

# Physics Education in Virtual Reality: An Example

Hannes Kaufmann and Bernd Meyer  
Kaufmann@ims.tuwien.ac.at, xberndmx@hotmail.com  
Institute of Software Technology and Interactive Systems,  
Vienna University of Technology, Vienna, Austria

## Abstract

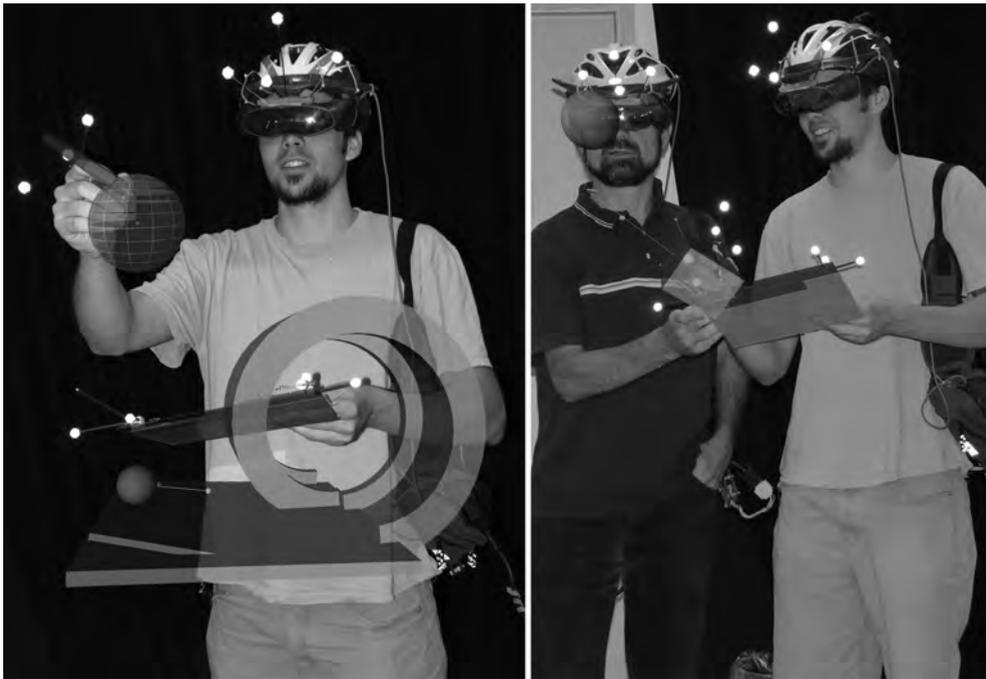
We present an immersive virtual reality (VR) application for physics education. It utilizes a recent physics engine developed for the PC gaming market to simulate physical experiments correctly and accurately. Students are enabled to actively build their own experiments and study them. A variety of tools are provided to analyze forces, mass, paths and other properties of objects before, during and after experiments. Innovative teaching content is presented that exploits the strengths of the 3D virtual environment. Physics Playground serves as an example of how current technologies can be combined to deliver a new quality in physics education.

## Motivation

Classical mechanics (Goldstein, Poole, & Safko, 2006; Pohl, 1931) is the oldest discipline in the field of physics. It describes the common motion of objects that humans perceive in everyday life. The fundamental laws of dynamics which were formulated by Issac Newton (1642 – 1727) are still of high importance and traditionally taught in schools. Even though our knowledge about physics changed throughout centuries, the ideas behind force, acceleration, speed and path are still up to date. Every mechanical machine and many physical domains relate to these laws. Due to the fundamental character and palpability of mechanics, a large number of physics books start with a chapter on mechanics (Gerthsen & Vogel, 1993; Körner, Hausmann, Kießling, Mende, & Spretke, 1988; Pohl, 1931).

Some students have problems understanding the physical concepts of mechanics. It might be that physics in the traditional sense is sometimes taught in an abstract, jejune way, and is, therefore, not very motivating to students. The result is that the theoretical models behind physical phenomena are often misunderstood. It is not necessary to stress that conveying these concepts correctly is of utmost importance since they are fundamental to physics. Many other theoretical models build upon the laws of dynamics.

Therefore, the authors developed an educational virtual reality (VR) application called PhysicsPlayground (Meyer, 2007) that is supposed to support students in studying and finally understanding the concepts of mechanics (Figure 1). To do so, PhysicsPlayground provides users with a believable three-dimensional virtual world with correct physical behavior. In this environment, learners and educators are able to freely create physical experiments that can be simulated in real time. In addition, analyzing capabilities of physical properties are provided in the virtual world. We consider the analyzing functionality an important strength of a virtual laboratory like PhysicsPlayground. It offers possibilities that are far superior to what can be done in a real physics lab. A direct connection between simulated reality and physical data is supposed to help students grasp the theoretical basics of mechanics. To establish a direct link to students' pre-knowledge, physical data - that can be acquired through the application - is presented in a way so that it closely relates to formulas and equations of school mechanics.



