p
148. Dark Messiah Half Life 2



149. Half Life 2: The best and most enjoyable use of physics I have yet played in a game. Meqon physics demo: Not a game per se, but a tech demo that places you in control of some scenarios where you have control of ragdolls etc.
150. Half-Life 2 - very few, if any, special cases, and a point to the interactivity.
151. Half-Life 2 - A very realistic simulation of object physics, and they were also integrated into gameplay very well via the Gravity Gun
152. The Elder Scrolls: Oblivion and Medieval 2: Total War.
153. Half life 2=In times of desperation, objects around you can be used as weapons
154. Half-life 2 Company of Heroes (realistic breakages, and object weights)
155. Elder Scrolls 4: Oblivion The Thief Games Deus EX
156. Alpha Centaury (Sid Meyer) had, in my oppinion, one of the best physics in any game. It was so good because it allowed the terrain to be modified while playing and that produced global effects like changes in rainfall.
157. Gish and Toribash, their main point may not be how they implement physics, but how gameplay and physics are mixed.
158. Dark messiah of Might and Magic Half life 2 (source engine games)
159. The Ageia Physics SDK has the best examples, so any game that leverages Ageia (Unreal 3 Engine, City of Heroes, Unity Engine, etc) has potential for very realistic realtime physics.
160. Half Life 2: excellent use of ragdoll physics, facial animation, and material interactions. Katamari Damacy: Bogus physics model, but ridiculously entertaining.
161. Half Life 2 – rigid body simulation Crysis – rigid body simulation, fracture
162. tombraider legend, half life
163. Cell factor, unreal tournament.
164. Half-Life 2: uses a modified version of Havok’s physics sdk. I would describe it as a good use of game physics because it furthered the gameplay and wasn’t just window dressing.
165. HalfLife2, Pray, Baloo (small freeware game, with gameplay based on phisic), PlasmaPong (fluid simulation on GPU)
166. Painkiller. Uses rigid body physics and rag-doll physics which is basically a set of constrained rigid bodies.
167. Half-Life - good interaction with objects, realistic movement (jumping, taking time to stop from a run) Pro Evolution Soccer 4 - the behaviour of a ball
168. Company of Heroes. Half-Life 2.
169. Psi Ops Half Life 2 But implemented new and unique control mehcanisms that allow the user to interface with the simualtion.
170. Half-Life 2 - gravity gun, and the general fluidity of models F.E.A.R. - fluidity of models Dark Messiah of Might and Magic - very well done ragdoll models, when fighting enemies, and they will roll down stairs / fall down cliffs etc realistically, and then, provided they don’t die, will recover and return to the fight.
171. Devastation for being among the first to use physics. Half-Life 2 for the puzzles.
172. Incredible Machine. ( advanced for the time ) Hubie ( cartoony but consistent )
173. Half Life 2 Dead Rising
174. Halo for example. Very good game physics... gravity and collision detection.
175. Half Life 2 - mostly rigid body dynamics, the first game to use game physics for something beyond rudimentary collision/”don’t fall through the world” - the game contained puzzles that required

you to remove support beams to collapse structures, etc.

176. Company of Heroes is the best yet Half Life 2
177. trespasser - an admirable effort to interact - miserably failed halflife2 - vehicles, constraints, balance - good but not enough of it gran turismo series - all vehicle dynamics - very well executed
178. What is good physics??
179. Half-Life 2 Company of Heroes
180. Half Life 2
181. Half-Life 2 Dark Messiah Garry's Mod 10 (excellent example) I believe all of the above are based on Valve's Source engine, and modified versions of the Havok Physics engine. Each of them has good simulations of basic mechanics, momentum, Newton's Laws + optical effects. Accurate simulation of these things makes for a good game physics model as they are physical principals encountered frequently in everyday life and so their inclusion in a virtual world adds to the immersion of the player in the world.
182. Trespasser, in that you could physically interact with game objects by manipulating them with your character's mouse-controlled hand. Further, all the objects had weight and momentum, meaning they could be used as in real life (for example, an empty gun or a plank could be used as a bludgeoning weapon, or to knock over an obstacle) Halo 2, for the movement of your vehicles over the uneven terrain, and also the satisfying deaths of opponents who can be sent flying or knocked limp by attacks.
183. Half life 2 (Havok) and Elder Scrolls: Oblivion (also havok)
184. Armadillo Run, and Dark Messiah makes its world feel quite sturdy.
185. pinball dreams. the gameplay is completely based on the simulation of a steal ball. the incredible machine. similar physics as in the pinball example. mainly gravity and some collisions.
186. Oblivion: simple gravity and object interaction physics on gameworld items.
187. Half Life 2 - Best simulation of real world physics I've seen in any game thus far
188. Half Life 2 springs to mind for excellent implementation of "havoc" physics, allowing different entities i.e crates or corpses to interact with each other. Bumping into each other and transferring forces between them.
189. Half Life 2 - whilst the physics is not perfect by any means (and what is), certainly the interactions between objects is fairly realistic
190. Halflife 2 : Bouyancy, Friction, Gravity, Acceleration, etc. Very realistic and enables proper physics puzzles such as weighing down one end of a seesaw to lift up the other. Gears of War : Corpses can be shot into different part and kicked realistically around the floor. Pointlessly gory, but quite amusing...
191. Half Life 2 - Havok Physics engine Oblivion - Havok Physics engine
192. Half life 2 - Not so much for the physics themselves as such (standard havok 2 if i remeber correctly) but for the correct weighting of objects. It gave them alot more feel and realism in the game world where as other games sometimes miss the mark. Objects feeling slighly too heavy or too light can break immertion but Half life 2 got it spot on.
193. Half Life 2
194. Plasma Pong. A great example of using fluid mechanics to make a big difference to gameplay (and it's fun and colourful). Half-Life 2 is good, including on the physics front, but some of the physics puzzles are a little contrived or obvious. (Cell Factor looks terrible; all the videos I've seen are basically hundreds of boxes flying around, which looks rubbish on many levels. Also, is it just me

or does the blood look like globs of jam?)

195. Half Life 2. Doom 3. Basically any 3D shooter in the last 4 years.
196. Half Life 2 –Havok physics? Used in a different and entertaining way. ES4: Oblivion – Used well in an engrossing setting, making it more believable and fun.
197. Half Life 2 started the physics revolution in games and remains quite true to life. Company Of Heroes uses physics accurately in a RTS enviroment and allows a great degree of interaction with the level.
198. Unreal Tournament 2, Half Life 2, Many racing games. They are good because they are good enough approximations to what you see in the real world.
199. HALF LIFE 2 (the most realistic simulation of real world physics that I have ever seen). TRESPASSER (a good attempt at simulating real world physics, at a time when computers weren't quite powerful enough to do the job totally convincingly).
200. halo, half-life 2 (havok), battlefield2
201. tribes 1+2 for movement, explosion reaction, etc. was a decent attempt of networking physics.
202. I particularly like many of the small games out there where some kind of rigid body or particle physics simulation are the core of the game. For example Impulse, Blueprint, Bridge Builder etc. The focus of those games are constructing structures , often using materials with different properties, springs or machines. The target is to build a structure that will perform a specific task when simulated.
203. Unreal tournament 2, Half-life 2
204. minesweeper (swept mine collisions) solitaire (playing cards move using Havok)
205. Half life 2, using modified Havok - I believe Valve wrote a custom set of far-field (i.e. broad phase) tests. It provided a very solid physics sim, that was integrated at the very core of the gameplay. It was also "fun".
206. Half Life 2 is the only commercial game I've played that has game physics which truly make a difference. Armadillo Run is truly awesome, and the old Bridge Builder Games were also really strong - both construction games where you are given materials to build a solution to a problem which is centred around a physics simulation. I imagine that a number of the leading driving sims have realistic car physics which feels just right to people that drive - but since I don't, I can't personally vouch for this.
207. HL2: gravity gun. TIM (most all games use parts of physics, gravity probably being the most common)
208. Half Life 2 Rainbow6:Vegas
209. Garrys Mod on Source - It's a physics sandbox and you can create almost anything you want with the objects from HL2. Armadillo Run - You have to create physics based courses to get a ball to an objective. I like them both because they are challenging and it is fun to experiment with the physics.
210. Best example of actual gameplay physics has to be Half-Life 2 - using the physical properties of various objects in puzzle solving (weighing down see-saw ramps with blocks, using floating barrels to raise things etc) and game combat was a big step and a real innovation.
211. Spaced Penguin and a few other casual games involving gravity and/or magnetism to steer an object towards a goal. It's a nice combination of simple physics with complex behaviour. Similarly, Cave Hunter and Moon Lander use gravity to complicate the gameplay. And obviously simulations of pool, golf etc.

212. Half-Life 2, Any source engine based game
213. Company Of Heroes - Buildings collapsing correctly depending on position of impact from artillery make the game feel a bit more lifelike as the remains fo the building can still be used and do not just disappear. Half-Life 2 - At times this game feels like a physics demo, as objects appear to carry weight, they can be manipulated and used, as opposed to being 'stuck to the ground'.
214. 1. Half Life 2 (Especialy from what I have seen of the up comming episode 2) 2. Oblivion
215. Deus Ex: Invisible War - Lighting (and shadows), collisions, stacking objects. They weren't hugely important to how the game played, but they massively added to immersion. Medieval 2: Total War - projectiles such as arrows and stones, and the impacts of charges. They make the battles seem far more organic
216. Half life 2 - the gravity gun. Americas army - bullet drop.
217. Half Life 2 - Fairly good use of gravity and physics, especially the manipulation of objects with the Gravity Gun weapon.
218. Half-life2 is too easy. I'll say oblivion (firing arrows and ragdoll were pretty good) and F.E.A.R (you could blow tin of paint off shelves etc).
219. Half Life 2 - Ragdolls, enviroment. Good because they're pretty accurate to real life.
220. Half-Life 2 - Havoc 2.0 physics. Because not only does the physical representation of the objects appear to be realistically done (unlike, say, Deus Ex: Invisible War), its inclusion is also an asset to the gameplay, especially by way of use of the gravity gun. Gran Turismo series. The realistic handling of the cars is the whole draw of the game.
221. Dead Rising: the crowd dynamics and collision calculations between literally hundreds of enemies on-screen at one time are superb and have to be seen to be believed. Oblivion: the mass and other properties of different items in the game are worked out very well, as is the way they interact with forces like gravity. Dropping an item while on a hill will inevitably end up in it running downhill in a very believable fashion.

13. Please rank the importance of game elements for you. You might want to choose your previously picked 'most liked genre' to relate these answers to a particular game style.

	Very important	Important	Nice to Have	Not Important	Who Cares?	N/A	Respons Average
3D Graphics and Image Quality	23% (55)	<b>40% (95)</b>	29% (70)	5% (11)	3% (6)	0% (1)	<b>2.23</b>
3D Sound and Audio Quality	12% (28)	33% (79)	<b>40% (94)</b>	10% (23)	5% (12)	0% (0)	<b>2.63</b>
Good Artificial Intelligence	39% (94)	<b>40% (96)</b>	18% (42)	2% (4)	0% (1)	0% (1)	<b>1.83</b>
Availability of Network Play	16% (37)	23% (55)	<b>41% (96)</b>	15% (36)	5% (11)	1% (2)	<b>2.70</b>
Size of Gameworld	17% (39)	<b>39% (91)</b>	32% (75)	10% (23)	2% (4)	1% (3)	<b>2.41</b>
Long Gameplay	27% (65)	<b>36% (86)</b>	28% (67)	8% (19)	1% (2)	0% (0)	<b>2.19</b>
Storyline in Game	29% (69)	<b>36% (86)</b>	22% (52)	9% (21)	4% (9)	0% (1)	<b>2.22</b>
Innovative Ideas	<b>41% (98)</b>	39% (92)	18% (43)	0% (1)	1% (3)	0% (1)	<b>1.81</b>
Cool Character Designs	14% (34)	28% (68)	<b>42% (100)</b>	13% (30)	3% (6)	0% (1)	<b>2.61</b>
Reality of Simulation	16% (37)	<b>30% (72)</b>	30% (70)	16% (38)	7% (16)	2% (4)	<b>2.67</b>
Usability of Interface	40% (96)	<b>43% (103)</b>	14% (34)	1% (3)	0% (1)	0% (1)	<b>1.78</b>
Entertainment Value	<b>76% (180)</b>	19% (45)	3% (7)	1% (2)	0% (1)	1% (2)	<b>1.29</b>
Educational Value	2% (5)	3% (7)	30% (70)	<b>36% (84)</b>	28% (67)	1% (3)	<b>3.86</b>
Force-Feedback Input	3% (8)	8% (19)	27% (64)	<b>31% (74)</b>	26% (60)	4% (10)	<b>3.71</b>
Customizability of Game	16% (39)	28% (66)	<b>41% (99)</b>	9% (21)	5% (13)	0% (1)	<b>2.59</b>
Availability and Price	23% (55)	<b>38% (90)</b>	29% (69)	8% (20)	2% (5)	0% (0)	<b>2.29</b>
<b>Total Respondents</b>							<b>239</b>
(skipped this question)							163

14. Do you think game physics is one of the elements worth rating as well?

		Response Percent	Response Total
Yes		92.4%	220
No		7.6%	18
<b>Total Respondents</b>			<b>238</b>
(skipped this question)			164

15. Please rate it on the same scale.

	Very Important	Important	Nice to Have	Not Important	Who Cares?	Response Average
Game Physics	29% (70)	41% (98)	27% (64)	2% (5)	1% (2)	2.04
<b>Total Respondents</b>						<b>239</b>
(skipped this question)						163

16. What kind of game physics have you experienced in games that you played? (check all that you know AND actually played in a game) - Note: For simplicity, we left out the physics of light (graphics) and sound (audio) here!

	Response Percent	Response Total
Collision Detection	94.5%	223
Vehicle Dynamics	94.9%	224
Flight Dynamics	72.9%	172
Projectile Dynamics	86.9%	205
Particle Systems	84.3%	199
Ragdoll Animation	95.3%	225
Breaking Bodies	76.3%	180
Swinging Bodies	86.4%	204
Velocity Based Friction	65.7%	155
<b>Simple Gravity</b>	<b>96.2%</b>	<b>227</b>
Day/Night and Weather Cycles	80.9%	191
Motion in Space (i.e. Solar System)	56.4%	133
Cloth Simulation	72%	170
Flowing Liquids	62.7%	148
Springs	71.6%	169
Other (please specify)	6.4%	15
<b>Total Respondents</b>		<b>236</b>
(skipped this question)		166

17. Have you heard of physics engines or physics accelerators?

	Response Percent	Response Total
Yes	97%	229
No	3%	7
<b>Total Respondents</b>		<b>236</b>
(skipped this question)		166

18. What do you think about the following statements regarding game physics?

	Yes	Maybe/Partially	No	Don't know	Response Average
Game physics is just another effect in a game.	21% (50)	34% (80)	<b>43% (100)</b>	2% (4)	<b>2.25</b>
The physics used in a game is "real" physics.	8% (19)	<b>43% (101)</b>	42% (99)	7% (16)	<b>2.48</b>
In a simulation game, it is important that the physics is accurate.	<b>72% (168)</b>	19% (45)	7% (17)	1% (3)	<b>1.38</b>
It would be cool to have a "reality slider" in a game.	<b>34% (80)</b>	28% (65)	29% (69)	9% (20)	<b>2.12</b>
There should be more and better "game physics" in games.	<b>62% (146)</b>	32% (75)	5% (11)	1% (2)	<b>1.44</b>
Game physics is just about Newtonian mechanics.	7% (16)	28% (66)	<b>49% (115)</b>	17% (39)	<b>2.75</b>
Games can teach the player something about physics.	<b>43% (102)</b>	40% (93)	15% (35)	2% (5)	<b>1.76</b>
The game physics is mostly invisible to the player.	10% (23)	35% (83)	<b>51% (120)</b>	3% (8)	<b>2.48</b>
It is OK, to fake physics in most games.	35% (82)	<b>45% (105)</b>	19% (45)	1% (3)	<b>1.87</b>
Game programmers should use real physics in games.	26% (60)	<b>55% (129)</b>	16% (38)	3% (7)	<b>1.97</b>
A physics simulation inside a game could be educational.	<b>49% (115)</b>	34% (80)	13% (31)	3% (8)	<b>1.71</b>
Some physics genres are not used in game physics.	<b>59% (138)</b>	20% (46)	4% (10)	17% (41)	<b>1.80</b>
Good game physics is important for the success of a game.	23% (53)	<b>43% (101)</b>	33% (78)	1% (3)	<b>2.13</b>
Better game physics makes a game more playable.	<b>42% (99)</b>	39% (92)	17% (40)	1% (2)	<b>1.76</b>
<b>Total Respondents</b>					<b>236</b>
(skipped this question)					166

19. If an artist makes an interactive multimedia installation about the topic of physics (for example it uses real physics data), would you consider that a "physics game"?

	Response Percent	Response Total
Yes, absolutely - if it's interactive, it's a game.	17.4%	41
<b>Maybe - but it depends on the how it's implemented.</b>	<b>73.3%</b>	<b>173</b>
No, definitely not. - art is not a game.	9.3%	22
<b>Total Respondents</b>		<b>236</b>
(skipped this question)		166

20. Roughly rate your knowledge in the following areas. (This is not a complete list of all the fields on purpose.)

	Expert	Average	Very Basic	Blank Sheet	Response Average
Physics	25% (58)	<b>52% (119)</b>	20% (45)	3% (8)	<b>2.01</b>
Programming	<b>33% (75)</b>	23% (54)	30% (69)	14% (32)	<b>2.25</b>
Engineering	15% (34)	<b>36% (82)</b>	36% (81)	14% (31)	<b>2.48</b>
Mathematics	31% (72)	<b>51% (116)</b>	15% (35)	3% (6)	<b>1.89</b>
Biology	9% (20)	<b>44% (101)</b>	37% (83)	10% (23)	<b>2.48</b>
History	14% (32)	<b>48% (110)</b>	29% (66)	9% (21)	<b>2.33</b>
Arts	16% (37)	36% (82)	<b>39% (89)</b>	9% (21)	<b>2.41</b>
Economics	3% (6)	34% (79)	<b>46% (106)</b>	17% (38)	<b>2.77</b>
Music	14% (32)	<b>41% (93)</b>	36% (81)	9% (21)	<b>2.40</b>
<b>Total Respondents</b>					<b>230</b>
(skipped this question)					172

21. Did you like to learn about science topics in school, at university or through self study?

		Response Percent	Response Total
Yes		77.7%	178
Somewhat		19.7%	45
No		2.6%	6
<b>Total Respondents</b>			<b>229</b>
(skipped this question)			173

22. Is some knowledge of science helpful in today's complicated technological world?

		Response Percent	Response Total
Yes, all the time		56.8%	130
From time to time		41%	94
No, almost never		2.2%	5
<b>Total Respondents</b>			<b>229</b>
(skipped this question)			173

23. What about physics? Is physics an important science that is important in day-to-day life?

		Response Percent	Response Total
Yes		48.5%	111
Somewhat		42.4%	97
No		9.2%	21
<b>Total Respondents</b>			<b>229</b>
(skipped this question)			173

24. Are scientists doing a good thing (or) for the world and our society?

		Response Percent	Response Total
Yes, all of the time.		14.8%	34
Yes, most of the time.		57.4%	132
Sometimes, sometimes not.		26.1%	60
No, with a few exceptions.		0.9%	2
No, never.		0.9%	2
<b>Total Respondents</b>			<b>230</b>
(skipped this question)			172

25. Is there a gender bias towards men in the sciences?

		Response Percent	Response Total
Yes		39.9%	91
No		60.1%	137
<b>Total Respondents</b>			<b>228</b>
(skipped this question)			174

20. How do you think mathematics, engineering and physics is perceived today by the general public?			
		Response Percent	Response Total
We love "Einstein", two thumbs up for science.		8.8%	20
Generally good but people see some black sheep as well.		33.8%	77
"Whatever!" would sum it up. But science is a recognized workhorse of the economy.		37.7%	86
Its there and produces results, but most of it is very suspect and problematic.		8.8%	20
Those lab-guys with their white coats will kill us all.		3.1%	7
Other (please specify)		7.9%	18
<b>Total Respondents</b>			<b>228</b>
(skipped this question)			174

27. What is the most interesting aspect of physics that you remember from school or university? If not about today's physics research – anything else total that you are really interested in?	
<b>Total Respondents</b>	<b>202</b>
(skipped this question)	200

1. Not really anything
2. The most interesting aspect of physics class would have to be the determination of certain missing variables when studying motion laws and formulas.
3. Cant think of any
4. Fluid physics. And today, energy and matter particles.
5. blank sheet
6. Physics of space flight and advances in space technology
7. I've always been interested in the physics of liquids.
8. the string theory or theory of all
9. I remember a small bug-looking robot which our teacher brought to us in school, which could simulate the 'real thing' pretty well. At least that's how i saw it then :)
10. I have always loved the physics of chemical reactions.
11. a teacher once showed us "ski stunt simulator"
12. ???
13. Emergent properties of complex physical interactions: surprising effects that are often best investigated using simulation.
14. space physics
15. Concentric Motion
16. Just watching two things collide at a high velocity. This lets us do things we cant do in real life.
17. Hmm, it would probably be gravity, and thats mostly it.
18. What goes up must come down :-P (how are we ever going to make an economical spacecraft with this kind of thinking...)
19. Interplanetary gravity
20. wormholes and whatnot



21. Magnets. Even though I didn't learn about them in high school (I'm in tenth grade) much, I still think that they are really interesting, and I always liked to learn about WHY things do what they do. Why does one molecule in an element give it a different color, a different texture, a different reaction to everything else, a different toughness, etc etc.
22. Fluid Dynamics look fairly interesting.
23. Learning how the physics are vital in our day to day life without realising it, and what life would be like without any sort of physics.
24. Nothing really...
25. magnetic water....
26. Well, I'm only in 10th grade, and so far, the only class that I've took that dealt with physics (so far) was Earth Science... which there was a lesson on some basic physics (motion, gravity, work, and I think there was some buoyancy somewhere in that class), but I'd like to take Physics in 11th grade (or 12th grade, if I don't have a hour clearing in 11th). Back on topic, I really liked learning about motion, I've always been slightly interested in physics.
27. Physics used for NASA-related jobs - such as space expeditions.
28. I think my decline into geekdom started with a library book I checked out in middle school about quantum physics. I've always been interested in space and time. I guess what actually is most exciting for me is the applications of newtonian physics, and the simple inventions that can be constructed when we understand forces, pressures, and gravity.
29. None.
30. I'm not sure i completely understand the question here so ill give it my best. The most interesting thing iv'e learned so far would be the electricity moves from place to place an how it provides us with everyday items like an Xbox 360... it does'nt seem like anything out there makes me extremely curious yet.
31. Not really.
32. I enjoy game design and consider most academic studies in light of how they can be used in games. Mechanics was no different. Electromagnetism I found to be fairly boring.
33. ummmmm blank sheet. im not in university or highschool to be learning physics but quantum physics is very interesting.
34. everything about physics is awesome
35. impacts, velocity vectors
36. Not really. maybe large scale physics, like planet attractions etc... [do I dare call it astro physics without having the slightest clue if its its name?]
37. Theoretical Physics.
38. i love all physics
39. I would probably say i really like the idea of inertia and spin on a sphere and how that interacts wih its surroundings.
40. The fact that physics made it into the game world is quite interesting.
41. don't know the english word for it .. its like when you try to lift sumtin heavy and put like a wood board under it and it is easier to lift it(ofcourse you can just lift a part of the thing if you do this) but this is good to know in today common situations(excuse my spelling i am from iceland) and also the heat-expanding thing, like when you heat some things the expand and shrink if they cool this is also important to know in today situations
42. Gravity and motion is probably all I've really been taught in physics. Today's physics reserach...

I'm really interested in liquid dynamics simulation, because that could be really fun in a game, but they haven't really made any good ones based on it so far.

43. Gravitation
44. Being able to model intuitively-correct forces was always interesting, if slow on paper. Not really aware of cutting-edge physics.
45. centrifical/centripital forces
46. Huge amounts of interaction in games such as Cell Factor: Revolution (Demo is called Cell Factor: Combat Training).
47. rfer
48. The wonderful feeling of gaining a glimpse into the workings of the universe. Showing how light behaves as both a particle and a wave is really rather neat.
49. .... i don't remember... =P
50. springs always amazed me about how they worked, also magnetism but i dont think thats really physics related.
51. fluid simulation
52. Building and testing a trebuchet in physics alevel, varying different parameters.
53. Tehoretical Physics sometimes catches my imagination.
54. aaaaah i dont really know.. im not interested in anything particulary (sorry for the spelling)
55. Fluid Dynamics, Rigid Body Simulation
56. It is all equally interesting to me. I really don't know much about todays physics research.
57. Video taping different materials being dropped from changing heights.
58. Not Really
59. I know little on physics (at the moment).
60. Not sure.
61. (sound)waves, collisions
62. Newtons Laws of Motion
63. no NOT AT THE MOMENT
64. Not many from school, we just crunched equations mainly.
65. Fluid dynamics (esp. real time simulation)
66. I enjoyed the mechanics of projectiles.
67. nothing really.
68. The Newtonian physics where the most interesting thing in school, at least for me.
69. still in school, so new aspects everyday.
70. MEMs and Nanotechnology. Definitely the future!
71. Laborations containing magnetics, electricity and so on.
72. Fluids physics. Nothing on the radar that I am really interested in.
73. ...
74. Havent really studied physics. No.
75. the proof that we cannot plan anything everything random we can only predict with a margine of error. this also strenghts my atheist beleifs that if evering it so random how can ther be a god controlling it to be so random.
76. Space is always interesting, and in Sweden we have just had our first Astronaut in space :) From school I vaguely remember a lot of experiments with springs, and some stuff about gravity.

