This book is provided in digital form with the permission of the rightsholder as part of a Google project to make the world's books discoverable online.

The rightsholder has graciously given you the freedom to download all pages of this book. No additional commercial or other uses have been granted.

Please note that all copyrights remain reserved.

**About Google Books**

Google’s mission is to organize the world’s information and to make it universally accessible and useful. Google Books helps readers discover the world’s books while helping authors and publishers reach new audiences. You can search through the full text of this book on the web at [http://books.google.com/](http://books.google.com/)
An Alien’s Guide
to Multi-Adaptive Educational Computer Games
Acknowledgements

The research and development introduced in this book is co-funded by the European Commission under the seventh framework programme in the ICT research priority, contract number 215918 (80Days, www.eightydays.eu). The opinions expressed in this publication do not necessarily reflect the position of the European Commission.
An Alien’s Guide to Multi-Adaptive Educational Computer Games

Contents

Around an Inspiring Virtual Learning World in Eighty Days
Michael D. Kickmeier-Rust .................................................................1

Designing Learning Games
Daniel Schwarz and Martin Stoecker ..............................................5

Developing Educational Games
Ivan Orvieto .....................................................................................21

Balancing on a High Wire: Adaptivity, a Key Factor of Future Learning Games
Christina M. Steiner, Michael D. Kickmeier-Rust, Elke Mattheiss, Stefan Göbel, and Dietrich Albert .........................43

Adaptive Digital Storytelling for Digital Educational Games
Stefan Göbel, Florian Mehm, and Viktor Wendel .........................89

Realizing Just-in-Time Personalization – A Technology Overview
Owen Conlan, Cormac Hampson, Kevin Koidl, Stefan Göbel, and Florian Mehm .................................................................105

Integrating External Content: A Possibility to Reduce Development Costs
Lorenzo Oleggini, Sam Nova, and Ivan Orvieto ............................121
Around an Inspiring Virtual Learning World in Eighty Days

Michael D. Kickmeier-Rust

Computer and video games have become a very successful genre and an important part of today’s entertainment landscape. With the increasing amount of time people of all ages spend playing computer games, the idea of utilizing the motivational and didactic potential of those games for serious, in particular educational, purposes is becoming a more and more popular and fascinating idea. The European project 80Days (www.eightydays.eu) is a leading-edge research effort, inspired by Jules Verne’s novel “Around the world in eighty days”. The project ran from 2008 to 2010 and focused on developing psycho-pedagogical and technological foundations for successful digital educational games – successful in terms of educational effectiveness as well as financial turnovers.

A major objective of psycho-pedagogical research efforts was a scientifically sound framework for a non-invasive assessment of knowledge, learning progress as well as motivational states, and a subsequent comprehensive adaptation to the learner on micro and macro levels. The micro level refers to subtle educational interventions strongly embedded into the game such as tailored feedback or hinting within specific learning situations. The macro level, on the other hand, refers to an educationally appropriate sequencing and pacing of learning situations tailored to the individual learner.

The project successfully elaborated a joint formal model of cognitive assessment of learning progress (on the basis of Competence-based Knowledge Space Theory) on a probabilistic and non-invasive level, of non-invasive motivational assessment, the provision of suitable support and interventions, and open interactive adaptive storytelling (cf. Kickmeier-Rust, Hockemeyer, Albert, & Augustin, 2008). From a technical point of view, within the project an accurate analysis of learning and game design requirements has been carried out and the results have constituted the starting point for the study on system architectures and soft-
ware modules that best could have fulfilled the requirements. Research in the area of open, interactive storytelling achieved a technical realization of the developed formal model in form of a story engine, which implements the psycho-pedagogical model and which drives and adapts the game (Kickmeier-Rust, Göbel, & Albert, 2008). Overall, psycho-pedagogical and technical efforts led to a compelling prototype game. Significantly, this demonstrator also represents an important step towards achieving a multi-adaptive system that not only adapts discrete elements of the game towards psycho-pedagogical purposes, but also adapts the story to accommodate larger educational objectives. Very briefly, the prototype game is teaching geography for a target audience of 12 to 14 year olds and follows European curricula. The game design includes premises, concepts, metaphors, structures, gameplay, learning objectives, contents, background story, game characters, visual design and game props. The key objectives can be summarized with “create an Earth report” and “save the Earth”. In concrete terms, an adventure game was realized within which the learner takes the role of an Earth kid at the age of 14. The game starts when a spaceship is landing in the backyard and an alien named Feon is contacting the player. Feon is an alien scout who has to collect information about Earth. Subsequently player and Feon have to explore Earth and learn facts as well as concepts in order to successfully proceed through the game (a more detailed description of the game prototype can be found in chapter “The 80Days game”, p. 153).

During the project’s three development phases we developed a prototype game to demonstrate the conceptual and technical research. The code name for this demonstrator game was Lizard. At the end of each development phase a more elaborated version of Lizard was released (version. 1.0, 2.0, and 3.0). In addition to the game prototype a mockup player named Bat Cave was developed in order to provide insights into the smart assessment and adaptation features developed in the context of the project. The game was subject to multi-faceted evaluation activities. This work was geared towards its objectives of defining an evaluation framework and of implementing an array of evaluative activities. In close collaboration of different disciplines, the game design concepts and adaptive features were investigated with children from Austria, Germany, and the United Kingdom. Empirical findings yielded beneficial effect of playing the game, as evident and an overall satisfying usability and user experience. Implications for the future development of the game prototypes and the design of evaluative activities were drawn.
In particular, the theoretical knowledge and practical experience thus gained will contribute to advancing the research area of evaluating usability and user experience in digital educational games.

Ultimately, this leads to the purpose of the guide book. In the following sections we want to introduce the idea (the necessity) of personalizing educational gaming, the theoretical concepts and ideas, as well as the scientific, conceptual solutions developed. We want to share our experiences in educational game design, which can be considered more than being the sum of good learning design and good game design, and we want to present a role model for a successful curriculum-related design. In essence, we want to provide a set of ideas, experiences, and solutions to guide the reader through the process of designing and developing the next generation of educational computer games.
Designing Learning Games

Daniel Schwarz and Martin Stoecker

The totality of all design aspects of a learning game is called the Learning Game Design. Learning Game Design is a relatively new craft that fuses two core building blocks, whereas each building block is a science of its own:

- Game Design
- Didactic Design

The goal of this fusion is to create a motivational and instructional learning game experience that turns the successful player into a successful learner and vice versa. There are further accompanying design disciplines who are involved in the realization of this fusion of Game Design and Didactic Design:

- Story Design
- Visual Design
- Sound Design
- Interface/Interaction Design
- Information Design
- Character Design

Those design disciplines provide the artistic means in order to realize the methods of the didactic design and the game design. Their design contributions make up the overall Learning Game Design. This chapter describes the goals and the approach of the overall design work on the 80Days Learning Game without going into the details of the respective design disciplines as they are described in the Game Design Document. Using the image of a tree for the whole Learning Game (Figure 1), we will describe the process of Learning Game Design as a growth process starting from the soil to the final fruit.
Soil: The Zeitgeist that Teaches Geography

The Zeitgeist is the nourishing soil in which our Learning Game-Tree is rooted. All branch wood high up in the air is grown on the common ground for Geography. The Zeitgeist is the cultural context that forms our understanding of what Geography means to us today. We had to ask ourselves, “What do we want to achieve with the teaching of Geography today?”, “What kind of Geographers do we want to have tomorrow?” To answer these questions, we must know the composition of this soil, the current “Zeitgeist of Geography”. In order to successfully plant our learning game tree and nourish the roots of our premises for learning, story and gameplay.

We undertook a comprehensive evaluation standard for Geography learning. We learned that the teaching approach to the Geography subject changed a lot from the introduction of the topic in teaching plans until today. The composition of the soil had changed since the Zeitgeist had sailed on wooden ships, following the stars with a needle of steel while the tint on the hand-drawn maps still was fresh. For the present Zeitgeist of Geography, we finally found a comprising answer in the UN-Decade for Education and Sustainable Development (2005-2014) that sets the goals for teaching Geography for the near future:
1. The Earth-human relationships are now the focus of the teaching approach.
2. The topic of Geography, therefore, becomes more interdisciplinary.

Furthermore, there was a continuous theme in all of the European curricula which we later conceptualized as the “Long zoom of Geography” – the British curriculum described it best by stating that “Scale matters” in Geography:

“Scale influences the way we think about what we see or experience. Any geographical enquiry benefits from being viewed from a range of scales to develop an understanding of how these scales are interconnected.”

Based on the themes that we had identified as the Zeitgeist for teaching Geography, we were able to define the premises for learning, story, and gameplay as roots for the Game. In parallel this analysis provided the information needed for the identification, grouping, and selection of the main learning themes to be included in the game. It also delivered as results the Learner requirement specifications and resulting detailed skill lists of selected learning objectives.

**Roots: The Premises of the Game**

Based on the research and findings in the breeding ground of the ZEITGEIST, it is up to inspiration and creativity to ingrain the games’ premises for teaching Geography.

**Why do we Need Premises for Designing a Learning Game?**

A premise is the literal root of all good or evil in a film, book, stage play, or video game. If the premise of an author is that the world is bad, there will be mainly bad events in her story that will happen to the main character: the main character will behave badly, will be treated badly, or both. And finally, it is very likely that there will be a negative outcome of the whole story in the end, because the author’s main message is that the world is bad. The premise gives the tune for the melody to be played in the play. It can be in a major or minor key, but it surely lays down the tracks for the outcome of the story, the essence of what an author wants to convey about the world to the audience. This holds
true for any art form that tries to convey a message or experience to the audience – including the art form video game or the art of giving school lessons.

**Premises Set the Goals of a Learning Game**

In video games the outcome of the story is equivalent to the goals to be achieved in the game by the player. Here the premises – similar to the process in screen writing - define the wished outcome of the game experience by setting the goals of the game that have to be achieved by the player. With the premises we lay down the tracks for the events that will make the player learn.

But unlike a stage play, novel, film, or video game which is conceived solely for convincing entertainment, a digital learning game has to achieve two main goals: On the one hand it has educational objectives, and on the other hand it has motivational objectives. The educational objectives deal with a comprehensible and sustainable transfer of the educational knowledge to be taught with the game, while the motivational objectives strive to excite the motivation of the learner to go on with the learning game experience and to finalize it successfully.

In order to achieve those motivational objectives the medium computer game offers various motivational means just as a film would make use of different dramaturgical means like music, plot points, or montage to tell its story. The two main motivational tools in digital games are “Storytelling” and “Game play.” Their deployment in the game allows for the achievements of the motivational objectives. Therefore we subsume for the following considerations the motivational objectives with these two motivational means.

Table 1 shows a confrontation of the educational objectives within the 80days-project and the motivational objectives that shall contribute to the achievement of these educational objectives. The motivational objectives are subsumed in the right column under the two main motivational tools.

In digital learning games there is a clear hierarchy of the two main goals:

- The supreme goal of the learning game is the achievement of the learning objectives.
The motivational objectives however must serve the learning objectives.

In order to realize those two main goals – education and motivation - there are three premises with which the learning game is rooted in the ground; one is the dominating one from which the other two branch off: the premise for learning Geography.

Table 1. Educational and motivational objectives.

<table>
<thead>
<tr>
<th>Educational objectives of the topic Geography in 80 days</th>
<th>Motivational objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Basic Geographic Instruments, use of maps and globes</td>
<td>• Create suspense</td>
</tr>
<tr>
<td>• Basic Knowledge of the Earth</td>
<td>• Create curiosity</td>
</tr>
<tr>
<td>• Physical Geography Processes and Implications</td>
<td>• Create meaning</td>
</tr>
<tr>
<td>• Human and Economic Geography processes and Implications</td>
<td>• Create challenge</td>
</tr>
<tr>
<td></td>
<td>• Create emotion</td>
</tr>
<tr>
<td></td>
<td>• Create affection</td>
</tr>
<tr>
<td></td>
<td>• Create desire</td>
</tr>
<tr>
<td></td>
<td>• Create immersion</td>
</tr>
</tbody>
</table>

The Premise for Learning Geography

What is the premise for learning Geography if the Zeitgeist demands a focus on “Earth-human relationships” together with the consideration of “interdisciplinarity” and a “range of scales”?

We found our premise for learning by defining “Life on Earth” as the crucial core statement that was given by the UN-Decade for Education and Sustainable Development. Therefore, we put life on Earth as the nucleus of our concept for learning Geography (see Figure 2).
The Premises for Story and Gameplay

The Premise for learning serves as an overall premise for the learning game. Having defined it, we can deduce the other two premises for story and gameplay in the following order:

1. The premise for learning determines
2. The premise of the story
3. Both determine the premise of the game play

The premise of the story has to honor the educational premise, while the premise of the game play at the end of the chain is determined by the educational premise and the premise of the story. Figure 3 shows the dependencies between the three premises that make up the base for the successive development of the game metaphor and concept.

Figure 2. The Premise for learning Geography with the 80Days game.
**Log: The Game Metaphor**

*What is a Game Metaphor?*

What is a game metaphor and why would we need one? According to George Lakoff and Mark Johnson (1980), the quintessence of any metaphor is “understanding and experiencing one kind of thing in terms of another” (p. 5). As Johnson and Lakoff stated, we literally live by metaphors and everyone uses them in his/her everyday activity. By means of the saying “Time is money” they explain its metaphorical concept: TIME IS MONEY, TIME IS A LIMITED RESOURCE and TIME IS A VALUABLE COMMODITY are all metaphorical concepts. They are metaphorical since we are using our everyday experiences with money, limited resources, and valuable commodities to conceptualize time. This isn't a necessary way for human beings to conceptualize time; it is tied to our culture. There are cultures where time is none of these things. By using the saying “Time is money” we explain time with money. Scientists use metaphors, too when they work with (mental) models of theories which explain the nature of physics or biology.

For game design the usage of metaphors plays an important role, since a game creates a virtual experience that can be completely detached from the experience the player makes in her real world. How would the players know what this strange world means and what they would have to do within it? To play and enjoy the game she must understand the goal, the challenge, the game elements, and the rules of play. Designing a game for a player means that you create an experience for the player that emerges from her interaction with the game world. To have a satisfying game experience it is crucial that the playing is meaningful for the player. Meaningful play can only emerge when the player understands the game world that has been created by the designers. From a more systemic view this means that the player, as an actor in the game system, understands the meaning of the objects in the game, their attributes, their internal relationships, and the environment of the game in which all of these things occur and take place (see Salen & Zimmerman, 2004). The player must be able to recognize familiar concepts in the game world that help her to understand their meaning. Since game worlds cannot copy the richness of the real world one-to-one or introduce alternative worlds the game designer must find a metaphor that helps the player to understand and experience the game world in terms of the real world.

The Game Metaphor for 80Days: The “Long Zoom”

What metaphor would we use in a computer game in order to make Geography understandable for the playing learners? The most overt metaphor is already in use: Google Earth takes the globe on our desks and in our classrooms and enhances it with the interactive qualities and real-time visualization of the computer. Thereby it transforms the metaphor of the good old globe as a physical object on your desk into a virtual and interactive Geography Information System (GIS), which is enjoyable to use both for tourist information retrieval and explorative discovery of the planet.

The learning game in 80Days is basically a “long zoom” into the domain of Geography with the aim to teach young learners the learning topic Geography with the premise for learning that we have defined before (please see Figure 3).

The “long zoom” of 80Days conveys a learning experience of Geography that spans from a global view of the whole earth down to the exploration of a specific geographical site. The key for a successful learn-
The intention of the Game Metaphor for the Didactic Design

As a teaching approach the concept of the “long zoom” enables the representation of the complexity and profundity of Geography by allowing the learner to experience geographical topics and sites from different levels of scales while maintaining the greater context of learning. The greater context of learning is 80Days’ special premise of Geography: Learning Geography should enable children to face the challenges of the human impact on the conditions of life on Earth – or in short: “Geography saves the Earth”. An appropriate background story tells this premise and its resolution and keeps it alive and visible throughout the “long zoom”.

The intention of the Game Metaphor for the Game Design

As a frame for the game play the game metaphor of the “long zoom” enriches the playing experience with different kinds of game plays for the different levels of scale. The game consists of three different game plays that build upon each other and can be seen as three different levels of scales of the “long zoom”. They could stand alone as single games, but together they form the cohesive geographic learning experience that makes 80Days unique.

Crown: The Game Concept

The soil (Zeitgeist), the roots (Premises) and the log (Metaphor) carry the game concept as the crown of the Learning game tree. Three main branches outgrow the log and roots and form the shape of the crown:

The branch of goals, the branch of structure, and the branch of content together shape the overall design concept of the learning game. In these branches later on the concrete learning situations will develop and hang like the proverbial apple from the tree.
Goals

In the branch of goals the game concept describes the goals of the learning game:

- Learning objectives
- Gameplay objectives
- Story objectives

Structure

In the structure branch the game concept describes:

- The domain knowledge and competences to be learnt as a skill tree.
- Possible learning paths and according story lines in form of game chapters (Intro, Tutorial, Micro-Missions) and an outline of story situations, gameplay situations and learning situations (Learning, Gameplay or Story Situation within a Micro-Mission) as well as the listing of skills that are coupled to the learning situations; typically the skills are pooled in topics (e.g. “natural hazards”) which allow for a structuring of the learning experience into thematic chapters
- The pedagogical rules and adaptive mechanics that will create the adaptive creation of learning paths for the learner that will lead her through the branches of the skill tree
- The gameplay modes and their according game mechanics that will create the gameplay experience for the player (flying the UFO, manipulating objects in a virtual terraforming simulation at the map desk in the UFO, interacting with the game character Feon in story and learning situations)

Content

In the content branch the game concept describes:

- the learning content in form of skill lists
- the respective learning action categories for those skills (see Figure 4)
• the listing of possible **gameplay elements** which translate the learning action categories into according gameplay in order to enable action-oriented learning (see Figure 5)

• the most important **game elements** that are required to realize the game concept
  
  o description of game world and its look (concept art, not finished design)

  o objects of the game world for the most important game features (the UFO for flying to different geographical sites as an equivalent to spinning the globe or hanging a map on the wall in the classroom)

  o game elements like the Head-up-Display for in-game information display or tools the player can use (e.g. mobile phone in order to communicate with other characters like the Geography professor who serves as an information source)

• the game characters and their roles and functions for para-social learning.

The game concept document describes the overall learning game design throughout all the design disciplines involved without going into the details of a single situation in the game or the detailed specification of a game element. Once it is finalized, its main branches will branch out into the more elaborated details of the Game Design Document, which is supposed to be the finalized and very detailed instruction for the programmers and artists on how to develop a game.

<table>
<thead>
<tr>
<th>Number</th>
<th>Learning action category</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>28</td>
<td>skills can be told by NPC</td>
<td>38</td>
</tr>
<tr>
<td>24</td>
<td>skills are learned by seeing and observing</td>
<td>33</td>
</tr>
<tr>
<td>11</td>
<td>skills are learned by experimenting and constructing (simulation)</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>skills are learned by measuring and taking samples</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>skills are taught by analysis and conclusion</td>
<td>4</td>
</tr>
</tbody>
</table>

*Figure 4. Distribution of skills on possible learning action categories in Lizard.*
Apple & Picker: The Learning Game Situation

The learning game situation is depicted with an action rather than an object like the log or tree crown: The learner must place a ladder, climb up the ladder, and reach for the goal – the apple that is to be picked and out of reach at first. This image is used to clarify that we do not talk about a “learning object” that is delivered and presented to the learner – as it is understood by the e-learning approach - but about a spatio-temporal situation in which the learner is in an experience which is embedded in a meaningful and goal-oriented context that is designed to foster her learning.

Figure 5. The mapping of identified learning action categories to gameplay elements in Lizard.

A learning game situation must fuse the dramaturgical (story), didactic (learning) and interactive (gameplay) means and deploy them to arrange the learning situation as a challenging constellation of:

- Actor (learner)
- Goal (learning objective, story objective and motivation)
- Obstacle (problem-solving by means of reasoning and action (gameplay)).
If the effort to master the learning situation is supposed to be worthwhile for the learner, we must design it like the tempting fruit hanging high above from the tree:

Being designed as a tempting or rewarding fruit, the situation has to offer a goal and an obstacle to the actor – causing her to learn to become a successful apple picker. The obstacle would be a challenging gameplay activity (→ use the ladder, stretch yourself to reach out to the apple, detach the apple’s stem from the twig). The goal would be a story reward or function (→ steal the apple from the forbidden tree and bring it to the evil magician in order to ransom the captured princess), whereas the story goal would be the fruit flesh that covers the apple core, which stands for the learning goal and the related skills to be learnt in this situation.

**Challenges of Learning Game Design**

**Game Metaphor**

- Find a game metaphor that is suitable for the complete range of the learning content and that can lead - at the same time - to a game concept that is realizable with the available resources.

**Game Concept**

- Create gameplay modes that allow the player to manipulate, act and interact within a learning game situation.

- Transform the factual knowledge to be learnt into action-oriented learning. For this the designer must collaborate with the didactic designer and domain expert to create a link between learning action categories that are defined on the side of the didactic design and gameplay elements that are defined on the side of game design.

- Create storylines that follow the hierarchical build-up of knowledge according to the prerequisite relations of the single skills in the skill tree. The goal is to link possible story paths to valid learning paths, which are constrained by the prerequisite relations of the skill tree. Concretely, this means that the story designer needs to arrange places, actions, and a basic drama-
turgical development in line with the development of knowledge as it is foreseen by the skill tree which determines the level design (see Figure 6).

Figure 6. Early draft of structuring the game chapters of Lizard into Intro, Micro-Missions and Extro by means of taking the skill tree as the underlying design structure for the level and story design.
Design Learning Game Situations

Creating a learning game situation that offers a good integration of gameplay, storytelling, and learning is the most difficult task in the whole process of learning game design. The best way to start the design work is to take a single skill or group of skills that belong together and work with the learning action categories that were defined for those skills by the pedagogical expert and the domain expert. Since the related gameplay elements are already defined in the game concept the designer can use it as a basis and creative limitation for the gameplay action to create the situation.
Developing Educational Games

Ivan Orvieto

Videogames have gained more and more importance over last years, becoming one of the most pervasive entertainment activities among people. Not surprisingly one of the latest surveys conducted in the USA reports that 67% of American households play videogames, spending something like 10.5 Billion of dollars. (Entertainment Software Association, 2010). Moreover, fully 97% of teens ages 12-17 play computer, web, portable, or console games; additionally 50% of teens played games “yesterday”, 86% of teens play on a console like Microsoft Xbox, Sony PlayStation, or Nintendo Wii, 73% play games on a desktop or a laptop computer, 60% use a portable gaming device like a Sony PlayStation Portable, a Nintendo DS, or a Nintendo Game Boy, 48% use a cell phone or handheld organizer to play games (Lenhart et al., 2008). The situation in Europe is pretty similar (Interactive Software Federation of Europe, 2010) - thanks also to the diffusion of broadband and related drop of internet connection costs.

Because videogames are so present in our lives, it becomes quite natural to look at them for a usage outside pure entertainment. Similarly to programs presented on television, dealing with entertainment, education, news, etc., videogames can be used for training, teaching, entertaining, informing, advertising, etc.

Videogames Development

Videogames development has changed very much from the first exemplars like Pong, developed by one person with very limited resources, to triple-A multiplatform games like Call of Duty Modern Warfare 2 (Infinity Ward, Activision) that required hundreds of people and many millions of dollars to be developed.

Games are now available for a great number of platforms (PC, consoles, portable consoles, mobile phones etc.), with several different
game genres (racing, sport, puzzle, action, strategy, etc.), for different target users (casual gamers, hardcore gamers, family, etc.).

This panorama describes a situation where game development has become a serious business, with significant budgets and worldwide diffusion.

**General Overview of Videogames Development**

First of all it is important to give a definition of what a videogame is. A videogame is an electronic game that involves interaction with a user interface to generate feedback on an audio-video device. Common elements of a videogame are visible things like 2D and 3D graphics, Sound, Music, Dialogues, Text, but also not visible components like Source code and Game design. Further in this chapter a definition of these elements will be provided together with a description of the process and professional figures involved in game development.

**The Process of Videogame Creation**

Despite the fact that every developer adopts his or her own methodology and tools to create a videogame, a general description of how a videogame is developed is possible. As we have seen above, a videogame is made of graphics, art, sound, and music, usually defined as assets, but also of software modules commonly mentioned as technology plus an intangible essence named game design that describes how the game should appear and behave.

The usual flow adopted for the development of a videogame is the following: assets are managed by technology to implement indications provided by game design. Technology is a broad term that incorporates many different software modules; the most common are game engines, editors, sound engines, network managers, physics engines, and rendering engines.
Figure 7. Schematic view of 80Days learning game development.

Figure 7 shows various phases of learning game development in 80Days together with their relationship in detail:

- *Game design* and *Learning design* serve as input for the entire process.

- Once they are defined several activities can start:
  - R&D: often required, especially when new features are required.
  - Graphics, audio and text production.

- *Engines and tools* are created starting from results of R&D and inputs coming from Game design and Learning design.

- Once engines and tools and all assets are ready it is possible to start developing the game defining gameplay situations via *script programming*.

- The game takes shape thanks to integration phase where assets are merged together with technology.
• **Beta testing** is one of the most important phases where functionalities, coherence with Game design and Learning design, usability are checked

• Depending on results of Beta testing, it is possible that the process turns back to script programming to slightly change features or to fix bugs or, seldom, back to other phases even to Game design and Learning if major issues arise.

**The Development Team**

Every team that develops videogames holds a certain number of different roles among its lines: 2D and 3D graphic artist, animators, programmers, game designers, testers, translators, producers, and many more depending on the complexity of the game.

Nowadays teams are made of hundreds of people, often split up in different locations, that work together for 2 to 4 years on a project. It is also quite common that a development team is dismissed at the end of the project since the successive one can start after some months of pause, thus reducing unfruitful costs. Once the new project starts the team is rebuilt on the basis of updated requirements.

For very big productions, and in general to reduce development costs, it is quite common to hire professionals that work in teams in countries like China and India under supervision of producers coming from the core development team.

**Videogames and Learning Games**

Having seen the success videogames have among young people it is obvious to think about their adoption for improving learning thanks to their impact on motivation and their immediate acceptance by targeted users. But of course there are some problems to be solved like reducing costs of development, designing effective learning games that convey traditional learning content within an innovative media, and finding profitable business models.

What is a learning game then? Unfortunately there is not a set definition since the subject is quite new, but it can be thought of like a videogame that provides learning elements while played. The efficacy of learning is the big question mark, and often it is also not so easy to validate such efficacy. The key factor is that players should not feel the
Developing Educational Games

sensation of being involved in a learning process but should enjoy the game they are playing and be involved in it. But this is also the tricky part of the story, how to create effective learning games that can find a place in a market that is not completely defined so far, but that requires, as always, a good cost-benefit ratio.

To accomplish a significant step towards successful and effective educational games, 80Days project has addressed the following two main objectives: first, integrating models of adaptive personalized learning with those of adaptive interactive storytelling, and, second, merging virtual game environments with existing learning resources, thus reducing development costs and time. These fusions has resulted in an adaptive and responsive system, enabling the understanding of active learning processes within a virtual learning environment, adapting to individual needs and abilities, and therefore fully exploiting the learners’ capabilities.

Compared to traditional videogame development processes, making a learning game presents some new challenges. First of all, from the point of view of contents, game designers should work in parallel with teachers and learning experts to define how the game should behave, what learning elements should be presented to players, and what type of communications should be used. The second main topic is technological: new software modules should be created to fulfil learning requirements and they should be integrated in traditional videogame architectures. The third challenge is about methodologies: the world of videogame development and the one of learning experts should find a way, a common language, and new procedures to work together, putting their expertise together toward a common goal.

Common Approaches to Learning Games Development

Looking at the actual panorama of learning games development, there are three main approaches for integrating games into the learning process: have students build games from scratch, integrate commercial off-the-shelf (COTS) games into the classroom and have educators and/or developers build educational games from scratch to teach students.

In the first approach, students take on the role of game designers; in building the game, they learn the content. Traditionally, this has meant that students develop problem-solving skills while they learn programming languages. Professional game development takes one to two years
and involves teams of programmers and artists. Even though this student-designed approach to digital game-based learning (DGBL) need not result in commercial-quality games, it is nonetheless a time-intensive process and has traditionally been limited to computer science as a domain. It is certainly possible for modern game design to cross multiple disciplines (art, English, mathematics, psychology), but not all teachers have the skill sets needed for game design, not all teach in areas that allow for good content, not all can devote the time needed to implement this type of DGBL, and many teach within the traditional institutional structure, which does not easily allow for interdisciplinarity.

A basic example of this approach is the so-called Microworlds. Microworlds are tools that allow students to explore and test their ideas as they create science simulations, mathematical experiments, and interactive multimedia stories. Usually provided with a simple programming language (e.g., LOGO) they are useful to build small game and learning experiences to very specific topics. Major disadvantages are the usual lack of storytelling and poor graphical appearance that doesn’t help motivation.

The second approach—integrating commercial off-the-shelf digital game-based learning (COTS DGBL)—involves taking existing games, not necessarily developed as learning games, and using them in the classroom. In this approach, the games support, deliver, and/or assess learning. This approach is currently the very cost-effective in terms of money and time and can be used with any domain and any learner. Quality is also maximized by leaving the design of game play up to game designers and the design of learning up to teachers. This approach to DGBL seems to be the most promising in the short term because of its practicality and efficacy and in the long term because of its potential to generate the evidence and support we need to entice game companies to begin developing serious games.

Integrating COTS games is not without its drawbacks. Commercial games are not designed to teach, so topics will be limited and content may be inaccurate or incomplete. This is the biggest obstacle to implementing COTS DGBL: it requires careful analysis and a matching of the content, strengths, and weaknesses of the game to the content to be studied.
Examples of COTS being used in classrooms are, among others, *Civilization* (Sid Meier, Microprose) to teach history, *CSI* (369 Interactive, Ubisoft) to teach forensics and criminal justice, and *SimCity* (Maxis) to teach civil engineering and government.

80Days falls in the third category, where games are designed by professionals to seamlessly integrate learning and game play. Touted by many as the “Holy Grail” approach to DGBL because of its ability to potentially address educational and entertainment equally and to do so with virtually any domain, this professionally designed DGBL process is more resource-intensive than the first two options. This is because the games must be comparable in quality and functionality to commercial off-the-shelf (COTS) games, which after all are very effective in teaching the content, skills, and problem-solving needed to win the game. The development of such “serious games” is on the rise, and the quality of the offerings is promising (e.g., *Environmental Detectives*, developed by the Education Arcade; *Hazmat: Hotzone* under development at the Entertainment Technology Center at Carnegie Mellon University; *Virtual U*, originally conceived and developed by Professor William F. Massy; and *River City*, developed by Professor Chris Dede, the Harvard Graduate School of Education, and George Mason University).

In the specific field of Online Multiplayer Learning Games some examples are *Revolution* (Massachusetts Institute of Technology), a Massively Multiplayer Online Role Playing Game set during the American War for Independence, *Wolfquest* (Minnesota Zoo, Eduweb), a 3D wildlife simulation game where players join a wolf pack made up of friends or computer-controlled wolves and, through trial and error, instinct, and experience, learn to maximize both individual and pack survival, NASA Massively Multiplayer Online Game "*Astronaut: Moon, Mars and Beyond*", currently under development.

**The 80Days Approach**

80Days takes its origins from previous projects, among them projects such as ELEKTRA, art-E-fact, U-CREATE, iClass, and INSCAPE.

From a technological point of view, there are certain software modules that will serve as a basis for the development activities of 80Days and that have been created (completely or partially) during those previous projects, in particular:
• **Game engine and rendering engine:** used to manage and visualize learning and gaming activities; they will be built starting from the basis created in ELEKTRA.

• **Scene editor:** used to prepare 3D scenes to be managed by game engine; it will be created by modifying the ELEKTRA editor.

• **Story Editor:** used by the authors/designers to manage the story structure; it will be derived from the prototype created during art-E-fact, U-CREATE and INSCAPE.

• **Story Engine:** used to manage the flow of the story and eventually intervene in it; it takes its basis from the projects Virtual Humans and INSCAPE.

• **Learning Engine:** used to facilitate personalization around a learner's skills, and to support the seamless integration of this dynamic component into the game’s non-linear storyline; it will be built upon the technology developed during the ELEKTRA project.

In a structured and interdisciplinary project like 80Days, it is of fundamental importance the definition of the software components and the general architecture of the system because they act as a pillar for the entire project. The main components of the 80Days can be summarized here:

**Game Engine (GE):** responsible for gameplay management and overall system coordination.

**Story Engine (SE):** responsible for story flow management and macro adaptivity.

**Learning Engine (LE):** responsible for learning management and micro.

**Toolset:** set of tools used by developers to create a story flow compliant with SE and a 3D scene compliant with the GE. More than this, it increases cost effectiveness by providing editors to speed up the process of game creation.

In few words, GE directly interacts with a user receiving his/her inputs and giving necessary output in order to provide full game (and learning) experience. During the game experience all significant game situations and learning activities are sent to SE. Using that information, the SE can manage the flow of the story and, in case, intervene in it, sending to
GE Adaptation Requests from a macro level point of view. Furthermore, the SE communicates with the LE by passing information (game situations, learning activity) received by the GE and getting Adaptation Requests that work from a micro level point of view. The SE may filter or not information coming from GE or LE in order to allow a direct communication between the other two engines.

Further on we will describe in particular how the Game Engine has been developed, its architecture, and how it is related with other modules of the system.

![Figure 8. 80Days’ Overall system architecture.](image-url)
Game engine choice

As seen above the Game Engine has been inherited from previous project ELEKTRA and improved during the 80Days project. A Game Engine is quite complex software; to avoid developing it from scratch at the beginning of the ELEKTRA project a deep analysis of available Game Engines on the market was conducted. Three main categories have been inspected: AAA Commercial, Open source, and Web-based engines. Among the category of AAA Commercial engines we have analyzed Gamebryo (Emergent Game Technologies), Unreal Engine 2 (Epic Games) and CryENGINE (Crytek).

Open source engines we considered were Nebula 2 (Radon Labs) and Ogre (Ogre Team).

In the last category of Web-based engines we tested Shockwave 3D (Adobe) and Virtools (Dassault Systemes). The requirements were various: flexibility, expandability, ease of use, cost, support, presence of a community, platforms supported (PC, Web, Consoles, Mobile devices), videogames already developed with, performances, quality of rendering, availability of tools/editors. Table 2 presents the main results of the analysis with pros and cons.

It became immediately clear that AAA Commercial engines were not suitable for a R&D project due to the very high costs of licensees. The choice was then between open source and Web-based ones. Game engines belonging to both categories were interesting since licensees costs were very low to zero and features interesting enough for the project’s purposes. The main difference between the two categories is the platform on which to deploy games made with them: analysed open source game engines are suitable for Microsoft Windows, Linux, and Apple Mac, while web-based ones can deliver games playable directly in web browsers for Microsoft Windows and Apple Mac.

Due to low performances and worse graphics quality we have decided to drop the category of web-based game engines. It is interesting to mention that this choice is more project related than an overall assessment of the value of one or the other category, both are good or bad depending on the requirements of the game that should be developed.
### Table 2. Comparison of various Game Engines.

