Automating Production of Cross Media Content for Multi-channel Distribution

www.AXMEDIS.org

DE8.3.1.2 Multilingual Guidelines and Technical Solutions 1st Update

Version: 1.1
Date:22/09/2006
Responsible: ILABS (d.fuschi@giuntilabs.it) (revised and closed by coordinator)

Project Number: IST-2-511299 Project Title: Automating Production of Cross Media Content for Multichannel Distribution Deliverable Type: Report Visible to User Groups: Yes Visible to Affiliated: Yes Visible to the Public: Yes.

Deliverable Number: DE8.3.1.2 Contractual Date of Delivery: 31/08/2006 Actual Date of Delivery: 22/09/2006 Title of Deliverable: Multilingual guidelines and technical solutions 1st Update Work-Package contributing to the Deliverable: WP8 Task contributing to the Deliverable: WP8.3 Nature of the Deliverable: Report Author(s): ILABS, XIM, ANSC

Abstract:
The present document updates and complements what previously provided in DE8.3.1 in respect to multi-language support.

Keyword List:
Content, multi-language, support, technology, meta-data, synchronisation, translation, vocabularies

Table of Content

1 Executive Summary and Report Scope

To be highly attractive and marketable, content should be available in multi-language format and possibly, deliverable on multiple platforms as is, or just with minor adaptations. To achieve such results is necessary to introduce, or support, highly structured content formats so to enable the adoption of the presently most evolved solutions for multi-lingual content management. Therefore the present document focuses on the issue of multi-lingual support both at metadata and content level. Technical solutions for both issues are presented and described in some detail, at least from the logical point of view. The work to be performed is related to the following Task as reported in the Technical annex.

WP8.3 – Content Integration and multilingual support. For cross media content delivering integration and multilingual management problem are extremely relevant. This sub-work-package has the goal of integrating content identified in WP8.2 with additional information addressing multilingual aspects. The integration will have to be supported by specific tools. Thus expected integration result will involve: improving metadata, adding text in other languages, addition of other related content components, synchronization among different media (such as an audio with the text, a video with the text, sequence of image with an audio) etc. Such process will have to be done in a large scale to generate enough valuable content for the validation phase and to prove the effectiveness of AXMEDIS tools for accelerating content composition and formatting thus making possible also content on demand provision. For text translation standard commercial tools will be used while for the other aspects specific activities and tools will have to be identified and developed. Development will be performed in WP5. Other details: A first version of guidelines and defined mechanisms for the production of content with multilingual support, multilingual metadata.

Having stated this is necessary to pint out that the present document is an update to the previously delivered DE8.3.1 and therefore will build on such document further refining and detailing certain aspects. In any case it is worth noting that issues related to multi-lingual support in coding and programming will be mainly covered in the guidelines related to the framework development and maintenance where are more suitable, while here will be pointed out issues are aspects that may have a direct impact also on coding (like impacts of Unicode on data-structures or of language support in data/code labelling).

1.1 Responsibilities

ILABS is primary responsible for the completion of the present document, yet all content owning partners are responsible for providing the required contributions including content samples (in terms of aspect, description and metadata). Technical partners are responsible for the corresponding part of the development guidelines plus the overall coherence check in terms of expected supported formats.

2 Assumptions

Given the scope of the document and its nature (update of DE8.3.1.2) it is worth recalling here the assumptions made in order to ensure consistency with previous document while avoiding useless duplications.

2.1 Language related issues

We have already mentioned that audio-visual content is often dubbed, or sub-titled, yet this is just an example of how to deal with multi-language content. Both methods present other kinds of issues like the synchronisation one, even when not taking into account the translation issue. There is then another full set of cases to be taken into account and among them the most relevant is the one related to metadata management.
Actually, for certain aspects, it is even more relevant to have metadata accessible in several languages, rather than content itself (it would be sufficient to think to language courses or to repositories of “original” content accessible to an international panel of scholars, etc.) yet in some cases if metadata are not supporting multilingual management, even a fully multi-lingual content may fail to be used or found.
It is generally accepted that there are several key issues to be considered in respect of multilingual content management. The most relevant have been already reported in the previous version of the document, while hereafter are reported the most relevant activities to be performed when dealing with multi-lingual content. It is important to take into account that some of these operations may occur either individually or combined with the others depending on several factors like the stage of product life cycle or new business opportunities coming form new markets opening.

Activity	Rationale	Solutions	Actors	Issues
Translation	• Whenever crucial or valuable information is not available in the original content • Whenever the target audience requires access to content in their language to accept and acquire it	• **Human based** (content and context oriented, expensive, usually accurate, always viable even if often reflecting knowledge domains related difficulties when dealing with some cultural dependent aspects) • **Machine assisted** (sentence oriented, inexpensive, potentially inaccurate, viable on limited well defined knowledge domains) • **Machine based** (word oriented, inexpensive, inaccurate, error-prone, viable if referred to limited well defined and controlled knowledge based vocabularies)	• Qualified personnel • External service • Specialized SW packages	• Proof reading of translated work is required for any adopted method • Sub-editing is needed as, depending on content domain, errors may be introduced despite the quality of the used system • For machine based is mandatory to use short, unambiguously structured sentences and to avoid idiomatic phrases • Content with significant translation requirements are likely to dwarf all other production costs except copyrights
Localization	• Whenever translation is not enough • Whenever a new market is opened to a product • Whenever a product has to be changed or adapted	• **Human based** (content and context oriented, expensive, usually accurate and performed with the support of several tools) • **Semi-automatic** (sentence oriented, reasonably inexpensive, potentially inaccurate, viable on limited well defined knowledge domains) • **Automatic** (word oriented, inexpensive, inaccurate, error-prone, viable if referred to limited well defined and controlled knowledge domain vocabularies)	• Qualified personnel • External service • Specialized SW packages	• Multiple language versions of content require different storage • Multiple language versions of content require different maintenance processes • Multi languages content management systems characteristics and functionalities will affect how a the products could be maintained and its cost • Cultural differences may be underestimated in their impact. Moreover even within a single language there are location related differences
Design	• Whenever a new product is conceived • Whenever a product has to be changed or adapted	• **Human based** (based on functional templates finalized and adapted manually to accommodate desired results, expensive, time consuming, difficult to maintain, usually accurate, performed with the support of several tools) • **Semi-automatic** (based on rigid functional templates adapted manually to accommodate needed changes, rather expensive, usually accurate, performed with the support of several tools) • **Automatic** (based on rigid functional templates, could lead to unexpected results in case of errors in the template design phase, requires expensive support tools like multi language CMS etc.)	• Qualified personnel • External service	• The most commonly, and an easily overlooked difficulty encountered in developing multi-lingual products • Text or graphic labels that fit the design constraints in one language may not work well in translation • Links between components should not lead unsuspecting users from one language to another • User feedback should not be solicited in a language if it cannot suitably handled • Content should be stored and edited as Unicode, published with an appropriate character set, language META tags should be published as html entities and direction tags should be specified, where appropriate

Activity	Rationale	Solutions	Actors	Issues
Workflow	• To properly and viably manage content production • To keep quality, time and costs under control	• Commercial solutions (costly, fully featured, requiring extensive customization efforts both installation and maintenance phases) • Open-source solutions (free, often limited in features, requiring significant customization efforts both installation and maintenance phases)	• Managers • Production personnel • Marketing personnel • Technicians • Suppliers	• Production, aggregation, distribution… work flow need to be properly modeled along with their interactions • Externalization of translation needs to be additionally modeled and handled if needed • Change management needs to be put in place and connected to both the production and translation cycles

It has been already mentioned that in the current global environment most organizations (commercial, governmental, nonprofit…) have a real need to present/offer/sale their content in multiple languages. This is not an effort free issue as apparent from what reported hereafter.

Content kind	Benefits	Issues
Multi lingual	• Addressing larger market • Facilitating customer acquisition & retention • Addressing specific company/institution needs • Enabling further exploitation of acquired content • Reducing production/aggregation costs for localized content distribution	• Qualified personnel needed • Multi-lingual CMS (m-CMS) needed • Could lack of specialization or adherence to local culture and habits • Translation activity/service required • Higher design/production/maintenance cost
Mono lingual	• Addressing specific/focused markets • Addressing specific company/institution need • Enabling a higher degree of specialization in offered content • Enables a more "customer-tailored" production • Enables a high degree of adherence to local culture and habits • Lower design/production/maintenance cost	• Qualified personnel needed • Mono-lingual CMS needed • Prevents access to those that do not know the specific language • Requires specific efforts in case of cross border exploitation • Reducing production/aggregation costs for localized content distribution

Taking as a starting point the just mentioned issues it is worth recalling here in some more details those that will have a higher impact level in the content management activity.

2.2 Content management tools

CMS and workflow engines can greatly reduce the troubles related to maintaining even a medium sized content collection (in a single language) thanks to direct editing and content management performed by the people responsible for the content production. Yet when the content has to be multilingual, the translation step occurs in the content production process and needs to be properly taken into account and handled. In this process multi-lingual CMS (m-CMS) bring a substantial help as the system organizes, speeds up and makes simpler the management of the translation process, yet this is not going to be costless.

2.3 Content structuring

If in most cases it is warmly recommended to have the same structure for every "language-based" content version, to greatly simplify the production & maintenance process; in the case of multi-language content (semi-)automated production process, it is mandatory to adopt fixed-structured functional templates. Such templates should basically fix content and navigation structure taking also into account the average difference in size among languages for the same item.

2.4 Content translation/localisation

In theory any multi-language content extension/update should be done simultaneously to all supported languages, but this is often not practical. Most often content is created in one "*master*" language, which is then translated into the other supported languages (often called "*target languages*"). It has to be accounted that there are several possibilities of approaching multi language content, where the difference lays in the extent to which "*master*" content is adapted and localized into "*target*" content (using the just reported way of addressing the language issue). What just said is roughly schematized in the following table.

Content type	Master language	Target language
Text	• Content is conceived, developed, approved, edited, composed, revised and finalized. • Content is archived and related data (including tags) stored in the CMS • Content is published & distributed • Content is made available to customers	• Master source content is searched in the CMS • Depending on content nature, business relevance, addressed market, intended audience, content nature, available budget… source content undergoes: o ***Full localization*** - Master source content is decomposed into elementary parts, each is translated, approved, edited, composed, revised and finalized o ***Partial localization*** - Master source content is summarized, summary and some selected excerpts are translated, approved, composed and finalized o ***Minimal localization*** - Master source content is summarized, translated, approved and finalized • Content is stored in the CMS and tagged • Content is published & distributed • Content is made available to potential customers
Audio-visual	• Content is conceived, developed, approved, edited, mounted and post-processed • Content is masterized • Content is archived and related data (including tags) stored in the CMS • Content is published & distributed • Content is made available to customers	• Master source content data is searched in the CMS and original source retrieved • Depending on content nature, business relevance, addressed market, intended audience, content nature, available budget… source content undergoes: o ***Full localization*** - Master source content is either dubbed or re-recorded in the target language o ***Partial localization*** - Master source content is dubbed for dialogues while other part of the soundtrack are left unchanged o ***Minimal localization*** - Master source content is simply subtitled • Content is masterized • Content is archived and related data (including tags) stored in the CMS • Content is published & distributed • Content is made available to potential customers
Web-site / CD / Application	• Content is conceived, developed, approved, debugged and documented. • Content is archived in the CVS and related data (including documentation) stored in the CMS • Content is released • Content is made available to customers	• Master source content is searched in the CMS/CVS • Depending on content nature, business relevance, addressed market, intended audience, content nature, available budget… source content undergoes: o ***Full localization*** - Master source content is decomposed into elementary parts, each is translated (including documentation), approved, re-assembled, tested and released o ***Partial localization*** - Master source content is decomposed into elementary parts, those to be translated are identified, needed structural changes (including those related to navigation are performed), selected components are translated and approved, the whole content is re-assembled, tested and released • Content is archived in the CVS and related data in the CMS • Content is released • Content is made available to customers

2.5 Internet & globalisation impacts

It is rapidly becoming routine for people to search for potential content and suppliers over the Internet. It is generally accepted that between 60% and 80% of companies Web site traffic is generated by searches for content/products, and is performed using search engines. Such searches will lead customers to the target only if the information provided/available presents the content in the language used for the potential customer's search.

Taking into account the content value chain we could briefly point out (for each phase) the major benefits/problems related to multi-lingual content in comparison to mono-lingual one.

Phase	Actions on content	Multi-lingual		Mono-lingual	
		Benefits	Problems	Benefits	Problems
Producing	Conceiving, designing, developing, validating and packing valuable objects along with related metadata	• Usually more marketable	• More demanding process (at same complexity)	• Usually more specialized	• Less demanding process (at same complexity)
Managing	Checking objects quality, assessing content related processes data, assess user feedbacks, check ROI	• World-wide 7x24 management • Wide coverage of customer needs	• More demanding process (at same complexity) • Need for specific m-CMS solutions	• Strong territorial binding • High correspondence with location specific customer needs	• Less demanding process (at same complexity) • No need for specific CMS solutions

Phase	Actions on content	Multi-lingual		Mono-lingual	
		Benefits	Problems	Benefits	Problems
Distributing	Producing/acquiring and delivering the right content for the right market with the right promotional effort	• Wider target markets • Easier searches	• A global distribution network is needed • Language skilled personnel is needed	• Usually more specialized	• Smaller target markets • Customer may require more assistance
Selling	Ensuring that the customer receives exactly what has been asking for	• Wider target markets • Easier searches	• A global sale network is needed • Language skilled personnel is needed	• Usually more specialized	• Smaller target markets • Customer may require more assistance
Maintaining	Updating, fixing, revalidating and repacking objects along with related metadata	• Possibility to compare languages • Easier searches	• More complexity • Need for cross consistency checks	• Less complexity • No need for cross consistency checks	• Impossibility to compare languages • Harder searches

Apart from the abovementioned value chain phases related to content owners, producers and distributors, it is worth taking into account (in the same fashion) also the others related to customers (both at B2B and B2C level).

