ETSI TR 101 568 V1.1.1 (2012-02)

Human Factors (HF);
A study of user context dependent multilingual communications for interactive applications

Reference
DTR/HF-00137

Keywords
application, HF, ICT, interaction, language, user

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://ipr.etsi.org).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Human Factors (HF).

Introduction

Of the study objectives

The present study addresses the issue of localisation for multilingual context-dependent interactive applications. "*Localisation*" is the process of adapting the application for a specific country. It implies not only translation of dialogues or phrases from one language to another, but also the adaptation of idiomatic and cultural characteristics. The same issue is relevant to multilingual interactive applications where several languages are to be supported simultaneously.

The objective of the study is to define how to simplify the development process of highly interactive multilingual applications and to ensure the top quality of their localisation.

The present study is intended for anyone dealing with complex localisation of context-dependent interactive products, such as dynamic online systems, video games, serious game and eLearning, smartphone applications, internet-based applications accessed by PC or mobile, etc. It concerns applications designers, developers, publishers, product managers and distributors, as well as all stakeholders who may benefit from its use, including service and application providers, end-users, etc.

In short, the present study primarily concerns those who have already experienced a serious localisation problem, when designing or using an application, especially designers, publishers, users or all those who don't want to experience this problem at all.

Therefore, the present study tries to achieve two key goals:

- To describe the state of art in the localisation issues and techniques, especially regarding context dependency;
- To describe a way forward, a proposed roadmap leading to guidelines or potentially standard in that area.

Although this roadmap may need additional collateral information, the study shows that the contents, the scope and the potential solutions for such guidelines is clearly defined, so that the Technical Committee may launch these guidelines study with no delay, nor additional research required.

Of the study background

ICT users are becoming increasingly involved and fully immersed in applications such as video games or one-to-one Internet-based applications. The more immersed the user, the more successful the application! Two key factors determine the extent of such immersion: an increasingly realistic environment (such as graphics), and a more in-depth textual or oral interaction. Applications therefore demand "online" textual or oral interactivity with the user in a complex, accurate and natural-sounding way. Texts are created on the basis of the user context, which, in turn, depends directly on the user's actions and his/her environment.

The complexity of dialogues and interactions with the environment in different context has become so important that it is practically impossible to plan for every potential combination. Text "*strings*" (chains of characters) to be created by the application are therefore constructed dynamically from scripts for human-machine dialogues, through "engines" generating at real-time phrases that are dependent on these context variables.

Once created, these applications are to be adapted into languages or countries different than the original ones they have been created for- a process known as *"localisation*". It implies not only the linguistic translation of dialogues or phrases from one language to another, but also the adaptation of idiomatic and cultural characteristics. In simple applications, with little dynamically generated text, the localisation process includes the translation of the whole User Interface (UI) and text strings from the source language into the target language.

This is however not possible for interactive applications based on variables and interactivity scripts. Localising such an application implies translating all possible UI and text strings from one language into another, identifying all variables and their potential values, and also translating these variable values into the target language. The fact that majority of applications are being written in English or Asian languages, which have a very simple grammar system, increases the difficulties when translating into other, more complex-structured languages, where grammatical agreements vary depending on case, number and gender. The problem becomes even more critical when having to adapt cultural variables. These issues can lead to limiting the number of countries in which the application can be marketed. Alternatives are either forcing users to use English, or releasing poor quality applications in localised languages, risking a poor audience or worse, a negative buzz.

Several types of industries are facing this critical problem, such as the game industry, education, telecom, internet, automotive industry, etc. Many of them are working around the problem by simplifying the dialogues to avoid grammatical barriers, therefore reducing the quality and the level of immersion. And there are no emerging languages technologies able to propose a valuable solution yet.

Therefore, there is a strong need, both for the designers and for the end users, to study issues relevant to the localisation of such context dependent multilingual interactive applications, approached from all relevant stakeholders' perspectives, to understand the complexity and specificity of the issue throughout all the involved application segments, to analyse how these sectors are addressing or working around the problem, and finally how the whole application development community can define together a common way to solve this increasingly critical issue.

Of the study boundaries

The study will focus on ***text-based*** interactivity, since this is the core of all communications, even audio ones. Indeed, applications are either explicitly text based (messages are displayed to the user or taken from him through keyboard) or they add an audio interface, as input or output. Audio inputs are based on Speech To Text (STT) and Automatic Speech Recognition (ASR) technologies to be able to record and process user input. Audio outputs are either pre-recorded audio (then static and with no link with our scope) or based on Text-To-Speech (TTS) technologies able to generate speech out of a dynamic text. The present study will then not consider audio at all, and the speech technologies STT, TTS, ASR, although of high interest in multi-language systems, will not be presented in the report as being totally out of scope.

It is also important to explain what part of the concept of UGC – User Generated Content – the present study is covering. UGC is a generic term covering all types of information, used in a broad range of applications, which is coming directly from the user, such as news, forums, comments, blogging, digital video or images, podcasting, etc. In the present study, since target applications are context dependent interaction, UGC is restricted to the user context, including his profile, his inputs, his history and previous actions, etc., which could be recorded in real-time context variables, as in 1-to-1 marketing, or games or role playing in eLearning scenarios. Typically, user input such as comment, chat, discussion, is out of scope. However, user input asked for a name, an answer to a question, a decision, a choice, are to be considered. In a first phase, "closed" inputs only will be taken into account, and later more open and informal answers.

1 Scope

The present document gives an introduction to and an analysis of the most important issues and areas of relevance to context dependent multilingual communications for interactive applications. It provides a clear description of the most common difficulties and problems faced by application designers and localisers today, and how they solve or work around these.

The scope of the present document is summarised through the following statements about the study:

1) It defines localisation and explains what is involved in the localisation process of interactive application, including the management of interactive and non-interactive applications translations.

2) It describes in detail problems and issues related to the localisation of interactive applications, to help understand the limitations, needs and existing solutions or work-around used in the field.

3) It identifies the different industrial and technical domains that are directly concerned by the issue. It looks at several innovation activities related to the domain and provides a state-of-art presentation of languages technologies and research in the domain of multilingual applications, translation management and localisation.

4) It collects information from the different industry sectors identified, examining their needs, their localisation process, their management of translations, and potentially, specific tools or processes they are using for solving or working around the problem.

5) It provides a generic analysis of the situation, plus a specific analysis related to each identified industry sector.

The present document addresses the localisation process, which covers a large spectrum of issues and activities. However, the study, after providing an exhaustive description of what localisation means, will focus on the localisation aspects relevant to the highly interactive applications heavily using context variables.

The present document had an initial focus on the game industry but it does not restrict its scope to video games. On the contrary, it will expand its vision beyond games, aiming at identifying all other technical and economical sectors facing similar issues.

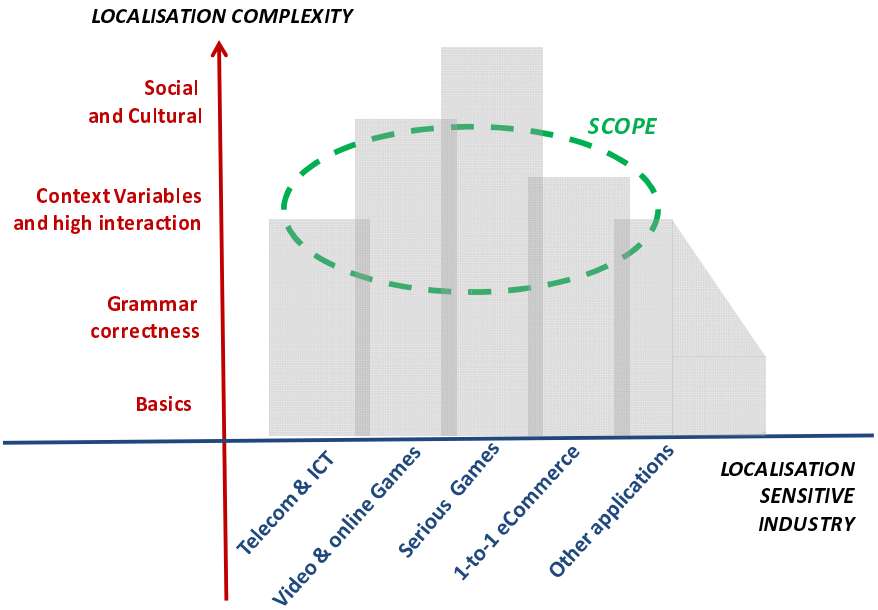

Figure 1: Technical Report Scope diagram

Finally, the present document will not provide an exhaustive analysis of advanced research projects and techniques, but it will review the main existing or known areas of innovation and analyse whether they may help solving our problem.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

Not applicable.

2.2 Informative references

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI EG 202 417: "Human Factors (HF); User education guidelines for mobile terminals and services".

[i.2] W.L. Johnson, S. Marsella, N. Mote, H. Vilhj´almsson, S. Narayanan, and S. Choi: "Tactical language training system: Supporting the rapid acquisition of foreign language and cultural skills". In Proceedings of InSTIL/ICALL2004 - NLP and Speech Technologies in Advanced Language Learning Systems, 2004.

[i.3] W.L. Johnson, S. Marsella, and H. Vilhj´almsson: "The DARWARS Tactical Language Training System". In Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), 2004.

[i.4] R. Klischewski: "No man's tool - Why we need games localization tools". LISA China Conference. Suzhou, China, 2010.

[i.5] US patent "System and method for generating grammatically correct text strings" - US patent office - number 7983895, July 19, 2011.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

avatar: See "playing character".

crowd sourcing: it is the act of outsourcing tasks, traditionally performed internally within a company or a group, to an undefined, large group of people or community

NOTE: It can be a community design (all or part of a development task), a community review and evaluation, or any other community activity.

context variables: See "variables".

dynamic dialogue (vs. static dialogue): user-application dialogue where interactivity and sentences depends on the user context, the user actions and his environment, therefore with an infinite number of possible phrases

end user: See "user".

function: abstract concept of a particular piece of functionality in a device or service

ICT devices and services: devices or services for processing information and/or supporting communication, which has an interface to communicate with a user

localisation: advanced process, which consists of adapting an application for a specific country, not only through the translation of dialogues or phrases from one language to another, but also through the adaptation of idiomatic and cultural characteristics

manual: See "user guide".

Machine Translation (MT): automatic translation, as by computer, from one natural language to another

NOTE: Initially restricted to word to word translation, current MT systems are using technologies such as rule-based translation, translation memories, dictionary based techniques or statistical techniques, as well as hybrid systems using several of these techniques.

modifier or **variable modifier:** modifier is tag to be put on a variable to indicate that the value of the variable needs to be modified to be in grammatical accordance with gender, numeral, case, etc.

NOTE: Modifiers could be use for social and cultural aspects also, as, for instance, for indicating formal versus informal addressing.

non-playing character (NPC): any fictional character of a game or an application not controlled by a player. In video games, the NPC will usually be controlled by the computer

NOTE: In other games, or in many applications (such as Serious Game), the NPC will be a character controlled by the game master, the educator or the Master of Ceremony.

1-to-1 marketing/eCommerce: personalized marketing/eCommerce as an alternative to mass marketing. The system analyses each client or prospect in order to adapt the communication and sales accordingly

player: denomination of any **user** in a game or a serious game

playing character: any fictional character of a game or an application directly controlled by a player

NOTE: It is also called the "avatar", as the game entity representing the user, its behaviour and acting on his behalf.

sim-ship: simultaneous shipment of a given application in several countries to avoid the need for any additional investment.

NOTE: Localisation should then be done for each country.

static dialogue (vs. dynamic dialogue): user-application dialogue where interactivity and sentences are well defined in a fixed tree of user actions or environment, therefore with limited number of possible phrases

Terminology Management System (TMS): translation memory tools (see TM below) use either segment based translation memory engine or a corpus based engine, or both.

NOTE: This is called a "terminology management system". This ensures consistency in the translation by automatically searching for previous examples of where phrases were translated in the document.

translation: process of taking textual or oral communication elements, in the form of sentences or phrases, from one source language and translating them into a target language

Translation Memories (TM): translation memory allows for re-use of what has been translated previously by a professional translator.

NOTE: There are TM by domains, industry, projects or groups of projects. The importance of terminology is huge

user: person who uses an application - see also "player"

user education: any information provided to users of a product or service on the functionality provided by the product or service and any instructions on how this functionality is to be used

User Generated Content (UGC): generic term covering all types of information, used in a broad range of applications, which is coming directly from the user, such as news, forums, comments, blogging, digital video or images, podcasting, etc.

NOTE: In the case of the present study, since target applications are context dependent interaction, UGC is restricted to the user context, including his profile, his inputs, his history and previous actions, etc., which could be recorded in real-time context variables, as in 1-to-1 marketing, or games or role playing in eLearning scenarios.

user guide: technical communication documents, intended to give assistance to users using a particular product

User Interface (UI): physical and logical interface through which a user communicates with a telecommunications terminal or via a terminal to a telecommunications service

NOTE: Also called man-machine interface, MMI.

variable: variable is an element of a sentence that can take a different value depending on the user context and the game context

NOTE: In highly interactive applications such as game, the possible values of a variable can be quite high, the combination of variables values being then infinite.

XLIFF: interchange format used in the localisation field, as an extension of XML

NOTE: The format is used widely but does not handle key elements such as variables table, needed for context-dependent applications.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

ASR	Automatic Speech Recognition
CAT	Computer Aided Translation tool
CM	Content Management
CMS	Content Management System
CRM	Customer Relation Management
GCMS	Global Content Management System
GMS	Globalization Management System
HTML	Hyper Text Mark-up Language
ICT	Information and Communication Technologies
MMORPG	Massive Multiuser On-line Role Playing Games
MT	Machine Translation
NPC	Non-Playing Character
PC	Personal Computer
PDF	Portable Document Format
R&D	Research and Development
STT	Speech to Text
TM	Translation Memory
TMS	Terminology Management Systems
TTS	Text to Speech
UGC	User Generated Content
UI	User Interface
XLIFF	XML Localisation Interchange File Format
XML	eXtensible Mark-up Language

4 Localising an interactive application

4.1 Localisation vs. translation

As defined, "*translation*" is the process of taking textual or oral communication elements, in the form of sentences or phrases, from one source language and translating them into a target language. Translation can be done from a well defined and fixed text (official translation of an official document or a book, for instance); translation can be also done simultaneously by human interpreters, while a person is talking in a conference. Translation of applications on the other hand consists in taking both fixed and dynamically generated texts displayed by the application from one language to another.

Existing and emerging languages technologies can help to translate applications using semi-automated systems such as Translation Memory or fully-automated systems such as Machine Translation (Goggle translation, internet automatic forum or consumers' comments translations, etc). These systems give variable results depending on the available corpus, the context, the content, the type of text and the format. For instance, Translation Memory, which has now reached maturity, is often integrated in GMS – Globalization Management Systems or GCMS – Global Content Management Systems. In the same way, Machine Translation is still limited but it gives great results when following a strict process (e.g. controlled English during authoring). This is the case of the automotive industry, for instance, which performs most of its translation using MT through a well-defined user interaction process.

