This Chapter presents the general background and fundamental concepts related to the domain problem that is addressed in this thesis. At the first section (\autoref{sec:cscl-and-scripted-collaboration}), an overview of CSCL field and scripted collaboration is presented to provide a most comprehensive and clear understanding about the research context. This section also describes in detail the motivation problem caused by the scripted, and the current computer-based solutions to deal with this problem. The \autoref{sec:gamification} elaborates an overview of gamification, and the best practices and theories related to this technology. Furthermore, the related works that use gamification in the context of CSCL and other contexts to deal with the motivation problem are presented in this section. Finally, the \autoref{sec:ontologies-and-ontology-engineering} presents the fundamentals of ontologies and ontology engineering. This sections also discusses how ontologies is currently used to support the systematic formalization of theory-based knowledge, and how this formalization, in the field of artificial intelligence in education, is used to overcome some problems that are similar to those that must be solved to provide a computational support in the gamification of CL scenarios with theoretical justifications based on the best practices and theories related to gamification.

\section{CSCL and Scripted Collaboration}

\label{sec:cscl-and-scripted-collaboration}

Although CL has a long history in education, it is not until the early 1990s that the research field dedicated to study how to provide support for the CL through the use of Internet and computational technology had gained attention and strength \cite{StahlKoschmannSuthers2006}. Such research field known as Computer-Supported Collaborative Learning (CSCL) is a multidisciplinary field that combines studies from the cognitive psychology education and from the computer science to effectively enhance the CL process through the use of computational technology \cite{HoppeOgataSoller2007}.

The general aim of CSCL field is to develop technologies to support or create situations in which two or more students learn together through the interaction among them \cite{Dillenbourg1999}. In these situations, the learning outcomes is consequence of students' interactions and how these interactions affect the individual. In consequence, to enable a well-though-out design of CL, the CSCL scripts have been proposed by the CSCL community as the technology to facilitate the social and cognitive processes of learning by describing the way in which the learners will interact with each other in a CL scenario \cite{HarrerKobbeMalzahn2007}.

\subsection{CSCL Scripts}

\label{sec:cscl-scripts}

CSCL scripts are the technology that describes how to structure and orchestrate the CL process to attain a set of pedagogical objectives \cite{DillenbourgJermann2007}. Such description is a set of prescribed instructions that indicates how to facilitate the social and cognitive processes in group activities \cite{Dillenbourg2002}. These prescribed instructions are defined by instructors, like teachers or instructional designers, as an effective way to attain a set of learning goals. Thus, the CSCL scripts indicate the way in which students should collaborate, they constrain the interactions among the participants, they specify the roles for the participants, and they indicate the distribution of task, tools, and resources used in the CL process.

In order to provide a common and sharable description of CSCL scripts, \citeonline{KobbeWeinbergerDillenbourgHarrerHamalainenHakkinenFischer2007} propose a framework to describe the elements of CSCL scripts. This framework consist in the common specification of components and mechanisms illustrate in \autoref{fig:components-and-mechanisms-of-cscl-scripts}.

\begin{figure}[htb]

\caption{Components and mechanisms of CSCL scripts}

\label{fig:components-and-mechanisms-of-cscl-scripts}

\centering

\includegraphics[width=0.95\textwidth]{images/components-and-mechanisms-of-cscl-scripts}

\fadaptada{Fischer2007}

\end{figure}

The structural \emph{components} of CSCL scripts are the participants, groups, activities, roles and resources. The component of \emph{participants} is used to describe the participants, such as learners, monitors, and teachers. Although this description can be abstract or concrete and simple or complex, it is often presented in a simple manner with rules that indicate conditions to participate in the CL process. The component of \emph{activities} describes what will be performed by the participants in the CL process to attain the learning goals defined by the instructional designers. The component of \emph{roles} describes the privileges, obligations and expectations of participants in the CL process. The component of \emph{groups} of participants defines through hierarchical structures how the students are grouped according to the participants' characteristics. The component of \emph{resources} describes the learning objects (e.g. content, material, and tools) that can be used by the participants during the CL process.

The \emph{mechanisms} of CSCL scripts are the group formation, component distribution and sequencing. The mechanism of \emph{group formation} consists in the specifications of how the participants will be distributed over the groups. The mechanism of \emph{task distribution} provides the specification about how the components of scripts are distributed over groups using the mapping of groups, activities, roles, and resources. The mechanism of \emph{sequencing} consists in the definition of how the components and groups defined in scripts are distributed over time. In general, this sequencing describes the execution order of activities in the CL process.