Phase	Actions on content	Multi-lingual		Mono-lingual	
		Benefits	Problems	Benefits	Problems
Acquiring	Searching, identifying, selecting and accessing to the right content	• Easier searches	• Weak local territory coverage • Low correspondence with local needs	• Strong local territory coverage • High correspondence with local needs	• Harder searches
Using	Retrieving and accessing to the right content	• Easier searches • Easier access	• Low correspondence with local needs	• High correspondence with local needs	• Harder searches • Harder access

3 Solutions and approaches

Whenever there is content that needs to be localized is usual, at professional level, to go for proper translation and localization process. As already mentioned, this is a complex and structured process that often presents problems and unexpected issues. In the previous version of the current document were pointed out how the process is rolled out and how it could be possible to smoothen it. Now we will examine how the available technologies (and in particular those supported or compatible with AXMEDIS) could help in addressing this issue possibly reducing costs and increasing quality of both the overall process and derived content.
In the previous version of the document several steps leading to the completion of the translation process were identified, in the following table we still refer to them but for each we point out rationale, supportive technologies, constraints and issues.

Actions	Rationale	Technology	Constraints	Issues
Process selection (step 0)	Whether to go for a in-house or outsourced solution and which technology, and process, to adopt	SWAT analysis	Costs, quality, timeliness, error-rate	• Confidence
Scope definition (step 1)	Which are going to be the boundaries of the process in terms of style, completeness and accuracy of the localization process	SWAT analysis	Expected usage, target audience, costs, quality, timeliness, error-rate	• Marketing • Cultural aspects • Domain aspects
Volumes definition (step 2)	What is going to be processed and what is going to be produced in terms of quantities[1] and formats (including tools dimensioning)	Quantitative analysis	Costs, quality, timeliness, error-rate	• Feasibility • Viability

[1] Typically translations are quoted based on the number of words in the source document (or characters in the case of ideographic scripts such as Chinese or Japanese) and the document type. An accurate word count will enable to measure quotes against each other as well as give an estimate for time frame. As a rough guide, a good translator, paying attention to detail and producing quality copy can translate 2000 words/day (depending on topic, knowledge domain, source and target language this count can vary quite substantially).

Actions	Rationale	Technology	Constraints	Issues
Actors identification (step 3)	Identifying who will do what and when, at which conditions, costs… so to fix all process details and enable proper process management	Project planning	Costs, quality, timeliness, error-rate	• Feasibility • Viability • Confidence • Cultural aspects • Domain aspects
Process deployment (step 4)	Actually implementing the devised process, monitoring actions performed by each involved actor, checking quality, constraints respect, milestones achievement…	Translation • Human-based • Machine-assisted • Machine-based Project management	Costs, quality, timeliness, error-rate	• Feasibility • Viability • Confidence • Adherence to requirements • ROI

3.1 Translation/Localization Specific Issues

As AXMEDIS is dealing with content aggregation, adaptation and delivery, for both B2B and B2C market, encompassing and respecting all constraints coming form proper DRM support, it is mandatory to take into account the fact that translation/localization of content is an activity that has usually constraints coming from rights acquisition and enforcement. To this extent it is necessary to take into account the implications and changes that DRM introduction will bring in a context usually simply ruled by service contracts, non disclosure and/or confidentiality agreements.

Translation technology	Legal Constraints	Issues
Human based (content and context oriented, expensive, usually accurate, always viable even if often reflecting knowledge domains related difficulties when dealing with some cultural dependent aspects)	• Translation/localization rights should have been acquired • A service contract is required • A non disclosure and/or confidentiality agreement is required	• Content is usually made available in source form to a third party • Work is usually performed on systems non necessarily controlled / belonging to the production chain DRM protected (those of the third party) • Work is often performed and delivered in formats different from the source ones[2]
Machine assisted (sentence oriented, inexpensive, potentially inaccurate, viable on limited well defined knowledge domains)	• Translation/localization rights should have been acquired • A service contract could be required • A non disclosure and/or confidentiality agreement would be required	• As in the previous case • Cultural aspects may be underestimated • Domain specific aspects may be also underestimated • Use of Vocabularies / Translation Memories[3] is mandatory but is often difficult to cover specific knowledge domains, technical terminology or jargons • Additional errors may be introduced by the lack of context
Machine based (word oriented, inexpensive, inaccurate, error-prone, viable if referred to limited well defined and controlled knowledge based ocabularyies)	• Translation/localization rights should have been acquired	• As in the previous case • Free service offer a limited capability • On-line services follow a word by word translation approach which is often not suited for translating phrases especially when the grammatical structure of the sentence changes between source and target language

So far there already several specific issues that have to be taken seriously into account in respect of AXMEDIS automated approach to content management. More specifically we have to take into account what follows:

Relation	Rationale	Issues
Translation ⇔ Workflow	The insertion of a translation/localization step requires proper modeling in the workflow engine set-up of the content factory	• Workflow installation and implementation will require that proper control and management interfaces are established among people/supportive tools to track and manage performed actions

[2] Often professional translator receive a content in a format (HTML, PDF…) work out the translation in the preferred tool (MS-Word, Notepad…) and then return a "*text*" file that then will have to be processed at destination with clear shortcomings, including but not limited to operator error (cutting & pasting into wrong location, dropping characters during the process, etc…) or document incompatibility (when working across different versions of software, and operating systems, data can become corrupted or misinterpreted)

[3] A translation memory is a catalogue of recurring terms, phrases and sentences in the source document and their equivalent in the target language. The translation memory comes into play as the translator works through the project, allowing speedy insertion of stock standard items from the translation memory. This can be very useful for large-scale projects such as product operating manuals etc. but it can also lead to stilted, repetitive copy when used on other kind of contents and especially with marketing materials.

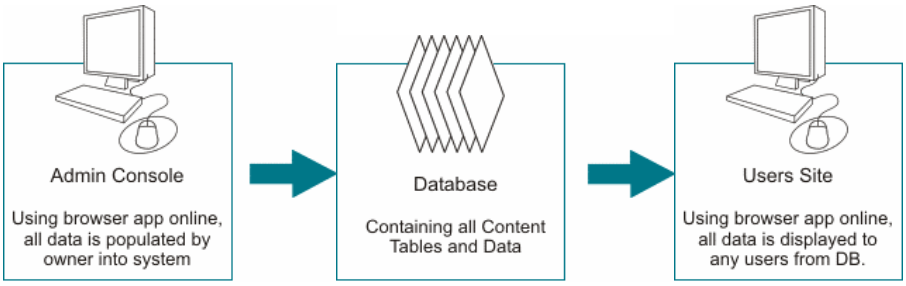
Relation	Rationale	Issues
Translation ⇔ Actors	The choice of a specific translation method / technology has a direct impact on involved actors included but not limited to the translation or proof-reading role	• Usage of in-house staff implies availability of required translation skills; has the benefit of not having to incur external expenses, but also has some embedded risks, as selected staff may not have enough language skills as the one required, or may not have the time to dedicate to the job, or may need to squeezed it in between all other tasks resulting in lover than expected quality. • Usage of professional translator implies to find a translator, and verify that is up to the level. • Usage of translation agencies is costly and still requires reference checking[4]
Translation ⇔ DRM	The choice of a specific translation method / technology has a direct impact on rights management (as already reported) and needs to be properly addresses	• Translation/localization rights should have been acquired • A service contract could be required • A non disclosure and/or confidentiality agreement would be required • Content may need to be made available in source form to a third party • Work could be performed on systems non necessarily controlled / belonging to the production chain DRM protected (those of the third party) • Work could be performed and delivered in formats different from the source ones

3.2 Translation/Localization, technology and workflow interaction

It is evident that the relation among the translation/localization process (including its extension and scope), the adopted technology and the availability of proper workflow tools is extremely relevant (even when ignoring the DRM aspect) therefore in this section we will examine the issue in more detail including suggestions and guidelines for managing the issue.

In the previous version of the document were already reported a set of diagrams related to the translation/localisation process (mainly in the web-based environment where such issue is extremely common) and to solutions commonly adopted for supporting it. Now we will use again those diagrams but with a different purpose, namely examine in some detail issues, problems and solutions that will occur or may affect individual steps, sub-processes or any other relevant aspect of the overall process.

As a starting point we will take into account a specific case, that is a multi-language web-site, as this will give us the opportunity to tackle all aspects of the production flow as well as several media which are usually found as components in a website. The basic flow that could occur would comprise only two steps: the database population and the content fruition. To this purpose basically three major SW components would be needed, namely the management system, the database/storage system, and the fruition system.



In such over-simplified process the management system may be composed of a single or a set of tools. Depending on its structure several actors may need to cooperate to achieve proper management and the presence or absence of an m-CMS could make the difference.

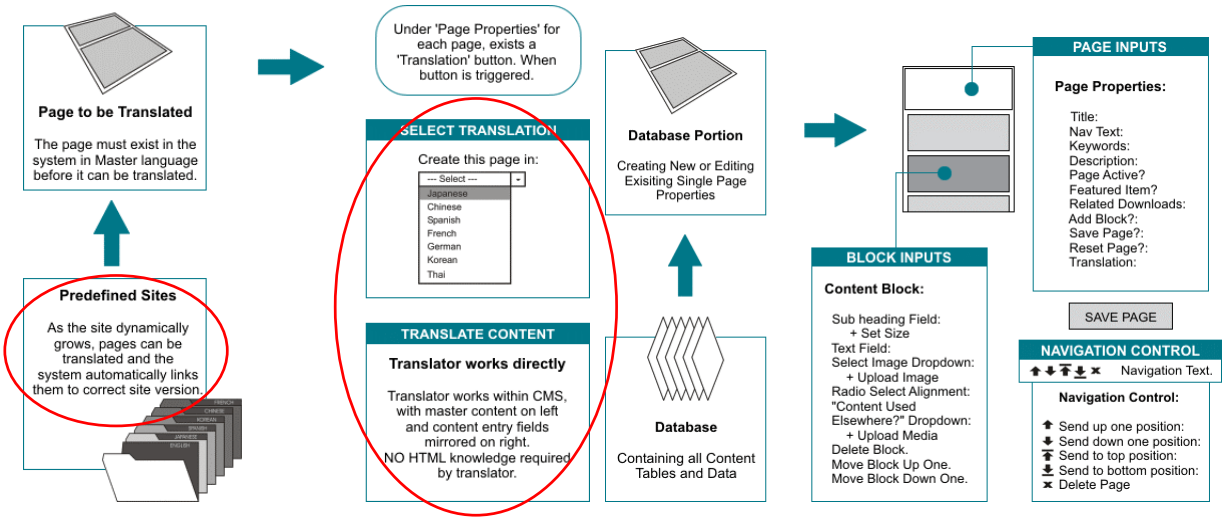
Also the content storage could be achieved in a set of different manners, staring from a simple file system, to get to an m-CMS encompassing databases or CMS. Also in this case, depending on nature and structure of the storage solution a quite broad set of actors will be needed.

The fruition side of the process may seem to be the simplest one but actually me be far more complex then expected if we take into account aspects like DRM or more in general security and payment methods.

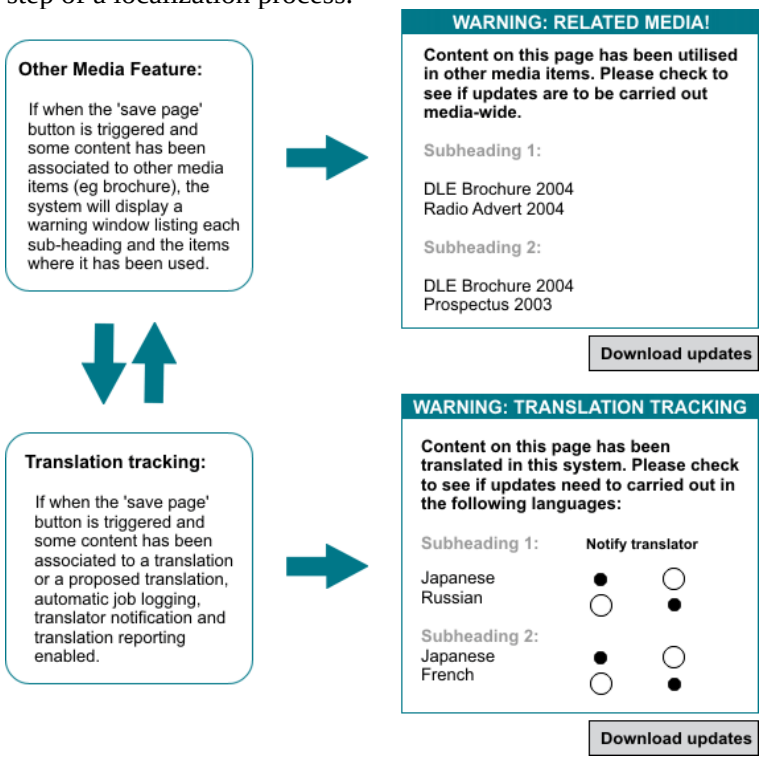
As a matter of fact depending on the kind of content that has to be accessed and the kind of usage that the consumer may be offered it may be necessary to add/use specific components (form simple cookies up to applets or specific clients).