As defined, "*localisation*" is a more advanced process, which consists of adapting an application for a specific country, not only through the translation of dialogues or phrases from one language to another, but also through the adaptation of idiomatic and cultural characteristics.

The document EG 202 417 [i.1] provides a good and clear definition of the localisation process and its implications for mobile terminals and services, which is fully applicable to the context dependent interactive applications:

"Localisation refers to the provision of product and user-guide variants for different markets taking into account local linguistic and cultural differences. This presents a special challenge as the ICT market is a global market and most manufacturers try to market their products globally. In many countries, the localisation of consumer products is required by regulation.

As the costs for localising products and services are considerable, most manufacturers and service providers restrict their localisation efforts to offering different language versions of the user interface (in particular in the menus) and of the user guides. The use of icon-based menus (currently state of the art at least on the main menu level) is an attempt to internationalise aspects of the user interface of many applications.

One of the main challenges related to localising applications is that as localisation is done fairly late in the development process, the localisation efforts of application comes after completion of master draft, and even sometimes after the first release in other countries, so always at a very late stage. Since all last-minute changes to the master also have to be made to all language variants, correct and complete language variants are only available in later editions".

The following clauses will explain the differences between translating and localising and therefore the technical and cultural aspects of localisation. Although the localisation of context dependent interactive applications may have some common aspects with other ICT sectors, such as mobile terminals and services, there provide a lot of highly specific aspects that make us understand why solving the problem is difficult and, as today, not solved at all.

4.2 Localisation aspects

4.2.1 Localisation and language complexity

The grammatical complexity of languages is endless, and the examples provided in this clause are only giving a quick flavour of problem. A more in-depth presentation of the most common complexity elements of different languages is provided in Annex A. Cultural and social aspects are adding to this "technical" complexity a human and emotional layer that is even different, within the same language, between cultures.

This aspect could have stay a minor and useless question in the way translation and localisation was several decades ago: why bother when your application is speaking English, German, French, Japanese and Chinese and when sentences to be translated are the text bars in menus or error messages? But, as explained in the introduction, three major phenomena are now forcing industries to address the problem of localisation fast:

- Applications should now offer a totally immersed interaction in an increasingly realistic environment. Users are not accepting to be forced in to a foreign language (English) or gender (male) any more.
- Globalization enforces distributors to provide a given application simultaneously in an average of 12 languages per release (typical industry standard), and to add additional language for each target markets.
- Minority policies and markets are pushing distributors to more and more provide applications in minority languages, either for social and economical reason (access to eSociety) or for cultural reasons (strong mobilisation, dense market, cultural heritage preservation).

As a result, localisation is becoming a critical aspect of commercial growth, which increases the pressure onto applications providers for porting applications onto a broad range of languages, whatever their complexity and cultural specificities. This is why understanding these complexities and solving the related localisation problem are the 2 steps that are described in the present study.

Here are several basic elements showing the complexity of languages and why localisation can become uneasy.

4.2.2 Grammatical aspects

- The basic grammatical specificities in languages such as gender, number, cases, plurals: e.g. French has four types of definite article for the English "the" ("le", "la", "l'" or "les"), and adjectives agree with gender and number; Russian has two plural forms Finnish has 14 different groups into which you can sort almost every noun and adjective, etc. So a simple sentence in English for instance, may have many possible translation depending on the context:

"$Player$ picks up the %color %item" → ***Alan ramasse le couteau bleu***
→ ***Alan ramasse la perle noire***
→ ***Alan ramasse l'épée bleue***
→ ***Alan ramasse les pièces rouges***
→ ***Alan ramasse la blanche colombe….***

Some languages even have different plural depending on the actual numerals. Here is an example from English to Polish localisation of a very simple phrase:

%numeral red flag(s)" → 1 red flag or x red flags

English	Polish (female gender)	Polish (male gender)	Polish (neutral gender)
%d red flag	%d czerwonej flagi	Czerwonego fotela	Czerwonego jeziora
1 red flag	1 czerwona flaga	1 czerwony fotel	1 czerwone jezioro
2, 3, 4 red flags	2, 3, 4 czerwone flagi	2, 3, 4 czerwone fotele	2, 3, 4 czerwone jeziora
5-20 red flags	5-20 czerwonych flag	5-20 czerwonych foteli	5-20 czerwonych jezior

Figure 2: Example of Polish localisation of numerals

- Some languages are handling plural accordings, in very specific ways. Some put just an S at the end (most EU languages) or at least another variant of the noun, while some languages just repeat the noun twice.

  e.g.: ***the man / l'homme (sing.) → the men / les hommes (pl) in English and French***

  orang - a person" (sing) → orang-orang – the people " in Indonesian

- Some languages differentiate between the collective, which is indifferent in respect to number, and a set of single entities (called "singulatives"). For example, in Welsh, ***moch*** ("pigs" as a whole) is a basic form, whereas a suffix is added to form ***mochyn*** ("pig" as a single one).

- In other languages, "singulatives" can be regularly formed from collective nouns:

  e.g.: Standard Arabic: حجر ***ḥajar "stone"*** → حجرة ***ḥajara "(individual) stone",***

  بقر ***baqar "cattle"*** → بقرة ***baqara "(single) cow".***

- Numerals may be handled not just as singular or plural in some languages. For instance, in Russian, there are 3 different types of according for numeral: Nominative singular when 1 unique object, genitive singular for 2, 3 or 4 of the object, and genitive plural for any number above 4.

- Possessive pronouns in German agree to both the gender of the owner and of the object, while in French accordance is only to the gender of the object:

  e.g.: Son frère → you don't know whether the 'owner' is male or female

- The dialect variants of particular languages: e.g. Dutch in the Netherlands and in Belgium (Flemish), German in Switzerland, Austria and in Germany; French in French Canadian and Creole, etc. This includes spelling-only variants, such as the OUR vs. OR between British and American English (e.g. ***colour*** vs. ***color***).

- Translation may vary depending on the context. Some words have several meanings, not the same from a language to another, so that localisation then takes context and situation into account.

  e.g.: (English) "I've got the POWER" → (French) "J'ai le POUVOIR" ou "J'ai de la PUISSANCE"

- Additional idiomatic words can be added to a sentence, which have no translation, but add an inflection, information the speaker want to give: an advice, an order, a question, a doubt, etc.

  (French) Vous devriez ouvrir la porte de gauche.

  (English) You should open the door on the left first, shouldn't you?

- Some languages, such as German for instance, are creating very long words and sentences for expressing a specific action or situation. A simple phrase in English or Chinese, with a certain length, will be translated into a German phrase more than twice as long. It is not uncommon for short texts, such as the titles of text commands and menu items, to be three times as long in German as they are in English, while the Chinese equivalent will be much shorter. For example, the English word "***Redo***" translated to German is "***Wiederherstellen***" up from 4 characters to 16, i.e. representing an expansion of 400 % !

- That might be (actually is) a problem when the application has been designed with strings with a specific maximum number of characters. Localisation should then adapt the sentence to fit in the maximum string length, often by using abbreviations, hoping that the user will understand them:

Figure 3: Example of text length limitation in German

At some point, even for a German speaker, the abbreviations may be so numerous that it is impossible to understand what they are finally referring to.

4.2.3 Social aspects

- The use of *formal addressing*: in some cultures, it is appropriate to address the user using formal language ("Vous", "Sie", "U", etc.), while in others an informal addressing ("Tu", "Du", "Jij", etc.) may be expected (e.g. French, Spanish or Italian) while other languages would expect the informal form (e.g. German or Swedish). And, in some cases, dialogues may expect an informal addressing with a formal response.

- Beyond the formal addressing, the functionality of the application itself can be based on hierarchy relations, such as chief to soldier in a game, or the manager to his employees in an eLearning management training. This hierarchy is expressed not only through *formal addressing*, but also through specific cultural coding: e.g. the same order or advice will be given in a very different tone and words in German (straight, strict) or in Japan (indirect, soft).

- Interactivity takes into account the social behaviour and social relationship between playing or non-playing characters. Localisation should then use the profile of the users/actors of the dialogues (static information) as well as the context, the situation, the intention (dynamic information). This will lead to vary between formal vs. polite, familiar vs. casual, friendly vs. unfriendly, informative vs. directive, handled different from one language to another, etc.

- Some languages use a different word depending on who is speaking, and its relation with who he speaks to: "I show you" has four possible translations depending on gender of the speaker and the listener (*i.e. male to male, male to female, female to male and female to female*). Other languages (Thai or Cambodian for instance) use different words depending on the hierarchy between both: therefore knowing that a male is talking to a female is not sufficient: a father talking to his daughter is different than a son talking to his mother, a master to his female servant, etc.

4.2.4 Cultural aspects

- Humour expressed in words (or in illustrations) is highly cultural, since some topics may be offensive in some cultures: e.g. comparing a man with a dog is the highest offense in Arabic countries. Some applications can just avoid using it or at least employing it with care: Mobile terminals and services (as mentioned in their recommendations), or purely informative and non-immersive applications such as for instance GPS automotive systems. However, some applications do use it intensively for the core of their efficiency, such as game, education applications, storytelling, etc.

- The same issue is to be taken into account concerning relations between man and woman, and the way sexism is perceived. E.g. German requires the use of male and female terms describing a person in order to circumvent sexism, while some Anglo-Saxon cultures consider this sexist and require both males and females to be addressed with the male term only.

- The usage of foreign terms, such as English or American terms, might be accepted in some languages, but not in others. Same with Latin expression used in Latin and Saxon languages but not the others, etc.

- Visual elements such as colours create a different stimulus depending on the culture. For instance, the ***green*** or ***red*** have a different positive/negative comprehension. Here is the example of way the Yahoo finance website colour market symbols:

Yahoo finance website	Market up	Market down
China - http://cn.finance.yahoo.com/	Red ▲	Green ▼
Korea - http://kr.finance.yahoo.com/	Red ▲	Blue ▼
France - http://fr.finance.yahoo.com/	Green ▲	Red ▼

Figure 4: Example of cultural differences in colour symbols

Colours may also be associated to specific events or feelings:

Color	China – Asia	USA - Europe
Red	Celebration, Luck, Marriage	Stop, Danger
White	Mourning, Death	Purity, Marriage
Green	Infidelity	Go, Safety, Environment
Blue	Immortality	
Yellow	Sacred, Imperial	Caution, Joy, Happiness
Black		Mourning, Death

Figure 5: Cultural differences in colour meanings

- Finally, each culture has taboos or fears, which should be taken into account not to shock the user with something that seems meaningless to foreigners. Several examples: In China, number 4 is considered as very bad luck, and number 8 is very good luck, while, in Occident, numbers 666 or 13 are to be avoided. In the US, 9/11 now refers to more than just a day of the year. Another example: criticizing royalty is taboo in Morocco, Spain, Belgium or Thailand, and so it is for religion in many countries, such as Islam in Middle-East.

4.3 Localisation of context dependent applications: games

4.3.1 Flavour of the game localisation problem

Most applications now require real-time textual or oral interactivity with the user in a complex, accurate and natural-sounding language. This is clearly demonstrated with video games and in particular to MMORPGs (Massive Multiplayer's Online Role Playing Game), in which players of all nationalities become fully immersed in a virtual reality. The greater the level of immersion in the game, the more successful it is. There are several key factors which determine how effective immersion is, including increasingly realistic graphics and real-time textual or oral interaction in the chosen language that comes close to normal human exchanges, even if this actually involves a human-machine dialogue.

This poses a major technical problem in cases where applications need to construct generic phrases with an infinite number of possible context-dependent combinations.

Figure 6: Simple interactivity script

Simplified examples

"you have received [numeral] %item(s)"	***"$avatar$ picks up the %item"***
enables generation of	enables generation of
"you have received 1 gold medal" *or* ***"you have received 2 gold medals"***	***"Queenie picks up the key"*** *or* ***"Mousy picks up the knife"***

Texts produced by the application are composed on the basis of the application environment (situation, level, plot, etc.) and most of all, of the ***user environment***, which, in turn, depends directly on the user's actions. This environment is represented by a certain number of **context variables**.

The key point in today's game is that ***it is physically impossible to plan for every potential combination of these variables*** and, therefore, every potential sentence to be pronounced or displayed in the game.

The UI or text strings character chains that need to be created within a game are built dynamically by the application such as the example above. There are described in pseudo-text files called the "*interactivity scripts*". These interactivity scripts are processed by the core of the game code, called the "*engine*", which generates phrases that are dependent on the context variables in real-time, as shown in Figure 7.

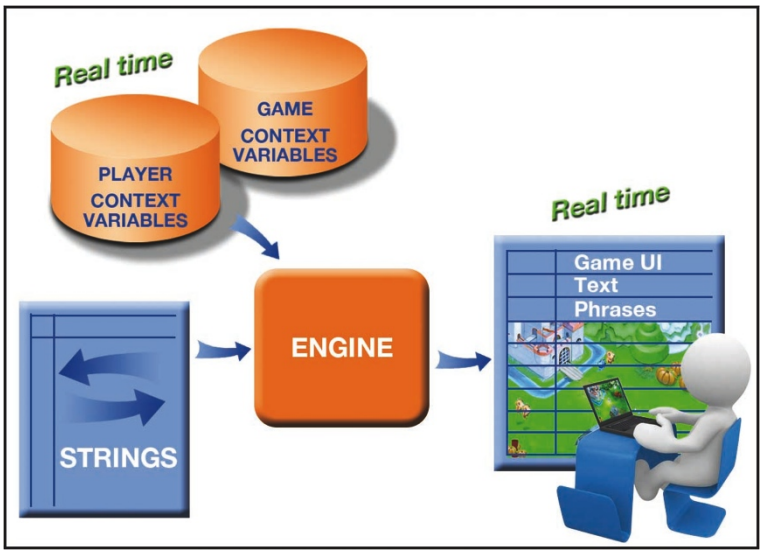

Figure 7: Simplified Game Structure

While the game is running, the game gets the structure of the phrase to be generated real-time, for instance, for

"$avatar$ picks-up the %item"

it gets the values of the context variables,

"$avatar$" = ***"Queenie"*** and ***"%item"*** = ***"key"***

and finally it builds the phrase to be displayed or said:

"Queenie picks-up the key".

Example of a localisation problem

Using the same example as above, a simple localisation from English to French, for instance, would consist of translating interactivity scripts and potential values of variables into French:

"$avatar$ picks-up the %item" → ***"$avatar$ ramasse le %item"***

"$Player$" can keep any value, not dependent on language
"%item" values: "key" → ***"clé"; "knife"*** → ***coutea, etc.***

But when phrases are generated in real-time by the game, linguistic issues occur:

"John ramasse le couteau" → correct
"Alan ramasse le clé" → **INCORRECT !** *(in French, « clé » is female, the article should then be « la »)*

Interactivity scripts as well as the game processing the information in real-time should be more sophisticated to handle localisation and multilingual versions of a game. This is the heart of the study.

