Challenges and Potentials of User Involvement in the Process of Creating Games

Jörg Niesenhaus
University of Duisburg-Essen, Germany
joerg.niesenhaus@uni-due.de

Abstract. This article gives a short overview about the history of user involvement in the area of digital games and describes the specific challenges and potentials of the participation and motivation of users in this application area. It specifies the different degrees and types of user involvement and outlines the current state of the art. Moreover, the article discusses the implications of user involvement for game companies and users with a special regard to user-generated content and gives an outlook on future development.

Introduction

Within the past years the game industry has gone through an overwhelming economic growth and analysts foresee a strong growth in the nearby future as well (despite the current depression). The branch of game development and publishing is already a major industry with its strongest markets in North America, Japan, and Europe. The nine European core markets[1] sold video and computer games worth € 7.3 billion (excluding hardware sales) (Nielsen Media Research, 2008) in 2007. Games software sales in the U.S. recorded € 6.9 billion (9.5 billion dollar) in 2007. The annual growth of the industry ranges from 17% (U.S. sales between 2003 and 2006) (Entertainment Software Association, 2008) up to 21% (German sales between 2006 and 2007) (BIU - German Association for Interactive Entertainment Software, 2008).

---

[1] The European core markets are Great Britain, Germany, France, Italy, Spain, the Netherlands, Switzerland, Sweden, and Finland.

User involvement in Digital Games

It is not a new phenomenon that user communities participate in the development or improvement of a product. In fact, it is extremly rare to find industries that do not attract user involvement (Edery and Mollick, 2009). Users do not only gather around software products, they build communities around sports, cars, food and even healthcare products. So, what is so special about user involvement in the area of digital games?

At first, the relation between game company and gamer looks like many other relations between producers and consumers: The companies are looking for profit and the users want to solve their problems, gain status in their community or - in most cases - just want to have fun (Edery and Mollick, 2009).

But over the years, game companies have learned to align the needs of user communities with their own needs by offering awards and incentives that make it entertaining and interesting to focus on user involvement (Edery and Mollick, 2009). The game companies provide the users with toolkits and support, which makes it easier for the community to get involved and to share the results. But the key factor is that most of the companies have staff focussing only on the support of the community. So called 'community managers' know the gaming community's own language and rules, are able to support the users and, even more important, guide the user participation into a positive direction for business and innovation to avoid e.g. piracy issues (Wera, 2008).

More than any other industry, video game companies have succeeded in channeling the positive aspects of user involvement not only to extend lifecycles of products but also to create valuable sources of innovation for the industry (Kücklich, 2005). How this was achieved and to what extent today's user involvement in games evolved will be explained in this article.

History of User Involvement in Digital Games

User-generated content and user participation in the context of digital games exist as long as the games itself but the possible degree of participation changed a lot over the years. Back in 1962 computer science students enhanced the original 'Space War'[2] game by adding new features and brought it back to the community afterwards, making it one of the first digital game modifications. Adding new features to a game in 1962 meant to reprogram specific parts of the original game which was reserved to a small group of skilled hobby programmers and computer science students. At that time, modifying commercial games was often connected with the use of hex editors to manipulate binary game files. While some people just 'cheated' by changing the binary code e.g. to make their avatar invulnerable, others created additional content by doing a great job of reverse-engineering.

---

[2]   'Space War' is one of the first computer games and was created by Steve Russell at a PDP-1 computer in 1961 at the MIT.

Years later, the first games with GUI-based toolkits[3] emerged, offering non-programmers the opportunity of generating content for games. With the release of 'Doom'[4] in 1993 gamers started not only to record their own movies based on game engine technology (called 'machinima'), but they also created more than 12.000 modifications of the game (Kushner, 2004). Although more and more people got involved in creating or modifying game content, they still remained as a small community without an access to a broader audience.

But this all changed when the internet became a mass medium. People could now share their own thoughts and creations with a continuously growing community. The user interaction via different web platforms, wikis and boards made it possible to work collaboratively on larger projects. One of these collaborations changed the game industry's view on user participation. Started as a small community project the modification 'Counter-Strike' for the game 'Half-Life' got so popular that Valve, the publisher of Half-Life, offered the community team contracts and made 'Counter-Strike' a part of the Half-Life franchise in 1999. Until now, 10 million products under the 'Counter-Strike' label were sold[5] and, in addition, extended the lifecycle of 'Half-Life' a lot. Impressed by the success of 'Counter-Strike' a lot of game publishers and development teams put more resources in offering better toolkits to the community or started releasing their engine source code to the public[6].