[4] Agencies typically work across a range of languages and document formats; they verify the translators' skills and certify the quality of the output

Furthermore, given the current trend of consumers to look for a 7x24 freedom of access to contents and services, also the systems that will be used to provide such access have to be designed accordingly.
Last but not least the possibility for the user to access form wherever around the globe implies not only a need for a constant (or extended) availability but also for a true support of multi-language aspects by content, GUI and application.



In the chosen example of a website, and given what just mentioned, we could easily see that the adoption of an m-CMS could enable to simplify the workflow (especially as far as translation/localization is concerned) as the tool will take care of ensuring content consistency as well as other controls that otherwise will have to be performed manually with a error-prone process due to the complexity and interactions among many components, tools and actors. From the picture is apparent where the m-CMS will help and improve the process. Yet not all content needs to be multi-lingual, nor all site/application components have to be localized, therefore a similar result could also be achieved combining a mono-lingual CMS (yet supporting Unicode content) with a workflow engine for which the actual work model has been properly modeled.
We chose the example of a website as usually in websites one could find a quite blended set of media, from text to images, from audio to video including animations. The reason for taking this into account is that translation is only one step of a localization process.



So far we have explained the issues and impacts related to translation, now it is necessary to take into account also the fact that whenever a structured content comprises also other media than text, the translation of text is just a step, a necessary one, but a step. Missing to localize other components could jeopardize the content value or is marketability. To this extent a specific set of checks should be applied in the translation process to ensure that all components in need of translation/localization are adequately taken into account. Also in this case an m-CMS represents a good solution, as it takes this aspect into account, yet it is possible to achieve a similar result thanks to proper workflow modeling. In the following diagram is explained how this would happen in an m-CMS. To achieve a similar result using a workflow engine will require a bit more effort, as the process of check will be performed in a separate step and by a different entity (the workflow engine) from the one actually performing the ingestion of the translated content. This also implies that the workflow engine could be properly interfaced with all other tools involved in the process and that all aspects of the content management procedure are modeled along with involved actors and relations.
Having provided now the big picture it is necessary to tackle more specifically the presently viable solutions for achieving what so far exposed. To do this, as already outlined in the previous version, we will need to examine a bit more in detail how the various kind of content could be handled and which are relevant inherent issues.

3.3 Text

Text is the widest category of content to be taken into account when dealing with multi-lingual issues as all other content will somehow include some text (either as a specific part of the content or as metadata or as specific support to multi-lingual issue management for example subtitling…). Language tags should be used for all text-based content. A simple example of the use of multilingual text and metadata is in the production of html, where keywords and page descriptions need to reflect the language of the page content. In these cases, to properly support multilingual search and content accessibility, the best user experience is achieved by declaring the primary language of each page using the ***<html lang="nn">*** tag, where ***"nn"*** is an international language code defined under ISO-639 (***'it'*** for Italian, ***'en'*** for English, ***'fr'*** for French, etc.). Once a page's language is correctly identified, keywords and descriptions will then be indexed in the relevant primary language. Multiple language versions of the same content are best presented as separate pages, each with an appropriate language identifier tag and with all keywords and descriptions as well as content translated. This ensures that the result will be returned in the same language as the query. The ***<link>*** tag of the header enables to link a version in a language to alternate versions in other language, as show in the example from http://www.w3.org/TR/html4/struct/links.html#h-12.3.3 presented below:

```
<HEAD>
<TITLE>The manual in English</TITLE>
<LINK title="The manual in Dutch"
      type="text/html"
      rel="alternate"
      hreflang="nl"
      href="http://someplace.com/manual/dutch.html">
<LINK title="The manual in Portuguese"
      type="text/html"
      rel="alternate"
      hreflang="pt"
      href="http://someplace.com/manual/portuguese.html">
<LINK title="The manual in Arabic"
      type="text/html"
      rel="alternate"
      charset="ISO-8859-6"
      hreflang="ar"
      href="http://someplace.com/manual/arabic.html">
<LINK lang="fr" title="La documentation en Fran&ccedil;ais"
      type="text/html"
      rel="alternate"
      hreflang="fr"
      href="http://someplace.com/manual/french.html">
</HEAD>
```

This standard approach can be generalised to all XML documents, using the ***'lang='*** meta-tag to describe the language of the content. Also to support text readers and search engines, secondary languages used within text should be encapsulated within a tag using ***'lang=nn'***. There is another set of extremely important issues to be taken into account when dealing with content translation/localisation, namely that text size will change across languages and that there is a well know difference in the complexity/difficulty of the translation process depending on the couple formed by source and target language. This applies indifferently to human or

machine based translation even though in the latter case there are several additional factors to be taken into account, for example the current state of Machine Translation (MT) systems per language pair that are currently mainly devoted to some couples of languages as apparent form the following table taken from J.Hutchins' Compendium of Translation Software, 10th Edition (see following table from [20]).

From: \ To:	English	German	French	Spanish	Italian	Portuguese	Polish	Dutch	Greek	Czech	Slovak	Hungarian	Swedish	Finnish	Danish	Latvian	Estonian	Lithuanian	Maltese	Slovene	
English	-	48	42	48	29	30	7	6	3	2	2	3	1	.	.	1	.	.	.	.	222
German	49	-	23	8	8	3	4	1	.	1	2	1	1	1	1	.	.	.	.	.	103
French	38	22	-	10	9	5	1	2	3	1	.	.	.	.	.	.	.	.	.	.	91
Spanish	42	7	8	-	7	6	1	.	.	1	.	.	.	.	.	.	.	.	.	.	72
Italian	25	8	9	7	-	3	1	.	.	1	.	.	.	.	.	.	.	.	.	.	54
Portuguese	25	4	4	6	3	-	1	.	.	.	.	.	.	.	.	.	.	.	.	.	43
Polish	6	3	1	1	1	2	-	.	.	.	.	.	.	.	.	.	.	.	.	.	14
Dutch	6	1	2	.	.	.	.	-	.	.	.	.	.	.	.	.	.	.	.	.	9
Greek	2	.	3	.	.	.	.	.	-	.	.	.	.	.	.	.	.	.	.	.	5
Czech	1	1	1	.	1	.	.	.	.	-	.	.	.	.	.	.	.	.	.	.	4
Slovak	1	1	.	.	.	.	.	.	.	.	-	.	.	.	.	.	.	.	.	.	2
Hungarian	1	1	.	.	.	.	.	.	.	.	.	-	.	.	.	.	.	.	.	.	2
Swedish	2	1	.	.	.	.	.	.	.	.	.	.	-	.	.	.	.	.	.	.	3
Finnish	2	1	.	.	.	.	.	.	.	.	.	.	.	-	.	.	.	.	.	.	3
Danish	.	1	.	.	.	.	.	.	.	.	.	.	.	.	-	.	.	.	.	.	1
Latvian	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-	.	.	.	.	1
Estonian	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-	.	.	.	0
Lithuanian	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-	.	.	0
Maltese	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-	.	0
Slovene	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	-	0
	201	99	93	80	58	49	15	9	6	6	4	4	2	1	1	1	0	0	0	0	

Statistical Machine translation (SMT) systems derived from the Europarl[5] corpora vary in quality, this is reflected also by the achievable BLEU/NIST score achieved by systems trained using the Europarl corpora.

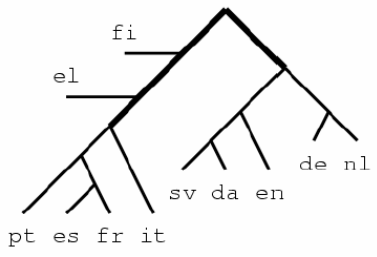
Source Language	Target Language										
	da	de	el	en	es	fr	fi	it	nl	pt	sv
da	-	18.4	21.1	28.5	26.4	28.7	14.2	22.2	21.4	24.3	28.3
de	22.3	-	20.7	25.3	25.4	27.7	11.8	21.3	23.4	23.2	20.5
el	22.7	17.4	-	27.2	**31.2**	**32.1**	11.4	26.8	20.0	27.6	21.2
en	25.2	17.6	23.2	-	**30.1**	**31.1**	13.0	25.3	21.0	27.1	24.8
es	24.1	18.2	28.3	**30.5**	-	**40.2**	12.5	**32.3**	21.4	**35.9**	23.9
fr	23.7	18.5	26.1	**30.0**	**38.4**	-	12.6	**32.4**	21.1	**35.3**	22.6
fi	20.0	14.5	18.2	21.8	21.1	22.4	-	18.3	17.0	19.1	18.8
it	21.4	16.9	24.8	27.8	**34.0**	**36.0**	11.0	-	20.0	**31.2**	20.2
nl	20.5	18.3	17.4	23.0	22.9	24.6	10.3	20.0	-	20.7	19.0
pt	23.2	18.2	26.4	**30.1**	**37.9**	**39.0**	11.9	**32.0**	20.2	-	21.9
sv	**30.3**	18.9	22.8	**30.2**	28.6	29.7	15.3	23.9	21.9	25.9	-

Table 2: BLEU scores for the 110 translation systems trained on the Europarl corpus

As already mentioned the difficulty of translation into and from a given language may differ widely as apparent from the following score table, more over, when clustered by scores the relations among language groups appear quite intuitively.

Language	From	Into	Diff
Danish (da)	23.4	23.3	0.0
German (de)	**22.2**	**17.7**	**-4.5**
Greek (el)	23.8	22.9	-0.9
English (en)	**23.8**	**27.4**	**+3.6**
Spanish (es)	26.7	29.6	+2.9
French (fr)	26.1	31.1	+5.1
Finnish (fi)	19.1	12.4	-6.7
Italian (it)	24.3	25.4	+1.1
Dutch (nl)	19.7	20.7	+1.1
Portuguese (pt)	26.1	27.0	+0.9
Swedish (sv)	24.8	22.1	-2.6

Average translation scores for systems when translating *from* and *into* a language. Note that German (de) and English (en) are similarly difficult to translate *from*, but English is much easier to translate *into.*



Clustering of languages based on system scores: Language families emerge

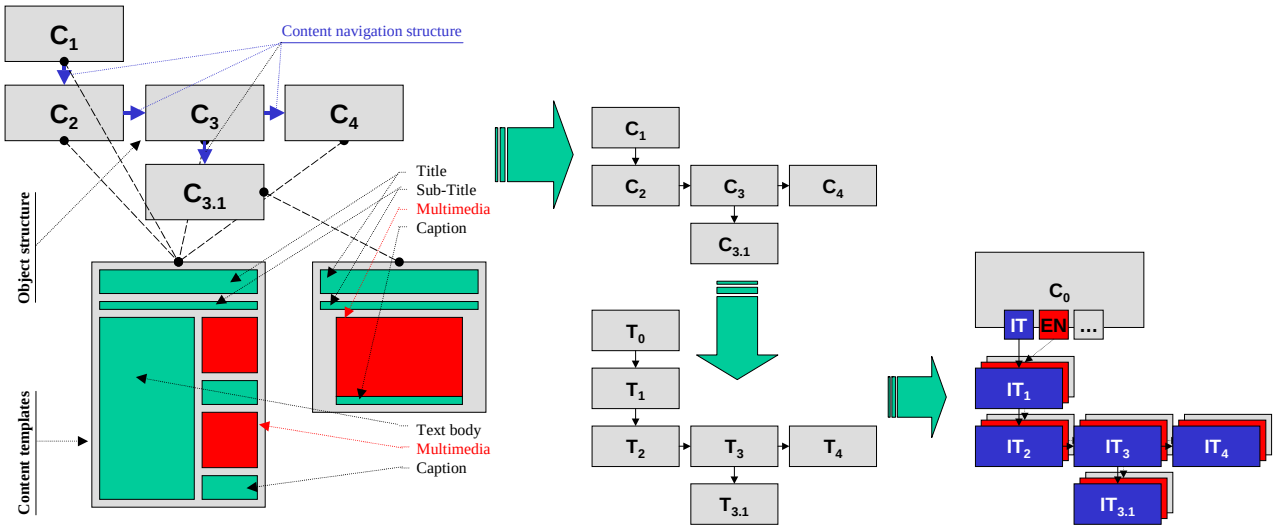
Given the relevance and occurrence of text among the kind of content that will have to be somehow localized is worth taking into account how this could be achieved and which are the major issues related to viable solu-

[5] Europarl: A Parallel Corpus for Statistical Machine Translation", Philipp Köhn, MT summit X, September 2005

tions. In doing this we will also often make reference to issues already reported in the assumption section of the present document as well as in the previous version.

