Video Game Localisation for Fans by Fans: The Case of Romhacking

Pablo Muñoz Sánchez

University of Granada

Abstract
The aim of this paper is to show the particularities of the so-called “romhacking”, a methodology developed by amateurs to localise mainly classic video games. In the first section, the concept and origin of the term “romhacking” is presented. The second section offers an overview of the workflow followed by romhackers to localise video games. In the third section, an analysis of the differences between professional and amateur translations is given. The fourth section includes a discussion of the legal aspects of this practice. The paper concludes with a reflection on the impact of amateur translations on the video game localisation industry.
Although the majority of video games are created in Japan or North America, some are never released from their country of origin, despite the fact that they yield large profits (up to 50%) in other international markets such as Europe (Chandler, 2006). Fortunately, the situation began to change in the mid-1990s (Bernal Merino, 2006, pp. 23-24), although some video games still are not localised when released in other countries. For instance, 490,000 copies of *Chrono Trigger* *DS* were shipped to Japan and 220,000 to North America as of December 2008 (*Square Enix*, 2009, p. 19), but it was released in Europe on February 2009 only in English.

On the other hand, some gamers think that today’s video games are not as good as several decades ago, and many nostalgic fans have opted to relive classic video games (Gieske, 2002, pp. 9-10). Indeed, some of the authors who have written about the history of video games, such as Burnham (2001), Poole (2000) or Sellers (2001), consider the 1990s as the golden age of these entertainment products, arguing that today’s outstanding graphics do not compensate for the lack of originality.

Thanks to the advent of the Internet and the so-called “console emulators” (i.e., software that allows playing games of a specific console on a different platform, usually a PC), now gamers can easily download a file containing the binary data of an old game (which is called ROM, i.e. Read Only Memory) and enjoy the old *Pac-Man* or *Space Invaders* as if they had the original hardware and software. Although the ideal aim of emulators is to preserve the cultural heritage of video games (Esposito, 2004, p. 19), gamers now find it possible to play video games that were never shipped from the country where they were developed.

However, as many popular games were released only in Japanese or English, many gamers could not have a complete, enjoyable experience. Thus, some fans started to study the structure of the ROM data of their favourite games, developing tools everybody could use to do
fan translations. These are the origins of the so-called “romhacking”, that is, the process of
modifying the ROM data of a video game to alter the game’s graphics and dialogues.

As the author of this paper has been a romhacker in the past, his aim is to give an
overview of the methodology followed by romhackers to localise video games and to discuss
some of the quality and legal aspects involved in these amateur translations from an academic
point of view.

The Concept of Romhacking

The term “romhacking” originates from the words “ROM” (Read Only Memory) and
“hacking”, as romhackers need to “hack” the ROM data of a game to alter its contents. The word
“hacking” refers here to the fourth definition of hacking in the Merriam-Webster’s 11th
Collegiate Dictionary (2009): “to write computer programs for enjoyment”. In fact, according to
Esselink (2002), the skills of a typical romhacker resemble those of a localisation engineer: “A
localisation engineer doesn’t need to be a programmer, a translator, or a designer. Most
importantly, a localisation engineer should be someone who is not afraid of technology, who
loves to experiment and troubleshoot, and who is extremely flexible” (p. 4).

Generally speaking, the term “romhacking” refers to the process of modifying the ROM
data of a video game to alter the game’s graphics, dialogues, levels, gameplay or other gameplay
elements. A romhacker may have two kinds of projects: editing a game to create new levels and
to change characters’ attributes; or translating it from one language to another. In this paper, only
translation projects are addressed, as the first type is beyond the scope of this research.

Another usual term related to romhacking is “fan translation”. However, this should not
be confused with actual romhacking, because fan translation is simply a translation carried out by
amateurs, no matter the product, as is the case of “fansubs” (subtitles created by amateurs for
Japanese cartoons; see Díaz Cintas & Muñoz Sánchez, 2006). As explained above, a romhacking project does not necessarily involve translation at all.