\autoref{qua:social-script-framework-kobbe} shows the description of social script \cite{WeinbergerErtlFischerMandl2005} using the framework proposed by \citeonline{KobbeWeinbergerDillenbourgHarrerHamalainenHakkinenFischer2007}. In this example, the CL scenarios orchestrated by the social script foster the acquisition of knowledge through a set of case studies (\emph{resources}) that are analyzed and reviewed by the students groups. The number of students in each group is equal to the number of case studies, and the ideal number is three. In the first step of sequencing, each learner playing the \emph{analysis} role writes down an analysis of case study, and then, he critiques the analyses made by other learners playing the \emph{critics} role. In the second step of sequencing, each learner revises his/her own analysis, taking into consideration the critiques received by the other learners in the case group.

\begin{quadro}[htb]

\caption{Social script describes using the framework proposed by \citeonline{ KobbeWeinbergerDillenbourgHarrerHamalainenHakkinenFischer2007}}

\label{qua:social-script-framework-kobbe}

\centering

\footnotesize

\begin{tabular}{l p{12cm}}

\toprule

\multicolumn{2}{l}{\textbf{Structural components:}} \\ \midrule

\textbf{Participants:} &

A number of participants that must be divisible by the number of case studies. \\

\textbf{Groups:} &

Case groups \\

\textbf{Activities:} &

(a) Applying theoretical concepts to the case study and constructing arguments \\

&

(b) Critiquing initially scaffolder with prompts for eliciting clarification, identifying conflicting views and constructing counter-arguments \\

\textbf{Roles:} &

\emph{Analyst} and \emph{Critic} \\

\textbf{Resources:} &

Case studies (minimal number is three case studies) \\

\toprule

\multicolumn{2}{l}{\textbf{Mechanisms}:} \\ \midrule

\textbf{Group formation:} &

All participants are grouped by the number of case studies. Each participant becomes member of all case groups although with different roles in each. Each participant is the responsible analyst for one case study and critic for all other cases \\

\textbf{Task distribution:} &

Each case group receives one case study, and the roles are distributed in a way that each participant assumes the role of analyst in one case group and the role of critic in all other case groups \\

\textbf{Sequencing:}

& - the analyst writes an analysis of case study. (a) \\

& - wait for all case group analysts to be done, and writes a critique for the analysis of case study. (b) \\

& - wait for all case group critics to be done, and the analyst considers each critique and writes a reply to each. (a) \\

& - wait for all case group analysts to be done each critic in turn reads the reply and writes a second critique. (b) \\

& - wait for all case group critics to be done... the analyst considers all critiques and revises the analysis of case study (a) \\

\bottomrule

\end{tabular}

%\fadapted{KobbeWeinbergerDillenbourgHarrerHamalainenHakkinenFischer2007}

\end{quadro}

\subsection{Levels of Abstraction and Granularity of CSCL Scripts}

\label{sec:level-of-abstraction-and-granularity-of-cscl-scripts}

CSCL scripts have different levels of abstraction and granularity in the description of CL scenarios \cite{Dillenbourg2002, DillenbourgJermann2007, Villasclaras-FernandezIsotaniHayashiMizoguchi2009}. The levels of abstraction classify a script according to the completeness of elements described by them (from the most abstract to the most concrete). The levels of granularity classify the scripts according to the aggregation level of elements described by them (from the most coarser grained to the finest grained).

According to \citeonline{DillenbourgJermann2007}, a CSCL script can be classified in one of the four levels of abstraction defined as follows as:

\begin{description}

\item[\emph{Script Schemata}:] are scripts used to describe the core design principles whereby is expected to trigger interactions among participants in the CL process. In this sense, these scripts are defined in a content free didactic form, so that they can be used to describe patterns of CL. Examples of script schemata are the Jigsaw script \cite{Aronson1978, KordakiSiempos2010}, conflict script \cite{WeinbergerErtlFischerMandl2005}, and reciprocal script \cite{King2007}. The jigsaw script describes a CL scenario in which the principle of interaction consists in the grouping and re-grouping of participants with complementary information to share their knowledge. The conflict script describes a CL scenario to group learners with contradictory knowledges or opinions to instigate the discussion. The reciprocal script describes a CL scenario that assigns alternate roles to the students for facilitating questioning and tutoring activities.