<table>
<thead>
<tr>
<th>3D ENGINE</th>
<th>PROS</th>
<th>CONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAMEBRYO</td>
<td>Flexible graphics engine and comprehensive set of tools for special effects and shader support. Multiplatform.</td>
<td>Licence price ($50,000.00 per SKU (i.e., per title per platform) with full source code), start-up cost</td>
</tr>
<tr>
<td>UNREAL ENGINE 2</td>
<td>Astonishing graphics, Performances, A.I.</td>
<td>Licence price (US $350,000 for one of the available Unreal Engine 2 platforms + royalty of 3% due on all revenue from the game), start-up cost</td>
</tr>
<tr>
<td>CRYENGINE</td>
<td>Astonishing graphics, Performances, A.I.</td>
<td>Licence price, startup cost</td>
</tr>
<tr>
<td>NEBULA 2</td>
<td>Great graphics, very well integrated scripting, toolkit for Maya. Mangalore game frame work.</td>
<td>No native 2D features</td>
</tr>
<tr>
<td>OGRE</td>
<td>Good 3D Rendering Engine</td>
<td>It’s only a renderer</td>
</tr>
<tr>
<td>SHOCKWAVE 3D</td>
<td>Good rendering, physics and character animation. Player well spread all over the Internet</td>
<td>Not very good performance</td>
</tr>
<tr>
<td>VIRTOOLS</td>
<td>Optimized workflow thanks to a robust development hub. Good for very rapid prototyping and game development (online, downloadable, budget games and license-based games)</td>
<td>Not very easy to be used</td>
</tr>
</tbody>
</table>

Among open source engines, choosing between Nebula 2 and Ogre has been quite difficult since both present good features and an appreciable easiness of use. Since Nebula 2 comes together with editors and tools (Mangalore), making it a complete system to develop games, compared to Ogre that is less complete but with a great community of users, has
convinced us to adopt the first so to be able to work on the game from day one instead of creating tools to be used then for development.

Nebula 2 has revealed to be a robust system throughout both ELEKTRA and 80Days that allowed us to concentrate on projects presenting very few technical problems in over 4 years of development.

**Engines integration**

The integration of the three main engines of the 80Days’ system, GE, SE and LE, has been a delicate and time consuming activity for various reasons:

- they were built with different technologies (SE and LE)
- they had to speak same language (both technically and semantically)
- they have been developed by different partners
- they had to work in real time without interfering (in a disrupting way) with player experience
- they had to provide consistent interventions in the game flow

The difference in terms of technology was one of the deals: we had to arrange communication between GE and SE on one side and from SE to LE on the other. The first part of the problem has been quite easy to be solved since both engines (GE and SE) have been developed in C++, so integration has been performed by considering SE as a DLL (Dynamic Linked Library) so to use it as a common external module of the GE.

Unfortunately communication between SE and LE has been more difficult than the previous one. SE had to communicate with LE and vice versa, but they were developed with two different technologies. The SE is implemented as a C++ native application (like the GE), while the LE is realized as a Java Application that works on top of a dedicated virtual machine. In order to communicate they had to use a low level of interaction using structures and communication channels provided by the operating system. After the first analysis was carried out, the most promising solution to have a fast data exchange (required to keep real time responses in the system) was to create a socket for communication through a dedicated communication protocol over a TCP/IP connection.
Another important issue was to provide consistent interventions in the game flow. The main idea, as can be seen in Figure 9, was to avoid the problem that multiple adaptation requests could arrive from LE and SE to the GE. Thus the architecture of the system has been designed so to have adaptation requests passing through the SE that acts like a manager and decides which data should arrive from or going to the LE.

**Game engine architecture**

The main role of the GE is to provide runtime functionalities for presenting and modifying game experience. It offers game core features and – driven by SE - works as controller for overall system.

In Figure 10 the main components of the Game Engine are depicted:

- **Nebula2 Rendering Engine**: responsible for the visualization of the game experience.
- **Mangalore Game Framework**: responsible for the management of the game features and game entities and provider of core functionalities for gameplay management.
- **Scripting Engine**: responsible for the provision of the high level interface used to develop gameplay and entities’ behaviours. A link and synchronization with interactive story parts
defined in the Story Editor (Action Set Editor) have been established.

- **80Days Core Application**: is the main runtime environment of the application and it’s responsible for the coordination of all game components.

- **External Modules**: responsible to provide non-core features to the GE (i.e. Physics Engine, Terrain Engine, GUI System, etc.)

In Figure 10 a more detailed scheme of Game Engine architecture is proposed and a description of the main modules is presented in subsequent paragraphs.

**Rendering engine: Nebula 2**

The very base of a 3D videogame is the rendering engine that is responsible for showing the game experience on users’ screens. Nebula2 is an open source multiplatform rendering engine developed in C++ developed by “Radon Labs GmbH”. It is like an “Operating System” for games because it provides core rendering functionalities and it is organized in modular layers like a real operating system.

![Game Engine main components](image-url)

*Figure 10. Game Engine main components.*
Technically speaking, the Nebula2 rendering pipeline is shader based and it can be completely defined externally in respect to the engine source code with no need to recompile the engine itself. The design of the engine code makes use of many design patterns and it’s quite intuitive. Key Nebula2 rendering engine features are:

- Object-Oriented design based on a plug-in architecture philosophy.
- Hierarchical Object Name Space: all Nebula2 Objects live in a hierarchical tree of named nodes like a file system.
- Commercial Maya exporting tools that support also pre-calculated lighting for static scenes and both hierarchic and bones animations.
- Native Dx9 renderer that implements Direct3D Effect Files
- Per-Vertex, Per-Pixel lighting including light mapping and Gloss maps.
- Vertex and pixel shaders including High Level Shading Languages
• High performances Key frame and Skeletal Animations including Morphing and Blend shapes.

• Scene culling algorithms like Octrees for outdoor scene rendering and Portals for indoors

• Advanced particles systems, lighting effects, dynamic shadows and other visual effects

**Game engine core: Mangalore and the core application**

As seen above, GE is responsible for gameplay management and overall system coordination. In ELEKTRA and 80Days the provision of such functionalities has been carried out by a software module developed starting from the base code of Mangalore. Mangalore is a game framework built on top of Nebula2 by Radon Labs. It is developed following the data driven principles that completely separate data from logic application allowing a very modular development. Entities, properties and managers are basic concepts in Mangalore development and they allow developers to expand it easily.

In 80Days we have expanded Mangalore framework by adding specific features that were required by game design and learning design.

**Scripting engine**

The Scripting Engine is the high level interface of the GE. It allows the development of game features and game situations by using a scripting language instead of a programming language. This offers great advantages in terms of cost effectiveness, reduction of errors, possibility to create game situations by non-expert programmer.

During the ELEKTRA Project the Python Software Foundation Python scripting language has been chosen because it is very well documented, widely adopted, and it has a big set of extensions and applications. Thanks to positive results achieved in ELEKTRA, we decided to adopt it in 80Days too. The scripting engine interprets Python scripts offering access to the GE structures and managers, allowing developers to control the game from a high level point of view. Within 80Days, functionalities of the scripting engine have permitted developers to access to a lower level of the GE (directly to the rendering engine when needed) in order to encapsulate core/engine functionalities in a more modular architecture.
Story engine interface

The Story Engine Interface is a module of GE that is responsible for the communication with SE. Its work consists in sending data regarding game situation and learning activity and receiving adaptation requests that intervene in game flow triggering particular events on the GE side.

Seen from the SE perspective, the Story Engine Interface (SEI) provides a link to the GE. Secondly, the SE builds an interface to the LE and passes progress information in both directions via the SE as interface.

![Diagram](Image)

*Figure 12.* The Story Engine (Narration Controller) as driving component for the 80Days framework. The Game Engine represents one Player from the Story Engine perspective.

External modules

External Modules are software modules that cover all non-core features, some of them are derived from ELEKTRA, like the Character Engine for example, others have been created from scratch in this project, like the Terrain Engine for example.
• **Physics Engine**: it provides runtime rigid body simulation and collision detection. It’s an external module based on the Open Source ODE Physics Engine, used successfully in many free and commercial products.

• **Dialog Manager** that provides data structures and procedures to manage and visualize character dialogs.

• **Trigger manager** that provides structure and functionalities to manage, load, and execute XML triggers, responsible for the gameplay flow.

• **Character Physics** that manages character game entities using capsule and other physics structures used in game development.

• **Terrain Engine**: responsible for terrain data retrieval, management and rendering.

• **GUI System**: it provides 2D User interface functionalities inside the 3D rendered scene, it’s based on the CEGUI third party library.

• **Cutscene System**: responsible for non-interactive 3D game situations and cinematics.

• **Network System**: responsible for network communication and multiplayer game situations.

• **Sound System**: responsible for audio output management.

• **Character Engine**: responsible for character animations and speech synchronization.

The Terrain Engine, specifically developed for 80days, has following features:

• Special tools were created to use real geographic data for 80days; these tools convert common Geographic Information System data into a format that could be used directly by the terrain engine.

• The setup of the engine allows for easy changing of terrains for different scenarios.

• The terrain rendering supports tiled texture to represent very large arrays; this solves problems with older graphics hardware that had strict limits on texture sizes.
Multiple sets of 3D objects can be added and placed onto the terrain.

One of the most important features of ELEKTRA in terms of creating emotions between the player and the learning game has been the usage of non-playing characters (NPCs) interacting with the player and carrying on the game experience. This characteristic has brought very good results in terms of effectiveness of the learning experience in the 80Days project. Technically speaking, the usage of NPCs is made possible by means of a Character Engine that is, as seen above, responsible for character animations and speech synchronization.

The Character Engine in 80days has following features:

- Characters support the semi para-social phenomena. This serves mainly the immersion feature the learning game should have.

- Autonomous behavior of the Character for an easy updating requested by developing phases with new game situations. This includes an evolved procedural animation system and an acceptance of programmed storytelling commands which may be formalized like in the PhD thesis of Caitlin Kelleher (2007). This feature helped to enhance the cost effectiveness of the project by using chunks of animations joined together to form bigger animations instead of being necessary to develop a great number of single bigger animations.

- High usability to generate and configure characters for the production pipeline.

- Handling the dialogues mainly between player’s character and NPCs.

- Executing behavior due to gameplay and adaptive learning elements.

**Development Pipeline**

A brief description of the development pipeline that has been set up during the 80Days project is presented here. The philosophy of 80Days is to strictly separate story creation and content production: Content (assets, objects) is created within familiar additional tools (standalone applications) such as 3DS and Maya for 3D or Photoshop for 2D and imported into the 80Days framework via ICML, the Story Editor and
Scene Editor. Figure 13 outlines the development pipeline with its actors, software modules, assets and dependencies/relationships. Let's now see how things work.

Starting from the top left corner, 3D artists produce all 3D assets using a 3D Authoring Software. In this case they have used Autodesk Maya coupled with the Radon Labs Nebula Toolkit 2.0 for Maya that allow Maya to export scenes and models compliant with Nebula2 specifications. 3D Assets are exported as files inside a game specific folder structure, these files are structured as a specific Nebula 2 engine file.
Developing Educational Games

format like .n2, .n3d2, .nanim2. Similarly other assets have been created by specific professionals:

- 2D assets have been produced by 2D artists using 2D authoring software like Adobe Photoshop, Adobe Illustrator or The Gimp. Usually 2D assets are image files in standard formats like Portable Network Graphics (.png), JPEG (.jpg), TARGA (.tga) that have been used as textures or 2d GUI pieces of artwork. When required we also used animated video contents like MPEG files.

- Audio assets, like sound effects and background music, have been produced by Sound artists through the usage of Sound Authoring tools like Propellerhead Reason or Sony Soundforge or The Audacity Team Audacity. Usually audio assets are media files in standard formats like mp3 (.mp3), ogg vorbis (.ogg) or wave (.wav) files.

Like 3D assets both 2D and Audio Assets have been placed in a specific folder structure to let GE loading them. Assets are the basic blocks to be mounted inside the game, when they are produced, they can be mounted with specific tools and scripts that describes how GE handles them and how assets behaves. 3D assets, in particular, are managed by Script Programmer or Level Designers using the Scene Editor so to be mounted in a scene that GE can understand and reproduce. In order to allow the GE to use those assets, they are instantiated and placed inside the 3D scene, labeled with a name or ID and coupled with game specific properties and attributes.

Script Programmers, using external script files, can access tagged scene assets and program specific behavior using the power of a High Level Scripting Language that exposes a complete set of game specific functionalities.

Using a scripting language like Python allows the creation of gameplay and learning situations by combining different type of assets and game elements in a fast and reliable way. In parallel to game scene assets development and mounting, also Story and Learning bases have been developed and linked to game scenes:

- Authors could define story flow using the Story Editor that produces input for the SE, responsible for the story management during the game experience. Through Story Editor it has been possible to define Macro story flow. Story editor ex-
ported ICML files that were interpreted/executed by SE at runtime. Very briefly we can say that SE tells GE if and how the story should be modified depending mainly from user’s activities.

- Learning experts wrote the knowledge base that is the starting point for the learning and skills assessment. This knowledge base is one of the inputs of the LE and it has been used to perform calculations about recommended learning interventions in the game experience. Knowledge base has been made consistently with learning requirements of the specific story / game situations.

Another important phase of the production pipeline has been the work of core programmers that are responsible for the development of three main engines of the system:

- GE is based on Nebula 2 Rendering Engine and Mangalore Game Framework, both developed in C++ as part of a bigger framework that comprises other external modules. GE core programmers have expanded functionalities of the ELEKTRA base framework to fulfill 80Days game and learning design requirements. Another important task has been the development/enhancement of the Scene Editor in order to implement game and learning design requirements and to allow Scene Editor working together with Story Editor and other tools that form the 80Days Toolset.

- The SE is developed as a .dll (Dynamic Linked Library) made in C++ language, so it has been quite comfortable to integrate it inside the GE framework. SE manages story flow receiving LE inputs and applying rules written with Story Editor.

- The LE has been developed with Sun Java language, and it runs within a virtual machine. Communication with SE is done via a TCP/IP protocol. LE will receive inputs from the SE and from the GE (through the SE).
Balancing on a High Wire: Adaptivity Key Factor of Future Learning Games

Christina M. Steiner, Michael D. Kickmeier-Rust, Elke Mattheiss, Stefan Göbel, and Dietrich Albert

With the research on game-based learning in the last years computer games have reached the status of being acknowledged as having a place in learning. Educational games have the potential to support learning inside and outside the classroom and to make learning fun. It would, however, be rather naïve to believe that all educational games are effective, and that all educational games are suitable for all learners (e.g., Van Eck, 2006). The design and development of successful educational games is highly challenging and most educational games are no match for their commercial counterparts. Aside from an appealing story and game design, an educational game needs to incorporate sound learning and didactic design. Game design and learning design need to be meaningfully harmonised and tied together and may not stay side by side unconnectedly. There are a series of well-established psychological and pedagogical principles, theories, and models on cognition, learning, and motivation that can help in understanding how games can be used best for learning.

Csikszentmihalyi’s concept of ‘flow’ (1990), for example, is frequently discussed in the context of educational games and instructional design, in general (e.g., Chan & Ahern, 1999; Shernoff, Csikszentmihalyi, Schneider, & Shernoff, 2003). Flow can be described as an intrinsically enjoyable state of deep absorption in an activity. It is a state of engagement characterized by the culmination of concentration, interest, and enjoyment. Good educational games should promote flow and avoid anything that forces the learner to step out and interrupt this experience. Traditional learning or assessment activities, like textbook reading or intermediate learning tests, would have a highly disruptive character during the game experience and are therefore not suited for educational games. Rather, learning and assessment need to be more subtly integrated in the game such that they are an integral part of the story and
the game world. This leads to another psycho-pedagogical approach commonly mentioned in the context of educational games: the principle of situated cognition and learning (Brown, Collins, & Duguid, 1989; Lave & Wenger, 1991; Van Eck & Dempsey, 2002). This approach emphasizes the connection between knowing and doing and the importance of realizing learning in meaningful contexts. In an effective educational game, learning should take place in (to the game) meaningful contexts. Learning is related to the game environment in which the knowledge is acquired and demonstrated. What is learned needs to be relevant in the game and practiced within the game context.

Another aspect that is important in educational games refers to challenge. An educational game needs to feature an appropriate level of challenge in order to be motivating and effective. Games that exceed or fall behind the capacity of players are not engaging and are likely to forfeit them. This aspect is highly relevant from a motivational point of view. An appropriate balance between perceived challenges and perceived skills is a precondition for flow (Czikszenmtihalyi, Abuhamdeh, & Nakamura, 2005). Challenge is also one of four key factors of computer games (Lepper & Malone, 1987, Malone, 1981) and is defined to be provided with appropriate levels of difficulty, progressive difficulty levels, and goals that are meaningful to the learner. Balancing challenge means creating cognitive disequilibrium in the sense of Piaget's (1952) theory of cognitive development.

Individual learners may, however, largely differ in their knowledge and competence, as well as in their preferences and motivational constitution, they bring along when facing an educational game. In order to keep an appropriate balance of challenge and motivation and to realise individually meaningful learning and gaming experiences, the issue of adaptation to the needs and characteristics of the individual learner becomes relevant (e.g., Brusilovsky, 1996, 1999). With the application of sound adaptation technologies that are grounded on well-established psycho-pedagogical foundations, educational games can be established as an approach to teaching and learning that is suitable for all learners. Adaptation is a key factor of future learning games that can help in creating educational games that are equally suited for different preferences and characteristics through the provision of personalised game experiences. As a result, adaptivity can contribute in further advancing educational games to educationally sound learning tools.
This chapter presents a comprehensive framework for intelligent adaptation in digital educational games. It grounds on a well-founded scientific basis of psychological and pedagogical theories and incorporates cognitive and motivational aspects, as well as narrative theory into a higher-level psycho pedagogical framework for competence development and motivation.

Competence-Based Adaptation

A Theoretical Basis for Competence-Based Adaptation

In traditional virtual learning environments personalization to the knowledge or competence of a learner may be realized through adaptive selection or sequencing of learning objects, adaptive navigation support, and adaptive problems solving support (e.g., Brusilovsky, 1996; De Bra, 2008). These conventional adaptation techniques are applicable in learning games only to a limited extent and need to be adapted in order to fit this application context. An educational game will, for example, normally not have a classical navigation menu for which navigations support could be provided. Moreover, in order to tailor the learning experience to the individual, information about a learner’s current knowledge needs to be gathered. Due to the requisite of embedding educational activities in the game, as outlined in the introduction of this chapter, traditional assessments and knowledge tests can hardly be realized in an educational game, either. To realize personalization to the knowledge or competence of a learner and, thus, to provide the appropriate level of challenge, a game needs more subtle ways of assessment providing continuous input and information about the learner. Adaptation of learning based on this information needs to be realized in terms of adaptation techniques that are plausible in an educational game context, e.g., through appropriate feedback and interventions.

Competence-based Knowledge Space Theory (CbKST; Albert & Lukas, 1999; Heller, Steiner, Hockemeyer, & Albert, 2006) constitutes a prominent approach and sound theoretical basis for intelligent, competence-based adaptation in technology-enhanced learning. CbKST constitutes an extension of the purely behavioristic Knowledge Space Theory for representing domain and learner knowledge (Doignon & Fal magne, 1985, 1999). A domain of knowledge is represented by a set of assessment problems; the subset of problems a person is capable of
solving makes up the knowledge state of this individual. A knowledge
domain is structured in terms of prerequisite relationships capturing
dependencies among the problems. Consequently, not all subsets of
problems will be potential knowledge states that are expected to be
observable. The prerequisite relation establishes a quasi-order on the
set of problems. The collection of knowledge states corresponding to
the prerequisite relation builds the knowledge structure, which also
contains the naive knowledge state (covering no problem) and the ex-
pert knowledge state (covering the whole set of problems). The knowl-
dge states are naturally ordered by set inclusion and give rise to a
number of meaningful learning paths. Thus, given the knowledge state
of a learner, a knowledge structure provides useful information, which
learning content should be presented next, but also which previously
learned material should be reviewed (Falmagne, Cosyn, Doignon, &
Thiery, 2006). Furthermore, a knowledge structure can be exploited for
realizing an adaptive knowledge assessment, i.e., an assessment proce-
dure that efficiently determines the current knowledge state of a learner
by presenting him/her with only a subset of problems (Dowling &
Hockemeyer, 2001; Falmagne et al., 2006). This can be realized by ex-
ploring the structure inherent to the knowledge domain and selecting
problems depending on previous answers of the learner. A knowledge
structure can, thus, build the basis for creating personalised learning
experiences by tailoring the learning path to the learner’s knowledge
and aiming in closing the gap between current knowledge state and the
desired knowledge state corresponding to the learning goal (Falmagne,
1993). In this way, Knowledge Space Theory constitutes a powerful
basis for realizing adaptive system behavior in the context of technol-
ogy-enhanced learning, allowing a tailoring of the learning process to
the learner’s knowledge, and has been successfully implemented in
several e-learning environments (Albert, Hockemeyer, & Wesiak, 2002).

Competence-based extensions of Knowledge Space Theory tried to
incorporate competencies or skills into the theoretical framework and
thus, to separate observable performance and underlying competence
(Doignon, 1994; Düntsch & Gediga, 1995; Korossy, 1997; Albert &
Held, 1994; Hockemeyer, 2003; Hockemeyer, Conlan, Wade, & Albert,
2003; Heller et al., 2006). These approaches aim at theoretically explain-
ing the observed behavior by considering underlying cognitive con-
structs in terms of skills. The basic idea of these approaches is to as-
sume a basic set of skills or elementary competencies that describe
abilities required for solving problems or taught by learning objects of a
particular knowledge domain. The competence state of an individual is the collection of skills that he or she possesses. It is not directly observable but can be uncovered based on observable solution behavior. Taking into account prerequisite relationships among skills, a competence structure can be established in analogy to a knowledge structure (Korossy, 1997, 1999). A competence structure constitutes the collection of all possible competence states that correspond to a prerequisite relation defined on the set of skills. By assigning to each problem the competence state(s) that is (are) sufficient for solving it (or, respectively, to each learning object the skills that it teaches), a knowledge structure on the set of problems (learning objects, respectively) is induced. This assignment of skills allows for deciding which learning objects are to be presented next, given a certain competence state, and thus builds the basis for generating personalized learning paths (Heller et al., 2006). CbKST has further nurtured research and development towards intelligent cognitive systems adapting to learners current competence state (Conlan, O'Keeffe, Hampson, & Heller, 2006).

Originally, CbKST has been applied in the context of traditional technology-enhanced learning systems for adaptive assessment and adaptive navigation support (for an overview see Albert et al., 2002). In recent years this theoretical framework has been elaborated and applied in the context of digital educational games (e.g., Kickmeier-Rust & Albert, 2008; Kickmeier-Rust, Göbel, & Albert, 2008; Kickmeier-Rust, Hockemeyer, Albert, & Augustin, 2008). These elaborations basically distinguish between two different types of adaptation that are tailored to learning environments with large degrees of freedom, like educational games, namely adaptivity on a macro level and adaptivity on a micro level.

**Macro Adaptivity: Pinpointing Challenge and Learning Sequence**

Adaptively providing a learner with suitable learning objects is a major mechanism of existing approaches to adaptive e-Learning. This is the usual way competence and knowledge structures have been used for personalization in traditional technology-enhanced learning (e.g., Conlan et al., 2006). Given the knowledge or competence state of a learner, suitable next steps for learning can be identified by consulting the knowledge and competence structure. In this way, learning paths can be realized that take into account the prerequisites among knowledge ele-
ments of a domain and meaningfully guide the learner on his/her way from an inferior knowledge or competence state to the desired state corresponding to the learning goal.

Macro adaptivity actually adopts this traditional technique of adaptation in terms of selecting and ordering learning activities to learning games. This may consist in the selection of learning situations in the game that fit the knowledge and competence of the learner. Another example would be the presentation or recommendation of meaningful options in the game, e.g., paths that lead to learning situations that fit the learner’s needs. Hence, macro adaptivity refers to adaptation on the level of learning objects or activities and can be realized in terms of multiple paths through the game environment. In an educational game macro adaptation is, therefore, typically performed between learning units, i.e., game situations or missions. Macro adaptivity deals with the adaptive sequencing of alternative learning situations in an educational game. Generally, adaptation on a macro level is based on a fixed learner model (e.g., traits) or adaptation model (e.g., pedagogical implications) and on typical (knowledge) assessments (via test items).

In the context of educational games the principle of macro adaptivity has, however, some drawbacks and needs to be complemented by further considerations in order to ensure to get the best out of personalization. Firstly, pure macro adaptation by following the theoretical structures in terms of CbKST means reordering learning objects and maybe skipping specific ones. This may contradict the narrative of the educational game. Skipping a specific situation might lead to corrupting the red thread through the game’s story and end up in an incomplete or implausible sequence of situations and story elements. As a result, macro adaptivity needs to be advanced in order to meld competence-based adaptation and adaptive storytelling, in order to give rise to sequences of meaningful learning as well as story elements (see section ‘Getting storytelling on board: Linking story and knowledge’ below). Moreover, as macro adaptation is realized only between learning units, it will lead to a limited number of interventions. As an educational game will likely consist in complex, explorative situations with large degrees of freedom, more regular support is needed. Moreover, knowledge assessment as known from conventional learning environments with computer-supported test items, even if it is realized in terms of adaptive assessment procedures, cannot be realized in an educational game – as it would mean a disruption of the game’s flow. This necessi-
Balancing on a High Wire

states the extension of the realm of adaptation techniques in learning games to the level of micro adaptivity (see section ‘Micro adaptivity: Taking a back seat with sensitive antenna’).

Getting Storytelling on Board: Linking Story and Knowledge

In a game context, the competence-based, cognitive models for adaptive educational technologies need to be harmonized with adaptive storytelling in order to come up to game experiences that are meaningful from a didactic as well as from a narrative point of view. From a didactic point of view, for instance, on the one hand rather novice learners will need to be presented with specific or additional learning situations, while those with more experience and pre-knowledge might skip certain situations. From a storytelling perspective, on the other hand, there might be game scenes or even learning situations – possibly in alternative variants (i.e., game paces) – that each learner has to go through in order to fit or follow the story.

Adaptive storytelling means to tailor a game’s storyline and story pacing to the individual learner. Adaptive storytelling may significantly support personalized learning experiences in educational games by automatically adapting the story to individual preferences of the learner. In this way it is possible to realize explorative learning processes in game and story contexts that are personally relevant to the learner and, thus, support approaches to constructivist and situated learning.

The sound linkage between story and knowledge to form appropriate game paths is a crucial aspect. On the basis of a pool of given story elements and game-based narrative learning objects, a suitable storyline needs to be established. This storyline needs to keep the red thread through the story, on the one hand, and has to be in line with the learner’s current knowledge and learning progress, on the other hand.

To find educationally meaningful, yet plausible and exciting storylines, formalisms and rules are required. A Markovian storytelling framework has been defined that merges story plots with competence structures in order to establish game paths through an educational game. This framework models the game elements of a game and the possible story lines, which provide a compatible sequence of game elements and are appropriate for certain user groups. A user may be characterized by an ‘inner state’ covering a list of attributes, like the current competence
state, as well as gender, age, etc. At each stage of the game for a learner in a certain ‘inner state’ possible next game elements can be selected.

By getting adaptive storytelling on board of macro adaptivity, personalization of educational game experiences can be extended. Consequently, macro adaptivity comprises competence-based as well as story-based adaptation. In addition to adaptive curriculum sequencing, as done in traditional learning systems, story elements or game paces are selected that correspond to the preferences and characteristics of the learner. Sensation seeking, gender, interests, video game preferences and habits are possible adaptation criteria for this. Adaptation may thereby consist of selecting game paces in accordance with learners’ characteristics. Learners who are high-sensation seekers and/or frequently play action and fighting video games might, for instance, be presented with a fast and hectic version of a game with competitors or persecutors. In contrast, learners who are low-sensation seekers and/or hardly play any video games would play a more relaxed and explorative version of the game without any time pressure. Moreover, if the cognitive model suggests several meaningful next steps of learning, a learning situation may be selected in terms of adaptive storytelling that features story elements matching with the learner’s interests or preferences. Adaptive storytelling may not only be realized based on static information about the user (i.e., traits) that is, for example, gathered in the intro screen of an educational game, but may also serve adaptation to more dynamic states like motivation (see also section ‘Adapting to motivation on a macro level’).

Further details on the conceptual work on adaptive and interactive storytelling in digital educational games can be found in the following chapter “Adaptive Digital Storytelling for Digital Educational Games” (p. 89).

**Micro Adaptivity: Taking a Back Seat with Sensitive Antenna**

Micro adaptivity takes place within learning tasks or situations. This approach to adaptation in educational games has been elaborated in order to allow intelligent adaptation that does not compromise game flow and immersion (Kickmeier-Rust & Albert, 2010). Micro adaptivity attempts to monitor learning progress by interpreting a learner’s actions in the game in terms of available and lacking skills. On the basis of this information non-invasive adaptive interventions are possible that pro-
vide the learner support and guidance with hints or feedback. Micro adaptivity actually constitutes a form of adaptive problem solving support adjusted for the personalization in complex learning situations with large freedom, in particular educational games.

**Non-invasive competence monitoring**

In order to be able to guide and support the learner according to his/her needs, the current competence state and learning progress, as well as probable misconceptions and competence gaps need to be assessed. The CbKST framework is a basis for effective adaptive skill assessment carried out in form of explicit testing procedures. In a game context, however this assessment needs to be realized in a more subtle manner that does not force the learner to leave the game situation and that, therefore, keeps the game flow. Micro-level assessment allows non-invasive competence monitoring through problem solving that is embedded into the game context and narrative. For realizing this, cognitive theories of problem solving (Newell & Simon, 1972; Newell, 1990) have been used and incorporated in CbKST in order to come up to a formal psychological model for interpreting learner behavior and actions within learning and assessment situations of an educational game (Augustin, Hockemeyer, Kickmeier-Rust, & Albert, in press).

A problem solving situation in an educational game can be modeled in terms of a problem space representing all possible problem solution states, i.e., the initial and the goal state of the problem as well as the intermediate states. Problem states can be represented by the consideration of objects relevant for a problem and transition rules specifying the possible changes from one state to another. Each problem solving state can be associated with a set of possible or admissible actions. This concept of a problem space can be connected with the competence structure of a domain to realize non-invasive competence monitoring. To this end, for a problem solving process the possible actions in terms of manipulations of the objects in the situations are linked to the underlying skills that a learner has available or lacks. To facilitate this linkage, the location and properties of an object can be described by means of position categories (Kickmeier-Rust & Albert, 2010).

An example from an educational game would be a learning situation on natural hazards that allows exploring flood prevention measures and causes in a terra forming simulation. One task and learning objective in this situation might be to reduce the flood risk level through human
interventions in the landscape. Other tasks could be to increase the risk level or to explore the natural causes of a flood. Possible actions in this situation would consist in manipulating objects in terms of selecting and setting different kinds of human interventions in the simulation playground. This could, for instance, be to plant trees, to build a dike, to straighten a river, etc. This means the learner can perform different actions to achieve the goal. The aim of micro level assessment is, in the first instance, to assign a problem solution state from the problem space to each action. This mapping is done by classifying actions according to a set of rules. Given the goal of reducing the flood risk, for example, clearing trees would increase the distance to the goal state. The second step is to assign a set of available and a set of lacking skills to each action and problem solution state. For example, planting trees would indicate that the learner knows that reforestation is a natural flood prevention measure – the respective skill can be assumed to be available. Building a dike far away from the river would indicate that the learner knows that dikes can contribute to flood prevention, but he/she does not know about the mechanics behind and contextual conditions.

As naturally a single observation is not very convincing, the micro adaptive assessment of CbKST usually applies a probabilistic approach. From the information on available and lacking skills based on an action, assumptions on the competence states and their probabilities are derived. One method of doing this is to increase the probabilities of all competence states, including the skills that are assumed to be available based on the learner action, and to decrease the probabilities of all competence states that cover skills that can be assumed as unavailable. With each learner action the probability distribution over the competence states can be updated. In this way, a more and more focused and clear picture of a learner's competence can be established (cf. p. 112 for technical details). At the end of this procedure stands a more or less well-founded assumption about the skills the learners have, the skills they don’t have, and their position in the problem solving process.

A crucial next step of micro level adaptation is now to have the game appropriately react on the basis of this competence monitoring. This refers to the provision of meaningful psycho-pedagogical interventions and feedback in an educational game.
Non-invasive guidance

Pedagogical interventions and feedback are an essential and integral element of the educational process. They guide and support the learning process, they make the learner aware of his/her progress and possible deviations from a planned learning path and goal, they provide the learner with appropriate information and direct the learner’s view on important information, and they stimulate reflection. In conventional educational settings, interventions come from teachers, fellow students, friends, or from one’s self. In the context of technology-enhanced learning, interventions and feedback may be provided by tutors (e.g., via discussion forum) or automatically by the system (e.g., through pop-up messages in the learning interface of conventional e-learning systems or through non-player characters in an educational computer game). The effectiveness of interventions and feedback in computer-based learning has been demonstrated in a variety of studies (e.g., Azevedo & Bernard, 1995; Moreno, 2004; Tan, Biswas & Schwartz, 2006). Mechanisms of feedback and intervention support several cognitive mechanisms for learners - they allow indication of a gap between current and desired performance level and may enhance motivation and task strategies; they are able to reduce learners’ cognitive load, especially for novice learners; and they may provide information that is useful for correcting inappropriate task strategies, errors, and misconceptions (Shute, 2008).

Feedback constitutes an important aspect of instructional design and several authors have tried to define general guidelines for feedback in a learning context (e.g., Economides, 2006; Narciss & Huth, 2004; Pivec, Koubek, & Dondi, 2004; Shute, 2008). Such guidelines may, for example, refer to the timing of feedback, to the learners’ level of achievement, to feedback types, or the form of their presentation. In accordance with such guidelines, research also tried to systematically describe and categorize types of feedback and interventions (e.g., Mason & Bruning, 2001; Narciss & Huth, 2004; Shute, 2008), for example, according to the timing of feedback (e.g., Dempsey & Wager, 1988), the complexity of feedback in terms of the extent of information provided (Mory, 2004), the delivery mode (Vasilyeva, Puuronen, Pechenizkiy, & Räsänen, 2007), or its ‘mind’ dimension (Economides, 2006). Advances in technology and technology-enhanced learning provide a wide spectrum of opportunities for providing effective feedback to learners; feedback that can be tailored to the individual situation and the individ-
ual’s current needs. It could be shown that adaptive computer-based learning systems provide effective feedback (Azevedo & Bernard, 1995).

Effective feedback and interventions require the system’s ability to monitor and model behavior and characteristics of learners in order to provide an appropriate reaction. An appropriate response to the learner’s behaviour plays also a crucial role in the personalization of digital educational games. The information from micro-level assessment in an educational game gives rise to the realization of psycho-pedagogically meaningful intervention mechanisms that provide personalized and non-invasive guidance and support to the learner (Kickmeier-Rust, Hockemeyer, et al., 2008; Kickmeier-Rust, Marte, Linek, Lalonde, & Albert, 2008; Kickmeier-Rust, Steiner, & Albert, 2009). From a micro perspective, i.e., within learning situations of an educational game, this primarily refers to adaptive, didactic interventions such as the provision of hints in problem solving processes, feedback about learning progress, or adjustment of game parameters (e.g., difficulty). The realization of interventions in educational games is, however, challenging as they hold a disruptive potential. Like the assessment procedures themselves, also the system’s response to the learner must be – of course – educationally meaningful and suitable, but also non-invasive and strongly embedded in the gaming context without corrupting immersion and flow. To equip the game with a set of potential interventions or feedback that can be triggered in an intelligent and systematic way, a menu of psycho-pedagogically inspired adaptive intervention categories and types has been elaborated (Steiner, Kickmeier-Rust, Mattheiss, & Albert, 2009). This allows realizing competence-based adaptation on the micro level of a learning game in a psycho-pedagogically meaningful way. Non-invasive interventions embedded in the game may consist of hints, suggestions, warnings, or feedback given by a non-player character or in modifications in the game display or interface. These interventions strive to enhance cognitive abilities and knowledge acquisition and to support the learner adaptively according to his/her behavior and underlying available or lacking skills. Depending on the results of the non-invasive assessment procedures an appropriate intervention can be triggered that fits the learner’s current competence and the situation. On principle, we distinguish two categories of interventions that may apply in this case: cognitive and meta-cognitive interventions. Both categories cover a variety of concrete intervention types.
Figure 14. Self-regulated learning and the involved metacognitive processes.