Example of a work-around

The output in another language is sometimes so random that you can end up changing the initial phrase itself in a way that soon means it becomes simplistic, just to avoid any accordance errors. For example:

"$avatar$ picks-up the %item"
may become
"$avatar$ ramasse: %item"

If no effort is made to correct this, the result is poor or low quality, and the game loses most of its immersion capabilities.

Dynamic dialogue vs static dialogue

Applications (not just games) are composed of *Playing* and *Non-Playing Characters*. Playing Characters represent actual users of the application. Non-Playing Characters are additional characters, either machine-generated (in games, for instance), or moderators, referees played by human users to monitor, control or orient the course of the application (in eLearning and Serious Games, for instance).

Experience shows that the quality of the NPC communication is a major factor of the immersion and efficiency of the applications. NPCs express what they are thinking or doing to appear more immersive. The goals of an expressive NPC can be achieved with *static dialogue*, but adding grammatically correct *dynamic dialogue* enables more detailed, precise, relevant and interesting expression. This in turn allows players to infer more to an NPC's character behaviour and thoughts:

- Context-sensitive conversation
  - Expressing needs and emotional state

    Static: ***I'll have a drink, please.***
    Dynamic: ***I'll have a beer, please.*** (Ordering at establishment)

    Static: ***He's got a weapon!***
    Dynamic: ***He's got a knife!*** (Expressing fear/alarm; transmitting information)

  - Expressing knowledge

Static: ***I heard someone was causing trouble around here recently***.
Dynamic: ***I heard there was a fight in London last night.*** (Emergent current events)

- Location-sensitive conversation

Static: ***In pursuit of suspect. Requesting backup.***
Dynamic: ***Suspect entering the Sophia office. Requesting backup***. (Recognizing location)

Static: ***I'm heading there now. Meet me there.***
Dynamic: ***I'm heading to room 9. Please, meet me there.*** (Talking on phone)

4.3.2 Variable and modifier: Example of interactivity script

Creation of the sentence in English

The designer of a new game is writing interactivity scripts in English. He wants to communicate a range of context-dependent sentences to the player:

"Take the red key !"
"Take the green rope !"
"Take the yellow coins !"

He then creates 1 variable ***(%item***) that depends on the user and game context.

"Take the %item.color %item !"

But the game should be localised in several other countries, such as France and Germany. The script above is not sufficient since in French, the defined article "***the***" may take 4 different forms: ***le*** (masculine), ***la*** (feminine), ***l'*** (both when next word begins with a vowel) and ***les*** (plural). The designer should then foresee the situation and add a function "***get_defined_article***" which will return the proper article upon the gender and numeral of the following variable %item, for each language it is designed. The phrase in the English script is then as follows:

"Take <get_defined_article> %item.color %item !"

In English, the function ***<get_defined_article>*** returns the unique word ***"the"***

Each item record contains also the associated attribute "color" of the item, context dependent too and also taken from a set of potential colors, usually common to all the game elements requiring a color. The interactivity script then contains the following phrase that will be used by the game engine, with the related variable table and color attribute:

"Take <get_defined_article> %item.color %item !"

Variables	Values
%item	
	key
	rope
	coin
	knife
	stone

Attributes	Values
.color	
	black
	white
	red
	blue
	yellow
	green

Figure 8: Related variable table and colour attribute

Localisation for French language

The work of the localizer is then to translate the script phrase plus the variable tables. A first adaptation will consist in changing the order of the words, since "color" should be after the noun in French. A second decision is also to always stay formal ("vous" vs. "tu") with the player, the imperative mode being different for both, to avoid the grammatical difficulty. The French script phrase and the related variable tables appear as follows:

"Prenez <get_defined_article> %item %item.color !"

Variables	EN values	FR values	Gender	Defined article
%item				
	key	clé	F	la
	rope	corde	F	la
	coin	pièce	F	la
	knife	couteau	M	le
	stone	caillou	M	le

Attributes	EN values	FR values
.color		
	black	noir
	white	blanc
	red	rouge
	blue	bleu
	yellow	jaune
	green	vert

Figure 9: Related localised variable table and colour attribute

But this is not sufficient since, in French, adjectives should be in accordance, gender and numeral, with the noun. Tables will then be enhanced to give all the possible gender/numeral cases depending on the %item. A specific tag, that we call a "modifier", tells the game engine that the value of %item is to be modified depending on gender and numeral. The script in French will then look like this:

"Prenez <get_defined_article> %item <check_gender> <check_number> %item.color !"

with

<get_defined_article> returns the right article for the following word
%item realtime value of the variable %item
<check_gender> word modifier making sure the gender will be respected (here for color)
<check_number> word modifier making sure the plural will be respected (here for color)
%item.color realtime value of the color attribute for the variable %item

So that correct French sentences will be generated:

"Prenez la clé rouge !"
"Prenez la corde verte !"
"Prenez les pièces jaunes !"

The related variable tables will look as follows:

Variables	EN values	FR values	Gender	Numeral	Defined article
%item					
	key	clé	F	S	la
	rope	corde	F	S	la
	coins	pièces	F	P	les
	knife	couteau	M	S	le
	stone	caillou	M	S	le

Attributes	EN values	FR values			
.color		M / S	F / S	M / P	F / P
	black	noir	noire	noirs	noires
	white	blanc	blanche	blancs	blanches
	red	rouge	rouge	rouges	rouges
	blue	bleu	bleue	bleus	bleues
	yellow	jaune	jaune	jaunes	jaunes
	green	vert	verte	verts	vertes
	orange	orange	orange	oranges	oranges
	violet	violet	violette	violets	violettes
	brown	marron	marron	marrons	marrons
	grey	gris	grise	gris	grises

Figure 10: Related localised variable table and colour attribute

Localisation for German language

Requirements for German language are also different. German language, like many other ones, respects ***case***. The word "***take***" becomes "***nimm***" + ***accusative***. The localiser should then use a new "modifier" stating that accusative case is needed, forcing the game engine to take the accusative version of the article. Gender and numeral are also to be checked. Therefore, the script phrase in German will look like as follows:

"Nimm <use_accusative_case> <check_gender> <check_number> <get_defined_article> %item.color %item !"

Localisation script principles

This simple example is showing how a simple text can be more complex upon target language characteristics. We could take the same example and add more and more complexity, adding not only grammatical aspects and rules, but also social and cultural ones. But the basic principles stay the same.

The fundamental elements to be used in interactivity scripts are then the following:

- "**variable**" or "**context variable**"
  A variable is an element of a sentence that can take a different value depending on the user context and the game context. In highly interactive applications such as game, the possible values of a variable can be quite high, the combination of variables values being then infinite.

- "**modifier**" or "**variable modifier**"
  A modifier is tag to be put on a variable to indicate that the value of the variable is to be modified to be in grammatical accordance with gender, numeral, case…etc: <check_gender> or <use_accusative_case> for instance. Modifiers could be use for social and cultural aspects also, as, for instance, for indicating formal versus informal addressing.

- "**attribute**" or "**variable attribute**"
  An attribute is an additional information attached to a variable, a field in its descriptive record, such as its color for instance, or a date, or a nickname. Some attributes may be associated with a finite list of values which will be the same for all variables using this attribute. For instance, a game defines 16 different colors, and each element with a color attribute can be set at these values.

4.4 Localisation of all context dependent applications

A focus has been given on the specific case of game applications for explaining the rationale and the issues of localisation of context-dependent interactive applications. Many other applications are facing the same problem, but games are the most complex applications in terms of context dependency and infinite possibility of variable-based generated interactivity. Understanding then solving the problem of the game industry will allow addressing and solving the localisation problems of many other sectors and their context dependent interactive applications.

Figure 11: Localisation concerns all sectors

4.4.1 Sectors sensitive to context-dependent applications localisation

Several industrial sectors are developing highly interactive applications handling user context dependency using variables, the resulting generated phrases and interactivity dialogue being nearly infinite. Therefore, localisation cannot be handled as for traditional less interactive applications for which all possible generated sentences are translated. As today, either the localisation process of such applications is consuming a lot of time and efforts, or the result is of poor quality, then threatening the application success in the target countries.

The main industrial sectors concerned today are the following ones – non-exhaustive list:

- **eLearning & Serious Games:** Applications in the eLearning and education domain are increasingly investigating in fully immersive environment, both hardware (Virtual Reality graphics, immersive simulators and caves, etc.) and software (immersive language and behaviour, real life interactivity, infinite possibilities, etc.). It uses more and more techniques from the gaming world, methods and structures, so that eLearning has developed a new branch called "Serious games". This sector is then highly sensitive to the localisation problem of highly interactive applications with context variables dependency. Its future will even go much further than games, since Serious Games are highly sensitive not only to grammar aspects, but a lot to social and cultural aspects.

- **Telecom & smart phone:** This clause is rather a supporting platform than a type of application, and it naturally overlaps with the 1-to-1 marketing and business as well as the game sector. But it is important to mention as a whole sector, because these industries are the one pushing their application providers to resolve the localisation issues. Most large enterprises (names available upon request) have localisation manager at the executive level. Moreover, telecom companies are providing also phone based applications, including pure telecom services, which interactivity is more and more context dependent. These applications may have a stronger need for voice controlled and speech processing than games, or eCommerce.

- **1-to-1 marketing and eCommerce:** The trend in eBusiness, paid or (apparently) "free", is to provide applications built on a 1-to-1 relation, where the application interacts with the client or prospect in a unique personalized way, based on the user profile, history and behaviour. The realness of the interactivity is the unique way to ensure efficiency of the selling process. Applications are then highly user context dependent, they need the same variable structure as games, and since there is no commerce without being global, localisation is critical for them. This sector is then highly sensitive to localisation, and demands a standardized way to solve the problem. Their demand is also pushing middleware and tools providers (engines, machine translation, terminology management system, globalization management system, computer aided translation, etc.) to provide the same standardized solution, since 1-to-1 marketing companies are integrating solutions, and have neither competencies nor research money to invest there.

Several eBusiness sectors are especially concerned (not exhaustive):

- Real estate
- Hotels and other related activities
- Travel and related activities
- Match making
- Technical information on-line (massive data)
- Etc.

Please note an important limitation in the localisation issues of these sectors regarding the User Generated Content (see "Of the study boundaries" in Introduction). The user content corresponds to the user profile, history, decision, answers, etc., but not his comments in related forum, chats or consumers' comments with free text. For instance, in the case of hotels reservations, the part of the application concerned will be the reservation, the promotion, the answer of the system to the request of the user. But it does not cover open text commenting a given hotel, or contributions in an open forum, which will have no impact on the application itself and will not be taken into account for one-to-one interactivity.

- **Automotive industry:** The sector is in increasing need of interactivity between cars and drivers, as well as between infrastructure and drivers. The 2 main directions are the GPS and other on-board computerized systems, which interface should be more and more immersive, and all security applications (especially infrastructure to drivers for security information). As for the telecom sector, these applications are strongly linked to speech processing, both speech recognition (ASR and STT) and speech synthesis (TTS). As explained in the clause 1"***scope***", speech is out of scope since all the context dependent localisation issue can be solved with text-only approach, but this aspect is to be analysed since in this case, user input in speech form is a key component for security.

4.4.2 Example of interactivity script: 1-to-1 real-estate sector

This is a real life example, which has been proposed to a leading localisation services company by a real estate on line client. The real estate on-line global application is offering property description in a formatted way, but it intends to use language as close to reality as possible and with no language mistake. The format should be as follows:

{new-build} {country} {property} {for sale}

The {new-build}	key could contain new, cheap, best, latest, popular, discount
The {country}	key could contain region, province, coast, town and any country
The {property}	key could contain apartment, property, villa, town house, country house, land,cave house garage, commercial, wooden home
The {for sale}	key could contain for sale, to let, holiday rental

such as:

new Spain property for sale

cheap Malta apartment to let...

The application should be able to generate every possible combination of these variables for translation. The localisation company has been asked to localise the application as is for several European countries.

Linguistic analysis:

There are a number of issues that need to be carefully managed for localising these variables:

First, the possible values of some placeholders are of different types (noun and adjective). So some combination of values will not work, even in English:

e.g. "*Discount coast*" or "*Latest region*" will be in the best case ambiguous or grammatically wrong.

In French, depending on the replacement values, the engine might invert the order of the placeholders:

{new-build} {country} {property} {for sale}

will work for:

populaire région - appartement à louer

But not for:

bon marché région - appartement à louer

In French, and many other languages, adjectives should be in accordance with noun. The script needs to take into account the gender of the replacement value for the placeholder {country} and pick-up the right form of replacement value for the placeholder {new-build}.

e.g. *"Meilleure région" or "meilleur quartier"*.

This analysis is not exhaustive and we could add a lot of issues and problems the application will face in French, in German or in Russian – at least.

Conclusion:

A script has been built to handle gender issues that may occur from the concatenation of variables {new-build} &{country}.

Check gender of value {country} and use correct gender for value {new-build}

An adaptation has been decided to limit the translation problems and helped to generate grammatically correct sentences in most languages: using a hyphen to separate *"{new-build} {country}"* from *"{property} {for sale}"*

This solution is not even guaranteed for 100 % of the possibilities, and some results are still in "not so good" French or German, but stay understandable, which is, for that type of application, acceptable. But the best solution would have been to create upstream a much better format, adapted to most target languages, so that the localisation process would have been limited to translating simple words in a variables table.

4.5 Impact on interactive applications

4.5.1 Localisation process

In most cases, the "developer/designer" creates the English (or Asian) version of the game using an English-oriented "engine" and the "publisher" is responsible for funding the "localisation " process (i.e. translation into the language of the target country). This renders the entire production chain particularly complex and leads to a significant increase in costs:

- **The developer:** designs the engine in English, invents his/her own coding rules (as shown in the example above), with varying levels of effectiveness, and corrects any language or comprehension problems at a very early stage, but only following a user request (bug).
- **The publisher:** is required to invest in the localisation of applications from English, normally achieving average results for simple languages and often unsatisfactory ones for more complex languages: the phrases generated are either grammatically poor and fail to sound natural or, worse, they are completely incorrect and incomprehensible.
- **The localiser:** has an extremely complex task, since he/she is required to translate each phrase in the code as accurately as possible, based on scripts with variables that hinder the translation process (reduced quality/productivity and complex validation process) and often limit its possibilities, where there are difficulties with other languages, such as plurals, genders or cases.
- **The player:** in most cases, is required to interact in English or to tolerate sentences such as "***you need [bullet] X 3 to do it***" or "***Alan picks-up: knife colour: blue***".

- **Translation tools:** the nature of the process, and particularly the presence of variables, limits the effectiveness of all translation productivity tools, such as translation memories, machine translation, Computer Aided Translation tools, and application engine/middleware providers.

Therefore, the only way for achieving a correct localisation implies to plan the localisation phase upstream in the design phase and to use an encoded way of model phrases whatever the language. This is highly complex, very costly, quite often inefficient, and never standardized.

All context-dependent interactive application industries stakeholders (see NOTE) beyond the sector of games and eLearning are seeking recommendations and possibly a standardized way to handle the problem, if necessary at the cost of some grammatical or contextual limitations.
This is the heart of the present study.