Adopted approach	Rationale	Issues
Direct translation by internal staff	Internal staff performs the translation job, no specific support tool or application is provided for content management	• The activity may be squeezed among other staff routine activities lowering the quality • The translation will be performed either directly on a copy of the source or on some text editing tool and then manually inserted into the final destination either by the staff performing the translation job or by some personnel involved in the management of the target destination (this could be a error prone process) • Timeliness and accuracy of the process are hard to be kept under control
Supported direct translation by internal staff	As before but some specific support tool or application is provided for content management	• The activity may still be squeezed among other staff routine activities lowering the quality • The translation will be performed either directly on a copy of the source or on some text editing tool but supported by some tool (DB or similar) • Translated text will be stored in a support system then inserted into the final destination either by the personnel involved in the management of the target destination using the supporting system (this could still be a error prone process even though fare less than the previous)
CMS/m-CMS supported translation (internal staff)	As before but CMS specific support is provided for content management	• The activity may still be squeezed among other staff routine activities lowering the quality • The translation will be performed thanks to specific features of the support tool • The support system will take care of storing and placing translated text into the final destination
External translator (freelance)	A selected freelance translator performs the translation job, no specific support tool or application is provided for content management	• The translator identification process may be long and difficult • The translator reliability and quality are hard to be defined/assessed beforehand • The translation will be performed either directly on a copy of the source or on some text editing tool and then manually inserted into the final destination by the personnel involved in the management of the target destination (this could be a error prone process) • Timeliness and accuracy of the process are hard to be kept under control • There are confidentiality and security issues
Supported external translator (freelance)	As before but some specific support tool or application is provided for content management	• The translator identification process may be long and difficult • The translator reliability and quality are hard to be defined/assessed beforehand • The translation will be performed either directly on a copy of the source or on some text editing tool but supported by some tool (DB or similar) • Translated text will be stored in a support system then inserted into the final destination either by the personnel involved in the management of the target destination using the supporting system (this could still be a error prone process even though fare less than the previous) • There are confidentiality and security issues
CMS/m-CMS supported translation (freelance)	As before but CMS specific support is provided for content management	• The translator identification process may be long and difficult • The translator reliability and quality are hard to be defined/assessed beforehand • The translation will be performed thanks to specific features of the support tool • The support system will take care of storing and placing translated text into the final destination • The confidentiality and security issues can be kept under control more easily
External translator (agency)	A selected translation agency performs the translation job, no specific support tool or application is provided for content management	• The translation will be performed either directly on a copy of the source or on some text editing tool and then manually inserted into the final destination by the personnel involved in the management of the target destination (this could be a error prone process) • There are confidentiality and security issues
Supported external translator (agency)	As before but some specific support tool or application is provided for content management	• The translation will be performed either directly on a copy of the source or on some text editing tool but supported by some tool (DB or similar) • Translated text will be stored in a support system then inserted into the final destination either by the personnel involved in the management of the target destination using the supporting system (this could still be a error prone process even though fare less than the previous) • There are confidentiality and security issues
CMS/m-CMS supported translation (agency)	As before but CMS specific support is provided for content management	• The translation will be performed thanks to specific features of the support tool • The support system will take care of storing and placing translated text into the final destination • The confidentiality and security issues can be kept under control more easily

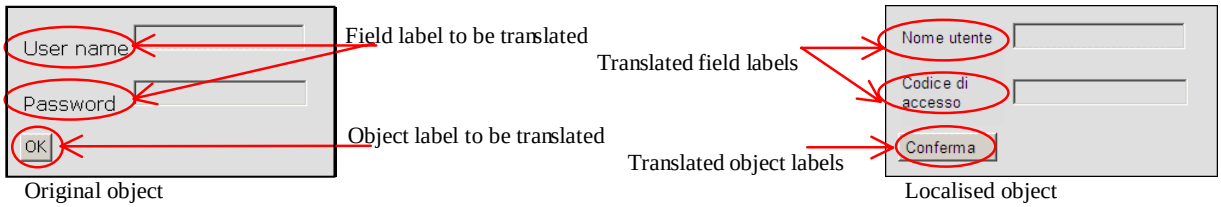
Given the extremely broad variety of cases to be taken into account when tacking text to be localised it will be advisable to adopt a web-based input system for possible supportive tool (including CMS/m-CMS).
As a starting point we will take into account simple support tools like the one achievable by combining a database and some active web-pages (.asp or .php or .jsp). Such an approach could be profitably used both in a client server or stand-alone basis depending on how it is implemented. Furthermore this approach could allow usage in all abovementioned situations (internal, freelancer or agency based translation) where front-end people are translators or content experts with no programming experience. In such an approach, for carrying out the translation, it is necessary to have a simple database that will be hosting the original master version of content and all translations. The database table(s) structure will have to reflect the content purpose and related structure so that the translation could be performed at an ***"atomic"*** level. Furthermore it will be necessary to have a table for each supported language and in each table (master or target) any record will correspond to a whole object and reflect its structure in terms of ***"record-elements"***.
This is in line with the approach for which a multi-lingual content could be seen as a common structure with a set of references to resources that are language dependent, but can be addressed in the same manner across languages. Once selected the language all content (including navigation/GUI elements) could appear to the user localised in a totally transparent fashion and consistent thanks to the fact that implied resources will be collected form all those available depending on the selected language. What just stated could be expressed graphically as reported in the following schema where a generic content (with a given structure represented by a navigation index) gives origin to a functional template where a layer ***"0"*** is added to manage the language selection. The same template is then used to give origin to an actual multi-language object where the navigation structure is consistent among languages and specific resources of each component are the one related to the selected language.



Generic content structure (based on components functional templates), derived structural template and multi-language instance

In the approach just presented there is the adoption of a double level of templates. Functional templates, used to give form to content's components, compose the first level; structural templates allowing access to resources depending on the selected language compose the second layer. In this manner components resources can be accessed in the same manner from each "***language dependent structure"*** thanks to the ***"functional template"*** approach in which resources are referenced and represented by objects external to the template itself (the title as well a the labels or buttons… are referenced and taken form external resource files) therefore offering the opportunity to refer to objects that can be equal in name but located in a different path that will be dependent by the language. In the base level of the structural template (***T_0***) it will be possible to accommodate all content specific characteristics that are ***"language"*** dependant like the text direction and orientation (right to left, left to right, top to bottom…) and operate the ***"language related path selection"***.
To keep content process management simple and allow easy access to people involved in translation, the interface should provide only basic functionality. In such an approach the interface could be kept (for simplicity) in the "***master language"*** only, even though it would be possible to follow the same approach to localise the GUI itself. This latter point will primarily depend on factors like how often the process of translation will have to be done, how relevant it is that the GUI is in a specific language or customisable into several languages, which is the budget available for the supportive tools, and many others that may vary from case to case on a company based approach. Last but not least even when following such an approach it will be pos-

sible to add some security management (for example by asking for a user log-in…), but once again this is an aspect of the support tool that will depend from a number of factors that are certainly out of the scope of the present document. In the following example we see a sample of a login screen that can be used as a guide to identify and explain in detail what just stated.

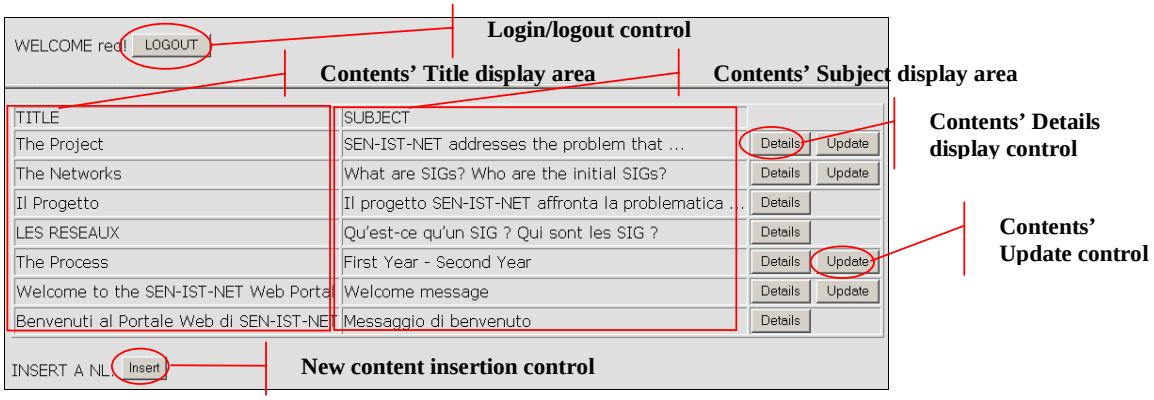


From the previous picture is also apparent the issue of different dimension in the size of resources across different languages. If the previously explained double template approach would have been adopted the present object will be structured into a set of functional parts: ***labels*** (label_1, label_2), ***input fields*** (in_field_1, in_field_2) and a ***control button*** (button_1). For each of this objects the caption of the component will be part of the object properties and would be referring to an external source for the content. Such sources will be located into the language dependent structure (one location per supported language) and the current one will be addresses by an environment variable.

As already mentioned it could be decided that for the sake of simplicity (or to keep costs bounded within a given budget) the support tool interface could be kept un-localised. In the specific case we will explain now such approach has been followed (the rational for the choice in the case was that all involved translators will use English as the starting and reference point therefore a GUI localisation was not required).

Furthermore it was decided to use a login procedure and to assign specific rights to each actor based on the role and master language of each user; therefore the Administrator and Master Editor would operate in English and be able to insert new items while Translators would only be able to access content, insert and revise translations.

The infrastructure will be based on a database and use a web-based access composed of ***".asp"*** pages. In the following picture is possible to see the basic interface that would appear to the Master Editor, main functions available are pointed out in the picture.



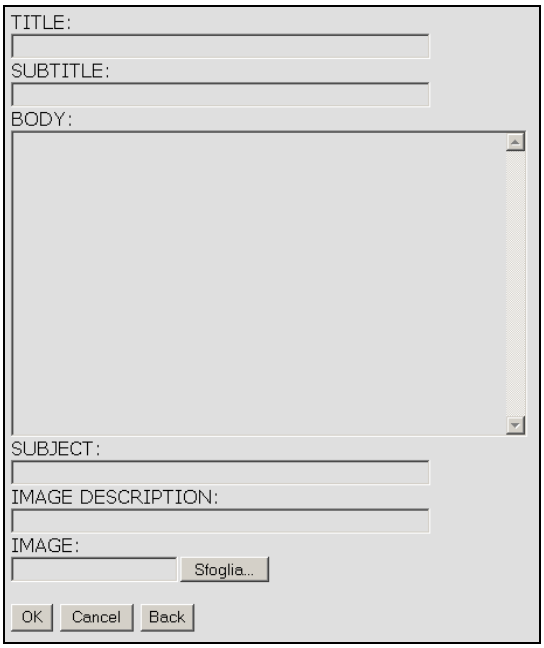
In the previous picture taken form the basic support tool taken into account and explained here, are also pointed out the main functionalities available to the Master Editor and the content display areas.

It is easy to identify basically three kinds of objects, namely, labels, I/O fields and control buttons. As far as I/O fields are concerned we have an area marked as **Title** where content titles are presented. Then there is an area marked as **Subject** where content subject is presented. Each line corresponds to a content stored in the system. In this approach not every field has an explanatory label, but fields have been grouped in a logical and functional manner so to limit the need to additional components (labels in this case) for the interface.

The controls available on the interface are basically divided in three groups. A first control rules the exit from the application (***logout*** button). A second group rules the management of content already present in the system; the **Detail** button will cause a window to open where it is possible to examine content details; the **Update** button will cause the opening of a window where it's possible to modify the previously inserted text.