Meta-cognitive interventions refer to self-reflective processes or knowing about one’s own knowledge. Meta-cognitive processes are central to meaningful learning and are an integral part of the self-regulated learning process (compare Figure 14; Zimmerman, 1990, 2002; Puustinen & Pulkkinen, 2001). The self-regulated learning paradigm is an appropriate approach for characterizing learning processes in educational games, as playing and self-regulated learning are closely related (Rieber, 1996).

Meta-cognition consists in introspections that are manifested as plans, strategies, reflections, self-control, self-monitoring, self-evaluation. These introspections are addressed and shall be supported by meta-cognitive interventions. Instead or before providing very concrete hints and pieces of information, such interventions shall prompt learners to focus and reflect on the task and their cognitive processes. Learners should be encouraged to reflect on their own competence, thinking, and problem solving behavior. This can be accomplished by different types of interventions that may be of different nature and may provoke different meta-cognitive actions.

A straightforward way of promoting reflection on one’s performance and competence is by directly telling the learner to do so – e.g., “Think it over again. Reflect on the goal of our task.” This is called an introspection intervention. Another way for provoking or initiating meta-cognitive processes is to pose a question addressing strategic planning, monitoring, or self-evaluation – i.e., meta-cognitive question. In this way, it shall be ensured that the learner understands a problem, consciously thinks about possible solution strategies, establishes connections and analogies...
to prior knowledge and problem solving, and mindfully monitors and controls his/her thoughts and actions. Another type of meta-cognitive intervention is to concretely ask the learner to carry out a task involving and requiring meta-cognition – i.e., meta-cognitive task. A request to write a diary of one’s thoughts or to sketch a plan of how to approach the situation etc. would be examples of this intervention type.

The above mentioned intervention types may be applied in each phase of the self-regulated learning process. As a result, meta-cognitive interventions may be further differentiated according to the learning phase in which they are provided – in forethought, performance, and reflection interventions. Forethought interventions try to promote or support planning and preparation of the problem solving process, e.g., through a question, such as “What do you think are feasible strategies for tackling this task?”, posed by a non-player character. Performance interventions are provided during the problem-solving process in order to prompt self-observation and -monitoring. This may, for instance, consist in the provision of a meta-cognitive task requesting the learner to record important occurrences or interim outcomes in a logbook. Reflection interventions shall foster reflection on problem solving performance and acquired skills after the problem solving process, e.g., by a hint “Think about what you have learned in this task and how it can help us in our future adventures.” Consequently, meta-cognitive interventions can be characterized on two orthogonal dimensions according to their nature and the learning phase in which they occur (compare Figure 15). This scheme of intervention types makes clear the theoretical background of these interventions and may serve as a valuable basis for actually designing concrete interventions in an educational game. In practice, of course, the line between the different types will be somewhat blurred.

In certain situations or at a certain point in time it may not suffice to prompt learners to think about their own thinking and cognitive processes. If the non-invasive competence monitoring, for example, shows that a learner seems to get stuck in problem solving and/or features significant competence gaps, he/she will need to get further support in order to successfully acquire the required skills and to accomplish the task. This is where cognitive interventions come into play, which directly focus on the available or lacking skills of the learner. This means, the learning objectives are directly targeted and competence acquisition shall be enhanced.
Cognitive interventions may have different purposes and targets. Moreover, cognitive interventions differ in the concreteness of information (or depth of information) they provide. An intervention may, for instance, provide a rather subtle hint; another one may point out very concrete information. Figure 16 gives an overview of the types of cognitive interventions that we distinguish. These different types have been specified on the basis of the theoretical framework of CbKST and...
its understanding of meaningful learning sequences by following competence learning structures (e.g., Heller et al., 2006). Principles of constructivist learning, such as cognitive apprenticeship (Collins, Brown, & Newman, 1989; Schroeder & Spannagel, 2006), have also been taken into account. In addition, theoretically founded principles for the design of informative tutoring feedback (Narciss & Huth, 2004) have built a basis for the definition of the cognitive intervention types.

An intervention may address skills that, determined by the non-invasive assessment, the learner has available. Competence activation interventions are applied if, according to prior assessment outcomes, it can be assumed that a learner possesses the skills required for solving the problem, or at least a subset of them. If the interpretation of the learner actions in the problem solving process shows that the learner seems to nevertheless get stuck in the task, the necessary skills are interpreted as temporarily unavailable or ‘inactive’. With a competence activation intervention, these skills shall be stimulated and reactivated, which can be done by a reference to a specific skill of the learner or by posing a question or task that involves the same knowledge but in a different manifestation. While such type of intervention is rather subtle and tries to lead the learner to discovering that he/she possesses the necessary skills, competence explication interventions are much more concrete in the information they provide. These are interventions in the context of complex problem solving or simulation situations, where the learner can freely experiment and explore. If the non-invasive assessment comes to the conclusion that a learner has available or acquired certain skills due to the actions he/she takes, this intervention type makes explicit and reinforces the knowledge involved in the certain action. In this way, it can be ensured that the learner is aware of the knowledge elements underlying the correct actions.

Cognitive interventions may, of course, also address skills that have been identified as lacking in the learner’s current competence state. Interventions providing problem solving support constitute hints or recommendations for possible next steps in the problem solving process. This intervention type mainly aims at decreasing the distance between the present solution state and goal state. The respective interventions are, in general, rather subtle; instead of directly explicating a certain knowledge element, they aim at leading the learner to promising solution ways. Interventions may consist of vague indications or more concrete hints of possible next steps, the provision of information that is
relevant for but more or less independent from the problem solving context, the tip to revisit prior learning situations, the suggestion of analogies or reference to similar problems, etc. A very similar intervention type is a competence acquisition intervention. These interventions tend to be a bit more concrete in the information they provide and are more common in learning game situations that are not characterized by a highly complex problem solving task. Competence acquisition interventions are applied if the system concludes that the learner lacks certain skills. In this case, an intervention provides the required information. The most concrete and tangible way of providing information in case of lacking skills is through dissolving interventions. This intervention type is used if the learner was not able to show the required answer behavior within a reasonable number of actions. By finally providing the correct solution to a problem or task, these interventions ensure that the game and game flow can continue. Such intervention, of course, for didactical reasons might not be suitable for every task or situation.

Instead of directly addressing available or lacking skills, a cognitive intervention may also target the learning progress. Progression feedback consists of information about the progress in learning and competence development in the course of the game. This may be feedback on specific learner responses or actions, such as correctness, location of mistakes or explanations why a certain answer/action is not leading to the desired goal. The feedback may also be goal-directed in terms of providing information about the progress towards a desired goal. Progression feedback may be realised in terms of scoring mechanisms or game character development, and it has also a strong relevance for motivation. Although these interventions normally do not directly provide information on individual skills, they relate to them. Progression feedback as a cognitive intervention type is, therefore, distinguished from meta-cognitive reflection interventions.

Finally, a cognitive intervention may address ambiguous skills. This means skills on which the non-invasive assessment led to unclear results after a certain number of actions are targeted. Cognitive assessment interventions aim at gathering additional information on the respective skills by explicitly posing a question or problem testing them. These interventions, though, differ considerably from the usual and typically highly disruptive assessments applied in technology-enhanced learning, as they are strongly embedded into the game context and narrative in order to keep immersion and flow. This is achieved through interactive
dialogues with different answer options, which may not only refer to correct and incorrect responses but may also have a storytelling function and lead to different story strands depending on the learner’s choice.

As can be seen, this elaborated menu of interventions provides a rich basis for competence-based guidance and adaptation at a micro-level of a learning game. All interventions require a manifestation in form of game assets (e.g., a sound file with a specific sentence). The translation of the theoretical intervention types into concrete interventions for a particular game is a matter of learning game design. Thereby, certain interventions need to be defined specifically for skills, while other types can be designed more or less skill-independent. Correspondingly, the different intervention types can be differentiated with respect to their skill-specificity. Specific interventions specifically address certain skill or subset of skills – i.e., contain concrete information associated with the respective knowledge elements. These interventions are specified and selected with reference to the respective skill(s). Actually, such an intervention may refer to only one skill (e.g., skill on artificial lake as flood risk-decreasing measure) or to a group of skills (e.g., skills on flood risk-decreasing measures in general), depending on the granularity level of the information covered. Generic interventions are of a more general and reusable character, as they are not referring to a specific (set of) skill(s). These interventions may be designed in terms of an intervention pool, from which an intervention in a concrete situation can be selected. Such a pool of generic interventions may, however, be restricted to a certain type of learning/gaming situation (e.g., flying situation, map desk situation) in order to ensure the contextual appropriateness of the hint. While certain intervention types are clearly skill-specific (virtually all cognitive interventions), certain intervention types may optionally be designed as either specific or generic interventions (e.g., introspection intervention).

Depending on the game context and knowledge domain of a concrete learning game, of course, not all interventions might be possible or feasible to be realized. To unfold the full potential of the different intervention types, sound selection rules need to be defined such that given a certain situation and learner the most appropriate intervention is triggered. This means, to allow a meaningful provision of adaptive interventions, appropriate psycho-pedagogical rules need to be set up for their selection. The selection rules translate the theoretical consid-
erations on the intervention categories and types into instructions and specifications that can be interpreted by the system in the concrete learning game context. Selection rules will usually consist of meta-rules specifying the overall conditions of selecting adaptive interventions (for instance, threshold values for skill probabilities) and rules for the selection of the appropriate intervention type. For selecting a concrete intervention type, the history of probability updates of skills relevant in the problem solving process, as well as the history of already provided interventions, are taken into account. In general, it is good practice to specify selection rules that lead to a rather conservative and sparing triggering of interventions; usually no interventions are more beneficial than too many or inappropriate ones. Moreover, it needs to be taken into account that repeated or inadequate interventions due to misinterpretations of a situation can significantly compromise the learning game experience.

Finally, with the provision of adaptive interventions it has to be considered that certain intervention types will (optimally) change a learner’s competence state. Correspondingly, the provision of certain adaptive

![Diagram](image)

**Figure 17.** Overview and example of the adaptation process on a micro level.
interventions should trigger an update of skill probabilities. A cognitive intervention that provides information on one or several skills that have been diagnosed as lacking will actually lead to an increase of the probability that a learner has available the respective skills. In case of an assessment intervention, naturally, the skill probabilities are updated according to the learner’s answer.

To summarize the principles and processes of micro adaptivity, Figure 17 gives a comprehensive overview of the adaptation process and illustrates how non-invasive competence monitoring and guidance are interwoven. For better understanding, the figure provides an example illustrating the successive steps of micro adaptivity.

**Motivational Adaptation**

Computer games have a high popularity and motivational potential – and it is exactly this motivational benefit that is aimed to be exploited in learning games. Educational games can help to motivate learners to learn, as learning is incorporated in gameplay (Oblinger, 2004).

It is, however, important to properly understand the processes and constructs underlying motivation of learners in the context of an educational game in order to successfully make it a motivating game. Motivation is essential for effective learning and extensive research has been carried out in order to understand, model, and support motivation to learn (e.g., Astleitner & Wiesner, 2004; Meece, Anderman & Anderman, 2005; Schunk & Pajares, 2002; Wigfield & Eccles, 2000). Motivation means the orientation and energization of behavior towards a positively perceived goal or target state. Learning motivation refers to the structures and processes that explain learning and learning activities and their effects, like activation, orientation, and persistence (e.g., Krapp, 1993). Motivation, thereby, has to be differentiated from motives. Motives are person specific dispositions (traits); they are relatively stable and situation independent behavioral tendencies. Motivation, in contrast, is a dynamic and momentary state that results from the interaction of different person- and situation-specific variables.

This dynamic nature is important to keep in mind when aiming in realizing motivationally sound learning games. It is not enough to have trust that a learner who starts a learning game with high motivation will automatically keep this motivation until the end of the game. A learner may for many reasons get disengaged quickly and possibly even stop
playing the game. Reasons for disengagement may be an inappropriate level of challenge of the game or disruptive traditional learning or assessment activities that destroy flow experience. In this regard, competence-based adaptation, as outlined in the previous sections, can contribute a lot to keep motivation. The aspect of motivation, however, should be considered more comprehensively and explicitly in a learning game context.

Motivation may be understood as a general and comprehensive concept of learner engagement. In addition, different aspects of motivation may be differentiated. Keller (1987) systematically considers motivation to learn in his model of instructional design and distinguishes four different components or conditions that have to be considered when aiming at people becoming and remaining motivated: attention, confidence, relevance, and satisfaction. These four components may be seen as different aspects of motivation that make up and may be used to model and represent a learner's current motivational state. Attention refers to the degree a learner directs perception and cognition to the situation or task in question. As an appropriate level of attention is a prerequisite for learning, it is an important aspect of motivation. Attention needs to be obtained and sustained, optimally by taking into account sensation-seeking needs and traits (Zuckerman, 1971), in order to arouse curiosity without over-stimulation. Relevance relates to the feeling of perceived meaningfulness and mindfulness of a certain task or situation to the learner. The level of relevance will be high for a learner whose interests or preferences are met with a certain instruction or if he/she understands that a certain learning task is relevant and applicable in real life. Confidence relates to the learner's self-efficacy and expectancy of being able to accomplish a certain task successfully. The level of confidence influences persistence and accomplishment in a learning task. It is, therefore, important to address this aspect of motivation by giving learners the impression and possibility that they are able to achieve some level of success if effort is exerted. Satisfaction, as the fourth motivational aspect, refers to the feeling and attitude towards a task and one's own accomplishments. If the task and outcome was in some sense rewarding (intrinsically or extrinsically), people will feel good and more motivated.

Motivation should be sustained throughout the learning game through an appealing game design itself, through active involvement, and through interaction and feedback. To exploit the motivational potential
of a learning game at the best, in addition to motivationally appealing design, approaches towards motivational adaptation should be taken. Motivational adaptation will ensure keeping the learner engaged while gaming. For adaptation a system, of course, requires information and an understanding of a learner's current motivation. In the following sections the theoretical foundations for understanding and assessing motivation are presented. Subsequently, it is outlined in what way motivational adaptation may be realized in a learning game.

**Motivation in Learning Games**

The inclusion of motivational aspects in the design of educational systems - like learning games - is of increasing interest for researchers and developers. Instead of only assuming that games are motivating per se, it becomes more and more important to consciously design and create games in a way to enhance the players' engagement. For this purpose specific design principles can be used for realizing static motivating features, which are the same in each game attempt and for all players. Related guidelines specifying desirable game characteristics can be found, for instance, in Garris, Ahlers, and Driskell (2002), who elaborated on the six game dimensions on the basis of previous research work: fantasy, rules/goals, sensory stimuli, challenge, mystery, and control (e.g., Keller, 1987; Malone & Lepper, 1987).

Aside from general design guidelines for educational games, an additional opportunity to enhance motivation in game-based-learning is to provide the player with motivating interventions like specific encouraging statements. It seems useful to keep the player engaged by praising and giving elating feedback. It is, however, reasonable to adapt these or similar interventions to the needs of the player, because motivational interventions can easily be annoying and time wasting for a person who is already engaged. Song and Keller (2001) investigated the effectiveness of a motivationally adaptive computer-assisted instruction, which provided motivational strategies dependent on the assessed motivational state of learners. This instruction was compared to a motivationally minimized and a saturated version of the instruction, which included no and, respectively, all possible strategies. The adaptive version resulted to be superior regarding attention and overall motivation.

The first step to motivational adaptation is the capability of an educational system to recognize whether a person is lacking motivation or not. This motivational assessment is necessary to provide the player
with a specific motivational intervention in the right moment and to allow engaged people to play without any disturbance. Both, the assessment of motivation as well as motivational interventions based on the assessment outcomes should ground on a common basis and understanding of motivation. In an attempt to establish such a basis, Mattheiss, Kickmeier-Rust, Steiner and Albert (2009) devised an advanced model of motivation in game-based learning. The model is based on the expanded cognitive model of motivation to learn by Heckhausen and Heckhausen (2006), the input-process-outcome model by Garris et al. (2002) and the ARCS model of instructional design by Keller (1987, 2008). It is a fusion of these three approaches for application in the field of game-based learning, integrating the different motivational theories and considerations. An illustration of this model can be found in Figure 18.

**Figure 18. Advanced model of motivation in game-based learning** (adapted from Mattheiss et al., 2009).

Game-based learning includes four levels or aspects. Firstly, the overall process of game-based learning consists of a specific sequence of stages, which can have or not have an incentive on their own. These stages are represented by conditions, activities, and outcomes on the basis of the input-process-outcome model (Garris et al., 2002). Following the expanded model of motivation to learn (Heckhausen & Heckhausen, 2006), the stages can be seen as made up by certain components. These components are related through expectations. Finally, there exist motivational strategies, which are relevant mainly – but not exclusively – in a certain stage. The stages can be seen like three distinct parts, although
it has to be mentioned that a strict isolation is not purposeful in some cases and that there is a certain amount of overlap.

**Conditions**

The *conditions* of game-based learning include aspects, which are basically determined before the gaming takes place. These aspects refer to characteristics of the player, such as gender, age, personality, prior experiences, or interests, as well as to characteristics of the game and the learning content, like game type, game characteristics, knowledge type, etc. Considering the interaction of these two components – person and situation – is crucial for a successful game-based learning experience. Right before the educational game is played, the person will develop specific expectations about the upcoming activity, which are dependent on this person-situation-interaction. For example, a person who has positive prior experience with learning games will probably expect the announced learning game activity to be fun or/and useful. This will influence the initial user judgments when the game starts. A self-confident and experienced player will expect the playing activity to be easy and enter the game with an according activity expectation. The present conditions and expectations can form an incentive for the player, in terms of a high interest in the learning content or in games of the specific game genre. It is important to create positive conditions and expectations, as far as it is possible, to foster a high motivation when beginning to play the learning game.

Motivational strategies particularly useful for a motivational enhancement within this preparation phase for the educational game are related to attention and relevance (Keller, 2008). By means of an interesting game-design, arousing curiosity by incongruity, mystery, or unresolved problems, attention of the player can be attracted to the game. Relevance of the learning game can be realized through matching the game content, teaching style, and social organization to the learners’ goals, learning styles, motives (achievement, affiliation, or power), and experiences. This means already the incorporation adaptation on a macro level.

**Activities**

*Activities* relate to the player’s as well as the system’s actions within an educational game. This stage actually refers to the learning game and
the interaction with the game. It can be seen as a cycle of user judgments, followed by a certain user behavior or reaction, which causes a specific system feedback leading to a renewed judgment. The performed activity can have an incentive by itself if the gaming activity is experienced as fun or if the person enjoys the learning process or competence demonstration through the reaction-response-cycle. Together with the conditions the activity forms specific expectations about the outcome of the game. An experienced player, who performs well in the game, will expect to master the game and be pleased with this outcome. On the contrary, a consistently unsuccessful player will probably not expect to succeed at any time and may even quit the game at a certain point.

Motivational strategies during the gaming activity are strongly related to micro adaptation, in terms of a non-invasive system feedback to specific user actions. To motivate a user continuously in an educational game, it is crucial to strengthen confidence. A confident player will have positive expectations for success and, therefore, engage in the game. To support confidence, the player should experience success in the game and attribute this success to his/her own ability. This can be realized by providing the player with tasks which are not too difficult in order to allow success, but no tasks that are too easy in order to not bore the learner. As a result, the challenge should fit the ability of the learner. This balanced level of challenge is also advantageous for the appearance of a flow experience (Csikszentmihalyi, 1990), which is considered to be the ‘optimal’ experience. Another strategy to enhance motivation at the activity stage is through interventions related to attention. Besides an attention-catching game play, dependent on user characteristics like the sensation seeking tendency, micro adaptive interventions towards better concentration are crucial. If the player shows a lack of attention the system could, for instance, trigger a dramatic statement, sharp noise, or quiet pause. To sustain attention also strategies in terms of macro adaptation may be followed. This would mean, for example, to change the game pace and storyline.

**Outcomes**

*Outcomes* are the results of a game-based learning activity, which are ideally increased cognitive abilities, positive affect, and high motivation to continue the activity.
Ensuring satisfaction after playing is the most important motivational strategy at this stage. The general aim is that the learner leaves the learning game with a positive feeling about the game experience. A learner can be intrinsically rewarded and, thus, satisfied because of being proud of his/her performance and accomplishment in the game. A feeling of satisfaction may also be fostered by making clear the applicability of the knowledge acquired in a real life context. In addition, also extrinsic rewards like sweets or points may be used to enhance satisfaction.

**Assessing Motivation**

**Approaches to motivational assessment**

For assessing motivation, one can simply directly query persons on their current motivation (e.g., Song & Keller, 2001). This approach, however, is not feasible in the context of a learning game as it may be perceived as disruptive and even as compromising motivation. As a consequence, ways of non-invasively measuring a person’s motivation are of interest. When looking for suitable motivational assessment methods one comes across various related research work in the field of e-learning.

In terms of a non-intrusive assessment it is desirable to develop a methodology based on the interpretation of players’ behaviour that can be easily measured and recorded by the system and allows real-time adaptation. De Vicente and Pain (2002, 2003) established a set of motivation diagnosis rules for an intelligent tutoring system, based on a motivation model including motivational traits and states. For the derivation of these rules participants with tutoring and/or teaching experience had to watch a recorded interaction of a specific person with the system. The participants were informed about the persons’ value of the traits (control, challenge, independence, and fantasy). According to this information and the observed interaction, they had to continuously estimate the students’ value of the motivational states (confidence, sensory interest, cognitive interest, effort, and satisfaction) and verbalize the reasoning behind these judgements. From this data the researchers deduced rules for detecting a person’s motivational state while interacting with the system. The estimations of motivation mainly rely on a person’s performance (such as speed and quality of performance, quantity completed, and mouse movements). For
instance a non-random mouse movement is interpreted as an indicator for high attention and/or interest, quick performance suggests — independent of other behavioural characteristics — either confidence or a lack of interest, and a good performance is assumed as indicating high satisfaction.

Cocca and Weibelzahl (2007, 2009) used a similar method of letting experts rate a learner's motivational state. They deduced and identified behavioural patterns from log-files of a web interactive learning environment in order to draw conclusions as to whether the learner is engaged or not. The authors identified attributes related to reading pages (number of pages, average reading time) and taking tests (number of tests, average time, number of correct and incorrect answers) as particularly important for predicting engagement. Two characteristic patterns of disengaged users were found: users who click fast through pages without reading them and users who spend a long time on a page. On the basis of such unobtrusive assessment of motivation or engagement, the authors argue for further steps targeting unengaged learners, which consist in assessing self-efficacy, self-regulation, etc. in a dialogue and providing them with a reasonable intervention.

Baker, Corbett, and Koedinger (2004) investigated a misuse of intelligent tutoring systems called 'gaming the system', which has been found to be negatively correlated with learning. Two types of this misuse can be distinguished: help abuse and systematic trial-and-error. With log data and human-coded observations, an algorithm was developed on a set of latent response models to detect a gaming-the-system behaviour. The algorithm allows, on the basis of specific predictors, to identify gaming behaviour.

The hitherto presented research shows ways of using easy to assess information, like the time a person spends on a specific task, for the assessment of a person's motivation. Commonly, human judgements are utilised in order to set up rules and models of learners' motivation and to investigate their correctness. These procedures may, however, feature flaws and lack reliability and validity, as the expert ratings ground on observations for deriving rules to predict a learner's motivational state with the aid of observable variables. A supplement to this approach would be to use self-assessed data to validate the rules and patterns.
Beck (2005) developed a formula to estimate students' engagement in an intelligent tutoring system including multiple-choice cloze questions, without the need for a human rating of user interactions. The introduced technique called 'engagement tracing' is based on item response theory and uses response time and other parameters to model and calculate disengagement.

A research field potentially useable also for motivational adaptation is that of affect perception in educational software. Virvou and Katsionis (2003; Katsionis & Virvou, 2004), for example, tried to recognize students' emotions from errors and detectable performance characteristics. On the basis of a virtual reality game for teaching English orthography and grammatical rules, the authors investigated, with the aid of human experts, what evidence about the emotional state of a learner can be drawn from log data – including use of keyboard and mouse as well as knowledge of domain. The authors identified certain patterns of action related to several different affects, such as, for example, the the number of times the delete button is pressed while forming an answer is a measure of certainty of the particular answer; mouse movements without any obvious intent are connected either to little concentration or to frustration (after the student has been asked a question); the number of drop outs indicates disappointment or boredom. The structure/categories of emotions according to the cognitive theory of emotions (Ortony, Clore, & Collins, 1988) was used to model students' emotions while they learn (Katsionis & Virvou, 2005). This addresses behavioural characteristics (like mouse movements, speed of answering, use of agent, use of map, and use of inventory) and cognitive characteristics (like error frequency, error persistent occurrence, correct answer frequency, and correct occurrence) to deduce learners' emotions.

**Indicators for motivation**

In order to realise a successful assessment and monitoring of learners' motivation, the selection and operationalisation of appropriate motivational indicators as well as their proper interpretation is crucial. Exploiting different aspects of log data, as can be seen from the research sketched in the foregoing section, appears a promising and suitable approach for motivational assessment.

The time spent on a specific task or page tells much about the motivation of a player. The more time a learner spends on a task, the
more effort he/she spends. Moreover, if a learner acts on a certain task (e.g., reading a text, answering a question, or solving a problem) rather hastily, he/she is likely to have a lack of ‘attention’ or a feeling of boredom. Quick actions also relate to a systematic trial-and-error, which is a form of gaming the system. If the time spent is very long and exceeds a threshold value, the learner may also feature low attention because of being distracted, but he/she may also be not just not confident enough to continue.

Backwards steps: if a learner tends to go back very often, this may be interpreted as a lack of confidence and a worry about failure.

The frequency of accessing help or asking for hints is another reasonable indicator. Help or hints are demanded frequently by persons who are unconfident. Obviously, this is not the only reason to ask for a hint. A person may also ask for help if he/she lacks the knowledge or skills required for a task. Another reason could be gaming the system through help abuse, which can be related to boredom and lacking attention. The time spent after receiving a hint is actually crucial. If the learner immediately asks for help again, he/she is probably abusing the help and wants to go through the task without spending much effort.

Pattern of mouse movements: A person who moves the mouse towards a specific goal and with an obvious intent is most likely attentive and confident about what he/she is doing. If the mouse movements are, however, random the player is probably bored or distracted and, therefore, may lack attention. If mouse movements are unassertive and little, this would rather argue for a lack of specific knowledge or confidence.

The performance of a learner is a good indicator for motivation, especially if it is combined with other observations. The number of mistakes or correct answers may be taken as a performance measure. A person who fails several times in a row would need a motivational intervention in terms of a confidence enhancement more than a person who continuously succeeds.

The degree of confidence in one’s own answer or response behaviour may be queried explicitly from learners and gives information about the current motivational state. If a person repeatedly indicates a low certainty on the correctness of a given answer this indicates a low confidence, in particular if the player nevertheless shows a satisfying performance.
For any motivational indicator used, the clear and thorough specification of mapping the indicator to the level of motivation is crucial and demanding. Thereby, the definition of threshold values suggesting and distinguishing between different motivational states is essential. Possible misinterpretations of certain behavioural patterns need to be considered and accounted for, which is a really challenging task. A very long reaction time or time of inactivity, for instance, might be mistakenly interpreted as lacking attention while the learner actually has gone to the bathroom. Often the triangulation of different indicator variables will be advisable and might lead to the best results. In this case it will probably make sense to use weighting parameters for each indicator, dependent on the specific learning or game situation.

In addition to these motivational indicators, another possibility to gather information on current motivation is by the presentation and use of dialogues. This can be realised in an educational game through interactive dialogue sequences between the player and a non-player character and, therefore, be smoothly integrated into the gameplay and narrative without having a disrupting character. The learner may be provided with different answer statements to choose from; the selection of specific options may be mapped to certain motivational states.

An example for such a dialogue would be:

- Question from non-player character: ‘Well, is there anything you want to tell me?’
- Answer alternatives:
  - ‘No, let’s go on with the challenges!’ → indicating high motivation
  - ‘I fear that I will not be able to accomplish this task!’ → indicating a lack of confidence
  - ‘Can we hurry up? This task is kind of boring.’ → indicates a lack of attention

**Motivational Adaptation in Learning Games**

Due to the dynamic nature of motivation, motivational interventions aiming at keeping the player engaged need to be adapted to the needs of the player. Only a player showing a risk of losing engagement should be provided with motivational adaptation in order to increase motivation again. On the contrary, a learner that is already highly motivated
should not get any motivational interventions, as these may even have a counterproductive effect.

As this adaptation should be provided throughout the game and, in particular, during complex problem solving situations, this calls for an extension of the framework of competence-based adaptation to incorporate also personalization to motivation. From a micro adaptivity point of view this means realizing a monitoring of the learner's current motivational state in terms of a non-invasive assessment, which can then be used as a basis to trigger motivation-enhancing interventions. From the perspective of macro adaptivity, motivational adaptation refers to the aspect of selecting and sequencing game situations or story elements in accordance with learner engagement.

The following subsections shall provide a more concrete picture of how motivational adaptation can be realized by presenting motivational assessment and interventions as it has been implemented in the demonstrator game of the 80Days project as a case study (see also Mattheiss, Kickmeier-Rust, Steiner, & Albert, 2010; Steiner et al., 2009). This work builds upon the advanced model of motivation in game-based learning as presented above, which integrates several theoretical models of motivation and motivational instructional design (Heckhausen & Heckhausen, 2006; Garris et al., 2002; Keller, 1987, 2008) into a common framework. The resulting model comprehensively covers relevant motivational aspects as a basis for adaptive educational games. Furthermore, the state of the art on motivational assessment and possible indicators for motivation has been incorporated in the considerations on the realization of a motivational assessment as well as on adaptive interventions as a complement to competence-based adaptation.

**Motivation monitoring**

The realization of motivation-based micro adaptivity consists of a non-invasive assessment of the current motivational state of a learner and adaptive motivational interventions that are selected according to this assessment.

When considering the advanced model of motivation in game-based learning, the activity stage is the stage of relevance when thinking about assessing and enhancing motivation during the learning game, as it refers to the actions and interactions between learner and game (cf. Figure 18). Grounding on this model ‘attention’ and ‘confidence’ have
been identified as the most important aspects of motivation during the activity stage of an educational game. Keller (1987) denotes attention as an important element of motivation and a prerequisite for learning. Therefore, attention needs to be directed to the appropriate stimuli and to be sustained. Confidence relates to the expectancy for success and influences learners' persistence and accomplishment. In order to maintain motivation, the development of confidence should be fostered.

These two aspects, attention and confidence, can be used to characterize a learner's motivational state. Broadly speaking, for each of these two motivational aspects two value levels can be assumed – i.e., high vs. low. Consequently, for the two motivational aspects (A = attention, C = confidence) and the different values on them (high vs. low) four possible motivational states result (see Table 3). Motivational interventions should be triggered only in case of one or two aspects being low, but not in case of both aspects being high, i.e., a motivational state (A+, C+).

<table>
<thead>
<tr>
<th>Confidence (C)</th>
<th>Attention (A)</th>
<th>low</th>
<th>high</th>
</tr>
</thead>
<tbody>
<tr>
<td>low</td>
<td>(A-, C-)</td>
<td>(A+, C-)</td>
<td></td>
</tr>
<tr>
<td>high</td>
<td>(A-, C+)</td>
<td>(A+, C+)</td>
<td></td>
</tr>
</tbody>
</table>

Directly asking the learner for his/her current status on these motivational aspects during the game would mean a disruption of the game flow; the learner would be forced to stop the game itself to answer to the question – which actually means compromising engagement, immersion, and motivation (Van Eck, 2006). Moreover, in case of attention this way of assessment is problematic, as an explicit assessment question would most likely influence attention itself. Instead of simply asking the learner for their current motivation, therefore, an approach for non-intrusive assessment of motivation has been implemented.

This approach consists of an assessment methodology interpreting a player's behavior during playing for measuring motivation and allowing real-time adaptation. Similar to the non-invasive assessment of skills, we can assign specific motivational assumptions to specific classes of actions. As outlined above, certain behavioral patterns and characteristics operationalizing different kinds of indicators (e.g., mouse move-
Balancing on a High Wire

ments, time measurements, help demands) can be utilized as indicators for certain aspects of a learner’s motivational state. If, for instance, the mouse movements of a learner are random, this can be interpreted as a lack of attention, and if hints are demanded frequently this can, under certain conditions, be a sign for a lack of confidence.

Two main indicators have been selected as indicators for attention and confidence. These indicators correspond to previous research on assessing motivation and are suitable for the realization of such an assessment in the scope of an adaptive educational game. The motivational assessment is realized on the basis of the following two main indicators:

- **Reaction time**: The time between a system feedback (e.g., a statement or hint given by a non-player character) and a player (re)action.

- **Number of errors**: This variable in general refers to the erroneous responses or actions of a learner. As motivational adaptation is assumed to complement competence-based adaptation, this indicator relates to the skill assessment. When a probabilistic approach of skill assessment is realized, this indicator is based on the skill probability updates (decreases) in the current game situation. The number of errors is operationalized by counting the number of learner actions (within a certain situation) leading to a decrease in the probability of at least one skill, i.e., taking into account actions that are not fully correct/desired.

Instead of distinguishing only two broad categories for each motivational aspect, i.e., high vs. low attention or confidence (as sketched in Table 4), the motivational indicators allow realization of a fine-grained measurement of a learner’s motivational state. The motivational assessment is realized in a similar way as the skill assessment, in terms of continuously updating the value of both motivational aspects throughout the game (cf. p. 114 for technical details). Both, attention and confidence are assumed to have values between 0 and 1, with 0 meaning no attention/confidence and 1 meaning full or very high attention/confidence. At the beginning of the assessment a medium level of motivation, i.e., a start value of 0.5 for both aspects, is assumed. By the use of the motivational indicators, learner actions are defined for the individual situations of the learning game, which lead to increases or decreases in the value of attention and/or confidence. The definition of
these actions or rules is based on the previous research on motivational
assessment and motivators as sketched above.

The estimation of attention grounds on the following considerations:

- If a learner reacts immediately or very fast after a system feed-
  back (e.g., already during assumed reading time), this is as-
  sumed to indicate a lack of attention. The learner will likely be
  not concentrating and might follow a trial-and-error method.

- If reaction time after system feedback is very long, i.e., exceeds
  a certain threshold, the player is assumed to lack attention as
  he/she does not seem to focus perception and action on the
  desired task.

- If reaction time is appropriate, the level of attention is appro-
  priately high.

If the reaction time is either appropriate or too long, the number of
errors is considered in a second step as an indicator for confidence:

- A high number of errors is assumed to decrease a player's con-
  fidence in his/her own ability.

- If, on the contrary, the learner shows only few errors and most
  actions are approaching towards the desired solution state, this
  is assumed to increase the learner's confidence.

These considerations lead to the basic mapping of the motivational
indicators to the four motivational states as depicted in Table 4. These
general rules need to be concretized for an individual learning game and
game situation by translating them into rules with threshold values for
game or action evidences. This means action classes are specified in
order to trigger updates of the attention and confidence values. These
specifications may, for example, be of the following type:

- If the response time until the learner reacts after having re-
  ceived an instruction/hint is ≤ 2 seconds, the attention value
  should be decreased.

- If the response time until the learner points at the map after
  having received an instruction/hint is ≥ 10 seconds and the
  number of errors is ≥ 3, the attention value and the confidence
  value should be decreased.
If the response time until the learner changes flight direction after having received a hint is 2.1 to 9.9 seconds and the number of errors is < 5, the attention value and the confidence value should be increased.