**The Methodology of Romhacking**

Depending on the complexity and amount of text to translate, romhacking projects may involve a team of several romhackers and translators or only one romhacker who also acts as a translator. For example, the romhacking project of the *Super Nintendo* game *Tales of Phantasia* involved a romhacker and two Japanese-to-English translators from the romhacking group *DeJap* because the script was very long, while the English-to-Spanish translation and romhacking of the *Super Nintendo* game *Secret of Mana* was carried out by only one person, as the script was not that lengthy. Moreover, the text display in each game is unique to that game, so the efforts devoted to romhacking may vary from one project to another.

The following paragraphs offer an overview of a typical romhacking project, although one must note that not all games can be translated following this methodology, as sometimes the text data in a ROM is compressed with different algorithms. For more information on the process, the comprehensive guide of Muñoz Sánchez (2007) explained in detail nearly all the different techniques to extract and insert text in a ROM. The website *Romhacking.net* (2009) also provides a good deal of romhacking documents and tools (the software cited in this paper can be found on said website).

*Text decryption.* The text contained in the ROM data is usually encrypted, meaning that it cannot be read with software like Notepad or Microsoft Word. Therefore, a romhacker must use software such as *SearchRX* or *Search Relative* (Figure 1) to decrypt the text and create a “table”, that is, a file containing the equivalence codes of the decrypted characters in a ROM so other software designed for romhacking can decrypt and thus show the text accordingly. For this task, a romhacker needs to search for a word or phrase appearing in the game, note down the
equivalence codes of the decrypted text (which are presented in hexadecimal, that is, a numeral system with a base of 16 that uses the symbols 0-9 to represent values zero to nine, and A through F to represent values ten to fifteen).

Figure 1. Searching for the word “Sword” in the Secret of Mana ROM using SearchrRX.

Once the text has been decrypted, the creation of a table file can be automated with software such as TaBuLar or TAG (Figure 2).
Figure 2. Automating the creation of a table file using TaBuLar.

**Graphic edition.** Normally, a non-localised ROM does not allow using accented letters, and Japanese ROMs may not even contain the Latin alphabet. Therefore, prior to the translation of a ROM, romhackers must edit its graphic data to insert all the necessary letters and symbols. In order to achieve this task, romhackers use software such as *Tile Layer Pro* or *Tile Molester* to edit unused characters in the target language (e.g., Japanese characters if the video game is in Japanese or rarely-used characters in Spanish such as the apostrophe if the video game is in English) and thus create a special character (Figure 3). This way, if a romhacker changes a
semicolon for an accented “a,” the video game will show an accented “a” instead of a semicolon when using the equivalence code for the semicolon.

Figure 3. Font of the Final Fantasy IV ROM (Super Nintendo version) in Tile Layer Pro.

Nevertheless, editing graphics in a ROM is not a process similar to retouching a normal image. The type of graphic data in a ROM depends on the console the video game was designed for, so romhackers first need to set up the software according to a specific console (e.g., Tile Layer Pro supports the Super Nintendo and Game Boy formats, among others) and then look for the ROM font with extra care, as graphics do not usually appear in order (Figure 3). If graphics are difficult to find, trying other formats may help.
Translation. Once romhackers have decrypted the text in a ROM, created a table file with most equivalence codes, and edited the graphic data to use special characters, they can start the translation phase. Romhackers use two main approaches to carry out this process: using special software called “hex editors” together with the table file and starting the translation; or locating all the portions of the script within the ROM, exporting them to normal text files, and then importing them once the translation is finished. While the first option is faster, the second gives more control over the text and favours the translation of a large script among different translators. The main hex editors used for romhacking projects are WindHex and Translhextion; both tools allow translating on the fly as well as extracting and inserting scripts to a text file.

Before starting to translate, romhackers need to complete the table file with some missing equivalence codes, such as dots, commas, line breaks and other special tags used to display text on screen. To do so, romhackers have to load a ROM together with its table file in a hex editor designed for romhacking and search for some text appearing in the game. Around the text will be non-recognised codes, so the romhackers need to deduce what the codes mean. The best way to finish the table is to search for text appearing at the beginning of the game and then compare the differences between the text in the hex editor and the on-screen text. In order to add new equivalence codes to the table file, romhackers need to open it with a text editor, such as Notepad, and follow the format of previous entries, which is normally “hexadecimal code” = “character”. Figure 4 shows an example of this process.
Figure 4. Completing a table file using on-screen text as reference.