**Figure 1.** Left: A student working with PhysicsPlayground. In the right hand he holds a wireless pen (used as input device), in the left hand the PIP. Right: Collaborative work in PhysicsPlayground.

## Working environment

The implementation of PhysicsPlayground is based on the Studierstube Augmented Reality framework (Schmalstieg et al., 2002). The standard hardware setup consists of an head-mounted display (HMD), a wireless pen and the so called personal interaction panel (PIP) (Szalavári & Gervautz, 1997). Pen and PIP are used to fully control the application in 3D space (Figure 2). The overall setup is very much like the one used in Construct3D (Kaufmann, 2004), since both applications were developed in the same lab. It allows direct manipulation and free roaming around virtual objects, and is favored by students. Each of the hardware components is tracked by an optical tracking system in six degrees of freedom. This gives the user freedom in motion – some students lie down on the floor to look at objects from below or step on a chair – and simultaneously enhances the feeling of immersion and fun. PhysicsPlayground runs on a standard desktop setup as well, but is more intuitive and more impressive to use in an immersive environment.

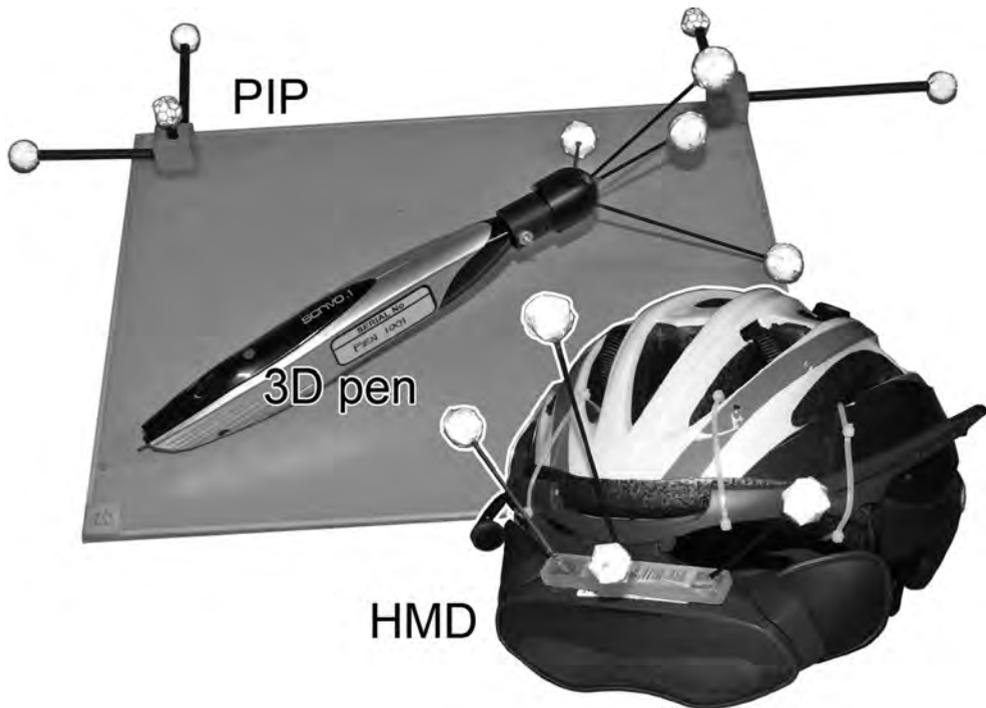


Figure 2. Each user is equipped with input devices, pen and PIP, and a head-mounted display when interacting with PhysicsPlayground.

## Interaction

The pen has one clickable button, which is used to select and drag objects and to control the application. Most of the application functionality can be invoked through the PIP, a touchable Plexiglas sheet providing haptic feedback. When looking through the HMD, control elements are displayed on the PIP which the user can click in order to trigger different actions (Figure 3). On purpose, it is similar to a 2D GUI to provide students with a known type of interface.