State of the Art

Nowadays, several online and offline games have huge fanbases which gather around single products or whole franchises and series. In most cases, web-based platforms work as a basis for the online communities which are established by the game's publisher or developer[7], by online and print magazines[8], or by individuals. In addition, games like 'Little Big Planet'[9] provide a first impression of the potential of integrating community platforms or features directly into the game, creating a smooth transition between the game and the user involvement. All community platforms have different focuses (or a combination of them): Some discuss game ratings, other provide users with helpful guides and walkthroughs or provide users with self-made addons and content.

---

[3]   Lode Runner offered one of the first level toolkits in 1983 (Amiga 800, Broderbund Software).
[4]   First-person shooter game by id Software.
[5]   Not included are downloaded versions of the franchise which were sold over Valve's internet platform 'Steam'.
[6]   As one of the most popular 3d gaming engines at this time, the 'Quake' engine by id Software, was published under the GPL in 1999.
[7]   The official 'World of Warcraft' community was established by the developer 'Blizzard Entertainment': http://www.wow-europe.com/de/community/ Last visit: 01/12/09
[8]   An example for a popular game magazine community is www.gamespot.com Last visit: 01/12/09
[9]   Jump'n Run game for the Playstation 3, developed by Media Molecule. Released in 2008.

Table I. Different types and degrees of user involvement in games.

Degree of Part.	Type of User Involvement	Preconditions of Users	Tools	Examples	Future Tools & Technologies
High	Creating own digital games	-Experienced Users -High skills in programming and 3d modelling tools	-Microsoft XNA -Visual Studio C++ -Maya, 3DS Max -Game-Engines	-Alien Hominid -Braid -Ragdoll KungFu -Crayon Physics	-MS Kodu -Programming by learning -Silverlight
High	Creating mods & total conversions	-Programming and scripting skills -Able to create own textures and models	-Engines, Toolkits -Visual Studio C++ -3DS Max, Blender -Photoshop	-Counter-Strike -WC3 DotA -Insect Infestation -Tower Defense	-Genetic algorithms -MS Popfly
Medium	Building content and assets	-Good understanding of game mechanics and / or design -Toolkit knowledge	-Level toolkits -Object editors -Photoshop	-Little Big Planet -The Sims 1 & 2 -Forza Motorsport -Spore	-Content creation will be part of the game
Medium	Writing fan fiction or creating fan art	-Talent for writing or arts	-Pen & Paper -Office applications -Photoshop -Web platforms	-Red vs. Blue (Machinima) -Blizzard.com/ Inblizz//Fanart/	-Extended video authoring tools
Low	Writing reviews & walkthroughs	-Basic understanding of game design, basic writing skills	-Office applications -Webpages -Blogs	-Gamespy.com -Gamefaqs.com -Supercheats.com	-Future community platforms
Low	Playing the game	-Interest in the genre and game world	No tools required	-Data Mining -Logfile Analysis	-Evaluation algorithms

Table I shows my proposal of a categorization of different types and degrees of user involvement in the gaming industry. For most of the higher degrees of participation additional tools and software are needed, and moreover, inside knowledge and experience of using the given tools.

By playing a game the 'user involvement' is limited to logfile analysis of player data or behaviour which is supported passively by the user. Several companies do research about their games and evaluate data of e.g. online gaming sessions in order to find bugs or resolve balancing issues.

The active participation starts with gamers giving feedback about their gaming experience e.g. in official boards or questionnaires of the developer. Gamers also write reviews, give recommendations how to solve technical issues or write whole walkthroughs for their favorite games. An economic study discovered that gamers reduce supporting costs of companies by helping each other to solve technical problems, to the extent that they solved 1.300% more problems than the support staff of the companies (Jeppesen, 2005).

A more creative extension of the writing skills are the generation of stories going beyond the game's plot (fan fiction) or creating own fan art works or machinima movies with characters or settings of the game world. The web platform 'Mod the

Sims 2'[10] has 670.000 active members who wrote more than 123.000 stories, many of them illustrated with movies taken in the game.