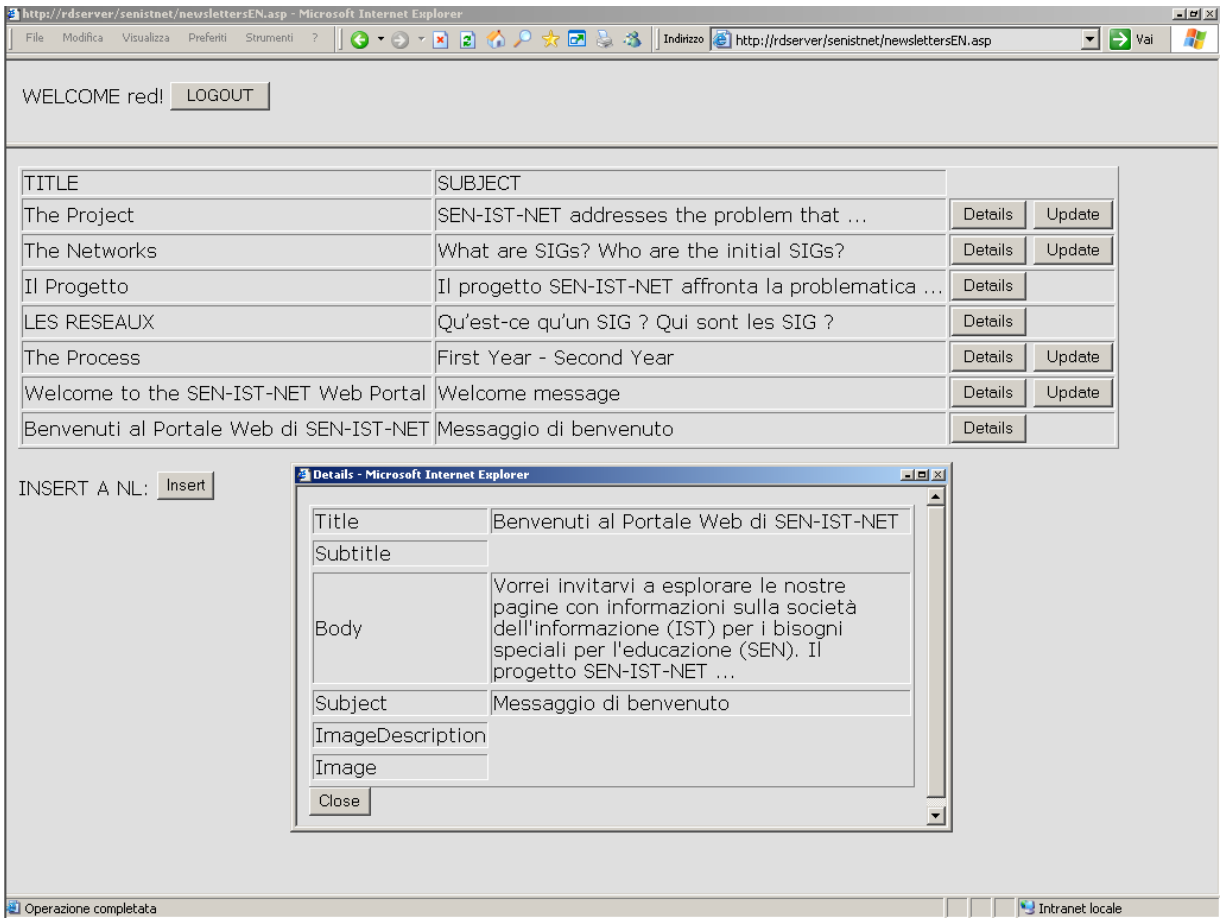
Finally there is an **Insert** button, which allows inserting a new text into the database. Insertion is allowed only for English text while for all other languages it is expected only to have either a translation (the 1st step in the process) or an update if the translated text needs modification.

In the following image is shown the input interface for a generic text insertion. In this example, each text shall present the following characteristics:

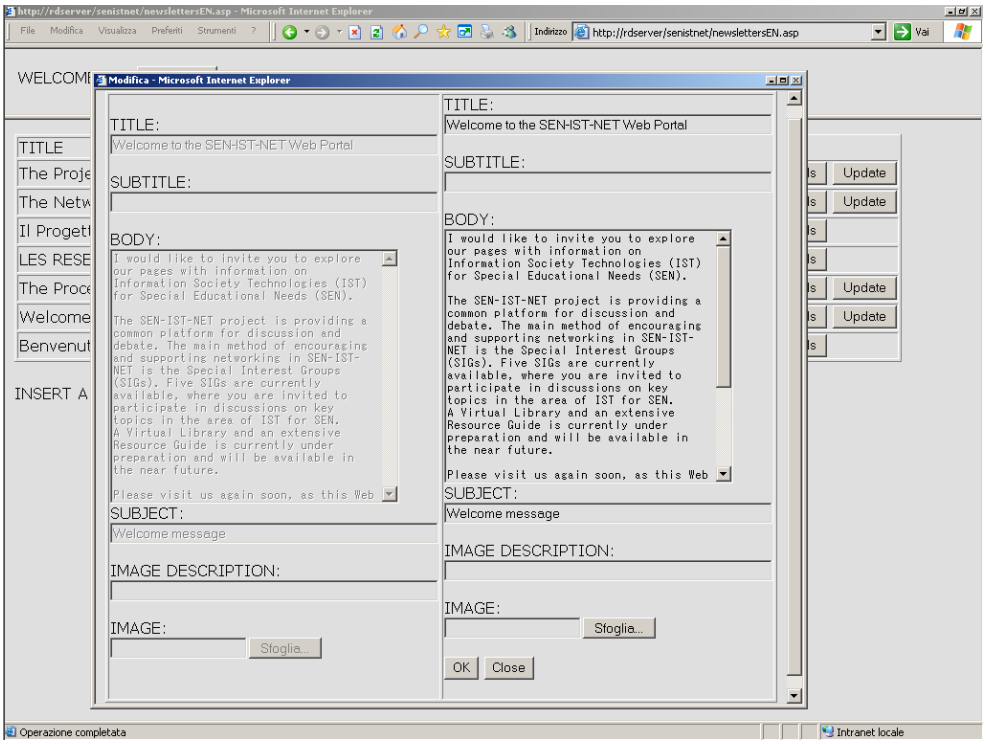


- **Title -** is the text title. It can be omitted if not necessary or it can be the only field in case it represents a section title.
- **Subtitle -** is the text subtitle. It can be omitted if not necessary or it can be the only field in case it represents a sub-section title.
- **Body -** it's the text body. The core information but can be omitted in case the inserted text has to be managed as a section / sub-section title. It can be left without title and / or subtitle in case it has to be treated as a plain paragraph
- **Subject -** this is a service field used to provide a support information to the user when accessing the main page as this description will provide information on the displayed record
- **Image description -** is the image caption and has to be loaded only if there is an image. It is possible that an image does not have a title/sub-title or associated text. If this field is present the next should hold a URL address.
- **Image -** is the URL address of the image referred by the caption.

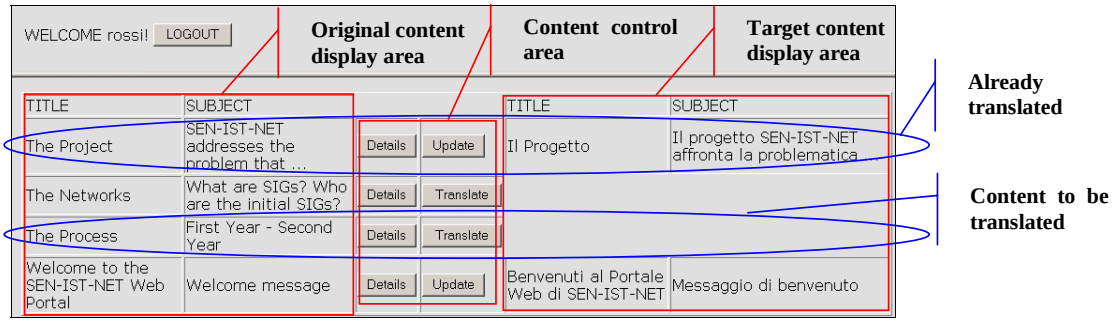
An image of the previous interface during operation is reported hereafter where the detail window has been opened. It this support tool the Master Editor will be able to see all records inserted (regardless of the language) while the translators will always see English text on the left and the translation one on the right.



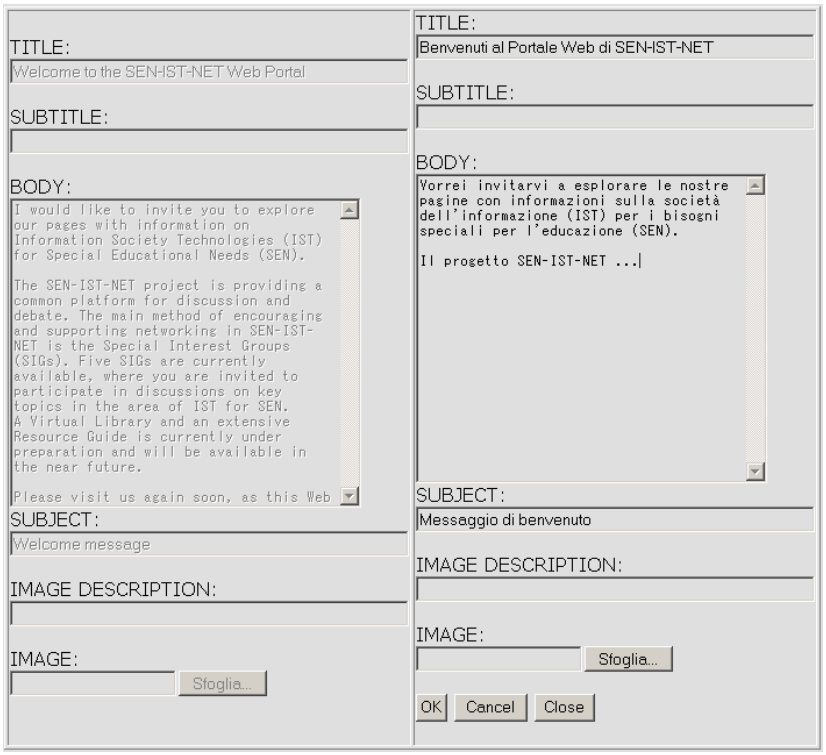
For the Master Editor, the update window has on the left side the original text (shaded as cannot be modified there) and on the right the window for modifying it.



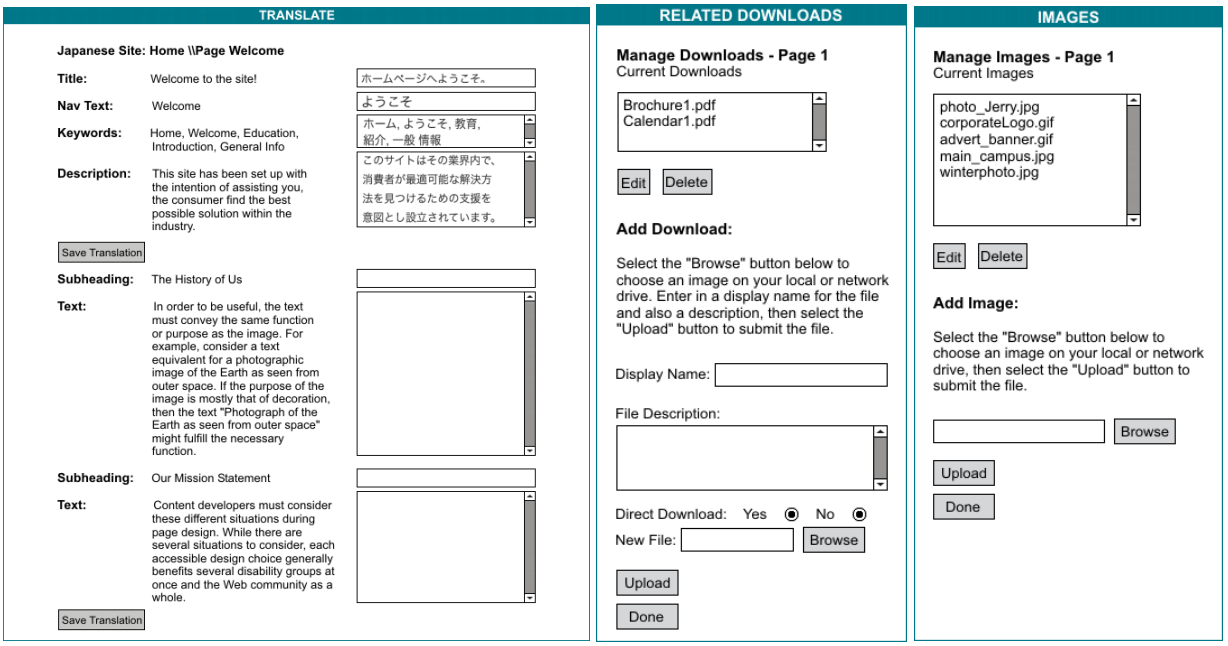
In the following picture is visible the main page for a generic translator. As already mentioned on the left side are reported the Master contents (in English), on the right the corresponding translation in the target language (in this case the target language would be automatically selected at login time given the design choice to map in a one to one ratio translators to languages). In the middle there are control buttons enabling the translator to access to content detail, translate it or update the already translated text. From the picture is evident another feature of this translation/localisation supporting tool as the set of available controls are dependent on the current status of the target language. In other words for contents that have just been inserted or not translated yet there will be a ***"Translation"*** control, while for contents already translated there will be an ***"Update"*** control.



So far we have pointed out the structure of the control and display interface of a translation support tool based on a simple database and a set of active web-pages. It is now necessary to enter in more detail in the translation related process. As already pointed out the database will have a master table for master content in English where each record corresponds to a specific content while each record field corresponds to a content component. Content components are the resources that will have to be translated. For each target language there is a target table in the database. The target table has exactly the same structure of the master content table. Records are initially inserted in the master table and indexed. Whenever a translation process is started a univocal association between current master record and target one is made, stored and properly indexed. This operation is automatically handled when the translation text is committed (stored by pressing the OK button on the translation insertion window). The aspect of the generic item translation window, is reported in the following picture, as usual on the left there is the original text and on the right the translated one.



What just stated is in essence what is done (in much more detailed and refined manner) by an m-CMS. This is already apparent from the following screenshots coming form an m-CMS.