77. Never took physics, and was always interested in learning. I will be taking physics class soon. However, from a friend of mine...she would tell me how Roller Coasters work, and they had a class trip learning about the mechanics and psychology of the structures.
78. Don't have a standard education. Selftaught. Of course there is fun to watch how they are frequently changing their views in extrasolar astronomy.
79. Quantum Physics and Quantum Physics
80. Nothing from school/university, I wasn't interested back then. Right now, I'm interested in everything I missed, with special mentions to mechanics and quantum physics.
81. Relativity Newtonian Mechanics
82. particle physics, multiverse theory, I guess. This is a weird section, I must say...
83. I personally liked the idea of travelling at high velocities (close to speed "c" scenario and the perceptions of time).
84. cloaking device
85. The dynamics of space flight
86. Light is quite interesting, but I'm not all that interested, all the equations and things really bug me.
87. fiber optics, public-key encryption
88. I loved Feynman...everything about Feynman and Quantum Electrodynamics. I don't think Newtonian physics is all that fun, but once you get past that it is fascinating. I have my eye on all of the string theory stuff...we'll see how that pans out.
89. I like to read about the stars, planets, blackholes etc. electronics, and magnetism was also cool.
90. black holes were pretty interesting. coming soon is wireless recharging of batteries, which I think will be a massive step forwards in technology.
91. can't think of any at this moment
92. Currently interested in quantum mech. (basic, layman level), also nuclear physics, both after reading Oppenheimer bio "American Prometheus"
93. remember from school: Special Relativity research: nothing in particular
94. Reality.
95. As I just started going to college online to get a bachelors in Game and Simulation Programming, I am sure I will be learning much more about physics than I do now.
96. Nuclear and quantum physics.
97. Quantum physics.
98. Gravitational dynamics, particularly planetary systems - some nice, self-contained problems with interesting associated mathematics. Interesting new research on the radar - detection and study of gravitational waves.
99. Grand Unified Theory.
100. probably particle/quantum stuff because i never really understood it.
101. I preferred chemistry. You could blow stuff up in chemistry.
102. aertbdstdfszer
103. I loved fluid dynamics and rigid body physics. Energy formulations of physics laws are nice too. Today: anything concerning computational advances, energy conservation, storage and generation.
104. Nanotechnology, Superconductor research, particle/quantum research & string theory
105. I truly enjoyed the study of gravitational effects in space on large masses, especially planets, stars, and asteroids. Kepler is my hero. ;)

106. How physics applies to everyday life, the sort of "Oh so THAT'S why that happens".
107. Well, I've just read a book called "Parallel Universes," and I found it all to be quite fascinating, the fact that our universe might be just a membrane in a sea of other universes.
108. Launching a rocket from Earth and landing on the Moon and returning. The math involved is staggering. Today, I'm following the different private companies interested in space travel, such as Virgin, Blue Origin and Armadillo.
109. Modern physics is beyond my level of knowledge, though I am interested in nanomachines and nanomanipulation techniques which could be used in genetic engineering and microsurgery. I am also interested in the idea of synthesizing matter from energy and the related concept of transporter beams. And of course FTL space flight. In school I enjoyed starting a fire with a lens, splitting light into colors, and vector physics involving velocity and acceleration.
110. I know that things happen, but knowing why they happen and how it will affect other things is what intrigued me about physics.
111. I was interested to hear that classical Newtonian dynamics break down at speeds near the speed of light, and at the quantum level. More recently I hear a lot of talk of "dark matter" and "dark energy" that supposedly makes up a large percentage of the universe. I hear the Russians are looking to further develop nuclear propulsion systems for long-distance spacecraft. With a refined system they claim to be able to travel to the outer reaches of the solar system in weeks, rather than years. Very cool.
112. Fusion
113. I'm particularly interested in the unknown questions of physics like time and its relation to matter.
114. The whole gamout from classic Newtonian through to the most bizarre quantum theory, and how the history of science changes based on current soci-political conditions. On the radar? Sure, quantum stuff and cosmology.
115. Being a math major, the mathematical aspects were interesting. Being in biomedical research, reasearch in medical imaging is interesting.
116. Not so much 'academic' physics, but astrophysics and various dimension theory (time theories, string theories, standing wave stuff) is pretty fascinating to me right now. I find myself very interested in the stuff that is trying to define the why and how of reality because in some ways it bridges two of my favorite things; philosophy and science.
117. Superconductors
118. I don't know.
119. The interactions between wires carrying current was interesting to me. Harmonics and collisions were also interesting.
120. Though I only studied physics in high school, I liked learning calculus in university and seeing how it would apply to the study of (mostly Newtonian) physics. I enjoy reading about quantum physics, and am excited about upcoming experiments with the Large Hadron Collider, materials science and continued space exploration.
121. Quantum computing and nanotechnology in general.
122. Can you tell me in three simple sentences what a Higgs Boson is and why it's so important to find one?
123. N/A
124. Black holes, superconductors, superfluids - just the cool stuff :)
125. Don't remember my education with physics

126. Quantum Computing...
127. I found most of the ridiculously theoretical stuff quite interesting. Wave partical duality, for instance.
128. At the minute, lasers and their potential applications.
129. Not really
130. I think the medical uses of physics interested me the most in school, while today nanotechnology (is that to do with physics?) and military uses (avionics in planes and the like) are what interest me the most, I'd say.
131. Newtonian physics is the most interesting aspect of physics. I'm particularly interested in quantum mechanics and the various implications of that (for example quantum computing), also some of work in particle physics is very interesting.
132. Quantum Physics and Grand Theorey of Everything being on the horizon.
133. The fate of the universe, especially black hole dynamics etc.
134. Finally getting a grip of vectors! Topping the national board in physics shortly after that!
135. Discovering how Newton's Laws of Motion contradict most of Einstein's work and vice versa, which led me to realise the world and universe is a lot more complex than most people realise. I'm particularly interested to see what becomes of String Theory.
136. Artificial intelligence and robotics (combined) SETI
137. Nature of the universe and basic mechanics
138. How the universe was made, what keeps it all together. What don't we know yet and various theories.
139. The great big donut in Western Europe, NASA space program, satellite communications
140. My favourite subject was Space. Most interesting prospect is Fusion.
141. Don't really know :/
142. Can't think of anything.
143. Physics put the universe in perspective for me, and made me feel endlessly small, the laws incorporated in such a large space is mind-boggling, but interesting none-the-less
144. Space, mostly.
145. Quantum physics - exploring the limits of the universe.
146. Multiple body problem and particle physics in astronomy were interesting. Also magnetism and electro-mechanical systems.
147. Space , particles , radiation
148. Subatomic particle physics, closely followed by quantum wave mechanics.
149. Fluid dynamics was very interesting to me at university. Today I think that device physics and quantum computing research are very exciting for the future.
150. experiments
151. I am personally involved in hardware acceleration of rigid body and fluid dynamics for games physics.
152. I write game physics so I'm more interested in Newtonian stuff.
153. My physics prof was insane.
154. Atom Teleportation, Quantum theory
155. Most interesting aspect from school seemed to be mechanics because it seemed more "applicable" to real life. Right now I'm mostly interested in Feynman's research, but just for the fun of it.

156. At college - inertia
157. Generally physics dealing with time and space; I find research into the "final frontier" fascinating.
158. theory of relativity, the 3 laws of motion.
159. Relativity and quantum mechanics.
160. Newton's laws and optics. Now, I'm interested in using geometry and topology to unify physics without dark energy or other nonsense.
161. magnets, lasers etc
162. EVERYTHING, er... I mean Physics today? yeah, i'm thinking I can provide a physical proof to the Prime Number Theorem AND, what the Big O really is.\*grin\* I am in love with Relative SpaceTime and a tool I'm working on called Relative Point Math for reducing the Time complexities of any process over time. Yeah, I think I found something. I am very interested in any process involving time and space and people.
163. I enjoyed the E&M, learning about how energy travels in waves and about the relation between electric and magnetic fields. I think nuclear and high energy physics is a very interesting research field today.
164. Teleportation
165. the notion that most of what we see, and even don't see (yet) is predictable to some degree of accuracy. Quantum computing life on other planets
166. Don't remember anything interesting.
167. Electronics
168. The principles of relativity and time dilation are a particularly fascinating area of physics.
169. The transmission and conversion of energy.
170. Gravity, its fascinating and complex, also sub atomic physics are interesting.
171. Mechanics, because it was useful for simulations in games and suchlike. Most modern physics research is beyond me, but quantum computing, if it counts.
172. lasers with changing colors are pretty interesting
173. The switching on of the large particle accelerator this year is very exciting. Personally, I am interested in soil mechanics and simulation of granular materials as part of my PhD. Physical simulations such as the Finite element analysis and the discrete element method are particularly exciting, and both are rapidly growing.
174. I learned little interesting about physics at school but independently of that I have a basic interest (in the grand theories without knowing the ins and outs of them) of string theory, quantum mechanics and especially the work being done on travelling faster than light.
175. I only studied science subjects up to GCSE as I went on to do literature based subjects, however I remember two experiments we did, one with a Van der Graff generator (which is always fun) and one cliché experiment about parachutes. However I made a point of keeping up with science and I've found I've been quite interested in advances in light manipulation technology and organic computing
176. Topics about space, time and quantum theory without a doubt. They are far more interesting than dull stuff about how fast object X moves on surface Y, or how much wire A stretches when object B is dropped.
177. anomalies within physics on a very small scale (i.e electrons under certain circumstances seeming to hit 2 or more places at once).
178. Electricity, creating electric circuits to do various jobs.

179. My interests this year are Atmospheric & Ocean Physics, and Astrophysics. Particle physics, etc, is interesting, but I prefer more 'tangible' aspects of the subject. New, large projects like the LHC and James Webb Space Telescope are interesting prospects for new science. (I'm currently doing a 4-year Physics course at uni)
180. Astronomy, cosmology. String theories and such
181. The creation of black holes and time travel.
182. Newtonian Mechanics + Calculus = rocks... Levitating vehicles are an area of research I'd like to see materialise...
183. I have always been interested in gravity (I own a piece of software that enables me to study the gravitational interactions of objects in the Solar System).
184. Space travels
185. navier simulations, and having alot done for you through API's like novodex
186. Since my current for-fun programming project is a rigid body simulation with proper resting contact, I'd have to say that that's what I consider interesting at the moment.
187. Time. What is "time"?
188. blackholes and sex
189. I can't remember many interesting topics from school, blowing stuff up in chemistry lessons was much more fun :). I'm not really in interested in physics itself per se, but rather what an understanding in it enables us to do e.g. space exploration etc.
190. fundamental particles, nanotech
191. electricity, nano, quantum mechanics
192. Manipulation of light.
193. Theory of Everything/cosmology are things I'm very interested in. LHC is going to be interesting too.
194. The most interesting was relativity; the bit I was best at was waves and springs. Generally I find maths and biology more interesting than physics.
195. Bouyancy
196. Mostly I enjoyed the physical aspects - but I also like the "What if..." senarios where we would come up with ideas for dooms day devices.
197. It was pretty cool to see how tonnes of things can just be described using Newton's second law, or the conservation of energy. But that could just be because I generally focus on the mathematical side and it makes learning formulae easier. I don't really keep up with todays physics so all I could do on the second question is mangle some buzzwords which I don't really understand.
198. Particle theory/chaos theory
199. Weightlessness has always facsinated me somewhat.
200. I was interesting in the solar sail. Unfortunate that it was lost.
201. All of it was interesting to me.
202. Particle and sub-nuclear physics along with quantum theory have always interested me and I've enjoyed reading popular science books on the subject.


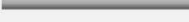



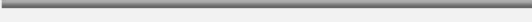


26. Movie physics: How accurately do movies show science, science facts and the work of scientists? (Consider your typical entertainment tick and leave out 'SciFi' in your assessment, since it is by definition 'fiction' as well as 'DocuS', because they are by definition 'true'.)

	Response Percent	Response Total
Science is always accurately shown	0.9%	2
More often accurately than incorrectly shown	7%	16
It's shown half the time true and half of the time false	24.7%	56
<b>Most of the time it is shown inaccurately and very seldom correctly</b>	<b>57.7%</b>	<b>131</b>
Science is never the real thing in movies	9.7%	22
<b>Total Respondents</b>		<b>227</b>
(skipped this question)		175

27. Current game physics is mostly dynamics (i.e. mechanics). Of the following physics fields, which one do you think could be interesting to implement or use in a video game (assuming it can be done). Or indicate that you think it is impossible to use in a video game.

	Yes, very cool	Maybe just a bit	No, boring	Huh, how would one do that?	No idea, my Physics background isn't that good.	Response Average
Thermodynamics (i.e. Heat Engine, Kinetic Theory)	<b>52% (116)</b>	29% (64)	8% (17)	5% (12)	6% (13)	<b>1.64</b>
Electromagnetism (i.e. Electrostatics, Maxwell's Equations)	<b>45% (100)</b>	33% (73)	8% (17)	4% (9)	10% (22)	<b>1.67</b>
Relativity (i.e. Special and General Relativity, Field Equations)	<b>31% (68)</b>	29% (65)	11% (24)	13% (29)	16% (35)	<b>2.08</b>
Relativistic Mechanics (i.e. Scattering Theory)	<b>36% (80)</b>	23% (50)	8% (18)	7% (15)	26% (57)	<b>1.80</b>
Statistical Mechanics (i.e. Entropy)	24% (52)	<b>31% (68)</b>	16% (35)	7% (16)	21% (46)	<b>2.09</b>
Advanced Classical Mechanics (i.e. Chaos Theory)	<b>43% (94)</b>	26% (57)	9% (19)	10% (22)	12% (27)	<b>1.84</b>
Quantum Mechanics (i.e. Schroedinger Equation)	22% (48)	21% (47)	13% (29)	<b>23% (50)</b>	21% (47)	<b>2.47</b>
Nuclear Physics (i.e. Dirac Equation)	21% (46)	23% (50)	15% (32)	18% (40)	<b>23% (51)</b>	<b>2.39</b>
Computational Physics (i.e. Weather Simulations)	<b>70% (154)</b>	21% (47)	6% (13)	1% (2)	2% (5)	<b>1.37</b>
<b>Total Respondents</b>						<b>223</b>
(skipped this question)						179



20. What should the experience of real-physics in video games be? (choose any that you feel are good ideas or contribute your own idea)			
		Response Percent	Respos Total
Configuration Experiences: A "reality-slider" implementation which allows to switch between "arcade" mode and "simulation" mode (at the expense of playability).		39.1%	84
About-Game Experiences: Online-community based structures such as forums to assist in a possibly steep learning curve.		30.7%	66
About-Game Experiences: Documentation about the physics used in the simulation (formulas, graphs, etc.).		23.3%	50
About-Game Experiences: A realism-rating or similar indicator (similar to an age-rating) about the physics on the package of the game.		27%	58
In-Game Experiences: The ability to accurately measure and monitor simulated parameters used in the game (i.e. using overlay graphs).		26%	56
<b>In-Game Experience: Immersion is enhanced and the feeling to be part of a believable alternate-reality is heightened.</b>		<b>84.7%</b>	<b>182</b>
In-Game Experience: Game physics capabilities of the software are such that actual science experiments are possible inside the game.		51.6%	111
Other (please specify)		16.3%	35
<b>Total Respondents</b>			<b>215</b>
(skipped this question)			187

21. Replacing pseudo-physics in games with more accurate implementation might have an impact on the player. To what degree could you agree with the following statements?				
	Agree	Neutral	Reject	Response Average
If physics is not done purely for entertainment, players might actually learn something while gaming.	<b>41% (89)</b>	40% (86)	19% (42)	<b>1.78</b>
Less "movie physics" in games make sciences more real and "positive".	26% (57)	<b>44% (96)</b>	29% (63)	<b>2.03</b>
It is necessary to improve the quality of VR artifacts (i.e. through better physics) because we use them so much in games.	<b>47% (102)</b>	41% (88)	12% (27)	<b>1.65</b>
It would be good to have "quality" physics and simulations - otherwise my mind gets all "screwed up".	31% (68)	33% (73)	<b>36% (78)</b>	<b>2.05</b>
Pseudo-physics in games create a kind of useless "Folk Physics". We better not use that too much in games.	18% (39)	38% (83)	<b>44% (96)</b>	<b>2.26</b>
A great educational resource - the video games - is possibly wasted at a time when better science education is needed.	27% (59)	<b>40% (87)</b>	33% (71)	<b>2.06</b>
Adding some real science in a game should attract more girls to this kind of entertainment genre.	2% (5)	40% (88)	<b>57% (125)</b>	<b>2.55</b>
<b>Total Respondents</b>				<b>219</b>
(skipped this question)				183

22. Do you think scientists could use a computer game to perform research and actually come up with some new finding, if the game physics would be implemented very good?			
		Response Percent	Response Total
Yes		33.6%	74
Maybe		38.2%	84
No		28.2%	62
<b>Total Respondents</b>			<b>220</b>
(skipped this question)			182

23. If you have any unique comments or conclusions about what game physics and its future might mean, please type them down below. For example, if a game implemented accurate physics, how would your gaming style change?	
<b>Total Respondents</b>	
<b>100</b>	
(skipped this question)	
302	

1. Video game physics may eventually get realistic enough to be used for science, but won't be accurate enough for a long time.
2. While realistic physics simulations are great for many games, it's important to realize that realism is not always the most entertaining option. Over-the-top (hyperrealistic) physics can be more effective, as can underpowered (dampened) physics. Sometimes, it may even be attractive to move away from the curve of physics that behave the way we expect (even hyperrealistic physics in games is somewhat predictable) to unnatural, unpredictable physics. It all depends on the type of experience that is intended by the game designers.
3. physics games should be a great way to make what we can't make in real life. We can't change any rules in real life so why are we trying to copy that on a system which can show us other worlds with other physics. (sorry for my english)

4. One of the big reasons for me is the opening up of new possibilities for games. That and allowing for the interesting simulation of possibilities that cannot be interactively explored in the real world. This includes game physics that wildly differ from real-world physics.
5. Physics in games is a wonderful thing that lets people simulate what would happen in a game, maybe i'm just saying this because I love watching a ragdoll fall down the stairs, so I don't have to break my neck, but who knows the future of game physics, if coders and programmers keep going at the pace they are now, then we'll know something amazing is going to come out of it.
6. Hm, accurate physics would mean much more replayability in my view, and replayability is better.
7. Toribash has (feature is there, but disabled) a feature where you can switch between modes of physics. One was called "newton.tbm" and the other was "hong kong.tbm" - Newton very much portrayed extremely excellent physics (very hard to rip off a head in this mode) and Hong Kong is just "movie physics".
8. I strongly believe that as physics gets more popular in games input devices will have to change to become more than 2-dimensional which I think will be a very good thing and give further possibilities for game physics.
9. I personally like physics in video games because it adds a fun variable. If everything had to be run from a script, nothing could ever be a surprise, but if you can flip a car over a house and land on your friend's tank, it's much more fun.
10. Better physics implementations in games could lead to better playability. I know, as a player, I love to see when, for instance, I kick a box and it rolls some only to settle a foot or two away. Things interacting the way you expect them in games is a far more engaging, entertaining experience in my mind.
11. Hm, well, one last comment would be that I love games that implement physics into it, and I'd like to see more of them. I also think if a game was to advance in realism of physics, games could launch into a new era... or something like that.
12. I believe I could learn much, as I have already learned a bit from playing physics games.
13. I don't think my gaming style would change at all. I just play the best games out at the time. If I like, I keep. If I don't like, I return. Better game physics just seems like a way to make games more realistic for the player and if they were implemented many would probably see the change.
14. Many of the questions were misleading I thought. What we consider the study of physics is just a model of the real world. The equations are never completely accurate and they are all approximations. To what degree you approximate depends on your domain. So the questions of whether game physics is "real," I feel the answer is yes even if the models they use are very simple. However, if a game is attempting to accurately portray a system in real life, such as a race car simulation, then the models must be fairly accurate in order to convince the player. As far as whether physics should be used more in games, it entirely depends on how it is implemented. If it is implemented simply for the sake of having it, then the game-play is unchanged and the cost of developing the physics is wasted.
15. More accuracy would probably result in me spending more time experimenting.
16. I believe that physics are a very cool and exciting aspect in gaming. The more advanced the physics get in the games the happier I am.
17. I LOVE games with/based on physics. If all games had good physics I would be a happy camper.
18. Game physics should come naturally and be implemented into the game in such a way that you feel as if you are directly affecting the world around you.
19. Well if a game lets you put a wall that would change the entire gameplay (a good thing).

20. aefradda
21. Having games implementing accurate physics would mean that the physics simulation would (for obvious reasons) be virtually identical (e.g. one game using a value for pi accurate to ten decimal places and another a value accurate to eight wouldn't make a whole lot of difference) which would increase the \*collective\* quality of immersion in games.
22. Good Luck =)
23. one of the main reasons people play games is to escape reality. if there are accurate physics implemented into more games, with more submersive control styles (i.e. the Wii control sticks) it would give people another reason to play games - to play out their fantasies, sick or otherwise, in a controlled environment, where noone gets hurt, noone gets offended, and players can turn it off when it gets too much for them. this could possibly be a great opportunity to relieve stress/anger
24. If a game implemented more accurate physics, I would be more creative while battling and probably have more fun.
25. The game would have more replay value, (I.E. killing enemies in different ways).
26. I would probably want more of those kinds of games and play them all the time
27. There seem to be some things in this survey suggesting that there's connection between female gamers and physics engines. I don't see this. I think the only thing keeping girls off games is games being made by boys. It's a male dominated industry and in order to get female gamers in the industry, there should be female game makers. There may be a connection to female gamers I don't see and I apologise for my ignorance if there is. And I would like to note that I feel that physics engines should be a medium for experimenting with new kinds of play. Thus, there should be more resources for independent companies (meaning make them cheaper) because the big companies won't always try new things.
28. Not much.
29. I just think that if physics code doesn't directly impact the gameplay experience for the better, it shouldn't have to be included just for the sake of it.
30. If physics were done and implemented very well into a game, I think that game would be much more popular than if the developers just put a "fake" physics engine into it.
31. Finding solutions would become a more relaxed and intuitive process (eg instead of finding a key to a locked door, being able to take the door off its hinges or setting the door on fire)
32. The possibilities presented by fluid dynamics are astounding. There would be some great gameplay opportunities there.
33. Comments on my answers: partially/maybe-answers were used as "partially" not as "maybe", rated relatively. I hope I could help.
34. I don't know about the gaming style, but I know that the physics in games will probably be used a lot more when "real-looking" liquidsimulations, particlesystems and other forms of physics are applied to a game. For example, you maybe will have to extinguish fire with water.
35. Action games might become very hard. Sports game very intense.
36. KEEP UP THE (GAME)PHYSICS! I like them, because they make a game feel "real" and they are a very cool feature, too! But for all the people with not that good computers: put in a "realityslider" sorry for my bad english: I am german :P
37. It would definitely change it a lot. Half-Life 2 has shown me that physics in games can radically change the way you play. For example, you can block off a doorway (at least temporarily) with a couple of boxes or something, to buy you some time to get away. In a game without physics, you just can't do that, unless it's scripted, which is no fun.

38. Games are about fun and entertainment and doing stuff you cannot do in real life. "Real Physics" can sometimes be the opposite.
39. I love learning new things, and if it's a game type of thing, I may learn something without intent or learn what physics is about without having any pressure from teachers and other people. I can just have fun.
40. Each of pseudo physics and real one have their place in gaming. Real physics means a hard learning curve for new players of a genre, which could mean less gamers in a sector. On the other hand, the "happyness" of mastering this kind of game is far greater. Pseudo-physics game based have almost no learning curve (on the physics point), but the fun is given somewhere else or by great sensation (like a good speed feedback, even if not realistic on the physics part).
41. \* the term 'real' physics has not been defined by this survey. \* game physics and 'real' physics (the study of) will never be the same, in game physics everything is approximate integer or floating point representations. It only has to look real. \* physics deployed in games should be done so for entertainment purposes, to make better games. Realism is a part of that but it's not everything.
42. If a game implemented accurate physics, I would stop playing games and just go play sports.
43. game physics should be unobtrusive, unless there is a specific physical effect that the designer wishes to create (e.g. zero-gravity). At a macro level, anything more subtle than Newtonian physics, fluid dynamics, and other large scale effects are probably not going to be noticed. Physics should be the slave of the game designer, not vice-versa. In the future, game physics will benefit more from the refinement of existing techniques, instead of the inclusion of new ones (e.g. there's no need to add quantum dynamics to game just yet, when there's still so much to do with )
44. The level and nature of physics depends entirely on the game genre. A game like Super Mario 64 needs only rudimentary physics, while a realistic World War 2 simulation (like the Forgotten Hope 2 mod being developed for Battlefield 2) would require more sophisticated physics, and could be improved with an engine with more perfected physics. On the other hand, RTS games don't really have much a need for complex physics at all.
45. I think that physics are a good idea, but I also love more extreme and impossible games ( such as F-Zero, the ludicrously fast and ridiculous racing game that many of us love, and Trackmania Sunrise ) and if we saw less of these I would be sad :( Okay, thanks, rather long though
46. Add a great plot, off load physics to a card, push the CPU and GPU hard. Developers have so much now in terms of CPU, GPU and networking bandwidth. Do something with it. Stop selling Doom, quake clones. Add tools - do not fear the end users and lock up the game to be play only. Physics in WW2 games like Red Orchestra should be a big hint - end users can make great things physics wise - if you give them the tools. If you have a few options to get the tools to make a new game - let Mac, Linux and XP users in on the physics - not just \$\$\$ for some XP only DRM lock in. As a developer do your homework on cross platform physics then code.
47. I think there is a difference in games that are done purely for entertainment and education/entertainment or serious games. I think games are underutilized in education but it is something to be careful of on a mass market scale. So I guess what I mean is that the degree of accuracy in your game is dependent on the audience and the intention of the game. If it is for entertainment it should be as realistic as the gameplay can handle, if it is edutainment it should be accurate but not to the point that it is not fun and engaging. If it is for research it shouldn't be a game at all but a realistic real time simulation that is as accurate as it possibly can be. I would be more inclined to get a game I knew would represent physics in an educational way if I knew the learning curve and content provided beforehand. I think any step to add fun to learning is a good one.

48. Yes, my game style would change. If military shooters (like Call of Duty etc.) had accurate physics the entire gameworld would change. Suddenly you could not hide behind a wooden fence to avoid bullets( they would shoot though) Grenades would blow doors of hinges. Artillery would make holes in the ground, demolish buildings etc. I do not believe that ANY game can be a "Trojan-horse-carrier" for teaching physics.
49. if everything in a game world was destructible, there would be more possibilities for real world cause and effect to take place. So, for example, you have many enemies in a building, either go in and take them out, or just destroy the building.
50. Accurate physics is good in simulations, eg sports sims. Accurate physics is less important in other games, as long as what is happening looks realistic, i don't care if it is accurate or not.
51. As development costs rise, I feel game physics should help game design by not having to deal with interactions on an object to object basis. Having realistic physics for the sake of it could complicate things for the player (fully destructible environment). I hope the augmentation of processing cores will be put to good use in physics rather than having to invest in a dedicated card for it.
52. When it comes right down to it, games are about having fun. If a game isn't fun, I don't want to play it. And for half of the games I play, immersion is a key concept, since I'm playing a representation of reality. The less I feel like I'm controlling a character on screen the better, and having in-game objects and assets react realistically to do what I do adds to the immersion of the game, thus heightening the the enjoyment of the game. I can appreciate good physics (and hate bad) but I certainly do not want the game to go out of the way to show them off. Details are everything in a game, and the more the player sees that is "out-of-place" in the real-world the more they'll feel like they're playing a game instead of riding across a sunny field with the grass blowing gently in the breeze, or charging up Normandy beach with bullets zipping all around them. Immersion.
53. More accurate physics means more immersive gameplay thus increasing the playability factor. Consumers are more likely to buy a game if they can really dive into the game, take WC3 for example, truly immersive - though physics isn't really a huge part of it but the sheer size of the game makes it more immersive, thus more appealing.
54. Accurate physics can destroy a potentially good game moment. Take the large Scarab in Halo 2. If the characters had "real world" physics, they would be thrown clear of the platform attached to the Scarab removing a very fun and entertaining moment.
55. I think the perspective of this study is wrong. People learn when they are entertained. Fiction (and fictional physics) are not wrong or bad, they are a way of understanding the world from a different, hypothetical angle. Most videogames do not take place in the real world or have gameplay where the player can manipulate physics, so it's silly to expect them to have realistic physics. However games (or software toys, the non-scored variety) which are intended to be educational, games which have puzzles based on manipulating a physical system, can be cool and might do lots of interesting things with realistic physics, but there's no point criticizing people who don't want to do that. If someone wanted to promote better science in games the thing to do is either make a game themselves or offer a grant to a game development company making that kind of game.
56. For a lot of games, extremely accurate physics are not necessary, and may even detract from the enjoyment. For example, many games are space-themed. If you had accurate physics, flying around in space would be tedious and very boring. There would be no fancy explosions or laser beams, and no sound. So, in some instances it is more fun to bend the rules a bit. There are some genres however that might benefit from a slight enhancement of the in-game physics. Flight simulators or

racing games should, and probably do already, have fairly detailed physical calculations, but even they sometimes lack in some areas. Platform games might be more instructive if they focused on giving a more accurate physics model. For example, if a Mario-type side scrolling game had things like friction, real gravity, aerodynamic forces, then kids might develop a better understanding or interest in these topics.