NOTE: Application designers and developers, publishers, distributors, engine and middleware providers, localisers, translators, CAT tools providers, Machine Translation providers, Translation Memory builders, end-users, etc.

4.5.2 Localisation impact on the economical world

The success of such an initiative will provide important economic, technical and scientific benefits to all stakeholders:

- It has an impact on **the technical and economic aspects** of these applications, enabling them to be developed by introducing the idea of context variable-based localisation and multilingualism at a very early stage in the development process, and by simplifying life for developers (by making it easier for the community to adopt).

- It allows **simplification of the techniques of the development environment**, and it **saves a huge amount of time** (the timescale for marketing the product is vital for this industry) **and money** (no more compromises between the time needed for translation and quality).

- It allows the **deployment of an application that is 100 % right from the design stage**, and which therefore has the advantage of immediate acceptance by users and does not run the risk of suffering any negative "buzz" either. In this way, it **increases the value of the application for the developer,** and it **enlarges the publisher's potential market** by a simultaneous shipment ("sim-ship") in several countries, without the need for any additional investment.

- It provides solutions that are elegant and **effective, enabling a mixture of human and language technologies** with the aim of achieving quality for customers and users alike.

- It would allow the applications community to be involved in the localisation process and put in place **crowdsourcing** strategies therefore reducing the cost and increasing the market acceptance of the application.

In conclusion, the constraints of today's globalization of the economy as well as of the society have made linguistic questions an ever-greater challenge. If companies wish to retain their competitive edge, they should localise their products to suit the needs of the local market in terms of language, culture and functionality. This initiative has been launched from the Game Industry but its impact goes far beyond, and addresses all context dependent multilingual communications for interactive applications.

4.5.3 Localisation impact on the society

Beyond the obvious economical impact, the success of this initiative will provide a powerful ***inclusion*** tool within our society:

- It would allow localise applications and services in ***minority or other non-supported languages*** at no or low cost. The EU has a proactive policy towards the protection and the promotion of regional and minority languages (Article 22 of the European Charter of Fundamental Rights) aiming at "respecting cultural, religious and linguistic diversity" (24 official languages, plus all the language regional variants). This initiative would strongly support this policy and provide a powerful inclusion tool.

- The support of Minority Languages would also be backed up by the economical and industrial sector, since these languages are always related to a very active and strong community, highly motivated, to that would largely adopt (and then buy) products and applications in their language. For instance, books, films or applications in Corsican have market segments more than 10 times larger than in French, in Corsica.

- It also foster the EU multilingualism policy, as well as the "early access to languages" policy, addressing not only the EU citizens and the foreigners trend to master 2 languages or more, but also the illiterate and poorly-literate persons, either for social or learning capacity reasons.

5 Localisation sensitive sectors

As explained in the introduction, the purpose of the present study is to ***set the scene*** of the problem, by explaining what type of applications are facing it and why providing a standardized solutions is demanded by all the localisation sensitive industries. The purpose is NOT to provide all the answers, which will be the objective of future work (through a dedicated work item and STF).

This is why the present study, based on the discussions with many localisation departments of different industries, detects which kind of problems they are struggling with and what solutions they would like to see developed.

The initial architecture of the study was planning to review the localization process of each industry sector. Interviews quickly showed that there was no point in making differences between industry sectors, because the actual problems were fairly generic so that the description of the issue coming from the game industry was general enough to illustrate the relevance of the context dependent localization issues for all other industries.

5.1 Context dependent application industry

5.1.1 Differences in industry size not sector

Innovative solutions for the localisation of context dependent application are today quite limited since either developers and publishers are working on internal proprietary solutions and procedures of script encoding, or their handle localisation by working around the problems.

For the middle-size companies, the choice is clear: either a poor quality of dialogues and interactivity, or a huge amount of valuable time and budget allocate to localisation of the application in post-production. Thankfully, they are some great well-localised games in the field, but only thanks to huge efforts.

Large companies (names available upon request) can afford post-production, and some of them (see below) have actually developed large and powerful systems solving internally some of the localisation issues described in the present document, but it is done at the cost of a large internal development support group that only large companies can afford. There are very few available environments today for developers and localisers.

Despite the relatively successful result, it is achieved at the cost of huge resources invested in a peripheral activity, so that interviews show that even them would like to have access to a common approach and shared tools (tools, not contents – see analysis) that would reduce the amount of non productive efforts in maintaining the localisation systems.

Localisation software providers have been also working on technologies improving translation and localisation. None of them are handling variables and therefore none is addressing the critical issue of context dependent interactivity applications. This has been confirmed by several localisation experts and tool and middleware providers.

5.1.2 Technical limitations of existing solutions

In variable-based application localisation, as explained above, there are very few advanced techniques. The promising techniques are actually coming from other applications using innovative components such as machine translation (not used at all for games, see analysis below), translation memories and so forth.

Therefore, the limitations of the current solutions are actually explained in details above in the general presentation of localisation problem for context dependent interactive applications. The present clause takes the opportunity to give more practical examples of the current limitations and results:

Practical example of interactivity limitation due to localisation problems

Let's take a real example from a large global company providing one of the most famous MMORPG games:

- In Russian, spelling of location names in the prepositional case depends on gender:
  - London (Masculine): Лондон à Лондоне
  - Siberia (Feminine): Сибирь à Сибири
- For the current version of the game, translation of location names is not feasible.
- Work-around should be taken to avoid using places in the prepositional case:
  - Option 1 (Bad) "You cannot do that while your ship is located in [place name]."
  - Option 2 (Better) "[place name] is not a valid location for that action."
  - Option 3 (Generic) "You cannot do that at your present location."

So at best, translators can work around the problem (and not always) but at the cost of an internal modification of the game (back to development) or at the price of a lower quality and precision of dialogues and interactivity.

Another recent example from a top European game developer

While localizing an existing Chinese game for Germany, the company has just realized that the code does not allow handling gender in their variables unless changing the whole structure of the application. As a result, the game cannot handle numerical variables and cannot have male/female differentiation.

Therefore, the company has officially decided to address all German players as if they were ***male***, and to handle the numerical with the sign X as in "X times":

*You need **$itemNum $itemName** to do it*
*He/She needs **$itemNum $itemName** to do it*

will appear in German (here the English translation)

***You** need **$itemName X $itemNum** to do it*
***He** needs **$itemName X $itemNum** to do it*

For example:

*He needs **Coin X 3** to do it (even if He refers to a Princess)*
*He needs **Coin X 1** to do it (instead of "she needs 1 coin")*

5.1.3 Existing technologies for interactive application localisation

Several technologies and perspectives exist in the localisation and translation industry and are used in other applications beyond games:

Translation Memory

Translation memory allows for re-use of what has been translated previously by a professional translator. The importance of terminology is huge. At present translation memory tools use either segment based translation memory engine or a corpus based engine, or both. This is called a "terminology management system". This ensures consistency in the translation by automatically searching for previous examples of where phrases were translated in the document.

This solution has several important limitations:

- It requires having a permanent access to a large data base, then it may only supports applications that are doing their localisation online at real-time in a central powerful server.

- The bigger the data base contents, the more efficient the result. The data base should then be increasingly updated, and can only be efficient when reaching a huge critical mass of data. As today, the biggest translation memory data bases of the industry have a success rate of 80 % or less.
- This technique does not allow embedded systems – only real-time online application may use it.
- This technique is based on memory of phrases; therefore it does not handle any context variable mechanism.

Interchange format

An interchange format is crucial to improving translation quality and process. The interchange format needs to be able to store all the information from the development environment needed for localisation and all the information returned from the localisation environment needed for development. There is already a localisation interchange format, XLIFF. This can be used as both a reference and starting point for the interchange format for games localisation. However, it is likely that there will be data needed for games localisation that XLIFF does not deal with. The interchange format will either be a namespace which extends the XLIFF namespace or a new XML vocabulary. However, the Interchange Format handles text only and does not support at all variables and multiple values [i.4]. It is then, today, totally unable to resolve our issue.

Machine Translation

MT technologies are starting to be used in many applications, including in Internet applications or large applications with massive data. However, most MT technologies do not fit the technical constraints of video games, especially the most demanding of them such as MMORPG. However, these technologies, mature for some domains, are to be considered as new and innovative for the game industry (see details in clause 6 below). They are totally inefficient for handling context variables. In fact, the Machine Translation providers are part of those advocating for a standardized solution for handling context variables, since their machine will then be able to work by processing encoded scripts with context variables, which they are unable to do today.

Embedded systems

The progress both on the translation quality and the miniaturization of the translation engine could lead to a real-time translation with a translation engine embedded into the game itself. It is then possible to design a local application tailor-made for a specific game that would translate any text in real-time, but it needs to be based on a standardized way of writing interactivity scripts (one of the objective of the present study). This will require developing specific resources to deliver the expected level of quality, and integration features in order to have a Machine Translation component that can easily be embedded in any game product. This solution is today impossible.

Hybrid systems

Machine Translation systems can be technically based on several types of translation mechanisms (see next clause for details): rule-based translation, translation memories, dictionary based techniques or statistical techniques, as well as hybrid systems using several of these techniques. These systems are the most promising ones today, but they are not addressing our problem for several important reasons: First it is too large and dynamic to be embedded; therefore hybrid systems could only work through central servers and high speed telecommunications. Second, such systems cannot handle the variable tables, since there is no standard way to encoded interactivity scripts.

Important: one could think that some language technologies could solve the problem of variable-based context dependent interactive applications such as games. This is not the case. On the contrary, the providers of languages technologies, MT community, engines and middleware providers and CAT tools providers are all supporting the initiative of defining guidelines, recommendations and a standard for the whole community. Having a common way and process to handle this issue would allow these providers to offer innovative solutions, elegant and powerful development and localisation environment.

5.1.4 Existing proprietary solutions for interactive application localisation

As explained above, several large companies have developed their own internal solution or/and process. Several of them have agreed to release some of their internal information to the limit of proprietary and protected information – (available upon request). In the game industry, we can take the real-life example of a company we will call XXX.

The case of XXX

XXX has really developed a full internal system aiming at solving most of the localisation problem for context based applications. Although focusing on the games, it is very interesting to see that XXX is using the same system for

solving the localisation problems of many other internal departments, such as their marketing (for the CRM and direct marketing), their Quality Assurance, and all their web applications and web commerce. Therefore it demonstrates that such a solution is usable not only for games but for all types of interactive applications such as eLearning, internet or phone applications, etc. The philosophy of their system – with very little technical content – is even fully described in their patents [i.5].

The requirements of XXX are heavy since they develop and support many games, especially MMO (Massive Multi-users Online games). In their oldest MMOs, word count was about 10 000 words. Today's MMOs provide an average of 1 000 000 words each (more than the Bible), per release with an update for new release/bug fix/enhancement of 100 000 words average every 6 weeks. Most games are released in 7 languages (English, French, German, Spanish, Japanese, Russian, and Portuguese) for all platforms and up to 12 languages for PSN - PlayStation Network (following the standard of their PSN partner).

All XXX applications are structured using "string macros" with "string IDs", and the games are fetching at real time the string as attached to the string ID. The system of macros is similar to the context variable system that the present study is proposing.

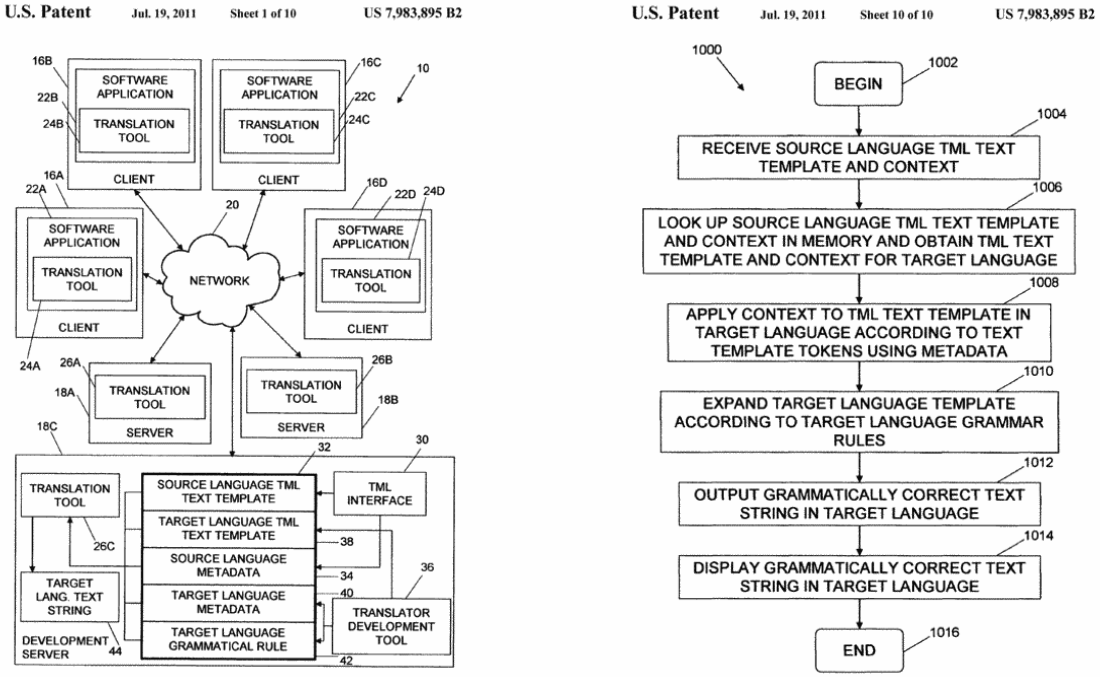

Figure 12: Schematics of XXX localisation system

They solution is first based on a large online data base (51 millions of text lines) used as translation memory, plus a large glossary. The Data Base indexing is working better and better overtime as the Data Base contents grows, and reaching high performances since, in all cases, an input sentence finds a matching phrase "80 % similar". As a result, they can rapidly provide the first localized version of the game. It is then given to human translators, who are supported by another XXX internal tool called the T4 or "grammatical engine T4". This tool contains all the gender/number/cases grammatical rules for 12 languages and resolve 80 % of these issues, given a macro description of the sentences when using context variables, depending on the quality of the code developed by the designing team.

Another internal tool, called the "CExplorer" is handling the main translation issue (for XXX): the context. The data base is attached to the CExplorer where the designers have stored and described all the elements of the games: characters, NPC, items, object, how they are related to each other, their profile, logs of activities, etc. Each string to be translated can then be seen with its context: who speaks, who he is, etc, so that it helps the translators to give the right translation based on game context. On another hand, designers also have access to a XXX development environment, attached to CExplorer, so that they can provide all the context information along with the creation of the game, details on each character, qualitative information that will be used later by translators when localizing the game. It saves them a lot of time since it avoids having long discussions with translators, long after they have actually created the game.

Finally, XXX provides both for designers and translators, a game play tool called the "CLog" which can record testers game play and then replay the game at real time, providing real time logs and the related code and text on the side, so

that when there is a correction to be done, it can be corrected on this side code/text right away and replayed to check it is then OK.