\item[\emph{Script Classes}:] are the specialization of scripts schemata defined to facilitate their adoption in a particular learning context. In scripts of this type, the content-domain and the student-data contain restrictions related to the learning context for which they are intended. For instance, the Universanté Script \cite{DillenbourgJermann2007} is a script class based on Jigsaw schema in which the constraints established to define the learning context in which it could be used are: having different thematic groups and participants from different nations.

\item[\emph{Script Instances}:] are scripts in which the content-domain are specified for a particular situation. The script instances are script classes with concrete content-domain but they are independent in participant-data. e.g. who are the participants in the CL scenario, and what are deadlines for the activities.

\item[\emph{Script Sessions}:] are the scripts used in the execution stage as support to orchestrate and coordinate the interactions among the participants. In a script session, the content-domain and participants-data have been specified in the most concrete level to be directly executed by a learning environment. The CL scenario described by a script session is commonly known as “\emph{CL session}”, and, in general, it has been obtained when the teacher or educational staff define the participants and deadline of CL activity in a script instance.

\end{description}

Regarding to the level of granularity \cite{FischerKollarStegmannWeckerZottmann2013}, the CSCL scripts can be classified in macro-scripts and micro-scripts.

\begin{description}

\item[\emph{Macro-scripts}:] are scripts that basically describe the CL process in a courser-grained level without detailing the specific interactions among participants. A macro-script describes how to attain a set of pedagogical objective indicating the sequencing of individual and group activities that must be follow by participants. Thus, for example, in the Jigsaw macro-script, to promotes the individual accountability and positive interdependence, the sequencing of activities consists in three activities: an individual activity, expert group activity, and jigsaw group activity. In the individual activity, each student studies a particular part of a whole problem. In the expert group, the students of different groups that study the same part of the whole problem meet together for exchanging ideas. At last activity, students of each jigsaw group meet to contribute with their expertise to solve the whole problem.

\item[\emph{Micro-scripts}:] are scripts that describe the CL process in a fine-grained level \cite{WeinbergerFischerStegmann2005}. They indicate, for example, the dialogues that must happen among student to achieve the pedagogical objectives, and they are intended to describe the communication model between participants. Thus, for example, to facilitate the negotiation and elaboration of a domain concepts, Weinberger, Ertl, Fischer, and Mandl \cite{WeinbergerErtlFischerMandl2005} describe a micro-scripts for online peer discussion using a sequence of sentence openers (e.g. my proposal for an adjustment of the analysis is….) that prompted learners to contribute with the discussion and critique one another's contributions.

\end{description}

The macro-scripts and micro-scripts have a hierarchical relationship to describe the CL process. The micro-scripts describe the communication process for a CL activity \cite{WeinbergerFischerStegmann2005}, and the macro-scripts describe groups, roles, and flow of CL activities \cite{DillenbourgHong2008}. This explicit hierarchical relationship is employed in some models and tools to support a combined a full-complete design of CL scenarios \cite{AlharbiAthaudaChiong2014, ChallcoBittencourtIsotani2016}. \citeonline{Hernandez-LeoVillasclaras-FernandezAsensio-PerezDimitriadisRetalis2006} propose a hierarchical model in which schemata and classes of macro-scripts and micro-scripts are used as templates to generate scripts in the different levels. In the work of \citeonline{ChallcoGerosaBittencourtIsotani2014}, the hierarchical relationships of macro-scripts and micro-scripts is employed to support the automatic generation of unit of learning based on hierarchical task networks.