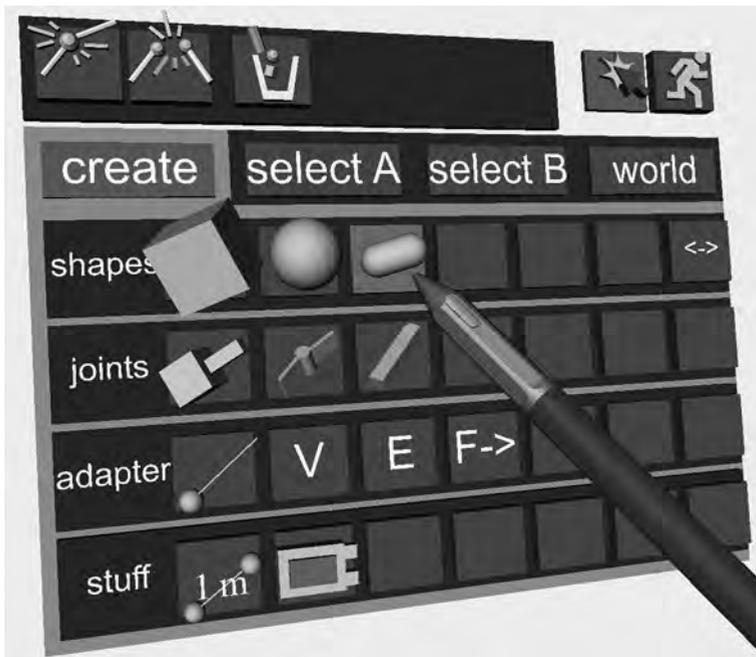


Figure 3. An example menu of PhysicsPlayground on the PIP. The virtual representation of the pen is shown as well.

In Figure 3, several menu buttons are shown. Many of them have 3D symbols placed on top. When moving the pen over these symbols they play an animation. This clarifies their specific functionality, in short, and is self-explanatory. The GUI is a stand-alone component which can flexibly be used by other VR applications. An additional layout manager system like it that is standard for 2D GUIs is not available at present.

## Features and design

In (Seugling & Rölin, 2006), a number of widely used physics engines have been evaluated in detail. Amongst them was the commercial engine PhysX, formerly known as NovodeX, from AGEIA (AGEIA, 2006) which was recently acquired by

Nvidia, and the open source engines Newton Game Dynamics and Open Dynamics Engine. The result of the evaluation was that PhysX is the most stable, feature rich, precise and fastest engine available at the moment. Therefore, the PhysX API was chosen as the base of PhysicsPlayground. PhysX from AGEIA is a commercial physics engine written in C++ that is licensed to developers of non-commercial software free of charge. Objects which can be used by PhysX in simulations are rigid bodies, soft bodies, deformable objects, fluids and cloth. Related to PhysicsPlayground, rigid bodies are of highest importance. A rigid body in its original definition is an object with fixed geometrical characteristics (Hecker, 1996). At each point in time every point within the rigid body stays fixed to each other. Additional dynamics provide the rigid body with the ability of translational and rotational motion. In PhysX, rigid bodies and rigid body dynamics are referred to as shapes and actors. In the following, the features of PhysicsPlayground will be described briefly.

### ***3D shapes***

Because the application is intended to simulate physical school experiments, it must be possible to integrate virtual models of real life objects. Therefore, PhysicsPlayground allows users to create, destroy, modify and interact with different kinds of shapes. Each shape can be either static or dynamic, and represents a solid object, enabled for collision detection during simulation. Static shapes stay in place while the physical simulation is running. They have an infinite mass. Dynamic shapes behave like real world objects. They have an adjustable mass, a center of mass and a surface friction, and are affected by force during simulation.

The appearance of the shapes can have various forms. Shapes can be simple objects like boxes, spheres, or cylinders, or more complex like a looping (Figure 1 left) or a car. A number of primitive shapes are integrated into PhysicsPlayground by default. More complex objects can be defined by the user or loaded on demand. Position, appearance and parameters of all shapes, for example, the width of a box, can be configured by the user after object creation. Modification takes place through the PIP or by direct manipulation of the shapes with the pen. Finally, shapes can be grouped into larger shapes making it possible to create advanced objects.

### ***Joints***

Rather than simply grouping shapes, our application allows the definition of mechanical linkages between different shapes (Figure 4). Currently implemented are a revolute joint, a prismatic joint and a stiff joint connection. In the case of the revolute joint, a motor can be added. It causes its attached shape to spin with constant radial velocity. With such connections at hand it is possible to create more complex physical scenes. A centrifugal regulator, for example, can be built and simulated in Phys-

icsPlayground this way (e.g. Figure 9 right). Modification and positioning of each joint is either done utilizing the PIP or directly by selecting a joint in 3D space.

## **Analyzing**

During every physical experiment, certain magnitudes and properties of the simulated objects might be of interest. To be able to compare a virtual simulation with a real one, it is important to offer possibilities to extract physical properties of objects before, during and after run time of the simulation. For educational purposes, these data can be used to learn about the theoretical background or to confirm results in a traditional way, for instance, by using appropriate formulas. Physical magnitudes with a high interest of analyzability are speed, acceleration, force, friction, energy and path.