Analysis*:** XXX presents the whole tool set as their "Swiss knife" of localisation. It seems to be highly efficient, but in reality, it is limited to the good will and good usage of the applications developers. It has also the disadvantage of having to train the translators working for XXX to become experts of the T4 environment, which is only useful for XXX. Translators also have to work directly on the macros, with no tools for displaying "real life" phrases to be translated that would simplify their task. Finally, XXX also admitted that the system is not used or too late in the process when integrating applications or games developed outside of XXX design team but distributed by them. In short, although they want to keep this environment as a competitive advantage, XXX is also interested in seeing a standardized way of handling localisation information for context variables based application developers. This is why XXX has accepted the idea (to be validated by Legal department) to become an ***industry reference in the potential future development of our guidelines.

The case of YYY

YYY is a major European company well known for one specific game, one of the most popular and powerful MMORPG in the world today. Their need for localisation is enormous and their application is, of course, highly dependent on context variables.

More than 1 year ago, the company has experienced the problem and decided to launch the development of an internal localisation system called MLS2 that will support both the translators at the localisation step and upstream the developers at the design level. Here are examples of typical technical specifications (partial copy provided by the company with their explicit authorization):

- Linguistic object attributes for writers, translators:
  - Every entity, action, etc. in the game requires linguistic information specific to each language
  - Don't rely on code to do this; there are too many exceptions in any language
  - The amount of data entry per item depends on the degree of freedom both authors and translators
  - When developing an object editor, it is helpful for translators to see the item and related data, to provide context and resolve ambiguities
- Linguistic information with grammar rules:
  - Need both, but code tells us that we should use the genitive plural in Russian when the quantity is 5, but a translator tells us what that genitive plural form actually is
  - This does not mean creating a dictionary
- Improved Interface:
  - In content editors, when working with text, show the objects that can be referenced in dialogue, and make visible any necessary attributes on those objects
  - Improve syntax readability; i.e. [a/an], [der/die/das] style input
  - Basic conditionals for gender support, etc.
  - Better hint text for translators, better training for writers and programmers
  - Show preview sentences before submitting using actual object data to catch mistakes
- Clear best practices:
  - Place responsibility for correct grammar on writers, not programmers
  - Write grammar rules in code only when they can handle all exceptions
  - Make writers and programmers aware of localization issues as it applies to them
  - Internationalization necessarily includes English as a supported language

- During development, no player-visible text should be hardcoded or otherwise enter the game outside this framework

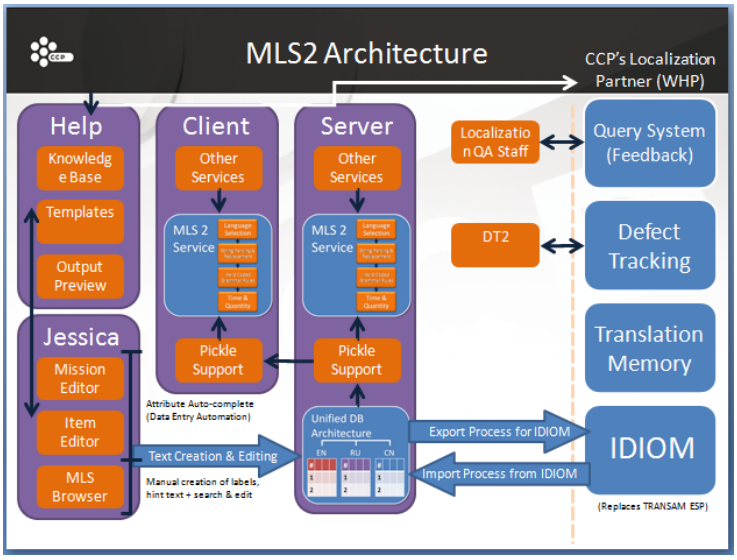

Figure 13: Schematics of YYY Games localisation system

The system is still under development, and no restricted information has been given so far about its status, its testing and its performances. The point was mainly to acknowledge that the same problem is seen by all and that the big players are trying to solve it by investing a lot of resources.

Please note that, like for XXX, YYY Games has been in permanent contact with us and is quite interested in following and even supporting our standardized initiative, since it will provide them with key elements they could build on and then they could avoid spending large money and time on non-core business efforts.

General specifications of localization systems for context dependent applications

The case studies above, as well as direct discussions with engine and middleware providers, clearly show that all target localisation systems are always based on the same technical principles:

- The localisation system requires high performance and efficiency to handle context variables correctly.
- A clear separation is to be made between the application code, indicating the dialogue with potential linguistic tags whatever the target language, and the system itself that is focusing on implementing the grammatical and cultural translation and accordance (gender, case, plural, social aspects, etc.).
- The dialogue within the code is written in a "markup language", as given along the present study – this language is one of the pieces that would benefit of a shared standardized way to write it.
- A common data base of glossary (individual words) or even translation memory (full sentences) may improve both the quality and the performance of the system.
- A link between the translator and the context is mandatory, the context being provided by the developer. Therefore a system should manage application context, filled by the developer along its design and used by the translator during his localisation process.
- Some limitations may be part of the best practices to avoid phrases that would be uneasy to translate in some specific languages, so the system should come with a "best practices" guide.

These are very straight forward directions that will be taken into consideration while performing the analysis and the recommendations.

5.2 Limitations of traditional ICT localisation rules

The reference document EG 202 417 [i.1] provides recommendations to ensure good localisation of mobile terminals and services. Some are general enough to be applied in context dependent interactive applications, but many are not; they however help to detect missing parts for our study and therefore the need for additional recommendations.

NOTE: The present clause is directly related to the clause "**Localisation aspects**" above. It intends to take existing localisation recommendations provided by ETSI for a specific ICT domain ("*Mobile terminals and services*") and show that, although some recommendations are general enough to fit out problem, other do not, and several topics are even missing and should probably be developed from this study. It then gives valuable elements and directions for scoping and developing the present study.

5.2.1 Optimized source texts

Initial recommendations, when relevant, are in *italic*, followed by the related comment and analysis.

- *Source text should be written as clear and short as possible (short and full sentences, use the active voice, do not use too many preposition and use standard phrases).*

  There are controlled language tools that help monitor how content is authored. This is used for Machine Translation, for instance, for manuals and users guides. This rule can be followed in general for static text and dialogue, when place holders are used in short repetitive sentences with simple structure (e.g. "**Press on %menufunction**"), however, it does not apply to the majority of the dialogue that is dynamic and context dependent, with infinity of possible generated phrases.

- *Editors and language specialists should help the technical communicators to write as optimized and consistent source texts as possible.*

  This is a very efficient rule. Experience shows that more than 50% of the localisation problems can be avoided when basic recommendations are followed in the source text. This gain is even verified in the game industry and game designers do not have the feeling that this is restricting their creativity. For instance, a rule can be to always address the user formally (e.g.: French "Vous" and not "Tu"). The cases where informal addressing would be required are very limited.

- *Re-use of phrases in the source text that have already been edited and approved.*

  As for the first bullet above, it fits simple structures and short sentences. This rule can then be followed for static text and dialogue, even if the dynamic text is generated on the fly. But it does not apply to the majority of the dialogue that is dynamic and context dependent, with a potentially infinite number of possible generated phrases. However, it gives an interesting direction for solving some of the problem, by being able to base localisation, either on the fly, or in post-production, upon memories of typical phrases. Some techniques exist that could be adapted or extended for our problem.

  It is important to note that Translation Memory (TM) systems are usually related to a project, a group of products or a specific company. For instance, HP uses one TM for its printers, another for its laptops and another again for its handheld translation. Too many strings in the same TM would pollute, even if great care was taken in the TM maintenance of attributes or meta-data to identify the origin of each string.

- *Language specialists should investigate which cultural and language specific adjustments have to be made to the target language.*

  This fits perfectly our problem. It is always possible to give guidelines and do a localisation analysis prior to starting a localisation project – this is what is done sometimes in the field. But this is still very limited. Therefore, languages specialists could provide a common *knowledge base*, whatever the format and tools around it, to be shared by all context-dependent interactive application developers and/or localisers, containing, in a common way, all cultural and language specific adjustments to take care of. This is to be analyzed in depth in the study and in future related tasks. Please note that, although highly relevant for localisation quality, this is not related to our focus on variable-based context dependent applications.

- *Project managers and editors should have a close cooperation with the group of people who manage the translations (often external partners). The constant workflow from source texts to target texts should have a very clear process, as this is crucial to deliver the mobile terminals at time.*

  Same comment as above. A key element for our study and its future outputs.

- *It is important to ensure that the style and terminology of a translated text/term correspond to the style and terminology used in the local market and within the company. This can be accomplished by having people who know the local markets and the industry, review the translations. The reviewers can give advice and comments on the choice of terminology and the overall translation to the translators.*

  Same comment as above. This means that all these language & culture related aspects relevant for localisations should be collected, stored and shared in a common way. It is important to keep in mind that although a lot of contents in terminology management can be shared, any system should be able to handle different flavours depending on the one using it. For instance, HP PC terminology may vary from the DELL one, because they have some preferred terms to say the same thing.

  Shared terminology knowledge should soon come with related recommendations and possibly tools. Its contents would be not only shared but also enhanced by all stakeholders as a crowd-sourced shareware of the community. There are companies or associations (such as TAUS, for instance) trying to foster such an approach. But this issue is out of our scope as it is not related to our focus on variable-based context dependent applications.

5.2.2 ICT localisation guidelines

Some localisation guidelines already exist in the field of ICT (see reference [i.1].

The present clause does not try to start providing localisation guidelines for the sector addressed in the present document (which will be the objective of a potential future work), but it intends to briefly analyse the existing ones in order to demonstrate gaps with our problem and to underline therefore the efforts to be made.

Initial recommendations from the ETSI document in reference [i.1], when relevant, are in *italic*, followed by the related comment and analysis.

Translation management guidelines

- To minimize mistakes as much as possible, it is recommended to use as many standard phrases/texts as possible that have been revised and approved earlier.

- Re-use translations efficiently (this depends on the quality of the source text, good terminology management, unified style and terminology in the target text, good version control of translation managements/content management, regular clean-ups and good CAT tools).

This can be followed for static text and dialogue, but it does not apply to the majority of the dialogue that is dynamic and context dependent, with infinity of possible generated phrases. However, it gives an interesting direction for solving some of the problem, by being able to base localisation, either on the fly, or in post-production, upon memories of typical phrases related to the context (game, learning domain, industrial target, etc.). Some techniques exist that could be adapted or extended for our problem.

- Differences among languages regarding the total number of characters required for a particular text should be taken into account in the process (i.e. when a text is translated from English to German or Finnish, the number of text will increase by 30 %).

This is related to the string length issue, which is critical for application parts where the maximum size is fixed (such as in menus, short lists, text commands, etc.). It is even worse with context variables applications since the final length of a string is not even predictable, since it may have any value. This point has been explained in details in clause 4.2 above.

- Translators should be provided with terminology databases and style guides.

Not only fits this well our problem, but it has even to go further, as explained above: All these language & culture related aspects relevant for localisations should be collected, stored and shared in a common way, with possible flavour differences (see above).

- Translations should be validated prior to shipping and evaluations of localised [~~user guides~~] applications with end users (e.g. usability tests or focus groups) should be conducted at regular intervals.

This is an indispensable part of the localisation process of most of these applications. However, the time required to 1) identify the problems and 2) to fix them, can be shortened by fixing recurrent placeholders/variable related issues at design and development stage and by using CAT tools giving enough context for the translation to be accurate.

Localisation guidelines

- Consider the target languages when producing the source text. Be aware of dialect variants, the adaptation of visual content to local cultures, formal and informal addressing, and the use of English-language terms.

This is the heart of our problem. This could be partially addressed through knowledge base and guidelines. However, this recommendation is a target and does not provide any methods or tools for ensuring it can be reached in the case of context dependent interactive applications.

- Avoid using humour, jargon and too informal language in the source language, as this can be easily misunderstood.

This is not always feasible in the case of immersive applications offering a high level of interactivity, such as immersive eLearning scenarios. Indeed, this sort of applications, to achieve its objective, should be as close as possible to reality; therefore humour, hints, feelings, etc are an integral part of the interaction between the user and the application or the NPC.

5.2.3 Missing localisation aspects

NOTE: Headings only are given here - See clause 4.2 "Localisation aspects" for details of each aspect mentioned.

As a short summary, here is a non-exhaustive check list of the localisation aspects that have been identified in this study, but for which there is neither study nor ICT-related recommendations as today, based on our current knowledge:

Grammatical aspects

- The basic grammatical specificities in languages such as gender, number, cases, plurals, etc.
- Translation may vary depending on the context. Some words have several meanings, not the same from a language to another, so that localisation should take context and situation into account.
- Additional idiomatic words that can be added to a sentence, that have no translation, but add an inflection or valuable information: an advice, an order, a question, a doubt, etc.

Social aspects

- The use of formal addressing in some cultures.
- Beyond the formal addressing, hierarchy relations, such as chief to soldier in a game, or the manager to his employees.
- Social behaviour and social relationship between the playing or Non-Playing Characters. This will lead to vary between formal vs. polite, familiar vs. casual, friendly vs. unfriendly, informative vs. directive, etc.
- Some languages use different words depending on who is speaking, and its relation with who he speaks to, depending on gender of the source and the target, and the hierarchy between both.

Cultural aspects

- Humour to be taken into account (contrary to traditional rules for ICT).
- The same issue concerning relations between man and woman, and sexism.
- Cultural taboos.

All the above statements are a reminder that problems related to the localisation of context dependent interactive applications are different from those of other applications. Respecting these new constraints is critical for the usage and

dissemination of these applications in a global world. It creates a new field of study, which requires rules, shared data and information and possibly tools and standards.

6 Language technologies' state-of-art

NOTE: As mentioned in the Scope, the present study focuses on text-based interactivity, since this is the core of all communications, even audio ones. Indeed, applications are either explicitly text based (messages are displayed to the user or taken from him through keyboard) or they add an audio interface, as input or output. Audio inputs are based on Speech To Text (STT) and Automatic Speech Recognition (ASR) technologies to be able to record and process user input. Audio outputs are either pre-recorded audio (then static and with no link with our scope) or based on Text-To-Speech (TTS) technologies able to generate speech out of a dynamic text. The present clause will not consider audio at all, and the speech technologies STT, TTS, ASR, although of high interest in Multilanguage systems, will not be presented in the report as being totally out of scope.

6.1 Machine translation (MT)

6.1.1 General

One of the main sources of innovation is also the components issued from ***Machine Translation.*** Machine Translation (MT) is a technology that was created more than 40 years ago, at the initiative of government intelligence agencies. Nowadays, the technology is mature and widely used for a great number of applications. However, most of these components are not adapted for the highly interactive games such as MMORPG.

The basic principle of MT is an automatic computer process performing a simple substitution of words in one natural language into another. As the mere substitution is insufficient, even for at least a correct result, technologies are used in order to improve the process: rule-based translation, translation memories, dictionary based techniques or statistical techniques, as well as hybrid systems using several of these techniques. Some components, whenever relevant to the present document, have been explained in the clause above.