\subsection{Instructional Design Process of CL Scenarios and Computer-Based Systems}

\label{sec:instructional-design-process-cscl-scripts}

The abstraction and granularity levels of CSCL scripts provide different benefits for the instructional design process of CL scenarios \cite{AlharbiAthaudaChiong2014, ChallcoBittencourtIsotani2016, MiaoHoeksemaHoppeHarrer2005}. This section discusses these benefits and the characteristics of some computer-based systems used to support the instructional design process of CL scenarios . \autoref{fig: diagram-instruction-design-tools} shows the diagram proposed by \citeonline{\cite{ChallcoBittencourtIsotani2016}} to classify these computer-based systems according to the stages defined in the general model of instructional design process.

\begin{figure}[htb]

\caption{Classification diagram of computer-based systems to support the instructional design process of CL scenarios}

\label{fig: diagram-instruction-design-tools}

\centering

\includegraphics[width=0.95\textwidth]{images/chap-general-background/diagram-instruction-design-tools}

\fdireta{ChallcoBittencourtIsotani2016}

\end{figure}

During the \emph{design stage}, script schemata and script classes are used as templates by authoring tools to support the definition of the core elements in a CL scenario. In these tools, the information are extracted from scripting patterns, and then, used as recommendation to help the instructional designers to select the more suitable elements according the instructional/learnings goals for which the CL scenario is intended. In this sense, (Web)Collage is an authoring tool in which, to help the users to define the learning activity flow in macro-scripts, the system provides advices

provides advices for the definition

To help users create CL scenarios, the system implements and uses interaction patterns

based on CL best practices. Thus, instead of creating a scenario from scratch, a user can use

these best practices that are provided as templates for guiding the design. Currently, it

implements six interaction patterns including brainstorming, jigsaw, pyramid, simulation,

TAPPS, and TPS. Figure 2.3 shows the initial interface of the system where it is possible to

select the patterns and configure the learning objectives.

The interface of Collage is especially friendly for novice designers and teachers who want

to introduce CL activities in their classes. To design a scenario in Collage is quite simple. The

user selects the main objectives for a CL session and the system can identify the interaction

patterns that support the achievements of these objectives. Then, the user can choose and

instantiate an interaction pattern to fit in a specific domain. It is also possible to associate CL

activities with available learning resources.

instantiate these scripts and to obtain CL sessions

they may provide advice along the design process. In addition, CSCL scripting

patterns whose solutions propose structures of scripts can be represented computationally and

implemented in authoring tools as a kind of templates that can be easily completed in order to create

computer-interpretable scripts.

To create CSCL scripts, teachers also need to select tools (not to generate them) that are to

support the activities. In this line, semantic search of tools using ontologies is being researched by

Vega-Gorgojo, Bote-Lorenzo, Gómez-Sánchez, Dimitriadis, & Asensio-Pérez (2005). Therefore,

some patterns at the resource level (this is applicable to tools and learning materials) can act as a

mediator between the resource searchers and the user.

During the instantiation of a CSCL script, tools for managing roles and groups are also

necessary. This type of tools should easily enable the creation of multiple groups or roles and the

further binding of individuals according to the knowledge captured in the patterns and the patternbased

structure of a script, which may be quite complicated.

Regarding the interpretation (i.e. execution) of CSCL scripts, the most important types of tools

are players and LMSs. A system that interprets CSCL scripts should consider the information

collected in the patterns. That is, it should be able of interpreting scripts at the learning flow level

or/and at the activity level, provide the needed resources, etc. (Bote-Lorenzo, 2005). The

information captured in patterns may be also used for feedback or adaptation purposes. In addition,

CSCL scripting patterns can be used by awareness tools. For instance, a CL flow awareness tool

(considering the patterns at the CL flow level) will allow participants to be aware of the

collaborative learning flow during execution: which activities have been accomplished, which are

the next ones, in which activities are involved the rest of the participants, etc. In many CL

situations, having such awareness is crucial since participants may change their groups depending

on the phase of the learning flow and may need to know the progress of their future team partners

\cite{MagnisalisDemetriadis2012a, PrietoAsensio-PerezDimitriadisGomez-SanchezMunoz-Cristobal2011,Alario-HoyosBote-LorenzoGomez-SanchezAsensio-PerezVega-GorgojoRuiz-Calleja2013}.

give recommendations according to learning goals. For example,

instation of provide

many authoring tool to describe the core design principles of CL scenarios.

Different benefits from the use of script schemata and classes as patterns are obtained in the instructional design process of CL scenarios \cite{AlharbiAth0/0/00 0:00:00 AMaudaChiong2014, ChallcoBittencourtIsotani2016, MiaoHoeksemaHoppeHarrer2005}.