57. Lighting in games might become more realistic - more realistic effects like caustics and volumetric stuff as realistic clouds
58. Improved physics means more possibilities wch IMO translates to more immersive gameplay.
59. Accurate physic....while possibly educational...might actually remove the fun of the "game." If a game isn't fun, people won't play it and the purpose is defeated. Most games are played (and movies watched) with "escapism" in mind. There is nothing wrong with having inaccurate physics as long as it contributes to the fun of the game. A couple of comments regarding your survey....some of your scales are not good. For example there is nothing between Expert and Average when talking about levels of knowledge in certain sciences. What is Average? The average person walking down the street? What is Expert? A PhD? Also, as I'm sure you'll find out, open ended questions are a pain to code. Open ended questions which are asked as Yes/No's are worthless. Finally, there is definitely a biased tone throughout your survey, almost like you have an axe to grind against inaccurate physics engines in games. If a game is a simulation, then fine....an accurate physics engine is appropriate. If a game is a platformer, such as Mario, you could really care less is gravity is accurately portrayed. You may wish to re-do this survey at some future time with a more unbiased tone. Good luck with your research.
60. Gaming style changes depending on the game being played. If the way the physics simulation was implemented in a game was changed people would change their play style to whatever works in that particular game.
61. A more interactive environment always encourages me to explore more and play or experiment with the environment around me. Incorporating more or just other aspects of physics than classical mechanics could enhance this aspect.
62. Part of the appeal of some games is that acts that are very difficult in real life, such as driving a Formula 1 car at race speed, are possible for untrained people. Making the simulation more accurate in such cases could be counterproductive. On the other hand, many in-game objects currently "feel" wrong, so improved simulation of everyday objects could greatly help players' suspension of disbelief.
63. I would probably start gaming a lot more. Truly accurate physics and a completely immersive control system would allow you to do all those things that you've thought of in the past. You know, the "wouldn't it be cool if..." ones.
64. In Half life 2, you can solve puzzles with physics
65. Accurate physics have the potential to increase the immersion and general fun-factor of games greatly, but I'm not so sure if games with accurate physics will have a place in scientific research. As for games with good physics being used for education, that is an exciting, if uncertain, prospect.
66. A game with accurate physics would lead me to plan traps and simple machines for both offense and defense which is different from my more "run and gun" style. Also there is a lot of possibilities in a physics puzzle type of game, think rude-goldberg style puzzles.
67. It's not how good, but how useful and inherent to the game/scenario the technology is; that matters. I think I enjoyed the first Test drive games much more that the newer gtr games etc Similarly I played SIm City and Sim City 3000 for hours on a nend - the SIm CIty 4 left me cold

— it gets too complicated after some time...

68. accurate physics could result in a much more immersive experience as you can interact in more ways with the environment. You would be less limited with what tasks you can perform and how you could perform them
69. If the quality of physics and AI in games increase I would be forced to change the way I play due to having to treat the world more as I would in real life.
70. Physics for games should be about improving the dynamic behaviour of the game first - but they're only applicable in games that contain some basis of reality. Second Life with 'Real Physics TM' would be highly boring.
71. Shooters could completely change. Games like Unreal Tournament would disappear and be replaced by Americas Army-type games. We should strike a balance between the two, have enough of each to have a complete experience. Sometimes realism can be tiring and uninspirational. At other times it is essential.
72. Learning to actually USE physics in something like HL2 really opens the game up for you... If you're out of ammo and you see some just out of reach, you can grab it with your gravity gun. If more accurate physics were included, anything would be possible. You could hurl your weapons at enemies to stun them, wash away bloodstains in stealth games by using water, that kind of thing.
73. Experimenting would be encouraged and new methods of completing tasks would be possible. You no longer have to find a key to open the door, you smash it open with your trusty crowbar. etc Though, if physics were to be fully implemented, there would be a need to have a better way to control what's going on on the screen. A keyboard and a mouse wouldn't be enough.
74. The act of gaming would require more thought. With the implementation of physics comes a living and breathing environment that will create its own problems, rather than having them scripted, this will offer immersion on a scale unseen
75. ...
76. Real physics will never be implemented and any attempts to do so will look really pathetic and - well physics doesn't sell as well as blood and gore.
77. Please separate effect physics (eye candy) with game play physics (player is affected) Also visit Bullet Physics forum: <http://bulletphysics.com>
78. Game physics are meant to serve the gameplay. If game physics can represent other physical phenomena without becoming prohibitively slow and add something valuable to the experience it will start to show up in games. (the only two things that spring to mind here are deformable/fracturing bodies and smoothed particle hydrodynamics)
79. This is an interesting subject, and although I fully support a realistic physics engines in gaming today, I don't share the author's implied belief that this is \*vital\* to today's gaming. I would fully support a committment by game companies to include a more realistic physics model in their products, but I can think of many other areas in game design that could benefit at least as much (if not more) from serious attention by the game's designers.
80. Right now the simulations we tend to do are often more than relaistic enough for the input devices we use to control them. Its no coincidence that driving games used physcis long before chanter based games.
81. My immersion in the game would be higher, and thus my gaming experience more profound and entertaining.
82. As a game designer/programmer, I can say that a little physics goes a long way. Just like art, game physics needs to be consistent, not accurate. It is easy for people to pick up on the particular



game-specific rules and limitations.

83. Currently, we are limited by the number of off screen objects or particles we can accurately simulate or "move" in a reasonable amount of time. The reason for this is the fixed bit width that is used when processing objects not related. It programs waste heat can be expressed in time and currently, we are FAR from optimal. The future physics engine will leverage this to reduce relative calcs to the appropriate and smallest bit width necessary for the movement of values(i.e physics, weather, electricity, gravity, light, sound, heat, liquids, etc) in parallel time.
84. I find it hard to make catch-all statements regarding physics in a game, much like any other part (rendering, audio, etc) 'Video Game' is an incredibly general term and wraps a wide array of experiences. I don't think that you can make the argument that video games, as a whole, need to move towards more realistic physics. There will always be games where fake physics is enough. The important thing to remember about games is that gameplay is king. It's not enough to render a highly detailed and realistic world with highly detailed and realistic physics - that just gives you a real-estate sales tool. Video games need gameplay, and gameplay a video game creator comes up with will have varying requirements in terms of rendering, physics, audio, etc. Realistic physics definitely has a place in the future of video gaming, but it's not a be-all end-all. In 20 years we might have incredibly accurate physics simulations in certain games, but I'm willing to bet that 2D side scrollers will still be around, with constant velocity falling and physically impossible jumps.
85. From doing this questionnaire I can see that you want to implement real physics for you PhD, but I simply think its is not needed or required and not enough people have super computers. Games give the illusion of a real world without it being real, games which have tried to use real physics have failed to ship.
86. Accurate game physics would demand too much hardware. I can't see ROI. If it won't make games more fun, then it's worthless. Only scientists would benefit, and that's not how the world works.
87. Better physics would open up the game world to more creative solving of problems and overcoming of obstacles. Sadly, there are a number of issues which prevent this. In many cases, physics only apply to certain objects in the game and not others. Or, your means of interacting with the objects is limited (such as Half-life 2, where you can lift cars but cannot tear doors off their hinges, or pull/push other characters)
88. The point of games is to have fun. If an accurate physics model is fun, then good. But not all games need it; not all are attempting to be accurate simulations of the real world, and nor should they be.
89. Physics are already quite accurate, but improving can only make things better. It might \*sometimes\* be good to have slightly unrealistic physics if the game would be greatly enhanced - maybe explosions hurl objects a lot further than normal?
90. The more realistic a game should be, the more realistic the physics. Should physics be more appropriate used artistically in a game then they should be able to be inaccurate. Physics are walking the same path Graphics did just a few years ago. Not everything is about realism.
91. (On the above point (32), scientists already use physics simulations to do physics. I don't know if a generally programmed videogame would help scientific research, though). Personally, I feel that gaming physics is generally a good thing; for example, I loved Plasma Pong. I don't think it's necessary at the moment to bring it beyond something that increases immersion and playability (which it always has, since even Mario has physics). It's possible that computers get faster and ubiquitous gaming physics changes gaming. Physics generally has the capacity to improve the flow

of gameplay more so than better graphics, but it can still be used as a gimmick with no apparent improvement to the genre or game itself (Cell Factor is an example of this, from what I've seen of the videos).

92. It would certainly be made more realistic.
93. I'd spend hours playing with stuff in an interactive room...
94. The single, most important contribution of "real world" physics in games is that real world logic can be used to solve problems. This is infinitely more appealing than trying to determine the single solution that the developer wants the player to use. Real world physics opens up the possibility of MULTIPLE solutions to game puzzles, and this is very important to me.
95. if it is a game, there is only one important word: "fun". physics can be "fun" because it can give you a different playing experience, but that's all there is to it. the math itself is totally unimportant - just as no one cares if an animation is based on vectors, matrices or whatever.
96. It's all about the game really, rather than the physics. If an accurate depiction of physics in-game makes the game more fun, then stick as much in as possible. If wildly inaccurate "movie physics" is more fun, then that's what we should have in-game. Games are/should be a form of entertainment. Once a game-like experience starts turning too obviously educational or evangelical about something the fun factor decreases rapidly.
97. It would depend entirely on the game. You could have a puzzle game that uses, say, fluid flow (in fact, we have one with that mercury blob game that came out recently); obviously adding physics to FPS games can change the way you play (I can blow up this building and make it fall on my enemy; I can knock this person over with a stream of water). On the other hand, real physics also adds chaos and unpredictability to games, and this can sometimes mean that the games are less satisfying (you may want to be able to retry a section and know what does and doesn't work)
98. I believe that in game Physics have the potential to make videogames become more realistic and improve their quality. They may also help to add variety to the genre - like how Half Life 2 added the gravity gun to create variety during combat and puzzles. It also has the ability to allow players to become more creative and experiment with the game world and by extension real world physics, like Garry's Mod has done.
99. For me, the thing that most appeals about physics in games is when it is replicating what we have. You are given a problem, and you know that if a solution would work in reality it will work here. Artificial blocks on this are a big immersion breaker in gaming. Real physics would allow you to use the game environment more, and with more certainty. However, there is a place for crazy movie physics too. It may not be educational, but it's often fun. And that is what games are ultimately about.
100. If we take 'realistic' physics too far, then many game types would become much less fun to play. Who would want to play a game where every bullet is fatal/crippling? We want games where the world feels real, but a game where we can still do things that we cannot do in real life.

## F.2 Game Physics and Physicists

### Introduction

Dear Reader,

this survey is part of my PhD research into "Game Physics in Video Games"; the general direction of my research is briefly summarized on this page of my website: <http://www.ferzkopp.net/joomla/content/view/55/15/> - feel free to review it before starting this survey.

The survey questions are designed to get an insight from practicing scientists - Physicists to be precise. For example it will try to answer the question how the use of physics and physical realities in video games is perceived by YOU. The survey will also explore some of the ideas you might have on how one could bridge the gap between real science and entertainment with video games as a medium.

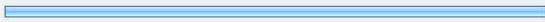

You will encounter 33 questions about physics, science, video games and their physics. Some questions will be about your specialization in physics and your views about the video games in general. You will probably take you about 15-20min to work through the survey - if interrupted, you may come back at any time and the survey will continue where you left off.

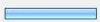
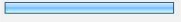
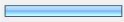


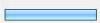
Please note: You do not need to have played computer games to participate!!!

While this survey is about video games, your answers are relevant even if you never played a computer game in your life - I encourage you to contribute your ideas and comments. If some of the questions you encounter seem to be not applicable to you or if you can't answer them, just skip the question and continue with the next one. At the end of the survey you can provide general written feedback.

Before you start, I want to assure you that this survey is purely for research and remains anonymous (i.e. no login, name or phone number is required). Please note however, that some of your written comments might be quoted in a publication such as my thesis. If you have any questions about this survey, the research, have suggestions or are interested in the results when they are published, feel free to write me an email: [aschiffler at ferzkopp.net].

Thank you very much for your time and assistance in my research! Andreas Schiffler



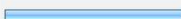

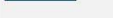

1. What's your gender?		
	Response Percent	Response Count
Male 	93.3%	42
Female 	6.7%	3
	<b>answered question</b>	<b>45</b>
	<b>skipped question</b>	<b>0</b>

2. Which age group are you in?		
	Response Percent	Response Count
20 years or younger 	15.6%	7
<b>21 to 30 years old</b> 	<b>28.9%</b>	<b>13</b>
31 to 40 years old 	20.0%	9
41 to 50 years old 	6.7%	3
51 to 60 years old 	13.3%	6
61 years or older 	15.6%	7
	<b>answered question</b>	<b>45</b>
	<b>skipped question</b>	<b>0</b>

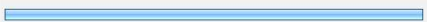
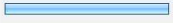
3. In Physics, what is your specific field of study or your current research focus? (if there are more than one, please list them all)	
	Response Count
	<b>answered question</b>
	<b>33</b>
	<b>skipped question</b>
	<b>12</b>


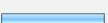
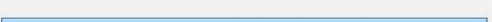
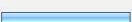

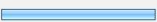
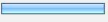
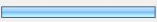


1. Space Physics, Upper Atmospheric Studies
2. General Relativity, Quantum Optics, Solid State Physics
3. mechanical physics.
4. Studying Mechanical Engineering at university this year, and have A-level Physics, and Maths with Mechanics.
5. mechanics
6. Physical modeling for sound synthesis and musical composition, with also an interest for computer animation
7. Applied Physics - Electromagnetics
8. experimental hadron physics
9. Cosmology
10. Condensed matter theory Statistical physics
11. Theoretical atomic physics, ab-initio calculations of atomic properties, spin-angular integration techniques
12. Theoretical physics (QED, QFT) Solid state physics
13. plasma physics
14. Atmospheric spectroscopy
15. Computational materials science Computer simulation Electronic structure theory and computations.
16. Solid-state physics Computational physics Optics biophotonics
17. Nuclear Power
18. Solar physics, K-12 pedagogy
19. fluid-structure interactions

20. Optics
21. Nanophysics (2x)
22. Astrophysics
23. granular materials
24. Elementary particles, History of ideas in physics
25. Theoretical Physics, Cosmology, Materials, Artificial Intelligence, Biophysics
26. Computational Acoustics
27. Nanostructured semiconductor single spectroscopy
28. Taking an undergraduate degree in Physics
29. high school physics (general)
30. Electromagnetics
31. Currently physics education, originally experimental low temperature physics

4. Categorize your kind of Physics work or research (choose the closest match or add your own category).			Response Percent	Response Count
Theoretical Physics			15.2%	5
Computational Physics			12.1%	4
<b>Experimental Physics</b>			<b>30.3%</b>	<b>10</b>
Applied Physics/Engineering			12.1%	4
Physics Education			18.2%	6
 Other (please specify)			12.1%	4
<b>answered question</b>				<b>33</b>
<b>skipped question</b>				<b>12</b>



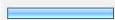
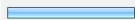






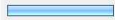
5. Roughly rate your knowledge in the following areas. (This is not a complete list of all the fields of study on purpose.)						
	Expert	Average	Very Basic	Blank Sheet	Rating Average	Response Count
Physics	<b>78.1% (25)</b>	21.9% (7)	0.0% (0)	0.0% (0)	1.22	32
Programming	31.3% (10)	<b>37.5% (12)</b>	25.0% (8)	6.3% (2)	2.06	32
Engineering	<b>34.4% (11)</b>	<b>34.4% (11)</b>	25.0% (8)	6.3% (2)	2.03	32
Mathematics	<b>50.0% (16)</b>	<b>50.0% (16)</b>	0.0% (0)	0.0% (0)	1.50	32
Biology	0.0% (0)	<b>53.1% (17)</b>	34.4% (11)	12.5% (4)	2.59	32
History	9.7% (3)	<b>41.9% (13)</b>	38.7% (12)	9.7% (3)	2.48	31
Arts	6.3% (2)	25.0% (8)	<b>50.0% (16)</b>	18.8% (6)	2.81	32
Economics	6.5% (2)	25.8% (8)	<b>45.2% (14)</b>	22.6% (7)	2.84	31
Music	22.6% (7)	22.6% (7)	<b>48.4% (15)</b>	6.5% (2)	2.39	31
<b>answered question</b>						<b>32</b>
<b>skipped question</b>						<b>13</b>

6. Do you (or have you) published or publicized anything related to Physics?		
	Response Percent	Response Count
Yes 	71.9%	23
No 	28.1%	9
<b>answered question</b>		<b>32</b>
<b>skipped question</b>		<b>13</b>

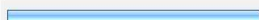
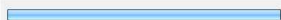

7. If you answered Yes in the previous question, where did you publish? (check all that apply)		
	Response Percent	Response Count
Scientific Journal (peer reviewed) 	87.0%	20
Scientific Book (focused audience) 	17.4%	4
Conference Poster or Proceedings 	82.6%	19
Educational Book (school or university) 	21.7%	5
Book (general audience) 	8.7%	2
Journal (general audience) 	26.1%	6
Patent Application 	17.4%	4
Blog or Website 	26.1%	6
Radio, TV or Film 	8.7%	2
Other (please specify) 	4.3%	1
<b>answered question</b>		<b>23</b>
<b>skipped question</b>		<b>22</b>

8. Please rate the following statements about Physics research.							
	Fully agree	Somewhat agree	Indifferent	Partially disagree	Totally disagree	Rating Average	Response Count
My research needs to be accepted within the science community through publications.	46.7% (14)	26.7% (8)	23.3% (7)	3.3% (1)	0.0% (0)	1.83	30
A scientist can get valuable input through comments from outsiders or other fields.	56.7% (17)	20.0% (6)	16.7% (5)	6.7% (2)	0.0% (0)	1.73	30
It is important to experiment; Trial-and-Error is a viable method for doing research in Physics.	50.0% (15)	40.0% (12)	6.7% (2)	0.0% (0)	3.3% (1)	1.67	30
Research results can only be valid when all the methods and instruments used are very precise.	26.7% (8)	33.3% (10)	6.7% (2)	23.3% (7)	10.0% (3)	2.57	30
The results of my science work needs to be communicated to the general public.	43.3% (13)	26.7% (8)	16.7% (5)	6.7% (2)	6.7% (2)	2.07	30
Fancy visualizations could be used in educating the public about Physics, even when they are not 100% accurate.	36.7% (11)	26.7% (8)	20.0% (6)	13.3% (4)	3.3% (1)	2.20	30
Popular media influences the public perception of Physics negatively.	10.0% (3)	30.0% (9)	30.0% (9)	30.0% (9)	0.0% (0)	2.80	30
There is a crisis in science education; we need to educate our children better in Physics.	56.7% (17)	30.0% (9)	6.7% (2)	6.7% (2)	0.0% (0)	1.63	30
	<b>answered question</b>						<b>30</b>
	<b>skipped question</b>						<b>15</b>

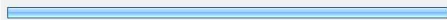
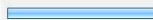
9. Does your research generate any numerical data, digital measurements or simulated datasets? Estimate the approximate amount of storage you would need in ONE year. (B = Bytes)

	Response Percent	Response Count
>1TB 	3.6%	1
100 GB - 1 TB 	3.6%	1
10 GB - 100 GB 	17.9%	5
<b>1 GB - 10 GB</b> 	<b>21.4%</b>	<b>6</b>
100 MB - 1 GB 	10.7%	3
10 MB - 100 MB 	7.1%	2
1 MB - 10 MB 	3.6%	1
100 KB - 1 MB 	7.1%	2
10 KB - 100 KB 	3.6%	1
< 100 KB 	3.6%	1
No data 	17.9%	5
<b>answered question</b>		<b>28</b>
<b>skipped question</b>		<b>17</b>


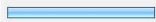
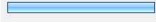
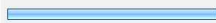

10. Is current Physics research important, in that it affects people's day-to-day life?

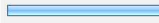
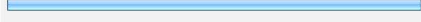

	Response Percent	Response Count
Yes 	42.9%	12
<b>Somewhat</b> 	<b>46.4%</b>	<b>13</b>
No 	10.7%	3
<b>answered question</b>		<b>28</b>
<b>skipped question</b>		<b>17</b>

11. Is there a gender bias towards men in Physics?

	Response Percent	Response Count
Yes 	75.0%	21
No 	25.0%	7
<b>answered question</b>		<b>28</b>
<b>skipped question</b>		<b>17</b>

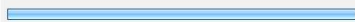

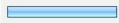




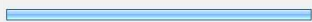

12. How do you think mathematics, engineering and physics is perceived today by the general public?		
	Response Percent	Response Count
People love "Einstein" and think "Two thumbs up for science!" 	7.1%	2
The mood is generally good, but people see some black sheep as well. 	25.0%	7
"Whatever!" would sum it up. But science is a recognized workhorse of the economy. 	25.0%	7
<b>Science works and produces results, but people think most of it is very suspect, incomprehensible or problematic.</b> 	35.7%	10
People think "Those lab-guys with their white coats will kill us all."	0.0%	0
Other (please specify) 	7.1%	2
	<b>answered question</b>	<b>28</b>
	<b>skipped question</b>	<b>17</b>


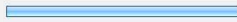
13. Movie physics: How accurately do movies show science, science facts and the work of scientists? (Consider your typical entertainment flick and leave out 'SciFi' in your assessment, since it is by definition 'fiction' as well as 'Docus', because they are by definition 'true'.)		
	Response Percent	Response Count
Science is always accurately shown	0.0%	0
More often accurately than incorrectly shown	0.0%	0
It's shown half the time true and half of the time false 	25.9%	7
<b>Most of the time it is shown inaccurately and very seldom correctly</b> 	70.4%	19
Science is never the real thing in movies 	3.7%	1
	<b>answered question</b>	<b>27</b>
	<b>skipped question</b>	<b>18</b>

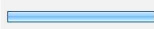







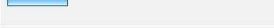
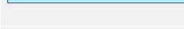




14. What's the best science movie you can recall?		Response Count
	<i>answered question</i>	21
	<i>skipped question</i>	24

1. Evolution
2. 2001 a space odyssey (3x)
3. The elegant universe
4. Star Trek
5. an inconvenient truth
6. Dante's Peak
7. Contact
8. Our Mister Sun Bell Labs Film
9. west side story
10. A Beautiful Mind
11. Frau im Mond
12. Les palmes de M. Schultz
13. None
14. Doomsday Gun
15. Mindwalk

15. Do you think there is a greater need to communicate with the public about Physics and Physics research? Take a broad approach (i.e. consider education, presenting research, debates in the scientific community).		
	Response Percent	Response Count
Yes, there is definitely a huge need and large gaps to fill. 	59.3%	16
Yes, but the 'need' is not that big or urgent. 	14.8%	4
Maybe we could do some more in particular areas. 	18.5%	5
No, because there is not really a 'need', but more wouldn't hurt either. 	3.7%	1
No, what for? We are doing just fine as it is. 	3.7%	1
	<i>answered question</i>	27
	<i>skipped question</i>	18

16. Have you experienced a multimedia art installation involving Physics or using data from a real Physics experiment?		
	Response Percent	Response Count
Yes 	51.9%	14
No 	48.1%	13
<b>answered question</b>		<b>27</b>
<b>skipped question</b>		<b>18</b>

17. How much video game playing do you do? (consider any interaction with a video game, even if it is on a cell phone)		
	Response Percent	Response Count
Almost every day 	14.3%	4
A few times a week 	14.3%	4
Once a week 	10.7%	3
A few times a month 	7.1%	2
A few times a year 	14.3%	4
Basically never 	39.3%	11
<b>answered question</b>		<b>28</b>
<b>skipped question</b>		<b>17</b>

18. If you play any video games, what kind of game are they? (select all that apply)			
		Response Percent	Response Count
Early action games (i.e. arcade emulators)		25.0%	5
Maze games (i.e. PacMan, Bomberman)		10.0%	2
2D Platform games (i.e. Super Mario Bros.)		20.0%	4
Fixed Shooters (Galaga, Area 51)		5.0%	1
Slide shooters (i.e. Raiden)		10.0%	2
Competitive fighting games (i.e. Street Fighter II)		10.0%	2
Beat 'em Up fighting games (i.e. Double Dragon)		5.0%	1
3D Platform games (i.e. SuperMario 64)		10.0%	2
First-Person shooters (i.e. Quake, Half-Life)		45.0%	9
Third-Person shooters (i.e. Resident Evil)		30.0%	6
Survival Horror (i.e. Silent Hill)		15.0%	3
Turn based RPG and strategy games (i.e. The Elder Scrolls)		15.0%	3
Real-time RPG and strategy games (i.e. Age of Mythology)		15.0%	3
Massively Multiplayer Online Role-playing Games (i.e. Ultima Online, EVE)		5.0%	1
Text Adventures (i.e. Zork)		0.0%	0
2D Graphics Adventures (i.e. King's Quest)		0.0%	0

Hidden Object Puzzles (i.e. Minesweeper)		35.0%	7
Character Control Puzzles (i.e. Lemmings)		10.0%	2
Construction Puzzles (i.e. TIM, Crazy Machine)		5.0%	1
Child Education Games (i.e. Carmen Sandiego Series)		0.0%	0
'Serious Games' for Training/Advertising (i.e. America's Army)		15.0%	3
Programming Games (i.e. Core War, Robot Battle)		10.0%	2
Other (please specify)		15.0%	3
		<b>answered question</b>	<b>20</b>
		<b>skipped question</b>	<b>25</b>

**19. What gaming platform do/did you use to play your games? (check all that apply)**


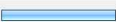
	Response Percent	Response Count
Game Console (i.e. Xbox, PS2)		5
<b>Personal Computer</b> (i.e. PC, Mac)		17
Handheld Console (i.e. Gameboy, PSP)		2
Other Portable Device (i.e. Cell Phone, iPod)		5
Commercial Arcade Console		0
Other (please specify)		2
		<b>answered question</b>
		<b>19</b>
		<b>skipped question</b>
		<b>26</b>

**20. Have you ever programed a computer game (or part of a game) yourself?**

	Response Percent	Response Count
Yes		6
<b>No</b>		15
		<b>answered question</b>
		<b>21</b>
		<b>skipped question</b>
		<b>18</b>

21. Whats the best game you know? (if you know any)		Response Count
	<i>answered question</i>	15
	<i>skipped question</i>	30

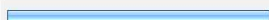
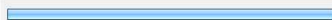
1. GTA: San Andreas
2. The Sims
3. MS Pinball Arcade
4. Xenosaga
5. Flight Simulator II
6. The legend of Zelda: The Wind Waker
7. Civilization 3
8. Outcast, Far Cry, No one lives forever 1/2
9. Civilization
10. Metroid
11. X-Plane
12. Lemmings

22. Do you think there a gender bias towards men when video games are made?		
	Response Percent	Response Count
Yes 	80.8%	21
No 	19.2%	5
	<i>answered question</i>	26
	<i>skipped question</i>	19

23. Without surfing over to wikipedia.org, consulting a book or asking an expert, please answer in a one or two sentences: What do you think "game physics" is?		Response Count
	<i>answered question</i>	26
	<i>skipped question</i>	19

1. The mathematics necessary to simulate animated motion, and the interaction of characters within a '3d' environment.
2. I'm guessing. The physics simulation that is included in games.
3. An "engine" that determines how objects interact in the game world.
4. Tests of physical intuition
5. Simulation of the physical laws (movement, dynamics, etc) of the "world" of a game