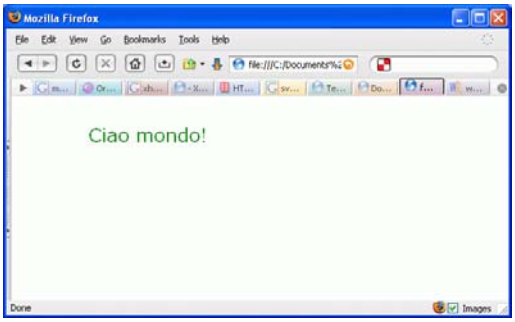
3.4 Images

Apart from the textual content that may appear on an image (an image could also be: a sales trend graphic, a market share pie representation…), images always present a relevant set of textual accompanying information. Some of this information is also embedded inside the image file format (JPEG header…). What we are referring here is the heterogeneous set of language related aspects that are implied by the specified kind of object. For example when translating a composite document, such as a poster, the disposition of the text is important. As already stated when dealing with text, a translated text is not necessarily as long as the original, it might be needed to adapt the layout or the styles, such as increasing or decreasing a font size. A similar problem exists with maps, where the tag over a location might need to be adjusted as a change in the width of the text can make two labels overlap. These are just examples of content that are usually composed

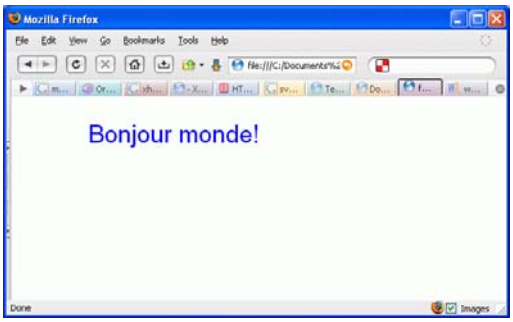
basically by images or transformed in images either for print or for usage as digital objects. As already stated this has also impact onto graphical user interfaces (GUI) as often there are portions exploiting graphic components holding text that is language dependent, like buttons. This is particularly relevant in the case of packed interfaces (for example PDAs or mobiles) where even a small change in dimensions of a specific object may alter the overall balanced aspect of the GUI. Last, but not least, frequently also in web-based applications is becoming common to have GIF objects used to achieve a better overall graphical result for the interface.
Even if to these problems there is no unique solution, what is advisable is to take into account such aspects at design time and organise graphical components of images / GUI to accommodate the most probable set of languages that may be addressed at distribution time (whenever possible). To this extent the previously pointed out solution based on a double level of templates could be an excellent approach as it will force a certain degree of rigidity in the graphical layout, but will accommodate the highest degree of flexibility in terms of content adaptation to language related needs (orientation, dimension…).
Furthermore, in some contexts a format such as SVG (Scalable Vector Graphics) may be considered when multi-lingual text must be provided within graphics. Through the conditional ***<switch>*** element and the use of the ***systemLanguage*** attribute multiple texts could be included in the same SVG. The switch will return the first child element for which the condition is true, thus care should be taken to set the primary language at the top. Also the event of and unsupported language can be easily considered and handled and, for example, a default text added.[6] The following snipplet is and example of the use of the ***<switch>*** element and of the ***systemLanguage*** attribute on the ***<text>*** element.

```
  <switch>
    <text systemLanguage="it" x="250" y="150" font-family="Verdana" font-size="60"
fill="green" >
            Ciao mondo!
    </text>
    <text systemLanguage="fr" x="250" y="150" font-family="Curier" font-size="80"
fill="blue" >
            Bonjour monde!
    </text>
    <text x="250" y="150" systemLanguage="en" font-family="Verdana" font-size="55"
fill="red" >
            Hello world!
    </text>
    <!--  The following is the default text which will be displayed if non of the above lan-
guages are detected-->
    <text x="250" y="150" font-family="Verdana" font-size="55" fill="red" >
            Hello world!
    </text>
  </switch>
```

In the above example is worth noting that different colors, different font types and sizes have been defined for different texts. Also the last ***<text>*** element (without the ***systemLanguage*** attribute) will be shown should the above languages not be detected. In the following picture what just explained in relation to the provided example is made graphically apparent.



Text displayed for "***it***"



Text displayed for "***fr***"

[6] For further details see the *Scalable Vector Graphics (SVG) 1.1 Specification* (http://www.w3.org/TR/SVG11/index.html#minitoc)

3.5 Audio / Video

We have already mentioned how relevant linguistic issues are for this kind of content and how subtle the difference could be between a good quality and a poor quality multi-lingual management for this kind of content. Nevertheless it is important to see how this issue has been, so far, tackled and as a starting point it is essential to place a clear distinction between audio and video content.

3.5.1 Audio

As far as audio is concerned, and considering the music market, we have to take into account the following issue: ***a song either is available in a specific language or is not***. For example there used to be the so-called "***cover***" versions of songs having had a relevant success. This phenomenon was quite common during the '50s and '60s and still applies to some countries (Italy, Spain, South America, China…) both with a localised version of the song or with a re-made one. Just to give some examples of the various possibilities is interesting to take into account that during their long staying in Hamburg, The Beatles, made a German version of Michelle, or that the Italian group Dick Dick made an Italian version of California Dreaming (Sognando California), or singers like Nec, Lura Pausini, Eros Ramazzotti, Rita Pavone, Raffaella Carra and others that have made Spanish versions of their best products. On the opposite side there is "*opera*" where usually no matter where it is performed, it is always performed in the original language (Italian, French or German) even if the public can access to both original and translated version of the text. As far as re-make of songs is concerned is sufficient to think to "Summer in the city" that has been sung at least from Lovin' Spoonfool, Joe Cocker, Joe Jackson each with his own stile and finally by Nelly in an Hip Hop version.
In any case there are other 'non-musical' (possibly multi-lingual) contexts where audio can play an important role that must also be considered.

- Public services like information services, ticket machines etc, where (also for the sake of accessibility) there is and audio recording; traditionally if the interface is multilingual, the first step is usually the language selection. The porting of some of these services to the Internet, or (for example on-the-spot information services) to portable devices could benefit from multi-lingual metadata and automatic language set-up.
- Museums and exhibitions "audio-guides" are another example. Many cultural contexts, now a day, offer such devices to the public. In this case each device is equipped with a certain set of languages the user may chose. If the audio guides were to be replaced (or joined) by portable devices such a as PDAs, the spoken audio content should be multilingual and the language clearly identified. In a similar way museum kiosk computers must provide multilingual content, again linguistically meaningful audio content must follow such criteria as well. In both cases it isn't uncommon to find both music and speech in many museum guides or kiosk where snippets of mixed content are often presented in a 'documentary' fashion. In these cases the music is to be considered linguistically 'neutral' while the speech must be clearly marked.
- Educational content, where a big portion of content involving audio, is usually video based (see below), but it is not infrequent to find presentation-like content with graphics/text and a commenting voice. Examples can be seen in many educational CD-ROMs where some sections (usually triggered by the user) present such content. In this case probably the most straightforward solution would be to deliver directly localized versions of the product. If similar content were to be delivered over the internet, the possibility to include different multilingual audio should be considered.
- Internet radio is now spreading, thanks to broadband, yet this makes new issues emerge. Spoken parts of programmes may be relevant in particular for news, sports… therefore, the possibility of delivering muli-language streams could be very valuable. The opportunity for mixed audio content could be considered as well, for example a user subscribing to an internet radio service could be delivered only streams in own language or only for certain content (e.g. speeches)

All the above examples, of course, rely on the assumption that specific budget have been (or will) spent on the production of the multilingual audio content. As already said, because translation and production of multilingual content (especially audio content) can be quite resource-demanding, this could be happening in a certain lapse of time, therefore both the framework and content should be 'multilingual-aware' so that the translations could be 'filled in' whenever ready or as part of an adaptation process for an acquired content.

3.5.2 Video

As far as video is concerned the issue is far more complex. In this case there are a set of possible solutions, spanning from dubbing up to subtitling. One could try to limit the burden by providing several audio tracks

to allow the user to select the desired one, this would be a very coarse solution as in case of absence (or poor quality) of synchronisation, the result may be quite distant from the user expectation, orn in some cases even disturbing or annoying.
To this extent it is worth taking into account the huge amount of money spent in dubbing audio-visual products for the mass market (both as far as TV and movies are concerned). A clear example of this are the TV-Fictions produced in the US (like Star Trek, Dinasty, Dallas…) that would be need to be dubbed/subtitled before being broadcasted in Europe. Initially this process would require years, now it has been squeezed to a few months. Moreover when dubbing such products it often happens that even the title is changed/adapt to overcome issues related to a mere translation process that could lead to an insignificant or little appealing title. Of course this issue (synchronization) is highly dependent on the kind of content presented: in some cases (like documentaries, educational content, promotional etc.) where a comment voice is usually 'off', synchronization may not be as much of a problem, in fact in such a context the informative capability of the audio may be the priority.
A totally different issue is the one related to the metadata associated to this kind of content as in this case it does not matter whether the content is audio or visual, it is just a matter of which languages are supported for the metadata management and how the info are stored. In this latter case (metadata) all that has been said for text and/or metadata applies.

3.6 Animations & Multimedia

This kind of content is basically a combination of the previously mentioned ones and therefore presents the combination of all problems reported so far. At the same time this kind of content has an advantage in respect to the previously mentioned ones, namely they can benefit of all achievement of synthetic graphics that span from the possibility to automatically tune lips-sync up to enable things that are unfeasible to humans. Moreover the product implicit digital nature, further enhance the chances to achieve real and effective multilingual support especially when adopting a multi-level templating approach as previously exposed.

3.7 Multi-language management and programming

In this paragraph we will just provide a tiny example of how the issue is tackled in the programming environment, where practices and policies are so different form place-to-place and even affected by application domain. The best example is the one of ERPs where BIPCS used to be the most prominent product (on the US market) but could not stand the comparison with SAP in Europe. This was basically due to a factor: despite BIPCS was localised in terms of linguistic support, it was not up to the level in terms of complexity management. What is intended here is that in Europe (for example in Italy) the complexity of the legal and administrative scenario was at least an order of magnitude more than in the US. Companies like SAP being based in Europe and having to deal with intra/extra border issues for goods exchange, different currencies and taxation systems even within the same country (e.g. old and new shillings or east/west deutsche mark…) were more suitable to accommodate for a wider and deeper set of customisation (at in this case localisation would basically comprise a good set of customisation). In other environments this was not the case (think to MatLab) and therefore localisation would be limited to the GUI. This process has to be carefully planned in any case to avoid the risk of translating also "***keywords***" or other components of the programming environment potentially leading to incompatibility among products of the same family but representing localised instances; just look to what happened in a certain period of time with Excel where functions names were localised along with the interface leading to the strange situation where the function ***SUM()*** of the USA/UK version would become ***SOMMA()*** in the Italian one, with all consequent problems of interoperability.

3.7.1 wxWidgets

In order to support multi lingual for the software designed, internationalization in *wxWidgets* has been investigated since *wxWidgets* is being utilised for various key components of the framework. A separate program called *gettext* needs to be installed in order to generate the language catalogues. A GUI wrapper for *gettext* called *poEdit* is available if preferred. The process is as follows:

- All translatable strings should be wrapped in a *_()* macro, e.g. *wxString s = _("this is a string")*;
- All menu items, labels, dialog messages, etc. should be converted.
- All non-translatable strings should be wrapped in a *_T()* macro so that character width conversion can be handled appropriately depending on whether Unicode is used on the system.
- *gettext* should be run on your source to generate a *.po* file containing all the translatable strings.
- These strings need manually translating.

- Finally, a catalogue file (*.mo*) can be generated for each supported language.
- The Locale can be set in the system initialization method to the system locale using *wxLANGUAGE_DEFAULT* and the correct translation of the strings will be used where available.

What just stated is based on the fact that *wxWidgets* uses a certain number of user-readable strings such as "*help*" or "*Load file*" which should be translated to the users language if it is different from English. *wxWidgets* has built in support for internationalization (*i18n*) which allows for this to happen automatically if the translations to the current language are available. Hereafter is reported the list of all existing translations. The table provides both the data of previous version and the actual ones. Columns names are quite self-explicative, yet is better to specify that in each row is reported the language and the status of its translations in the current CVS version of *wxWidgets* (take into account that data are regenerated once a day, therefore this table should be taken just as a trend reference). Languages that have had a significant evolution in the period are in bold while completed ones are highlighted.