Even more ambitious and time-consuming is the process of creating own content which can be implemented in a game. This does not only imply knowledge about the game world and - if applicable - the game mechanics, but also in how to use the tools in order to create appropriate content. While some games extend their lifespan by level editor tools allowing the community to create endless levels and stages, other editors allow players to create characters or costumes (e.g. for the game 'Spore'[11] people created more than 2.5 million different creatures) to broaden the variety of the scenarios. While some tools give the users all the options they need to create content, some need additional support of scripting languages like LUA or texture painting tools. Game community members also release their own patches in order to clear bugs[12] or build interface modifications[13] to optimize their gaming experience through additional functionality and better usability.

Modifications often base upon the engine technology and the system of rules of one specific game but change the visual appearance or the gameplay so that the experience of playing the modifications differs slightly or significantly[14] from the original game. Creating a professional modification often needs a whole team of hobby programmers and artists, e.g. to program new game mechanics and model new graphical content. Current editor tools for games offer a lot of possibilities of participation in the creation of additional game content or complete games. The most powerful editors in terms of degrees of freedom and state of the art technology are toolkits of first-person games like 'Crysis'[15] and 'Half-Life 2'[16]. These tools offer physics engines, complex lighting and shader systems, AI scripts and a lot more. But only high skilled users can access and use all the tools to reach a product that matches commercial game standards. In most cases, it takes at least a basic knowledge of additional 3D modelling tools like 'Maya' or '3D Studio Max' and a scripting or programming language to use the full power of these tools.

The highest level of user involvement is the creation of own original games by using tools or engines from available games. Creating a new game challenges not only the technical knowledge of the participants but sets also high demands for the creativity needed to build a game world with working systems of rules and logics, believable characters, interesting plot and so on.

---

[10] http://www.modthesims2.com/ Last visit: 01/12/09
[11] 'Spore' was released 2008 by Electronic Arts and lets the player create creatures at different evolutionary stages of the game.
[12] So called 'community patches' often emerge when the development studio cancels the official support for a popular game.
[13] On the platform www.curse.com community members are sharing more than 4.500 self-written interface modifications for the game 'World of Warcraft'. Last visit: 01/12/09
[14] Modifications with significant changes in gameplay and visual appearance of a game are called 'total conversions'.
[15] Awarded first-person shooter released 2007 by Electronic Arts.
[16] Released 2004 by Vivendi Universal but still popular due to modifications and extensions.

Implications for Publishers and Users

Gamers decide to what degree they want to get involved. They can spend both time and creativitiy for a product they like, or lean back and consume the content created by others. Community members no longer have to wait until their idea will be implemented - they can grab the provided tools and create additional content for a game on their own. Futhermore, they do it not only for themselves, but for thousands of players. The attention of the other community members, playing their level, using the interface modification or just giving positive feedback and recommending it to others is one of the highest rewards for a 'modder'. An even higher but much more uncommon award is the feedback of the developers, by showing interest for the gamers work or even integrating it into the next version of the game.

Publishers and developers get a lot of relevant feedback through user involvement, not only in terms of questionnaires, board postings or reviews (Niesenhaus and Lohmann, 2009) but also in form of e.g. game modifications. With each modification of his game the developer learns something about the gamer's needs and wishes and can integrate well-received ideas into his future products. A survey of modders (Prügl and Schreier, 2006) of 'The Sims' revealed that over half of all active modders spent more than six hours a week developing new content for free, and a smaller group of 12% spending more than 20 hours a week. Given this enormous potential of motivated users participating in the creation of content or support of a product, the possibilities for game companies saving money and generating additional benefits are obvious (Postigo, 2007). As a matter of fact, the investment for a web platform and the tools to support the user participation is comparatively small-sized, seeing the potential of benefits like extending the lifecycle of products, getting free viral marketing campaigns and higher sales numbers.

There are also some downsides of user involvement for the game companies, though. Well-done game modifications can not only increase the value of a game, but may also distract customers from the publisher's own official expansion products (Edery and Mollick, 2009). Moreover, communities are not only able to increase the gamer's loyalty for a product but can also work as an amplifier when things are not working as intended. Another problem game companies are facing are copyright violations, and moral or ethnic offences caused by user-generated content.

Benefit and Cost Calculations for the Participating Users, Publishers and the Community

Given the fact that some *participating users* spend more than 20 hours a week creating content for their favorite game without the perspective of a monetary benefit (Prügl and Schreier, 2006), the calculation of costs and benefits becomes obvious for them. Of course, some developers offer incentives for the outstanding community members but - from an economic perspective - these incentives still are out of proportion to the hours of work contributed by the gamers. This leads to the as-

sumption that most users do not judge the incentives by their monetary value but by emotional factors or prestige.