Two mechanisms are provided to output such data:

- a) A simple measurement tool to measure distances and
- b) A powerful analyzer which can extract and visualize all interesting, relevant physical magnitudes from shapes (Figure 4 right).

As archetype for the analyzer we used an oscilloscope. An oscilloscope has two axes, a time axis and a voltage axis. Both axes are equivalently used in our application. On the value axis we can visualize – in addition to absolute values, like voltage - also components of 3D vectors. This is needed because magnitudes like velocity or path have three dimensions plus a value. Therefore, our analyzer has four input fields - x, y, z and an absolute value.

The input fields can be connected to different shape attributes, like speed or kinetic energy. The attributes are represented through so called adapters. Velocity, energy and a path adapter are supported. Adapters have to be added to each shape manually. An added adapter interfaces a specific physical magnitude, and, via the adapter, data can be plugged into the analyzer. During simulation, the adapter will continually send its value to the analyzer and the accordant time-value function is drawn in real time. This allows students to study all data during the running experiment.

Moreover, these adapters do not only act as interfaces to the analyzer but can visualize their values. For example, the path-adapter (which is attached to an object) will record and subsequently draw its trajectory, whereas the velocity-adapter will split and visualize the absolute velocity value in its x, y and z direction.

## **Force adapter**

In addition to adapters with analyzing capabilities, we added a force adapter. With the force adapter, a directed and dynamically changeable force can be put on a shape to affect its motion. This gives many new possibilities on physical scenarios. For instance imagine a simulation of precisely accelerating and decelerating a car.

Configuration of the force adapter is done through the PIP. When selecting the adapter, a force-time function can be defined using a built-in key frame editor. The function is finally processed during simulation. This functionality is demonstrated in the next section in the example “Teaching Lesson: Speed and Velocity”.

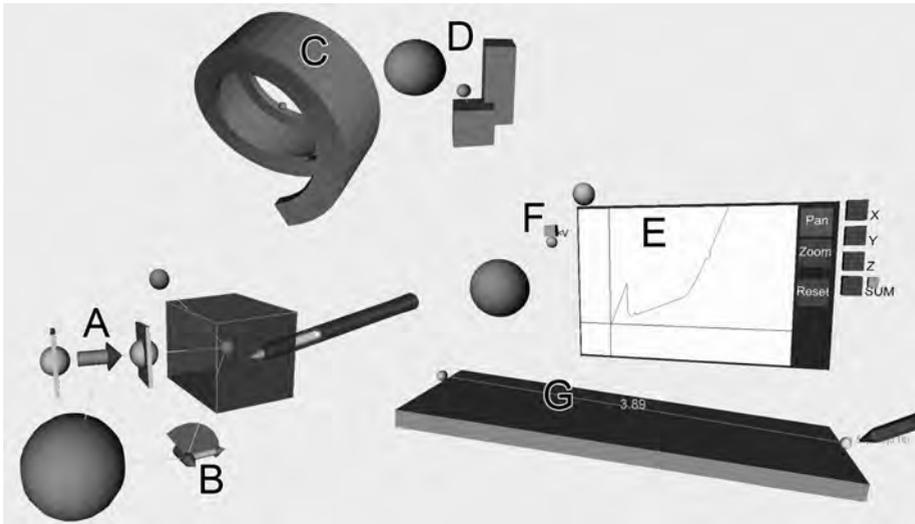


Figure 4. The PhysicsPlayground elements and their visualisations in VR.

In Figure 4, the above – mentioned building blocks of PhysicsPlayground are shown. A and B represent mechanical linkages. Both connect the same shapes, box and sphere. The revolute joint B restricts the rotational motion between both bodies along the axis of rotation (darker arrow pointing outwards). The prismatic joint A restricts the translational motion between both bodies along the vector drawn below A. C is a more complex shape, a loop-the-loop. D shows grouped shapes which act together during simulation. E denotes the analyzer which is connected to the velocity adapter F of a sphere. During simulation the velocity value of F is recorded. The orange curve on the analyzer results from an earlier run of the experiment where the sphere bounced onto the underlying plane and continued to roll down. The recorded time-velocity values can be read exactly by just moving the pen over the orange line. G demonstrates the measurement tool. Currently, the length of a box is measured.

## Educational use cases

### *Force and counterforce*

In the following, the applicability of PhysicsPlayground for school experiments and school exercises will be demonstrated by means of force and counterforce. The effect

of force and counterforce is described through the third axiom of dynamics. It states that every force which is invoked by a body A on a body B leads to a counterforce in the opposite direction with the same absolute value.