**Table 4. Motivational indicators and their mapping to the motivational states.**

<table>
<thead>
<tr>
<th><strong>Attention (A)</strong></th>
<th>low</th>
<th>high</th>
</tr>
</thead>
<tbody>
<tr>
<td>low</td>
<td>- reaction time too long AND - number of errors too high</td>
<td>- reaction time appropriate AND - number of errors too high</td>
</tr>
<tr>
<td>Conf. (C)</td>
<td>high</td>
<td></td>
</tr>
<tr>
<td>low</td>
<td>- reaction time too short OR - reaction time too long AND - number of errors appropriate</td>
<td>- reaction time appropriate AND - number of errors appropriate</td>
</tr>
</tbody>
</table>

A critical point when defining the action classes for the motivational assessment is the appropriate specification of the threshold values on response time and errors, i.e., the decision which time frame for response time is still appropriate and which response times are too short and too long and, respectively, which number of errors might be critical for an increase or decrease in the level of confidence. For this psychopedagogical considerations and a careful analysis of the actual learning game situations have to be carried out in order to come up with the most suitable values. Once a motivational assessment is implemented into the educational game, the appropriateness of the values can be tested and, if necessary, threshold values can be refined.

As can be seen, behavioral patterns in the interaction with a learning game can be used to realize a non-invasive assessment and monitoring of a learner’s motivational state. The non-invasive assessment of a learner’s motivational state can on principle be carried out throughout the whole game in order to gain a complete picture about the learner’s motivation and motivational curve during the gameplay and learning experience. To allow this for each and every game situation or scene, the relevant learner actions need to be identified and the conditions under which an update of a motivational aspect should be triggered.
need to be specified. On the basis of this assessment motivational interventions may be initiated if the system detects a lack of motivation referring to one or both motivational aspects.

**Motivation-enhancing interventions**

Motivational or motivation-enhancing interventions are in general any kind of system feedback (e.g., hint from a non-player character, change in the user interface) that are supposed to enhance and retain learners’ motivation and engagement on a high level. Motivational interventions, in general, may be based on the non-invasive assessment of skills as well as on the motivational assessment.

As learning and the perceived level of challenge are tightly related with a learner’s motivation, motivational interventions may be realized on the basis of the non-invasive assessment of skills. Such interventions may be encouraging in case of failure or praising in case of success and – aside from supporting motivation, in general – are assumed to be incentives for learning and skill acquisition. These interventions address motivation on a general level by trying to foster positive affect and engagement for learning. The following motivational interventions can be distinguished and provided on the basis of the skill assessment:

- **Praising interventions** congratulate in case of success. They are assumed to be incentives for learning and reinforcing motivation. As, however, success is assumed to have a certain motivating character anyway, this intervention type shall be used rather sparingly. **Encouraging interventions** are used in case of failure in order to promote further trials and keep motivation. If, though, a learner lacks confidence, this type of intervention will not be sufficient. Rather, interventions addressing confidence in particular are advisable (see below). **Incitation interventions** announce pleasant outcomes such as rewards in order to foster motivation to carry on in the game or to continue in a problem solving situation. **Affective interventions** address emotional-affective aspects of the learning game experience and social interaction with other game characters and are supposed to foster a positive affect.

Motivational interventions, however, may go beyond the use of general motivational enhancements and consist in specific interventions in response to the non-invasive assessment of the current motivational state of a learner. Interventions that are provided on the basis of the motivational assessment directly target the specific motivational aspects.
of attention and confidence. A range of interventions types targeting the motivational state of the learner can be distinguished and are shortly explained below.

Attributional interventions address the aspect of confidence by providing suggestions of self-worth enhancing attributions to success and failure, corresponding to attributional theory (Weiner, 1974). They aim at fostering self-worth enhancing attributional styles through a re-attributional training (Dresel & Ziegler, 2006) and are applied in case of lacking confidence or dysfunctional attributional styles. In general, the factors affecting the attribution of success and failure of achievement are ability, effort, task difficulty, and luck (Weiner, 1974). These factors can be classified according to the two dimensions of stability (stable vs. variable) and locus of control (internal vs. external). Favorable attributional styles explain success by internal and failure by variable factors. Attributional interventions realize a motivational training based on attribution theory in case of a determined lack of confidence (for an overview see Figure 6). Attribution to success should first be made to effort and after some time and with increasing expertise to ability (Dresel & Ziegler, 2006). Failure, on the contrary, may in the beginning be attributed to bad luck (false answer behavior may just have a careless slip); afterward it should be attributed to effort in order to make the learner understand that the reason for failure is not stable (as it would be in case of attributing to ability) and the locus of causality lies within the learner and, thus, to encourage the learner to try harder. Correspondingly, there are four different subtypes of attributional interventions depending on the outcome (failure or success) and the explaining factor: success intervention effort, success intervention ability, failure intervention effort, and failure intervention bad luck. Attention catchers address the aspect of attention if the system detects a lack of attention; they try to raise attention through unexpected statements or occurrences, such as a dramatic statement, a sharp noise, or even a quiet pause, and in this way increase the variability and appeal of the game. The learner can be prompted to focus action and perception on the desired task and curiosity is evoked. Finally, there are two different types of motivational assessment interventions, which serve for clarifying information on the current motivational state in case of unclear or ambiguous assessment results. In this case, an explicit querying is realized in form of a question posed by a non-player character, in a way that it is as highly embedded into gameplay and story as possible. Such an intervention will consist of an interactive dialogue with the different
answer options providing indications on the current motivational state. *Attention assessment interventions* ask about the learner’s attention. Although this intervention itself will likely increase attention, such kind of query is helpful in order to determine boredom, lack of interest, or perceived irrelevance which probably lead to inattention immediately after again. *Certainty questions* ask the learners for their confidence in the correctness of a specific action or response and therefore gather direct information on this motivational aspect. This type of assessment intervention could also be considered as a kind of meta-cognitive intervention.

Only in case of a low level of attention or confidence, a motivational intervention may be triggered, tailored to the specific motivational state of the learner. In case of appropriately high levels of attention/confidence, on the contrary, a learner should not be distracted by a motivational intervention, as this might even have a counterproductive and, thus, motivation-decreasing effect.

![Diagram](image)

*Figure 19.* Factors explaining success and failure in re-attributional training.

Regarding the selection of appropriate interventions for motivation-based adaptation, similarly to competence-based micro adaptivity, selection rules for the individual intervention types have to be defined. These selection rules translate the theoretical, psycho-pedagogical considerations presented above into concrete rules that are understandable and processable by the system. Also, in this case, meta-rules will specify
the general scope of motivational interventions, e.g., to ensure that no inappropriate coincidence of two interventions occurs. In addition, rules for the selection of a certain intervention type will ensure that the most appropriate intervention is provided at each time. An intervention is triggered based on an update of the value of a motivational aspect due to some learner action – if the value of either attention or confidence falls below a specified threshold level an intervention is triggered. For the selection of intervention types realizing an attributional training not only information from motivational assessment itself is utilized, but also information stemming from the skill assessment is explicitly taken into account in order to trigger the appropriate attributional factor for success and failure, respectively. This refers especially to the sequencing of intervention types, for example, to the case that attribution of success shall initially be suggested to effort and with increasing skill level, i.e., successful learner actions shall be related to ability.

It can be assumed that a change in the motivational state of a learner does not necessarily need to be directly tied to a certain, observable learner action. Rather, the provision of certain motivational interventions – as they are directly targeting a change in the respective motivational aspects – is assumed to lead to an update of the values of the considered motivational aspects. Consequently, a motivational assessment intervention will, depending on the aspect it targets, directly lead to an update of the attention value – by immediately translating the answer option selected by the player into an according decrease or increase of the respective motivational aspect value. In addition, it could also be assumed that that the provision of an attributional intervention or, respectively, an attention catcher leads to an increase in confidence or, respectively, attention. In order to avoid an overestimation of the respective motivational states, in this case it appears reasonable to either do without such an immediate update and rather only realize a decrease and increase of the motivation value by the interpretation of the learner’s behavior or to carefully select the update parameters such that an intervention leads to only a slight change in the motivation value. A summary and overview on the motivational adaptation process in terms of motivation monitoring and motivation-enhancing interventions is provided in Figure 20. An example included in the Figure illustrates the different stages of the adaptation process.
Adapting to motivation on a macro level

Aside from the provision of motivational interventions in terms of micro adaptivity, the system may react and adapt to the learners motivational state also on the macro level.

In case of a lack of attention, there is the possibility of adaptation on a macro level in terms of a change of the game pace. If the learner shows a continuous lack of attention and according micro level interventions have already been provided but were more or less unsuccessful, the game may switch to a faster game pace. This will in general not only mean or require a faster gameplay, but will also involve variations in a game’s story. A change of game pace based on the motivational assessment in a certain game situation will, in general, become evident in the subsequent game situation. As a result, a learner who starts in a relaxed game pace may not necessarily end the game in this pace, but may successively be presented with increasingly hectic game and story paces.

In case of a continuous lack of confidence the system may come to the conclusion that the level of challenge is too high for the respective learner. Even with the provision of appropriate attributional interventions, a continuous experience of failure will compromise confidence in one’s own abilities. This may lead to an adaptation on the macro level consisting of switching to an easier task (i.e., decreasing difficulty). This might be done by the presentation of a less demanding version of the current problem solving task or a different task. Such kind of macro adaptation addressing confidence will allow the learner to be successful, to demonstrate his/her capability, and to experience himself/herself as competent.

Conclusion

This chapter elaborated on the importance of adapting learning experiences in educational games to the individual needs of the learner. Adaptation to the current knowledge and competence level of a learner as well as to his/her current motivational status should thereby accompany and complement each other. This is because learning and motivation are tightly interconnected. Providing an appropriate level of challenge by tailoring learning activities to the skills a learner has available is a crucial aspect for keeping a learner engaged in a learning game and to evoke flow experience. Conversely, an adequate level of motivation is
necessary in order to bring the learner to actually deal with a learning game. In order to provide the learner with most appropriate support, adaptivity in a learning game should be realized between learning situations, i.e., selecting and sequencing learning activities in a sound manner, as well as within learning situations, i.e., providing support in complex problem solving tasks. The concepts of macro and micro adaptivity characterize these two different levels of adaptation. The smooth embedding of adaptive mechanisms in the game is, thereby, a critical precondition for effective personalization. Above all, the narrative of a learning game needs to be taken into account such to realize not only meaningful learning paths but also coherent story lines.

Figure 20. Overview and example of motivational adaptation process.

CbKST provides a sound mathematical psychological basis for competence-based adaptation. This theoretical framework gives rise to establishing knowledge and competence structures for a knowledge domain, which can be exploited as a sound cognitive basis for the implementation of adaptation technologies. The established structures can not only be used for macro adaptive sequencing of learning situations and thus, the realization of meaningful learning paths, but also for non-invasive
assessment and interventions in terms of sophisticated problem solving support.

Adaptive educational mechanisms often mean providing the learner with automated customization or giving the learner specific recommendations, for example, which learning object should be attended or how a problem could be solved. There is a broad body of research in the area of adaptive educational systems, including that on CbKST and a variety of other theoretical frameworks (e.g., Ohlsson & Mitrovic, 2006), covering this kind of adaptation. There is, though, a parallel research community that is focusing on similar problems, namely (commercial) recommender systems. Recommender systems are used, for example, in online shops to suggest specific products to potential customers in a most tailored way. Despite an obvious overlap in terms of theory and technology, exchange between those research communities is sparse (one approach is described by Drachsler, Hummel, & Koper, 2009). Recommendation techniques are often based on collaborative filtering (Konstan et al., 1997; Liu, Lai, & Lee, 2009), which exploits item preferences of similar users, content-based filtering (Pazzani, 1999), which determines recommendations using item preferences of the current user, or knowledge-based recommenders (Felfernig & Burke, 2008), which conclude recommendations on the basis of an explicit representation of the advisory knowledge (e.g., in the form of rules or constraints).

Future research on adaptivity in educational systems and learning games should account for the experiences and existing solutions in the area of recommender systems and incorporate such approaches towards novel methods for knowledge diagnosis and subsequent interventions. Research towards an integration of the CbKST approach to intelligent adaptation and the recommender and filtering techniques could further enrich the potential of personalization in technology-enhanced learning. A specific research goal would be to enable the adaptive educational technology for educational games to autonomously learn from the learners and groups of learners playing the game. Model-based diagnosis (Reiter, 1987; Steinbauer & Wotawa, 2009) can help to automatically identify minimal sets of inconsistent rules and thus support a significant reduction of development and maintenance efforts triggered by inappropriate learning path recommendations and interventions. Collaborative and group recommenders can help to personalize the selection of repairs (Felfernig et al., 2009) and to achieve
consensus regarding the acceptable set of path selection and intervention rules. These concepts may be developed for single user scenarios as well as for groups based on existing work in the fields of dialog learning (Felfernig, Friedrich, Teppan, & Isak, 2008; Mahmood & Ricci, 2007), group recommendation (Jameson, Baldes, & Kleinbauer, 2004), and contextual recommendation (Cocea & Magoulas, 2009).

The aspect of collaboration is actually getting more and more prominent in the context of educational games. This is due to the increasing prevalence and popularity of multiplayer games. This trend has been recognized and taken up in the context of technology-enhanced learning, by trying to exploit multiplayer games for educational purposes (e.g., Childress & Braswell, 2006; Lee, Eustace, Fellows, Bytheway, & Irving, 2005; Padilla Zea, González Sánchez, Guriérrez, & Paderewski, 2009). Through the incorporation of psycho-pedagogical know-how on group learning, in general, and computer supported collaborative learning, in particular, multiplayer games have the potential to foster communication and interaction between learners and to facilitate cooperative learning. To fully exploit the potential of such collaborative learning activities, the adaptation mechanisms in an educational game should be translated to the group level. In order to be able to properly apply the principles of micro level assessment and interventions as well as macro adaptivity on the group level further research is needed. The theoretical developments and technical implementations of adaptive interventions for individual learners cannot be adopted for groups of learners in an un-reflected manner. Rather, the specific characteristics of collaborative learning need to be taken into account and relevant state of the art on computer supported collaborative learning, group adaptation, and cognitive as well as social psychology of group learning and groups need to be incorporated and synthesized in the theoretical framework. The perspective of macro adaptivity, in this case, has to care for the realization of learning and story sequences that are meaningful to both the individual as well as the group. Micro level assessment may serve different purposes in a collaborative learning context. The continuous monitoring and interpretation of individual learners’ behavior may serve the adaptive formation of appropriate learning groups in terms of adaptive collaboration support (e.g., Brusilovsky, 1999). Furthermore, non-invasive assessment can serve the creation of a group model (e.g., Jameson & Smyth, 2007) characterizing the competence of a group of learners. In addition to the consideration of individual assessment profiles for collaborative learning, the assessment
procedure may be exploited to realize a continuous assessment on the group level itself. Based on and triggered by the assessment results, appropriate adaptive interventions could be selected and presented. In case of collaborative learning, the interventions provided need to be adapted to the group model. Moreover, in case of group learning there will be further intervention types specifically suited for the context of collaborative learning, such as group formation interventions. In sum, future research should aim to extend the presented considerations on adaptation in educational games from the level of individual to the level of group learning. The theoretical approaches to individualized instruction and adaptation can be used to support collaboration and problem solving in collaborative learning scenarios (Kickmeier-Rust, Steiner, & Albert, 2009; Baghaei & Mitrovic, 2007). This may be assisted by research in the field of recommender systems, as mentioned above, which has extensively dealt with collaborative filtering and group recommendation techniques.

The incorporation of motivation in the adaptive technologies of educational games brings enormous added value and potential to adaptivity. Motivation is key for learning and, thereby, different aspects of motivation may be considered. Attention, which itself can be denoted as one component of motivation (Keller, 1987), is crucial for learning and needs to occur simultaneously with learning in order to allow the perception, understanding, and acquisition of learning contents (e.g., Anderson, 1982; Hidi, 1995). Consequently, it is important to attract and direct learners’ attention to the significant aspects of the game. Interest is also closely related to motivation and sometimes referred to as interest-based motivation (Hidi, Renninger, & Krapp, 2004). Interest may be evoked in terms of individual interest (e.g., preference for a certain topic) or situational interest (e.g., appealing learning environment). Situational interest is assumed to be evoked by the general nature of game-based learning; individual interest may be enhanced by tailoring game type, pace, or narrative to a learner’s individual preferences. This actually also refers to the creation of relevance of learning situations (Keller, 1987). Confidence or self-efficacy (Schunk & Pajares, 2002) as a motivational concept denoting the belief in one’s own ability to master a certain task is also crucial for learning as a person’s self-efficacy beliefs influence goal setting, task choice, willingness to learn, and persistence. A learner’s confidence should therefore be enhanced by learning tasks of the appropriate level of challenge and feedback mecha-
nisms that foster beneficial attributions to success and failure (Dresel & Ziegler, 2006).

We have presented how the assessment of the actual motivational state of a learner and adaptive interventions tailored to the learner's current motivation can be realized in an educational game. Motivational assessment is needed to provide the player in the right moment with a specific motivational intervention, while allowing engaged people to play without disturbance. The realization of micro level adaptation for motivation entails the translation of adaptation mechanisms and principles that are grounded on sound psycho-pedagogical theories, models, and considerations on instructional design (e.g., Keller, 1987; Weiner, 1974) into the concrete design of an educational game. This implies the identification and classification of learner actions and their mapping onto the level of motivational aspects by the use of appropriate motivational indicator variables, the design of instances of the different types of motivational interventions, and the specification of rules for selecting these interventions. The adaptation principles presented include motivational interventions that aim at enhancing motivation by directly targeting attention and confidence as important aspects making up a learner's motivational state. As these adaptive interventions ground on assumptions on the current motivational state of a learner as derived from a continuous, non-invasive assessment and interpretation of a learner's actions in the game, the provision of such interventions can only be as good as the underlying motivational assessment. A precondition for realizing successful interventions is, therefore, a successful assessment process yielding valid assumptions on the learner's characteristics on which the interventions are relying. This calls for appropriate indicators of motivational aspects, on the one hand, as well as for the proper interpretation of learner actions, on the other hand. The assessment of motivational aspects is not perfect and misinterpretations may occur, leading to over- or underestimation of motivation. It is necessary to carefully define the indicators and rules for drawing assumptions on the motivational state. Often the triangulation of different indicator variables will be advisable. In addition, it is sometimes reasonable to strengthen the conclusions drawn by more explicit information gained from motivational assessment interventions.

Future research on motivational adaptation should further investigate the suitability of different motivational indicators for assessment in the context of educational games and how they could be harmonized with
other sources of information in order to further increase accuracy and efficiency of the assessment results. Moreover, aspects of attention and confidence for modeling and representing a learner’s motivational state are important and how they could be complemented by further aspects in order to gain a comprehensive picture of motivation in and educational game.

The conduction of empirical investigations in order to thoroughly investigate the functioning and actual benefit of the competence-based and motivational adaptivity provided by an educational game is essential. The effects of micro- and macro level adaptation need to be systematically and thoroughly examined in order to demonstrate their added value and to be able to further refine adaptation principles. Analyses conducted in the context of the ELEKTRA project and the 80Days project with educational games teaching physics and, respectively, geography revealed that adaptation leads to better learning performance and superior game experience compared to non-adaptive versions of the games (Kickmeier-Rust, Marte, et al., 2008; Kickmeier-Rust & Albert, 2010). Future research should investigate in even more detail the benefits of the different adaptation and intervention types and their possible combinatorial effects. The in-depth evaluation of adaptivity in an empirical context will give evidence of the applicability and significance in the educational practice of game-based learning and will serve further elaboration of the theoretical frameworks for adaptivity.
Adaptive Digital Storytelling for Digital Educational Games

Stefan Göbel, Florian Mehm, and Viktor Wendel

Digital Educational Games (DEG) use the positive properties of games to educate their players. These properties include the motivation intrinsically found in games, which is connected to games inherently requiring and encouraging learning (of game mechanics, level layouts, etc.), as well as the fact that (digital) games are well accepted especially in the case of younger audiences. This interplay between Learning and Gaming establishes the field of Game-based Learning.

Figure 21. Symbiosis and Interplay of Learning, Storytelling and Gaming in Digital Educational Games.

Apart from Learning and Gaming, a DEG can also combine these fields with an interesting and motivating Story to keep players engaged with the subject matter. While the presentation of learning content embedded into a well-structured and narrated story results in the field of Narrative Learning (Kickmeier-Rust, Göbel, et al., 2008), the combination of all three areas can lead to an engaging Story-based DEG.
which uses both elements of games and of storytelling to educate or train.

Going even further, a DEG can try to react to characteristics and choices made by players, leading to a personalized game by means of adaptation mechanisms (Koidl et al., 2010; Peirce et al., 2008). As an example, a learner with deficits in a certain area could be presented with more content teaching this skill. In order to keep the game entertaining, the story and gameplay would have to react to this adaptation by also adapting these two fields.

In this chapter, the characteristic concepts underlying each of these fields are presented as a basis for further discussion of the technical realization of their interplay in the context of the 80Days software platform. Concerning Storytelling, the basic theory of Interactive Digital Storytelling in general and the notion of story models in particular will be described. For the area of learning, the theory of Competency-based Knowledge Space Theory (CbKST) will be presented (Albert et al., 2007; Conlan et al., 2006). For gaming, the concept of player modeling will be introduced, which allows adapting a game to be most pleasing for a certain player type.

In conjunction, these concepts will be unified into the notion of so-called Narrative Game-Based Learning Objects (NGLOBs) (see Göbel et al., 2010), which describe parts of a game using these concepts in order to be able to adapt them in an ideal fashion to the players of the game. The chapter will be concluded with a section on authoring tools, which are an important means to introduce experienced authors and educators in the process of creating a Story-based DEG (Göbel et al., 2009) and a brief summary with the main achieved results and suggestions for next research and technical development (RTD) steps.

**Interactive Digital Storytelling**

Since the beginning of human kind, stories have been used to share and transfer knowledge and personal experiences. While in the early days this was limited to oral presentations, recently, within the digital age and information and knowledge society, new forms of storytelling and narrative concepts came up – for instance Digital Storytelling, Interactive Storytelling or Interactive Digital Storytelling.
Adaptive Digital Storytelling

In one sentence, our approach to Interactive Digital Storytelling might be characterized as (the use of) ‘stories as instruments for interactive knowledge transfer of content, knowledge, (personal) experiences and facts.’ Hereby, the focus is set on the combination of interactivity (interactive, non-linear knowledge transfer and story representation), the use of Digital Media (multimedia, interactive media) and well-proven Storytelling concepts (story models, suspense, and dramaturgy):

- The story should be presented in a suspenseful and exciting manner, which fascinates the users (audience, recipients of the story).
- The users should become engaged by a story and dive into an immersive narrative environment - flow effect/flow theory introduced by (Csikszentmihalyi, 1997).
- Digital, interactive media (such as virtual environments or computer games) should be used to enhance the knowledge representation and visualization by graphic-interactive features and the possibility to interactively explore the content (story) instead of 'purely consuming' some content in a passive way.

This approach is in accordance with the definition and foundations of Interactive Storytelling provided by the IGDA (International Game Developers Association):

*Stories have most likely been part of the human experience from the earliest days of language, but until recently the storytelling medium has been largely static. Barring different versions of the same story, any given tale unfolds the same way every time one reads it. Computer games promise the potential to move beyond this strictly linear form by offering stories that interact with the player, allowing them to participate in the decisions or actions that shape the narrative. However, at the current time this field is still in its infancy.*

Chris Crawford, as one of the pioneers in Interactive Storytelling, defines the term as “a form of interactive entertainment in which the player plays the role of the protagonist in a dramatically rich environment” and continues that “the experience of interactive storytelling differs substantially from that of a conventional linear story. A linear story ‘runs on rails’ from start to finish in the most powerful and expedient manner possible. The interactive storytelling experience meanders through a dramatic universe of possibilities. It lacks the sense of directed inevitability that gives conventional stories such power. It is
An Alien’s Guide to Multi-Adaptive Educational Computer Games

like a butterfly flitting across a meadow, not a hawk plummeting down on its prey. The closest form of traditional storytelling is the soap opera, which concentrates on the relationships among the characters rather than the particular plots” (Crawford, 2004).

The definition of Chris Crawford points out the difference (conflict) between linear, plot-based Storytelling systems and non-linear, emergent narratives. 80Days contributes to this field and the symbiosis of Storytelling and Gaming for (Technology enhanced) Learning.

The main questions tackled within 80Days have been ‘What makes a good, Story-based learning game?’ on a fundamental level and, more concrete ‘How to build such an exciting game?’ as well as ‘How does a story continue at a certain moment during play?’ – which is closely connected to the ‘conflict’ (narrative paradox) being addressed by Crawford.

To answer these questions, the consortium investigated research in adaptive, interactive digital storytelling as well as learning and psychology including competence development. From a Storytelling perspective, the major challenges we addressed concern a) integration issues, both on a conceptual and technical level, b) the Narrative Paradox describing the conflict between author and player control (Louchart & Aylett, 2003), c) the question about what happens during run-time when some conflicts occur among the different approaches (storytelling, learning, gaming) respectively among the corresponding run-time components (Game Engine, Story Engine and Adaptive Learning Engine) and how the game continues? The following sections of this chapter concentrate on those adaptive storytelling aspects for DEGs.

During the 80Days project, we have elaborated an adaptive storytelling framework addressing the RTD issues mentioned above. Central aspects include the analysis and use of appropriate story forms, structural story models (serving as the threads of stories), and the conceptualization of Narrative Game-based Learning objects (NGLOBs) as atomic story units to compose such Story-based DEGs.

**Story Forms – Sequencing in DEG**

With respect to macro adaptation and the question how a story continues at a specific moment during play, Figure 22 presents the main classes of story forms for games and the pros and cons of its usage for sequencing and macro adaptation in Story-based DEGs.
The simplest form represents the linear approach. The advantages are that the author has full control of the story and it is quite easy to implement that approach from an author/game developers’ perspective. Contrary, the major drawback is the lack of flexibility and possibilities for personalization and adaptation in the sense of macro adaptivity and sequencing. Branching is a little more complex than the pure linear approach and more expensive in terms of content production; however, there is still full authorial control and a lack of flexibility for macro adaptation. The non-linear approach is more flexible and provides some space for macro adaptation and sequencing of story units due to the variety of transitions per story unit. Nevertheless, there is a limited possibility to combine and re-use story units in different scenarios, story lines, and contexts, for instance, different user groups, game lengths, or game modes in 80Days. Within the modular approach, the set of story units might be understood as ‘a sea of pearls’ (story modules), which might be (in principle) freely connected and combined with each other. This approach builds the basis for emergent narrative Storytelling systems (Figueiredo et al., 2008; Louchart & Aylett, 2004) and offers best opportunities for macro adaptation and an almost endless set of possible storylines/paths or sequences of story modules. On the other side, authorial control is very limited – contrary, the player gets more or less full control over the scenario – and it becomes quite difficult to ‘guarantee’ a suspenseful story.
In sum, it is definitely not possible to determine which story form is most appropriate for adaptive, Story based DEGs. Contrary, from our perspective, the challenge is to combine the different forms and to find some good balance among linear and modular concepts to provide enough flexibility for macro adaptation and sequencing on the one hand and, simultaneously, to ‘guarantee’ some suspenseful story based on well-proven and elaborated story structures such as story models used in literature, theatre, TV/film or Story-driven adventure games.

**Hero’s Journey as Red Thread in 80Days**

From a top-down perspective, that combination of linear and modular concepts might result in some guided but still open and flexible story structure such as the Hero’s Journey (Campbell, 1949; Vogler, 1998), well proven in the field of learning games, especially learn adventures. We identified the Hero’s Journey (see Figure 23) as the most appropriate existing story model matching the requirements of 80Days to create highly flexible and adaptive, yet suspenseful and informative DEGs following the RTD paradigms of 80Days.

![Figure 23. Hero’s Journey story model (left) – linear and modular story units (right).](image)

In release 1 of the project’s demonstrator game (Lizard 1.0), being an ‘early demonstrator’ after project month 15, the first part of the Hero’s Journey (the departure) and a good portion of the middle part (the
Adaptive Digital Storytelling

initiation) have been covered: A cinematic intro has been produced to cover the story start and first steps of the Hero's Journey, a tutorial (introducing geography as subject and explaining the gameplay) might be understood as first threshold. Further, different micro missions (~story units or quests/ level in a typical game) refer to the dramaturgic function of the road of trials, which represents the modular part of the story model. This means (in theory) the players can decide the order of micro missions on their own, whereby, (in practice) not all sequences (=learning paths) do make sense. For instance, based on CbKST and underlying skill structures (Conlan et al., 2006), it might be better to visit a micro mission 1 first and to learn something about subject A, before mission 2 with a subject B (where it is useful or even necessary to have some background knowledge on skill A). Therefore, the skill structure and useful transitions among micro missions are used within the authoring tool to define reasonable transitions among micro missions and its corresponding associated skills. The rest of the story model, especially the climax of the story, where the transfer of the moral of the story ("Save the earth!" in the DEG on geography in 80Days) has been realized within the Lizard demonstrator versions 2 and 3 during the course of the next project phases.

Learning

Adapting DEGs, while incorporating learning issues, consists of tracking the players' knowledge state and adapting the game's story structure according to it in a personalized, adaptive manner. The gathered information is used for sound decisions how to continue the game on a macro level (sequencing of scenes/situations) and in terms of micro adaptation. The aim is to let the player be neither unchallenged nor overwhelmed by the complexity of the tasks covered by the selected scene. Rather it is strived to steadily increase the player's learning performance. In 80Days the CbKST is utilized for this purpose (Conlan et al., 2006) - i.e., a set of not observable skills S and a prerequisite relation R among these skills. S and R are elaborated by the author and stored in the DEG's learning context. The competence structure C defined by R over S then allows us to determine meaningful learning paths for the player. For this purpose each scene is assigned with two subsets of skills P and A. Prerequisite Skills (P) are assumed to be necessary to solve the contained tasks, whereas Associated Skills (A) are meant to be improved by the covered learning objects. Each skill s_i has a probability
value $p_i$ that is updated according to the player's behavior (e.g., right or wrong answers, clicking at the right position, time response). It is assumed that the learner has achieved a skill $s_i$ if $p_i$ exceeds a certain threshold $p_{\text{min}}$. The update of $p_a$ for skills $s_a$ in $A$ usually occurs at the end of a scene, but can also be triggered by user actions at any moment during play. Scenes are evaluated and ordered by comparing the scenes' prerequisite skills with the skills already achieved by a player. The topmost candidate is then proposed as next scene.

**Gaming**

Adaptation in terms of gaming means adapting a game such that it fits a player's preferences and gaming style. Therefore, it is necessary to model the player in order to characterize those preferences. One of the first player models was designed by Bartle in 1996 (Bartle, 1996). It defines four types of player types: ‘Killer’, ‘Achiever’, ‘Socializer’, and ‘Explorer’. Bartle’s model assumes that every player fits into one of these archetypes. A player who prefers action-packed games or who is a rather aggressive player seeking confrontation is, for example, classified as a ‘Killer’. Houlette (2004) introduced a player model that keeps track of several player traits to create a model which can be used to adapt the behavior of Non Player Characters (NPCs). In Façade (Mateas & Stern, 2003) the player can join an interactive story that adapts its agents' behavior according to the player's actions.

In Bat Cave, a mockup player to demonstrate the adaptive features and functionalities of the technological framework (see Figure 28), we use a modified version of Bartle’s player model, which does not simply sort a player into one of the four categories. Instead we model to which extent a player is a ‘Killer’ or an ‘Achiever’, etc. Therefore, we assign a vector to the player containing a number between 0 and 1 for each player type (‘Killer’, ‘Achiever’, ‘Socializer’, ‘Explorer’), which indicates how much the player is a ‘Socializer’, etc.

Player decision options in Bat Cave are, in many cases, related to a player type, so that a player has the option to solve a game situation the way he/she prefers. This can be a more aggressive solution, which implies that the player is rather a ‘Killer’ like choosing to attack an opponent, or a rather social solution like talking to the opponent in order to avoid a conflict implying a ‘Socializer’. Whenever a player chooses his/her option, the player model is updated accordingly. In Bat Cave
the player model is always visible on the ‘right side’ of the GUI. Whenever Bat Cave comes to a ‘free transition’, which means that the next scene is not predefined, but instead will be chosen by the story engine according to the player model, the current player model gets compared to the player model appropriateness of each possible next scene. Therefore, each scene has to be annotated by the author with values for each player type indicating the appropriateness of the scene for the respective player type.

The appropriateness value \( av_i \) of scene \( i \) then is

\[
av_i = \frac{1}{4} \sum_{j \in \{killer, achiever, socializer, explorer\}} (1 - |a_{i,j} - p_j|)
\]

with \( a_{i,j} \): the value of the \( j \)-th attribute of the annotated player model appropriateness of scene \( i \) and \( p_j \) the value of the \( j \)-th attribute of the player model. The factor \( 1/4 \) is a normalization to the interval \([0; 1]\), as there are 4 player types. Inside the sum, for each of the player types the difference between the model value \( p_j \) and the appropriateness value \( a_{i,j} \) of scene \( i \) is calculated as \( |a_{i,j} - p_j| \) and subtracted from 1. This way, the matching of model value and scene value is normalized between 0 (no match) and 1 (total match).

This comparison is executed for every possible next scene, and the scene which fits best is then chosen as the next scene. This way we can provide an individual story path for different types of players. However, it must be mentioned that a significant authoring effort is necessary as the author needs to create the different shaping and has to specify the player model appropriateness for the different scenes.

**Narrative Game-Based Learning Objects**

In order to combine the mechanisms of the learning, gaming, and storytelling contexts, we created so-called NGLOBs (Göbel et al., 2010), Narrative Game-based Learning OBjects. In sum, the model for an NGLOB is built by a composition of context information resulting in a triple vector \( C_N \times C_G \times C_L \). The narrative context \( C_N \) provides a list of tuples \((storymodelStep, appropriatenessFactor)\), whereby \( storymodelStep \) is encoded by the initials of a story model (for instance ’SM_HJ’ for the Hero’s Journey story model) plus a number for the step/part of that model. The parameter \( appropriatenessFactor \) indicates how much the scene fits to the according \( storymodelStep \) and is normalized in the range \([0,
1]. The gaming context $C_G$ primarily tackles the appropriateness of individual gaming objects and gaming situations for different player types. Analogously to the narrative context $C_N$, the gaming context $C_G$ also provides a list of tuples \((\text{playerType}, \text{appropriatenessFactor})\). Here, \('PM\_BA\_x'\) describes the player type based on the classification of Bartle (1996). For example, \('PM\_BA\_E, 0.9'\) indicates that the NGLOB is very appropriate for the player type 'Explorer' according to Bartle.

The model for the learning context $C_L$ provides a vector composed of two parts listing all Associated ('Axyz') and Prerequisite ('Pxyz') Skills for a specific learning situation/LOB, whereas 'xyz' is a unique identifier. Apart from that quantifiable part described above, the model for NGLOB contains further descriptive elements such as short texts/abstracts summarizing the synopsis of narrative, gaming and learning functions of a specific NGLOB. In Figure 24, an example for such an NGLOB is provided:

\[
\begin{align*}
(HJ\_2, 0.1),
(HJ\_4, 0),
(HJ\_4\_1, 0.2),
(HJ\_5, 0.85),
\cdots
\end{align*}
\]

\[
\begin{align*}
(PA\_B1, 0.15),
(PA\_B2, 0.4),
(PA\_B3, 0.2),
(PA\_B4, 0.9)
\end{align*}
\]

\[
\begin{align*}
(AI030, AI033),
(B2122, B2297)
\end{align*}
\]

Figure 24. Example for a Narrative Game-based Learning Object.