Most current machine translation systems provide customisation by domain in order to improve output by limiting the scope of possible substitutions. This technique is particularly effective in domains where formal language is used. On the other hand, MT is poorly efficient in domain with informal interactivity, speech based applications and real-life real-time translation.

6.1.2 MT technologies

Rule-based, statistical and hybrid machine translation

The first systems that were developed were relying on rules describing the languages and dictionaries. The process consists in analyzing the source sentence, transfer it to a target language and build a new sentence applying grammar rules. The statistical techniques that were more recently developed are based on the processing on large bilingual corpora to create a translation model. A new sentence to be translated is then broken into pieces in order to find similarities with the model and a translated sentence is assembled from that.

Very recently, hybrid solutions were created in order to take advantage of both technologies through a combination of these. It is now generally admitted as the most promising way to obtain a good translation quality with a minimal customization effort.

Controlled language

The quality of the machine translation is highly dependent on the quality of the source text. Controlled language is a set of tools and methods in order to normalize the redaction of the source text, which allows increasing the quality of its Machine Translation.

Source correction

It is not always possible to control the quality of the source text. Another option is to analyze it before the translation process, in order to resolve ambiguities and to correct some of the errors that are obvious. This is a way to increase the Machine Translation quality.

Entity recognition

In any documents are figuring proper nouns, location names, titles, expressions that do not need a literal translation. It is key to identify those and to handle them properly, with a "*Do Not Translate*" for the appropriate conversion process.

Translation profiles

Every document or even part of document may be related to a specific domain and require specific resources in order to be translated properly. Using Translation Profiles is a way to manage the linguistic resources such as Dictionaries or Translation Models that should be used to translate a specific part of text.

Stylesheets

More and more, the text that is handled in document or localisation workflows is in the XML format. This allows the identification of sub-parts of the text and to have a translation process that is differentiated for each of these sub-parts, using a Style-sheet that will determine the action to take and the Translation Profile to use.

6.1.3 Advanced terminology management technologies

There is the need for a new technology handling the management of multilingual dictionaries. Those dictionaries are not only the correspondence of terms between various languages, it is also the management of the grammar category (e.g. verbs, nouns, adjectives, etc.) and other attributes such as the domains of use.

Source analysis

The first step in the translation process is the analysis of the source sentence. The "parsing" allows the attribution of a grammar category to each segment of the sentence, which can eventually be corrected by a user (writer or translator) to make it correct. This segmentation can then be used by a third-party application to ease the translation process or by the tool engine through the end of the translation process. Tags identifying the variables can be handled to take the appropriate actions.

6.2 Multilingual dialogue systems

6.2.1 General

There are very few multilingual dialogue systems today, and most of them (actually all) rely on direct translation. However, people communicating through machine translators cannot easily tell what the purpose of their communication is or what other people's intentions are because of the poor quality of translation, especially in case of speech-to-speech translation. In constrained contexts, spoken dialogue systems are appropriated for very specific applications, but remain very constrained in terms of tasks and languages.

The actual long term need, for which research is ongoing, is multilingual conversational systems that fully understand the intention of the user(s), independently of their language, supported through shared knowledge and understanding of the problem. These systems will also operate on the social aspects of communication, participating in multiparty conversations in a natural and expressive manner. Such systems will be platform-agnostic, and able to provide natural spoken interaction functionality to a wide range of applications.

What is typically presented as multilingual dialogue systems are systems indeed capable of "handling multiple languages", however, only in the sense that either different language variants of the given system can be run, for instance, by selecting a language at the beginning of the interaction, or the interaction language of the system can be changed in the course of interaction. Examples of projects from which such multilingual systems stem include 2 MIT projects, the Voyager project (http://groups.csail.mit.edu/sls/research/web.shtml) and Jupiter project (http://groups.csail.mit.edu/sls/research/jupiter.shtml), TALK project (http://www.talk-project.org/) (for English and German), LUNA (http://www.ist-luna.eu/) (Italian and Polish), the Xu et al. system (English and Japanese), or the DUMAS project (http://www.sics.se/dumas/).

More than bilingual systems are rare, an example involving English and Asian languages is the CU FOREX (http://www.se.cuhk.edu.hk/hccl/demos/cu_forex/) (Cantonese, English and Putonghua) and Amities (http://www.dcs.shef.ac.uk/nlp/amities/index.html) (English, French and German). Simultaneous interaction with a dialogue agent of speakers of different languages has not been so far widely addressed. In particular in the European context, addressing not only the main most researched spoken languages of Europe, but also under-resourced languages have not been receiving much attention.

6.2.2 Human-computer dialogue

Deployed, operational spoken dialogue systems' conversational capabilities are often restricted to pre-scripted, fully system-driven models. Practical task-oriented dialogues, e.g. information seeking dialogue systems or voice-operated device control, based on such models have been around for a while.

The complexity of dialogue, and consequently of the interaction model required for a computational system, depends, however, to a large extent on the complexity of the conversational domain or task. Complex domains which require reasoning or expert knowledge in order for dialogues to be possible at all, for instance, problem solving or tutoring, and let alone flexible mixed-initiative dialogue in such complex domains remain within the realm of research and experimental systems and are often built within large academic or collaborative industry-academia projects. Dialogues do not only involve strictly "on topic" exchanges addressing the domain or task at hand, but comprise a range of meta-level phenomena, such as grounding or patterns of social conversational behaviour.

A more natural dialogue is obviously more difficult to model, for instance, in terms of specifying the conversation's flow, representing the dialogue participant's beliefs, or recognizing and generating social cues. Several research projects are attempting to address the human factors in conversational human-computer interaction.

Dialogue systems are typically restricted to interactions within their dedicated domain or task, most often even to a single specific one. This is in no way surprising because the coherent flow of a conversation, i.e. the types of topics raised and the order, in which this happens, is strongly related to the domain or task structure. It is a known issue that porting a dialogue system to a new domain is a major task, which might even involve redesigning the entire dialogue system. While it is not realistic to think of entirely domain/task-agnostic systems at the moment, the strategy in advanced "easily adaptable" dialogue systems is to share domain representations if the different domains are closely related or to implement "pluggable" dialogue components as agents linked within a domain-independent larger architecture. A recently emerged trend in dialogue systems research is to exploit machine learning for various sub-tasks involved in dialogue modelling: from dialogue act recognition to modelling the interaction itself. Both supervised and unsupervised techniques are employed, with reinforcement learning slowly becoming a new paradigm. However, portability and scalability of reinforcement learning based systems remains an open issue. Moreover, training a learner for such a system typically involves a larger-scale data collection based on a dialogue corpus.

6.2.3 Multiparty dialogue

A typical setup addressed in spoken dialogue systems is that a given dialogue agent addresses one speaker (user) at a time. While multi-party dialogue has been receiving much attention in the dialogue research and conversation analysis communities lately, both in the context of focused tasks (http://www.amiproject.org/) as well as open-ended user-generated content, such as chat, research on multiparty interaction with real dialogue agents is still at its early stages.

KomParse (http://komparse.dfki.de/) is a recent project exploiting advanced natural language processing technologies to bring conversational capabilities to virtual characters in a MMO game. Perhaps the most comprehensive commercial serious game system, which emerged from a collaboration between academia and industry, and which supports spoken dialogue is DARWARS/Tactical Language and Culture Training System (TLCTS). TLCTS, [i.2] and [i.3] developed by ALELO, Inc., (http://www.alelo.com) funded by the US Defence Advanced Research Projects Agency (DARPA), is a virtual 3D world-based educational system for teaching foreign languages which has been built to provide language training to soldiers who prepare for US military missions. TLCTS comprises self-contained, adaptive language courses in Iraqi Arabic, Pashto, French and Levantine Arabic, and includes spoken language tasks, such as interactive dialogues with feedback on speech errors. The system helps learners acquire communicative competence while being guided by an intelligent agent through a number of 3D animated games. Learners perform different "missions" in an interactive story environment. The system has been deployed and has been used to train thousands of US soldiers. Aside from military scenarios ALELO, Inc. offers games involving simulated real-life social interactions with spoken dialogs and cultural protocols with "socially intelligent virtual humans" communicating with game players as autonomous, animated characters. To our knowledge, in Europe there is no project of comparable size and scope (and character) dedicated to dialogue-based interaction in serious games.

6.2.4 Current limitation of Multilingual Dialogue Systems

As a summary of the above technical presentation, the limitations of the existing technologies for Multilingual Dialogue Systems are as follows:

- Simple, restricted (pre-scripted) multiparty dialogue systems
- Mono-or multi-lingual grammar-based speech understanding

- Dialogue systems mainly limited to single users (one user at a time)
- Analysis of social signals from multiparty conversation

7 Generic analysis

7.1 Localisation requirements of the industry

As explained in the introduction part of the present report, the purpose of this study is to ***set the scene*** of the problem, explains what type of applications are facing it and why providing a standardized solutions is demanded by all the localisation sensitive industries.

The present study has been based on numerous discussions with localisation departments of different industries, as well as localisation experts and middleware providers, in order to detect which kind of problems they are struggling with and what kind of solutions they would like to see coming to simplify their life.

The first important conclusion of the study is that, although the problem was first identified and endured by the Game industry, ***the localisation problem of context dependent applications is totally generic*** and concerns all other industries developing such applications on their own field, either today or in a very near future. They all acknowledged the issue and the need for ***a common shared solution, guidelines or standard, and development tools***.

7.1.1 Necessity for absolute correctness for all languages

The technical study has shown that the industry requirements are more and more increasing in two directions: more and more languages and an absolute correctness in localisation results. Both directions can only be handled by industries and applications providers by offering a stable environment to both developers and translators all along the production chain, including a standardized methods and process for encoding dialogues and handling context variables:

- ***Large companies*** may (and sometimes do) develop their internal system, keeping it proprietary and making it a competitive advantage, but wasting a huge amount of time in R&D resources for developing and maintaining the system, and training localisers/translators onto their internal standards and tools. Since their system is usually made of encoding, which could be shared, plus translation memories and data, that is unique and their real core value, they all agree that the former part could be easily shared so that they could focus on their real added value.
- ***Other companies*** have no time or resources to develop their internal system. Some software companies are proposing localisation tools or middleware, but none are addressing the issue of context variables in localization yet (see NOTE). So these companies are today working around the problem, by investing a lot in post production human translation – to the limit of the existing code -, or by enforcing rules strongly reducing the quality of the result (see examples above). These companies, as well as the localisation tools and middleware ones, are strongly supporting the initiative of defining guidelines/standard to be used and shared by all.

NOTE: One company has already developed part of the tool – they are part of our expert team and agreed to participate to a future STF if any. In the same way, having a standard would help them focusing on their tool which is their real added value.

The study also shows that the number of target languages is increasing. In the near past, localisation could be limited to English, German, French, Japanese and Chinese, and sentences to be translated were mainly menus, lists of words and simple text with no or few context variables. Nowadays, the trend is clearly to more languages, more complexity and absolute correctness. Three major phenomena are now forcing industries to address this problem of localisation:

- Applications should now offer a ***totally immersed interaction*** in an increasingly realistic environment. Users are not accepting to be forced in to a foreign language (English) or gender (male) any more. Immersive applications are always based on context dependency then ***context variables*** at real-time.
- ***Globalization*** enforces distributors to provide a given application simultaneously in an average of 12 languages per release (typical industry standard) at least, and to add additional language for each target markets.

- ***Minority*** policies and markets are pushing distributors to more and more provide applications in minority languages, either for social and economical reason (access to e-Society) or for cultural reasons (strong mobilisation, dense market, cultural heritage preservation).

As a result, localisation is becoming a critical aspect of commercial growth, which increases the pressure onto applications providers for porting applications onto a broad range of languages, whatever their complexity and cultural specificities. The grammatical complexity of languages is endless (see summary above and annex A), and the cultural and social aspects, now to be taken into account more and more, are adding to the "technical" complexity a human and emotional layer very different, even within the same language, between cultures. Only a common approach and guidelines may help the applications development to address the localisation issue correctly.

7.1.2 Key localisation success factors

The situation is more or less the same in every industry, when facing variable-based context dependent content. But this is today highly critical for the game and eLearning industries because they are the ones handling today the largest amount of such context variables. However, many industries are about to reach the same requirements, such as telecom, internet applications and eCommerce.

As an example, we could take the actual market constraints of one major company of the game industry: this company is developing and/or supporting numerous games, especially MMO (Massive Multi-users Online games). In the oldest MMOs, word count was about 10 000. Today's MMOs provide an average of 1 000 000 words each (more than the Bible) per release with an update for new version/bug fix/enhancement of 100 000 words average every 6 weeks. Most games are released in 7 languages (English, French, German, Spanish, Japanese, Russian, and Portuguese) for all platforms and up to 12 languages for PlayStation Network. The need for an extremely powerful solution is then critical.

Several key factors of great importance are therefore shared by all the application providers:

- ***The Performance*** of the localisation process should be achieved in a very short time delay. The quantity of elements to be localised is drastically increasing. See the example above. Context dependency is exploding the amount of potential output dialogue to infinity. A simple solution translating words into other words is now irrelevant. So whatever the provided solution, it has to be fast and efficient.

- ***The Time-to-Market*** or rather the lack of it, is another factor which can have a very negative effect on sales of applications. If the localised application is not released at the same time or very soon after the original language version, potential customers may get tired of waiting and decide to acquire competitors' one. Software pirates can also end up releasing the original language version in the target market. In application localisation it is important that the process is optimised so that the software can be localised quickly without a loss of quality.

- ***The Quality*** of translation is crucial to both having a product that the customer is happy with and having this software released in sync with or soon after the original language version. Quality problems solved closer to the source save time and money. In order to maximise the cost effectiveness of application localisation there should be a standard translation process, which is designed for target type of software, and a seamless process where errors are checked and retrieved and translation errors are removed quickly.

- ***The Personalisation*** is more and more important for reaching the required level of immersion and correctness of the localisation. This includes a perfect utilisation of the user profile (male or female, age, social level, even accents, dialects and language variants, etc.) as well as sociolinguistic and cultural aspects (informal addressing, social relationship between speaker and listener, taboos, etc.). See clause 4 for details.

- ***The Terminology*** is important to translation in any context. When localizing an application, terminology can decide whether a product thrives or dies. Imagine a 14 year old boy in Madrid switching on his computer and anticipating the clever tricks and story of the games he has just purchased. Everything looks perfect. Can you imagine what would happen if the language was not the language of escape and adventure and youth but that of business and law? Terminology management technology should be used in order to ensure that the correct language and styles are used.

- ***The Context*** has to be taken into account at any level of the translation and localisation. It goes far beyond the personalisation and the terminology. It covers all contextual aspects of the application itself, of the user profile, of the user behaviour and history in the application, of the other users, players or non-playing characters (game terminology that is understandable in any other application) etc., all these elements that are stored in context variables which are to be taken into account, so that a translation will be different depending on the current full snapshot of the instant. This factor is the core of our study.

These 6 key factors are shared by all the industry sectors in their localisation objectives.
Any standard, system, process or environment to be put in place should address these 6 pillars of success.