Table II shows various game-related incentives with their user-perceived value in comparison to the real costs for the publisher or developer providing these incentives. Although it may be easy to understand that incentives with high monetary value are perceived as high-valued by the users, it is not general necessary to spend a lot of money in order to create appealing incentives for the community as the examples illustrate.

Table II. Relation between user-perceived and real value of game-related incentives.

Incentives in the area of games	User-perceived value	Costs	Examples
Trips to trade fairs or meet and greets	High	High	Ticket and trip to game show
Product-related merchandise	Medium	Medium	Action figure of game character
Product-related virtual goods	High	Low	Ingame items with special abilities in multiplayer games
Access to beta versions	Medium	Low	Earlier access to closed beta
Fame & reputation	Medium	Low	Special discussion board ranks

An interesting factor is the perception of special game world-related items (also called 'ingame items') in multiplayer games or special ranks for discussion boards. Whereas the creation of a special version of an existing virtual item or the creation of a new board rank is only a small investment for the developer, the prestige in the community will be orientied on the availability of the item or title and the effort involved to achieve it. Another prestigious incentive for gamers is an earlier access to the beta version of an upcoming game because some developers and publishers give away 'beta codes' only to the most successful or 'hard-playing' users of previous games. Codes which are given away randomly or via contests do not have this prestige, of course.

Some platforms like Microsoft's 'XBox Live Marketplace' or Apple's 'App Store' show also first possibilites of how to sell user-generated games. While the Xbox Live Marketplace will sell only products evaluated extensively by Microsoft, the App Store allows hobby developers to sell their own applications and games with very few restrictions, making it possible to generate monetary benefits out of self-made games.[17]

---

[17]   Both platforms cost a fee for developers. In case of Microsoft's 'Xbox Live Marketplace' developers can keep up to 70 percent of the revenue generated by their games.

The benefit and cost calculations for *publishers and developers* may seem easier to manifest at first glance. Some studies estimate the amount of work users contribute to a certain product and try to determine the value by comparing the amount of user work to company standards (Jeppesen, 2005). These estimations are appropriate in order to get a rough impression of the overall investment of all users involved in a game product but they often fail to provide an accurate estimation of the contribution's value. The following factors are often disbanded when it comes to estimations of the value of user involvement:

(1) *Quality of user involvement.* Based on the fact that there is often no gatekeeper mechanism controlling the quality of user-generated content before it goes online, the quality of user-generated content is mostly very heterogenous. Keeping the high numbers of content creations of the games 'Spore' (2.5 million creatures) or 'Little Big Planet' (300,000 level designs) in mind, it is not possible to evaluate every piece of content unless there is the possibility to establish an automatic review process. Hence, it is not possible to imply an average level of quality when calculating the value of the user involvement.

(2) *Viral marketing effects.* It is very difficult to calculate the value of viral marketing for a game generated by its user involvement, because the impact of the different communication channels is nearly impossible to measure. Users often promote their own creations via Youtube videos, blog entries, board postings, or tell friends via instant messager systems or ingame messages about their creations. From time to time, user creations are featured by gaming magazines and portals, again amplifying the marketing effects for the content and the related game.

(3) *Effect on sales numbers.* Even more difficult to put in numbers are the effects of the user involvement of a specific product on its sales numbers. A purchase decision of a user is a complex and multi-facetted process with not only rational but also emotional factors having an influence on the final decision to buy a product or not. User involvement, especially user-generated content, is often supposed to have an influence on sales numbers, especially regarding long-term sales, but there are no sufficient studies proving these assumptions, yet.

Despite this criticism on the estimated calculations of the value of user-generated content, some of these calculations help to get a better idea of the effort contributed by the involved users, e.g. by trying to compare the overall development costs for a game to the value of the user-generated content of this specific game. The following example estimates the costs and the value of the user-generated content of the successful game 'Little Big Planet' released for Sony's 'Playstation 3' platform in 2008:

A game developer costs an U.S. company around 90,000 dollars a year including all taxes and supporting costs (Siwek, 2007). A creation of a basic and runnable level design may cost a high-skilled developer one day, which makes about 350 dollars of the companies costs. More complex and high-quality designs may lead

up to one week or more, but in comparison to the quality of the user-created designs only one day per level design is calculated. Now, take 300.000 level designs created by the users of 'Little Big Planet' and multiply them with the costs of a basic level design done by a developer. This ends up with a total content value of more than 100 million dollars.