An experiment that we use to demonstrate force and counterforce is described in Pohl (1931). Thereby, two persons stand face to face, each on their own frictionless moving carriage (Figure 5). At the same time, both persons hold a rope which connects them. With this setup, a series of tests can be conducted:

1. Both participants pull the rope at the same time.
2. Only the left person pulls the rope, whereas the right person holds the rope.
3. The same as before, but now only the right person pulls on the rope.

Independent of which participant pulls the rope, both wagons will move the same distance from start to the point where both collide in the middle. This is a result of force and counterforce.

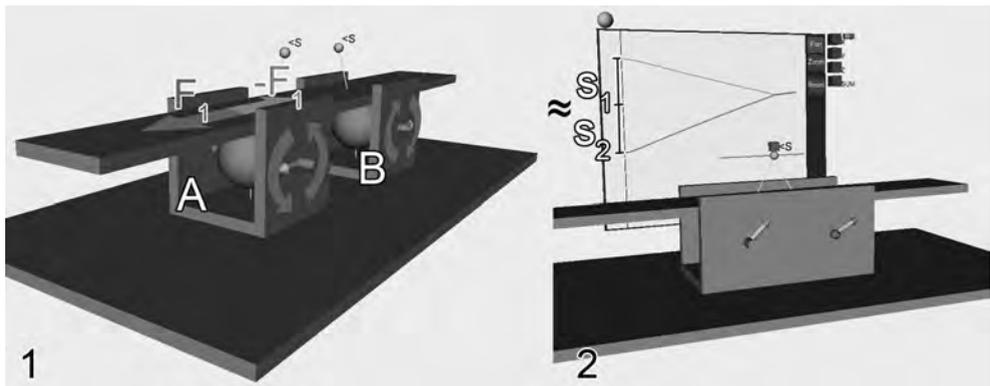


Figure 5. Demonstration of force and counterforce.

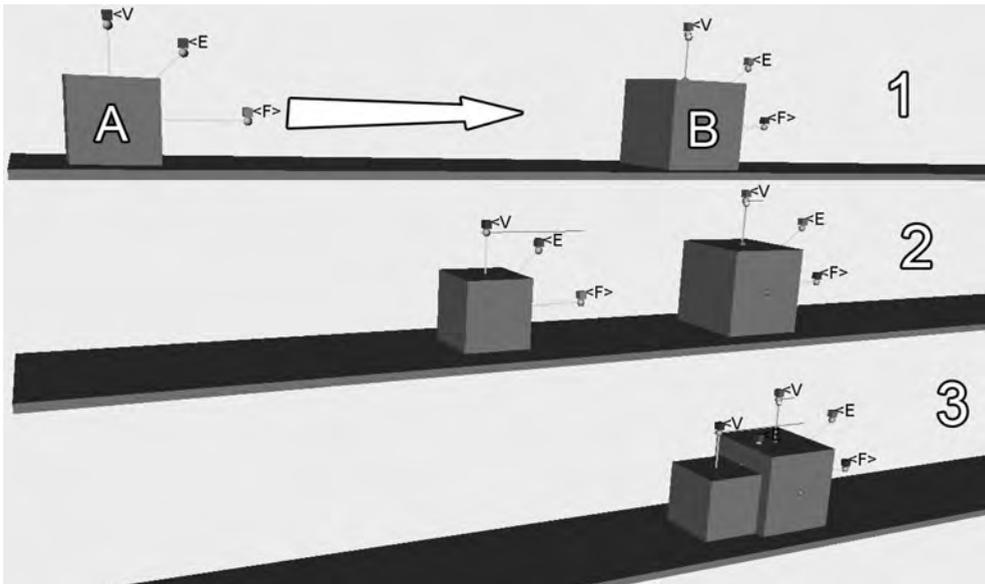
In our simulation of the experiment, the wagons (where the persons stand on) are replaced by two identical, frictionless slidable box frames A and B. Each of the boxes has a rotatable sphere in the middle which is mounted on the box frame via two revolute joints. These spheres represent the two persons. The rope is then realized through a bar which lies on top of both spheres. Additionally, bar and sphere must have a maximized friction coefficient of 1 so that everything works out right. Next, we want the participants to pull the rope during simulation. Therefore, a motor has to be attached to one revolute joint of each box frame. On body A, this motor has to spin counter-clockwise, whereas on body B, it has to turn clockwise. When one of the motors is activated during simulation, it will put a predefined torque along the rotational axis of its attached sphere. This will put the sphere into a rotational motion. The sphere again will try to transfer this motion to the bar at the point where the bar

touches the sphere. This leads to a force  $F_1$  which affects the sphere tangentially opposed to its rotational direction. In the other direction, it invokes a counterforce  $-F_1$  on the bar which has the same absolute value as force  $F_1$ . No matter if one or both motors are activated, the person who watches the simulation will notice that both box frames will always get together in the middle (the mid position in between their starting positions). Additionally, this can be proven in PhysicsPlayground by analyzing the path of each box frame through a trajectory adapter. During simulation, the analyzer will record a curve similar to the one shown in Figure 5 right. Students can read the distances  $s_1$  and  $s_2$  from the analyzer. Regardless of which person pulls the rope, both carriages always move equivalently.