By use of NGLOBs, an author is supported in the creation of reusable scenes which are ordered automatically (in the non-linear parts of the game like the 'Road of Trials' which represents the greater part of a game's story) according to his/her focus.

**Authoring**

In order to apply the mechanisms described in this chapter, it is essential for the creators of a game to be able to configure them all in an easy and efficient way, with an authoring tool specifically constructed for this task. Using such an authoring tool, it is possible to create individual NGLOBs as parts of a game, edit the metadata required for their functioning, and link them together to form the overall game. In the context of the 80Days project and other project and RTD activities, such an authoring tool was created in the form of the StoryTec authoring tool (Göbel et al., 2008; Mehm et al., 2010).
Figure 25. A broad overview of the architecture of the 80Days platform.

Figure 25 shows a broad overview of the technical architecture partly underlying the 80Days project and how the associated technical elements and concepts are connected to the authoring tool. This architecture including the following components:

- The StoryTec authoring system, in which an author can create the structure of a game and place content in the game.

- A repository in which structures and content created in StoryTec is saved using the ICML format along with other data, such as log information, created during gameplay.

- The Story Engine, which parses the ICML files placed in the repository and uses the information in them to control the game at runtime. To this end, it communicates with both the Learning Engine for learning-related decisions as well as a Game Engine for control and feedback.

- A Game Engine, which provides the actual frontend to the user, which includes handling input, the implementation of gameplay mechanics, and graphics rendering.

In this section, we present in general terms one possible structured process of creating a story-based educational game using StoryTec,
thereby introducing the individual components that make up the authoring tool.

After starting the StoryTec application (see Figure 26 for a screenshot of the main user interface), the user has the choice of loading an existing project, creating a new one from scratch, or utilizing one of the templates that are provided. A template provides the author with a pre-configured structure which is either based on a story model that can assist authors with little dramaturgic writing skills in creating a coherent and complete story (e.g., by conforming to the Hero’s Journey story model) or on a project-specific structure that must be complied with (e.g., the structures a certain player component expects).

Figure 26. The main user interface of StoryTec. The main components in clockwise order from the top left: Stage Editor, Objects Browser, Property Editor, Story Editor.

Creating a Story

Constructing the story structure

If the users opted to create a project from scratch, a possible first step is defining the structure of this project. This can be achieved in the Story Editor component, in which the overall story is partitioned into
individual scenes or complex scenes (see Göbel et al., 2008). By drawing transitions as arrows between story units, the possible paths through the story are defined. Unconnected scenes should be interpreted as being freely combinable in an adaptive modular storytelling fashion, i.e., the most appropriate scene is selected during runtime, taking into account dramaturgic, learning, and gaming aspects (see Göbel et al., 2009). While other components like the Stage Editor are found in many authoring systems, the Story Editor as an abstract view on a story is a component that is an extension compared to many systems.

Scenes can be annotated in detail according to several categories. Authors can specify the expected time a user will remain in a given scene or which function of a story model a scene fulfills. A component visualizing which functions of the model are covered and which are missing in the story assists the user in creating a well-formed story. Skills which can be learned in the scene can be added to the annotation, along with skills the user should possess as prerequisites for the presented learning content.

**Configuring stages**

Having defined the overall structure of a story, the user can continue by configuring the details of each scene. The first step towards this lies in defining which objects are placed in this scene. The Stage Editor features a view of the current scene in which the objects featured in the scene are visible in a WYSIWYG fashion. Users can drag objects from the Resource Center and drop them onto the Stage Editor or into the scene visualization in the Story Editor.

Objects will then appear and can be manipulated in both editors, with the Story Editor just displaying an abstract visualization. Stage Editor plug-ins are not required to follow an exact WYSIWYG approach. For example the Stage Editor developed for the 80Days project abstracts from the 3D gameplay by offering a 2D map on which geographical locations can be placed and interactions with them can be defined.

**Defining actions in scenes**

After deciding which objects (such as virtual characters or props) will take part in a given scene, authors can define the logic that governs the flow of events in this scene. In comparable tools this process is often...
realized either using a scripting language such as Python or a visual programming approach as in Kelleher and Pausch (2006). Since the design of StoryTec is primarily geared towards non-programmers, the latter variant was chosen. In the ActionSet Editor (see Figure 27), the user can enter the sequence of events that will occur once this scene is encountered in the runtime environment by adding actions, represented by boxes, into a tree structure. Actions are always applied to objects which the author placed in the context of the current scene, therefore, only those objects are available in the ActionSet Editor. Furthermore, authors can trigger the transitions drawn in an earlier step, indicating that the active scene at runtime should change.

In order to be able to react to user input and other events, Stimuli are added to the scenes as a special object type which can also have attached actions.

![Figure 27. The ActionSet Editor of StoryTec.](image)

**Testing: Iterative design**

After the first phase of authoring a story-based game has been completed in StoryTec, the game can be exported into the ICML format and be tested in a player application, for example in 80Days using the Nebula2 game engine as player. During testing, various information are saved into the Context Database, including the actual time users spent in a certain scene and the events which took place during a session. This information can then be imported back into the authoring system, where it is consolidated and can be used, for example, to update the
expected time of scenes to be closer to an actual play through. This allows an iterative design process – compare with Fullerton et al. (2006) – using which authors can improve their stories over several testing cycles. The approach of iterative design of games or stories is further supported by offering text-to-speech-functionality to all plug-ins in the system.

‘Bat Cave’ Demonstration, Testing and Evaluation Platform

In addition to the prototype game Lizard, we developed a mockup player named Bat Cave (Figure 28). The purpose of this platform is to link the intelligent technological solutions and services that are running in the background of a game and that cannot directly be observed with a user interface. The idea is to have a platform that allows observing the interaction between player and system as well as testing and evaluating the effectiveness, robustness, reliability and most importantly the validity of assessment and interventions.

Bat Cave features a ‘player’ part, which is used to interact with a game that is displayed in a prototypical fashion, and a testing and evaluation part, in which the information gathered during playing such as the history of visited scenes, the current state of variables, or the current
NGLOB-related data, such as the currently measured player model, are shown to the user. Using this tool, a user can both quickly test the game under evaluation in order to find out how changes that were made to the game affect it as well as evaluate the effects of metadata used to control the adaptive algorithms of the 80Days/Bat Cave-platform. As an example, an author might find that macro-adaptive decisions are made too early in the course of the game, when not enough data about the player such as the knowledge state of the player has been collected. More information about Bat Cave can be found in Mehm et al. (2010).

Conclusion

In this chapter, the concept of Narrative Game-Based Learning Objects was introduced as a basis for creating adaptive and personalizable Story-based Digital Educational Games. Grounded in the fields of storytelling, gaming, and learning, these objects incorporate all fields which are important for creating and realizing adaptation in such games.

Since the concept of NGLOBs is new for experienced story authors and game designers as well as educators, tools can be advantageous for introducing this concept to these groups of authors. Therefore we introduced both the area of authoring tools, exemplified by the StoryTec authoring tool, as well as the area of Rapid-Prototyping and Evaluation tools, with the ‘Bat Cave’ (testing and simulation) platform and application as an example.

While this chapter focused on the theoretical background of the mechanisms found in adaptive educational games, the technical realization of these concepts in the 80Days software platform for Digital Educational Games is described in the chapter “Realizing Just-in-Time Personalization – A Technology Overview” (p. 105).

Further RTD activities in the field of adaptive, Story-based DEG will be investigated and further cultivated towards collaborative/multiplayer scenarios (Wendel et al., 2010), effective authoring mechanisms as well as the transfer and use of the developed methods and concepts to other application domains such as personalized serious games for health (Göbel et al., 2010).
Realizing Just-in-Time Personalization –
A Technology Overview

Owen Conlan, Cormac Hampson, Kevin Koidl,
Stefan Göbel, and Florian Mehn

Digital Game-based Learning (DGBL) strives to maintain a balance between learning effectiveness and the inherent motivational experience of playing a game. The main challenges lie in ensuring the learning offered is embedded in the gameplay in such a way that it does not interrupt the flow of the gaming experience. This may be achieved by tightly integrating the gaming environment with the educational material, but such tight coupling leads to poor reusability as the gameplay, story narrative, and learning material are difficult to separate. In 80Days the responsibilities for gameplay, storytelling, and learning are separated into three engines that interoperate to deliver a cohesive learning experience through a compelling gaming setting.

The learning offered in 80Days is tailored to the learner’s individual skills and needs. This form of dynamic personalization presents several technological challenges, such as implicitly acquiring information about the learner, cooperating with the story engine to ensure the narrative evolves to accommodate appropriate learning objectives, and performing adaptations in a timely and consistent manner.

This chapter focuses on the operation of the Learning Engine, specifically how it models learners and make adaptive recommendations to the Game Engine. The chapter also discusses the interoperation between the Learning Engine and Story Engine to ensure the game meets the needs of the learner.

Challenges in Realizing Personalized Digital Game-based Learning

Producing a game that is both motivational and educationally sound presents a number of significant challenges, the primary of which is
ensuring that the pedagogical premise of the educational elements of the game and the gameplay elements are sufficiently intertwined to present a holistic experience. A common failing of Digital Game-based Learning has been the lack of integration of these elements. This often leads to a fractured experience for the learner with the educational elements appearing as dull interludes in the otherwise exciting game.

Simply combining traditional technology enhanced learning techniques with a game does not automatically produce a motivational learning experience. The flow of experience is often interrupted by the learning material, which appears to be flat and non-interactive when compared with the gaming experience. However, there is a compelling reason for combining games and learning and it stems from the inherent motivation that most games inspire. The learner is driven to complete the game by the feeling of satisfaction that comes with successfully mastering the in-game challenges. Every computer game exhibits some form of learning activity, albeit quite informal. For example, a player must learn the controls of the game, details of the game setting, and how to interact successfully with non-player characters (NPCs). As these skills are learned as part of the gameplay and are integral to success in the game, players are willing to invest time in learning them. The vision for successful Digital Game-based Learning is that the educational skills learned through the game should be seen as just as integral to successfully completing the game.

Assuming that an appropriately compelling means can be found to integrate the educational skills and game skills, there exists a fundamental problem – no two learners have exactly the same needs or prior learning experience. ELearning typically suffers from high dropout rates. This is for a variety of reasons, but the material not meeting the learner’s needs and their lack of motivation rate highly for not completing a course. If appropriately combining eLearning material with an engaging computer game helps to resolve the motivation issue, then how can the material not meeting the needs of the learner be addressed?

Adaptation and personalization technology offer the potential to tailor an offering to meet the needs of a user. Adaptation techniques, such as those proposed by the Adaptive Hypermedia research domain, often construct an information offering for the user by combining several smaller pieces together. It is imperative that a sound narrative is followed to ensure that the pieces form into a coherent whole. In addition
Realizing Just-in-Time Personalization

to a sound theoretical foundation, adaptive education requires quite demanding technological realizations of theories and models. Over the past number of years, a variety of intelligent and adaptive educational systems have been introduced. To date, these systems have focused largely on adaptively ordering and presenting learning tasks.

Intelligent, adaptive, and personalized tutoring systems were developed by different researchers; a detailed review on such technologies was published by Brusilovsky (1999), general reference frameworks are described, for example, by De Bra (1999) or Albert and Mori (2001). Techniques of adaptation and individualization are primarily adaptive presentation, adaptive navigation support, and adaptive problem solving. In the framework of the ELEKTRA project (Kickmeier-Rust et al., 2006) a new terminology was introduced because game-based approaches to learning are substantially different from traditional eLearning approaches. The new concepts, which are tailored to learning environments with large degrees of freedom, are adaptivity on macro and micro levels (Kickmeier-Rust & Albert, 2010; Kickmeier-Rust et al., 2007). Macro-adaptivity refers to traditional techniques of adaptation such as adaptive presentation and adaptive navigation on the level of learning objects (or learning situations in a DGBL). Generally, macro-adaptive interventions are based on a fixed learner model (e.g., traits) or adaptation model (e.g., pedagogical implications) and on typical (knowledge) assessments (via test items). Micro-adaptive interventions, on the other hand, are non-invasive (meaning that an overall narrative is not compromised) and affect the presentation of a specific learning object or learning situation.

Within Digital Game-based Learning a different approach to that seen in intelligent tutoring systems has to be taken since the learning tasks are profoundly embedded the narrative of the game. As the game has its own storyline, any personalization offered must remain cognizant of this narrative structure whilst presenting the educational material in a meaningful manner. Thus, simply reordering learning tasks in a game may make educational sense, but would most probably result in an implausible rearrangement of the game’s narrative plot elements. Therefore, existing approaches to intelligent and adaptive educational technology are inappropriate for gaming learning environments. Due to the nature of immersive DGBL the adaptation within such games needs to be continuous and less periodic. This issue can be resolved by integrating micro-adaptivity into the environment where adaptation
occurs within the various learning situations as opposed to around them. Micro-adaptivity creates challenges of its own due to the nature of the experience of game play and the impact that game world changes can have on a player’s experience.

Micro-adaptation allows the details of a learning situation to be tailored to the specific needs of a learner. Macro-adaptation allows the sequence of learning situations to be reordered and more learning situations added to accommodate more learning objectives. This form of adaptation requires close coordination with the storytelling elements of the game to ensure the story remains coherent. Both of these forms of adaptation may be performed dynamically as part of the game. The macro-adaptation tends to happen on a slightly longer time frame as the need for more or fewer learning objectives emerges. The micro-adaptation happens rapidly as the learner interacts with the game. In both forms the information about the learner and their needs must be inferred through observing their behavior and interaction with the game. Interrupting the game to explicitly query the learner about their needs would break the immersion of the game and disrupt its flow.

There are a number of technical challenges that must be addressed in order to introduce micro- and macro-adaptation into DGBL.

1. The modeling of the learner must be implicit and time efficient. It should be possible to model several facets of the learners, such as their knowledge, gameplay skills and motivation in a timely manner.

2. Micro-adaptation should be able to personalize a learning situation to meet the needs of a learner. These adaptations again should be timely and consistent with the story elements of that situation.

3. Macro-adaptation should able to introduce more learning objectives into the game in order to meet the learner’s evolving needs. The introduction of these learning objectives may require the modification of the storyline to ensure a contiguous narrative experience.

The key aspects of these challenges are timing and consistency. The learner should not be aware that the game is adapting to meet his or her needs, other than a sense that it is appropriate for him or her. If the gameplay slows down or behaves in an unexpected or inconsistent
manner then the adaptations begin to produce a negative experience for the learner. The danger is that immersion may be broken and the flow of experience disrupted.

**Introducing the Story Engine and Learning Engine**

In order to ensure that consistency is not compromised the components responsible for the learning and storytelling need to work in tandem. In 80Days the components responsible are the Learning Engine (LE) and Story Engine (SE), respectively. They are separately implemented and maintained logic engines that operate within their respective tasks. However, it is important that they cooperate closely to ensure this consistency is maintained. Broadly speaking the Story Engine is responsible for ensuring the sequence of learning situations chosen produces a sensible storyline. The Learning Engine is responsible for the micro- and macro-adaptation decisions that pertain to learning. These may impact the sequencing of learning situations when it is determined that more learning objectives are required. The LE has two main areas in which it performs adaptations – skills and motivation. Skills relate to learning outcomes that may be gained by the learner through a learning situation. Motivation relates to the degree of immersion and involvement experienced by the user whilst learning and playing through the game.

In order for the LE to successfully adapt to the learner’s needs it must have a model of that learner. Evidence comes from the Game Engine, via the Story Engine to the Learning Engine and this evidence must be interpreted to build a coherent model of the learner. Figure 29 shows a simplified version of the 80Days architecture with the separation of the three main engines. In order to build a coherent model of the learner the LE has two components – the Skill Assessment Engine (SAE) and the Motivation Assessment Engine (MAE).

![Figure 29. Simplified 80Days Architecture.](image-url)
These sub-engines must, in a timely manner, determine the skills the learner is acquiring and the degree of motivation from the evidence received. This interpretation stage must also gather information about the current game state and adaptations that have already been triggered. This information is important to ensure consistency and appropriateness of future adaptations that may be recommended. This is just one of four stages followed by the LE. These stages are discussed in more detail in the next section.

In 80Days the effectiveness of the different engines were assessed in two demonstrators. The first, known as Lizard, was developed through several iterations and was evaluated in authentic learning settings. Its primary focus was to assess the appropriateness of the micro-adaptations offered through a compelling and appealing game. The second demonstrator, known as Bat Cave, was a technical tool used to assess the potential of macro-adaptation to adapt to the changing needs of a learner by adding additional learning objectives to a game. This tool was assessed by validating that the additional learning situations added to cover these learning objectives remained consistent to the storyline being presented. Lizard involved integration of all three main engines, Game, Story and Learning, whilst Bat Cave focused predominantly on the latter two.

The Four-Stage Approach to Just-in-Time Personalization

The nature of the game evidence sent from the Game Engine (GE) is game specific and consists of player actions, movements, and task successes or failures. This information however is not immediately useful for educational adaptation, requiring a degree of inference by the Learning Engine (LE). Inference within the LE is the first step in the Four-Stage Approach to Just-in-Time Personalization employed to provide effective non-invasive adaptation. The four stages employed are inference, context accumulation, adaptation constraint, and adaptation selection. Further details on the background to the four stage approach are detailed in Peirce (2008).

The design of the LE and the four stage approach allows for the educational adaptation to be performed without regard for the game specifics. The LE effectively infers and abstracts game actions into educational evidence that can be reasoned over in a generic manner, thus
enabling it to be employed for different games with minimal alteration. A key example of this is the abstraction of skills provided through the Skill Assessment Engine (SAE). The SAE effectively maps user actions within the game to skill evidence, and further generates a probabilistic skill model for the learner. More information on the SAE is detailed in the next section.

The second stage of the adaptation process involves accumulating game and learner evidence. In consideration of the large quantity of evidence accumulated, potentially dozens of items per second, the use of XML based models, a traditional approach in many Adaptive Hypermedia Systems such as APeLS (Conlan, 2004), becomes impractical due to manipulation and reasoning speed. Consequently all data is accumulated in a working memory provided by the Drools rule engine. The use of the Drools rule engine provides an efficient means to reason over large data sets using declarative logic.

In order to perform adaptation within the GE the LE must have an a priori abstracted understanding of the adaptations possible. Within the LE these adaptations are negotiated with the Story Engine (SE) and are represented as Adaptive Elements. An Adaptive Element consists of an identifier which represents the corresponding skill and type of intervention such as “B2337_CE” (B2337 being the skill identifier and CE being the identifier for the intervention type). Based on these identifiers the SE can decide which dialog should be rendered in the game engine. An example Adaptive Element in the 80 Days would be the Non Player Character (NPC) Feon giving a cognitive hint such as, “Okay, that’s Budapest – the capital of Hungary.”

The following are the benefits of using Adaptive Elements:

- Educational adaptation does not need to be concerned with realizing adaptations
- Facilitates the independent authoring of the game engine and the adaptation logic
- Based on SE negotiation story constraints can be considered

The third LE stage of adaptation constraint is concerned with ensuring that only appropriate Adaptive Elements are used. By using constraint rules, only feasible and appropriate Adaptive Elements are made available for selection in the final LE stage. The selection of adaptation is
An Alien’s Guide to Multi-Adaptive Educational Computer Games

achieved through adaptation rules that examine the accumulated learner data and the available Adaptive Elements.

**The Skill Assessment Engine**

Knowledge Space Theory (KST), introduced by Doignon and Falmagne (1985), provides a theoretical framework within which the knowledge or skill state of a learner can be determined. It is based on a prerequisite skill structure that describes the relationships between different skills. For example, a learner should typically be able to multiply whole numbers before he can multiply decimal numbers. If the learner exhibits evidence of being able to multiply decimal numbers it may be assumed that he can also multiply whole numbers. Such probabilistic reasoning enables a system to infer a learner’s skill state based on partial evidence (Conlan 2006).

The fundamental approach taken in KST is to reduce the number of possible pieces of evidence needed about a learner to an optimal set. In this way the Knowledge State of a learner may be assessed through the minimum number of inferences, thus achieving maximum efficiency. This is only possible by examining the domain in which the learning is occurring and identifying the underlying prerequisite relationships that exist between concepts. This is a time consuming and expert task that involves describing a learning domain, such as mathematics, in terms of formal prerequisite relationships.

Specific educational tasks, such as the learner interacting with a virtual experiment, are broken down into specific sub-tasks. Success or failure in these sub-tasks forms evidence that facilitates the probabilistic update of the learner’s model. The certainty is dependent on the level of inference required. However, as only partial evidence is needed to assess a skill state it can be done very efficiently. When applied to DEGs KST has the potential to provide the basis of a time sensitive approach to modeling a learner’s acquisition of knowledge and skills.

Interpreting evidence sent by the Game Engine (GE) is central to the first inference step of the four stage approach to Just-in-Time Personalization. The Skill Assessment Engine (SAE), a component of the Learning Engine (LE), is responsible for translating a learner’s actions within the game into a list of probabilities that show the likelihood of a skill having been acquired by the learner. This assessment of a learner’s
Realizing Just-in-Time Personalization

...skills must be done in an implicit fashion so as not to negatively impact their flow through the game.

The domain specific skills to be acquired in the 80Days game were organized according to KST into a prerequisite knowledge structure, which was represented as a binary matrix and then parsed by the SAE at design time. The resultant skills file was then loaded into the runtime component of the SAE. Previously the SAE was limited to working with modestly sized knowledge structures due to scalability issues with the initial algorithm implemented. This meant that in the ELEKTRA game (Peirce, 2008) the SAE was limited to a knowledge structure of just 25 skills. However in the 80Days game a more efficient simplified updating algorithm was developed and incorporated into the SAE. This meant that knowledge structures of much larger sizes could now be updated efficiently at runtime, with the 80Days game containing a knowledge structure of 156 skills. Furthermore, the current version of the SAE can accommodate knowledge structures much larger than this if required.

During the game, the user faces various learning challenges, with specific educational rules triggered depending on their interactions with learning objects, such as maps and flood prevention simulators. Learning objects are traditionally seen as static pieces of content, usually HTML, with associated metadata. In 80Days a learning object is an interactive experience that is woven into the game narrative. Each learning object has skills associated with it, thus if a rule relating to a learning object is fired through a learner’s interaction with the game, the SAE runs its algorithm to determine which skills have increased or decreased in probability.

If a user is performing badly at a specific task then the associated skills will have their probabilities decreased in the SAE. If these probabilities drop below a specified threshold then a related intervention may be triggered in the game (typically through NPC dialogue or in-game display) to help the user overcome this task and acquire the relevant knowledge. Likewise if a user is performing well at a task, the associated skill probabilities are increased within the SAE. If these probabilities exceed a pre-determined value, the user is then deemed to have acquired these skills.

These calculations must be done in less than 200ms (MacKenzie, 1993) so that the delay in the LE selecting an appropriate intervention for the
GE is not noticeable to the user. For the purposes of the work presented here, below 200ms is the definition of *Just-in-Time*. The adjustment in skill probabilities is taken into account in stage two of the four stage approach to *Just-in-Time* Personalization, where all evidence from the game and user is accumulated. Thus any change in skill probabilities has influence over which adaptive interventions are eventually presented to the user within the game environment.

**The Motivation Assessment Engine**

As illustrated in the previous section, the assessment of the learner’s skills during the runtime of the 80 Days game is essential for the Learning Engine (LE) to recommend appropriate cognitive interventions to assist the learner. This selection of interventions is based on the inferred skill state or knowledge of the learner. In addition to the cognitive assessment within the LE, 80 Days supports the assessment of motivational aspects also. This functionality required the development of a separate engine to sit alongside the Skill Assessment Engine (SAE) called the Motivation Assessment Engine (MAE). The MAE recommends motivational interventions as it assesses the learner’s attention and confidence. Similar to the SAE, both of the motivational aspects, attention and confidence are managed in a probabilistic model.

In order to assess the motivational aspects of the learner the Game Engine (GE) sends messages that relate to their specific actions within the game, such as changing the flight direction of the UFO. The LE can then use the time stamp of these actions to assess the motivational aspects of the learner, e.g., it can infer low attention if the learner receives a cognitive recommendation but does not react to it within a specific time frame. This can result in the LE sending a motivational recommendation to prompt the learner to react if the MAE’s value for attention drops below a predefined threshold. In addition to the timing of specific actions, the skill probability updates are used to indicate motivational aspects. For instance, a continuous stream of actions leading to a decrease of skill probabilities can be used to infer a lack of user confidence and attention and can prompt a motivational intervention.

Finally, the update history and probability reflecting the learner’s attention is used to assess if the game’s pace should be changed. A game pace change recommendation does not manifest in a dialog, such as the cognitive and motivational recommendations, but allows the game to switch to a faster game pace, e.g., from a relaxed pace to a driven pace,
Realizing Just-in-Time Personalization

or from a driven pace to a hectic pace. The result of a change in the
game pace could be the introduction of time limits, or specifically in
80Days, the introduction of additional UFO’s within the game that
apply pressure to achieve the mission goals as fast as possible.

To conclude, the MAE uses three different types of evidence to assess
the motivational aspects of the learner: first, the time between different
actions within specific learning situations, such as changing the flight
direction of the UFO after receiving a cognitive hint or remaining idle
for a specific time frame; second, the amount of skill updates leading to
a probability decrease of at least one skill. Finally, the update history of
the motivational aspects can lead to a change in game pace.

Personalizing the Story

**Story Description Language – ICML**

In order to describe interactive stories and Storytelling-based applica-
tion scenarios, much research has been invested into the development
of the ICML format (INSCAPE Markup Language). ICML was devel-
oped within the INSCAPE project (Balet, 2007), has successfully been
used in other projects and was further advanced for Story-based DEG’s
in 80Days.

The global aim of ICML is to provide not only a standardized and
comprehensive description language for a specific storytelling frame-
work, but also a good basis for a standard for any Interactive Storytel-
ling applications in general. Used as the underlying data format for the
Storytelling platform/framework, ICML describes the entire structure
of the story in a declarative way. In its first version (ICML V1.0), each
ICML document was separated into three top-level nodes, which dis-
tinguish different levels of narrative structures and their building primit-
tives: content, story and strategies.

The content node contains a global listing of logical content elements
which are part of the story. The mapping to their physical source (e.g.,
image URL) is done by the framework. Each content element node
includes a type description (sound, image, 3d model, etc.), the actions
(‘play’, ‘walk to’, ‘talk’, etc.) and properties.

The story node encodes the ‘state model’ of the story, including all
scenes/complex scenes and all possible transitions between scenes.
Every scene contains a stage set and an action set node. The stage set contains references to all content elements which are part of the scene and their size, position, etc. in that specific scene. In contrast, the action set node contains the high level story logic expressed as condition/action rules.

The strategies node contains a list of strategies. A strategy tells the system about corrective actions to be taken in case of a specific performance situation. In other words: it is a rule-base for triggering some methods that may influence the story flow at any point in time during the entire performance of the story.

A strategy is comparable to an action set, with the difference that it describes global, story wide rules and doesn’t refer to a particular scene. The strategies concept has been introduced within INSCAPE’s Story Pacing research topic whose main purpose was to guarantee a fluent and suspenseful experience in the performance of a highly interactive story by assigning the author certain control over time conditions and duration.

**Story Engine**

The Story Engine is responsible for executing and performing the interactive stories. The Narration Controller (Story Engine) forms the core of the overall runtime system. It interprets the created stories encoded in ICML described above and executes them by building an executable story graph out of it. During runtime, the graph is traversed by following the transitions from scene to scene – being evaluated at each step of the Narration Controller’s main loop. If a transition is executed, the Narration Controller is responsible for informing the Player component (the Game Engine in the 80Days approach) about a change of the current scene. On the other side, the Player has to inform the Narration Controller about user events (game evidence from the GE), which has to be processed by the Narration Controller and/or directly passed to the Learning Engine (LE). Taking into account these user events, the Narration Controller traverses the action set for every scene, evaluates the conditions, and informs the Player/GE about the actions to be executed.

How does a story continue at a specific moment? In order to answer that question the central idea is to use the concept of priorities. This means all the aspects discussed in the previous sections are considered,
Realizing Just-in-Time Personalization

and a set of rules conceptualized. These rules are used by the Narration Controller to find the ‘best’ next state and to decide which scene is loaded next.

Hence, the key challenge is to find a fair balance between the initially created story and the ‘exceptions’ caused by user interactions (unforeseen or at least not intended by the author). As an example, in the remainder of this section a tour through a museum is used. In this case, examples for such exceptions are wrong paths (not following the instructions of a virtual guide), skipped stations (passing artifacts without interacting), too long or short interactions at artifacts (causing problems with external and internal time constraints), etc. The following list provides some of the rules, ordered by priority:

1. External time constraints have the first priority. In the reference example of a story-driven museum tour for young visitors (each pupil explores the museum on his own; no guided tour in a group) the typical period of a museum visit represents a good example of such a fixed external time constraint. A teacher is unlikely to just say ‘let’s enjoy the interactive tour, walk around and explore the world of dinosaurs by yourself’, but rather add constraints such as ‘let’s meet all together in 1h at the same place’.

2. Dramaturgic aspects and story models. Correlating to the characteristics and the rules-of-thumb story models and narrative structures, plot points should be set at specific times, introductory explanations shouldn’t take too much time, the story climax shouldn’t be acquired too early, etc.

3. Importance of content and individual story elements. As far as individual story elements (scenes, specific dialogues, content etc.) are concerned, they are attributed by the author with an indicator for importance with the higher weighted elements preferred. For instance, an author might classify the importance of specific dialogue fragments of a chat station as ‘very high’, since it provides the answer to a leading question, or ‘essential knowledge’ which the pupils should take out of the museum visit. In contrast, background information about the artist of a painting might be classified as ‘interesting’ and should only be selected/visualized to the user if there is enough time for it.
Cooperation between the Story Engine and Learning Engine

The communication between the Story Engine (SE) and Learning Engine (LE) is bi-directional. To help support this communication the SE sends a specific ID together with a parameter as a message to the LE. Based on the ID and the parameter, the LE can determine the Learning Situation, Learning Action, and, in appropriate cases, the success or failure of the Learning Action. The received information is then used to perform updates in both the Skill Assessment Engine (SAE) and Motivational Assessment Engine (MAE). If specific skill probabilities or motivational probabilities, such as attention and confidence, reach a predefined threshold, the LE sends a message with an ID and parameter relating to a recommendation. The SE then receives this message and determines the exact dialog match from the recommendation sent by the LE. This avoids the possibility that the same dialogue is displayed twice to the learner. After the SE has sent the selected recommendation dialogue to the GE, the LE is informed about which dialogue was selected. This information is used for a further probability update on the related skill (as the user will have received a hint relating to the skill), although this specific update will not prompt the sending of a further recommendation.

It is important to note that this communication has to be very fast in order to provide the appropriate recommendation at the right time. The flying mission within the 80 Days game is one example in which a recommendation which arrives too late could be confusing to the learner. For example if the learner is flying in the wrong direction for a specific time period the LE will issue a recommendation. However if the recommendation arrives too late the learner may have already corrected the flight direction which would mean the recommendation would be incorrect.

To avoid timing issues the interface between the LE and the SE is based on a TCP/IP interface which supports bi-directional sending and receiving of game evidence and adaptive recommendations in milliseconds. The TCP/IP interface also provides means to request the skill state of the individual learner, or if needed the probability of a specific skill. The interface between the SE and the Game Engine (GE) is built on function calls allowing a seamless integration of the SE with both the GE and the LE.
Meeting the Challenges of Game-based Learning

This chapter has introduced three main technical challenges of Digital Game-based learning, namely, implicitly modeling the learner, providing detailed micro-adaptations to adjust individual learning situations, and offering macro-adaptation that can adjust the number of learning situations in order to introduce more learning objectives. Across these challenges timeliness and consistency must be adhered to. In other words, everything presented to the learner must remain coherent, flow naturally and be offered without any slowdown in the gaming experience. The Learning Engine has been discussed as the component of 80Days that is responsible for meeting these challenges. It achieves these by cooperating with the Story and Game Engines. Specifically, the cooperation with the Story Engine must ensure that macro-adaptation is performed in a manner consistent with the storyline being told.

Modeling the learner is divided into two sub-components of the Learning Engine – the Skill and Motivation Assessment Engines. The chapter has described how these engines operate in order to determine information about the learner in a timely manner. Moreover, it has discussed the four-stage process that takes this modeled information and recommends appropriate adaptive interventions to be rendered to the user in the Game Engine.
Integrating External Content: A Possibility to Reduce Development Costs

Lorenzo Oleggini, Sam Nova, and Ivan Orvieto

Similarly as is the case for commercial computer games, the realization of competitive Digital Educational Games (DEGs) implies a high level of costs. The specific focus of DEGs on selected (e.g., national or regional) curricula or on limited age groups restricts, however, in a significant way the potential market of such kind of games and, therefore, the possible related profit. The consequence is an increased difficulty in finding publisher willing to invest money in DEGs.

In order to address this specific challenge related with the development of immersive DEGs, one of the focuses of the 80Days project has been the identification of approaches leading to a less cost-intensive development of DEGs. Beside the specification of modules and frameworks for adaptive intervention and storytelling, one of the main strategies to achieve cost-reduction consisted in the integration of already existing learning resources in the implementation of the DEG. In particular two different aspects of the integration of existing external resources have been considered: on the one hand the integration of (learning) material such as images, videos, websites etc. in form of game assets; on the other hand the utilization of existing geographic datasets for the automatic generation of the virtual environment using traditional computer games’ rendering engines to be used as a stage for the developed DEG.

In both cases the specific generation of the corresponding content by computer graphic artists and/or developer becomes unnecessary and leads to the reduction of the production costs and of the development lifecycle.

The present chapter describes the challenges and solutions of the utilization of existing external material as a game asset or for the generation of a virtual environment. In a first sub-chapter an analysis of the situation in the computer game market is presented highlighting characteristic and difficulties of DEGs. In the following sub-chapter the strategies for the integration of external resources are presented.
An Alien’s Guide to Multi-Adaptive Educational Computer Games

Costs of Developing DEGs

Development of DEGs is for some aspects similar to classic Video-games development, even if with some specificities. A videogame can be defined as art and technology living together (with variable proportions) in an interactive product. Creation of DEGs introduces an additional element to the equation: the learning aspect.

Factors that influence development costs of Videogames and DEGs are manifold, among which we can mention length of gameplay (usually expressed in hours), game type (racing, First Person Shooter, Role Playing Game, etc.), graphical quality, target platform(s) (consoles, PC, mobile, web, etc.), geographical distribution (single country or worldwide), number of languages implemented, quantity of CG movies integrated in the game, etc. Addressing such requirements impacts on size of the development team (from few persons to hundreds of people), duration of the development phase (few months or several years), software and hardware to be bought, buildings to be run to allow people work, or travels to be done.

In addition to this, for developing DEGs we have to consider the above mentioned learning aspect that, from a cost of development perspective, is usually an extra effort in order to have learning experts participating to the production phases, to have learning resources available to be integrated in the game, and to have a specific target group (students) to be involved during testing phase of the game.

In order to give an overview of the current (2010) situation of costs for developing a so called “Triple A” game (best games available on the market), it is interesting to mention that such costs are planned to be recovered with the sales of the game itself, but profit is expected only with sequels of the game, made with (almost) same technology of the first one.