The developer of the game has about 25 employees[18] and the development of 'Little Big Planet' took about 2,5 years. This leads to estimated development costs of approximetly 5 million dollars including hardware and software licenses[19].

This roughly estimated example shows that the development costs of 'Little Big Planet' are considerable lower than the value added by the users. The example also shows impressively, that the participation of users allowed the developers to create a game with a higher content value than they could have done on their own.

Regarding this example from the perspective of the *community*, it is clearly a best-case scenario: The community members pay the same amount of money for a game with user-generated content support as for other games but get far more content to play through (even if it is unlikely that there are users who will play through all available levels). Everyone who wants to participate in the creation of content actively is able to but he or she can also just benefit from the additional content by consuming it. A downside of the user involvement for the community can be the already mentioned heterogenous quality of e.g. user-generated content in combination with a lack of feedback or rating systems. Community members may have to download several content units until they find something they are looking for.

Lessons for Other Application Areas

A lot of lessons learned by the games industry are also well-established in other application areas. Sharing code resources and modification tools is not a unique feature of the games industry but it is not being used to a comparable extent in other commercial application areas. Electronic Arts, the worldwide second biggest game publisher released toolkits allowing users to create the avatars for the game 'Spore'. Can you imagine Microsoft handing out tools to users in order to develop the next 'Karl Klammer'[20] for MS Office? It is comprehensible that some companies fear negative consequences of the user involvement: Having their product not under full control might lead to unintended results like inconsistencies in the overall product appearance or - when it comes to code modifications - errors in the system functionality or the logical structure.

The area of games shows that these problems can be limited to a minimum by setting up boundaries for the influence of the user creations. A successful example for establishing clear boundaries is the game 'World of Warcraft' which allows

---

[18]   Due to the fact that developers start often with a small core team at the beginning of the development, the first year of development was calculated with 10 employees

[19]   The estimated calculation consists of employee salaries, taxes, hardware, software licenses and office rental

[20]   Well-known but often critizised avatar which shall support users of Microsoft Office.

players to write their own interface modifications by using XML and LUA. These modifications do not only change the appearance of the interface but, even more important, enable also the creation of shortcuts and macro functions.

In contrast, artistic content creations cannot be controlled by code boundaries or algorithms but there are other possibilities of rejecting flawed content. Electronic Arts lets the users of 'Spore' sort out the avatars which do not meet their overall expectations. Other games like 'Little Big Planet' use rating, tagging and review functions to ensure that gamers can search for high-quality content. These methods introduce a kind of subsequent gatekeeper mechanism and help users to orientate theirselves in the mass of content.

When it comes to the motivation of users not every gratification model in the context of games is transferable to other application areas. Product-related merchandise incentives like action figures, posters or virtual goods do not work for other application areas due to the absence of strong characters, storytelling and a persistent game world one can relate to. But other incentives, like user reputation work also in a social context in other application areas.

Another important lesson from the games industry is to recognize and to know your users in order to create a mutually beneficial situation in which both the community and the developers are happy (Edery and Mollick, 2009). Game developers and publishers often provide boards or even regular chats where gamers can meet the game developers or community managers to discuss current issues or improvements of the product and new ideas. A study about the motives of 'user innovators' showed that they are highly motivated by feedback from the company that created the product which serves as the basis for their innovation (Jeppesen and Frederiksen, 2006). In contrast to these positive experiences, some application areas stay very anonymous e.g. when asking for user feedback. A good example are the bug report systems of several operating systems or office applications which ask the user to comment on an error. After sending the message, the user neither has the chance to get any feedback from a developer related to his comment nor to be aware of the overall status of the specific error.

These examples show how the application area of digital games creates ways to channel the user involvement and innovation into positive directions for both the developers and the users. This relationship is strengthened by the communication between the developers and the gamers and the experience and knowledge that both sides can benefit from it.

Future Development of User Involvement in the Area of Digital Games

Nowadays, the degree of user involvement does not only depend on the motivation of the audience of gamers. The degree of involvement and the quality of the results relate a lot to the tools and interfaces the gamers are using. Seeing the commercial success and the positive media coverage of games like 'Spore' or 'Little Big

Planet', future games will see more and more tools and platforms supporting user involvement and the creation of user content. In addition, to make toolkits more usable and accessible for a broader audience, there are tendencies which try to give the interested users even more degrees of freedom in creating content.