During the authoring and design stages, repositories of script schemata and classes facilitate the sharing and reuse of these scripts in distributed learning environments \cite{PrietoAsensio-PerezMunoz-CristobalDimitriadisJorrin-AbellanGomez-Sanchez2013, PrietoTchounikineAsensio-PerezSobreiraDimitriadis2014},

and the structures of script schemata and classes are used as templates to create new script schemata and classes \cite{AndreasHarrerH.UlrchHoppe2007, RonenKohen-Vacs2009}.

Benefits of granularity and

Thus, The script sessions require a formalization in computer-readable form to be executed in a learning environment and to support the CL process.

These scripts are object of study in this PhD thesis,

During the production stage, script schemata and classes provide advice and recommendation that help

Script schemata and classes facilitate the generation of computer-interpretable scripts, they provide information to support the search of applicable learning material and tools for the CL scenario \cite{Bote-LorenzoVaquero-GonzalezVega-GorgojoDimitriadisAsensio-PerezGomez-SanchezHernandez-Leo2004, IsotaniMizoguchi2008a, Vega-GorgojoBote-LorenzoGomez-SanchezDimitriadisAsensio-Perez2005}.

The script schemata and classes are also uses to obtain recommendation about how to bind individuals in groups and roles according to the knowledge described in these scripts \cite{IsotaniMizoguchiIsotaniCapeliIsotanideAlbuquerqueBittencourtJaques2013,Villasclaras-FernandezHernandez-GonzaloLeoAsensio-PerezDimitriadisMartinez-Mones2009}.

As can be noticed above, the macro-scripts and micro-scripts have a hierarchical relationship to describe the CL process of CL activities. The micro-scripts describe the communication process in a CL activity \cite{WeinbergerFischerStegmann2005}, and the macro-scripts describe groups, roles, and flow of CL activities \cite{DillenbourgHong2008}. Despite this explicit hierarchical relationship, there are few models and tools in which all the elements of macro-scripts and micro-scripts are combined to support the design of CL scenarios \cite{AlharbiAthaudaChiong2014, ChallcoBittencourtIsotani2016}. \citeonline{Hernandez-LeoVillasclaras-FernandezAsensio-PerezDimitriadisRetalis2006} propose a hierarchical model in which schemata and classes of macro-scripts and micro-scripts are used as templates to generate scripts. In the work of \citeonline{ChallcoGerosaBittencourtIsotani2014}, the hierarchical relationships of macro-scripts and micro-scripts is represented as hierarchical task networks to support the automatic generation of unit of learning.

In the CL ontology \cite{IsotaniInabaIkedaMizoguchi2009}, and therefore in the ontology OntoGaCLeS, the hierarchical relationship of macro-scripts and micro-scripts is not explicitly described as a direct link between macro- and micro-scripts. The hierarchical relationship is implicitly described as part of the conceptualization of events and processes proposed by Galton and Mizoguchi \cite{GaltonMizoguchi2009}. Based on in this conceptualization in which the representation of an event can be constituted by many distinct sub-events to describe a process, the hierarchical relationship of macro- and micro-scripts can be inferred from these events that are explicitly described in the CL ontology and the ontology OntoGaCLeS.

\ddddd

Having the description of CSCL scripts in natural language does not allow the computers programs to interpret them, and to give support for the execution of a CL scenario following the instructions indicated by the scripts. In this sense, to represent the CSCL scripts in a computer readable manner, the IMS-Learning Design\footnote{\url{http://www.imsglobal.org/learningdesign/}} (IMS-LD) specification has been adopted by different tools, such as (web)COLLAGE \cite{Hernandez-LeoVillasclaras-FernandezAsensio-PerezDimitriadisJorrin-AbellanRuiz-RequiesRubia-Avi2006,Villasclaras-FernandezHernandez-LeoAsensio-PerezDimitriadis2013}, CIAN \cite{MolinaRedondoOrtega2012}, LeadFlow4LD \cite{Palomino-RamirezBote-LorenzoAsensio-PerezDimitriadis2008}, NUCLEO \cite{SanchoFuentes-FernandezFernandez-Manjon2008}, CoLearn \cite{StylianakisArapiMoumoutzisChristodoulakis2013}, CeLS \cite{RonenKohen-Vacs2009}, and LAMS \cite{Romero-MorenoOrtegaTroyano2007}, as the language to describe CSCL scripts.