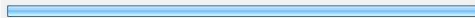

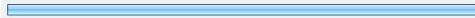





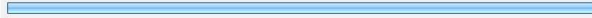
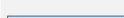
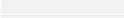
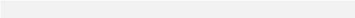
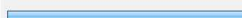
6. No idea. Maybe related to the theory of games in economics?
7. showing how physical laws work through modelling in a game situation
8. no idea
9. you have to solve a problem with tools. I do like 'the incredible machine' for that. Use a natural phenomenon to succeed.
10. Rules for simulating the behaviour of virtual objects appearing in computer games
11. Games which use realistic physics in the motion of objects and electromagnetism.
12. The accurate, as filtered by visual and auditory perception, depiction of the laws of motion, electromagnetics, optics, thermodynamics, materials properties, and similar realizable macroscale phenomena.
13. Simulation of movements in the game according to some laws of Physics
14. Something to simulate external conditions or similar
15. When things obey the laws (or certain) laws of physics in a video game, i.e., an tool falling to the ground with gravity, water flowing in a stream, a character's hair waving with the wind...
16. The incorporation of Physics principals/models, such as gravity, into a video game to make it life like. game physics is the result of programing in the game that makes game environment behave in a manner that is realistic. falling, recoil, conservation of momentum in explosions etc.
17. Game physics is a fundamental part of all games (To be more precise, it is part of the game 'engine'). It is supposed to simulate and re-create a virtual environment resembling reality with the highest accuracy possible using our contemporary technology.
18. The simulation of reality for objects in the game world using physical laws.
19. It is maybe the algorithms used to simulate the physics behind the "world" used in the computer game.
20. physics laws needed to be considered in programming or playing games
21. The physical "laws" used within a computer game (e.g. Newtonian, explosions, fluid dynamics, or for even background animations)
22. My son is a game producer/programmer for Blizzard Entertainment (World of Warcraft, etc.) He is not a physicist but has consulted physics books in his work, so I guess that is "game physics." However, he tells me that game producers never let faithfulness to physics laws stand in the way of producing a game with the desired qualities.
23. Using games to teach physics.
24. Application of dynamics to motion of objects in games.
25. Physics involved in simulation needed to make game look mor real

24. Have you heard of "physics engines" or "physics accelerators" in relation to PCs or video games?			
		Response Percent	
		Response Count	
Yes		44.4%	12
No		55.6%	15
		<b>answered question</b>	<b>27</b>
		<b>skipped question</b>	<b>18</b>

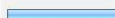
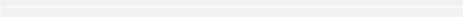

25. Based on your definition of game physics above and your knowledge or perception of video games: What's your assessment of the following statements regarding game physics?						
	Yes	Maybe/Partially	No	Don't know	Rating Average	Response Count
Game physics is just another effect in a game.	20.0% (5)	<b>44.0% (11)</b>	20.0% (5)	16.0% (4)	2.32	25
The physics used in a game is "real" physics.	4.0% (1)	36.0% (9)	<b>44.0% (11)</b>	16.0% (4)	2.72	25
In a simulation game, it is important that the physics is accurate.	<b>60.0% (15)</b>	28.0% (7)	12.0% (3)	0.0% (0)	1.52	25
It would be cool to have a "reality slider" in a game.	<b>44.0% (11)</b>	24.0% (6)	0.0% (0)	32.0% (8)	2.20	25
There should be more and better "game physics" in games.	<b>44.0% (11)</b>	<b>44.0% (11)</b>	0.0% (0)	12.0% (3)	1.80	25
Game physics is just about Newtonian mechanics.	20.0% (5)	32.0% (8)	<b>40.0% (10)</b>	8.0% (2)	2.36	25
Games can teach the player something about physics.	<b>52.0% (13)</b>	40.0% (10)	4.0% (1)	4.0% (1)	1.60	25
The game physics is mostly invisible to the player.	<b>44.0% (11)</b>	24.0% (6)	20.0% (5)	12.0% (3)	2.00	25
It is OK, to fake physics in most games.	<b>44.0% (11)</b>	20.0% (5)	32.0% (8)	4.0% (1)	1.96	25
Game programmers should use real physics in games.	32.0% (8)	<b>60.0% (15)</b>	4.0% (1)	4.0% (1)	1.80	25
A physics simulation inside a game could be educational.	<b>64.0% (16)</b>	32.0% (8)	4.0% (1)	0.0% (0)	1.40	25
Some physics genres are not used in game physics.	<b>56.0% (14)</b>	24.0% (6)	0.0% (0)	20.0% (5)	1.84	25
Good game physics is important for the success of a game.	12.0% (3)	32.0% (8)	<b>36.0% (9)</b>	20.0% (5)	2.64	25
Better game physics makes a game more playable.	12.0% (3)	<b>52.0% (13)</b>	20.0% (5)	16.0% (4)	2.40	25
					<b>answered question</b>	<b>25</b>
					<b>skipped question</b>	<b>25</b>



26. You might have heard of, seen or even played the game 'Half Life 2' (HL2). Which of the following physics elements do you think are part of this game? - Note: For simplicity, we left out the physics of light (graphics) and sound (audio) here. - If you don't know anything about HL2, just SKIP this question.

		Response Percent	Response Count
Collision Detection		80.0%	4
Vehicle Dynamics		80.0%	4
Flight Dynamics		0.0%	0
Projectile Dynamics		80.0%	4
Particle Systems		80.0%	4
Ragdoll Animation		40.0%	2
Breaking Bodies		60.0%	3
Swinging Bodies		60.0%	3
Velocity Based Friction		40.0%	2
<b>Simple Gravity</b>		<b>100.0%</b>	<b>5</b>
Day/Night and Weather Cycles		20.0%	1
Motion in Space (i.e. Solar System)		20.0%	1
Cloth Simulation		0.0%	0
Flowing Liquids		60.0%	3
Springs		40.0%	2
	<b>answered question</b>		<b>5</b>
	<b>skipped question</b>		<b>40</b>

27. If an artist makes an interactive multimedia installation about the topic of physics (for example it uses real physics data), would you consider that a 'physics game'?

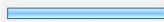
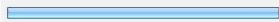
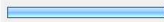
		Response Percent	Response Count
Yes, absolutely - if it's interactive, it's a game.		18.5%	5
<b>Maybe - but it depends on the how it's implemented.</b>		<b>77.8%</b>	<b>21</b>
No, definitely not. - art is not a game.		3.7%	1
	<b>answered question</b>		<b>27</b>
	<b>skipped question</b>		<b>18</b>

**28. Current game physics is mostly dynamics (i.e. mechanics). Of the following physics fields, which one do you think could be interesting for the player when integrated into a video game (assuming it can be done) -or- indicate that you think it is impossible (or extremely difficult) to use in a video game.**

	Yes, very interesting	Maybe, but wouldn't add much	No, boring topic	Impossible! How would one do that?	Rating Average	Response Count
Thermodynamics (i.e. Heat Engine, Kinetic Theory)	57.7% (15)	19.2% (5)	23.1% (6)	0.0% (0)	1.65	26
Electromagnetism (i.e. Electrostatics, Maxwell's Equations)	60.0% (15)	28.0% (7)	8.0% (2)	4.0% (1)	1.56	25
Relativity (i.e. Special and General Relativity, Field Equations)	53.8% (14)	38.5% (10)	3.8% (1)	3.8% (1)	1.58	26
Relativistic Mechanics (i.e. Scattering Theory)	48.0% (12)	36.0% (9)	12.0% (3)	4.0% (1)	1.72	25
Statistical Mechanics (i.e. Entropy)	30.8% (8)	38.5% (10)	19.2% (5)	11.5% (3)	2.12	26
Advanced Classical Mechanics (i.e. Chaos Theory)	65.4% (17)	26.9% (7)	3.8% (1)	3.8% (1)	1.46	26
Quantum Mechanics (i.e. Schrodinger Equation)	40.0% (10)	36.0% (9)	12.0% (3)	12.0% (3)	1.96	25
Nuclear Physics (i.e. Dirac Equation)	26.9% (7)	34.6% (9)	26.9% (7)	11.5% (3)	2.23	26
Computational Physics (i.e. Weather Simulations)	73.1% (19)	23.1% (6)	3.8% (1)	0.0% (0)	1.31	26
	<b>answered question</b>					<b>26</b>
	<b>skipped question</b>					<b>19</b>

29. What should the experience of 'real-physics' in video games be? (choose any that you feel are good ideas or contribute your own ideas)		
	Response Percent	Response Count
Configuration Experiences: A "reality-slider" implementation which allows to switch between "arcade" mode and "simulation" mode (at the expense of playability).	72.0%	18
About-Game Experiences: Online-community based structures such as forums to assist in a possibly steep learning curve.	24.0%	6
About-Game Experiences: Documentation about the physics used in the simulation (formulas, graphs, etc.).	52.0%	13
About-Game Experiences: A realism-rating or similar indicator (similar to an age-rating) about the physics on the package of the game.	56.0%	14
In-Game Experiences: The ability to accurately measure and monitor simulated parameters used in the game (i.e. using overlay graphs).	44.0%	11
In-Game Experiences: The ability to accurately measure and monitor simulated parameters used in the game (i.e. using overlay graphs).	44.0%	11
In-Game Experience: Immersion is enhanced and the feeling to be part of a believable alternate-reality is heightened.	52.0%	13
In-Game Experience: Game physics capabilities of the software are such that actual science experiments are possible inside the game.	56.0%	14
In-Game Experience: Incorporate actual live data of physical measurements or simulations in a game.	36.0%	9
Other (please specify)	8.0%	2
	<b>answered question</b>	<b>21</b>
	<b>skipped question</b>	<b>20</b>

30. Game entertainment usually uses very limited, inaccurate simulations. Replacing pseudo-physics in games with more accurate implementations might have an impact on the player. To what degree could you agree with the following statements?					
	Agree	Neutral	Reject	Rating Average	Response Count
If physics is not done purely for entertainment, players might actually learn something while gaming.	57.7% (15)	30.8% (8)	11.5% (3)	1.54	26
Less "movie physics" in games make sciences more real and "positive".	34.6% (9)	42.3% (11)	23.1% (6)	1.88	26
It is necessary to improve the quality of virtual reality (VR) artifacts (i.e. avatars and objects) through better physics because players use them so much in games.	42.3% (11)	57.7% (15)	0.0% (0)	1.58	26
It would be good to have "quality" physics and simulations - otherwise the players perception of reality gets all "screwed up".	46.2% (12)	38.5% (10)	15.4% (4)	1.69	26
Pseudo-physics in games create a kind of useless "Folk Physics". We better not use that too much in games.	40.0% (10)	44.0% (11)	16.0% (4)	1.76	26
A great educational resource - the video games - is possibly wasted at a time when better science education is needed.	57.7% (15)	34.6% (9)	7.7% (2)	1.50	26
Adding some real science in a game might attract more girls to this kind of entertainment genre.	15.4% (4)	38.5% (10)	46.2% (12)	2.31	26
Adding some real science in a game might attract more girls to this kind of entertainment genre.	15.4% (4)	38.5% (10)	46.2% (12)	2.31	26
The addition of physics that can be 'experienced' in a game might entice younger players to consider sciences as a career choice.	46.2% (12)	42.3% (11)	11.5% (3)	1.65	26
<i>answered question</i>					26
<i>skipped question</i>					16

31. Do you think scientists could use a computer game to perform research and actually come up with some new findings (assuming the game physics in the game is implemented correctly)?			
		Response Percent	Response Count
Yes		26.9%	7
Maybe		46.2%	12
No		26.9%	7
<i>answered question</i>			26
<i>skipped question</i>			19

32. If you agreed in the previous question that games might be used as research tool, do you have a specific example from your field of research or any ideas?	
	Response Count
<i>answered question</i>	5
<i>skipped question</i>	36

1. simulating an experiment run by the current laws programed into the computer may yield some results in a more cost effective manner. however this is useless if the laws of physics need to be rewritten.

2. i remember a game with sand flow... it was fun to play with the repose angles.
3. Implement scanning probe microscope and its control. Molecular dynamics.
4. no
5. complex system of interacting (intelligent) agents
6. Not in my field, in information theories, biophysics
7. Orbital mechanics - longterm space flight.
8. This question is barely answerable. Depending on the level of accuracy and computational power of future tabletop computers, scientific discoveries could be implemented more or less precisely. The problem of using a virtual environment as a research tool is, that the outcome of a "virtual" experiment is predictable. Quantum Mechanical features, such as Uncertainty, or aspects of reality yet unknown to us are hardly programmable -today-. To make research in a virtual environment sensible, one would have to create a new "virtual" universe embedded in a real machine. Very tough job. Maybe in a few hundred years we will be capable of doing exactly that, but certainly not nowadays. But on the other hand - If mankind could manage to build a real physics simulator utilizing all -currently- known knowledge, it would be a great step forward to a better scientific education.
9. no, but I could envision having simulations running on thousands of game machines, producing different outcomes, while the results are reported on-line to researchers. Kind of a computational physics analogy to SETI at Home

# Appendix **G**

## Science Art Reviews

### **G.1 Artworks using known game physics design principles**

**Bettina Brendel's *Molecular Visualization*** are a collection of abstract images representing the subatomic world of elementary particles, photons and electrons which explore patterns and symmetries in their interactions. The artist experiments with the role of visual imagery in the field of physics and physical optics through paintings and computer graphics in an attempt to come up with other ways of thinking about the scientific theory and create meaning for the individual. (Brendel 1994, YLEM 2000).

**Mr Snow & Zina Kaye's *Firmament*** is an installation which interfaces with data coming from a radio telescope and continues the artists investigations into the low-tech poetics of space. The system is a listening device with accompanying visualizations related to the data acquisi-

tion. (Kaye & Mr. Snow 2001)

**Peter d'Agostino's *Traces*** is a commemorative documentary short-film which is based on a collage of footage taken during atomic bomb tests and deployment. (Electronic-Arts-Intermix 1997) The physics presented in the piece is limited to the documented theme. However one of the featured characters of the narrative is the physicist Oppenheimer.

**(art)n Laboratory's *Data Portraits*** are renderings of the “real” from a variety of science fields as non-interactive images. ((art)n 2006) Several datasets from physics are incorporate into various productions, either for their visual (i.e. *Shock Wave Physics*, 1996) or metaphoric content (i.e. *Nuclear Physics Detector*, 1998).

**Sonia Sheridan's *Time Concepts*** consists of pioneering works, partially as part of the artists *Generative Systems* program, which explore the issues of perception of time and space. The artist collaborated with scientists, engineers and industry and produced many creative pieces. (Sheridan 2003) Physical realities and the use of scientific methods are not a major theme in the artists works.

**Agnes Hegedus' *Handsight*** is an interactive environment where the “endoscopic eye” or virtual camera projects spherically distorted geometric shapes of the virtual world. The anamorphic perspective produces precipitous spatial relationships which are explored through the interaction. (Hegedus 1992) While the physics is limited to the uncommon optical transformation algorithm.

**Tom Kemp's *Particle Painter*** is a graphics generator that simulates charged particles to algorithmically or interactively create computer graphics. While the artist questions the approach of a computer to create images as *limiting*, he recognizes it as “an interesting tool to create brushes that are physically impossible to build.” (Kemp 2004)

## G.2 Artworks not applicable to game physics design

**James Acord's *Hanover Monument*** project seeks to create a series of monumental sculptures to delineate the historic significance of creation of the world's first full-sized production reactors as an event and place with art. The sculptural site addresses the history of the technical processes and contributes to that area's environmental restoration. (The Columbia Free-Net 2007)

**Gudrun Bielz's *Rays, an additional catastrophe*** is an artwork which creates an environment that uses radioactive materials and a Geiger counter together with video projections, photos and sculptural elements. The theme is a poetic analysis of the material property of "radioactivity", its presence (background radiation) and effect (Hiroshima). (Transmitter-X 2005)

**Jay Lee and Bill Keay's *Suspended Window*** is an interactive video installation that interlaces multiple layers of real and virtual surfaces. Player movement creates an "organic disturbance" focusing attention on the nature and function of spatial boundaries in physical and virtual worlds. (SIGGRAPH 1999)

**Jane Marsching's *About Here and Later*** is an installation composed of a series of digital images and sculptures that explore the relationship of science and myth. (Creative Capital 2006) Geophysical and meteorological scientific data is mixed with science fiction, circus acts, and architectural drawings.

**C. Gerstl, J. Keijser's *Dj Vu of fresh water*** is a CAVE installation where the viewer is exposed to the unconscious behavior one exhibits toward



our environment as predators of nature. (Gerstl & Keijser 2004) Neither theme nor presentation feature much physics.

**Carsten Holler's perceptual experiments** are a collection of works which are intended to raise doubts and elicit questions regarding the systems underlying our existence, including themes such as perception, the human as animal, theories of evolution and generally involving the audience as participant. (Shipper 2006) The audience participation is termed a "science experiment" by the artist as well as the reviewers, but the topics of physics is not a focus of this artwork.

**Eve Andree Laramée's *Apparatus for the Distillation of Vague Intuitions & Instrument to Communicate with Kepler's Ghost*** are object installations which exploit the aesthetic potential of scientific imagery and processes to create a visually complex "apparatus". The installations are narrative pieces that present something fictional as fact or critique the complex historical and cultural dimensions of science. (Honigman 2002, Laramée 2006) The connection of the artist to the physicist Kepler arises more from her personal interest rather than a specific scientific motivation.

**Cornelia Hesse-Honegger's *After Chernobyl*** is a collection of insects and drawings from areas with nuclear fallout and was presented as an exhibition as well as a book. The artists uses scientific methods (protocol and text) during the production of the works. (Hesse-Honegger 2001) The environmental theme of such a collection is unique by tangential to physics.

**Eric Orr's *Electrum Tesla Coil*** is a sculpture that employs high-voltage discharges as the primary component. The sculpture is an evolution of the artists work towards absolute simplicity for his "industrial" art. (ASKlabs 2000, Orr 1998) While the viewer experiences a raw force

of nature, the piece does not focus on an explanation of the physics behind the technology.

# Appendix H

## Source Code

### H.1 Pendulum Game

#### Chromakey Pixel Shader

**Listing H.1:** Chromakey Pixel Shader (HLSL source code)

---

```
sampler TextureSampler : register(s0);
float3 referenceColor;
float threshold;
float4 PixelShader(float2 textureCoordinate: TEXCOORD0) : COLOR
{
    float3 color = tex2D(TextureSampler, textureCoordinate);
    float alpha = 1.0f;
    float distance = distance(color, referenceColor);
    if (distance < threshold)
    {
        alpha = 0.0f;
    }
    float4 newcolor = float4(color.r, color.g, color.b, alpha);
    return newcolor;
}
technique
{
```

```
    pass P0
    {
        PixelShader = compile ps_2_0 PixelShader();
    }
}
```

---

## H.2 Gamebase Database

---

```
#!/bin/bash

# The database user
USER=root

echo -n "Enter database password for user $USER > "
read PASS
echo pass=$PASS

echo Copying files ...
cp -v ../Data/*.mdb .

echo Creating databases ...
echo "DROP DATABASE IF EXISTS gamebase;" >>database.sql
echo "CREATE DATABASE gamebase;" >>database.sql
echo "USE gamebase;" >>database.sql
echo "CREATE TABLE games ( " >>database.sql
echo "  source char(32) NOT NULL, " >>database.sql
echo "  gameid char(64) NOT NULL, " >>database.sql
echo "  platform char(32) NOT NULL, " >>database.sql
echo "  name char(64) NOT NULL, " >>database.sql
echo "  category char(64) NOT NULL, " >>database.sql
echo "  subcategory char(64) NOT NULL, " >>database.sql
echo "  reldate date" >>database.sql
echo " ) ENGINE=MyISAM DEFAULT CHARSET=latin1;" >>database.sql

for FILE in *.mdb; do
    echo file=$FILE
    BASE=${FILE%%.mdb}
    echo base=$BASE
    DB=gamebase_$BASE
    echo "DROP DATABASE IF EXISTS $DB;" >>database.sql
    echo "CREATE DATABASE $DB;" >>database.sql
done
mysql -f -u $USER -p$PASS <database.sql
```

```

echo Extracting schema from MDB ...
for FILE in *.mdb; do
  echo file=$FILE
  BASE=${FILE%%.mdb}
  echo base=$BASE
  DB=gamebase_$BASE
  echo "USE $DB;" >$BASE-schema.txt
  mdb-schema $FILE postgres | grep -v "-" | grep -v "relation" | sed "s/\\tMemo\\t/
  Hyperlink/Text (255)/" | sed "s/Text/Char/" | sed "s/Byte/Integer/" | sed "s
  /Long //" | sed "s/Unknown 0x00/Char (255)/" | sed "s/Postgres_Unknown 0x0c/
  Char (255)/" | sed "s/(255) (4)/(255)/" | sed "s/(256)/(255)/" | sed "s
  /(510)/(255)/" | sed "s/DROP TABLE/DROP TABLE IF EXISTS/" >>$BASE-schema.txt
  mysql -f -u $USER -p$PASS $DB <$BASE-schema.txt
done

echo Extracting and importing MDB ...
for FILE in *.mdb; do
  echo file=$FILE
  BASE=${FILE%%.mdb}
  echo base=$BASE
  DB=gamebase_$BASE
  for TABLE in `mdb-tables $FILE`; do
    echo table=$TABLE
    mdb-export $FILE $TABLE | tail +2 >$BASE-$TABLE.csv
    echo "LOAD DATA LOCAL INFILE '$BASE-$TABLE.csv' INTO TABLE $TABLE FIELDS
    TERMINATED BY ',' ENCLOSED BY '\"' ESCAPED BY '\\\\' LINES TERMINATED BY
    '\\n';" | mysql -u $USER -p$PASS $DB
  done
  cat add_index.sql | mysql -u $USER -p$PASS $DB
  echo "SELECT 'gamebase' AS source, '' AS gameid, '$BASE' AS platform, name, PGenres.
  ParentGenre AS category, Genres.Genre AS subcategory, CONCAT(Years.Year
  , '-01-01') AS reldate FROM Games, Years, Genres, PGenres where Games.YE_Id=
  Years.YE_Id AND Genres.GE_id=Games.Ge_ID AND Genres.PG_Id=PGenres.PG_ID" |
  mysql -u $USER -p$PASS $DB | tail +2 >$BASE-summary.txt
  echo "LOAD DATA LOCAL INFILE '$BASE-summary.txt' INTO TABLE games FIELDS
  TERMINATED BY '\\t' ENCLOSED BY '\"' ESCAPED BY '\\\\' LINES TERMINATED BY
  '\\n'" | mysql -u $USER -p$PASS gamebase
done

echo Cleaning up ...
rm database.sql
rm *.csv
rm *.mdb
rm *.txt
rm *~

```

```

echo Stats ...
echo "select * from games limit 1;" | mysql -t -u $USER -p$PASS gamebase
echo "select count(1) as GamesInDatabase from games;" | mysql -t -u $USER -
    p$PASS gamebase

```

---

## Listing H.2: Gamebase Database Processor

---

```

#!/bin/bash

# The database user
USER=root

echo -n "Enter database password for user $USER > "
read PASS
echo pass=$PASS

echo Category mapping ...
if [ ! -f physicsmapping.txt ]; then
    sudo rm /tmp/physicsmapping.txt
    echo "SELECT DISTINCT category,subcategory,' ' AS physics,' ' AS space " >map.sql
    echo " FROM games " >>map.sql
    echo " ORDER BY category,subcategory" >>map.sql
    echo " INTO OUTFILE '/tmp/physicsmapping.txt' " >>map.sql
    echo " FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '\"' " >>map.sql
    echo " LINES TERMINATED BY \"\\n\" " >>map.sql
    mysql --column-names -u $USER -p$PASS gamebase <map.sql
    cp /tmp/physicsmapping.txt physicsmapping.csv
    echo Edit physicsmapping.csv and rename to physicsmapping.txt afterwards.
    wc -l physicsmapping.csv
else
    echo Creating mapping database ...
    echo "USE gamebase;" >database.sql
    echo "DROP TABLE IF EXISTS mapping;" >>database.sql
    echo "CREATE TABLE mapping ( " >>database.sql
    echo "  category char(64) NOT NULL, " >>database.sql
    echo "  subcategory char(64) NOT NULL, " >>database.sql
    echo "  physics int" >>database.sql
    echo " ) ENGINE=MyISAM DEFAULT CHARSET=latin1;" >>database.sql
    mysql -f -u $USER -p$PASS <database.sql

    echo Load mapping file ...
    echo "LOAD DATA LOCAL INFILE 'physicsmapping.txt' " >import.sql
    echo " INTO TABLE mapping " >>import.sql
    echo " FIELDS TERMINATED BY ',' " >>import.sql
    echo " ENCLOSED BY '\"' ESCAPED BY '\\\\' " >>import.sql

```

```

echo " LINES TERMINATED BY '\\n'" >>import.sql
echo " IGNORE 1 LINES " >>import.sql
mysql -u $USER -p$PASS gamebase <import.sql

echo Adding index to tables ...
mysql -f -u $USER -p$PASS gamebase <add_index_games.sql
mysql -f -u $USER -p$PASS gamebase <add_index_mapping.sql
fi

echo Clean up ...
rm -f database.sql import.sql map.sql *~ inc

```

---

### Listing H.3: Gamebase Database Mapper

---

```

ALTER TABLE Games ADD INDEX ( YE_Id );
ALTER TABLE Games ADD INDEX ( GE_Id );
ALTER TABLE Years ADD INDEX ( YE_Id );
ALTER TABLE Genres ADD INDEX ( GE_Id );
ALTER TABLE Genres ADD INDEX ( PG_Id );
ALTER TABLE PGenres ADD INDEX ( PG_Id );

```

---

### Listing H.4: Gamebase Add Index

---

```

ALTER TABLE mapping ADD INDEX ( category );
ALTER TABLE mapping ADD INDEX ( subcategory );

```

---

### Listing H.5: Gamebase Add Index (Mapping)

---

```

ALTER TABLE games ADD INDEX ( name );
ALTER TABLE games ADD INDEX ( category );
ALTER TABLE games ADD INDEX ( subcategory );

```

---

### Listing H.6: Add Index (Games)

---

```

Category,Subcategory,Physics
Activity,"Draw, 2D",0
Adventure,3D Construction Kit,1
Adventure,Adventure Creator,0
Adventure,AdventureWriter/Quill,0
Adventure,AGI Interpreter,0
Adventure,Animated Graphics & Text,0
Adventure,Arcade 2D,1
Adventure,Arcade 3D,2
Adventure,Arcade Isometric,1
Adventure,Click and Type,0

```

Adventure,"Combat, 2D",1  
Adventure,"Combat, Side Scroller",1  
Adventure,Comic,1  
Adventure,Construction Kit,0  
Adventure,GAC,0  
Adventure,Graphic(Charset)/Text,0  
Adventure,Graphic(Hi-Res)/Text,0  
Adventure,Graphics & Text,0  
Adventure,Isometric 3D,2  
Adventure,Joystick only,1  
Adventure,Miscellaneous,0  
Adventure,Move & Type,0  
Adventure,Move and Type,0  
Adventure,"Multi-Game, Multi View",1  
Adventure,"Platform, 3D",2  
Adventure,"Platform, Isometric Scroller",1  
Adventure,"Platform, Side Scroller",1  
Adventure,"Platform, Top View",1  
Adventure,"Platform, Vertical Scroller",1  
Adventure,Point & Click,1  
Adventure,Point and Click,1  
Adventure,Reality Sim,3  
Adventure,RPG 2D,1  
Adventure,RPG 3D,2  
Adventure,RPG Isometric,1  
Adventure,RPG Text,0  
Adventure,"RPG, 3D",2  
Adventure,"RPG, Isometric Scroller",1  
Adventure,"RPG, Isometric View",1  
Adventure,"RPG, Menu based",0  
Adventure,"RPG, Multi View",1  
Adventure,"RPG, Side Scroller",1  
Adventure,"RPG, Side-Scroll",1  
Adventure,"RPG, Text Driven",0  
Adventure,"RPG, Top View",1  
Adventure,"RPG, Versus",1  
Adventure,Selectable Answers,0  
Adventure,STAC (Illustrated),0  
Adventure,STAC (Text only),0  
Adventure,TaleSpin,0  
Adventure,Text only,0  
Adventure,"Versus RPG, 2D",1  
Adventure,"Versus RPG, 3D",2  
Adventure,"Versus RPG, Multiview",1  
Adventure,"Versus RPG, Top View",1  
Adventure,[uncategorized],0