If a Triple-A game can cost more than 100 M$ (as in the case of Rockstar’s GTA IV), this doesn’t mean that lower budgets bring necessarily worse or less successful games; the matter is to find the right balance between costs and profits (as always) that can be obtained with selling copies, by integrating advertisement, by making users paying a monthly fee to play, etc. In this sense it is very important from the very beginning to identify the reference market and to design the game accord-
Integrating External Content

Integrating External Resources: A Possibility to Reduce Costs of DEGs

As mentioned in the introductory section of this chapter one of the goals aimed by the EU funded ICT research project 80Days is the development of frameworks and methods allowing the reduction of development costs for DEGs. In particular this shall be accomplished by developing, on the one hand, new psycho-pedagogical approaches such as adaptive intervention or interactive and adaptive storytelling. On the other hand, cost-reduction shall be achieved by creating re-usable game assets principles and dynamics as well as by integrating in the game existing resources in different forms. The psycho-pedagogical innovations are described in detail in the previous chapters. In the present chapter two different aspects of the integration of existing external resources in the DEG development process are explained and the obtained results are shown. The developed frameworks and methods for the generation of a virtual environment starting from freely available geographic datasets are presented in this section, followed by the description of the inclusion in form of game assets of existing digital material.

Using Geographic Datasets for the Generation of a Virtual Environment

Premises

Powerful tools for geographic information analysis and description are Geographic Information Systems (GIS). A GIS is defined as “an automated set of functions that provides professionals with advanced capabilities for the storage, retrieval, manipulation and display of geographically located data” (Ozemoy, Smith, & Sichermann, 1981, as cited in Burrough & McDonnell, 1998, p.11). The purpose of a GIS is to allow geographers to carry out spatial analysis of geographic data and information in order to describe and study specific geographic processes. In order to do this a GIS is, among other things, developed for dealing with different geographic datasets characterized by heterogeneous coordinate systems, data structure, spatial resolution, projections, gener-
alization, etc. Even though the main purpose is to allow 2D spatial analysis, GI-Software also offers, since at least a decade, efficient 3D visualization possibilities for geographic data. As mentioned by Zlatanova, Rahman, & Pilouk (2002) tools such as ArcScene by ESRI, Imagine Virtual GIS by ERDAS, or Geomedia Terrain by Intergraph were already at the beginning of the 2000s able to satisfactorily handle geographic data for 3D visualization purposes. Nevertheless 3D visualization of terrain done with GIS is generally static or with limited fly-through possibilities and it is actually rather a so called 2.5D visualization (Raper, 1992) than a real 3D visualization due to the possibility to store only a single height value for each cell of the terrain model grid (representations of caves or overhangs is therefore not possible).

Game engines have been developed to ease the work of developing new games and also make it possible for developers to use high class rendering without having to invest in a full development team to develop specific engines. One philosophy behind game engines is the idea of avoiding reinventing the wheel each time a computer game is produced. This allows a reduction of production time and costs. There exist a great number of Open Source engines but at the same time there are commercial engines on the market often developed by games companies or by companies whose only business is to provide these engines. Game engines come in different shapes; some have limited functionalities and some others have all the features that are needed for top class games. These features include physics, rendering, terrain, AI, scripting, 2D and 3D graphics tools, 2D and 3D sound systems, localization frame works, and so forth. Due to the nature of the majority of games, game engines have been developed to handle manually created data. Entire worlds (levels) are often manually created using various custom made programs or using already existing 3D modeling software. The rendering engines in these game engines are therefore also optimized to handle this kind of data. Creating elevation data for virtual environments as large as in 80Days (Western and Central Europe) would not be possible without requiring a lot of resources if handled 'manually' in a traditional way. Modeling the whole world or even smaller parts would be a very demanding task. Converting existing (geographic) data into a format that can be used by the engines is an option but not all engines can handle these large data sets directly. Therefore modifications to the engines often have to be made. Selecting the right engine is a very important task as some engines only can handle indoor scenes; some others are very hard to extend. The license
Integrating External Content

under which an engine is released is also very important. For instance, many open source licenses require that any changes done to the engine is made available to the public; this might not always be in the interest of the company doing the modifications.

Technological advances in the gaming industry in the last years led to an increase of the rendering power of 3D game engines and, at the same time, to a reduction of the costs of the mentioned engines as well as the development of Open Source versions. If put in relationship with the characteristics of both GIS and game engines this suggests the integration of the two mentioned technologies. Several published scientific papers (Germanchis & Cartwright, 2003; Germanchis, Cartwright, & Pettit, 2007; Herrlich, 2007; Friese, Herrlich, & Wolter, 2008) already discussed these issues underlining the cartographic potential of the computer game technology. In particular the merging of the two technologies allows the reconstruction of 3D virtual environment based on real geographic data and the real time navigation of the environment as well as an increased immersive-ness thanks to the 3D computer games common effects, such as lighting and shadows.

The described premises have been considered in the initial phase of the 80Days project for the definition of the general game design to be adopted for the developed DEG. As a consequence, the consortium decided to put the focus of the external resource integration process in the generation of a virtual terrain representing Europe using geographic datasets freely available in the internet. The virtual terrain should act as the stage of the flying parts of the DEG allowing the player a new and interesting approach for the exploration of the European continent. The steps necessary to achieve the final result are described in a general way in the next sections and better explained with concrete examples from the 80Days project.

Data selection

Nowadays probably the most important and meaningful source of external resources to be used as part of the approach described in this chapter is the internet. Searching the World Wide Web it is possible to identify a large number of suitable and accessible resources which can possibly be integrated in a DEG. Besides the necessity to find geographic datasets allowing the generation of a virtual environment corresponding in terms of extent and visual appearance to the desired final result, one of the most important aspects to be considered is the copy-
right of the identified data. Not all the geographic data that can be found in the internet are free of copyrights. The most significant example is the Google Map collection of map, satellite, and terrain data. Even though this material can be accessed without having to pay any fee, this does not mean that the viewed data can be further used for other products. As explicitly formulated in the Google Maps/Earth Terms of Service:

"Unless you have received prior written authorization from Google […], you must not: […] copy, translate, modify, or make derivative works of the Content or any part thereof; redistribute, sublicense, rent, publish, sell, assign, lease, market, transfer, or otherwise make the Products or Content available to third parties […]". (Google, n.d.)

The large majority of data that can be found on the internet is characterized by some terms of use to be considered when using the data. The restrictions imposed by the copyright specifications and the terms of use are very heterogeneous and range from the necessity to pay a utilization fee for the usage of the data to the possibility to freely use the data without any conditions. Despite this range of different levels of limitations a couple of general principles regarding the terms of use of the different available resources can be expressed:

- Data produced and distributed by the U.S. government are generally in the public domain and can therefore be used without any restriction. Nevertheless, in the terms of use of the different data collections it is often asked to mention the data source as a courtesy.

- The open source community offers a growing number of geographic resources possibly utilizable without the necessity to pay any fee. Nevertheless the license model sometimes requires the publication of the derivative work under the same conditions than those of the original data. It is generally the case for the resources published under creative common license (http://creativecommons.org/). This nonprofit organization developed simple and standardized ways to grant copyright permissions ranging from all rights reserved to the public domain. In this way the author of an original work can control which use of his or her own resources is going to be allowed and which not. In detail, authors can decide conditions to be applied for attribution, further distribution, possibility to generate derivative work, or additional restrictions for commercial
purposes. This means that a commercial utilization of the final product could eventually problematic. This is not always necessarily a problem but it is an aspect which should always be considered.

Other important aspects to be considered in the selection process of the geographic datasets to be used for the generation of a virtual terrain with a computer game rendering engine includes the format of the files, the file dimensions in terms of disk space, or the spatial resolution of the data:

- Only file formats that can be submitted as input to the game engine (or to specifically prepared conversion tools) should be selected. Modern GIS offers a large number of conversion tools both for importing and exporting data so that more or less every kind of file format should be possibly utilizable. Nevertheless, in some cases specific know-how and experience in handling geographic datasets with such tools may be necessary.

- Due to the continuously increasing storage spaces of hard disks, to the improvement in the data transfer capacities, and to the high quality of some of the nowadays available maps and satellite images, the file dimensions of geographic datasets is often in the order of magnitude of several hundreds of megabytes to some gigabytes. This should be considered not only in terms of necessary disk space but also in terms of amount of data manageable by the computer game engine.

- The spatial resolution (smallest area or element recognizable as a separate unit, in case of raster images, the pixel dimension) of the data available for free in the Web is, in the majority of the cases, not comparable with the spatial resolution we are confronted with when looking at geographic data visible through Google Maps, Google Earth, Bing Maps, etc. As an example, the better level of details of the Landsat satellite images offered by the NASA is 20m. This means that since pixels are characterized by 20m by 20m side length, nothing smaller than 20m can be recognized on the data. However, depending on the aimed final result, this kind of spatial resolution can be even too big. In particular the general rule “better resolution – bigger files” should be considered when choosing the dataset to be used for the generation of the virtual terrain: it does not
make any sense to download and use gigabytes of data with 20m resolution if a virtual terrain is aimed, where the player will only have the possibility to fly at very high altitude and at high speed.

- Since data are offered by different providers, the level and the quality of the generalization process will differ from source to source. Similarly errors in the processing of the data are possible from the side of the data provider, for example, in case of community generated data. These points can lead to different datasets coming from different sources not geographically matching at 100% together. Another possible cause of non-matching data is the different level of details of the different datasets. An example of non-matching geographic datasets is shown in Figure 30. It is not possible to generalize a common solution of this problem. The knowledge of its existence can help in avoiding surprises in the final result and should be kept in mind when selecting the data to be used.

![Figure 30. Example of different data sources not perfectly matching.](image)

In 80Days a large number of resources potentially useful for the generation of a virtual environment based on geographic datasets has first been identified. The data have been collected in a repository and the
characteristics (spatial resolution, format, extent, terms of use, etc.) of the different selected datasets have been registered in the form of metadata. This allows an easy and fast search and identification of datasets corresponding to certain previously identified criteria. As a secondary result of the resource identification and selection process, it has been possible to pinpoint a couple of interesting and useful providers of geographic datasets. These can be considered as a good starting point for someone interested in searching geographic data for an approach similar as the one proposed in the 80Days project:

**Global Land Cover Facility.** The Global Land Cover Facility (GLCF) ([http://glcf.umd.edu/index.shtml](http://glcf.umd.edu/index.shtml)) offers to users from the scientific, environmental, resource and disaster management, or computer science communities a collection of geographic datasets to be used for the respective different goals. In particular it is possible for everyone to download for free a large number of mainly satellite generated data with focus on earth science and global environmental systems (University of Maryland, n.d.). Specifically GLCF offers satellite imagery from the satellites ASTER, IKONOS, Landsat, MODIS, QuickBird, OrbView, and SRTM as well as a certain number of derived products (land cover maps, flood maps, etc.) and reference layers in vector form.

**European Environmental Agency.** According to their web page the European Environmental Agency (EEA) ([http://www.eea.europa.eu/data-and-maps](http://www.eea.europa.eu/data-and-maps)) is an agency of the European Union providing sound and independent information on the environment. As part of this task data and maps related with the environment are available for download and divided in sections such as air pollution, biodiversity or population, and economy. Not only data owned directly by EEA are available but also data provided by different partners such as BirdLife or the British Geological Survey.

**OpenStreetMap** ([http://www.openstreetmap.org/](http://www.openstreetmap.org/)) is an open source project founded in 2004 with the purpose of creating a freely available map of the world thanks to the single contributions of the single users. By continuous digitalization of known areas or processing of collected GPS track by common people willing to participate in the project the number of areas covered by the OpenStreetMap map is constantly grow-

ing at the same time with the precision and up-to-date-ness of the mapped situation. Even though OpenStreetMap is basically a viewer and editing tool, some data providers offer the possibility to download subsets of the data in a format readable by a GIS (e.g., http://downloads.cloudmade.com/).

Virtual terrain generation

Based on the desired final result, the geographic datasets to be used for the generation of the virtual terrain should be chosen. Generally a digital elevation model (DEM) of the rendered area can be chosen as input for the definition of the elevation of the different surface’s points. The spatial resolution of the DEM should fit to the number of elevation points manageable by the rendering engine and to the extent of the scene to be rendered. If, for instance, the engine is able to manage 4000 by 4000 different points and the goal is to generate a virtual environment of an area covering 2000 by 2000 km, the ideal DEM to be used in this case should have pixels covering an area of 500 by 500 m. The second step consists in choosing a texture to be projected on the virtual surface. For a photorealistic terrain a satellite or aerial photograph can be used. Also, in this case the spatial resolution of the chosen dataset should match the covered area and the maximal dimension manageable by the computer game engine. It is also possible to create a thematic instead of a photorealistic surface to be navigated. In this case it can, for instance, be a good idea to use the DEM as an input for the shape of the terrain (allowing therefore a certain level of recognition of the redder area’s geographic situation) and a thematic map (e.g., air pollution) as a texture to be projected on the modeled surface. Of course it is also possible to choose a dataset different than a DEM for the definition of the surface’s elevation. By choosing, for instance, unemployment rates as input for height values and GDP as the texture to be projected on the surface, it is possible to obtain navigable surfaces representing economic geography’s thematic. The number of possible combinations of elevation and texture inputs is enormous and show how important an accurate definition of meaning and purpose of the aimed virtual environment is.

The elevation input data will be, first of all, converted in a format readable by the rendering engine and then used by the engine to generate a regular grid of vertexes. The distance horizontally/vertically will be
constant across the whole grid but the height of each grid vertex is
taken from the elevation data as illustrated in Figure 31.

As it is possible to see, this grid is represented by triangles connecting
the vertexes as almost all graphics hardware works directly with trian-
gles. The texture is placed on top of this elevation grid. Each vertex
from the grid will know its position within the texture allowing a 3
three dimensional distortion of the texture. The graphics card of the
computer is able to render this data from any position and any angle.
To optimize the rendering less detailed elevation grids are used in the
distance avoiding the draw of unnecessary details when the rendered
area covers only few pixels of the display.

By submitting an elevation and a texture input to the game engine a
virtual terrain is also generated through the process described in the
previous paragraph. The result is a virtual terrain that can be navigated
in real time without loading time interruption. It is possible to change
the view point and the viewing angle at real time allowing, therefore,
the exploration of the representing topic in a similar way as by flying
over it. Figure 32 shows some examples realized within the 80Days
project. As the figure shows, both photorealistic and thematic surfaces
can be realized. Once the approach has been implemented, the time
and the efforts necessary for the generation of the virtual terrain are
very low. New terrains can be generated simply by defining new input
files.

Figure 31. Schematic representation of the conversion
of a DEM into an elevation grid.
Integrating Existing External Content in Order to Save Content Production Costs

Premises

In addition to the geographic datasets used for the generation of the navigable virtual terrain, other kinds of resources have also been considered in 80Days as possible assets to be included in the game, such as pictures, videos, or 3D objects to be placed on the terrain. The same can be done for any general type of DEG. In the Web it is currently possible to find an enormous number of freely available resources that can be possibly included in a DEG, allowing a reduction of the production costs. Pictures can be used to illustrate phenomena or locations instead of specifically designed drawings. Videos can be used for similar purposes instead of cut scenes or animations. Of course, a broad range of materials is available. In 80Days, free geographic maps for navigation purposes have been used since they were adequate for the teaching of the geography subject. It has also been possible to use 3D objects to be placed on the virtual terrain and increase the immersive-ness of the rendered scene. In both cases it was possible to save time and efforts.
necessary for the specific realization of the mentioned assets. More generally it is, for instance, possible to think about the utilization of freely available audio files for DEGs about music (songs or instruments' sounds) or biology (animals' cries). Another possibility is to use pdf files or power point presentations already prepared as tutorials. Many other utilizations of external content are possible. However some general consideration, similar to those required for geographic datasets are necessary.

Data selection

When searching for material to be integrated as assets in the DEG production workflow, one of the most important aspects to consider (beside the kind of use foreseen for the resource) is the copyright. As it is the case for geographic datasets, the use of other resources that can be found in the internet is generally defined by specific terms of use. The World Wide Web is too vast to summarize some common principle. As a starting point the considerations expressed previously can be observed. In order to facilitate the process of resource identification a couple of web pages used in the 80Days project are briefly described here:

- Flickr (http://www.flickr.com/) is one of the most known websites for hosting and sharing images on the web. Even though the users of Flickr can choose to publish their pictures labeled as "all rights reserved", there is a large number of pictures published under creative commons license. In the searching filter it is then possible to specify restrictions in order to select only pictures characterized by specific licenses. The number of results is very large also after a limitation to pictures protected by creative commons license. A search of files with the key word "Paris" and protected by creative commons license leads for instance to more than one million results. Even though the possibility to modify the pictures or use them for commercial purposes should still be checked, such a large number of results should allow the identification of material suitable for the integration in a DEG.

- A very good source of maps in the public domain is the Perry-Castañeda Map Library maintained by the University of Texas Library at http://www.lib.utexas.edu/maps/. As it is stated in
the frequently asked questions section, “Most of the maps
scanned by the University of Texas Libraries and served from
this web site are in the public domain. No permissions are
needed to copy them. You may download them and use them
as you wish. We appreciate credit to ‘University of Texas Li-
braries’ as the source of the scanned images. A few maps are
copyrighted, and are clearly marked as such” (University of
Texas Library, n.d.)

• Another important source of external material is the Wikime-
dia commons database, where more than 4.5 million files are
offered for reuse. Terms of use of are different for the different
files however “almost all may be freely reused without in-
dividual permission” (Wikimedia Commons, n.d.). For in-
stance, a lot of different maps can be found in the database by
consulting the entry ‘Atlas’ which serves as maps and cartogra-
phy portal on Wikimedia.

**Integration in the DEG**

As already mentioned previously, the different possibilities to integrate
external resources other than geographic datasets into a DEG are very
numerous and heterogeneous. Therefore, a general description of the
concrete integration process is in this case not really possible. Never-
theless, some examples from the 80Days project are shown in this sub-
section with the purpose of being a possible starting point from which
to draw on for the integration of external resources in other DEGs.
Concretely in 80Days Flickr pictures have been included in order to
show images of visited geographic locations as illustrated in Figure 33.

![Figure 33. Pictures from Flickr used in the scientific, non-profit project 80Days for illustrating the city of Bern.](image-url)
As it is possible to see, the author is attributed as requested by the creative commons license. Additionally videos from the National Geographic Society have been included for a better explanation of natural geographic phenomena such as earthquakes, volcanoes, etc. It was possible to include this material into the DEG because of the educational purposes of the final product as specified in the terms of use (http://www.nationalgeographic.com/community/terms/).

Further on, a map from the Perry-Castañeda Map Library representing the Carpathian basin has been used as a basis for a specific task in 80Days. This allowed the production team saving the manual efforts necessary for producing a similar map starting from geographic data from other providers.

Finally, external 3D objects have been placed in the navigable terrain created for 80Days and improved, in that way, the visual appeal of the scene. By placing buildings and trees corresponding to the land cover information of the city of Budapest it has been possible to represent the situation in a more appealing style without having to specifically model the scene manually in 3D. An image of the city of Budapest with the 3D objects is shown in Figure 34.

Figure 34. Area representing the city of Budapest enhanced with iconic 3D representation of the land cover.
Evaluation and Validation Methodologies for Adaptive Digital Educational Games

Effie L.-C. Law

One of the major challenges facing the emerging research on adaptive digital education games (DEG) is to develop robust evaluation methodologies to address several intertwined aspects: usability of such a game (e.g., learnability), user experience it is designed for (e.g., fun), and learning efficacy (e.g., domain-specific knowledge gain) that it is purported to bring about in users. An even bigger challenge is to evaluate the distinct feature of adaptivity, which is implemented, at the micro-level, as cognitive hint as well as motivational encouragement and, at the macro-level, as story pacing. Given the complexity of adaptive DEG environments, it is imperative to employ mixed-method evaluations to triangulate multi-source, multi-perspective, and multimodal data.

Usability evaluation of digital games can employ a combination of empirical, analytic, and model-based methods. While measuring effectiveness and efficiency (ISO 9241-110) typically in terms of task completion rate and task completion time is important, what has to be considered more relevant for adaptive DEGs are user satisfaction, motivation, and learning progress that is closely related to a range of user experience (UX) attributes such as pleasure, challenge, frustration, tension, and boredom. An intriguing observation is that evaluation of innovative adaptive systems is largely performed with traditional methods, techniques, and tools (van Velsen et al., 2008), which are limited by their underlying assumption about predictable behaviors of a system.

Furthermore, as a DEG is a dynamic learning tool, evaluation must be tightly coupled with the actual learning process (i.e., constructivist-situated learning). It is also indispensable to apply multiple measures of learning and performance along multiple dimensions: technical, orientational, affective, cognitive, pedagogical, social, and others. Hence, an evaluation framework addressing conceptual, practical, and methodological challenges in evaluating adaptive DEGs – a maturing field – should take the following aspects into account:

i) Extensibility of conventional usability evaluation methods (UEMs) for evaluating usability and user experience (UX) of interacting with DEGs;

ii) Multi-dimensionality of the learning effectiveness of DEGs;

iii) Dynamicity of adaptive systems needs to be addressed with innovative and more powerful evaluation approaches;

iv) Power of sound theoretical models for explicating phenomena in relation to learning and acting with technology;

In the ensuing text, we first present our evaluation and validation framework for adaptive digital educational games (acronym EVADEG). The initial version of EVADEG has been built upon our systematic reviews of existing approaches relevant to the evaluation of DEGs. It has been applied to the evaluation studies of three game prototypes of 80Days. These practical experiences have enabled us to refine and enrich EVADEG.

**EVADEG: Four-Dimensional Framework**

Our framework EVADEG comprises four major dimensions:

- Learning Effectiveness Validation (LEV)
- Gaming Experience Evaluation (GEE)
- Game Usability Evaluation (GUE)
- Realtime Interaction Trajectory for Adaptivity Evaluation (RITAE)

Each dimension consists of a cluster of quality attributes (Figure 35), which are evaluated with specific techniques and tools. There exist inevitable overlaps of these attributes, given their intricate interdependencies.

Individual dimensions are elaborated in the subsequent sections. Reviews of relevant literature informing the selection as well as construction of specific approaches undertaken by 80Days are presented. Practical experiences gathered through the evaluation studies of 80Days are shared whenever appropriate.
Learning Effectiveness Validation (LEV)

Two schools of thoughts concerning the feasibility of deploying digital games as an e-learning tool exist: Proponents presume that digital games can balance the monotony of traditional teaching strategies and materials, whereas antagonists argue that high quality digital games are
simply not available (Wideman, 2007). Hence, it is imperative to provide evidence about the positive effects of digital games.

Current research efforts on evaluating user experience in games address mostly entertainment rather than educational types (e.g., Bernhaupt et al., 2008), though increasing attention is given to Serious Games. It is a dual challenge to evaluate both the fun and serious aspects of games. Measuring the success of educational technology and particularly of DEGs is a complex issue. Much of the literature in the field of educational technology revealed inconsistent, and, at best, non-significant differences were found between technology-based and traditional delivery media. Specifically, our learning effectiveness validation (LEV) approach is grounded in scientifically sound criterion-based designs and comparison-based designs. The former utilize a-priori specified criteria to measure educational effectiveness (e.g., whether students learn what they are supposed to learn), whereas the latter can potentially be applied to evaluating and validating adaptive sequencing of learning material, adaptive story generation, and non-invasive assessment and interventions. Three significant factors (Lockee et al., 2002) to be included in our LEV approach are:

- **Performance outcomes** (e.g., time and efforts required for achieving educational objectives)
- **Attitude outcomes** (e.g., attitude towards the learning environment/media; motivation and interest)
- **Programmatic outcomes** (e.g., is there a return on investment? Does the learning environment reach its target audience?)

In 80Days, performance and attitude outcomes were largely measured with questionnaires tapping on the related attributes (Figure 36). Specifically, the so-called Assessment of Learning Questionnaire (ALQ) was a ‘home-grown’ artifact, drawing on meticulous analyses of knowledge and skills covered by the game. The other questionnaires were grounded in the literature (e.g., Garzotto, 2007; Poels et al., 2008; Prensky, 2001). In addition, semi-structured interviews supplementing the questionnaires data were conducted.

**Game Usability Evaluation (GUE)**

In accordance with the philosophy of user-centered design (UCD), it is imperative to involve representative users in all phases of game devel-
opment to elicit their ideas as well as feedback on design concepts and prototypes to ensure that the game meets the users’ need and is usable (Figure 37). This widely employed design and evaluation approach is known as participatory design.

**Figure 37.** Game development phases with usability evaluation methods.

Evaluation of DEGs is particularly challenging as it involves two types of goals – pragmatic (learning) and hedonic (pleasure/fun) – with the former being associated with usability and latter with user experience (UX) and each subsuming different sets of quality attributes (Figure 36). With research on demarcating UX still in progress, current methods, techniques, and tools for evaluating and measuring UX are largely drawn from the usability tradition (Vermeeren et al., 2010). Nonetheless, the two major usability evaluation methods (UEMs), namely, expert review and user-based tests, remain highly relevant to evaluation of DEGs. Subsequently, we present an overview of recent research work on PD and the two common UEMs.

**Participatory Design (PD)**

PD has commonly been used in the field of Human-Computer Interaction (HCI) and, more recently, has become a prevailing approach that enables children’s involvement in the design of technology (e.g., Danielsson & Wiberg, 2006; Vaajakallio et al., 2009). Indeed, there has been ever-increasing interest in the roles of children in the design of software aimed for them (Bruckman & Bandlow, 2003). It has led to the emer-
gence of a HCI subfield coined as child-computer interaction CCI (Read, 2005); the design of game-based learning has broadened the HCI perspective to include the pedagogical one. Thus, the learner-centered design (LCD) has also come into play (Nousiainen, 2009). Within the CCI/LCD viewpoint, different techniques have been created to involve children in the development of applications designed for them. Several studies have employed requirements gathering techniques adapted for children, including the KidReporter technique (Bekker et al., 2003), the “Mission from Mars” method (Verhaegh et al., 2006), creative techniques including drawings (e.g., Bilal, 2003), storytelling (e.g. Hall et al., 2004), collaborative low-fidelity prototype creation (e.g., Druin, 2005). According to Nousiainen (2009), involving children in the design of technological applications that are used as educational tools is deemed as a convenient means for them to give their voices in issues concerning their own learning. Presumably it will enhance and sustain their motivation to learn.

In 80Days, we have employed the PD approach. Surveys were conducted in several games fairs. Focus groups were also conducted at different schools to collect representative target groups’ opinions on initial game concepts. Results of these activities were fed into the improvement on the related procedure.

**Expert Review**

Expert review is a common usability inspection method to be performed in the relatively early phase of software development. In essence, a usability specialist systematically inspects the design of a prototype to identify any violation of heuristics (i.e., usability problem [UP]). Outcomes of expert review normally lead to elimination of severe UPs and redesign proposals. As traditional usability heuristics (i.e., Nielsen, 1994) addressing performance- and task-based design are deemed inappropriate for games, alternative heuristics have emerged, including *game usability* and *game playability*; Table 5 illustrates some examples (e.g., Desurvire et al. 2004; Federoff, 2002; Korhnone & Koivisot, 2007). It is recommended to launch an expert review to identify serious UPs before implementing user tests.
User Tests

A typical user test involves asking a representative user to carry out specific activities with the system under scrutiny whilst her performance and behavior are observed and measured. The test can take place in a usability lab or in the context where a user normally deploys the system. Traditional usability measures for effectiveness (e.g., task completion rate) and efficiency (e.g., time till event) are not particularly relevant for evaluating games which fulfill hedonic rather than pragmatic goals, though arguably such measures may be applicable to educational games. Indeed, identification of usability problems (UPs), which hinder players from interacting with the game is a crucial goal of user tests on DEGs. Specifically, in 80Days, user tests have mainly been implemented for detecting UPs and have been conducted in three formats with each of them serving an additional specific purpose:

a. Group field tests: To meet the dual purpose of collecting sufficient number of datasets for validating the learning effectiveness, user tests were conducted in groups with size ranging from 5 to 14 children, who played the game on an individual basis, in the presence of at least two researchers. UPs experienced by children were primarily reported by themselves in post-gameplay questionnaires. These data are supplemented by researchers’ observational notes.

b. Dyadic field tests: To elicit real time running comments from players, which presumably are more valid than retrospective reporting, the think-aloud technique should be employed. However, children may have difficulty to follow a standard think-aloud protocol (Dumas & Redish, 1999), and they tend to verbalize their thoughts more easily when playing the game together than alone. Dyadic tests, as the name suggests, involves pairing up two players who collaboratively deal with the game’s quests. Furthermore, as children often behave differently according to how well they know each other (Benedikte et al., 2005), it is recommended that the two participants are no strangers to each other. A researcher should be nearby to observe players’ interactions and to provide help when solicited. A video camera is mounted behind participants to record their interactions with the game. When front view videotaping is not allowed because of the data
protection act for children enforced at school, in situ observations can compensate.

c. **Individual tests.** Typical setting for user tests was arranged with some sessions being conducted at schools (i.e., field study) or some at the university campus (i.e., lab study). The latter additionally involved the use of eye-tracker to capture visual attention (see below).

Table 5. Examples of Game Usability (GU) heuristics and Game Playability (GP) heuristics.

| GU1: Provide visual representations that are easy to interpret and that minimize the need for micromanagement |
| GU2: Provide players with information on game status |
| GU3: Allows players to skip non-playable and frequently repeated content |
| GU4: The player can easily turn the game off and on, and save games in different states |
| GU5: Sounds from the game provide meaningful feedback or stir a particular emotion |
| GP1: The game should give rewards that immerse the player more deeply in the game by increasing their capabilities (power-up), and expanding their ability to customize |
| GP2: Player’s fatigue is minimized by varying activities and pacing during game play |
| GP3: Provide consistency between the game elements and the overarching setting and story to suspend disbelief |
| GP4: Pace the game to apply pressure but not frustrate the player. Vary the difficulty level so that the player has greater challenge as they develop mastery. Easy to learn, hard to master. |
| GP5: Player is taught skills early that are expected to use later, or right before the new skill is needed |

**Game Experience Evaluation (GEE)**

Game experience is an emerging research topic, which has stimulated an ever-increasing number of research studies on game experience
conceptual models and evaluation methods. In this section, we present overviews of the two respective areas.

**Game Experience Conceptual Models**

Gameplay experience can be seen as an amalgamation of a player's sensations, thoughts, feelings, actions, and meaning-making in a gameplay setting. Players actively construct their experiences: they bring in their desires, anticipations and previous experiences, and interpret, reflect and reinterpret the experience dynamically evolving during the gameplay. In other words, gameplay experience is an intricate psychological and physical construct and is something that emerges in a unique interaction process between a particular game and a particular player (Emri & Mäyrä, 2005). Considering its inherent complexity, gameplay needs to be studied systematically as a multi-dimensional and multi-layered concept. Drawing on the related literature with particular focus on children, we identify a set of so-called experience-based criteria pertaining to players' feelings, emotional, and affective responses. Specifically, the six criteria: immersion, goal, autonomy, feedback, concentration, and challenge, are subsumed by the key concept flow experience proposed by Csikszentmihalyi (1998). Methods of measuring these criteria include both quantitative and qualitative approaches, including survey questionnaire, interview, focus group, in-situ observation, diary writing, and psycho-physiological techniques.

A relevant conceptual framework known as Digital Gameplay Experience Model (DGEM) was proposed by Calleja (2007). In brevity, DGEM describes aspects of players' experience with reference to six frames (viz., tactical, performative, affective, shared, narrative, and spatial) with each of them representing a modality of meaning. Each of the six frames describes a variety of experiences, ranging from conscious attention to internalized knowledge, which will result in some form of incorporation. According to Calleja (2007), macro-involvement refers to players' general motivations for engaging with games whereas micro-involvement refers to the moment-by-moment instances of gameplay. Basically, the DGEM is a descriptive framework that enables qualitative comparisons between different instances of play. From a learning perspective, it is of particular interest to understand how players incorporate different frames, because it can give insights into the relationship between the learning and involvement experienced and the
distinction between general motivation to play games and the engagement invoked during the gameplay (Iacovides, 2009).

**User Experience Evaluation Methods**

According to ISO 9241-110:2010, user experience is defined as “a person’s perceptions and responses that result from the use and/or anticipated use of a product, system or service” (clause 2.15). Given the soaring interest in UX and its complexity as well as richness, a diversity of UX evaluation methods (UXEMs) and measures have been developed. The scoping issue - which methods and measures to be adopted in which context and for which purpose - remains a critical challenge for the UX research community. The well-recognized principle of employing the mixed-method approach is particularly relevant for UX, given that it is inherently subjective and highly dynamic (Law et al., 2009).

In the context of adaptive DEGs, UX is multifaceted and particularly complex given a cluster of interrelated aspects: players’ perceptions of the quality of the game user interface components (i.e., text, graphics, sound, video); their affective responses to the game characters, game narrative and adaptive features (i.e., hint and encouragement provided); and their subjective evaluation of the quality of the domain-specific content delivered by the game. Furthermore, players’ expectation about the game, based on their past interactions with games of a similar or different genre, is an important factor influencing their UX when playing the current one, at least in the initial phase.

A series of scientific activities over several years, ranging from workshops to literature reviews, have led to a collection of 96 UXEMs deployed in academia and industry. Descriptions of these methods with a standard template with 18 attributes are accessible online (http://uxems.shorturl.com). Analyses of these UXEMs have systematically been carried out and documented (Vermeeren et al., 2010). Specifically, results from the content analysis on the strengths and weaknesses of the UXEMs show that scientific quality (i.e., psychometric properties: reliability and validity of the related tool and process) and practicability (i.e., usability (e.g., ease of use), feasibility (e.g., equipment/expertise required) and motivation (e.g., fun) are central concerns for the UXEMs applied in different study periods, namely ‘before’, ‘during’ and ‘after’ gameplay.
In 80Days, standardized questionnaires for UX like AttrakDiff (Has-
zenzahl, 2006) and home-grown ones based on the existing literature
have been used (e.g., Poels et al., 2007). They tap UX quality attributes
such as challenge, competence, flow, immersion, negative affect, posi-
tive affect, and tension. Post-game semi-structured interviews and in-
situ observations (facial expressions, body movements) also provide
important evaluation data.

**Real-time Interaction Trajectory for Adaptivity Evaluation (RITAE)**

There have been very few, if any, validated evaluation frameworks for
adaptive DEGs, though adaptive/personalized learning has been re-
searched since the inception of intelligent tutoring systems in the mid
70s. More recent work on evaluating adaptive systems (e.g., UMAP
2009; Weibelzahl, 2005), however, is regarded as too general for digital
games. Other research work focusing on the design framework for
adaptive digital games (e.g., Tan et al., 2007) addresses evaluation issues
to a limited extent. Van Velsen et al. (2008) have recently conducted a
systematic review of 63 studies of user-centered evaluation methods
applied to personalized systems. Interestingly, the methods identified
are classic ones like control group comparison, questionnaires, interviews, focus
groups, think aloud, and data log analysis (cf., innovative but not yet vali-
dated approached identified in Vermeeren et al., 2010).

In 80Days, for evaluating the effectiveness of micro-adaptivity (i.e.,
provision of cognitive and motivational intervention contingent on
gameplay performance), we relied on continuous trails of gameplay
activities (i.e., interaction trajectory) in the form of video-records and log
files, which were verified with outcomes of the assessment of learning
questionnaire (ALQ). Log files were proved useful for evaluating adap-
tive system (Cocea & Weibalzahl, 2007).

We have coined the approach as Real-time Interaction Trajectory for Adaptiv-
ity Evaluation (RITAE) (Figure 38). Specifically, the entire game session
was captured by an eye-tracker when the test session took place in a
university lab or by a video-camera when it did at a school premise. The
Learning Engine of the game automatically generated XML log-files
which were parsed into formatted HTML files with ten fields. After the
test, we viewed the videos and meticulously tracked every intervention,
be it a hint or an encouragement, and documented what the player had
done to *trigger* its delivery and how she reacted to it. The (mis)match between the player's action and the content of the intervention indicates its (in)appropriateness. In addition, we categorized the player’s reactions to the intervention such as ignoring it, correcting her previous action, re-engaging in a game activity, etc. With such data extracted for individual players, we evaluated correlations between the number of hints delivered and the improvement scores demonstrated by in-game assessment and by the difference between pre- and post-gameplay learning assessment questionnaires.