### ***Teaching lesson: Speed and velocity***

This lesson demonstrates how PhysicsPlayground can be integrated into a traditional physics lesson. An appropriate physical exercise from (Stark, 2002) was chosen. It was part of an actual high school final physics exam.

In this exercise, two bodies A and B slide on top of a plane along the global x-axis with different speeds. At some point in time, both bodies might collide. Many physical scenarios could be played through with this setup. We will recreate the scenario which relates most to the school task (Stark, 2002) and extend it in the following. The starting setup in PhysicsPlayground contains a plane and the two bodies A and B as shown in step 1 of Figure 6.



**Figure 6.** Complete setup for a PhysicsPlayground based school exercise. Steps one to three visualize the simulation progress in PhysicsPlayground.

The task description for students is the following:

A toy car which is represented by body *A* is accelerated along the positive *x*-axis. Starting at its standing position, it is accelerated by a time dependent force  $F_A(t)$ , which is given in Figure 7. The friction of all shapes is zero. Use PhysicsPlayground to acquire the final speed  $v_{end}$  of the toy car *A* and check your result by calculating it manually.

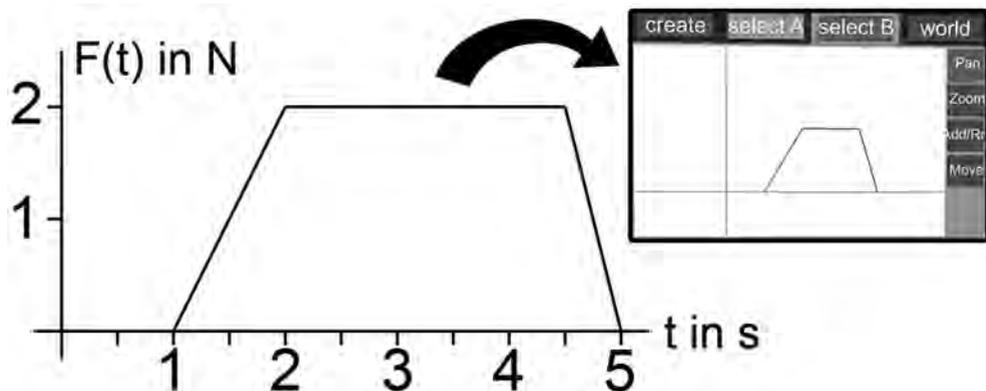
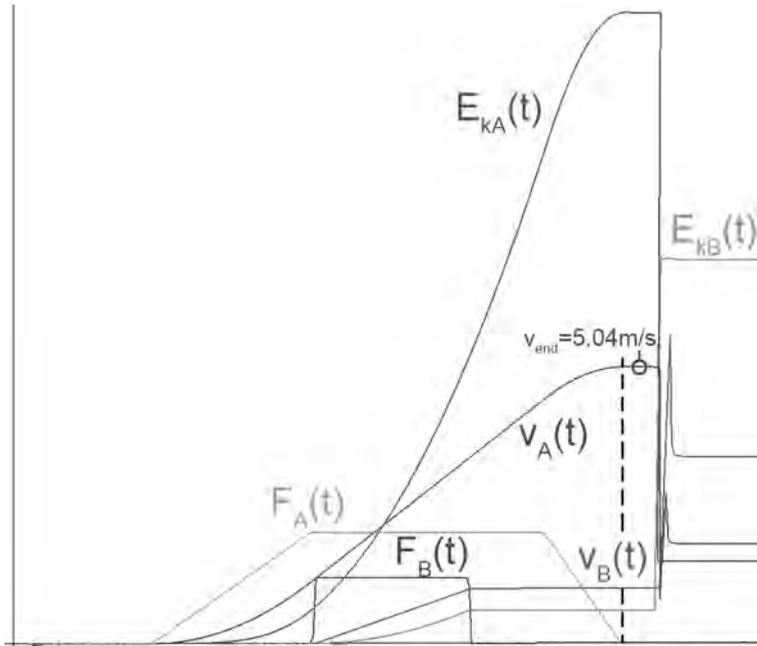


Figure 7. The given time force-function (left) and force adapter conversation (right).