Language	Oct-05			Sep-06		
	Translated	Fuzzy	Un-translated	Translated	Fuzzy	Un-translated
Afrikaans (af)	75.90%	12.80%	11.20%	75.90%	12.80%	11.20%
Albanian (sq)	**0.00%**	**0.00%**	**100.00%**	**96.70%**	**0.40%**	**3.00%**
Basque (eu)	14.00%	6.80%	79.20%	14.00%	6.80%	79.20%
Catalan (ca)	65.10%	19.00%	15.80%	65.10%	19.00%	15.80%
Chinese (zh_CN)	**89.20%**	**7.10%**	**3.70%**	**99.50%**	**0.50%**	**0.00%**
Chinese (zh_TW)	**74.80%**	**13.20%**	**11.90%**	**100.00%**	**0.00%**	**0.00%**
Czech (cs)	68.00%	15.10%	16.90%	68.00%	15.10%	16.90%
Danish (da)	**73.00%**	**18.70%**	**8.30%**	**83.40%**	**12.10%**	**4.50%**
Dutch (nl)	64.50%	19.40%	16.10%	64.50%	19.40%	16.10%
Finnish (fi)	71.10%	14.60%	14.40%	71.10%	14.60%	14.40%
French (fr)	100.00%	0.00%	0.00%	100.00%	0.00%	0.00%
Galician (gl_ES)	**0.00%**	**0.00%**	**100.00%**	**57.30%**	**41.90%**	**0.80%**
German (de)	**84.40%**	**8.40%**	**7.30%**	**100.00%**	**0.00%**	**0.00%**
Greek (el)	88.20%	5.80%	6.00%	88.20%	5.80%	6.00%
Hindi (hi)	**75.50%**	**13.80%**	**10.70%**	**75.50%**	**13.80%**	**10.70%**
Hungarian (hu)	**92.20%**	**4.60%**	**3.20%**	**100.00%**	**0.00%**	**0.00%**
Indonesian (id)	**76.90%**	**11.80%**	**11.30%**	**76.90%**	**11.80%**	**11.30%**
Italian (it)	100.00%	0.00%	0.00%	100.00%	0.00%	0.00%
Japanese (ja)	**89.10%**	**6.00%**	**4.90%**	**100.00%**	**0.00%**	**0.00%**
Latvian (lv)	19.10%	6.00%	74.80%	19.10%	6.00%	74.80%
Norwegian Bokmal (nb)	92.20%	4.60%	3.20%	92.20%	4.60%	3.20%
Polish (pl)	100.00%	0.00%	0.00%	100.00%	0.00%	0.00%
Portuguese (pt_BR)	87.70%	11.90%	0.40%	87.70%	11.90%	0.40%
Russian (ru)	92.20%	4.60%	3.20%	92.20%	4.60%	3.20%
Slovenian (sl)	100.00%	0.00%	0.00%	100.00%	0.00%	0.00%
Spanish (es)	**89.70%**	**5.50%**	**4.90%**	**98.50%**	**1.50%**	**0.00%**
Swedish (sv)	100.00%	0.00%	0.00%	100.00%	0.00%	0.00%
Turkish (tr)	92.20%	4.60%	3.20%	92.20%	4.60%	3.20%
Ukrainian (uk)	58.20%	24.90%	16.90%	58.20%	24.90%	16.90%

As already stated *wxWidgets* uses the standard GNU *gettext* tools for *i18n* therefore it is possible to contribute to work completion. Here are the steps that should be followed to get the most current version and contribute:

1. Get the latest version of the file locale/wxstd.po from the wxWidgets source tree; otherwise retrieve it directly from the cvs repository via the Web interface.
2. Rename it to ll_CC.po where "ll" is the 2 letter ISO 639-1 language code for your language and "CC" is the 2 letter ISO 3166 country code. If the country is the default one for this language (e.g. Italy is default country for Italian), then the country part and underscore preceding it should be omitted -- this is, in fact, the most common case.
3. Translate the strings in this file using either your favourite text editor or a specialized tool.

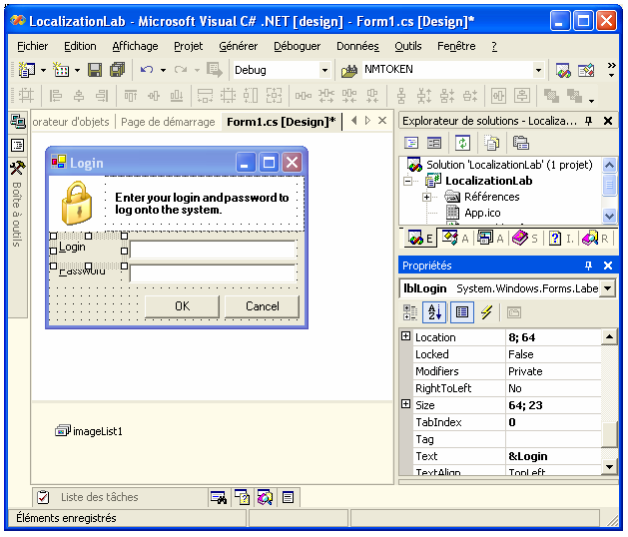
4. Verify that your translations can at least be compiled (even if they are yet incomplete) by running msgfmt -v ll_CC.po command: please note that you *must* use the -v option. In particular, please fill the header fields because msgfmt doesn't accept the default values for them.
5. Send the finished translation that will be added to the next wxWidgets release or snapshot.

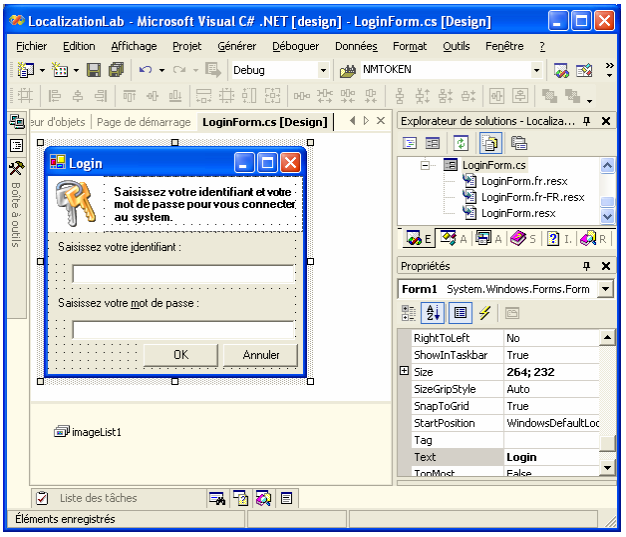
3.7.2 MFC Microsoft Foundation Class

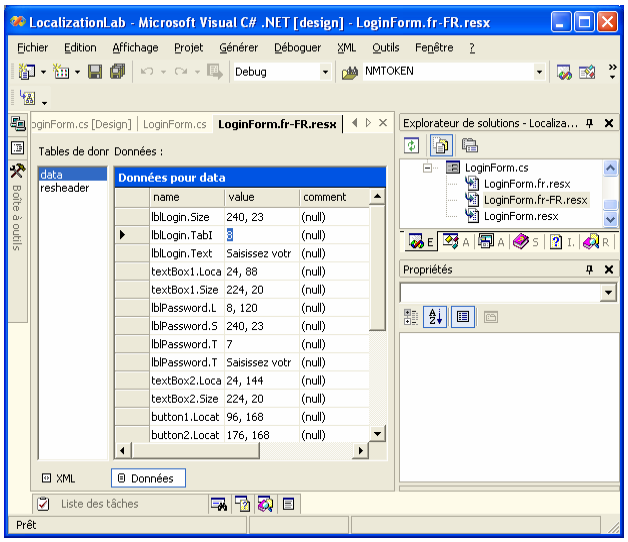
The main problem of using dictionary to translate the user interface is that it lacks flexibility, because sometimes the form should be completely reorganized to meet the target language constrains. This is why, in the MFC, Microsoft used “resource DLL” to handle localization. A resource DLL only contains the resource of an application, like string tables, error messages, but also the complete layout of a dialog box. Only the resource of the language of the user is loaded in memory and, as the identifiers of the resource are language independent, the application just have, for instance, to load the dialog with the id IDD_ABOUTBOX, and the English version will be loaded for the English, French version for the French.

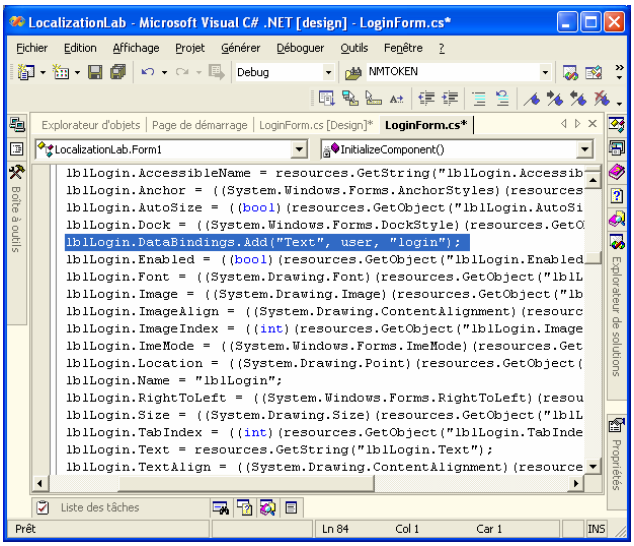
3.7.3 Microsoft *.Net*

In ***.Net***, this approach as evolved a little. First of all, there is a fallback mechanism, so that, for instance, if the resource for de-CH is not found, then the system search in the de resources, and if it does not found it either, it searches in the neutral language. Also, when you design a form, you can design it for several cultures. In the designer of the forms as also evolved so that some properties can be localized, but not all of them. The form presented is designed in the neutral language as apparent from the first picture. When translating it in another language, the picture may change, the labels might be widen and reposition. The accelerator keys also changed.









In the solution explorer panel, you see that the ***LoginForm*** has several resource files, one for the neutral language, one for the language (fr) and also, if needed, one for a sublanguage (fr-FR). The resource file contains values for the all the localizable properties of the form and the controls. But, as you may see below, not all

the properties are localizable. For instance, the ***DataBinding*** property, used to bind a control and the property of an object, is not used for layout but for managing the form. This approach to handle multilingual user interface is very flexible because it is possible to redesign the form, an issue that can be important when dealing with right to left text or top to bottom text. But the drawback is that it may make the translation process more difficult if the editor is not good enough to leverage the difficulty.

3.8 Multilingual Content Management Systems (m-CMS)

As already stated, m-CMS represent the most complete solution to most of the previously mentioned issues (both in terms of content production and support tools), but nevertheless they are still presenting also a non-marginal drawback, namely the cost.

Like for a workflow tool, each content management system needs to be customized to the client's individual needs and specifications. This need is based on the fact that exactly like in the workflow tool case it is not enough to acquire the tool but it is necessary to model the company set of activities, related actors and mutual interactions; for a content management tool is needed to model the content, its components and the related storage. It is necessary to clarify that most CMS tool do embed a set of features related to Content workflow management (either minimal or not) that will have to be used to handle the modeling of at least the content production phase and/or the subsequent content life cycle phases including functionalities related to versioning that, once again, may be developed to a different extent – form a very minimum to a CVS like. For this reason the set-up and configuration of a CMS/m-CMS is going to be a long and complex process that will certainly add additional expenses to the pure tool acquisition one (even when ignoring maintenance and future upgrading costs).

In most of the case CMS are used to support company content production for marketing and customer support services. This is certainly not the only usage for CMS, but is often the more frequent; m-CMS (multilingual CMS) do not differ much in this from CMS, except for the fact that they address directly the coordination issues encountered when trying to maintain consistency of content across multiple media and languages. As a matter of facts m-CMS were born to deal with issues of organizations where multilingual marketing is part of global marketing needs or distributed production and support causing different teams and workgroups to operate together on a 7 days a week 24 hours a day operation coverage, has nowadays become a reality that spans several business and market domains.

Having clarified these points it is worth now to examine some of the processes that will greatly benefit from the introduction of a CMS/m-CMS into a company daily operation especially as far as the content translation/localization is concerned.

Activity/Process	Rationale	Related issues	Possible solutions
Content re-use	Where the same content is used across multiple media there is a waste of resources if it is re-translated instead of recycled	• Reduce cost by removing skill set required of translator • Recycle content translations across multiple media (web, print. etc.)	• A master language is set in the CMS • All screens can then be translated within the CMS from anywhere, by anyone with Internet access, with no special HTML knowledge. • Where unique content is required for specific language/market(s), a master in the default language is created first, guaranteeing a central, monolingual master catalogue of all materials. • This default language version can be disabled to prevent viewing from outside the administration system, presenting only the translated unique content to the target market
Content consistency management	When working across multiple languages and media it is a coordination challenge to ensure that all media remain consistent with one another, and updates are applied universally and in a timely fashion across all media and languages	• To provide a system whereby multiple media releases can be linked so to be managed consistently • When content is updated/altered administrator is alerted • Same/similar content to be updated is automatically pointed out for management	• Set the master platform; this means the one that is always updated first as procedure. • Where content is re-used in alternative media, link specific blocks of content to specific media. • As master is updated, administrator is alerted to where same/similar content exists. • Updates to other media can be applied immediately or catalogued for batch updating at a later date