When the table file is complete, a romhacker can then start translating. As using a hex editor to translate a ROM is not very comfortable, many romhackers opt to dump all the text into text files. In fact, this is what game developers do to localise their products: they prepare Excel files with all the translatable content (usually with many comments and macros to help translators) so it can be put back into the game easily once the translation is finished.

On the other hand, romhackers face a handicap when trying to achieve a good translation, as they cannot use more characters than in the original message because the text may not appear properly on screen, so concision sometimes plays a key role in the translation. However, advanced romhackers know how to locate and modify the so-called “pointers”, that is, internal codes in the ROM that specify when dialogues should be displayed. Thus, they can expand some messages and shorten others so the entire script fits into the ROM. Although software that makes this task easier is available, a romhacker must always know how to locate these pointers. Consequently, a team of expert romhackers and translators who can divide the tasks will be ideal to work on large projects and produce good quality translations.
Fortunately, as romhackers usually have programming skills, they often develop localisation kits so translators can see how the text will be displayed on screen, which can avoid many bugs after importing the translated text into the ROM, especially those related to layout and character limitations. For instance, *FF3 Editor* (Figure 5) is software that allows for editing almost everything in the *Final Fantasy III* (US version) ROM, including the dialogues and font.

![Figure 5. The dialogue editor of FF3 Editor.](image)

Finally, when the translation is finished, a long testing process begins to spot and fix bugs. Once the text is considered bug-free and accurate, romhackers create a “patch”, that is, a file that can be applied to the original ROM in order to create the localised version.

**Differences between Professional and Amateur Translations**

Theoretically, the quality of fan translations should be below par compared to professional translations. Indeed, anything from typos to mistranslations are often found, but as in the case of fansubs (Díaz Cintas & Muñoz Sánchez, 2006, p. 45), sometimes amateurs’ efforts to be perfectionists have yielded good results, *Tales of Phantasia* (translated from Japanese to English by the group DeJap) being a reference for high standards in the romhacking community.
The outstanding quality of the translation for this game is to be found — apart from using good translators — in the fact that romhackers are very familiar with the games they translate, so they have context during the translation of the script. Unfortunately, this is not the situation for most professional translators, as they do not usually have access to the game itself (Bernal Merino, 2007, p. 2). Currently, only large companies, such as Nintendo or Square Enix, appear to hire translators to work in-house so they can play the game and have control over the translation from the beginning to the end of the localisation process (Mangiron, 2006, p. 310).

Another characteristic of romhacking projects is that they do not need to comply with censorship rules that forbid foul language, sexual content or copyright issues with brand names, as translations are unofficial and therefore do not undergo a legal check. For instance, the DeJap romhackers’ translation of the Super Nintendo game Tales of Phantasia contains a scene in which the language is appropriate only for adults, while in the Game Boy Advanced version, which was localised by Namco in collaboration with Nintendo, the language is much more adequate for all audiences (Figure 6). Furthermore, some locations in the Super Nintendo version include swastikas, which do not appear in the Game Boy Advanced version (Figure 7).
Figure 6. Dialogue differences between the unofficial (left) and official (right) translations of Tales of Phantasia

Figure 7. Graphic differences between the unofficial (left) and official (right) translations of Tales of Phantasia

As the game was translated unofficially by DeJap romhackers before the release of the Game Boy Advanced version, many fans criticised the official translation because of the differences, since the transcriptions of some character names were different from what they were used to seeing (as shown in Figure 7, “Klarth” appears as “Claus” in the official translation) and because the original Japanese voices were dubbed in English. This is the reason why some video games that were translated officially have been retranslated by romhackers in the interest of satisfying the fans. As Mangiron (2006) stated:

Fan translations have on occasion received higher praise than the official ones, such as the English translation of the Japanese RPG Final Fantasy IV, because fan translators, as gamers, know what the game should be like in order to appeal to other gamers like themselves. (p. 315)

Finally, some romhackers also rewrite the script of a video game to make it funny and give players the opportunity to play the video game from a different perspective. For instance, the Spanish romhacking group Charnego Translations made a parody of the famous video game
The Legend of Zelda: A Link’s Awakening for the Game Boy Color. Apart from changing the graphics significantly from the original version (Figure 8), this group changed the entire story to make it hilarious, sexual and violent. This translation achieved so much popularity that the host of a Spanish well-known radio programme on video games called Game Over (2001) interviewed the authors themselves.