In order to solve this task with the help of PhysicsPlayground, the student has to transfer the time-force function to PhysicsPlayground first. Basically, this is done by adding a force adapter, which points towards the positive *x*-axis to the toy car. The adapter is displayed in Figure 6, attached to object *A* (the dot denoted <F>). In the next step, the key-frames of the adapter have to be adjusted so that they represent the time-force function, as can be seen in Figure 7. If everything was done correctly, the body should start to move along the *x*-axis after the simulation was started. In order to check the movement, students may create an analyzer and connect the force adapter to the sum-input of the analyzer. If the recording of the analyzer is equal to the time-force function in Figure 7, the previous steps were performed correctly.

The next step is to investigate the velocity of the toy car. This is quite straightforward as we only have to attach a velocity adapter to body *A*. After doing so, the velocity adapter has to be connected with the *x*-input of the analyzer so that students can read off the final speed of the toy car from the analyzer. The print out of the analyzer is finally shown in Figure 8.



**Figure 8. Analyzer output for bodies A and B.  
The single curves were labelled afterwards.**

The only relevant graphs are  $F_A(t)$  and  $v_A(t)$ . There are some additional printouts of body A and B just to demonstrate what can be done with the analyzer (they are not relevant for this task). The graphs labelled E stand for kinetic energy. The graph of  $F_A(t)$  represents the time force function which is applied to the toy car during simulation. It is obvious that its analyzer graph is much the same as in Figure 7. This is exactly how it should be.

The velocity graph  $v_A(t)$  of the velocity adapter derives from  $F_A(t)$ . Final velocity is reached when the force  $F_A$  stops to affect body A. From this point on, the dotted line perpendicular to the abscissa (where  $F_A(t)$  stops affecting A) intersects  $v_A(t)$  at a point from which on  $v_{end}$  is reached. According to the analyzer,  $v_{end} = 5.04 \text{ m/s}$ . Since the solution, namely  $v_{end}$ , has been found using PhysicsPlayground, students can try to calculate it next.

In the solution of (Stark, 2002), the area below the force function  $F_a = \sum F * \Delta t$  is calculated first. To calculate  $v_{end}$ , the area  $F_a$  is used as a parameter in a formula for constant accelerated movement  $v_{end} = F_a / m$  whereby  $m$  is the mass of body A. After determining its mass within PhysicsPlayground, we calculate the result  $v_{end}$  and get a final result of  $5.00 \text{ m/s}$ . This is very close to the result of the simulation.

Apart from the above – mentioned exercises and experiments, PhysicsPlayground can be used to demonstrate a wide variety of mechanical devices, as shown in Figure 9.

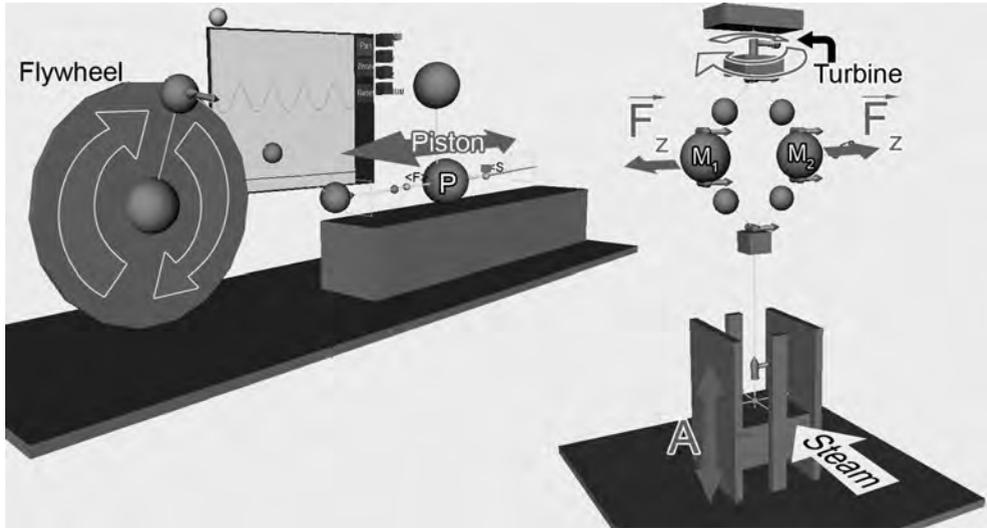


Figure 9. Left: A simple scenario that simulates the motion of a car crankshaft. The analyzer in the background is connected to the piston P in order to study the motion of the piston. Right: A centrifugal regulator. If the turbine spins faster, body A will rise and the steam supply will be decreased.