<table>
<thead>
<tr>
<th>Data Capture</th>
<th>Logfiles</th>
<th>Servers</th>
<th>Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Eye Tracker</td>
<td>- Video Camera</td>
<td>- Game console</td>
<td></td>
</tr>
<tr>
<td>XML BLOB</td>
<td>Event set</td>
<td>- Event</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Contextual info</td>
<td>- Timestamp</td>
<td></td>
</tr>
<tr>
<td>Data storage</td>
<td>Logfile parser</td>
<td>- In-game Assessment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Triggering event</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Player response</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Contextual info</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Timestamp</td>
<td></td>
</tr>
</tbody>
</table>

*Figure 38. Real-time Interaction Trajectory for Adaptivity Evaluation (RITAE).*

**Special Evaluation Technique: Eye-Tracking**

A common goal of eye-tracking studies is to understand how visual attention manifest as eye-tracking data is related to cognitive processes (Jacob & Karn, 2003). In eye-tracking studies, fixations (i.e., moments when eyes are relatively stationary and encoding of information takes place) and saccades (i.e., when quick eye movements occur between fixations without any information intake) are two basic metrics. A mul-
Evaluation and Validation Methodologies

titude of derivatives such as scan path (i.e., sequence of fixations in a target area) are available (El-Nasr & Yan, 2006). While these eye-tracking measures are commonly used, their interpretations remain malleable.

Jennett and her colleagues (2008), who investigated the immersion in a videogame with the aid of eye-tracking technology, found a decrease in eye movements measured with the number of fixations per second in the immersive condition as compared to an increase in a non-immersive control condition. The authors claim that in an immersive game the attention of the players becomes more focused on visual components relevant to the game (i.e., fixate on a selected set of objects within a specific timeslot), and in a non-immersive activity the individuals more likely get distracted by other items (i.e., fixate at various objects within a specific timeslot). In addition, they observe that their participants tended to change the nature of the game given (i.e., alter the game rule to make it more engaging), rendering it difficult to predict fixation behavior. A similar observation was noted by Sundstedt and his colleagues (2008) in their eye-tracking study with a maze videogame. They argue that such a player-effect (we coin it) of redefining a given game-task undermines the predictive power of saliency maps. However, in a non-game situation, the validity threat posed by the player effect seems insignificant or the non-game-based task can more effectively be controlled because of its lower complexity and dynamicity (Buscher et al., 2009).

Furthermore, the eye-tracking study of Buscher et al. (2009) on the user’s browsing behavior of web pages reveals some interesting observation. They notice that the lowest right corner of a webpage attracts almost no visual attention during the first second of each webpage view, suggesting participants’ low expectations of information content on the right side of most web pages. This finding seems consistent with those of the previous studies that identify triangular or F-shaped scan patterns on web pages. However, the observations derived from webpage viewing seem not applicable to videogame sighting (Law et al., 2010).

In 80Days, we employed the eye-tracking technology to investigate how players’ visual attention to interface objects was related to the respective learning effect. Specifically, we extracted the fixation duration data and correlated them with improvement scores in terms of in-game performance as well as pre-/post- learning assessment. In addition, data
related to the most relevant areas of interest (AOIs) - like the non-player-character window - were studied in terms of a hot spot analysis (see Figure 39) and the number of fixations per second on various AOIs.

![Figure 39. Hot spot analysis of visual attention in the flying situation of 80Days.](image)

**Specific Analytic Framework:**

**Activity Theory**

Activity theory (AT) is rooted in the work of Vygotsky (1987) and his student Leontiev (1978). AT is an intricate framework that aims to understand individual human beings (i.e., subjects) and social systems they constitute through an analysis of the origin, structure, and processes of their activities (Kaptelinin & Nardi, 2006). Activity is defined as a unit of subject-object interaction with the object having the status of a *motive* that meets a certain need of the subject. Engeström (1987) expanded AT to address systems of activity at the collective level. He defined the activity system model with seven components: subject, object, tools, division of labor, community, rules, and outcome (Figure 40):

- **Subject:** an individual engaging in an activity;
- **Object:** an objective or motive held by the subject for performing an activity;
- **Tool:** material or mental agency mediating the subject-object interaction;
• Rule: regulations of actions and interactions in an activity;
• Community: one or more people sharing the objective with the subject;
• Division of labor: how tasks, power and status are divided among cooperating members of the community;
• Outcome: an idea, a situational status, or a specific positive/negative emotional response.

Figure 40. Engeström's (1987) activity system model.

From an AT perspective, the computer is the tool mediating the interaction of humans with their environment (Bannon & Kaptelinin, 2000). In the field of education, AT has been employed to study the design and implementation of learning supported by technology (e.g., Barb et al., 2000). AT can facilitate understanding of how technological advances influence change in education (Issroff & Scanlon, 2002). Roussou et al. (2006) applied AT to study the interaction in learning through virtual reality and concluded that AT is a useful framework to understand the relationship between the user and virtual environment. However, their analysis did not explicitly address user experience.

We find AT particularly promising to shed light onto the understanding of user experience, which is essentially a psychological construct determined by actors' motives and needs, which are in turn shaped by the socio-cultural context where actors are situated. In the context of a DEG, the two seemingly competing motives (i.e., learning and playing) can be seen as inherent inconsistencies within the system and discrepancies between the system and its environment (cf., a traditional formal learning setting of a classroom vs. informal game-based learning). This well exemplifies the notion of contradiction of AT (Kaptelinin & Nardi, 2006). On the micro level, contradiction can be seen as breakdown,
which has drawn much research attention in HCI since 1980s and is also broadly used in different disciplines. The breakdown analysis highlights the spontaneous, subjective problems in the use of a technology and connects them to specific aspects of users’ action. In summary, Activity Theory is a promising framework relevant to the design and evaluation of DEGs.

**Concluding Remarks**

In this chapter, an evaluation framework for adaptive DEGs is presented. It is constituted by four dimensions with each of them being defined by a cluster of quality attributes, methods, and tools. While there is a plethora of existing methods and research for the evaluation of software as well as conventional educational technology and learning media, this knowledge cannot be transformed one-to-one to the genre of DEGs (e.g., de Freitas & Oliver, 2006). One significant observation is that traditional evaluation methods are largely employed for the emerging domain. Some innovative approaches have been proposed, but their scientific quality and practicability are yet to be established (Vermeeren et al., 2010).

The investigation of the adaptive features in a narrative learning environment with a large degree of freedom is challenging the existing evaluation and validation methods. Moreover, aspects of motivation, audio-visual preferences, personality, or individual ability and their interactions with game design, learning design, usability, or narrative are a complex and novel field of research. The ambition of 80Days is to fill the gaps thus identified and develop a reference framework for the in-depth evaluation of adaptive DEGs at design time (accompanying the design and development stages) and at runtime (evaluating interim and final versions of games). Last but not the least to mention is the very significance of a sound theoretical framework that can inform the selection and application of appropriate evaluation methods, especially gaining insights into implications to be drawn from qualitative data. Amongst others, Activity Theory (AT) is seen as a promising conceptual framework, which has already been applied to various studies in the domain of DEGs (e.g., Rousson et al., 2006). Whether other conceptual frameworks such as Distributed Cognition (Salomon, 1997) are even more powerful to explicate phenomena related to adaptive DEGs remains to be a challenging research question in the area of game-based learning.
The 80Days Game

Daniel Schwarz, Lorenzo Oleggini, Christina M. Steiner, and Martin Stoecker

This chapter is a summary of the learning game 80Days as it was designed and developed in the project. During the life span of the project, there were three major design and development cycles of the game with according releases named “Lizard 1.0”, “Lizard 2.0” and “Lizard 3.0”. Each release was tested in classrooms and focus groups and the user validation results fed into the design of the next version.

The description of the design process is structured into the three categories of our objective-driven design for learning, storytelling, and gameplay. We therefore describe the essence of the learning game design process for each of these three design categories.

Essence of Gameplay Design

The player is a 14-year-old teenager boy who is hijacked by an alien named Feon. Feon pretends to be an intergalactic travel author and needs help from a human – as the most intelligent life-form on this planet – to write his travel guide about the Earth. The boy makes a deal with Feon and agrees to help him. As a return for that Feon lets the boy fly his UFO. Together they discover Earth and compile geographic knowledge about the planet during their expeditions with the UFO. This is accomplished in different Micro-Missions by the player by means of flying to different destinations on Earth and to explore the geographical features on-site.

The overall mission of Feon and the boy is to conjointly create an “Earth report” with the gathered geographical knowledge. The main goal is then to send this “Earth report” as the assumed travelogue about Earth to the employer of the alien. In the evolving course of the game’s story, the boy sees through the alien’s game (of preparing the conquest of the earth) and reveals his “real” main goal of the game: He must save the planet – and the only way to do it, is to draw the right
conclusion from the “Earth report”, which was actually planned as a betrayal of Earth to the upcoming alien invasion.

Figure 41 describes (a) the didactic approach (“Geography concepts”) of teaching Geography with the game and (b) the three resulting gameplay modes of that approach (“Gameplay concepts”). Therefore the gameplay has got two main goals:

1. The first goal is to help the alien to complete the geographical “Earth report”
2. The second goal – save the planet - is revealed in the course of the story, when the player realizes the true intention of the alien.

As a result we have two levels of gameplays associated to the two goals. On the level of goal (1) the gameplay is based on explorative missions that cover several topics of geographical knowledge as mission goals. On the level of goal (2) the gameplay is to find a solution to save the planet.

80 Days’ game metaphor for learning Geography – “The Long Zoom” (please see Chapter 2) covers both main goals ranging from an action-oriented micro level (“Micro Missions”) to the macro level of strategic thinking, reasoning and reflection (“Geography saves the planet!”).

Taking the game metaphor as a conceptual frame for the game design, the idea of the “Long Zoom of Geography” was then transformed into a concrete design for the gameplay. For this we first identified three discrete zoom levels of the “Long Zoom” that represent didactically discrete and important “levels of scale” in teaching the subject of Geography. Having those discrete didactic levels enables the didactic designer to categorize the entire Geographical learning content into the following three level of scales:

- Knowledge about a specific point on Earth and its geographic features (Geographical Site)
- Knowledge about an area and its geographic features (Geographical situation)
- Knowledge about the entire planet and its geographic features from a global view (Globe)
Teaching Geography with the Game and (b) the Three resulting Gamplay modes of that approach ("Gamplay concepts") of "The 80 Days Game" metaphor of the "zone zoom" describes (c) the didactic approach ("Geography concepts") of "The 80 Days Game".
The goal of the game designer is then to develop a concrete design for the gameplay that incorporates those three Zoom levels. By doing so we can make sure that our gameplay is able to transport all kinds of Geographical Knowledge and transform it into enjoyable and meaningful learning and gaming experiences.

As a result of this design process, the 80Days game consists of three different game plays that build upon each other and can be seen as three different zoom levels (levels of scales) of the “long zoom”.

The background story and the overall mission of the game play (“Geography saves the Earth!”) is always the same through all the levels, but there is a specific gameplay for each zoom level. These three gameplay levels could stand alone as single games, but together they form the cohesive geographic learning experience that makes 80Days unique.

In each of the three levels the player can discover planet Earth together with the Alien in his UFO, the difference is the (geographical) scale of the discovery. Throughout all three zoom levels their main mission is always to learn Geography by discovering, acquiring, reporting, and testing geographical knowledge about the Earth. In the following paragraphs the three gameplay modes are briefly described.

**Gameplay Mode “Adventurous Exploration”**

In Gameplay mode (1) “Adventurous exploration” (Figure 42) the UFO can be landed. The player and Feon disembark the UFO and can explore the location on-site by foot - just as you would discover a level/stage in a classic adventure game by walking, looking and collecting things as well as information about the place.

This gameplay mode was realized differently in the demonstrator versions “Lizard 2.0” and “Lizard 3.0.” Instead of landing the UFO, we developed a simulation gameplay about the Geographic topic “Flood” at the map desk in the UFO providing the player with action-oriented and constructivist learning (please see a screenshot of the map desk simulation in Figure 46).
Gameplay Mode “Air Race”

In Gameplay mode (2) “Air Race” (Figure 43) the player is in the cockpit of the UFO and piloting the spacecraft. He concentrates on flying and navigation covering a wider geographic area with his actions: He flies the UFO over a wider area, exploring different geographic phenomena or topics in this specific area. By skillful flying and the application of navigation he is in the process of enactive learning of Geography. The flying parts of the Micro Missions of the three demonstrator versions of the game are played in this game mode.

Figure 43. Conceptual drawing of gameplay mode “Air Race” (left panel) and screenshot of gameplay mode “Air Race” in the game (right panel).
Gameplay Mode “Global View”

In Gameplay mode (3) “Global View” (Figure 44) the player isn’t flying the UFO. Instead of that the on-board computer is keeping the UFO in a “parking position” – usually the UFO is hovering in a safe planetary orbit.

The player is – together with Feon - in the observatory of the UFO. There both can gather around the map desk in the middle of the observatory. The map desk (Figure 45) is an interactive table that can show maps and a holographic 3D-projection of the Earth. At this table the player and Feon can plan their missions and flight routes. But they also can classify and locate their collected geographical information and knowledge about Earth in the interactive 3D-Hologram of the Earth.
In fact, this interactive 3D-Model of the Earth is the storage for the report that has to be sent to star fleet command. Actually, the map desk and its Earth-Hologram IS the report about Earth. In the beginning of the game this report is just as empty as white spots on a map, but in the course of the game, the Earth model gets filled more and more with the geographical data that is discovered and collected during the missions of the learning game. Since all the world-wide collected geographical data and information can be located, classified, stored, and always viewed in the global context of the whole planet, this game play mode is called “global view”: it lets you see your collected geographical knowledge in a global context. This is important for the dramaturgical development of the plot in which the fate of the Earth is decided.

In the first version of game demonstrator “Lizard” the map desk was only realized as a 2D-map desk, mainly used for navigation, mission planning and the representation of geographical learning content in form of pictures and text.

In the second and third versions of “Lizard” the map desk was also able to display learning content in the form of videos. As a complete new feature, the map desk was enhanced to a 3D map desk simulation tool with which the player could simulate causes, preventions, and effects of the natural hazard flood by means of simulating the flood that they observed in the real world in Budapest (see Figure 46).
Essence of Didactic Design

Transforming Learning Content into a Knowledge Space

In the very first phase of the 80Days project the learning content of the DEG has been identified. To do that geography curricula of different European countries and concrete geography lessons and material have been considered. On the basis of the mentioned analysis general learning objectives and learner requirements have been defined. Only the geographic topic identified as being commonly taught to the children in the target age 12-14 have been further considered in the design process. Concretely, the following learning objectives have been suggested as possible topics of the 80Days learning game:

Basic Geographic Instruments, use of maps and globes
   A1) Learning how to use atlas, globes and maps

Basic Knowledge of the Earth
   B1) Earth as a planet
   B2) Location of places and geographic features
   B3) Weather and climate

Physical Geography Processes and Implications
   C1) Physical processes that shape Earth’s surface

The analysis’ process of the European curricula led to the definition of specific skill lists corresponding to the five sub-topics mentioned above. On the basis of single lessons and concrete teaching material, such as school books or power point presentations, learning objectives have been defined at the granular level of detail of single skills. In order to match the requisites of the adaptive learning engine developed for the 80Days project it has been necessary to explicitly define for every single skill of the list a variable number of other skills judged as being pre-requisites. The resulting prerequisite structure allows the estimation by the learning engine of the probable knowledge state of the player and consequently the submission of the most adequate learning content for the prosecution of the game. More details about the learning engine are provided in the corresponding chapters of this methodology guide book. The skill lists have been filled following the logical structure of the analyzed teaching material. Due to this reason a basic didactically sound teaching sequence of the skills was already present in the first
version of the lists. It can in fact be basically assumed that a traditional learning unit will start by introducing the basic concepts before approaching the more complex ones. This fact is very useful since it allows having a very rough prerequisite structure constituted by the skill sequencing in the skill list.

In the initial definition of prerequisite skills, the overall picture of the prerequisite structure has not been considered. In other words, the further prerequisites of the prerequisites skills have not been considered at the very beginning. This led to inconsistencies, for instance, when trying to visualize the prerequisite structure of two similar skills characterized by similar prerequisites. A concrete example is shown in Figure 47, where the skills B3046 and B3050 are represented together in a graph of the knowledge space. Since the skills B3042, B3045, B3041, B3001 and B3002 are considered necessary for both B3050 and B3046, they appear twice in the structure.

This is due to the definition of prerequisites in the initial phase being done only focusing on a single skill at a time. Due partly to this reason and partly to the necessity to attentively review the initial skill selection work, a review of the prerequisites structure has been carried out in a second phase of the learning content definition process. In particular the following points had an important influence in the revision work:

- For the elaboration and definition of prerequisites and competence representation in this document the formal framework and principles of Competence-based Knowledge Space Theory (CbKST) were used (e.g., Albert & Lukas, 1999; Doignon & Falmagne, 1999), in particular the notion and features of a prerequisite relation (a reflexive, transitive binary relation) and its representation in form of Hasse diagrams.
A more global view has been adopted in defining/reviewing the prerequisites; groups of skills characterized by the same or similar (sub-)topic have been analyzed together trying to define the prerequisite relationships between them.

Particular attention has been paid to the relationship within the different topics (e.g., plate tectonic or Earth as a planet). Less attention has been paid to the relationships between different groups of skills although the more important and evident cross-topic prerequisites have been defined also in order to allow the building of a single complete structure of the skills.

Generally, starting from a certain skill it has tried to define what previous knowledge is necessary in order to understand or learn this skill. For example, for the skill ‘B3050 rainfall has three major types’ it is evident that it is necessary to know what rainfall is. The corresponding Skill is ‘B3046 – rain is one kind of liquid precipitation’, which again requires the knowledge of precipitation (Skill B3041) and so on. This leads to a Skill structure as presented in Figure 48.

Since it is almost always possible to imagine a prerequisite skill of a certain other skill the prerequisite definition process could hypothetically continue endlessly or almost. Due to this reason we defined some rules for the prerequisite definition process in 80Days. Generally we searched for prerequisite skills only among the skills defined in the skill lists. For instance it would be possible to argue that in order to know that “Altitude is the elevation of a point or object from a known level, commonly the mean sea level” (Skill B3011) previous knowledge about the sea level is necessary. Since the corresponding skill is not in the skill lists, the prerequisite searching process stops at Skill B3011 (no requisites are defined for this skill). With other words we assume the knowledge necessary for B3011 as being available in children of the target age. In other cases we defined the “missing” knowledge as being necessary in order to play to the DEG. When an important skill identified through the recursive prerequisite identification process was missing in the skill set but could not really be assumed as a general available knowledge for the target age we corrected the skill lists by adding the new skill.
On the basis of the identified sub topics and skill lists the design team decided which skills were most suitable for the different parts of the game considering gameplay, didactic, and content related aspects. Based on this decision a subset of the skill lists have been identified and provided to the story designer with the purpose of including them in the game design document. The proposed learning situations have then been evaluated from a content and psycho-pedagogical point of view in order to ensure the correctness and adequateness of the resulting DEG.

**Didactic Design**

The didactic design of the 80Days game follows an approach in the tradition of constructivism (Fosnot, 1996). This paradigm of teaching and learning sees learning as an active process and is reflected by teaching strategies that promote active learning, or learning by doing. Games by nature possess almost all the key features of constructivist learning environments (Tsai, Yu, & Hsiao, 2007). Thus, the adoption of a constructivist didactic design perspective appears suitable and valuable for the design of an educational game.

Constructivism is not a specific or single pedagogical approach, but rather it describes how learning happens, suggesting that learners construct their own knowledge out of their experiences and that learning builds upon prior knowledge of an individual. Constructivist learning theory is based on Piaget (1950) and builds the basis for a range of pedagogical approaches. These approaches feature overlaps and similarities and concordantly argue for active generation of knowledge and
meaning from experience. The 80Days didactic design was inspired by the constructivist conception of learning and took up ideas from several constructivist pedagogical approaches in order to form a comprehensive and effective basis for the design of an educational game.

The Nature of Constructivist Learning Environments

Constructivism sees learners in the centre of the learning process, actively constructing their knowledge instead of being passively exposed to teaching (e.g., Ally, 2004). Learning is characterized by construction and discovery of knowledge. The design of constructivist learning environments should follow the following principles (Greeno, Collins, & Resnick, 1996):

- Realizing environments that allow participation in social practices of inquiry, sense-making and learning
- Supporting learners’ personal identities

Particular implications for the creation of constructivist e-learning environments that have been identified are (Ally, 2004; Hadjerrouit, 2006):

- Learning needs to be realized as an active and constructive process with interactive and illustrative instruction using examples and use cases
- Authentic learning activities relating to real-world situations should be realized
- The learning process should be encouraged by collaboration and cooperation
- Learners should have the opportunity to control and direct their learning process
- Learners’ meta-cognitive skills should be enhanced. Self-reflection should be realized

Self-Regulated Learning

From the perspective of constructivism, learning does not purely consist in a phenomenon characterized by stimulus and response (von Glaserfeld, 1995). Rather, learning requires self-regulation and the creation of internal conceptual structures through reflection and abstraction.
The concept of self-regulated learning (e.g., Zimmerman, 1990) has been taken up as an appropriate means to represent the learning process in the 80Days game. Self-regulation in learning refers to taking over responsibility and control over one's own learning processes.

Self-regulated learning describes the ways in which individuals regulate their own cognitive processes in educational settings and, therefore, refers to learning experiences that are directed by the learner (Puustinen & Pulkkinen, 2001). Self-regulated learning is seen as an activity that learners carry out for themselves in a proactive manner (Zimmerman, 2002), thus realizing the basic idea of constructivism. Self-regulated learning can be seen as a cyclical process consisting of three phases: forethought, performance, and self-reflection. The forethought phase involves activities of goal setting and strategic planning carried out before learning. Furthermore, it involves processes of self-motivation based on self-efficacy beliefs, outcome expectations, intrinsic interest and values. The performance phase refers to the actual process of learning and involves strategies aimed at fostering the quality and quantity of learning performance through self-instruction, self-control, and self-observation. The self-reflection phase involves processes of self-evaluation, causal attribution (i.e., referring to beliefs about cause of error and success), and self-reaction. These processes influence the forethought with respect to subsequent learning efforts. In sum, self-regulated learning can be seen as learning that is guided by metacognition, strategic action, and motivation to learn.

Self-regulated learning fosters self-satisfaction and motivation of learners to continue improving their learning methods. Due to this superior motivation to learn self-regulated learners are assumed to be more likely to succeed academically and view their futures optimistically (Zimmerman, 2002).

Although self-regulated learning argues for a high level of control for the learner, it does not mean to leave learners to their own resources. If necessary, the learner nevertheless needs to be supported (e.g., Aviram, Ronen, Somekh, Winer, & Sarid, 2008). Self-directing one's own learning requires self-regulatory skills which a learner does not necessarily possess to a sufficient degree. As a result, dedicated and tailored support is needed in order to guide the learner towards increased self-regulatory competence.
**Cognitive Apprenticeship**

The term ‘cognitive apprenticeship’ was coined by Collins, Brown, and Newman (1989). Cognitive apprenticeship is an instructional approach in the tradition of constructivism that consequently sees learning as a process that is active, self-directed, situated, and interactive. In this active learning process different stages can be distinguished (Schroeder & Spannagel, 2006):

- Observing experts in problem solving (complex problem modeling, externalization)
- Accomplishing simple tasks on one’s own – imitating the observed behavior, and getting feedback during the task (coaching, feedback)
- Gradually fading out support by expert while increasing problem complexity (scaffolding, fading)
- Self-reflecting on problem solving strategies and discussion with peers (reflection)

This conception of supporting the learner and gradually decreasing this support while the learner matures in his knowledge and expertise actually also corresponds to the principle of scaffold instruction, which argues for an approach of giving learners assistance in decreasing degree according to their learning progress, until they are able to apply new knowledge, skills, or strategies independently (Larkin, 2002; Lipscomb, Swanson, & West, 2004).

**Situated Cognition**

Constructivism in particular also emphasizes the idea of situated learning. This refers to learning activities that enable the learner to contextualize the information (Brown, Collins, & Duguid, 1989; Lave & Wenger, 1991). This can be achieved by relating learning contents to an authentic context and to the needs and concerns of the learner. Situated learning can be characterized by the following features (Anderson, Reder & Simon, 1996; Wilson, 1993):

- Learning is related to actions of real life and everyday situations
- Knowledge is acquired in relevant situations and is transferable to similar situations
• Learning is a result of a result of social processes of participating, thinking, perceiving, problem solving, and interacting
• Learning is not separated from the world of action

Participative and cooperative teaching methods are the main means of acquiring knowledge in the sense of situated cognition. By interacting with others and the environment, the learner creates and negotiates knowledge (Stein, 1998). Situated learning takes place in meaningful contexts and uses cooperative and participative teaching methods as the means of acquiring knowledge. Knowledge is created or negotiated through the interactions of the learner with others and the environment.

**The Aspect of Motivation**

Constructivism considers motivation for learning as an important aspect of learning. Learners should be constantly challenged and engaged in an appropriate way in order to sustain their motivation. In constructivist learning environments, where learners are assumed to discover and create knowledge, intrinsic motivation is relevant and crucial. Intrinsic motivation refers to experiencing learning activities as inherently interesting and enjoyable (Ryan & Deci, 2000). The feeling of autonomy, which is realized by giving the learner control in the sense of self-regulated learning, and competence, in terms of self-efficacy or confidence are important factors for enhancing this type of motivation. Confidence or self-efficacy (Bandura, 1977) is influenced by the experience of previous successes and failures and by causal attributions explaining them.

Attribution theory is concerned with how individuals interpret events of success and failure and how this relates to their thinking and behavior. In his theoretical approach Weiner (1974, 1985) focused on achievement and identified important factors affecting the attribution of success and failure - ability, effort, task difficulty, and luck. These can be classified according to the dimensions of stability (stable and variable) and locus of control (internal and external). This theoretical approach has been taken up in education for motivational training (e.g., Dresel & Ziegler, 2006; Rheinberg & Krug, 1999), by providing feedback in terms of attributions that are fostering self-worth and motivation – explaining success through internal factors and failure through variable factors.
In addition, approaches to motivational instructional design have been taken into account and contributed to the formation of the didactic background for the design of the 80Days game. Keller’s ARCS (1987) model elaborates on strategies on how to design motivationally sound instruction. The model is based on the assumption that a learner is engaged in an activity if it leads to the satisfaction of a personal need and if the person expects to be successful (in the sense of expectancy-value theory, e.g., Heckhausen & Rheinberg, 1980).

The model provides

- four conceptual categories – attention, confidence, relevance, and satisfaction – which characterize human motivation
- and strategies to enhance motivation in each category
- and implications for motivational design.

The four categories can be seen as four conditions to motivate a learner in a learning environment. The original model was recently expanded by volition and self-regulation as a fifth principle and aspect of motivation (Keller, 2008). The input-process-outcome model of instructional games by Garris, Ahlers, and Driskell (2002) builds upon established motivational approaches (e.g., Bandura, 1977; Csikszentmihalyi, 1990; Keller, 1987; Lepper & Malone, 1987; Malone, 1981). It describes the key characteristics of educational games (i.e., fantasy, rules/goals, sensory stimuli, challenge, mystery, and control) and their implications for game design. In the context of the 80Days project an advanced model of motivation in game-based learning has been devised that builds upon and integrates these two models of motivational instructional design (Garris et al., 2002; Keller, 1987, 2008) with a theoretical model of motivation to learn (Heckhausen & Heckhausen, 2006) into a common framework comprehensively covering relevant motivational aspects as a basis for adaptive educational games (Mattheiss, Kickmeier-Rust, Steiner, & Albert, 2009). This model has in particular nurtured the elaboration of principles and technologies for motivational adaptation (for further details please refer to the chapter “Balancing on a high wire: Adaptivity, a key factor of future learning games, p. 43).
Translation of the Didactic Background to the 80Days Game Design

The 80Days game concept and story incorporates the requirements and characteristics of a constructivist learning environment and integrates the ideas of the pedagogical approaches reviewed above. The learner is considered as in the center of the learning process. An overview is given in Figure 49.

The gaming character of the learning environment given by an educational game perfectly and necessarily fits the constructivist conception of an active and participating learner and the idea of interactive learning. Learning is embedded into the story and gameplay. The adventure and simulation character of the 80Days game strongly contributes to the paradigm of knowledge construction. The learner has the opportunity to discover knowledge through exploration and experimentation, like flying around with the UFO and visiting different European cities or investigating the effects of human interventions to the landscape on flood risk level in the terraforming simulation. The active learning concept incorporated in the 80Days game involves multiple encoding of information, deep elaboration, different levels of processing, and problem solving.

Figure 49. Overview of the didactic background of the 80Days game.

Computer games typically allow a high degree of learner control. The gaming experience in 80Days features a high level of learner control and self-direction. Following the paradigm of self-regulation, the three phases of the self-regulated learning process are realized in the game.
The forethought phase is characterized by introducing the goals of a mission or task and by steps of strategic planning, like the definition of flight routes. The performance phase consists in the actual gaming in terms of flying around with a UFO or experimenting with the terraforming simulation. During these activities, the learner is provided with tools for monitoring his/her performance, like the compass and map view or risk level and status bars. Reflection on learning and oneself is realized by recapitulating on the achievements in a mission or task and reviewing the knowledge acquired, for example through the preparation of a micro-mission report.

The hero of the 80Days game, in which learners slip into, has been deliberately chosen as a game character that supports learners’ personal identities and allows learners’ identification with the player character. This is further enhanced by the game story and macro-adaptive variations in narrative and game pace according to the learners’ preferences. A friendly alien as travel companion, with whom a new friendship establishes during the game ensures a sense of relatedness and allows cooperation and interaction.

Learning takes place in an authentic context. Authenticity is realized through learning situations that are authentic within the game’s reality, on the one hand, and through a real-world reference, on the other hand. The game, although packed into a science-fictional story, through the realistic context of geography and the utilization of real geographical data and information conveys learning situations that relate to the real world (Oleggini, Nova, & Hurni, 2009). The tasks to accomplish and knowledge acquired in the game are relevant to the game environment as well as transferable to the real world. The examination of geographic topics, like natural hazards, on the basis of real world examples (e.g., a flood in Budapest) allows a meaningful contextualization of the learning content and can be seen as a realization of a situated cognition approach (Cognition and Technology Group Vanderbilt, 1993; Lave & Wenger, 1991).

Meta-cognition and reflection, which are essential components of meaningful and self-regulated learning, are enhanced by the provision of tools that support the learner in the reflection on him-/herself and the task – for instance through possible self-evaluations on the location of geographic features or through a scoring system linked to and representing learning progress. In addition, meta-cognitive interventions in terms of hints provided by game characters prompt learner’s meta-
cognitive introspection and reflection (Steiner, Kickmeier-Rust, Matthe-  
iss, & Albert, 2009). The learner is encouraged to think about his/her  
own thinking processes and competence development during the learning and problem solving process (Flavell, 1976).

While providing the learner choice and control, at the same time a didactic philosophy of providing guidance and support is realized. The idea of supporting the learner in problem solving and providing him or her with assistance when needed, is realized through adaptation of the learning game experience to the learners’ needs (see chapter ‘Balancing on a high wire: Adaptivity, a key factor of future learning games’ in this book and Kickmeier-Rust & Albert, 2010; Kickmeier-Rust, Göbel, & Albert, 2008; Kickmeier-Rust, Steiner, & Albert, 2009). This consists in the realization of meaningful learning paths with growing difficulty of the learning tasks (i.e., macro adaptivity) as well as by the provision of adaptive interventions in form of tailored hints during gaming (i.e., micro adaptivity). With growing competence a learner will gradually need and get less support. This corresponds to the principles of cognitive apprenticeship and scaffold instruction and realized learning in terms of guided discovery.

Motivation is enhanced and maintained in the game by a generally appealing game design and story taking into account basic principles of motivational instructional design and key characteristics of games. The opportunities of self-regulation conceded to the learners in the game are assumed to have an intrinsically motivating effect. Moreover, motivation is supported through the realization of competence-based as well as motivational adaptation (Kickmeier-Rust & Albert, 2010; Steiner et al., 2009). These adaptation mechanisms allow personalizing the level of challenge to the learners competence, which critical for supporting learner’s self-efficacy beliefs (Bandura, 1977) and a prerequisite for the arousal of flow experience (Csikszentmihalyi, 1990). Attention and confidence of a learner are monitored as motivational aspects throughout the game and tailored motivational interventions are provided (for further details please refer to the chapter ‘Balancing on a high wire: Adaptivity, a key factor of future learning games’ in this book). In this context, motivational training (Dresel & Ziegler, 2006) has also been implemented in the game through feedback mechanisms that are related to attribution theory.

In addition to the general didactic background of constructivism, also results from the analysis of European school curricula for geography

and characteristics of the teaching approach in this specific subject informed the learning game design. In this regard, the multi-disciplinary and heterogeneous character of geography needed to be incorporated in the 80Days game. The earth-human relationships are in the focus of teaching geography. Geographic factors are not considered in an isolated and independent way, but rather in an interdisciplinary way touching also other school subjects such as physics or natural science. Geography has a bridge function (Haubrich et al., 2006), and this is also reflected in the description of the different national curricula. The reflection of the interdisciplinary and heterogeneous character of geography had to be considered in the selection of learning objectives for the 80Days game. Another important aspect of geography teaching is the relationship to every day life. This relates to the creation of relevance and meaningful contexts in the sense of situated learning. In addition, the observation of geographic phenomena and features from different perspectives and at different scales as realized in classroom teaching was to be carried on in the game design. The inclusion of learning objectives from different geographical topics in the 80Days game corresponds and underlines the heterogeneous and multi-disciplinary character of geography. It allows a diversification of the learning content covered by the educational game, thus reaching 'polyvalent' or horizontal learning experiences. This aspect of diversification is also assumed to have an enhancing effect for engagement in the game. Different perspectives and scale levels are realized in the game through the metaphor of a long zoom, considering geographical topics and sites from different levels of granularity, for instance locating a city on a map of Europe and flying there to take a photo.

Summing up, the 80Days game follows a sound and theoretically founded learner-centered didactic philosophy that involves active learning through self-regulation, exploration, and experimentation in an engaging context. Support and guidance mechanisms during learning tailored to the needs of the individual learner are incorporated and ensure the realization of meaningful learning paths. This is further enriched by motivational design and enhancement during gaming. The ultimate ambition of the 80Days game is to convey learning objectives from geography; storytelling and game play serve this main goal. Learning is embedded in a compelling game play and narrative, which allow active discovery of knowledge on geographical facts. Learners are enabled to experience and find out geographical information from differ-
The 80 Days Game

ent perspectives and granularity levels and are thus, supported in establishing an integrated understanding of geography.

**Categorization of Learning Actions**

The selection of learning objectives is one of the main and first steps in the design of an educational game. In 80 Days, this selection process follows a sound procedure on the basis of the knowledge modeling framework of Competence-based Knowledge Space Theory (Albert & Lukas, 1999; Heller, Steiner, Hockemeyer, & Albert, 2006). This means, on the basis of a thorough analysis of European curricula, geographical (sub)topics are selected and described in more detail through the definition of a set of relevant skills underlying and related to this topic. The skills, and the prerequisite structure established on them, constitute the main cognitive basis for the realization of adaptive technology in terms of adaptive sequencing of learning situations, non-invasive assessment of competence, and adaptive interventions tailored to a learner’s current competence state.