Arcade,Amidar,1  
Arcade,Avoid it,1  
Arcade,Bat & Ball,2  
Arcade,Beat'em Up,1  
Arcade,Beat'em Up - Progressive,1  
Arcade,Bomberman,1  
Arcade,Boulder Dash,2  
Arcade,Boulderdash,2  
Arcade,Breakout/Pong,2  
Arcade,Catch it,1  
Arcade,Collect'em Up,1  
Arcade,Collect'em Up (Racing),2  
Arcade,Construction Kit,1  
Arcade,Frogger,1  
Arcade,Joust,2  
Arcade,Labyrinth/Maze,0  
Arcade,Lander,2  
Arcade,Logical Game,0  
Arcade,Miscellaneous,0  
Arcade,Multigenre,0  
Arcade,Pac Man,0  
Arcade,Pengo,0  
Arcade,Pinball,2  
Arcade,"Pinball, 3D",3  
Arcade,Platformer (3D),2  
Arcade,Platformer (Multi Screen),1  
Arcade,Platformer (Scrolling Screen),1  
Arcade,Platformer (Single Screen),1  
Arcade,SEUCK,0  
Arcade,"Tennis, 2D",2  
Arcade,Tetris,1  
Arcade,Tron,1  
Arcade,"Vintage Arcade, 2D",1  
Arcade,"Vintage Arcade, Isometric Scroller",1  
Arcade,"Vintage, Top View",1  
Arcade,Worm/Snake,0  
Arcade,[uncategorized],0  
Board Game,Backgammon,0  
Board Game,Chess,0  
Board Game,Draughts/Checkers,0  
Board Game,Kalaha,0  
Board Game,Miscellaneous,0  
Board Game,Monopoly,0  
Board Game,Othello,0  
Board Games,Backgammon,0  
Board Games,Chess,0

Board Games,Draughts,0  
Board Games,Jigsaws,0  
Board Games,Mahjongg (Solitaire),0  
Board Games,Mastermind,0  
Board Games,Miscellaneous,0  
Board Games,Monopoly,0  
Board Games,Othello,0  
Brain,4 Wins,0  
Brain,Arcade,1  
Brain,"Board Game, 3D",0  
Brain,Crossword,0  
Brain,Hangman,0  
Brain,Logical Game,0  
Brain,Mastermind,0  
Brain,Memory,0  
Brain,Minesweeper,0  
Brain,Miscellaneous,0  
Brain,Puzzle,0  
Brain,Solitaire,0  
Brain,Tic Tac Toe,0  
Brain,Tile Removal,0  
Brain,Towers of Hanoi,0  
Brain,[uncategorized],0  
Cards,Baccarat,0  
Cards,Blackjack,0  
Cards,Bridge,0  
Cards,Miscellaneous,0  
Cards,Poker,0  
Cards,Skat,0  
Cards,Solitaire,0  
Cards,[uncategorized],0  
Construction,Game Creation,0  
Construction,Level Designer,0  
Construction,Level Editor,0  
Construction,Miscellaneous,0  
Driving,"Combat, 3D",2  
Driving,"Combat, Isometric View",1  
Driving,"Combat, Side Scroller",1  
Driving,"Motorsport, 3D",3  
Driving,"Motorsport, Isometric Scroller",2  
Driving,"Motorsport, Top View",2  
Driving,"Platform, Side Scroller",1  
Driving,"Space Theme, 3D",3  
Educational,Geography,0  
Educational,Maths,0  
Educational,Miscellaneous,0

Educational,Quiz,0  
Educational,Spelling,0  
Educational,Typing,0  
Educational,Vocabulary/Language,0  
Educational,[uncategorized],0  
Edutainment,"Platform, Side Scroller",1  
Fighting,"Boxing, 2D",1  
Fighting,"Combat, 2D",1  
Fighting,"Combat, 3D",2  
Fighting,"Combat, Isometric View",1  
Fighting,"Combat, Multi-View",1  
Fighting,"Combat, Side Scroller",1  
Fighting,"Combat, Top View",1  
Fighting,"Flying Combat, Side Scroller",1  
Fighting,"KO, 2D",1  
Fighting,"Multi-Game, Multi View",1  
Fighting,"Pool, 3D",3  
Fighting,"Versus RPG, 2D",1  
Fighting,"Wrestling, 2D",1  
Fighting,"Wrestling, Stadium View",1  
Flying,"Combat, 3D",2  
Flying,"Flight Sim, 3D",3  
Flying,"Flying Combat, 3D",3  
Flying,"Shooter, Vertical Scroller",2  
Flying,"Skill, 3D",3  
Flying,"Skill, Multi View",2  
Flying,"Space Theme, Side Scroller",1  
Gambling,Backgammon,0  
Gambling,Battleships,0  
Gambling,Board Game,0  
Gambling,Cards,0  
Gambling,Casino,0  
Gambling,Chess,0  
Gambling,Dice,0  
Gambling,Draughts,0  
Gambling,Horseracing,0  
Gambling,Miscellaneous,0  
Gambling,Othello,0  
Gambling,Quiz,0  
Gambling,Racing,0  
Gambling,Slot Machine,0  
Gambling,Solitaire,0  
Gambling,Solitaire,0  
Gambling,Yahtzee,0  
Gambling,[uncategorized],0  
Girl,"Multi-Game, Multi View",1

Kids,"Draw, 2D",0  
Kids,"Reality Sim, Side Scroller",2  
Miscellaneous,Adult,0  
Miscellaneous,Demo,0  
Miscellaneous,Undefinable,0  
Miscellaneous,Weird!,0  
Miscellaneous,[uncategorized],0  
Parlour Game,"Gambling, 2D",0  
Parlour Game,"Gambling, Multi View",0  
Platform,"Combat, Side Scroller",1  
Puzzle,"Board Game, 2D",0  
Puzzle,"Breakout, 2D",2  
Puzzle,Game Show,0  
Puzzle,"Hunt, 2D",1  
Puzzle,"Matching, 2D",0  
Puzzle,"Matching, Isometric View",0  
Puzzle,"Matching, Top View",0  
Puzzle,"Maze, 2D",0  
Puzzle,"Maze, Stadium View",0  
Puzzle,"Multi-Game, Multi View",0  
Puzzle,"Pachinko, 2D",2  
Puzzle,"Platform, 2D",1  
Puzzle,"Platform, Side Scroller",1  
Puzzle,Quiz,0  
Puzzle,"Roll-a-ball, 3D",2  
Puzzle,Tetris,1  
Racing,Cars,3  
Racing,Formula One,3  
Racing,Isometric,2  
Racing,Miscellaneous,1  
Racing,Motorcycle,2  
Racing,"Motorsport, 3D",3  
Racing,Overhead,1  
Racing,"Platform, 3D",2  
Racing,Stay on Track,2  
Racing,[uncategorized],2  
Shoot'em Up,"1st Person, 3D",2  
Shoot'em Up,3D,2  
Shoot'em Up,Asteroids,2  
Shoot'em Up,Bomber,1  
Shoot'em Up,Centipede,1  
Shoot'em Up,Chase View,1  
Shoot'em Up,"Combat, 3D",2  
Shoot'em Up,"Combat, Isometric View",1  
Shoot'em Up,"Combat, Side Scroller",1  
Shoot'em Up,"Combat, Top View",1

Shoot'em Up,"Combat, Vertical Scroller",1  
Shoot'em Up,"Creator, Multi View",1  
Shoot'em Up,Crosshair,1  
Shoot'em Up,D-Scrolling,1  
Shoot'em Up,Defender,1  
Shoot'em Up,Duel,2  
Shoot'em Up,"Flying Combat, 3D",3  
Shoot'em Up,"Flying Combat, Isometric Scroller",1  
Shoot'em Up,"Flying Combat, Multi View",1  
Shoot'em Up,"Flying Combat, Side Scroller",1  
Shoot'em Up,"Flying Combat, Vertical Scroller",1  
Shoot'em Up,FPS,2  
Shoot'em Up,Gauntlet,1  
Shoot'em Up,H-Scrolling,1  
Shoot'em Up,Horizontal,1  
Shoot'em Up,Isometric,1  
Shoot'em Up,"Light Gun, 2D",1  
Shoot'em Up,"Light Gun, 3D",2  
Shoot'em Up,"Lightgun, Multi View",1  
Shoot'em Up,"Lightgun, Side scroller",1  
Shoot'em Up,Miscellaneous,0  
Shoot'em Up,Multi-Directional,1  
Shoot'em Up,Multi-Scrolling,1  
Shoot'em Up,"Platform, Side Scroller",1  
Shoot'em Up,"Platform, Top View",1  
Shoot'em Up,Platformer,1  
Shoot'em Up,Racing,2  
Shoot'em Up,Scramble,2  
Shoot'em Up,SEUCK,1  
Shoot'em Up,"Shooter, 3D",2  
Shoot'em Up,"Shooter, Side Scroller",1  
Shoot'em Up,"Shooter, Vertical Scroller",1  
Shoot'em Up,Space Invaders,1  
Shoot'em Up,"Space Theme, 2D",1  
Shoot'em Up,"Space Theme, 3D",2  
Shoot'em Up,"Space Theme, Side Scroller",1  
Shoot'em Up,"Space Theme, Vertical Scroller",1  
Shoot'em Up,"Submarine, 2D",2  
Shoot'em Up,"Submarine, 3D",2  
Shoot'em Up,"Superscope, 3D",2  
Shoot'em Up,Uridium,1  
Shoot'em Up,V-Scrolling,1  
Shoot'em Up,Vertical,1  
Shoot'em Up,"Vintage Arcade, 2D",1  
Shoot'em Up,"Western, 3D",2  
Shoot'em Up,[uncategorized],0

Simulation,Flight (Civil),3  
Simulation,Flight (Military),3  
Simulation,Marine,2  
Simulation,Miscellaneous,1  
Simulation,Space,3  
Simulation,Spy,1  
Simulation,Tank,3  
Simulation,Train,1  
Simulation,[uncategorized],1  
Skill,"Platform, Isometric View",1  
Sport,"Boxing, 2D",1  
Sport,"Ice Hockey, Stadium View",0  
Sports,American Football,2  
Sports,Athletics,2  
Sports,"Athletics, 2D",2  
Sports,"Athletics, 3D",2  
Sports,Baseball,2  
Sports,"Baseball, 3D",2  
Sports,Basketball,2  
Sports,"Basketball, 3D",2  
Sports,"Basketball, Isometric View",2  
Sports,"Basketball, Stadium View",2  
Sports,"Basketball, Top View",2  
Sports,Bowling,2  
Sports,"Bowling, 3D",3  
Sports,Boxing,1  
Sports,Cricket,2  
Sports,"Cricket, 3D",2  
Sports,Cycling,1  
Sports,Darts,3  
Sports,Fighting,1  
Sports,Fishing,1  
Sports,"Fishing, 3D",1  
Sports,"Football, 3D",2  
Sports,"Football, Isometric View",1  
Sports,"Football, Stadium View",1  
Sports,Football/Soccer,2  
Sports,Football/Soccer (Arcade),1  
Sports,Football/Soccer (Manager),0  
Sports,Golf,2  
Sports,"Golf, 3D",3  
Sports,"Hockey, 3D",2  
Sports,"Horse Racing, Stadium View",0  
Sports,Ice Hockey,1  
Sports,"Ice Hockey, 3D",2  
Sports,"Ice Hockey, Top View",1

Sports,Icehockey,1  
Sports,Miscellaneous,1  
Sports,"Mountain Bikes, 3D",2  
Sports,Multi-Event,1  
Sports,"Multi-Game, Multi View",1  
Sports,"Pool, Top View",3  
Sports,Riding,0  
Sports,"Rollerball, 3D",1  
Sports,Rugby,2  
Sports,"Rugby, 3D",2  
Sports,Shooting,2  
Sports,Skating,2  
Sports,Skiing,2  
Sports,"Skiing, 3D",2  
Sports,Snooker/Pool,3  
Sports,"Soccer, 3D",2  
Sports,"Soccer, Isometric",1  
Sports,"Soccer, Stadium View",1  
Sports,"Soccer, Top View",1  
Sports,Squash,2  
Sports,Table Tennis,2  
Sports,Tennis,2  
Sports,"Tennis, 3D",2  
Sports,"Tennis, Stadium View",1  
Sports,"Versus RPG, Split View",1  
Sports,Volleyball,2  
Sports,"Volleyball, 3D",2  
Sports,Watersports,2  
Sports,Wrestling,1  
Sports,[uncategorized],1  
Strategy,"Board Game, 2D",0  
Strategy,"Board Game, 3D",0  
Strategy,"Combat, 3D",1  
Strategy,"Combat, Multi View",1  
Strategy,"Combat, Top View",1  
Strategy,Company,0  
Strategy,Detective,0  
Strategy,"Football Management, Multi-View",0  
Strategy,"God, Isometric View",2  
Strategy,Life,1  
Strategy,Miscellaneous,0  
Strategy,"Platform, Side Scroller",1  
Strategy,Politics,0  
Strategy,"Reality Sim, Isometric View",1  
Strategy,"Reality Sim, Multi View",1  
Strategy,"Reality Sim, Top View",1

```
Strategy,"RPG, Menu based",0
Strategy,"RPG, Top View",1
Strategy,"Space Theme, Vertical Scroller",1
Strategy,Trading,0
Strategy,"Versus RPG/War, 2D",1
Strategy,Wallstreet,0
Strategy,War,1
Strategy,"War Sim, Multi View",1
Strategy,"War Sim, Top View",1
Strategy,[uncategorized],0
Uncategorized,Uncategorized,0
```

---

**Listing H.7:** Gamebase Physics Index Mapping Table

## H.3 Gamespot Database

---

```
#!/bin/bash

echo -n "Enter 'y' to continue to scrape, ctrl-c to exit > "
read DUMMY

echo Scraping gamespot pages ...
perl analyze.pl

echo Moving .csv files to Data ...
mv -f *.csv ../Data
```

---

**Listing H.8:** Gamespot Scraper Driver

---

```
#!/usr/bin/perl

#
# Gamespot game info scraper
#

use IO::Handle qw(flush);
use LWP::Simple qw(get);
use Date::Parse;
use Date::Format;

# Load all platform names and codes ...
$filename="categorycodes.txt";
printf STDERR "Loading $filename ...\\n";
```



```

%platforms=();
$count=0;
open (INPUT, "< $filename");
while (defined ($line = <INPUT>)) {
    chomp $line;
    if ($line =~ m/(.*)/(.*)/) {
        $platforms{$1}=$2;
    }
    $count++;
}
close(INPUT);

# Log summary
printf STDERR "Read $count platforms ...\n";

# Define source URLs for the various platforms needs &platform=### and &page=###
$sourceurl = "http://www.gamespot.com/games.html?type=games&mode=all&sort=views&
    dlx_type=all";

# Loop over all platforms
$gamecount=0;
foreach $platform (sort keys %platforms) {
    $gamecount_platform=0;
    # Open .csv file for this platform
    $filename = $platforms{$platform};
    $filename =~ s/\W//g;
    $filename .= ".csv";
    print "Writing $filename ...\n";
    open (OUTPUT, "> $filename");
    # Print csv header
    print OUTPUT "\"source\", \"gameid\", \"platform\", \"name\", \"category\", \"
        subcategory\", \"reldate\"\n";
    # Get baseurl
    printf STDERR "Processing " . $platforms{$platform} . " ($platform) ...";
    $baseurl=$sourceurl . "&platform=" . $platform;
    # Loop over possible pages
    PAGE: for ($i=0; $i<1000; $i++) {
        $url=$baseurl . "&page=" . $i;
        printf STDERR "Loading URL for page $i ...";
        # Get data from website
        $content = get($url);
        printf STDERR " read " . length($content) . " bytes ...\n";
        # Check for error to terminate page loop
        $errormessage="Sorry. There are no results for the filter you have selected.";
        last PAGE if ($content =~ m/$errormessage/);
        # Scrape/Parse HTML

```

```

$content =~ m/\<div class="body"\>.*?<table summary="(.*?)\<div class="
    page_nav"\>/gms;
$table=$1;
if ($table) {
    $table =~ s/ +/ /g;
    $table =~ m/\<tbody\>(.*?)\<\tbody\>/gms;
    $tbody=$1;
    # Grab rows
    while ($tbody =~ m/\<tr.*?\>(.*?)\<\tr\>/gms) {
        $row=$1;
        @cells=();
        $id="";
        $cellcount=0;
        # Grab other cells
        while ($row =~ m/\<t[dh].*?\>(.*?)\<\t[dh]\>/gms) {
            $cell=$1;
            # Remove links
            if ($cell =~ m/\<a(.*?)\>(.*?)\<\a\>/gms) {
                $anchor=$1;
                $cell=$2;
                if ($anchor =~ m/.*/(.*?)\/index.html.*/) {
                    $id = $1;
                }
            }
            # Remove spans
            if ($cell =~ m/\<span.*?\>(.*?)\<\span\>/gms) {
                $cell=$1;
            }
            #remove quotes and store
            $cell =~ s/"//g;
            $cells[$cellcount]=$cell;
            $cellcount++;
        } # cells from row
        # Print CSV
        if ($cells[0]) {
            # Convert date
            $date=$cells[4];
            $datetime = str2time($date);
            $sqldate=time2str("%Y-%m-%d", $datetime);
            # Output row
            print OUTPUT "\"gamespot\", \"$id\", \"$platforms{$platform}\"\", \"$cells
                [0]\", \"$cells[1]\", \"\", \"$sqldate\"\\n";
            flush(OUTPUT);
            $gamecount++;
            $gamecount_platform++;
        }
    }
}

```

```

    } # rows from body
    # Progress logging
    printf STDERR "Games so far $gamecount ... \n";
    flush(STDERR);
  }
}
close OUTPUT;
printf STDERR "Game for $platforms{$platform}: $gamecount_platform\n";
}
printf STDERR "Game in total: $gamecount\n";

```

---

### Listing H.9: Gamespot Scraper

---

```

15,3DO
1058,Acorn Archimedes
1042,Adventurevision
24,Amiga
1059,Amiga CD32
1035,Amstrad CPC
1061,Android
1020,APF-*1000/IM
25,Apple II
26,Arcade Games
27,Arcadia 2001
1021,Astrocade
28,Atari 2600
29,Atari 5200
30,Atari 7800
31,Atari 8-bit
1034,Atari ST
1052,Bandai Pippin
1053,BBC Micro
32,BBS Door
1063,BlackBerry
1060,Casio Loopy
1056,Cassette Vision
33,CD-I
34,Channel F
35,Colecovision
36,Commodore 64
1038,Commodore PET
1044,CPS Changer
1048,CreatiVision
1,Dreamcast
1023,DVD Player
1013,e-Reader

```

1040,EACA Colour Genie 2000  
1008,Famicom Disk System  
1064,Flash  
1041,FM Towns  
1055,FM-7  
2,Game Boy  
9,Game Boy Color  
38,Game.com  
16,GameGear  
10,Genesis  
1030,Gizmondo  
37,GP32  
39,Intellivision  
1047,Interton VC4000  
17,Jaguar  
1004,Jaguar CD  
1046,LaserActive  
40,Lynx  
42,Macintosh  
1039,Mattel Aquarius  
43,Microvision  
41,MSX  
1006,N-Gage  
1065,NEC PC88  
1033,NEC PC98  
1007,Neo-Geo CD  
18,NeoGeo  
3,NeoGeo Pocket Color  
19,NES  
4,Nintendo 64  
1012,Nintendo 64DD  
14,Nuon  
1011,Odyssey  
44,Odyssey^2  
1010,Online/Browser  
1037,Oric 1/Atmos  
1018,OS/2  
1015,Palm OS Classic  
1062,Palm webOS  
45,PC-FX  
1009,Pinball  
46,Playdia  
6,PlayStation  
1045,RCA Studio II  
1019,Redemption  
8,Saturn

```
1003,Sega 32X
1002,Sega CD
20,Sega Master System
1017,SG-1000
1066,Sharp X1
1032,Sharp X68000
1050,Sinclair ZX81/Spectrum
21,SNES
1054,Sord M5
1057,Super Cassette Vision
1043,SuperVision
1036,Tandy Color Computer
1051,TI-99/4A
1005,Turbo CD
22,TurboGrafx-16
47,Unix/Linux
49,Vectrex
48,VIC-20
50,VirtualBoy
1014,Windows Mobile
1022,WonderSwan
23,WonderSwan Color
1027,Zodiac
13,Xbox
11,GameCube
12,Game Boy Advance
1049,iPhone
5,PC
7,PlayStation 2
1028,PlayStation 3
10281,PlayStation Network
1024,PSP
1029,Xbox 360
10291,Xbox 360 Live Arcade
1026,DS
1031,Wii
10311,Wii Virtual Console + WiiWare
```

---

### Listing H.10: Gamespot Scraper Category Codes

---

```
#!/bin/bash

# The database user
USER=root

echo -n "Enter database password for user $USER > "
```

```

read PASS
echo pass=$PASS

echo Copy csv files ...
cp ../Data/*.csv .

echo Creating gamespot database ...
echo "DROP DATABASE IF EXISTS gamespot;" >database.sql
echo "CREATE DATABASE gamespot;" >>database.sql
echo "USE gamespot;" >>database.sql
echo "CREATE TABLE games ( " >>database.sql
echo "  source char(32) NOT NULL, " >>database.sql
echo "  gameid char(64) NOT NULL, " >>database.sql
echo "  platform char(32) NOT NULL, " >>database.sql
echo "  name char(64) NOT NULL, " >>database.sql
echo "  category char(64) NOT NULL, " >>database.sql
echo "  subcategory char(64) NOT NULL, " >>database.sql
echo "  reldate date" >>database.sql
echo " ) ENGINE=MyISAM DEFAULT CHARSET=latin1;" >>database.sql
mysql -f -u $USER -p$PASS <database.sql

for FILE in *.csv; do
  echo file=$FILE
  echo "LOAD DATA LOCAL INFILE '$FILE' " >import.sql
  echo " INTO TABLE games " >>import.sql
  echo " FIELDS TERMINATED BY \",\" " >>import.sql
  echo " ENCLOSED BY \"\\\" \" >>import.sql
  echo " ESCAPED BY \"\\\\\" \" >>import.sql
  echo " LINES TERMINATED BY \"\\n\" " >>import.sql
  echo " IGNORE 1 LINES " >>import.sql
  mysql -u $USER -p$PASS gamespot <import.sql
done

echo Cleaning up ...
rm database.sql
rm import.sql
rm *.csv

echo Stats ...
echo "select * from games limit 1;" | mysql -t -u $USER -p$PASS gamespot
echo "select count(1) as GamesInDatabase from games;" | mysql -t -u $USER -
  p$PASS gamespot

```

---

### Listing H.11: Gamespot Database Processor

---

```
!/bin/bash
```

```

# The database user
USER=root

echo -n "Enter database password for user $USER > "
read PASS
echo pass=$PASS

echo Category mapping ...
if [ ! -f physicsmapping.txt ]; then
sudo rm /tmp/physicsmapping.txt
echo "SELECT DISTINCT category,subcategory,' ' AS physics,' ' AS space " >map.sql
echo " FROM games " >>map.sql
echo " ORDER BY category,subcategory" >>map.sql
echo " INTO OUTFILE '/tmp/physicsmapping.txt' " >>map.sql
echo " FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '\"' " >>map.sql
echo " LINES TERMINATED BY '\\n\" " >>map.sql
mysql --column-names -u $USER -p$PASS gamespot <map.sql
cp /tmp/physicsmapping.txt physicsmapping.csv
echo Edit physicsmapping.csv and rename to physicsmapping.txt afterwards.
wc -l physicsmapping.csv
else
echo Creating mapping database ...
echo "USE gamespot;" >database.sql
echo "DROP TABLE IF EXISTS mapping;" >>database.sql
echo "CREATE TABLE mapping ( " >>database.sql
echo " category char(64) NOT NULL, " >>database.sql
echo " subcategory char(64) NOT NULL, " >>database.sql
echo " physics int" >>database.sql
echo " ) ENGINE=MyISAM DEFAULT CHARSET=latin1;" >>database.sql
mysql -f -u $USER -p$PASS <database.sql

echo Load mapping file ...
echo "LOAD DATA LOCAL INFILE 'physicsmapping.txt' " >import.sql
echo " INTO TABLE mapping " >>import.sql
echo " FIELDS TERMINATED BY ',' " >>import.sql
echo " ENCLOSED BY '\"' ESCAPED BY '\\\" " >>import.sql
echo " LINES TERMINATED BY '\\n' " >>import.sql
echo " IGNORE 1 LINES " >>import.sql
mysql -u $USER -p$PASS gamespot <import.sql

echo Adding index to tables ...
mysql -f -u $USER -p$PASS gamespot <add_index_games.sql
mysql -f -u $USER -p$PASS gamespot <add_index_mapping.sql
fi

```

```
echo Clean up ...  
rm -f database.sql import.sql map.sql *~ inc
```

---

## Listing H.12: Gamespot Database Mapper

---

```
category,subcategory,physics  
2D Fighting,,1  
2D Platformer,,1  
3D Fighting,,2  
3D Platformer,,2  
Action,,1  
Action Puzzle,,1  
Action Role-Playing,,1  
Adventure,,0  
Alt. Sports,,2  
Baseball,,2  
Baseball Management,,0  
Baseball Sim,,3  
Basketball,,2  
Basketball Management,,0  
Basketball Sim,,3  
Beat-'Em-Up,,1  
Biking,,2  
Billiards,,3  
Board,,0  
Bowling,,3  
Boxing,,2  
Business Strategy,,0  
Car Combat,,2  
Card Battle,,0  
Casino,,0  
Compilation,,1  
Computer Role-Playing,,0  
Console Hardware,,0  
Cricket,,2  
Demo Disc,,0  
Demolition Derby,,2  
Drag Racing,,2  
Driving,,3  
Edutainment,,1  
Fantasy Action Adventure,,2  
Fantasy First-Person Shooter,,2  
Fantasy Online Role-Playing,,2  
Fantasy Shooter,,2  
Fantasy Turn-Based Strategy,,1  
Fishing,,1
```



Flight Action,,2  
Flight Simulation,,3  
Football,,2  
Football Management,,0  
Football Sim,,3  
Formula One Racing,,3  
Futuristic Jet Sim,,3  
Futuristic Racing,,2  
Futuristic Sports,,2  
Futuristic Sub Sim,,2  
Golf,,2  
Golf Sim,,3  
GT / Street Racing,,2  
Hardware,,0  
Helicopter Sim,,3  
Historic Action Adventure,,1  
Historic City-Building,,0  
Historic First-Person Shooter,,2  
Historic Online Role-Playing,,2  
Historic Real-Time Strategy,,1  
Historic Shooter,,2  
Historic Tactical Shooter,,2  
Historic Turn-Based Strategy,,1  
Horse Racing,,1  
Horror Action Adventure,,2  
Hunting,,2  
Ice Hockey,,2  
Ice Hockey Management,,0  
Ice Hockey Sim,,3  
Interactive Movie,,0  
Jet Sim,,3  
Kart Racing,,2  
Light Gun Shooter,,1  
Logic Puzzle,,0  
Matching Puzzle,,0  
Mech Sim,,2  
Miscellaneous,,0  
Mission-based Driving,,2  
Modern Action Adventure,,2  
Modern City-Building,,0  
Modern First-Person Shooter,,2  
Modern Online Role-Playing,,2  
Modern Shooter,,2  
Modern Tactical Shooter,,2  
Modern Turn-Based Strategy,,1  
Motocross Racing,,2