7.2 Localisation environment architecture requirements

The description of the few existing localisation systems handling context variables (only proprietary so far) as well as the deep analysis of the technical and performance requirements for such a system are leading to propose a localisation environment that is able to address the 6 key success factors described above. This generic architecture will help understanding what could be addressed by the ETSI Technical Committee if launching an STF, and what will be the responsibility of industries and localisation tools providers in the future localisation environments.

7.2.1 Localisation environment principles

As it has been established in the analysis of the different case studies above, as well as from direct discussions with engine and middleware providers, future localisation systems should always be based on the same technical principles, when willing to resolve the problem of ***context variables***:

- A ***clear separation*** is to be made between the application code, indicating the dialogue with potential linguistic tags whatever the target language, and the system itself.

  ➔ This is the ***application ENGINE***, designed by the application developer

- A separate part of the system itself is focusing on implementing the ***grammatical and cultural translation and accordance*** (gender, case, plural, social aspects…) once given a string, a macro, its tags and its context.

  ➔ This is the central GRAMMATICAL ENGINE and/or Data Base, part of the company know-how

- The dialogue within the code is written in a "markup language", as given along the present study – this language is one of the pieces that would benefit of a shared standardized way to write it.

  ➔ These are the interactivity SCRIPTS, designed by the application developer - could follow a standard.

- A common data base of glossary (individual words) or even translation memory (full sentences) may both be improving both the quality and the performance of the system.

  ➔ This is the TRANSLATION MEMORY and DATA, part of the company know-how

- A link between the translator and the context is mandatory, the context being provided by the developer. Therefore a system should manage application context, filled by the developer along its design and used by the translator during his localisation process.

  ➔ This is the application CONTEXT data base, filled by the application developer

- Some limitations may be part of the best practices to avoid phrases that would be uneasy to translate in some specific languages, so the system should come with a "best practices" guide.

  ➔ This is part of the localisation GUIDELINES, proprietary know how or shared if defined as a standard

7.2.2 Localisation result for the application end-users

Once the application is released and localised using the localisation and development environments below, the end-user will never see the mechanism but should get 100 % correct sentences and words whatever the situation and the context variables he is generating and using.

- At run time, the embedded application engine is taking context variables values and generates, real time, the output sentences based on the dialogue encoded in the embedded scripts.

- In the case of large online applications using central data stored in a central server, translation memory and data might be used, depending on the architecture of the system.

- In case of fully embedded applications; everything stay transparent and is stored in the application itself without any need for connection or access to databases.

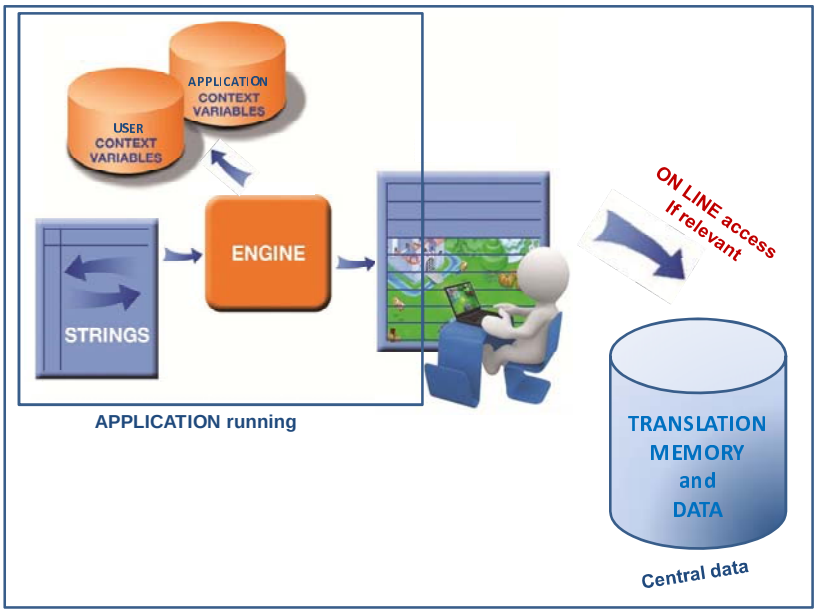

Figure 14: Localisation results for the application user

7.2.3 Localisation environment for the application developer

From the application developer (and potentially the engine developer if different) point of view, the localisation environment should then be architectured as shown in figure 15.

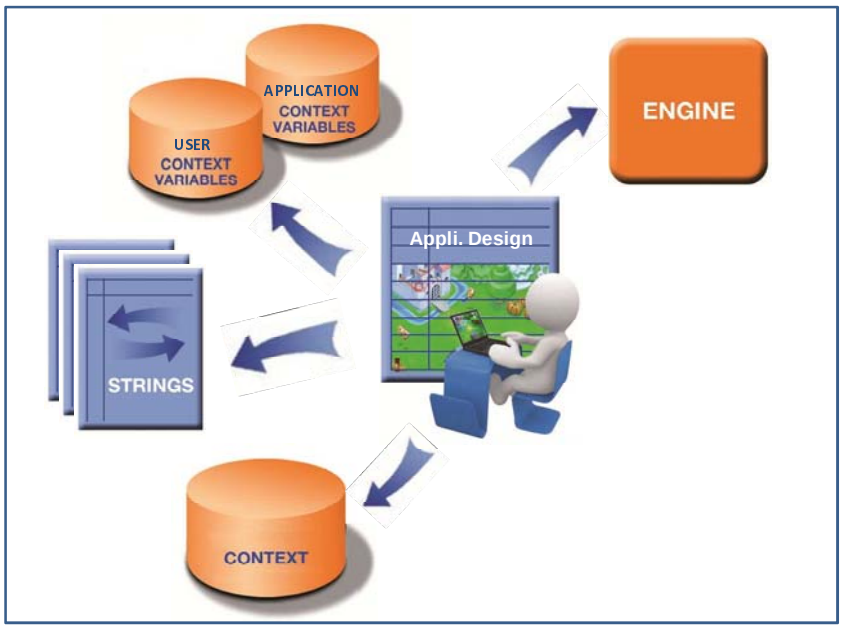

Figure 15: Localisation environment for the application designer

- The application designer is describing the interactivity, independently of the target language, within the application engine, making reference to phrases ids or links.
- He encodes the actual language dependent phrases within the interactivity scripts, using macros, tags, modifiers, etc., as defined in more details in the next clause. The localisation guidelines should be clear enough to motivate him to put all tags and modifiers future localisers will need, even if the current language he is using does not need them, for instance, mentioning numbering or cases.
- He records all potential context variables the application will need, as well as the related potential values (type and size, enumeration if limited, etc.).

- Any element, any character, any action of the application should have its description, its profile and the links with other object within the application context database. All these information will be used by the localiser later to better understand in which context a sentence is being output.

7.2.4 Localisation environment for the localisers/translators

From the localiser point of view, the localisation environment should be architectured as shown in the figure below.

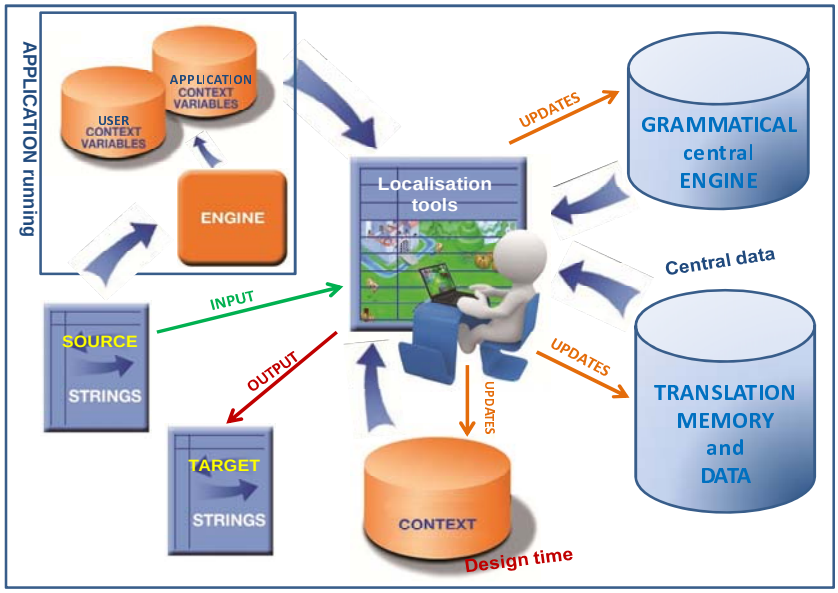

Figure 16: Localisation environment for the localiser

- The localiser has access to the scripts, the files containing the interactivity encoded using macros, tags, modifiers, etc., as defined in more details in the next clause.
- All the possible sentences are stored there, but many of the elements will get their value at run time from the context variables, as defined in the application context variables data base or the user context variables data base.
- Either the translator is working directly on the scripts (as it is done today in some companies, which is tedious and source or error), or his localisation tools are helping him seeing the resulting sentences. In both cases, he should provide the related translation of elements, context variables values, variables, etc.
- The localiser has access to the context database of the application, storing context oriented description, tags or links attached to the variables or the situation in the application. As seen before, access to the context of a text phrase is key to the quality of translation.
- The localiser may have access to the Translation Memory and Data of the company, to support him in initial translation, whenever there is a matching.
- In the same way, the localiser can have access and use the grammatical engine storing all grammatical mechanisms such as number, gender and cases rules of the target (and source) language. On the other hand, he may also improve this grammatical engine by adding rules.
- Future cultural and social information of localisation will be implemented in this grammatical engine.

7.2.5 Localisation environment for the society

Finally, it is quite interesting to analyse what will be the impact of such a localisation environment for the Society. It actually offers new opportunities for the society and in particular non or less-supported languages including minorities:

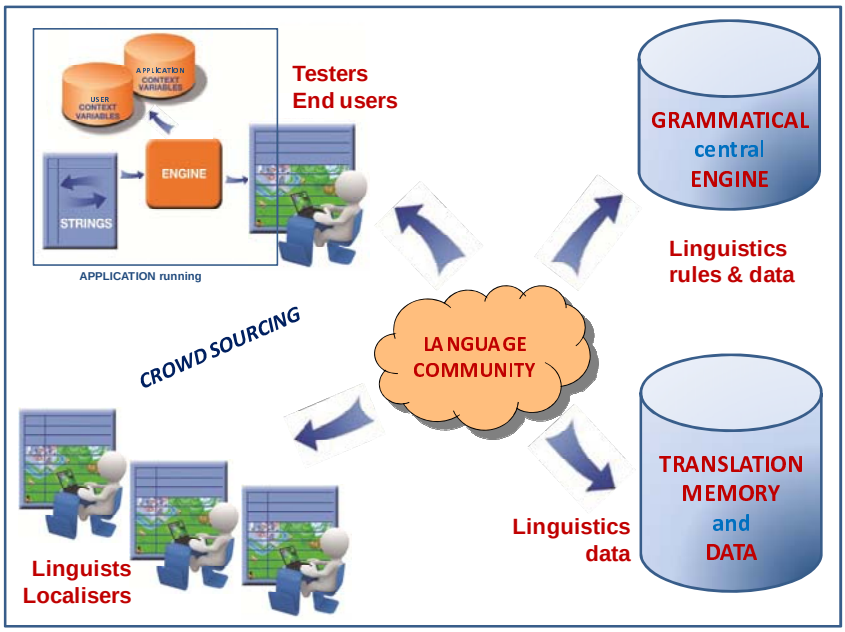

Figure 17: Localisation environment for the society

- This clause addresses situation of less or non-supported languages, such as minorities (Occitan, Sami, Basque, Corsican, etc.), and less-supported actual national languages (Finnish, Polish, Thai, etc.) as well as variants of supported languages (Austrian German, French Canadian, etc.).
- Cultural communities based on non or less supported languages may take advantage of the flexibility and portability of this localisation environment.
- Any language community may provide application publishers and developers with the linguistic and cultural data for creating the related grammatical engines, as well as for improving/completing Translation Memories.
- Any language community may provide linguists and localisers expert in their language for improving or utilizing the localisation environment for their culture/language.
- In the same way, any language community may provide testers and end-users in their own language, to validate applications and/or improving the whole localisation environment for their language.
- In general, such localisation environments enable to efficiently use and take advantage of the huge crowdsourcing capability of the community (easily mobilizable and motivated).
- The proposed localisation environment is a powerful tool for eInclusion and eSociety European policies, including the preservation of cultural inheritance.

Conclusion: The role of the present study has helped defining which part of the overall required solution described above could be taken by ETSI as a future guidelines or standard, and which part stays the responsibility of industries and localisation tools and environment providers:
→ ETSI Guidelines should focus on how to write Scripts and Grammatical Engine in a common manner.
→ Industries and tools providers should focus on application engines, translation memory and application design.

8 Conclusions and recommendations

As mentioned above, in clause 4.5.1, the strategic conclusion of the technical study is the following:

> **All context-dependent interactive application industries stakeholders (see NOTE), beyond the sector of games and eLearning, are seeking recommendations and possibly, a standardized way to handle the problem, if necessary, at the cost of some grammatical or contextual limitations.**
> NOTE: Application designers and developers, publishers, distributors, engine and middleware providers, localisers, translators, CAT tools providers, Machine Translation providers, Translation Memory builders, end-users, etc.

The previous clause has explained in details the way a global system for localisation of context variables based applications should be architecture and who is responsible for each elements of this overall system.

Proposed actions suggested to be taken immediately:

Recommendation #1:

A new generation of localisation environment is required in order to handle the increasing need of context based applications. Core engines and middleware, as well as the related development and localisation tools are the responsibility of industrials and the private sector. However, the whole community is expecting a common and standardized approach for handling the encoding of context variables and the encoding of the related basic grammatical rules such as gender, number and cases. Guidelines might seem more appropriate than a pure standard.

Recommendation #2:

Although the problem of localisation of context dependent applications have been detected by the game industry, it is clear that all industries developing such applications are or will soon be facing the same critical problem, and then they will all benefit from such an initiative. Therefore, it is recommended to launch the guidelines study with the industrial support of the game and eLearning industries, with the strong implication of the telecom industry, which is already facing the problem, and in open collaboration with any other field which might overlook it for their future, such as the Intelligent Transport Systems.

Recommendation #3:

A first stage would be to define guidelines for interactivity scripts in order to address the handling of context variables as well as related basic grammatical aspects of localisation, especially gender, number and cases which is then covering about 90% of localisation problems. **This part is independent from the languages**. As a reminder, the technical study has defined the fundamental elements to be used in interactivity -See above for details.

- "***variable***" or "***context variable***" / "***modifier***" or "***variable modifier***" : "***attribute***" or "***variable attribute***"

It is also suggested to work immediately on a second stage to be able to integrate social and cultural aspects of the localisation, which are becoming more and more important for a perfect immersion and correctness. This includes indications such as formal versus informal addressing, for instance.