### *Motion paths*

Last but not least, the application can also be used to visualize mathematical and geometrical contents such as kinematics (Figure 10). Through an appropriate linkage and rotational forces between three shapes A, B and C, we can cause shape B to rotate

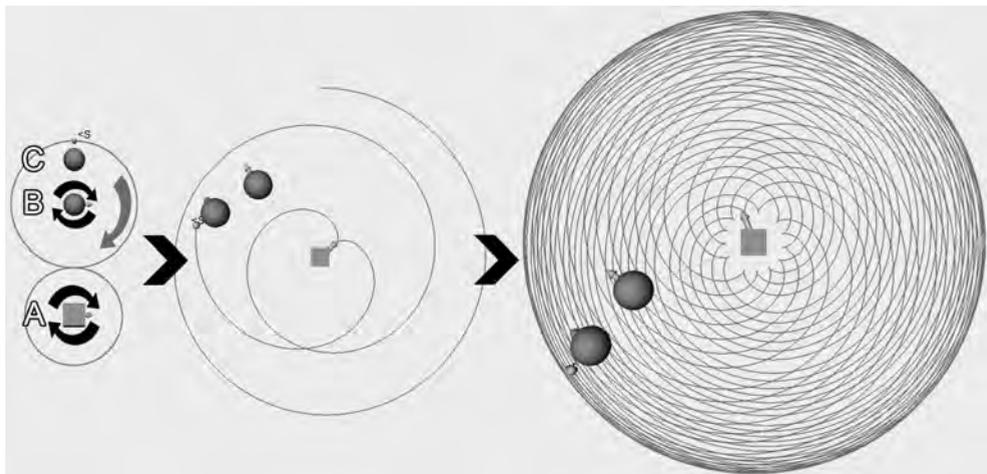


Figure 10. An epicycloid in PhysicsPlayground.

around the static shape A and shape C rotate around B. When the trajectory gets visualized through a path adapter on C, mathematical curves like epi- and hypocycloids are drawn.

## Conclusion

With PhysicsPlayground, a VR application was developed that aids teaching and learning mechanics. The physics engine PhysX on top of the Studierstube framework provides the technical basis and manages all physical calculations. The accuracy of the simulations is good, and considered sufficient for educational purposes compared to exact calculations which we used to verify the results.

The educational use cases demonstrate how versatile PhysicsPlayground can be integrated into physics lessons. The potential applications are manifold. Students can build virtual mock-up models of experiments to study physical properties, verify formulas, develop theories and actively participate in physics education in general. It fosters experimentation and understanding.

Up to now, PhysicsPlayground has only been evaluated by staff members of the Interactive Media Systems Group at Vienna University of Technology. No evaluations about its effectiveness for learning have been conducted yet. It would also be interesting to gather feedback on usability of the application as it uses its own GUI and interaction scheme. Therefore, an expert-based usability evaluation including physics teacher and students from different grades would be a meaningful next step.

## References

- AGEIA. (2006). *AGEIA Technologies, PhysX documentation*. Retrieved from [www.ageia.com](http://www.ageia.com)
- Gerthsen, C., & Vogel, H. (1993). *Physik - Ein Lehrbuch zum Gebrauch neben Vorlesungen* (17th ed.). Berlin, Germany: Springer.
- Goldstein, H., Poole, C. P., & Safko, J. L. (2006). *Klassische Mechanik*. Weinheim, Germany: Wiley-VCH.
- Hecker, C. (1996). Physics, the next frontier. *Game Developer Magazine*, 12-20.
- Kaufmann, H. (2004). *Geometry education with augmented reality*. Ph.D. thesis, Vienna University of Technology.
- Körner, W., Hausmann, E., Kießling, G., Mende, D., & Spretke, H. (1988). *Physik - Fundament der Technik* (10th ed.). Lipsk, Germany: VEB.
- Meyer, B. (2007). *Physics education in the field of mechanics with virtual reality*. Master thesis, Fachhochschule Deggendorf.
- Pohl, R. W. (1931). *Mechanik und Akustik - Einführung in die Physik* (2nd ed.). Berlin, Germany: Springer.

- Schmalstieg, D., Fuhrmann, A., Hesina, G., Szalavári, Z. S., Encarnacao, L. M., Gervautz, M., et al. (2002). The Studierstube augmented reality project. *Presence - Teleoperators and Virtual Environments*, 11(1), 33-54.
- Seugling, A., & Rölin, M. (2006). *Evaluation of physics engines and implementation of a physics module in a 3D-authoring tool*. Master thesis, Department of Computing Science, Umea University, Umea, Sweden.
- Stark. (2002). *FOS, Abschluss-Prüfungsaufgaben mit Lösungen, Fachoberschule*: Stark publishing company mbH.
- Szalavári, Z. S., & Gervautz, M. (1997). The personal interaction panel - A two-handed interface for augmented reality. *Computer Graphics Forum*, 16(3), 335-346.