Complementing this procedure of transforming learning content into a knowledge space, in 80 Days an additional step of categorizing learning actions has been taken in the context of the demonstrator 2 learning game design. For the realization of a sound learning game design the skills selected can be operationalized in terms of learning actions. This means that for each skill learning actions are defined and suggested that are didactically appropriate for acquiring the respective skill. This process actually constitutes the translation of the latent, cognitive construct of skills to concrete learning activities (e.g., Marte, Steiner, Heller, & Albert, 2008). The definition of learning actions is thereby guided by the active nature of games in general and the constructive and experiential character of game-based learning in particular. Starting from the learning actions defined for the skills selected for Lizard 2.0, the provided learning actions were thoroughly analyzed in order to identify regularities in the actions. This analysis process led to the derivation of categories of actions subsuming and classifying the learning actions in more general action categories. The following categories of learning actions related to skills could be identified:

1. being told by a non-player character or reading news/messages
2. seeing and observing
3. measuring and taking samples
4. analyzing and concluding
5. experimenting and constructing

The deduction of these action categories constitutes a suitable way of systematization for the design process, enabling translation and transformation of those action classes into gameplay features and thus supporting the translation of the didactic content into a tangible design of an adaptive educational game. Actually, the derived categories are not only applicable to the specific skill set characterizing the learning objectives of the demonstrator game. Rather, the established categories constitute a more general categorization system applicable also for other learning objectives and subject matters.

When reviewing the derived categories, as listed above, it can easily be recognized that these categories, in principle, can be characterized by different activity levels referring to the degree of active involvement and learner action. Category (1) ‘being told by a non-player character or reading news/messages’ denotes rather passive and receiving actions. Category (2) ‘seeing and observing’ appears a bit more active and tangible than just being told or reading, though still not requiring high activity – and thus takes some kind of intermediate position. Categories (3) to (5) constitute actually active and experiential learning actions with a high level of learner involvement. The introduction of three distinct categories referring to really active learning has been made deliberately in order to better differentiate between the learning actions for their translation into gameplay.

The categories of learning actions as derived from the above described analysis can be related to the established and widely known taxonomy of learning objectives devised by Bloom (1956) and revised by Anderson et al. (2001). This taxonomy classifies intended learning behavior in and for educational practice. The purpose of this taxonomy is mainly to enhance the definition, exchange, and communication of ideas and material on educational objectives by using a common language. Mainly, such kind of educational framework serves for classifying learning objectives and increasing their manageability for instructional planning, learning, and assessment. The taxonomy comprises six categories that characterize different levels of cognitive processing and are assumed to form a cumulative hierarchy. The individual categories of learning actions as derived from our analysis can be associated with the categories of Bloom’s revised taxonomy (see Table 6). For almost each learning action category, a single corresponding level of the taxonomy
can be identified – except for category (3), which can be associated or split up into two distinct categories of the taxonomy.

Table 6. Association of the categories of learning actions with Bloom’s revised taxonomy.

<table>
<thead>
<tr>
<th>Category of learning actions</th>
<th>Assoc. category of Bloom’s revised taxonomy</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) being told by NPC or reading news/messages</td>
<td>remember</td>
</tr>
<tr>
<td>(2) seeing and observing</td>
<td>understand</td>
</tr>
<tr>
<td>(3) measuring and taking samples</td>
<td>apply</td>
</tr>
<tr>
<td>(4) analyzing and concluding</td>
<td>analyze, evaluate</td>
</tr>
<tr>
<td>(5) experimenting and constructing</td>
<td>create</td>
</tr>
</tbody>
</table>

A classification scheme like Bloom’s revised taxonomy (Anderson et al., 2001) is a helpful instrument for planning learning objectives and teaching, as lower-level activities and higher-level activities indicating different levels of understanding and cognitive processing can be distinguished. Effective learning objectives and well-designed courses should at optimum include also learning activities from a higher level. Foundational knowledge and comprehension requirements referring to lower levels of the taxonomy, however, will have to be achieved and planned before higher-order activities can be addressed or done. Thus, a classification instrument as it is provided by such a taxonomy may serve as a foundation to structure a course. The harmonization of skills in terms of Competence-based Knowledge Space Theory with the taxonomy can support even further this planning and structuring process through exploiting the benefits of this structural theoretical framework. Conversely, advantage can be taken from the educational taxonomies for structuring skills (e.g., Marte et al., 2008). In this way, such a taxonomy can serve the generation of the cognitive basis for intelligent adaptation in educational technologies.

When defining learning actions for the learning objectives selected for an educational game, actually each skill can in the simplest case be taught by a learning action of category (1), and thus to the lowest category of the taxonomy. The aim of a successful educational game should and will, however, be to use more active learning actions for acquiring skills when- and wherever possible. This corresponds to the maintenance of the engaging character of computer games and, thus, the utili-
zation of the appeal of this technology for learning, on the one hand, as well as to the effective incorporation of a didactic strategy following constructivist learning principles, on the other hand. In addition, this also corresponds to the consideration that a well-designed course should include learning activities from a higher level of Bloom’s taxonomy. In some cases, it will nevertheless be necessary or suitable to involve also learning actions of a more passive character.

In sum, the categorization of learning actions established for the skills representing the learning objectives of an adaptive educational game supports the effective translation into game design in terms of gameplay and didactics.

### Essence of Story Design

#### The Role of Story in a Learning Game

The story is one of the two main motivational tools to foster a satisfying and successful learning game experience. The other motivational tool is gameplay. These two motivational tools have to serve the learning objectives by motivating the learner with their respective means.

The role of 80Days’ story is to create a “virtual extrinsic motivation” in order to bring the learner/player to play through the learning game from the beginning to its end. Thereby the story works on a level of premises, main goals, quintessence, and conclusion. The player/learner wants to know, “How will this story end? What is the true meaning of this all? What is my goal? Will I succeed to reach the final goal?” This mixture of curiosity and suspense transforms into an extrinsically evoked motivation to go on with the gaming and learning experience. Moreover, the embedding of learning and gaming in the context of a frame story provides challenges and goals that turn the player’s actions into meaningful play and endow her learning and gaming experience with meaning.

In order to achieve this, the background story creates

- a believable world and setting for the characters, the plot and the player actions
- a meaningful & challenging context for the gaming and learning experience in the learning game
The 80Days Game

- the two main goals of the game (“Create Earth Report” and “Save the Earth!”)
- the fulfillment of the story’s premise (→“moral lesson”)
- the embedding of the learning premise (→overall learning objective of the curricula (“Geography saves the Earth!”) in a meaningful and therefore challenging and enjoyable context
- charging the gameplay and with values and reasons for the gameplay action (“Collect Geographical knowledge to create Earth report”, “Defeat the Alien Threat!”)

• the fate of characters

Development, Deployment, and Adaptation of Storytelling in 80Days

The development of the story design in 80Days can be summarized in three evolutionary steps, which developed the story throughout the project from its

1. DEVELOPMENT: initial idea and outline as a background story to its
2. DEPLOYMENT: concrete realization of the story in plots for game chapters (the Micro Missions) of the demonstrator versions as story events and situations, which is then followed by the
3. ADAPTATION of the plot during the running game (story pacing).

The development of the background story

Summary: The background story is the overall story that describes the universe of the 80Days game, its setting, its rules, its characters and their goals, relations, and conflicts. From this background story any plot for a game chapter, game situation, or even a complete game sequel can be derived. The background story is built around the premise for learning and the premise for the story, whereas the premise for learning with its educational objectives prescribes the premise and resulting content of the story.
An Alien’s Guide to Multi-Adaptive Educational Computer Games

Briefly, the story can be summarized as follows: Our hero, a 14-year-old teenager, gets abducted by an alien called Feon, who first tries to convince him to investigate the earth to fulfill his job as a star searcher. The alien has to investigate new planets as a military scout (star searcher), so that the planet can be captured afterwards and be used as a new home planet by his own race (the Ladrions) for a while, until the natural resources of the planet are exhausted. Then the Ladrions move on to the next suitable planet. This way of surviving is their policy, which is in harsh military hands. Feon is supervised quite rigorously, for he is not only an outstanding star searcher, but also known to be somewhat critical of this policy. Not knowing about Feon’s true task, the boy helps him exploring the earth. While traveling and managing obstacles, they both not only find out that the earth is in an endangered state, but the boy also finds out about Feon’s job, his race, their plans, and that he and his whole home planet Earth are in great danger. Being able to form a friendship with the alien, he may convince him to fight together against the Ladrions’ leadership and save the planet by portraying both societies’ alternative sustainable ways of surviving.

The goal of the story is to save planet Earth on a global level and to save a new friendship (between the alien and the boy) on a personal level. Both topics deal with honesty, respect, responsibility, and boldness. It is also a lesson on the necessity to learn and to see the world with different eyes in order to better understand the world around you and to finally recognize the manifold beauty, astonishing complexity, and the unique value of the world you are living in. In the end it is, of course, also a personal lesson about yourself which not only deals with the overcoming of personal conflicts in the course of the story (caused by the mutual betrayal of the alien and the boy), but also addresses personal responsibility, e.g., when the alien has to carry out the inner conflict between saving his family’s life or to turn a whole planet over to annihilation.

The story combines the two story goals and makes the motivation for the player both global and personal. This represents the best approach for creating a convincing need to win the game for the player.
Heroes learn on journey: The functions of story for gameplay and learning. In our story we meet two potential heroes: the Boy and Feon. Both figures comprise parallel characteristics, but reverse attributes: While the boy already lives on earth, longs for something more interesting, and is being pulled out of his normal earth-life, Feon, in a way, intrudes on it by abducting him in order to invade the Earth to find a new home for the alien race. Their destinies stand in each other’s way: Only one can survive this adventure, so it seems at first (cf. Figure 50).

Both are really curious and enthusiastic when it comes to investigating stars, creatures, nature, and travel, but do it for different reasons. The boy begins to develop his interest not only for a test, but because of having fun with flying a UFO and just “playing around”. Feon’s natural interest in being a real scientist of life and nature is being suppressed by his government by not letting him explore for the truth. At the same time, using his skills makes him a quite cold character at first. He does not want to get involved into personal feelings to living creatures and nature anymore, because he is going to lose it anyway. The boy is quite carefree about this aspect, being just a normal youngster, still being full of emotions and able to form relationships. He just lacks educational/scientific knowledge, thinking methods (to analyze, reason, validate, etc.), which Feon as a wide-ranging scientist already has. They both, in a way, want the same, but cannot reach it because of a respec-
tive flaw in their character. Both need each other to really become a hero and overcome themselves (cf. Figure 51).

![Figure 51. The “mirrored Heroes” Feon and the Boy in 80Days.](image_url)

Furthermore, their “races” (humankind and Ladrions) share a lot of parallels: they are consumers and therefore enforce the same resource problems to ensure survival of their kind. The Ladrions, who evolved from a primitive amphibian creature combined with a very intelligent sort of fish, go on consuming and exhausting planets. Their “solution” of just being very technically skilled and traveling from planet to planet, exhausting and leaving it, lacks wisdom. It mirrors a part of human nature as well, and where it might lead when it is put to an extreme. Additionally, in the anthropological discussion, the question about “how much animalistic parts and drive there is still in us” is not solved and comes up from time to time again.

**The heroes determine their fate.** As already mentioned, it is not determined whether the boy or Feon will achieve becoming a hero in the end. The player in the role of the boy may influence his own and Feon’s development throughout the game until the end of stage two of our story. The main aspect is to save the world, and therefore becoming a hero means to save the world. In our story this means, first, prevent the Ladrions, especially the General, from destroying all life on earth and, second, convince and enlighten both societies (Alien & Human) about sustainability. The second aim fits our pedagogical target and gives an adequate premise to develop a storyline that contains geo-
graphical understanding and taking responsibility in the end. To solve these two big tasks, two great efforts are to be made. First of all the player/boy has to come up to a good level of geographical knowledge and understanding, to be convincing enough in front of his human fellowships in the end. Additionally he will need Feon’s help and therefore his friendship to absolutely save both worlds and find the best solution for it.

Depending on the learning progress and on the emerging relationship between kid and alien, the game may have different endings. The didactical rationale here is to understand that the best solutions cannot be found thinking egoistically and that sustainability is a not only a question of skill but also a question of community-thinking. Developing these skills takes place in the second stage of the game. By the way we generate a motivating side effect in the story, to find out about Feon, influencing the events and characters and luring with interesting information about the fantasy background story. This will make the player want to find out more, immerse more into the game, and therefore

Figure 52. Game objectives for becoming a Hero in 80Days related to story (yellow), learning (red) and gameplay (blue).
An Alien’s Guide to Multi-Adaptive Educational Computer Games

learn more. As in any good hero’s journey in a film or video game, the Heroes act self-determined and take fate in their own hands – be it as a Hero of the movie or the player being the Hero in the video game.

The player will solve the quests (Micro-Missions) by collecting items, information, finding out about special connections between acquired knowledge. The player will gain her geographical knowledge and raise her skill level from collected information as well as the act of traveling itself (Goethe: “Travelling educates”; see also Figure 52).

The Deployment of Lizard’s Story

Storytelling in “Lizard 1.0”

As mentioned, the prototype game of the project was released in three consecutive versions. The final interactive setup (as realized in releases 2 and 3) is illustrated in Figure 53.

At the beginning, the game has a “cinematic intro” – a linear video that is presented to the player without any possibilities to interact. Those cut scenes are video sequences that serve as bridges between interactive game scenes and, most often, these passive scenes are used to convey the story. The video shows the abduction of the boy and introduces the setting, the player character (protagonist), and the start of the story. Following the intro, the game starts the 3D-game environment. Storytelling from now on takes place in so-called “interactive cut scenes” which are story situations that happen in the game and propel the background story and development of characters in the game. In the first interactive cut scene, the player takes over and now controls the character of the boy. The player finds himself in the observatory of the UFO and encounters Feon for the very first time. After this introductory story situation the player knows about the setting of the game, what it is all about, Feon’s intentions and their next missions. After this the learning game lets the player go through a tutorial for teaching the player how to navigate and fly the UFO. Then it decides which Micro-Mission should be taken as the first mission. This decision is made on basis of the questionnaire that was filled out by the player before the game started. “Lizard 1.0” has got 4 Micro-Missions (MM) whose order of appearance can be adapted to let the player experience different learning paths according to her profile and performance during the
Figure 53. Overall interaction diagram of Lizard 2.0 and Lizard 3.0.
Figure 53 (continued). Overall interaction diagram of Lizard 2.0 and Lizard 3.0.
game. The three possible didactic sequences of Micro-Missions in Lizard 1.0 are:

1. MM2 MM3 MM4 MM1
2. MM2 MM4 MM3 MM1
3. MM4 MM2 MM3 MM1

**Storytelling in “Lizard 2.0”**

Lizard 2.0 continues after the player has accomplished the four Micro-Missions of “Lizard 1.0”. The learning content covers the geographic topic “natural hazards”. While the UFO is hovering over Europe, Feon tells the boy that he is interested in the natural hazards that can occur on planet Earth. An incoming radio message informs both that there is a flood in Budapest, Hungary. They decide to fly there. After they have observed the consequences of the flood in the city Budapest, Feon invites the player to simulate the flood with a virtual model of Budapest and its surroundings in order to learn more about the natural causes of a flood, human made risk factors, and possible prevention measures that humans can deploy in order to minimize the risks of a flood. Storywise this learning experience offers an adaptation in terms of story pacing: There is a relaxed, driven, and hectic story pacing that depends on the profile of the player and the acquisition of skills during the game. All of the three story pacings lead to different variations of the game’s ending. In the relaxed ending Feon thanks the boy for the great adventure; in the hectic one, they must flee from attacking scouts that were send out by the Alien General. All of the three variants share the same outcome: a cinematic closing scene in which the player sees how the boy is beamed back to his house after the adventure.

**Storytelling in “Lizard 3.0”**

Lizard 3.0 is mainly an enhancement of the existing game content of Lizard 2.0 and implements the design recommendations that were given by the validation team after the pupils had tested Lizard 2.0. Most of them affected the improvement of learning situations. The storytelling was enhanced in order to make the different story pacing and resulting endings more visible for the player. For that reason every story pacing has its own interactive cut-scene that shows the respective consequences of that story mode before the cinematic closing scene is shown.
**The Adaptation of the Storytelling in the Game**

While in “Lizard 1.0” the focus was on the Macro adaptation for the story (the story engine decides which one of the four Micro-Missions will be presented next), in the subsequent two versions of the game the adaptation of the story was focused on the story pacing in the game. As mentioned in the chapter above, the story pacing of “Lizard 2.0” and “Lizard 3.0” offers a relaxed, driven, and hectic story mode with respective dialogues, cut-scenes with appropriate events. For example, in the hectic mode, Feon and the boy are in the observatory and all of a sudden Alien scouts attack them with their space ships, while in the relaxed mode there are no incoming radio messages from an angry Alien General or any other hint at an Alien threat.
Concluding Remarks

Michael D. Kickmeier-Rust

Undoubtedly, ‘serious’ computer games will become a ‘serious’ part of the future educational landscape. In contrast to many educational technologies and practices, the immersive worlds of computer games are not compatible with a one-fits-all approach to learning. All the great strength and advantages computer games have, as discussed in this book, heavily depend on meeting individual interests, characteristics, and needs. This holds not only for a gaming related but also for the learning related hemisphere of this complex construct ‘immersive educational computer game’. Accordingly, on the basis of the recommendations and findings of a broad range of studies and surveys, a key aspect of a successful educational medium, in particular an education game, a smart and appropriate personalization of various factors is essential. The complex set of relationships, however, makes it difficult to appropriately and successful find a state of balance which assure immersive gaming and effective learning for a specific student/player.

In this book we have tried to give a brief overview of approaches to assess various factors and dimensions and to adapt accordingly in order to establish a promising balance of learning and playing, of challenge and ability, of serious and entertaining aspects, and of virtual and real worlds. In this book we have tried to introduce a technological and conceptual framework to realize such vision of intelligent multi-adaptive educational computer games. And although this and other technologies are not fully mature yet, they are recommended to be as strong part of designing the next generation of serious games – making serious games serious business, making computer games a ‘real’ effective and efficient medium for learning and teaching.
References


References

Design and Children (IDC'05) (pp. 9-16), June 8-10, 2005, Boulder, CO.


References


References


Keller, J. M. (2008). First principles of motivation to learn and e3-learning, Distance Education, 29, 175-185.


Kelleher, C., & Pausch, R. (2006). Lessons learned from designing a programming system to support middle school girls creating...


References


Adaptation in Digital Educational Games (pp. 55-65). University of Graz, Austria.


Journal of Continuing Engineering Education and Life-Long Learning, 17, 337-357.


206


List of Authors

Prof. Dietrich Albert. Since October 2010 Dietrich Albert is professor emeritus of psychology at University of Graz. He is an active senior scientist at Graz University of Technology, Knowledge Management Institute and key researcher at the Know-Center Graz. From 1993 to 2010 Dietrich was the head of the Cognitive Science Section at the University of Graz, the Department of Psychology’s largest working group. In the preceding years he was with the Universities of Göttingen, Marburg, Heidelberg, and Hiroshima. His research topics cover several areas, including learning and memory, psychometrics, anxiety and performance, psychological decision theory, computer based tutorial systems, values and behavior. Dietrich’s actual focus is on knowledge and competence structures, their applications, and empirical research. By working with psychologists, computer scientists, and mathematicians several academic disciplines are represented within his research team. Beside national activities, his expertise in European research and development projects is documented by several successful European projects.

Dr. Owen Conlan is a Lecturer with the Knowledge and Data Engineering Group (KDEG) in the School of Computer Science and Statistics of Trinity College, Dublin. As a senior member of the group he oversees many research activities related to eLearning, visualisation and personalisation. Owen has contributed to over sixty publications and his primary research interests lie in investigating the practical application of adaptation, personalisation and visualisation approaches to a variety of real world situations to facilitate increases in cognition and understanding. His research interests also include strategies for adapta-
tion, semantically meaningful visualisations of adaptive information, personalisation in highly immersive visual environments and the evaluation of cognitive and meta-cognitive gain in personalised eLearning systems.

**Dr. Stefan Göbel** graduated in Computer Science at TU Darmstadt in 2002. The topic of his dissertation is based on applied research he has investigated in the GIS department at Fraunhofer IGD starting late 1997 and deals with metadata-based graphic-interactive access to geodata archives. From 2002-2008, Dr. Göbel has been heading the Digital Storytelling department at the Computer Graphics Center e.V. in Darmstadt and has been involved in numerous RTD activities and projects on European, national and regional level in the field of Storytelling based Edutainment applications. Examples for this include the EU funded projects art-E-fact, INSCAPE and U-CREATE or the DinoHunter applications developed for the Senckenberg museum in cooperation with T-Online. In 2003 Dr. Göbel originated the int’l TIDSE conference on Technologies in Interactive Digital Storytelling and Entertainment, which merged with ICVS into ICIDS as leading conference in the field of Interactive Storytelling. Since late 2008 Dr. Göbel is heading the Serious Gaming group at the Multimedia Communications Lab at TU Darmstadt and continues the interdisciplinary research in the field of Technology-enhanced Learning, DEG and Serious Games. Research challenges include – among others – authoring, personalization and adaptation of Story-based DEG’s, collaborative learning scenarios or different facets of serious game applications, for instance in the field of serious games for health.
Cormac Hampson is a Ph.D. candidate with the Knowledge and Data Engineering Group (KDEG) in the School of Computer Science and Statistics of Trinity College Dublin. Cormac has received a B.Sc. in Multimedia from Dublin City University and a M.Sc. in Ubiquitous Computing from Trinity College Dublin. His primary research interest is in supporting casual users to explore information by leveraging domain expertise. Cormac has also worked extensively on adaptation and personalisation techniques that can be employed within Digital Educational Games.

Dr. Michael D. Kickmeier-Rust is a psychologist and software programmer. Since 2001 he is with the Department of Psychology at the University of Graz, Austria. His research and development activities are concentrating primarily on intelligent, adaptive educational systems. In particular, Michael is working on the evolution of psycho-pedagogical frameworks and models of adaptivity on the macro and micro levels. This work includes individualized navigation within learning environments, individualized presentation of educational material, as well as non-invasive assessment and interventions. Since 2006 his very focus is on the psychological aspects of learning with immersive digital computer games and the integration of intelligent educational technology in game-based learning. Since 2008 Michael is coordinating the FP7 ICT project 80Days that is dealing with game-based learning. Related fields of interest are human-computer interaction and specific fields of aviation psychology, dealing with education and training.
Kevin Koidl is a Ph.D. candidate with the Knowledge and Data Engineering Group (KDEG) in the School of Computer Science and Statistics of Trinity College Dublin. Kevin has received a dual master degree in Business and Computer Science (Dipl. Wirtschafts. Inf.) at the Technical University of Darmstadt in Germany. His primary research interest is Web Engineering specifically personalisation and adaptation strategies. In relation to this interest Kevin has worked extensively on adaptation strategies that can be employed within Digital Educational Games and other related application.

Dr. Effie Lai-Chong Law is a Lecturer at the Department of Computer Science at the University of Leicester (UK) and a visiting Senior Research Scientist of ETH Zürich (Switzerland). She obtained her PhD in Psychology (summa cum laude) from the University of Munich (LMU), Germany. Her research domains are human-computer interaction (HCI) and technology-enhanced learning (TEL) with a specific focus on usability and user experience design and evaluation methods. She has published a number of papers in international conferences, books and journals in these domains. She has been chairing two COST Actions: MAUSE “Towards the Maturation of Usability Evaluation and TwinTide “Towards the Integration of Trans-sectorial IT Design and Evaluation” in which leading HCI experts from more than 20 European countries have been involved. Besides, she has been a work-package leader in several international and national research projects, including 80Days or ROLE.
**Elke Mattheiss** is a psychologist working at the Department of Psychology at the University of Graz since 2008. Her research work focused hitherto mainly on the integration of motivational aspects and concepts in educational games. Currently she is concerned with approaches of adaptivity - on a macro and micro level - in the field of game-based learning, aiming for the non-invasive and individualized presentation of motivational interventions. Her research experience also implies the scientific evaluation of usability issues and learning effectiveness within the FP7 ICT project 80Days. Related fields of interests are cognitive psychology of learning and memory, and human-computer interaction, especially interaction design.

**Florian Mehm** is currently a PhD student at the Multimedia Communications Lab at Technische Universität Darmstadt. His main topic of research are game authoring systems for non-specialists (e.g., those lacking programming skills) as well as assembled topics from the field of Digital Games and Serious Games.

**Samuel S. Nova** was born in Denmark and now resides in Switzerland. He has been involved with game development for more than 20 years and has games released on various platforms such as PlayStation, PlayStation Portable, Game Boy Color, Game Boy Advance, PC, iPhone, J2ME devices. He has mainly been involved in development but also producing, design, tests and team management. From 2008 to 2010 he worked at the Institute of Cartography, ETH Zurich as part of the EU funded research project around an inspiring virtual
An Alien’s Guide to Multi-Adaptive Educational Computer Games

learning world in eighty days (80Days). In the project he focused in particular on rendering of virtual terrains.

**Lorenzo Oleggini** was born and resides in Switzerland. He holds a MSc in Geography, with Major in Remote Sensing issued by the University of Zurich. He graduated with a thesis on “Rule-based crop area classification using multi-temporal, multi-source data”. His focus lies on GIS and remote sensing. From 2008 to 2010 he worked at the Institute of Cartography, ETH Zurich as part of the EU funded research project “Around an inspiring virtual learning world in eighty days” (80Days). In the project he focused in particular on the semi-automatic generation of navigable virtual terrains based on geographic datasets.

**Dr. Ivan Orvieto** was born in Rapallo (GE) in 1971, and he graduated the first time in 2002 in Information Science and the second time in Specialized Informatics in 2005. He works in the videogame field since 1998, first 2 years as 3D artist and then as Project Manager in many videogame projects in GMM Entertainment (GIUNTI publishing group). In 2002 became Executive Producer in Gruppo Mafrau (Italian SpA, devoted to e-learning innovative solutions) and since 2004 he’s currently Executive Producer and Game Designer in TESTALUNA.

**Daniel Schwarz** was born in 1976 and holds a master’s degree for audio-visual media from the Academy of Media Arts in Cologne (KHM). His master thesis dealt with visual metaphors and storytelling in arts, information design, and modern sciences by the use of advanced 3D-computer visualisation. Daniel Schwarz works more than 10 years in the intersection of science, arts, media and technology. In 2003, he co-
founded the company “TAKOMAT”, a multimedia production company based in Cologne, where he holds the position of a managing shareholder and creative director. In 2005, he initiated the European research project ELEKTRA which pioneered interdisciplinary research on learning with computer games. From 2006-2008 he coordinated the project, which was awarded as one of the “Star projects” of European research and showed how pupils can learn with games. From 2008-2010 his company was partner in the European Research project 80Days. TAKOMAT developed a character animation technology that will enable the next generation of learning games. The company recently developed the most complex serious game about energy for the German Government.

**Christina M. Steiner** completed her Diploma (MS) in Psychology at the University of Graz. She is currently a researcher at the Cognitive Science Section, Department of Psychology of the University of Graz. Her work within several European R&D projects on e-learning focuses on the representation, modelling, and assessment of knowledge and competence. She is doing research on psycho-pedagogically founded design and adaptation of digital educational games. Furthermore, her work includes the evaluation of the effectiveness of learning technologies based on sound psychological methods and techniques. Further interests lie in the use of concept maps as a means to build prerequisite structures among learning objects and competences, on the validation of concept maps, and their application as a learning and teaching strategy.

**Martin Stoecker** was born in 1980 and holds a master’s degree in computer science of the University of Koblenz. During his study he worked as student assistant in the field of facial animation for the Fraunhofer Institute for Media Communication in the EU project “Virtual Human” in 2003. In 2006 he worked for the Laboratory of Mixed Realities (LMR) in Cologne as a software developer and gained experiences in programming and research on e-Learning environments. Here, he participated in the FP6-ICT-TEL projects ELEKTRA and iClass. He devised game concepts for the Nintendo DS and the Nintendo Wii.
and was involved in game development and game design. In 2008 he joined the coordinator of the FP7-ICT-TEL project 80Days. Here, Martin Stoecker works on research and development of believable behaviour generation for game characters and the integration of research knowledge into a character engine.

**Viktor Wendel** is pursuing a PhD title at the Multimedia Communications Lab at Technische Universität Darmstadt. His main areas of research are multiplayer scenarios in Serious Games, including Game Mastering and the design of collaborative (learning) scenarios.
## Index

activity theory, 150
adaptation, 29, 44, 45, 50, 62, 72, 82, 96, 106, 177
adaptive element, 111
adaptive storytelling, 1, 48, 49, 50, 123
framework, 92
attribution theory, 167
authoring, 40, 41, 98
bat cave, 202
Bat Cave, 2, 96, 103, 104, 110
CbKST, 45, 47, 48, 51, 52, 57, 83, 90, 95, 161, See also competence-based knowledge space theory
child-computer interaction, 142
cognitive apprenticeship, 166
cognitive intervention, 56, 57, 58, 59, 62, 80
competence state, 47, 50, 51, 58, 61, 173
competence-based adaptation, 45
competence-based knowledge space theory, 1, 45, 161, 173, 175, See also CbKST
computer game, 8, 12, 53, 106, 187
computer game market, 121
computer game technology, 125
constructivism, 164
COTS, 25, 26
DEG. See digital educational games
DGBL, 26, 27, 105, 107, See also digital game-based learning
didactic design, 5, 13, 160, 163
digital educational games, 50, 89, 104, 137, 202, 211, 212
costs, 121
digital elevation model, 130
digital game-based learning, 26, 105, See also DGBL
digital gameplay experience model, 145
digital storytelling, 50, 89, 90, 210
interactive, 90
e-learning, 16, 46, 53, 68, 106, 139, 164, 214, 215
EVADEG, 138
evaluation, 2, 6, 55, 88, 103, 104, 137, 206, 210, 212, 213, 215
for adaptive digital educational games, 137
games usability, 140
of DEGs, 141
real-time interaction trajectory for adaptivity, 147
expert review, 142
eye-tracking, 148
feedback, 1, 22, 45, 51, 52, 53, 54, 58, 59, 63, 64, 67, 75, 76, 78, 86, 99, 141, 144, 145, 166, 167, 171
definition, 43
flow experience, 145
game design, 1, 2, 3, 5, 12, 13, 15, 17, 19, 22, 26, 36, 43, 60, 63, 152, 153, 169, 171, 172, 173, 176, 216
game design document, 163
game engine, 28, 29, 37, 92, 99, 105, 109, 110, 112, 116, 119
main components, 34
game engine architecture, 33, 34
game engine choice, 30
game engine core, 36
game engines, 124
game experience evaluation, 144
game metaphor, 11, 12, 13, 17
game-based learning, 89
gameplay design, 153
gameplay modes, 156
geographic information systems, 123
Hero’s Journey, 94, 100
ICML, 115
interventions, 56, 78
just-in-time personalization, 110
knowledge space, 1, 45, 46, 90, 112, 160, 161, 173, 175
knowledge space theory, 45, 46, 90, 112
knowledge states, 46
learner model, 48, 107
learner-centered design, 142
learning actions, 173
learning design, 43
learning effectiveness validation, 139
Index

learning engine, 28, 92, 99, 105, 109, 110, 114, 116, 118, 119, 147
learning environment, 25, 69, 86, 140, 152, 168, 169
learning experience, 12, 13, 14, 39, 45, 66, 77, 105, 106, 156, 176, 185
learning game, 5, 6, 8, 10, 11, 12, 13, 14, 15, 16, 23, 25, 39, 54, 59, 60, 62, 63, 66, 68, 73, 75, 76, 82, 92, 171, 172, 173, 176, 182
80Days summary, 153
definition, 24
learning game design, 5, 17
learning game situation, 16, 17, 19
learning motivation, 62, 197
learning object, 16, 47, 84, 107, 113
learning paths, 14, 17, 46, 47, 83, 95, 171, 172, 182
Lizard, 2, 15, 16, 18, 94, 110, 153, 156, 159, 173, 182, 185, 186
macro adaptivity, 47, 107
macro level, 1, 29, 47, 48, 50, 66, 82, 88, 95, 154
Mangalore, 36
micro adaptivity, 50, 107
micro level, 1, 29, 47, 52, 54, 82, 85, 87, 154, 213
microworlds, 26
motivation, 62
definition, 63
motivation assessment engine, 114
motivation for learning, 167
motivational adaptation, 64, 70, 72, 75, 81, 83, 87, 168, 171
motivational assessment, 68
motivational indicators, 70
motivation-enhancing interventions, 78
narrative game-based learning objects, 90, 97, 104
narrative paradox, 92
NEBULA 2, 31
NGLOB. See narrative game-based learning objects
non-invasive assessment, 1, 54, 58, 59, 73, 74, 77, 78, 84, 85, 87, 140, 173, 211
non-invasive competence monitoring, 51
participatory design, 141
personalization, 104, 105, 110
prerequisite relation, 46, 47, 95, 161
probability distribution, 52
<table>
<thead>
<tr>
<th>Term</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>problem space, 51, 52</td>
<td></td>
</tr>
<tr>
<td>problems solving, 45</td>
<td></td>
</tr>
<tr>
<td>real-time interaction trajectory</td>
<td></td>
</tr>
<tr>
<td>for adaptivity evaluation, 147</td>
<td></td>
</tr>
<tr>
<td>scripting engine, 36</td>
<td></td>
</tr>
<tr>
<td>self-evaluation, 165</td>
<td></td>
</tr>
<tr>
<td>self-regulated learning, 55, 56, 164, 165, 169, 170</td>
<td></td>
</tr>
<tr>
<td>situated learning, 166</td>
<td></td>
</tr>
<tr>
<td>skill assessment engine, 109, 111, 112, 118</td>
<td></td>
</tr>
<tr>
<td>skill structure, 95, 112, 161, 163</td>
<td></td>
</tr>
<tr>
<td>stage editor, 101</td>
<td></td>
</tr>
<tr>
<td>story description language, 115</td>
<td></td>
</tr>
<tr>
<td>story editor, 28, 34, 39, 41, 100, 101</td>
<td></td>
</tr>
<tr>
<td>story engine, 2, 28, 37, 92, 97, 99, 105, 109, 116, 118, 119, 186</td>
<td></td>
</tr>
<tr>
<td>story engine interface, 37</td>
<td></td>
</tr>
<tr>
<td>story pacing, 116</td>
<td></td>
</tr>
<tr>
<td>story structure, 100</td>
<td></td>
</tr>
<tr>
<td>StoryTec, 98, 100, 102, 104</td>
<td></td>
</tr>
<tr>
<td>terrain engine, 34, 38</td>
<td></td>
</tr>
<tr>
<td>usability, 2, 24, 39, 137, 141, 212, 213</td>
<td></td>
</tr>
<tr>
<td>usability evaluation methods, 138</td>
<td></td>
</tr>
<tr>
<td>usability problem, 142</td>
<td></td>
</tr>
<tr>
<td>user experience, 2, 137, 140, 141, 151, 212</td>
<td></td>
</tr>
<tr>
<td>attributes, 137</td>
<td></td>
</tr>
<tr>
<td>definition, 146</td>
<td></td>
</tr>
<tr>
<td>user experience evaluation methods, 146</td>
<td></td>
</tr>
<tr>
<td>user test, 143</td>
<td></td>
</tr>
<tr>
<td>UXEM. See user experience evaluation methods</td>
<td></td>
</tr>
<tr>
<td>videogame definition, 22</td>
<td></td>
</tr>
<tr>
<td>videogames, 21</td>
<td></td>
</tr>
<tr>
<td>virtual terrain generation, 130</td>
<td></td>
</tr>
</tbody>
</table>
Computer and video games are an important part of today's entertainment landscape. Can we use these games to motivate and educate? That was the question explored by the European project 80Days [www.eightydays.eu], a leading-edge research effort. The project ran from 2008 to 2010, focusing on developing the psycho-pedagogical and technological foundations for successful digital educational games - where success is measured by both educational and financial effectiveness.

This book is the result of that research.