Proposed actions suggested to be taken next:

Recommendation #4:

It is suggested to launch another piece of work regarding guidelines on developing grammatical engines and data bases, one per language, handling the grammatical rules regarding gender, number and cases, and later any other grammatical specificity. It should include any type of grammatical information that the interactivity scripts and localisation tools may need. This is then tightly related. **Although the guidelines will be the same for all languages, actual data will be language dependent**, of course.

Recommendation #5:

Although not participating directly in the development of localisation tools, the ETSI and its Technical Committees should **support any industrial or research initiative** aiming at developing tools, engines, data bases and environment related to the above guidelines and standards. These initiatives could be collaborative projects, EU Framework projects proposals, or publicly funded private projects.

Annex A:
Linguistic complexity

The clause 4 of the present document is giving the basic flavour on how complex localisation can be because of massive grammatical differences between languages. It helps to understand the urgency to define shared localisation rules among applications developers and translators.

The present annex intends to give a broader and less trivial picture of the complexity of languages, and therefore to allow non-linguistic to grasp the high level of difficulty for ensuring grammatical correctness in the localisation process. ***However, this annex is far from exhaustive and languages can be far more complex than what is presented here***.

This annex will also help realizing how existing technical solutions such as Machine Translation are totally powerless and why a shared approach and system, through strong guidelines or standard, are mandatory.

The three main grammatical characteristics of a language are the number, the gender and the case. Languages also have more characteristics, in the way they handle conjugation, negation, etc., plus sometimes specifities far more idiomatic. The present document covers only the main ones.

A.1 Grammatical number

In linguistics, **grammatical number** expresses the count of the attached noun or pronoun, such as one, two, thousand… The rules and accordance may vary among languages.

In most languages, there are only two numbering: singular (1) or plural (more than 1). Plural of noun is created by adding a suffix (***key, keys***) with some irregularities (***man, men***). In so called "Romance" languages (French, Italian, Spanish, etc.), nouns, adjectives and articles are declined according to number (singular or plural). Verbs are conjugated for number as well as persons:

> ***la maison est belle, les maisons sont belles***.

Some languages, mainly from Asia, do not handle any numbering at all. The plural is given by repeating the noun or pronoun twice. For instance,

> ***Orang (a person) → orang-orang (people) in Indonisian***

In other languages, numbering can be more complex and handle singular (1), dual (2), plural (more than 2 that can be counted) and collective plural (plural that cannot be counted):

Example of dual form in ***Hebrew***: most nouns have only singular and plural forms, but some have distinct dual forms using a distinct dual suffix (largely nouns pertaining to numbers or time, or body parts coming by two, such as:

אלפיים /al'pajim	***two thousand***
שבועיים /ʃvu'ajim	***two weeks***
עיניים /eɪ'najim	***two eyes***

Dual numbering still exists in most Semitic languages (***Arabic, Hebrew***) some indo-european languages such as Tamil, Indonesian or Slovene, and their trace can be found in modern languages such as the English ***both***. Some very small languages even carry trial and quadral numbering.

Example of collective form in ***Welsh***: ***moch*** ("pigs") is a basic form, whereas a suffix is added to form ***mochyn*** ("pig").

In some languages, singulatives can be regularly formed from collective nouns, such as in ***Arabic***:

> حجر ***ḥajar "stone"*** → حجرة ***ḥajara "(individual) stone",***
> بقر ***baqar "cattle"*** → بقرة ***baqara "(single) cow"***

In some languages, even if the numbering is only limited to singular and plural, cases might vary upon the number, such as in ***Russian*** and many other Slavic languages: singular (1) requires nominative, numbers from 2 to 4 requires genitive singular and plural beyond 4, genitive plural:

У меня есть одна книга ***I have one book*** ***(nom.sing)***
У меня есть три книги ***I have three books*** ***(gen.sing)***
У меня есть пять книг ***I have five books*** ***(gen.plur.)***

The numeral "one" has even a plural form, for expressing specific plural-only words:

Одни джинсы ***one pair of jeans***
Одни часы ***one clock/watch***

Finnish language has a plural form of every noun case.

talo ***house***
talot ***houses***
taloissa ***in the houses***

Moreover, when a number is used, or a word implying a number such as "many", the singular partitive case is used.

kolme taloa – three houses

A.2 Grammatical gender

In linguistics, **grammatical gender** for a noun, a pronoun, an adjective or even a verb is the inflection given according to the gender of the subject. In many indo-European languages there are two or three gender classes usually called masculine, feminine and neutral gender. But the majority of languages carry 4 genders: male, female, animated and neutral. Some languages do not carry any, some carry more than three.

These types are used purely for linguistic classification and have no real-world implications. For instance, a ***car*** in English is neutral, in French ***voiture*** is female and in Spanish, ***coche*** is male.

Most Romance languages (French, Spanish, Italian, etc.) have only 2 genders. English and most Saxons languages have 3.

In some languages, such as ***Danish***, the former masculine and feminine genders of nouns have merged with time, into a new class called the ***common gender***, which however remains distinct from the neutral gender:

e.g. Danish pronouns - ***male, female, common, and neutral: "han", "hun", "den", "det"***

The grammatical rules for gender inflection also vary between languages. In Romance languages, gender inflects the noun, the pronoun and the adjective, but not the direct object which keeps is own gender:

e.g. French: La vieille femme aime son chien. Elle le regarde.
e.g. French: La vieille femme aime son chien. Elle le regarde.

In English, the pronoun will change, and also the pronoun of the direct object (possession):

e.g. English: The old woman loves her dog. She looks at it.
e.g. English: The old woman loves her dog. She looks at it.

Gender has also to be given in the case of undefined gender or dummy pronoun. In that case, masculine gender is usually the default one. For instance, in French, which has no neutral, the action of raining requires a dummy pronoun. ***"Il" ("he")*** will then be used:

e.g. Il pleut – It's raining
e.g. Il faut que tu partes- you must go

In many languages, the gender associated to unanimated objects is unpredictable. It is not neutral by default, but takes, even differently in the same language group, masculine or feminine gender.

e.g. Russian: луна (*the moon****) is feminine –*** *but* ***солнце (****the sun****) is neuter***

e.g. Polish: *the same word* ***księżyc (****the moon****) is masculine***

Some languages are also making a gender difference between ***animated*** and ***unanimated*** subjects. ***Polish*** is even carrying ***five genders***: personal masculine (male human beings), animate non-personal masculine, inanimate masculine, feminine, and neutral.

Finally, it is even interesting to mention even more complex specificity in gender: the ***Aboriginal*** tongue is known to carry a specific gender for fruit and edible things, and another one for insects only. The ***Basque*** language has no masculine and feminine, just animate and inanimate. Not mentioning the gender associated to abstractions, countries, words borrowed from foreign languages,etc.

This short description of gender is showing that there is infinity of patterns, which all lead to the fact that there is no or few rules, and plenty of exceptions, and that any localisation system should just learn it word by word.

A.3 Grammatical cases

In linguistics, **grammatical cases** for a noun, a pronoun or an adjective indicates its grammatical function in the sentence. For instance, a noun can play the role of a subject (the ***ball*** is blue), a direct object (Paul kicks the ***ball***), an indirect object (Paul has been hit by the ***ball***) or a possessor (This is ***Paul's*** ball).

The point is that, from a given root, each case is modifying the noun or pronoun or adjective root, (this is called inflection) usually by adding the related suffixes. Speaker then has to handle sentence analysis (to define the target case), use the related case-modified word, at the proper place in the sentence. In case of case conflict (for instance a genitive used with an accusative noun), there is a hierarchy of cases that gives the rule for precedence to define the right inflection to be used.

Some languages do not handle any case, such as French. Most European case based languages are using a set of 7 basic cases (the nominative, accusative, instrumental, dative, ablative, genitive, and locative) or a subset of them.

Some languages have a very complex case system; the most impressive language is certainly the ***Finnish***.

In Finnish, there are ***15 cases*** which can be divided into five groups, each of which consists of three cases.

- Basic cases include nominative, genitive, and accusative,
- General local cases include partitive, essive, and translative,
- Interior local cases include inessive, elative, and illative,
- Exterior local cases include adessive, ablative, and allative,
- Means cases include abessive, comitative, and instructive.

Some case has declension in singular and in plural, some are handling only plural (***comitative),*** some like the genitive with highly complex rules for genitive plural. This means that a given noun in Finnish will have at least 16 different inflections depending on its number and usage in the sentence:

	Singular	Plural
Nominative	-	-t
Accusative	-n / - / -t	-t
Genitive	-n	-en / -in / -den / -tten / -ten
Partitive	-a / -ä / -ta / -tä	-a / -ä / -ta / -tä
Essive	-na / -nä	-na / -nä
Translative	-ksi	-ksi
Inessive	-ssa / -ssä	-ssa / -ssä
Elative	-sta / -stä	-sta / -stä
Illative	-·n / -h·n / -seen	-in / -hin / -siin
Adessive	-lla / -llä	-lla / -llä
Ablative	-lta / -ltä	-lta / -ltä
Allative	-lle	-lle
Abessive	-tta / -ttä	-tta / -ttä
Comitative		-ne
Instructive	-n	-n

Figure A.1: All grammatical cases in Finnish

In addition, there are also a lot of adverbial cases whose usage is limited to some words, these forms being usually regarded as adverbs:

	Singular	Plural
Superessive	-alla / -ällä	
Delative	-alta / -ältä	
Sublative	-alle / -nne	
Lative	-s	
Temporal	-lloin / -llöin	
Causative	-ten	-ten
Multiplicative	-sti	
Distributive	-ttain / -ttäin	-ttain / -ttäin
Temporal distributive		-sin
Prolative	-tse	-tse
Situative	-kkain / -kkäin	
Oppositive	-tusten / -tysten	

Figure A.2: All grammatical adverbial cases in Finnish

A.4 Other grammatical specificities

It is broadly admitted that the basic three grammatical elements (gender, number, case) are covering and solving more than 90 % of the linguistic complexity of a given language. However, languages also have many additional specificity that increases their complexity, especially for localisation or translation systems. This clause, without pretending to be exhaustive, is giving a short flavour of some of these particularities.

***Chinese*:** All the Chinese dialects are very similar syntactically although phonologically they are quite different. They are "isolating" languages where grammatical functions are not marked by inflection as in English or Russian; rather, words are immutable and not marked for subject agreement, tense, grammatical gender, number, or case.

***Japanese*:** There are only two tenses: past and non-past. A peculiarity of Japanese is that not only verbs, but also adjectives have the category of tense:

e.g. mi-ru 'see' (non-past) – mi-ta 'saw' (past),
e.g. utsukushi-i 'beautiful' (non-past) – utsukushi-katta 'id.' (past).

In order to express relationships and various grammatical categories (mood, voice, etc.), Japanese uses a great number of particles or affixes. Word formation and composition are then very productive. Adjectives can easily be formed from nouns with the help of the particle no:

e.g. Nihon 'Japan' > Nihon-no ochya 'Japanese tea'.

Hungarian*:** It is a richly inflected language with complex noun and verb forms. Nominals for instance are formed by stems followed by inflectional suffixes: ***stem + number + person + case; depending on context some or all of these suffixes may be omitted. The case system, consisting of ***seventeen distinct cases***, makes intricate distinctions. Ten of them are actually expressing various spatial and temporal distinctions, such as movement into an interior, onto a surface, into an immediate proximity, up to a point and no further, and so forth. On another hand, nouns are not marked for gender at all.

Telugu: is an elliptical language allowing for expressions to be omitted when their meaning can be deduced from the context. It makes it one of the most "elusive" of the world.

Khmer: The system of personal pronouns is one of the richest of the world. It is sensitive to the social standing of interlocutors, such as perceived status, age, and level of intimacy, so that a simple sentence like "thanks" or "hello" will have dozens of wordings depending on who speaks to who, their gender and social relationship.

***Chechen*:** Unlike the languages of Europe, Chechen first person plural pronouns are divided into inclusive and exclusive forms. The inclusive form of the first person plural refers to both speaker and hearer, while the exclusive form refers only to the speaker and one or more referents excluding the hearer.

Basque: is a highly agglutinative language. There are about ***sixteen postpositional affixes*** attached to the last element of nominal constituents. The verb is formed periphrastically with the aid of auxiliary verbs ***'edun'/'have' and 'izan'/'be'***.

Annex B: Bibliography

- D. Bohus and E. Horvitz. Dialog in the open world: Platform and applications. In Proc. ICMI-MLMI. 2009.
- Tanja Schultz. Globalphone: A multilingual speech and text database developed at Karlsruhe University. In Proceedings ICSLP 2002, Denver, CO, 2002
- Stephan Kanthak and Hermann Ney. Context-dependent acoustic modeling using graphemes for large vocabulary speech recognition. In Proceedings of the ICASSP, pages 845–848, Orlando FL, 2002.
- Mirjam Killer, Sebastian St¨uker, and Tanja Schultz. Grapheme based speech recognition. In Proceedings of the 8th European Conference on Speech Communication and Technology, Eurospeech 2003, Geneva, Switzerland, 2003.
- Sebastian St¨uker and Tanja Schultz. A grapheme based speech recognition system for Russian. In Proc. SPECOM Speech and Computer, St. Petersburgh, Russia, 2004.
- F. Jelinek. Statistical Methods for Speech Recognition. MIT Press, 1998.
- Xu Y., Araki M., and Niimi Y. A multilingual-supporting dialog system across multiple domains. Acoustic Science Technology, 24(6):349–357, 2003.
- Pentland. Honest signals: how they shape our world. MIT Press, 2008.
- S. Raaijmakers, K. Truong, and T. Wilson. Multimodal subjectivity analysis of multiparty conversation. In Proc. EMNLP, pages 466–474, 2008.
- Vinciarelli. Speakers role recognition in multiparty audio recordings using social network analysis and duration distribution modeling. IEEE Transactions on Multimedia, 9(6):1215–1226, 2007.
- D. Jurafsky, R. Ranganath, and D. McFarland. Extracting social meaning: Identifying interactional style in spoken conversation. In Proc. NAACL/HLT, pages 638–646, 2009.
- J. Hakulinen, M. Turunen, R. Gil, R. Santos de la Camara, and N. Crook. Parallel processing of interruptions and feedback in companions affective dialogue system. In Proc. Interspeech, 2010.
- Tanja Schultz and Alex Waibel. Language independent and language adaptive acoustic modeling for speech recognition. Speech Communication, 35(1–2):31–51, 2001.
- L. Burget, P. Schwarz, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, K. Nagendra Goel, M. Karafiat, D. Povey, A. Rastrow, R. Rose, and S. Thomas. Multilingual acoustic modeling for speech recognition based on subspace gaussian mixture models. In Proc. International Conference on Acoustictics, Speech, and Signal Processing, pages 4334–4337, 2010.
- J. Dines, J. Yamagishi, and S. King. Measuring the gap between HMM-based ASR and TTS. IEEE Selected Topics in Signal Processing, 2011.
- J. Latorre, K. Iwano, and S. Furui. New approach to the polyglot speech generation by means of an HMM-based speaker adaptable synthesizer. Speech Communication, 48(10):1227–1242, 2006.

History

Document history		
V1.1.1	February 2012	Publication