Motorcycle Racing,,2  
Music Maker,,0  
Naval Sim,,2  
Olympic Sports,,1  
On-foot Racing,,1  
Party,,0  
Pinball,,3  
Puzzle,,0  
Racing,,2  
Rail Shooting,,1  
Rally / Offroad Racing,,2  
Real-Time Strategy,,1  
Real-Time Wargame,,1  
Rhythm / Dancing,,0  
Rhythm / Music,,0  
Role-Playing,,0  
Rugby,,2  
Sci-Fi Action Adventure,,1  
Sci-Fi Adventure,,1  
Sci-Fi First-Person Shooter,,2  
Sci-Fi Online Role-Playing,,2  
Sci-Fi Real-Time Strategy,,1  
Sci-Fi Shooter,,2  
Sci-Fi Turn-Based Strategy,,1  
Shoot-'Em-Up,,1  
Simulation,,3  
Skateboarding,,2  
Skating,,2  
Skiing,,2  
Snow / Water Racing,,2  
Snowboarding,,2  
Soccer,,2  
Soccer Management,,0  
Soccer Sim,,3  
Space Combat Sim,,3  
Sports,,2  
Stacking Puzzle,,0  
Stock Car Racing,,0  
Strategy,,0  
Submarine Sim,,3  
Surfing,,2  
Tank Sim,,3  
Tennis,,2  
Text Adventure,,0  
Train Sim,,0  
Trivia / Game Show,,0

Truck Racing,,2  
Videos,,0  
Virtual Life,,2  
Volleyball,,2  
Wakeboarding,,2  
Wargame,,1  
Web Browser,,0  
Wrestling,,1  
WWI Flight Sim,,3  
WWII Flight Sim,,3

---

### Listing H.13: Gamespot Physics Index Mapping

---

```
ALTER TABLE games ADD INDEX ( name );  
ALTER TABLE games ADD INDEX ( category );  
ALTER TABLE games ADD INDEX ( subcategory );
```

---

### Listing H.14: Add Index (Games)

---

```
ALTER TABLE mapping ADD INDEX ( category );  
ALTER TABLE mapping ADD INDEX ( subcategory );
```

---

### Listing H.15: Add Index (Mapping)

## H.4 MAME Database

---

```
#!/bin/bash  
  
# The database user  
USER=root  
  
echo -n "Enter database password for user $USER > "  
read PASS  
echo pass=$PASS  
  
echo Creating mame database ...  
echo "DROP DATABASE IF EXISTS mame;" >database.sql  
echo "CREATE DATABASE mame;" >>database.sql  
echo "USE mame;" >>database.sql  
echo "CREATE TABLE games ( " >>database.sql  
echo "   source char(32) NOT NULL, " >>database.sql  
echo "   gameid char(64) NOT NULL, " >>database.sql  
echo "   platform char(32) NOT NULL, " >>database.sql
```

```

echo " name char(64) NOT NULL, " >>database.sql
echo " category char(64) NOT NULL, " >>database.sql
echo " subcategory char(64) NOT NULL, " >>database.sql
echo " reldate date" >>database.sql
echo " ) ENGINE=MyISAM DEFAULT CHARSET=latin1;" >>database.sql
mysql -f -u $USER -p$PASS <database.sql

echo Copying files from Data ...
if [ ! -f mameinfo.xml ]
then
  cp ../Data/mameinfo.xml .
else
  echo Using existing file mameinfo.xml
fi
if [ ! -f catver.ini ]
then
  cp ../Data/catver.ini .
else
  echo Using existing file catver.ini
fi

echo Processing data into csv ...
perl analyze.pl >mame.csv

echo Importing data into db ...
#mysql -u $USER -p$PASS mame <import.sql
FILE=mame.csv
echo file=$FILE
echo "LOAD DATA LOCAL INFILE '$FILE' " >import.sql
echo " INTO TABLE games " >>import.sql
echo " FIELDS TERMINATED BY \",\" " >>import.sql
echo " ENCLOSED BY \"\\\" \" >>import.sql
echo " ESCAPED BY \"\\\\\" \" >>import.sql
echo " LINES TERMINATED BY \"\\n\" \" >>import.sql
echo " IGNORE 1 LINES " >>import.sql
mysql -u $USER -p$PASS mame <import.sql

echo Cleaning up temp files ...
rm -f mameinfo.xml catver.ini database.sql mame.csv

echo Stats ...
echo "select * from games limit 1;" | mysql -t -u $USER -p$PASS mame
echo "select count(1) as GamesInDatabase from games;" | mysql -t -u $USER -
  p$PASS mame

```

---

**Listing H.16: MAME Processor**

---

```
#!/usr/bin/perl
#
# mame_analyze.pl
#
# Mame XML file data-extractor: process Mame's INI and XML files into a CSV
  format
# for relational database import
#
use IO::Handle qw(flush);
# Load gamelist and create category stats from catver
printf STDERR "Loading catver.ini ...\\n";
%games=();
%category=();
$count=0;
open (INPUT, "< catver.ini");
while (defined ($line = <INPUT>)) {
  chomp $line;
  if ($line =~ m/(.*)=(.*)/) {
    if (!(%games{$1})) {
      %games{$1} = $2;
      $count++;
      %category{$2}++;
    }
  }
  $count++;
}
close (INPUT);
# Log summary
printf STDERR "Read $count games ...\\n";
printf STDERR "Categories are:\\n";
# Log category statistics
foreach $item (sort keys %category) {
  printf STDERR $item . " - " . %category{$item} . " games\\n";
}
# Read main mameinfo.xml file (slurp mode)
print "Loading mameinfo.xml ...\\n";
undef $/;
open (INPUT, "< mameinfo.xml");
$mameinfo = <INPUT>;
close (INPUT);
# Process gameinfo to generate CSV
printf STDERR "Finding per-game info ...\\n";
$gamecount=0;
foreach $item (sort keys %games) {
  $gameid=$item;
```

```

$description="";
$year=0;
$size=0;
if ($mameinfo =~ m/\<game name="\$item".*?\>(.*?)\</game\>/gms) {
    $gameinfo=$1;
    # Get data out of XML file
    if ($gameinfo =~ m/\<description\>(.*?)\</description\>/) {
        $description=$1;
    }
    if ($gameinfo =~ m/\<year\>(.*?)\</year\>/) {
        $year=$1;
    }
    # Determine total rom bytes
    while ($gameinfo =~ m/\<rom .*size="(.*?)".*\>/g) {
        $size += $1;
    }
}
$category=$games{$item};
@categories=split(/\/, $category);
$main_category=$categories[0];
$main_category =~ s/^\s+//;
$main_category =~ s/\s+$//;
$sub_category=$categories[1];
$sub_category =~ s/^\s+//;
$sub_category =~ s/\s+$//;
# Output CSV line
print "\"$gameid\", \"$description\", \"$year\", \"$size\", \"$main_category\", \"$
    $sub_category\"\n";
# Progress logging
$gamecount++;
if (($gamecount % 100)==0) {
    printf STDERR "$gamecount ... ";
    flush(STDERR);
}
}
printf STDERR "Done!\n";

```

---

### Listing H.17: MAME Analyzer

---

```

#!/bin/bash

# The database user
USER=root

echo -n "Enter database password for user $USER > "
read PASS

```

```

echo pass=$PASS

echo Category mapping ...
if [ ! -f physicsmapping.txt ]; then
  sudo rm /tmp/physicsmapping.txt
  echo "SELECT DISTINCT category,subcategory,'" AS physics " >map.sql
  echo " FROM games " >>map.sql
  echo " ORDER BY category,subcategory" >>map.sql
  echo " INTO OUTFILE '/tmp/physicsmapping.txt' " >>map.sql
  echo " FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '\"'" >>map.sql
  echo " LINES TERMINATED BY '\\n'" >>map.sql
  mysql --column-names -u $USER -p$PASS mame <map.sql
  cp /tmp/physicsmapping.txt physicsmapping.csv
  echo Edit physicsmapping.csv and rename to physicsmapping.txt afterwards.
  wc -l physicsmapping.csv
else
  echo Creating mapping database ...
  echo "USE mame;" >database.sql
  echo "DROP TABLE IF EXISTS mapping;" >>database.sql
  echo "CREATE TABLE mapping ( " >>database.sql
  echo "  category char(64) NOT NULL, " >>database.sql
  echo "  subcategory char(64) NOT NULL, " >>database.sql
  echo "  physics int" >>database.sql
  echo " ) ENGINE=MyISAM DEFAULT CHARSET=latin1;" >>database.sql
  mysql -f -u $USER -p$PASS <database.sql

  echo Load mapping file ...
  echo "LOAD DATA LOCAL INFILE 'physicsmapping.txt' " >import.sql
  echo " INTO TABLE mapping " >>import.sql
  echo " FIELDS TERMINATED BY ',' " >>import.sql
  echo " ENCLOSED BY '\"' ESCAPED BY '\\\\' " >>import.sql
  echo " LINES TERMINATED BY '\\n'" >>import.sql
  echo " IGNORE 1 LINES " >>import.sql
  mysql -u $USER -p$PASS mame <import.sql

  echo Adding index to tables ...
  mysql -f -u $USER -p$PASS mame <add_index_games.sql
  mysql -f -u $USER -p$PASS mame <add_index_mapping.sql
fi

echo Clean up ...
rm -f database.sql import.sql map.sql *~ inc

```

---

### Listing H.18: MAME Mapper

---

```
category,subcategory,physics
```

Ball & Paddle,,1  
BIOS,,0  
Breakout,,2  
Breakout \*Mature\*,,2  
Casino,,0  
Casino \*Mature\*,,0  
Climbing,,1  
Driving,,2  
Driving,1st Person,2  
Driving,Boat,2  
Driving,Plane,2  
Driving,Race,2  
Driving,Race (chase view),2  
Driving,Race (chase view) Bike,2  
Driving,Race 1st P Bike,2  
Driving,Race 1st Person,2  
Driving,Race Bike,2  
Driving,Race Track,2  
Fighter,2.5D,1  
Fighter,2D,1  
Fighter,3D,2  
Fighter,Field,1  
Fighter,Misc.,1  
Fighter,Multiplay,1  
Fighter,Versus,1  
Fighter,Versus \*Mature\*,1  
Fighter,Versus Co-op,1  
Fighter,Vertical,1  
Maze,,0  
Maze,Digging,1  
Maze,Digging \*Mature\*,1  
Maze,Driving,2  
Maze,Fighter,1  
Maze,Outline,0  
Maze,Outline \*Mature\*,0  
Maze,Shooter Large,2  
Maze,Shooter Small,1  
Maze,Shooter Small \*Mature\*,1  
Maze,Surround,1  
Maze \*Mature\*,,0  
Mini-Games,,0  
Misc.,,0  
Misc. \*Mature\*,,0  
Multiplay,,0  
Multiplay \*Mature\*,,0  
Not Classified,,0



Pinball,,3  
Pinball,Pachinko \*Mature\*,3  
Pinball \*Mature\*,,3  
Platform,Fighter,1  
Platform,Fighter Scrolling,1  
Platform,Run Jump,1  
Platform,Run Jump \*Mature\*,1  
Platform,Run Jump Scrolling,1  
Platform,Shooter,1  
Platform,Shooter Scrolling,1  
Puzzle,,0  
Puzzle,Cards,0  
Puzzle,Cards \*Mature\*,0  
Puzzle,Drop,0  
Puzzle,Drop \*Mature\*,0  
Puzzle,Match,0  
Puzzle,Match \*Mature\*,0  
Puzzle,Maze,0  
Puzzle,Outline,0  
Puzzle,Outline \*Mature\*,0  
Puzzle,Sliding,0  
Puzzle,Sliding \*Mature\*,0  
Puzzle,Toss,1  
Puzzle,Toss \*Mature\*,1  
Puzzle \*Mature\*,,0  
Quiz,Chinese,0  
Quiz,English,0  
Quiz,English \*Mature\*,0  
Quiz,French,0  
Quiz,German,0  
Quiz,Italian,0  
Quiz,Japanese,0  
Quiz,Japanese \*Mature\*,0  
Quiz,Korean,0  
Quiz,Music English,0  
Quiz,Music Japanese,0  
Quiz,Spanish,0  
Rhythm,,0  
Rhythm,Dance,0  
Rhythm,Instruments,0  
Shooter,1st Person,2  
Shooter,3rd Person,2  
Shooter,Command,2  
Shooter,Driving,2  
Shooter,Driving (chase view),2  
Shooter,Driving 1st Person,2

Shooter,Driving Diagonal,1  
Shooter,Driving Horizontal,1  
Shooter,Driving Vertical,1  
Shooter,Field,2  
Shooter,Field \*Mature\*,2  
Shooter,Flying,2  
Shooter,Flying (chase view),2  
Shooter,Flying 1st Person,2  
Shooter,Flying Diagonal,1  
Shooter,Flying Horizontal,1  
Shooter,Flying Horizontal \*Mature\*,1  
Shooter,Flying Vertical,1  
Shooter,Flying Vertical \*Mature\*,1  
Shooter,Gallery,1  
Shooter,Gallery \*Mature\*,1  
Shooter,Gun,2  
Shooter,Misc.,1  
Shooter,Misc. Horizontal,1  
Shooter,Misc. Vertical,1  
Shooter,Versus,1  
Shooter,Walking,1  
Sports,,1  
Sports,Armrestling,1  
Sports,Baseball,2  
Sports,Basketball,2  
Sports,Bowling,3  
Sports,Boxing,2  
Sports,Bull Fighting,2  
Sports,Darts,3  
Sports,Dodgeball,2  
Sports,Fishing,1  
Sports,Football Amer.,2  
Sports,Football Rugby,2  
Sports,Golf,3  
Sports,Handball,2  
Sports,Hang Gliding,2  
Sports,Hockey,2  
Sports,Horse Racing,1  
Sports,Horseshoes,2  
Sports,Multiplay,1  
Sports,Ping pong,2  
Sports,Pool,3  
Sports,Pool \*Mature\*,3  
Sports,Shuffleboard,2  
Sports,Skateboarding,2  
Sports,Skiing,2

```
Sports,SkyDiving,2
Sports,Soccer,2
Sports,Sumo,2
Sports,Swimming,2
Sports,Tennis,2
Sports,Track & Field,2
Sports,Volleyball,2
Sports,Wrestling,1
Tabletop,,0
Tabletop,Hanafuda,0
Tabletop,Hanafuda *Mature*,0
Tabletop,Mahjong,0
Tabletop,Mahjong *Mature*,0
Tabletop,Othello,0
Tabletop,Othello *Mature*,0
Tabletop,Renju,0
Tabletop,Shougi,0
Tabletop *Mature*,,0
```

---

### Listing H.19: MAME Physics Index Mapping

---

```
ALTER TABLE games ADD INDEX ( name );
ALTER TABLE games ADD INDEX ( category );
ALTER TABLE games ADD INDEX ( subcategory );
```

---

### Listing H.20: MAME Add Index (Games)

---

```
ALTER TABLE mapping ADD INDEX ( category );
ALTER TABLE mapping ADD INDEX ( subcategory );
```

---

### Listing H.21: MAME Add Index (Mapping)

---

## H.5 Database Merge

---

```
#!/bin/bash

# The database user
USER=root

echo -n "Enter database password for user $USER > "
read PASS
echo pass=$PASS
```

```

echo Creating databases ...
echo "DROP DATABASE IF EXISTS gamestats;" >database.sql
echo "CREATE DATABASE gamestats;" >>database.sql
echo "USE gamestats;" >>database.sql
echo "CREATE TABLE games ( " >>database.sql
echo "  source char(32) NOT NULL, " >>database.sql
echo "  gameid char(64) NOT NULL, " >>database.sql
echo "  platform char(32) NOT NULL, " >>database.sql
echo "  name char(64) NOT NULL, " >>database.sql
echo "  category char(64) NOT NULL, " >>database.sql
echo "  subcategory char(64) NOT NULL, " >>database.sql
echo "  reldate date, " >>database.sql
echo "  physics int " >>database.sql
echo " ) ENGINE=MyISAM DEFAULT CHARSET=latin1;" >>database.sql
mysql -f -u $USER -p$PASS <database.sql

echo Processing gamespot ...
sudo rm -f /tmp/result.txt
mysql -t -u $USER -p$PASS gamespot <extract.sql
cp /tmp/result.txt gamespot.csv

echo Processing gamebase ...
sudo rm -f /tmp/result.txt
mysql -t -u $USER -p$PASS gamebase <extract.sql
cp /tmp/result.txt gamebase.csv

echo Processing mame ...
sudo rm -f /tmp/result.txt
mysql -t -u $USER -p$PASS mame <extract.sql
cp /tmp/result.txt mame.csv

echo Merging data ...
FILE=gamestats.csv
cat gamespot.csv gamebase.csv mame.csv >$FILE
echo file=$FILE
echo "LOAD DATA LOCAL INFILE '$FILE' " >import.sql
echo " INTO TABLE games " >>import.sql
echo " FIELDS TERMINATED BY \",\" " >>import.sql
echo " ENCLOSED BY \"\\\" \" >>import.sql
echo " ESCAPED BY \"\\\\\" \" >>import.sql
echo " LINES TERMINATED BY \"\\n\" \" >>import.sql
echo " IGNORE 1 LINES " >>import.sql
mysql -u $USER -p$PASS gamestats <import.sql

echo Adding index to tables ...
mysql -f -u $USER -p$PASS gamestats <add_index_games.sql

```

```

echo Updating mame and gamebase platform labels ...
echo "UPDATE games SET platform=\"Arcade Games\" where platform=\"arcade\;" >
    update.sql
echo "UPDATE games SET platform=\"Commodore 64\" where platform=\"c64\;" >>
    update.sql
echo "UPDATE games SET platform=\"VIC-20\" where platform=\"vic20\;" >>update.
    sql
echo "UPDATE games SET platform=\"Sinclair ZX81/Spectrum\" where platform=\"zx\;"
    " >>update.sql
echo "UPDATE games SET platform=\"Atari ST\" where platform=\"st\;" >>update.sql
mysql -f -u $USER -p$PASS gamestats <update.sql

echo Cleaning up ...
rm -f gamespot.csv gamebase.csv mame.csv *~

echo Stats ...
echo "select * from games limit 1;" | mysql -t -u $USER -p$PASS gamestats
echo "select count(1) as GamesInDatabase from games;" | mysql -t -u $USER -
    p$PASS gamestats

```

---

### Listing H.22: Merge Processor

---

```

DROP DATABASE IF EXISTS gamestats;
CREATE DATABASE gamestats;
USE gamestats;
CREATE TABLE games (
    source char(32) NOT NULL,
    gameid char(64) NOT NULL,
    platform char(32) NOT NULL,
    name char(64) NOT NULL,
    category char(64) NOT NULL,
    subcategory char(64) NOT NULL,
    reldate date,
    physics int
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

---

### Listing H.23: Merge Database Create

---

```

LOAD DATA LOCAL INFILE 'gamestats.csv'
    INTO TABLE games
    FIELDS TERMINATED BY ","
    ENCLOSED BY "\""
    ESCAPED BY "\\"
    LINES TERMINATED BY "\n"
    IGNORE 1 LINES

```

---

### Listing H.24: Merge Database Import

---

```
SELECT games.*,mapping.physics
INTO OUTFILE '/tmp/result.txt'
FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
LINEs TERMINATED BY '\n'
FROM games,mapping
WHERE reldate>"1970-01-01" AND reldate<"2010-01-01" AND games.category=mapping.
category AND games.subcategory=mapping.subcategory;
```

---

### Listing H.25: Merge Database Extract

---

```
UPDATE games SET platform="Arcade Games" where platform="arcade";
UPDATE games SET platform="Commodore 64" where platform="c64";
UPDATE games SET platform="VIC-20" where platform="vic20";
UPDATE games SET platform="Sinclair ZX81/Spectrum" where platform="zx";
UPDATE games SET platform="Atari ST" where platform="st";
```

---

### Listing H.26: Merge Database Update

---

```
ALTER TABLE games ADD INDEX ( source );
ALTER TABLE games ADD INDEX ( platform );
ALTER TABLE games ADD INDEX ( name );
ALTER TABLE games ADD INDEX ( category );
ALTER TABLE games ADD INDEX ( subcategory );
ALTER TABLE games ADD INDEX ( reldate );
ALTER TABLE games ADD INDEX ( physics );
```

---

### Listing H.27: Merge Add Index (Games)

## H.6 Graph Generation

### Graph Generation Script

---

```
#!/bin/bash

# The database user
USER=root
echo -n "Enter database password for user $USER > "
```

```

read PASS
echo pass=$PASS

for i in *.plt *.bar; do

BASE=${i%.plt}
BASE=${BASE%.bar}
echo base=$BASE

echo Creating data ...
if [ -f $BASE.sql ]; then
  mysql -N -u $USER -p$PASS gamestats <$BASE.sql >data.txt
fi
if [ -f $BASE.1.sql ]; then
  mysql -N -u $USER -p$PASS gamestats <$BASE.1.sql >data.1.txt
fi
if [ -f $BASE.2.sql ]; then
  mysql -N -u $USER -p$PASS gamestats <$BASE.2.sql >data.2.txt
fi
if [ -f $BASE.3.sql ]; then
  mysql -N -u $USER -p$PASS gamestats <$BASE.3.sql >data.3.txt
fi

echo Creating graph ...
if [ -f $BASE.plt ]; then
  echo Gnuplot ...
  gnuplot <$BASE.plt
  mv graph.eps $BASE.eps
fi
if [ -f $BASE.bar ]; then
  echo Bargraph.pl ...
  cat $BASE.bar data.txt >bargraph.data
  ./bargraph.pl bargraph.data >$BASE.eps
fi

echo Creating pdf ...
epstopdf --outfile=$BASE.pdf $BASE.eps

#echo Creating png ...
#convert -density 200 $BASE.eps -flatten $BASE.png

done

echo Cleaning up ...
rm -f data.txt data.?.txt *~ inc

```

---

## Listing H.28: Plot Processor

### **bargraph.pl**

The helper script *bargraph.pl* used to generate bar-graphs with gnuplot is a *Perl* based open source software by Derek Bruening and can be obtained from <http://www.burningcutlery.com/derek/bargraph> or <http://code.google.com/p/bargraphgen>.

### **Games per Category Bar-Graph**

---

```
select category, count(1)/1000 as N from games group by category order by N desc
limit 20;
```

---

## Listing H.29: Games per Category (SQL)

---

```
font=Helvetica
colors=grey4
extraops=set size 1.0,1.3
extraops=set mytics 5
xlabelshift=0,2
ylabelshift=3,0
fontsz=8
yformat=%4.0f
xlabel=Video Game Category
ylabel=Number of Games (1000s)
```

---

## Listing H.30: Games per Category (bargraph.pl)

### **Games per Platform Bar-Graph**

---

```
select platform, count(1)/1000 as N from games group by platform order by N desc
limit 20;
```

---

## Listing H.31: XYZ (Gnuplot)



---

```
font=Helvetica
colors=grey4
extraops=set size 1.0,1.3
extraops=set mytics 5
xlabelshift=0,3
ylabelshift=3,0
fontsz=8
yformat=%4.0f
xlabel=Video Game Platform
ylabel=Number of Games (1000s)
```

---

### Listing H.32: Games per Platform (bargraph.pl)

## Games per Year Graph

---

```
select year(reldate), count(1) as N from games group by year(reldate);
```

---

### Listing H.33: Games per Year (SQL)

---

```
# set term pos eps color enhanced defaultplex "Helvetica" 12
set term postscript eps enhanced
set output 'graph.eps'
set key left top
set xlabel "Year of Release"
set ylabel "Games Released"
set format y "%.0f"
set yrange [0:10000]
set mxtics 5
set mytics 5
set grid
set style line 2 lt 1 lw 4
#  $f(x)=a*x+b$ 
# fit  $f(x)$  "data.txt" via  $a,b$ 
# plot "data.txt" smooth csplines title "Gamespot + Gamebase + Mame Databases
  Combined" with lines ls 1,  $f(x)$  title "Linear Fit" with lines ls 2
plot "data.txt" smooth csplines title "Gamespot + Gamebase + Mame Databases
  Combined" with steps ls 2
```

---

### Listing H.34: Games per Year (Gnuplot)

## Physics per Year Cleaned/Smoothed Graph

---

```
select year(reldate) as 'Year', avg(physics) as 'Average', stddev(physics)/sqrt(
    count(1)) as 'Error' from games where year(reldate)>1974 AND platform!="
    BlackBerry" AND platform!="iPhone" AND platform!="Pinball" group by year(
    reldate);
```

---

### Listing H.35: Physics per Year Cleaned/Smoothed (SQL)

---

```
# set term pos eps color enhanced defaultplex "Helvetica" 12
set term postscript eps enhanced
set output 'graph.eps'
set key left top
set xlabel "Year of Release"
set ylabel "Average Physics Index"
set format y "%.1f"
set yrange [0:3]
set mxtics 5
set mytics 5
set grid
set style line 2 lt 2 lw 5
set key top right
plot "data.txt" smooth csplines title "All Databases/Some Platforms" with lines
    ls 2, "data.txt" title "Error of Average" with yerrorbars
```

---

### Listing H.36: Physics per Year Cleaned/Smoothed (Gnuplot)

## Physics per Year Graph

---

```
select year(reldate) as 'Year', avg(physics) as 'Average', stddev(physics)/sqrt(
    count(1)) as 'Error' from games where year(reldate)>1974 group by year(
    reldate);
```

---

### Listing H.37: Physics Per Year (SQL)

---

```
# set term pos eps color enhanced defaultplex "Helvetica" 12
set term postscript eps enhanced
set output 'graph.eps'
set key left top
set xlabel "Year of Release"
set ylabel "Average Physics Index"
set format y "%.1f"
set yrange [0:3]
set mxtics 5
set mytics 5
```

```

set grid
set style line 2 lt 2 lw 5
set key top right
# f(x)=a*x+b
# fit f(x) "data.txt" via a,b
# plot "data.txt" smooth csplines title "Gamespot + Gamebase + Mame Databases
  Combined" with lines ls 1, f(x) title "Linear Fit" with lines ls 2
plot "data.txt" smooth csplines title "All Databases/All Platforms" with lines ls
  2, "data.txt" title "Error of Average" with yerrorbars

```

---

### Listing H.38: Physics Per Year (Gnuplot)

#### Physics per Year Linear-Fit Graph

```

select year(reldate) as 'Year', avg(physics) as 'Average' from games where year(
  reldate)>1974 AND year(reldate)<1985 AND platform!="BlackBerry" AND platform
  !="iPhone" AND platform!="Pinball" group by year(reldate);

```

---

### Listing H.39: Physics per Year Linear-Fit 1(SQL)

```

select year(reldate) as 'Year', avg(physics) as 'Average' from games where year(
  reldate)>1984 AND year(reldate)<2004 AND platform!="BlackBerry" AND platform
  !="iPhone" AND platform!="Pinball" group by year(reldate);

```

---

### Listing H.40: Physics per Year Linear-Fit 2 (SQL)

```

select year(reldate) as 'Year', avg(physics) as 'Average' from games where year(
  reldate)>2003 AND platform!="BlackBerry" AND platform!="iPhone" AND platform
  !="Pinball" group by year(reldate);

```

---

### Listing H.41: Physics per Year Linear-Fit 3 (SQL)

```

# set term pos eps color enhanced defaultplex "Helvetica" 12
set term postscript eps enhanced
set output 'graph.eps'
set key left top
set xlabel "Year of Release"
set ylabel "Average Physics Index"
set format y "%.1f"
set mxtics 5
set mytics 5
set grid
set style line 2 lt 2 lw 5

```

```

set key top left

f(x)=a1*x+b1
a1 = -0.1
b1 = 200
fit f(x) "data.1.txt" via a1,b1

g(x)=a2*x+b2
fit g(x) "data.2.txt" via a2,b2

r(x)=a3*x+b3
a3 = -0.1
b3 = 200
fit r(x) "data.3.txt" via a3,b3

set multiplot
set yrange [0:2]
set xrange [1975:2010]
plot "data.1.txt" title "Arcade Era" with points, "data.2.txt" title "Console and
PC Era" with points, "data.3.txt" title "Portable Device Era" with points
set key top center
set noautoscale
plot f(x) title "linear fit" with lines, g(x) title "linear fit" with lines, r(x)
title "linear fit" with lines
set nomultiplot

```

---

**Listing H.42:** Physics per Year (Gnuplot)

## H.7 playtrulyrandom.com

---