Microsoft already did a good job with the release of the XNA framework[21] and is now going one step further with the game creation tool 'Kodu'[22]. This hybrid form of tool and game will run on the Microsoft gaming console XBox360 and Windows-based PCs and allows kids and adults to create their very own games. 'Kodu' goes beyond creating visual content or stages and levels for games: With a visual programming language gamers will be able to set up rulesets or behaviours of characters and objects and, given this fact, having a major influence on the game design. The interface of 'Kodu' can be controlled completely via the standard console game pad, making it not necessary to type in code commands. Although, visual programming is not new to the academic community - similar approaches were made by e.g. Squeak (Ingalls et al., 1997) and Alice (Pausch et al., 1993) - Microsoft's 'Kodu' looks like an interesting opportunity to teach programming in a creative and motivating way.

Another approach for user-generated games, also in development by Microsoft, is 'Popfly'[23] which currently has beta status. It describes itself as the 'Youtube of applications' and combines the strengths of Microsoft's 'Silverlight' and JavaScript to give users the opportunity to create custom web-applications, multimedia mashups and games. The Silverlight technology is comparable to 'Adobe Flash' which is often used to create browser games or multiplatform titles. These technologies give users the possibilites to develop platform-independent games which can be played in the browser or on mobile phones.

Another interesting future development will be the growing online distribution of user-generated games or content. As mentioned earlier, Apple and Microsoft are already giving users the chance to publish their self-made games on their online platforms and share the revenues with them. As internet connectivity becomes a basic feature for most of the gaming consoles and the mobile phones and handhelds, we will not only see new types of user involvement in the nearby future but also new types of distribution channels and business models for user creations.

---

[21]   XNA is a set of tools with a managed runtime environment that facilitates computer game development.

[22]   Kodu was shown at the CES keynote 2009 in Las Vegas and will be available in Spring 2009.

[23]   More information: http://www.popfly.com/ Last visit: 02/23/09

References

BIU - German Association for Interactive Entertainment Software (2008): *Industry Facts*.

Edery, D. and E. Mollick (2009): *Changing the game. How video games are transforming the future of business*. Upper Saddle River, New Jersey: Pearson Education.

Entertainment Software Association (2008): *Industry Facts*.

Ingalls, D., T. Kaehler, J. Maloney, S. Wallace, and A. Kay (1997): 'Back to the future: the story of Squeak, a practical Smalltalk written in itself'. In: *OOPSLA '97: Proceedings of the 12th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*. New York, NY, USA, pp. 318–326, ACM.

Jeppesen, L. (2005): 'User Toolkits for Innovation: Consumers Support Each Other'. *Journal of Product Innovation Management*, vol. 22, no. 4, pp. 347.

Jeppesen, L. and L. Frederiksen (2006): 'Why Do Users Contribute to Firm-Hosted User Communities? The Case of Computer-Controlled Music Instruments'. *Organization Science*, vol. 17, pp. 45–63.

Kücklich, J. (2005): 'Precarious Playbour: Modders and the Digital Games Industry'. *Fibreculture*, vol. 3, no. 5.

Kushner, D. (2004): *Masters of Doom: How Two Guys Created an Empire and Transformed Pop Culture*. New York, NY: Random House Trade Paperbacks.

Nielsen Media Research (2008): *Nielsen Games Survey: Video Gamers in Europe 2008*. Interactive Software Federation of Europe.

Niesenhaus, J. and S. Lohmann (2009): 'Marrying Knowledge Management with Game Development: Challenges and Potentials'. In: S. Schaffert, K. Tochtermann, and T. Pellegrini (eds.): *Networked Knowledge - Networked Media: Integrating Knowledge Management, New Media Technologies and Semantic Systems*. Berlin, Springer.

Pausch, R., M. Conway, R. Deline, R. Gossweiler, and S. Miale (1993): 'Alice and Diver: A software architecture for building environments'. In: *CHI '93: INTERACT '93 and CHI '93 conference companion on human factors in computing systems*. pp. 13–14.

Postigo, H. (2007): 'Of Mods and Modders: Chasing Down the Value of Fan-Based Digital Game Modifications'. *Games and Culture*, vol. 2, pp. 300–313.

Prügl, R. and M. Schreier (2006): 'Learning from Leading-Edge Customers at The Sims: Opening the Innovation Process Using Toolkits'. *R & D Management*, vol. 36, no. 3, pp. 237–250.

Siwek, S. (2007): *Video Games in the 21st Century - Economic Contributions of the US Entertainment Software Industry*.

Wera, J. (2008): 'Online Community Management: Communication Through Gamers'. *Gamasutra*.