Despite the benefits that brings the use of the IMS-LD specification to represent CSCL scripts, several researchers indicate that this language is insufficient to fully support the modeling of CSCL scripts \cite{AlharbiAthaudaChiong2014, CaeiroAnidoLlamas2003}. The purpose of IMS-LD specification is not to provide a full support for describing CSCL scripts. We cannot describe in the IMS-LD the goal groups, the conditions

the IMS-LD has been developed as a neutral, generic and flexible educational modeling language to describe a wide range of pedagogies approaches (the teaching strategies, pedagogical goals and their associated activities) \cite{Koper2005}. In this sense, to support the representation of CSCL scripts in a computer-readable manner, a wide variety of extensions on the IMS-LD elements has been proposed in by several researchers \cite{Bote-LorenzoVaquero-GonzalezVega-GorgojoDimitriadisAsensio-PerezGomez-SanchezHernandez-Leo2004, LeoPerezDimitriadis2004, MagnisalisDemetriadis2012, MiaoHoeksemaHoppeHarrer2005, Vega-GorgojoBote-LorenzoGomez-SanchezDimitriadisAsensio-Perez2005}.

Instead, to simply provide a computer-readable representation of CSCL scripts, the work of \citeonline{Isotani2009} proposes the formalization of these scripts in a computer-understandable manner through the use of ontologies. This solution consists in a set of structures, known as ontological structures, that makes the description of CSCL scripts more semantically-rich. In the ontology, the learning goals, purposes, and other relevant information are expl that cannot be represented using the IMS-LD specification, i.e., learning strategies, group goals, interaction patters from learning theories. Providing this formalization in the CL ontology, \citeonline{IsotaniMizoguchiIsotaniCapeliIsotanideAlbuquerqueBittencourtJaques2013} demonstrates that intelligent-theory aware systems can interpret these scripts and provide advice and recommendation to support for the modeling of learners' development \cite{InabaIkedaMizoguchi2003}, the formation of effective groups \cite{IsotaniMizoguchi2008a}, and the instructional design of CL activities \cite{IsotaniMizoguchiIsotaniCapeliIsotanideAlbuquerqueBittencourtJaques2013}.

The abstraction and granularity give them an enormous flexibility to be reused in the instructional design process of CL scenarios as detailed in the next subsection. Thus, multiple scripts are used to describe different stages and of development and of CL scenario.

\\*\*\*\* clycle of lyfe of a cscl scripts and

\\*\*\*\* tool to support the clscl

\section{Gamification of Learning and Instruction}

\label{sec:gamification}

\subsubsection{Player Types Models}

For example, the Bartle’s player type model has been compiled into four portraits (killer, explorer, achiever, and socializer) to determine what is fun for players populating virtual worlds, such as the virtual worlds defined by MUD (Multi-User Dungeon role-playing games) and MMORPGs (Massively Multiplayer Online Role-Playing Games) \cite{Bartle1996, 2004}.

Later marketing literature has attempted to more accurately reach different modes of segmenting customers. The goal of segmentation is to identify groups of people that are as homogenous as possible, but that differ from each other in a significant way. In marketing literature, the following four overarching categories of segmentation have acquired an established standing appeal a particular audience. These portraits described as taxonomies of player types are established to indicate the features and requirements of games are defined through the identification of. These portraits.

could be obtained from player type models in which

These taxonomies are adapted

For example,

\section{Ontologies and Ontology Engineering}

\label{sec:ontologies-and-ontology-engineering}

\section{continue here}

caCL ontology and the ontology OntoGaCLeS represent the

as a set of events, and these events can also be part of events that represented CL activities described in a macro-script \cite{Villasclaras-FernándezIsotaniHayashiMizoguchi2009}.

, the communication process among the participants in a CL activity

Concluding remarks

This ontology was used as the basis of ontology to gamify CL scenarios (*OntoGaCLeS*), and its concept and formalization are presented in the Section **Error! Reference source not found.**.

\section{Gameplay, Gameplay experience and Gameful design}

user experience in a game-like system is known as gameplay experience, and its consists in the player’s sensation, thoughts, feelings, actions and meaning-making produced by the interaction between the player and the game-like system \cite{DjaoutiAlvarezJesselMethelMolinier2008, MäyräErmi2005, Richard Rouse2004}.

design consists in guidelines about how the game elements in the game-like system interacts with the users to lead them to have an enjoyable and motivated experience \cite{Schell2008}. The gameplay experience is different for each person and