```

<?php
$config = array("server" => "localhost",
               "user"    => "xyz",
               "password" => "xyz",
               "database" => "xyz");
?>

```

---

**Listing H.43:** PTR Configuration

```

<?php

/*
playtrulyrandom.com Webservice Component

```

(c) A. Schiffler, 2008-2010, GPL

*data.php - Retrieve the latest entropy pool data as raw binary blob.*

```
*/  
  
include ("config.inc.php");  
  
/* make DB connection */  
$mysqli = new mysqli($sconfig['server'], $config['user'], $config['password'],  
    $config['database']);  
  
/* check connection */  
if (mysqli_connect_errno()) {  
    printf("Connect failed: %s\n", mysqli_connect_error());  
    exit();  
}  
  
/* get source name */  
$sourcename = $_REQUEST["source"];  
if (!$sourcename)  
{  
    printf("");  
    exit(0);  
}  
  
/* reference position for bits reader */  
$lastbitsid=0;  
$lastbitsindex=0;  
  
/* Find sourceid, bits marker */  
$sourceid=0;  
$sourcename = mysqli_real_escape_string ($mysqli, $sourcename);  
$query = sprintf("SELECT sourceid FROM source WHERE name='%s'", $sourcename);  
if ($result = mysqli_query($mysqli, $query)) {  
    if ($row = mysqli_fetch_assoc()) {  
        $sourceid = $row["sourceid"];  
    }  
    $result->close();  
}  
  
/* do we have a source now? */  
if (!$sourceid)  
{  
    printf("");  
    exit(0);  
}
```

```

/* count hits on source */
$query = sprintf("UPDATE source SET hits=hits+1 WHERE sourceid=%d",$sourceid);
if (!($result = $mysqli->query($query)) {
    printf("0");
    exit(0);
}

/* how many bits were requested*/
$bytesRequested=$_REQUEST["n"] + 0;
if ($bytesRequested<1)
{
    printf("");
    exit();
}

/* limit request to 1Mbyte */
if ($bytesRequested>1024*1024) {
    $bytesRequested = 1024*1024;
}
$bitsRequested=$bytesRequested * 8;

/* Retrieve last submission ID of source */
$query = sprintf("SELECT bitsid AS lastId FROM bits WHERE sourceid=%d ORDER BY
    bitsid DESC LIMIT 1", $sourceid);
if ($result = $mysqli->query($query)) {
    if ($row = $result->fetch_assoc()) {
        $lastId = $row["lastId"];
    }
    $result->close();
}

/* loop to fill bits request */
$bitsLen = 0;
$bitsBucket = "";

/* do we need more bits */
while ($bitsLen < $bitsRequested) {
    /* add bits from records we have not retrieved yet */
    $query = sprintf("SELECT bitsid,data FROM bits WHERE bitsid<%d AND sourceid=%d
        ORDER BY bitsid DESC LIMIT 1", $lastId, $sourceid);
    if ($result = $mysqli->query($query)) {
        if ($row = $result->fetch_assoc()) {
            /* add these bits to the bucket */
            $bitsBucket .= $row["data"];
            $bitsLen = strlen($bitsBucket);
        }
    }
}

```

```

        $lastId = $row["bitsid"];
    } else {
        /* return what we've got */
        $bitsLen = $bitsRequested;
    }
    $result->close();
}
}

/* did we read enough bits? determine bits to send */
$bitsLen = strlen($bitsBucket);
if ($bitsLen < $bitsRequested) {
    /* adjust to actual length */
    $bitsToSend = $bitsLen;
} else {
    /* send what was requested */
    $bitsToSend = $bitsRequested;
}

/* close connection */
$mysqli->close();

/* convert bit string into binary blob */
$binary = "";
for ($pos=0; $pos<$bitsToSend; $pos += 8)
{
    $byteBits = substr($bitsBucket, $pos, 8);
    $number = bindec($byteBits);
    $binary .= pack("c", $number);
}

/* HTTP header for binary data */
$bytesToSend = $bitsToSend / 8;
header("Content-Type: application/octet-stream");
header("Content-Disposition: attachment; filename=\"entropy.dat\"");
header("Content-Length: " . $bytesToSend);

echo $binary;
?>

```

---

### Listing H.44: PTR Data Server

---

```
<?php
```

```

/*
playtrulyrandom.com Webservice Component

```

(c) A. Schiffler, 2008-2010, GPL

```
display.php - Generates a black and white image from entropy pool data.
*/

include ("config.inc.php");

/* make DB connection */
$mysqli = new mysqli($sconfig['server'], $config['user'], $config['password'],
    $config['database']);

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

/* reset data */
$data = "";

/* get source name, use default if needed */
$id = $_REQUEST["id"] + 0;
if ($id)
{
    /* Retrieve some data from bits-store */
    $query = sprintf("SELECT data FROM bits WHERE bitsid=%d", $id);
    if ($result = $mysqli->query($query)) {
        if ($row = $result->fetch_assoc()) {
            $data = $row["data"];
        }
        $result->close();
    }
}

// Blocksize
$size=6;

// Create a new image instance
$width=$size*strlen($data);
$height=$size;

$canvas = imagecreatetruecolor($width+2, $height+2);

// Allocate colors
$black = imagecolorallocate($canvas, 0, 0, 0);
$white = imagecolorallocate($canvas, 255, 255, 255);
```



```

$gray = imagecolorallocate($canvas, 222, 222, 222);

// Frame
imagerectangle($canvas, 0, 0, $width+1, $height+1, $gray);
imagefilledrectangle($canvas, 1, 1, $width, $height, $white);

// Bits
$xpos=1;
$bits=$data;

$bitsLen = strlen($bits);
for ($i=0; $i<$bitsLen; $i++) {
    $nextBit = $bits{$i};
    if ($nextBit == "0") {
        $curColor = $black;
    } else {
        $curColor = $white;
    }
    imagefilledrectangle($canvas, $xpos, 1, $xpos+$size, $size, $curColor);
    $xpos += $size;
}

// Handle output
if(function_exists('imagegif')) {
    // For GIF
    header('Content-type: image/gif');
    imagegif($canvas);
}
imagedestroy($canvas);

?>

```

---

### Listing H.45: PTR Image Display

---

```

<?php

/*
playtrulyrandom.com Webservice Component
(c) A. Schiffler, 2008-2010, GPL

retrieve.php - Retrieve entropy pool data as string of 0/1 characters.
*/

include ("config.inc.php");

/* make DB connection */

```

```

$mysqli = new mysqli($sconfig['server'], $sconfig['user'], $sconfig['password'],
    $sconfig['database']);

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

/* get source name */
$sourcename = $_REQUEST["source"];
if (!$sourcename)
{
    printf("");
    exit(0);
}

/* get user name */
$username = $_REQUEST["user"];
if (!$username)
{
    printf("");
    exit(0);
}

/* reference position for bits reader */
$lastbitsid=0;
$lastbitsindex=0;

/* Find sourceid, bits marker */
$sourceid=0;
$sourcename = $mysqli->real_escape_string ($sourcename);
$query = sprintf("SELECT sourceid FROM source WHERE name='%s'", $sourcename);
if ($result = $mysqli->query($query)) {
    if ($row = $result->fetch_assoc()) {
        $sourceid = $row["sourceid"];
    }
    $result->close();
}

/* do we have a source now? */
if (!$sourceid)
{
    printf("");
    exit(0);
}

```

```

/* count hits on source */
$query = sprintf("UPDATE source SET hits=hits+1 WHERE sourceid=%d",$sourceid);
if (!($result = $mysqli->query($query)) {
    printf("0");
    exit(0);
}

/* Find userid, bitmarkers */
$userid = 0;
$username = $mysqli->real_escape_string ($username);
$query = sprintf("SELECT userid,lastbitsid,lastbitsindex FROM user WHERE name='%s
    '", $username);
if ($result = $mysqli->query($query)) {
    if ($row = $result->fetch_assoc()) {
        $userid = $row["userid"];
        $lastbitsid = $row["lastbitsid"];
        $lastbitsindex = $row["lastbitsindex"];
    }
    $result->close();
}

/* have we seen this user? */
if ($userid == 0)
{
    /* create new user entry */
    $query = sprintf("INSERT INTO user (userid,name,lastbitsid,lastbitsindex,
        timestamp) VALUES (NULL,'%s',0,0,NOW())", $username);
    if ($result = $mysqli->query($query)) {
        $userid = $mysqli->insert_id;
    }
}

/* do we have a user now? */
if (!$userid)
{
    printf("");
    exit(0);
}

/* how many bits were requested*/
$bitsRequested=$_REQUEST["n"] + 0;
if ($bitsRequested<1)
{
    printf("");
    exit();
}

```

```

}

/* limit bits request */
if ($bitsRequested>1024) {
    $bitsRequested = 1024;
}

/* loop to fill bits request */
$bitsLen = 0;
$bitsBucket = "";
/* maybe add bits from last record which we haven't used yet */
if ($lastbitsindex>0) {
    $query = sprintf("SELECT bitsid,data FROM bits WHERE bitsid=%d LIMIT 1",
        $lastbitsid);
    if ($result = $mysqli->query($query)) {
        if ($row = $result->fetch_assoc()) {
            $bitsid=$row["bitsid"];
            if ($bitsid==$lastbitsid)
            {
                /* add remaining bits in previous segment to the bucket */
                $lastdatalen = strlen($row["data"]);
                $bitsBucket .= substr($row["data"], $lastbitsindex);
                $bitsLen = strlen($bitsBucket);
            }
        }
    }
    $result->close();
}

/* do we need more bits */
while ($bitsLen < $bitsRequested) {
    /* add bits from records we have not retrieved yet */
    $query = sprintf("SELECT bitsid,data FROM bits WHERE bitsid>%d LIMIT 1",
        $lastbitsid);
    if ($result = $mysqli->query($query)) {
        if ($row = $result->fetch_assoc()) {
            /* add these bits to the bucket */
            $lastdatalen = strlen($row["data"]);
            $bitsBucket .= $row["data"];
            $bitsLen = strlen($bitsBucket);
            $lastbitsid = $row["bitsid"];
        } else {
            /* return what we've got */
            $bitsLen = $bitsRequested;
        }
    }
    $result->close();
}

```

```

    }
}

/* calculate size of bits we have now */
$bitsLen = strlen($bitsBucket);

/* did we overread? determine segment index */
if ($bitsLen > $bitsRequested)
{
    /* calculate new index for next retrieval */
    $lastbitsindex = $lastdatalen - $bitsLen + $bitsRequested;
} else {
    /* no need anymore for this segment */
    $lastbitsindex = 0;
}

/* did we read enough bits? determine bits to send */
if ($bitsLen < $bitsRequested) {
    /* adjust to actual length */
    $bitsToSend = $bitsLen;
} else {
    /* send what was requested */
    $bitsToSend = $bitsRequested;
}

/* mix in PRNG numbers to make this returned bit string unique */
$bits="";
$bitsLen = strlen($bitsBucket);
for ($i=0; $i<$bitsToSend; $i++) {
    $nextBit = $bitsBucket{$i};
    if (mt_rand(0,1)) {
        /* keep bits */
        $bits .= $nextBit;
    } else {
        /* flip bits */
        if ($nextBit=="0")
        {
            $bits .= "1";
        } else {
            $bits .= "0";
        }
    }
}

/* return result */
if ($bitsToSend) {

```

```

    /* return result */
    printf($bits);
    /* update source record, ignore errors */
    $query = sprintf("UPDATE source SET retrieved=retrieved+%d WHERE sourceid=%d",
        $bitsToSend, $sourceid);
    $result = $mysqli->query($query);
    /* update user record, ignore errors */
    $query = sprintf("UPDATE user SET lastbitsid=%d,lastbitsindex=%d WHERE userid
        =%d", $lastbitsid, $lastbitsindex, $userid);
    $result = $mysqli->query($query);
}

/* close connection */
$mysqli->close();

?>

```

---

### Listing H.46: PTR Entropy Retrieval

---

```

<?php

/*
playtrulyrandom.com Webservice Component
(c) A. Schiffler, 2008-2010, GPL

status.php - Displays the current status of PTR as webpage /w image.
*/

include ("config.inc.php");

/* make DB connection */
$mysqli = new mysqli($sconfig['server'], $config['user'], $config['password'],
    $config['database']);

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

/* Retrieve statistics of bits-store */
$query = sprintf("SELECT COUNT(1) AS numData, SUM(datasize) AS numBits FROM bits"
    );
if ($result = $mysqli->query($query)) {
    if ($row = $result->fetch_assoc()) {
        $numData = $row["numData"];
    }
}

```

```

        $numBits = $row["numBits"];
    }
    $result->close();
}

/* Retrieve number of sources */
$query = sprintf("SELECT COUNT(DISTINCT name) AS numSources FROM source ORDER BY
    name");
if ($result = $mysqli->query($query)) {
    if ($row = $result->fetch_assoc()) {
        $numSources = $row["numSources"];
    }
    $result->close();
}

/* Retrieve last submission ID */
$query = sprintf("SELECT bitsid AS lastId, bits.created AS lastTime, source.name
    AS lastSource FROM bits LEFT JOIN source ON bits.sourceid=source.sourceid
    ORDER BY bitsid DESC LIMIT 1");
if ($result = $mysqli->query($query)) {
    if ($row = $result->fetch_assoc()) {
        $lastId = $row["lastId"];
        $lastTime = $row["lastTime"];
        $lastSource = $row["lastSource"];
    }
    $result->close();
}

/* close connection */
$mysqli->close();

/* show statistics */
printf("Number of Submissions = %s<br>", $numData);
printf("Number of Bits in Pool = %s<br>", $numBits);
printf("Number of Sources = %s<br>", $numSources);
printf("Last submission on %s from %s<br>", $lastTime, $lastSource);
printf("<img src=\"display.php?id=%s\"><br>", $lastId)
?>

```

---

### Listing H.47: PTR Status Display

---

```
<?php
```

```

/*
playtrulyrandom.com Webservice Component
(c) A. Schiffler, 2008-2010, GPL

```

```

    submit.php - Submit random bits to the PTR entropy pool.
*/

include ("config.inc.php");

/* make DB connection */
$mysqli = new mysqli($sconfig['server'], $config['user'], $config['password'],
    $config['database']);

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

/* get source name */
$sourcename = $_REQUEST["source"];
if (!$sourcename)
{
    printf("0");
    exit(0);
}

/* Find sourceid */
$sourceid=0;
$sourcename = $mysqli->real_escape_string ($sourcename);
$query = sprintf("SELECT sourceid FROM source WHERE name='%s'", $sourcename);
if ($result = $mysqli->query($query)) {
    if ($row = $result->fetch_assoc()) {
        $sourceid = $row["sourceid"];
    }
    $result->close();
}

if (!$sourceid)
{
    printf("0");
    exit(0);
}

/* count hits on source */
$query = sprintf("UPDATE source SET hits=hits+1 WHERE sourceid=%d",$sourceid);
if (!($result = $mysqli->query($query))) {
    printf("0");
    exit(0);
}

```



```

}

/* check if we got any raw bit input */
$rawBits=$_REQUEST["bits"];
if (!$rawBits)
{
    printf("0");
    exit();
}

/* clean input, keep only 1 and 0 */
$bits="";
$bitsLen = strlen($rawBits);
for ($i=0; $i<$bitsLen; $i++) {
    $nextBit = $rawBits{$i};
    if (($nextBit == "0") || ($nextBit == "1"))
    {
        $bits .= $nextBit;
    }
}

/* check if we have bits left */
$bitsLen = strlen($bits);
if (!$bitsLen)
{
    printf("0");
    exit();
}

/* determine quality of bitstring */
$quality=0.0;

/* store bits in database */
$bits = $mysqli->real_escape_string ($bits);
$query = sprintf("INSERT INTO bits (bitsid, sourceid, data, datasize, dataquality
    , created, changed, used) VALUES (NULL, %d, '%s', %d, %f, NOW(), NULL, 0);",
    $sourceid,
    $bits, $bitsLen,
    (float)$quality
    );
if ($result = $mysqli->query($query)) {
    printf($bitsLen);
    /* count bits submitted on source; ignore errors */
    $query = sprintf("UPDATE source SET submitted=submitted+%d WHERE sourceid=%d",
        $bitsLen,$sourceid);
    $result = $mysqli->query($query);
}

```

```
} else {
    printf("0");
}

/* close connection */
$mysqli->close();

?>
```

---

### Listing H.48: PTR Entropy Injector

---

```
<?php

/*
playtrulyrandom.com Webservice Component
(c) A. Schiffler, 2008-2010, GPL

usage.php - Track usage statistics of user by RNG mode and duration.
*/

include ("config.inc.php");

/* make DB connection */
$mysqli = new mysqli($sconfig['server'], $config['user'], $config['password'],
    $config['database']);

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

/* get source name */
$sourcename = $_REQUEST["source"];
if (!$sourcename)
{
    printf("-1");
    exit(0);
}

/* get user name */
$username = $_REQUEST["user"];
if (!$username)
{
    printf("-2");
    exit(0);
}
```

```

}

/* Find sourceid */
$sourceid=0;
$sourcename = $mysqli->real_escape_string ($sourcename);
$query = sprintf("SELECT sourceid FROM source WHERE name='%s'", $sourcename);
if ($result = $mysqli->query($query)) {
    if ($row = $result->fetch_assoc()) {
        $sourceid = $row["sourceid"];
    }
    $result->close();
}

if (!$sourceid)
{
    printf("-3");
    exit(0);
}

/* Find userid */
$userid = 0;
$username = $mysqli->real_escape_string ($username);
$query = sprintf("SELECT userid FROM user WHERE name='%s'", $username);
if ($result = $mysqli->query($query)) {
    if ($row = $result->fetch_assoc()) {
        $userid = $row["userid"];
    }
    $result->close();
}

/* Do we have a user */
if (!$userid)
{
    printf("-4");
    exit(0);
}

/* Get mode, assume pseudo */
$hrngmode=$_REQUEST["hrngmode"] + 0;

/* Get duration length */
$duration=$_REQUEST["duration"] + 0;
if ($duration<1)
{
    printf("-5");
    exit(0);
}

```

```

}

/* Store usage */
$query = sprintf("INSERT INTO usagstats (sourceid,userid,hrngmode,durationsec,
    timestamp) VALUES (%d,%d,%d,%d,CURRENT_TIMESTAMP);", $sourceid, $userid,
    $hrngmode, $duration);
if ($result = $mysqli->query($query))
{
    $result->close();
}
else
{
    printf("-6");
    exit(0);
}

/* close connection */
$mysqli->close();

?>

```

---

### Listing H.49: PTR Usage Display

---

```

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--
-- Database: `ptr`
--
--
-- -----
--
-- Table structure for table `bits`
--
CREATE TABLE IF NOT EXISTS `bits` (
    `bitsid` int(10) unsigned NOT NULL auto_increment,
    `sourceid` int(10) unsigned NOT NULL default '0',
    `data` text character set ascii collate ascii_bin NOT NULL,
    `datasize` int(10) unsigned NOT NULL default '0',
    `dataquality` double NOT NULL,

```

```
'created' datetime NOT NULL,  
'changed' timestamp NOT NULL default CURRENT_TIMESTAMP,  
'used' int(11) NOT NULL default '0',  
PRIMARY KEY ('bitsid')  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

-----

```
--  
-- Table structure for table 'source'  
--
```

```
CREATE TABLE IF NOT EXISTS `source` (  
  `sourceid` int(10) unsigned NOT NULL auto_increment,  
  `name` varchar(64) NOT NULL,  
  `hits` int(10) unsigned NOT NULL,  
  `submitted` bigint(20) unsigned NOT NULL,  
  `retrieved` bigint(20) unsigned NOT NULL,  
  `timestamp` timestamp NOT NULL default CURRENT_TIMESTAMP on update  
    CURRENT_TIMESTAMP,  
  PRIMARY KEY (`sourceid`),  
  KEY `name` (`name`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

-----

```
--  
-- Table structure for table 'usagestats'  
--
```

```
CREATE TABLE IF NOT EXISTS `usagestats` (  
  `sourceid` int(10) unsigned NOT NULL,  
  `userid` int(10) unsigned NOT NULL,  
  `durationsec` int(10) unsigned NOT NULL,  
  `timestamp` timestamp NOT NULL default CURRENT_TIMESTAMP on update  
    CURRENT_TIMESTAMP,  
  KEY `userid` (`userid`),  
  KEY `sourceid` (`sourceid`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