Figure 8. Comparison between the original (left) and parody (right) versions of The Legend of Zelda: A Link’s Awakening

The Legality of Romhacking

As commercial ROMs contain copyrighted material, it is actually illegal to manipulate a ROM in order to translate it, according to the Digital Millennium Copyright Act (1998). However, game publishers do not usually take legal actions against romhackers because they normally translate old video games, which does not impact sales since these games become unavailable. Furthermore, this is a way to promote old video games within one of today’s popular sagas, such as Final Fantasy.

Nevertheless, Pettus (1999, p. 88) described a case that dates back to 1999 in which the lawyers of a Japanese company called ASCII sent an e-mail to KanjiHack, a group of romhackers who were translating a computer game called RPG Maker 95, warning them to cancel the project
to avoid facing legal consequences. A more recent event took place in May of 2009; while it was not a translation project but a complete new game, the Japanese publisher *Square Enix* sent a cease-and-desist letter (Legal Department, 2009) to a group of fans who were creating a side story of the video game *Chrono Trigger* by using software used for romhacking projects. As they had not released their work to the public yet, they decided to cancel the project to comply with *Square Enix*’s demands.

Despite the fact that romhacking projects are carried out by and for fans with no profit motive in mind, they are considered illegal under current international laws, so romhackers must stop their work, if the copyright owners require this, to avoid facing legal action.

**Conclusion**

Video game localisation is different from any other type of translation due to the fact that end users are actually playing (that is, enjoying) when they read the translated texts. Therefore, as Mangiron and O’Hagan (2006) put it, “the main priority of game localisation is to preserve the ‘look and feel’ of the original” (p. 14). This is the reason why video game translation is often seen as “transcreation” (Mangiron & O’Hagan, p. 11), as translators opt for a domesticating approach in a Venutian sense (Venuti, 1995) so that players can amuse themselves without realising they are playing a translated game. Thus, translators must familiarise themselves with the video game subculture, and ideally, they should be gamers who know best the expectations of the target audience.

Thanks to romhacking, many gamers have had the opportunity to play video games which were never localised to their mother tongue. When this obscure process of video game localisation became easier with more user-friendly tools and new documentation, some gamers with good language skills even were able to translate by themselves the games they enjoyed in
their childhood. At no time did they think they were working for free so others could play in their mother language, as they were simply enjoying what they were doing.

Romhackers are rarely qualified translators, but they have the advantage of having a thorough knowledge of the game domain. Therefore, using the skills they develop, they find it easier to get involved with the video game industry. As Mangiron (2006) concluded, “Sometimes job specifications for localisation specialists do not require previous experience in this field, only some previous translation experience, experience as a gamer, and knowledge of the gaming world” (p. 315).

In this sense, participating in a romhacking project can be seen even as an educational aid for those seeking to work in the video game industry. In fact, such projects are a way for them to start familiarising themselves with common translation issues in video game localisation, such as the use of variables, text limitations within menus, dealing with special tags to insert pauses in the text, or other game specific issues. Further research is required to see if romhacking might help to fill the gap between academia and business, since to date, only a very limited number of universities offer video game localisation in their curricula.
References


Retrieved November 22, 2009, from


http://www.jostrans.org/issue06/art_bernal.pdf


http://www.archimuse.com/publishing/ichim04/3391_Esposito.pdf

Game over [Interview with two Spanish romhackers from the group Charnego Translations].


http://charnego.frikikipedia.es/downloads/mp3/entrevista1.mp3


http://www.merriam-webster.com/dictionary/hack


http://www.chronocompendium.com/CEOrder.PDF


http://www.overclocked.org/emufaq/EmuFAQ.pdf

Pablo Muñoz Sánchez