-----

```
--  
-- Table structure for table 'user'  
--
```

```
CREATE TABLE IF NOT EXISTS `user` (  
  `userid` int(10) unsigned NOT NULL auto_increment,  
  `name` varchar(64) NOT NULL,  
  `lastbitsid` int(10) unsigned NOT NULL,  
  `lastbitsindex` int(10) unsigned NOT NULL,  
  `timestamp` timestamp NOT NULL default CURRENT_TIMESTAMP on update  
    CURRENT_TIMESTAMP,  
  PRIMARY KEY (`userid`),  
  KEY `name` (`name`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

---

### Listing H.50: PTR Database

## Video Documentation

### I.1 Script

#### 0:36 - Main Menu (30 sec)

The Pendulum Game is a game prototype which serves as research platform in support of the authors dissertation.

The software is written in the C# language using the XNA framework published by Microsoft and can run on any Windows PC satisfying the hardware requirements.

Following a common game design, a Main Menu screen is shown after the application was launched. With the physical user interface the player can select options which start the game, get help and information or quit the application.

The background to the menu screens is a video composed of 37 double pendulum simulations.

### **1:19 - Controls Help (15 sec)**

Selecting the Game Controls Help option and activating it displays a screen describing the user interface.

The game is designed for two players denoted as player A and player B using two Xbox game controllers or the keyboard.

### **1:40 - Game Information (36 sec)**

Selecting the Game Information menu item displays a screen with the following text:

The Pendulum Game is a simple dynamics simulation of double regular and square pendulums - hence the name. The goal is to touch the 'targets' at the top of the screen for points.

A player needs to control the friction actuators in the hinges of the pendulums to inject energy into the system and make the pendulum arms swing high enough to reach the 'targets'. Options include various gravity environments and friction models.

The oscillatory systems together with the player interactions are used to extract physical entropy or randomness during gameplay.



### **2:21 - Pendulum Video (36 sec)**

The main menu of the Pendulum game also contains an option which allows the player to access some of the media produced for the game. This is a feature which differs from common game designs.

After selecting the View Pendulum Video menu item, a randomly chosen pre-recorded video clip of a chaotic magnetic pendulum is shown.

The game contains 13 of these clips which highlight the aesthetics of chaotic pendulum motion and are used during gameplay as visual element for "reality blending".

### **3:00 - Start Game (8 sec)**

To play, the Start Game menu item is selected, which always continues to a Game Options screen.

### **3:10 - Game Options Celestial Object (25 sec)**

Integral to the game design is the interactive choice of a gravity environment for the pendulum. The player can choose from several celestial objects of the solar system including planets and moons.

Players may select the Celestial Object menu item to cycle through available objects.

A scrolling text at the bottom of the screen displays scientific information about the currently selected object.

### **3:38 - Game Options Gravity Model (25 sec)**

For some celestial objects such as Earth and Mars, the game offers the user a choice of gravity models.

In the game, each model generates a high-resolution gravity map on the surface of a celestial object. The models are based on current physical research and are used during the next stage of the game which is the "Gravity Chooser".

### **4:09 - Game Options Friction Model (19 sec)**

The simulation of the pendulum used in the game includes the choice of a more or less physically accurate friction model at the hinges.

Players can select the Rotation Friction Model menu item to cycle through the available options: None, Linear or Stribeck - an empirically determined model used in research.

### **4:30 - Game Options Entropy Source (19 sec)**

Selecting the Entropy Source menu item allows the player to switch between two sources of randomness to shuffle in-game elements such as the pendulum videos.

The Pseudo option uses the standard random number generator of the operating system.

The Play Truly Random option connects to a webservice to retrieve physically random.

#### **4:57 - Choose Gravity (11 sec)**

To play the pendulum game on the selected celestial object and the current option settings, the player must continue to the Gravity Chooser screen by activating the first option in the menu.

#### **5:12 - Gravity Chooser (1:11 min)**

The next interactive option control is called Gravity Chooser, because its main purpose is to pick a unique gravity number which will be used in the pendulum simulation that follows.

The previously selected celestial body - such as Mars in this example - is shown as a 3D sphere with a high-quality, scientifically accurate texture.

The player can rotate the sphere with the controls allowing one to place the cross-hair indicator in the center of the screen over any geographical position. Two value gauges in the upper left corner show this position as latitude and longitude numbers.

The previously selected gravitational model is applied to the location and used to calculate a position dependent gravity number which is shown in the the upper right corner.

A scrolling text at the bottom of the screen displays scientific information about the currently used gravity model.

To continue with the game, the player presses the action button.

### **6:29 - Game Screen (14 sec)**

Players now see the animated game screen which contains spherical moving targets at the top, two pendulums in the center and info displays along the bottom of the screen.

The game over screen is reached, by completing the goal of the game.

### **7:01 - Game Goal (1 min)**

The goal of the game is to swing the pendulum high enough so it will touch the moving spherical targets at the top of the screen. Each touch will count one point towards the score of the player whose pendulum hit the ball.

The dynamics of the game objects - the double pendulums and spherical targets - are controlled through a simulated physical environment that uses the previously selected options.

To make the game more interesting, two different types of double pendulum have been implemented.

The left pendulum is called a regular double pendulum and consists of two connecting rods with a weight at each end. The right pendulum is a double square pendulum consisting of two square plates.

Each piece is connected with a hinge. Similarly each pendulum is anchored in the virtual environment with a hinge.

Visual enhancements are green halo-like graphics surrounding the targets and lines traced by the pendulum ends.

### **8:04 - Pendulum Animation (41 sec)**

Using the game controller, the player can inject rotational energy into the pendulum using a simulated friction coupling at a hinge.

In this example, the left pendulum's top hinge is controlled. The force exerted on the hinge is visualized by an animated dashed-circle which is drawn around the hinge point.

Due to this action, the pendulum starts to swing back and forth.

As the energy in the double pendulum increases, the swings get successively higher while the motion becomes more unpredictable and chaotic.

When the pendulum touches a target - one of the moving balls - it falls down and a point is scored for the player.

### **8:49 - Game Over (19 sec)**

A double pendulum system driven to a high energy state moves with circular motion, thus guaranteeing a win.

When all targets have been touched, the game is completed and the game over screen is shown.

The screens features point and entropy statistics, a video of a real pendulum and options to repeat the game or quit to the main menu.

### **9:26 - Xbox Controller (17 sec)**

The Xbox controller - which is a standard video game interface - is used to manipulate the double pendulum.

Players will use their left thumb to move the analog stick in a swinging motion left and right to exert a force on the hinges.

The right thumb can be used on buttons A or B to select the top or bottom hinge.

### **9:48 - Pendulum Motion (15 sec)**

When moved out of their initial at-rest position through a player action, the pendulum is free to swing.

The pendulum motion is further visualized by a line which is continuously drawn from the end of the pendulum as if it were a scientific chart recorder.

### **10:06 - Player Action (18 sec)**

During play, a player will continuously use their analog stick in a periodic motion left and right.

After a short time, players will generally attempt to synchronize their analog stick motion with the visualized double pendulum on the screen to maximize the energy input to the system.

### **10:27 - Friction Actuator (13 sec)**

The friction actuator display indicates which hinge is being forced. The rotation of the circle corresponds to the amount of the action; the direction of the action is shown using red and green color coding.

### **10:44 - Game Strategy (1 min)**

A winning game strategy for players will be to swing-up the pendulum as fast as possible to reach the targets.

Initially a player must find the resonance frequency of the pendulum system in the current gravity environment and move the controller accordingly.

The value indicators can be used to determine the direction of the analog stick for energy injection.

At a later stage in the game, the motion of the pendulum becomes more chaotic. Players must now have good hand-eye coordination to avoid removing energy from the system through their hinge actions.

If a pendulum has entered a phase of chaotic motion, players may need to stabilize the pendulum through their actions before attempting to inject more energy.

### **12:02 - Info Browser Concept (42 sec)**

The “Info Browser” is a game physics element designed to create a scientific narrative during gameplay using a simple concept.

Certain events in the game trigger a contextual broadcasts on a specific topic

- for example gravity when exiting the Gravity Chooser.

Pre-created informational links contained in a database are selected based on the context and send to the network using a packet broadcast.

One or more specialized viewer applications on the network can now be used to receive such info links, construct a narrative from them and allow the user or bystanders to navigate the information during or after gameplay.

### **12:47 - Info Browser Screen (21 sec) +5sec**

The Info Browser application is a sample viewer implementation which is included with The Pendulum Game. The central viewing area functions like a web-browser and the received info links are collected and categorized on the right pane.

By navigating to some of the received links, the user creates a narrative which has its origin in the game.

### **13:13 - Info Browser Experience (16 sec)**

In the pendulum game, links on game related narrative categories such as gravity or pendulums are broadcast during gameplay.

Any running InfoBrowser application on the network will receive these links and allows other users to browse them.



# Bibliography

AIG (2010), '*American Institute of Physics: About the Journal - Journal of Mathematical Physics*'.

<http://jmp.aip.org/jmp/staff.jsp> (accessed 2010-04-01).

amazon.com (2006), '*Physics for Game Developers (Paperback)*'.

<http://www.amazon.com/exec/obidos/tg/detail/-/0596000065?v=glance> (dated 2006-04-16, accessed 2010-04-24).

(art)n (2006), '*portfolio*'.

<http://www.artn.com/portfolio> (dated 2006-12-16, accessed 2010-04-24).

ASKlabs (2000), '*Electrum*'.

<http://www.asklabs.com/electrum.html> (dated 2007-02-17, accessed 2010-04-24).

Babin, W. (1989), '*Truth and Physics - A Critique*'.

<http://www.wbabin.net/babin/critique.htm> (accessed 2010-04-04).

Boker, L. (2005), '*AtomicMPC: Get Serious*'.

<http://www.atomicmpc.com.au/article.asp?SCID=27&CIID=24825&p=2>  
(accessed 2008-01-05).

- Bourg, D. (2006), '*Homepage: David Bourg*'.  
<http://www.davidbourg.com> (accessed 2008-01-04).
- Bousiges, A. & Butler, K. (2007), '*Battlezone at arcade-history*'.  
<http://www.arcade-history.com/index.php?page=detail&id=210>  
(accessed 2008-01-10).
- Boyer, C. B. (1949), *The History of the Calculus and its Conceptual Development*, Dover unabridged 1959 edn, Hafner Publishing Company, New York, NY, USA.  
Also online: [http://books.google.com/books?id=w3xKLt\\_da2UC](http://books.google.com/books?id=w3xKLt_da2UC)  
(accessed 2010-03-30).
- Brendel, B. (1994), '*Bettina Brendel*'.  
<http://www.ylem.org/artists/bbrendel/bb.html> (dated 1994-01-01, accessed 2010-04-24).
- Brockman, J. (1991), '*The third culture*'.  
[http://www.edge.org/3rd\\_culture/](http://www.edge.org/3rd_culture/) (dated 1991-01-01, accessed 2010-12-19).
- Brockman, J. (1995), *The Third Culture: Beyond the Scientific Revolution*, Simon & Schuster, New York, NY, USA.
- Cajori, F. (1917), *A History of Physics*, The MacMillan Company, London, UK.  
Also online: <http://books.google.com/books?id=eegKAQAAIAAJ>  
(accessed 2010-03-27).
- CERN (2008), '*The safety of the LHC*'.  
<http://public.web.cern.ch/public/en/LHc/Safety-en.html>  
(accessed 2010-04-05).
- Charles River Media (2005), '*CRM – Mathematics and Physics for Programmers*'.  
<http://www.charlesriver.com/Books/BookDetail.aspx?productID=84482> (accessed 2008-01-05).

- Chen, S. (2005), '*Dame Dev Blog: Serious Games*'.  
<http://damedev.blogspot.com/2005/10/serious-games.html>  
(accessed 2008-01-06).
- Chris, C. (n.d.), '*Erasmatazz: Library*'.  
<http://www.erasmatazz.com/Library.html> (dated 2006-03-26,  
accessed 2008-01-05).
- Cohen, S. (1987), *Zap: The Rise and Fall of Atari*, McGraw-Hill,  
Columbus, OH, USA.
- Collegiate Presswire (2003), '*Liemandt Foundation Hosts Hidden Agenda  
Contest*'.  
<http://www.cpwire.com/archive/2003/11/10/1441.asp> (accessed  
2006-04-01).
- Crawford, C. (n.d.), '*Storytron Staff: Chris Crawford*'.  
[http://www.storytron.com/staff/stf\\_chris.html](http://www.storytron.com/staff/stf_chris.html) (accessed  
2008-01-05).
- Creative Capital (2006), '*Grantee Page: Jane D. Marsching*'.  
[http://channel.creative-capital.org/grantee\\_247.html](http://channel.creative-capital.org/grantee_247.html)  
(accessed 2006-12-23).
- croteam (2003), '*Croteam - Get Serious! Engine Overview*'.  
[http://www.croteam.com/engine\\_overview.shtml](http://www.croteam.com/engine_overview.shtml) (accessed  
2006-07-20).
- Crutchfield, J. P. (2008), Five questions on complexity: Responses, *in*  
C. Gershenson, ed., '*Complexity: 5 Questions*', Automatic Press,  
Copenhagen, Denmark.  
Also online:  
<http://cse.ucdavis.edu/~chaos/papers/QsOnComplexity.pdf>  
(accessed 2010-03-29).
- Dirac, P. A. M. (1928), 'The quantum theory of the electron', *Proc. R. Soc.  
Lond. A* **117**(778), 610–624.  
Also online: <http://rspa.royalsocietypublishing.org/content/117/778/610>  
(accessed 2010-12-19).

- Director Online (2007), '*Article: Danny Kodicek Profile*'.  
<http://director-online.com/buildArticle.php?id=860> (dated 2007-12-06, accessed 2010-04-24).
- Eastman, T. E. (2004), *Physics and Whitehead: quantum, process, and experience*, State University of New York Press, Albany, NY, USA, chapter 2, Duality Without Dualism.  
Also online: <http://books.google.com/books?id=fwxK6hVnOM0C> (accessed 2010-04-04).
- Einstein, A. (1916), *Relativity: The Special and General Theory*, 1920 translation edn, H. Holt and Company, New York, NY, USA.
- Electronic-Arts-Intermix (1997), '*Traces, Peter d'Agostino*'.  
<http://www.eai.org/title.htm?id=2919> (dated 2007-02-17, accessed 2010-12-26).
- Feldhay, R. (1998), The use and abuse of mathematical entities: Galileo and the Jesuits revisited, in P. Machamer, ed., '*The Cambridge Companion to Galileo*', Cambridge University Press, Cambridge, UK, chapter 3, pp. 80–133.
- Fleischmann, M. & Pons, S. (1989), 'Electrochemically induced nuclear fusion of deuterium', *Journal of Electroanalytical Chemistry* **261**(2A), 201–308.
- Fox, M. (2005), '*Megan Fox: Homepage*'.  
<http://shalinor.circustent.us/megan/resume/index.html> (accessed 2006-07-19).
- Freedman, R. A. (1996), '*Challenges in Teaching and Learning Introductory Physics*'.  
<http://www.physics.ucsb.edu/~airboy/challenge.html> (accessed 2010-04-04).
- Garage Games (2002), '*Games: Chain Reaction*'.  
<http://www.garagegames.com/pg/product/view.php?id=7> (accessed 2006-05-02).
- Garage Games (2005), '*Profile: Kevin Ryan*'.  
<http://>

[//www.garagegames.com/my/home/view.profile.php?qid=105](http://www.garagegames.com/my/home/view.profile.php?qid=105)  
(accessed 2008-01-06).

Gerstl, C. & Keijser, J. (2004), '*VIDA 7.0: Déjà Vu of fresh water - a nightmare environment*'.

<http://www.fundacion.telefonica.com/at/vida/vida10/paginas/v7/deja.html> (accessed 2008-01-09).

Godfrey-Smith, P. (2003), *Theory and Reality: An introduction to the philosophy of science*, University of Chicago Press, Chicago, IL, USA.

Goldberg, F. M. & McDermott, L. C. (1987), 'An investigation of student understanding of the real image formed by a converging lens or concave mirror', *American Journal of Physics* **55**(503).

Gorini, R. (2003), 'Al-haytham the man of experience, first steps in the science of vision', International Society for the History of Islamic Medicine, Institute of Neurosciences, Laboratory of Psychobiology and Psychopharmacology, Rome, Italy.

Gorp (2004), '*Incredible Physics Contraptions*'.

<http://trails-project.org/webapps/gorp/ViewCurriculet.do?id=21&themeId=21&state=displayAll> (accessed 2006-04-27).

Häggström, O. (2009), Interactive real time cloth simulation with adaptive level of detail, Master's thesis, Umea University, Department of Physics, SE-901 87 Umea, Sweden.

Also online:

<http://publications.dice.se/attachments/clothThesis.pdf>  
(accessed 2010-04-25).

Hague, J. (2002a), '*Halcyon Days: Interview with Chris Crawford*'.

<http://www.dadgum.com/halcyon/BOOK/CRAWFORD.HTM> (dated 2006-03-26, accessed 2006-03-26).

Hague, J. (2002b), '*Halcyon Days: Interview with Ed Rotberg*'.

<http://www.dadgum.com/halcyon/BOOK/ROTBURG.HTM> (dated 2006-03-26, accessed 2006-04-15).

Halloun, I. & Hestenes, D. (1985), 'The initial knowledge state of college physics students', *American Journal of Physics* **53**(1043).

- Hansen, B. (n.d.), '*E-Pro (Video Clip)*'.  
<http://www.youtube.com/watch?v=n0U-N-IYya8> (accessed 2006-10-05).
- Havoc Inc. (2008), '*Havoc - Homepage*'.  
<http://www.havok.com/> (accessed 2008-01-12).
- Hegedus, A. (1992), '*Handsight*'.  
[http://csw.art.pl/new/99/7e\\_agndl.html](http://csw.art.pl/new/99/7e_agndl.html) (accessed 2006-12-16).
- Hegel, G. W. F. (1805), *Jenaer Realphilosophie: Vorlesungsmanuskripte zur Philosophie der Natur und des Geistes von 1805-1806*, 1969 edn, Akademie, Berlin, Germany.
- Heim, B. (1977), 'Vorschlag eines weges einer einheitlichen beschreibung der elementarteilchen', *Zeitschrift für Naturforschung* **32a**, 233–243.
- Hesse-Honegger, C. (2001), '*Cornelia Hesse-Honegger: Biography*'.  
<http://www.mendel-museum.org/eng/lonline/popart/bio/c.pdf> (accessed 2007-02-17).
- Honigman, A. F. (2002), '*Questioning Authority: A Conversation with Eve Andrée Laramée*'.  
<http://www.sculpture.org/documents/scmag02/julaug02/laramee/lar.shtml> (accessed 2006-12-18).
- Huygens, C. (1678), '*Traité de la lumière*', published in 1690.
- Intuitor (2006), '*Insultingly Stupid Movie Physics*'.  
<http://www.intuitor.com/moviephysics/index.html> (accessed 2008-01-09).
- Jacobs, J. (2004), '*Blog: Stealth education*'.  
<http://joannejacobs.com/2004/09/14/stealth-education/> (dated 2006-04-02, accessed 2008-01-06).
- James E. Cote, A. L. A. (2007), *Ivory Tower Blues: A University System in Crisis*, University of Toronto Press, Toronto, Canada.
- Kant, I. (1781), '*The Critique of Pure Reason*'.  
<http://www.gutenberg.org/ebooks/4280> (dated 2003-07-01, accessed 2011-03-13).

- Kaye, Z. & Mr. Snow (2001), '*Firmament*'.  
<http://laudanum.net/firmament/> (accessed 2006-12-18).
- Kemp, T. (2004), '*tom kemp: particle*'.  
<http://www.tomkemp.com/particle/index.html> (accessed 2007-02-03).
- Kendall, D. S. (2007), '*Power to Learn - Crazy Machines 1.5*'.  
[http://www.powertolearn.com/articles/software\\_reviews\\_for\\_parents/article.shtml?ID=16](http://www.powertolearn.com/articles/software_reviews_for_parents/article.shtml?ID=16) (dated 2007-06-06, accessed 2010-04-25).
- Kline, M. (1959), *Mathematics and the Physical World*, Dover unabridged 1981 edn, Thomas Y. Crowell Company, New York, NY, USA.
- Kuhn, T. S. (1962), *The Structure of Scientific Revolutions*, 3rd edn, University of Chicago Press, Chicago, IL, USA.
- Laramee, E. A. (2006), '*Eve Andree Laramee*'.  
[http://www.home.earthlink.net/~wander/\\_wsn/page3.html](http://www.home.earthlink.net/~wander/_wsn/page3.html) (accessed 2006-12-18).
- Liemandt Foundation (2006), '*Hidden Agenda Website*'.  
<http://www.hiddenagenda.com> (accessed 2006-03-28).
- MachineDesign (2006), '*NASCAR simulators keep it real*'.  
<http://www.machinedesign.com/ASP/viewSelectedArticle.asp?strArticleId=59900&strSite=MDSite> (accessed 2006-04-24).
- McDermott, L. C. & Shaffer, P. S. (1992), 'Research as a guide for curriculum development: an example from introductory electricity, part i: Investigation of student understanding', *American Journal of Physics* **60**(994).
- Mill, J. S. (1851), *A System of Logic*, Vol. 1, 3rd edn, John W. Parker, West Strand, London, UK.  
Also online: <http://books.google.com/books?id=eegKAQAAIAAJ> (accessed 2010-03-27).
- Minsky, M., Ming, O., Steele, O., Brooks, F. P. & Behensky, M. (1990), 'Feeling and seeing: issues in force display', *Proceedings of the 1990 Symposium on interactive 3D Graphics*.

- Newbert, T. N. (2009), *A Critique of Pure Physics: Concerning the Metaphors of New Physics*, Xlibris Corporation, Bloomington, IN, USA.  
Also online: <http://books.google.com/books?id=sf-M-b8Jct8C>  
(accessed 2010-04-04).
- Newton, I. (1687), 'Philosophiae naturalis principia mathematica', London, England.
- Newton, R. G. (2007), *From Clockwork to Crapshoot: A History of Physics*, The Belknap Press of Harvard University Press, Cambridge, MA, USA and London, UK.
- Noe, G. (2007), 'Half-Life 2 - The First Hour'.  
2007-08-08 (dated 2007-08-08, accessed 2010-04-25).
- O'Reilly Press (2001), 'Press Release: Physics for Game Developers'.  
<http://press.oreilly.com/pub/pr/880> (accessed 2006-04-16).
- Orr, E. (1998), 'Lightning on Demand: The Electrum Project'.  
<http://www.lod.org/Projects/electrum/index.htm> (accessed 2007-02-17).
- Paal, G. (2009), 'Fachübergreifende Kooperation'.  
[http://content.stuttgarter-zeitung.de/stz/page/2247571\\_0\\_2147\\_-pro-funktioniert-fachuebergreifende-kooperation.html](http://content.stuttgarter-zeitung.de/stz/page/2247571_0_2147_-pro-funktioniert-fachuebergreifende-kooperation.html)  
(dated 2009-10-22, accessed 2011-03-13).
- Pepper Games (2005), 'Crazy-Machines: Download page'.  
[http://www.crazy-machines.com/CMNADL\\_ENG/CMNADL\\_EN.htm](http://www.crazy-machines.com/CMNADL_ENG/CMNADL_EN.htm)  
(accessed 2006-07-19).
- Polson Enterprises (1996), 'Notes about Force Feedback joystick developments at Atari'.  
<http://www.virtualpet.com/vp/media/fishing/fishing.htm#force>  
(accessed 2006-04-16).
- Powell's Books (2005), 'Mathematics and Physics for Pogrammers'.  
<http://>



[//www.powells.com/cgi-bin/biblio?inkey=2-1584503300-1](http://www.powells.com/cgi-bin/biblio?inkey=2-1584503300-1)  
(accessed 2008-01-05).

Prager, B. (2009), '*Heim Theory: Homepage*'.

<http://www.heim-theory.org> (dated 2009-11-30, accessed 2010-12-19).

Pritchard, D. E. (2010), '*MIT Physics 8.01 - Classical Mechanics - Section L02 - Course Notes 2*'.

<http://web.mit.edu/8.01t/www/coursedocs/current/guide.htm?SEC=L02> (dated 2010-03-30, accessed 2010-03-30).

Rabounski, D. (2005), '*Progress in Physics - Declaration of Academic Freedom*'.

[http://www.ptep-online.com/index\\_files/rights.html](http://www.ptep-online.com/index_files/rights.html)  
(accessed 2010-04-04).

Rotberg, E. (1982), '*Zounds!*'.

<http://www.atarimagazines.com/v1n4/zounds.html> (dated 2006-04-18, accessed 2008-01-06).

Russell, B. (1931), *The Scientific Outlook*, 1st edn, George Allen and Unwin Ltd., London, UK.

Also online: <http://books.google.com/books?id=cV7Zo8z4tDwC>  
(accessed 2010-03-27).

Sambursky, S. (1974), *Physical Thought from the Presocratics to the Quantum Physicists*, Vol. 1, Ithaca, Ithaca, NY, USA.

Schiffler, A. (1997), SuperDARN Measurements of Double-Peaked Velocity Spectra, PhD thesis, University of Saskatchewan, Saskatoon, Canada.

Also online:

<http://www.ferzkopp.net/joomla/content/view/26/15/> (accessed 2010-10-15).

Schiffler, A. (1999), '*FAST - Font Asteroids*'.

<http://www.ferzkopp.net/joomla/content/view/69/12/> (dated 2006-09-18, accessed 2010-04-24).

Scientific American (2003), '*The Future of String Theory - A Conversation with Brian Greene*'.

- <http://www.sciam.com/media/pdf/Greene.pdf> (accessed 2010-04-03).
- Sheridan, S. L. (2003), '*Sonia Landy Sheridan: Generative Systems*'.  
<http://www.sonart.org/book/chapter01/chapter.htm> (accessed 2010-09-17).
- Shipper, E. (2006), '*CARSTEN HÖLLER: A Selection of Works and Exhibitions 2001 - 2006*'.  
<http://www.esterschipper.com/tate/hoeller.pdf> (accessed 2007-01-18).
- SIGGRAPH (1999), '*Jay Lee and Bill Keays: Suspended Window*'.  
[http://www.siggraph.org/artdesign/gallery/S99/artists/Lee\\_Jay.html](http://www.siggraph.org/artdesign/gallery/S99/artists/Lee_Jay.html) (dated 2000-05-08, accessed 2011-04-03).
- Snow, C. P. (1959), *The Two Cultures and the Scientific Revolution*, 7th, 1961 edn, Cambridge University Press, New York, NY, USA.  
Also online: [http://sciencepolicy.colorado.edu/students/envs\\_5110/snow\\_1959.pdf](http://sciencepolicy.colorado.edu/students/envs_5110/snow_1959.pdf) (accessed 2010-04-04).
- Stewart, B. (2001), '*An Interview with David Bourg: Author of Physics for Game Developers*'.  
<http://www.oreillynet.com/pub/a/network/2001/11/20/bourg.html> (accessed 2006-04-16).
- Tajmar, M., Plesescu, F., Marhold, K. & de Matos, C. J. (2006), '*Experimental Detection of the Gravitomagnetic London Moment*'.  
<http://arxiv.org/abs/gr-qc/0603033> (dated 2006-03-09, accessed 2010-12-19).
- Taylor, J. C. (2001), *Hidden Unity in Nature's Laws*, Cambridge University Press, Cambridge, UK.
- Taylor, J. R. (1997), *An Introduction to Error Analysis*, 2nd edn, University Science Books, Sausalito, CA, USA.
- TED (1998), '*Talks: Brenda Laurel on games for girls*'.  
[http://www.ted.com/talks/brenda\\_laurel\\_on\\_making\\_games\\_for\\_girls.html](http://www.ted.com/talks/brenda_laurel_on_making_games_for_girls.html) (accessed 2010-11-01).

- The Atari Library (2006), '*Atari Battlezone*'.  
<http://markn.users.netlink.co.uk/Arcade/battz.html> (dated 2006-08-03, accessed 2010-04-24).
- The Atari Library (2010), '*Atari Asteroids*'.  
<http://markn.users.netlink.co.uk/Arcade/aster.html>  
(accessed 2010-12-24).
- The Columbia Free-Net (2007), '*James L. Acord: B - Reactor Museum Association Commemorative Sculpture Project*'.  
[http://www.tcfm.org/timecapsule/html/james\\_l.\\_acord.html](http://www.tcfm.org/timecapsule/html/james_l._acord.html)  
(accessed 2007-02-16).
- The International Arcade Museum (1995), '*Killer List of Video Games: Hard Drivin*'.  
[http://www.klov.com/game\\_detail.php?game\\_id=8072](http://www.klov.com/game_detail.php?game_id=8072) (accessed 2006-04-13).
- Transmitter-X (2005), '*Gudrun Bielz*'.  
<http://matrix.kunst.transmitter-x.org/index.php?www2/katalog.php?kid=60&s=&selwert=~rFrame> (accessed 2007-02-17).
- Twin Lanes Technologies (2000), '*Twin Lanes Technologies Homepage*'.  
<http://twinlanes.com> (accessed 2006-04-24).
- van Gaal, R. & Dolphinity B.V. (2007), '*Racer Car and Racing Simulator*'.  
<http://www.racer.nl/> (accessed 2008-01-12).
- Venn, J. (1880), 'On the diagrammatic and mechanical representation of propositions and reasonings', *Dublin Philosophical Magazine and Journal of Science* **9**(58).
- von Borries, F., Walz, S. P. & Boettger, M. (2007), *Space Time Play: Computer Games, Architecture and Urbanism: The Next Level*, Birkhaeuser, Basel, Switzerland.
- Wegner, M. (2006), '*Homepage: Fun-Motion Physics Games*'.  
<http://www.fun-motion.com> (accessed 2006-05-26).
- Wikipedia (2006a), '*Article: Battlezone (1980 video game)*'.  
<http://>

[//en.wikipedia.org/wiki/Battlezone\\_%281980\\_video\\_game%29](http://en.wikipedia.org/wiki/Battlezone_%281980_video_game%29)  
(dated 2006-04-15, accessed 2010-04-24).

Wikipedia (2006b), '*Article: The Incredible Machine*'.

[http://en.wikipedia.org/wiki/The\\_Incredible\\_Machine](http://en.wikipedia.org/wiki/The_Incredible_Machine)  
(accessed 2008-01-06).

Wikipedia (2008a), '*Article: Arkanoid*'.

<http://en.wikipedia.org/wiki/Arkanoid> (dated 2008-01-11,  
accessed 2010-04-24).

Wikipedia (2008b), '*Article: Asteroids (arcade game)*'.

[http://en.wikipedia.org/wiki/Asteroids\\_%28arcade\\_game%29](http://en.wikipedia.org/wiki/Asteroids_%28arcade_game%29)  
(dated 2008-01-10, accessed 2010-04-24).

Wikipedia (2008c), '*Article: Grand Prix Legends*'.

[http://en.wikipedia.org/wiki/Grand\\_Prix\\_Legends](http://en.wikipedia.org/wiki/Grand_Prix_Legends) (dated  
2008-01-25, accessed 2010-04-24).

Wikipedia (2010a), '*Article: Battlefield 2142*'.

[http://en.wikipedia.org/wiki/Battlefield\\_2142](http://en.wikipedia.org/wiki/Battlefield_2142) (dated  
2010-04-14, accessed 2010-04-24).

Wikipedia (2010b), '*Article: Half-Life 2*'.

[http://en.wikipedia.org/wiki/Half\\_life\\_2](http://en.wikipedia.org/wiki/Half_life_2) (dated 2010-04-22,  
accessed 2010-04-24).

Wikipedia (2010c), '*Article: Mario Bros.*'.

[http://en.wikipedia.org/wiki/Mario\\_Bros](http://en.wikipedia.org/wiki/Mario_Bros) (dated 2010-04-20,  
accessed 2010-04-24).

Wikipedia (2010d), '*Article: Space Invaders*'.

[http://en.wikipedia.org/wiki/Space\\_invaders](http://en.wikipedia.org/wiki/Space_invaders) (dated  
2010-04-24, accessed 2010-04-24).

Wikipedia (2010e), '*Article: Super Mario Bros.*'.

[http://en.wikipedia.org/wiki/Super\\_Mario\\_Bros](http://en.wikipedia.org/wiki/Super_Mario_Bros). (accessed  
2010-03-28).

Wikipedia (2010f), '*Article: Theory of Everything*'.

[http://en.wikipedia.org/wiki/Theory\\_of\\_Everything](http://en.wikipedia.org/wiki/Theory_of_Everything) (dated  
2010-03-18, accessed 2010-03-29).

Ye, C., Li, J. & Luo, J. (2009), 'Precision measurement physics and its methodology', *Frontiers of Physics in China* **4**(2), 143.

Also online: <http://www.springerlink.com/content/p3263681761w3873/fulltext.pdf> (accessed 2010-04-01).

YLEM (2000), 'Artists using science and technology', *YLEM Newsletter* **20**(10).

Also online: <http://www.ylem.org/Journal/2000SepOct-clr.pdf> (accessed 2000-09-01).

# Glossary

**2D** two-dimensional. 22

**3D** three-dimensional. 19, 23, 31

**avatar** is a term that describes the computer user's representation as a 3D model, 2D image or textual username and may appear in a wide range of applications such as virtual worlds, computer games or social networks. 19, 22, 27

**calculus** is a branch of mathematics which is concerned with the study of how functions change when their inputs change and covers differentiation as well as integration methods. 12, 13

**CPU** Central Processing Unit. 25

**CRT** Cathode Ray Tube. 23

**dichotomy** is any splitting of a whole into two jointly exhaustive but mutually exclusive parts. 16

**error analysis** is the study of the type and quantity of errors that occur as the output of a model changes when the parameters to the model vary about a mean. 14

**FPS** First-Person Shooter. 31, 34

**GUT** Grand Unified Theory. 12

**IT** Information Technology. 8

**LAN** Local Area Network. 31

**LHC** Large Hadron Collider. 15

**meme** is a term derived from Greek *mneme*, or “memory,” describing a theoretical unit of cultural information, a building block of culture or cultural evolution which spreads through diffusion propagating from one mind to another; popularized by R. Dawkins *The Selfish Gene* (1976) as well as studied in the controversial field of *memetics*. 30

**moniker** is a nickname or pseudonym used in a small subculture. 12

**NES** Nintendo Entertainment System. 27

**physical evidence** is any evidence that proves a fact based on the demonstrable physical characteristics of a material object. 10

**platformer** (or platform game) is a computer game genre that originated in the early 1980s in which jumping on platforms via a jump button is the core part of the gameplay. 20, 27

**proportionality** is the relation of two quantities if they exhibit a constant ratio; the mathematical symbol  $\propto$  is used to indicate that two values are proportional. 12

**TOE** Theory of Everything. 12

**VR** Virtual Reality. 23, 24