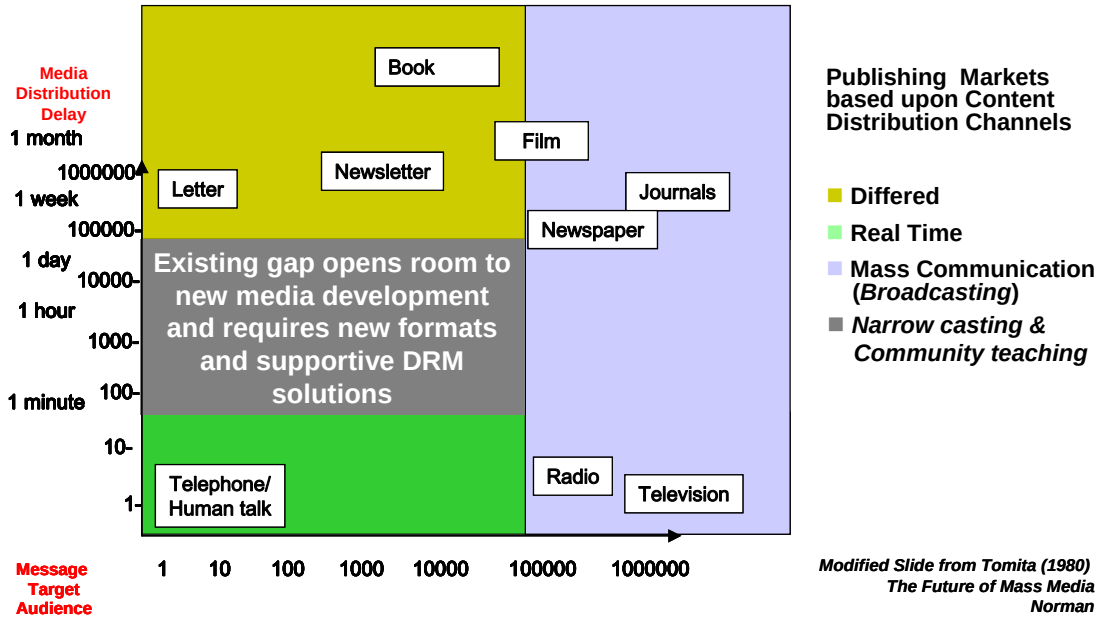
Activity/Process	Rationale	Related issues	Possible solutions
Content updating and maintenance	Need to maintain, update and expand content is a must, but related operations could be long and error prone impacting on time to market and costs	• Content should be easy to update • Content creator enabled to be content loader	• Create CMS that operates very much like a word processor, with NO HTML knowledge required • Allow content creation source to edit website directly
Search optimization	Content search is a constant activity in content production, maintenance, marketing and distribution, therefore search optimization is a must to save time and costs	• Fully optimizable search	• Manually editable title, description and keywords meta statements • Manually editable URLs (insert keywords to URL) • Create multiple landing pages for same product/service
Time to market	In an online world, time is crucial (both in terms of time to market and in terms of up-time) as Users expect immediate results	• Timeliness of operation • Development duration • Localization duration • Maintenance/Updating	• Exploiting and properly configuring CMS implicit workflow support to manage communication among actors and their proper synchronization • Exploiting the CMS implicit CVS to ensure content consistency and up-to-date status • Usage of powerful, optimized search tools • Usage of simple non-specialistic editing tools

4 Media & multi-language support

This is in essence a very broad field and a whole collection of books will hardly cover it if one would like to enter in the detail of how media do handle different languages. This is mainly due to the fact that language management within media is closely related to aspects like local culture, habits, interests, trends, fashion, local variances and daily jargon. Furthermore each media has its own "language"; saying this we refer to the communication aspect of the media. Each media exploits different aspects of the human psychology to achieve an effective communication process. Furthermore each actor in the field (either TV, radio, movie maker, publisher…) tries to convey also a "branding" message thanks to some aspects of the produced content. Nevertheless, it is possible to identify the major categories of approaches related either to time aspects of the distribution or to covered audience as reported in the following tables and summarised in the following diagram.

Distribution in time	Media	Followed approach
Differed	Books, newspapers, magazines	Different editions (one per language) Co-editions (one per language) Co-editions (multi-language)
	Letters, newsletters,	Different editions (one per language) Single edition (multi-language)
	Movies, TV series, Advertisement	Dubbing Subtitling
Real time	Telephone	Different versions (one per language)
	Radio	Different editions (one per language) Real time dubbing
	TV news…	Different editions (one per language)
	e-mails, SMS, MMS	Different editions (one per language)

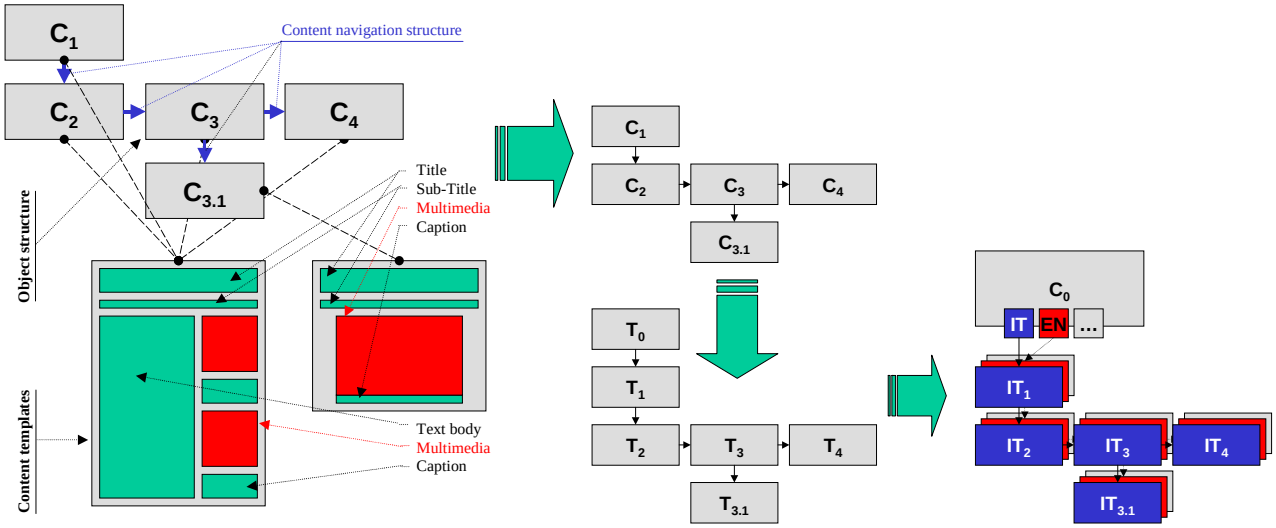
Distribution coverage	Media	Followed approach
Mass communication (Broadcasting)	Books, newspapers, magazines	Different editions (one per language) Co-editions (one per language) Co-editions (multi-language)
	Movies, TV series, Advertisement	Dubbing Subtitling
	Radio	Different editions (one per language) Real time dubbing
	TV news…	Different editions (one per language)
Narrowcasting	Letters, newsletters,	Different editions (one per language) Single edition (multi-language)
	e-mails, SMS, MMS	Different editions (one per language)
	Telephone	Different versions (one per language)



Whenever content is available in several language but formatted according to the principle ***"one language, one object"*** there are also additional way to deal with the content fruition in respect of user's language preferences. Basically the idea is that the user will be allowed to define an ordered list of preferred languages, so to ensure that it will be possible to end up accessing content in a secondary language when some content would not available in the favourite one. For instance in case the user prefers reading in French, but does not mind reading in English, such two languages could be selected as primary and secondary preferred for content and/or GUI. This approach could provide the system with useful information to exploit whenever content and/or components are missing in some specific language while they are available in other.
Actually, we have to say that this is a rather good solution especially whenever an on-line approach is followed or whenever content is retrieved thanks to some sort of query mechanism. At the same time, it is necessary to point out that this solution is rather demanding on the content administration side and could be easily pursued when exploiting a m-CMS, but rather hard in all other cases.

4.1 Authoring, Packaging & multi-language support

In this section we will focus on how content owner and authors usually deal with the multi-lingual issue mainly at authoring and production level taking as example what is presently achievable in multimedia e-learning. The rationale for such choice is that in such domain authors have to deal with complex objects, embedding several media (text, audio/visual, animation…), structured in complex manners and usually tagged and indexed to enable LCMS based management. Moreover contents (as well as related metadata) are often in several languages, and therefore proper solutions for addressing the issue have been not only defined, but also standardised. Having stated this is worth examining how the process could be achieved in an effective way, where to act and how so to address issues related to multi-language management in the process.
Creation and usage of the same content from different groups of people that may also have different sets of languages has been traditionally approached by publishers through co-editions in which publishers belonging to the different linguistic environment would edit the same content in their own language adapting images and any other multimedia content to the needs, resulting in a set of objects differing basically in language and "format" but not in "content". Actually a slightly different approach would lead to the production of a single object now holding the different languages (and related formats) within a single structure. In essence the approach is the same previously exposed when dealing with text in the context of translation/localization, technology and workflow interaction (a related schema is reported later on). The process would then be implemented as follows: a set of templates will be designed for each content component ensuring full respect of a functional based design approach. Then a master template will be designed to handle the content structuring and language management related issues. Once achieved this, content could be edited in a master form (comprising the two templates and all related resources) and then stored. The stored resources could be easily localized and saved. Similarly a localized version of the localized versions of the content component functional template should be either produced or retrieved (depending on the situation). Then the content structural template should be localized (if needed) and the newly produced/localized language-dependent resources (including related templates style-sheets…) could be linked together.



Generic content structure (based on components functional templates), derived structural template and multi-language instance

In case content is described via XML (as is the case of a IMS content package) then the abovementioned process could be also automated and achieved directly operating on the original content adding sections prepared (this means populated with the proper references to localized resources) starting from empty ones. In the following example we can see how this could be achieved in terms of XML coding. The starting point will be a section like the following one:

```
<Page xp:deliverytype="SCORM" xlink:label="L5415" xp:description="S1" id="Page01">
    <SubPages xp:deliverytype="SCORM" xlink:label="L5416" xp:description="Introduction" kind="R">
        <SubPage xp:deliverytype="SCORM" xlink:label="L5417" xp:description="Introduction_text">
            <ElementGroup xp:deliverytype="SCORM" xlink:label="L5418" xp:description="introduction_text_en" language="en">
                <ElementGroup xp:deliverytype="SCORM" xlink:label="L5419" xp:description="Introduction_text_a" type="Tourist">
                    <Text xlink:label="L5420">Welcome. In this section you will discover…</Text>
                    <audio src="L06_Madonna_Pomegranate_high/L06_Madonna_Pomegranate_high_Page01_en_tourist.wav"/>
                </ElementGroup>
            </ElementGroup>
        </SubPage>
    </SubPages>
</Page>
```

To add a new language version is enough to add an element with the same structure but different language identification and updated set of links to localized resources, this will result in what follows.

```
<Page xp:deliverytype="SCORM" xlink:label="L5415" xp:description="S1" id="Page01">
    <SubPages xp:deliverytype="SCORM" xlink:label="L5416" xp:description="Introduction" kind="R">
        <SubPage xp:deliverytype="SCORM" xlink:label="L5417" xp:description="Introduction_text">
            <ElementGroup xp:deliverytype="SCORM" xlink:label="L5418" xp:description="introduction_text_en" language="en">
                <ElementGroup xp:deliverytype="SCORM" xlink:label="L5419" xp:description="Introduction_text_a" type="Tourist">
                    <Text xlink:label="L5420">Welcome. In this section you will discover …</Text>
                    <audio src="L06_Madonna_Pomegranate_high/L06_Madonna_Pomegranate_high_Page01_en_tourist.wav"/>
                </ElementGroup>
            </ElementGroup>
            <ElementGroup xp:deliverytype="SCORM" xlink:label="L5425" xp:description="introduction_text_it" language="it">
                <ElementGroup xp:deliverytype="SCORM" xlink:label="L5426" xp:description="Introduction_text_a" type="Tourist">
                    <Text xlink:label="L5427">Benvenuti. In questa sezione scoprirete …<br/>
                    </Text>
                    <audio src="L06_Madonna_Pomegranate_high/L06_Madonna_Pomegranate_high_Page01_it_tourist.wav"/>
                </ElementGroup>
            </ElementGroup>
        </SubPage>
    </SubPages>
</Page>
```

Formats and aspects issues could also be dealt at this level as it is apparent form the following example (actually the same already used) in which also the formatting area is reported.

```
<?xml version="1.0" encoding="utf-8"?>
<!-- XML file generated by eXact Packager -->
<!-- eXact Packager licensed to: monbile2.giuntilabs.com -->
```

```
<LOMobile xp:deliverytype="SCORM" xlink:label="L5414" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xp="http://www.giuntilabs.com/exact/xp_v1d0" href="LOMobile.html" xp:description="L06_Madonna_pomegranate" Tracking="AICC-
HACP" id="L06_Madonna_pomegranate">
    <style xmlns="http://www.giuntilabs.com/exact/xp_v1d0">
        <colors c1="#000000" c2="#FFFFFF" c3="#D4D4D6" c4="#005FAE" c5="#003060" c6="#01D4FA"/>
        <linkcolors normal="#0000FF" visited="#800080" active="#FF0000"/>
        <tracking>AICC-HACP</tracking>
        <pagesize width="760" height="640" type="1"/>
        <textstyles>
            <textstyle type="Title">
                <normal fontface="Arial" fontsize="28" color="#000000" bold="false" italic="false" underline="false"/>
                <emph fontface="Arial" fontsize="28" color="#000000" bold="false" italic="true" underline="false"/>
                <moreemph fontface="Arial" fontsize="28" color="#000000" bold="true" italic="false" underline="false"/>
            </textstyle>
…
        </textstyles>
    </style>
    <Page xp:deliverytype="SCORM" xlink:label="L5415" xp:description="S1" id="Page01">
        <SubPages xp:deliverytype="SCORM" xlink:label="L5416" xp:description="Introduction" kind="R">
            <SubPage xp:deliverytype="SCORM" xlink:label="L5417" xp:description="Introduction_text">
                <ElementGroup xp:deliverytype="SCORM" xlink:label="L5418" xp:description="introduction_text_en" language="en">
                    <ElementGroup xp:deliverytype="SCORM" xlink:label="L5419" xp:description="Introduction_text_a" type="Tourist">
                        <Text xlink:label="L5420">Welcome. In this section you will discover …</Text>
                        <audio src="L06_Madonna_Pomegranate_high/L06_Madonna_Pomegranate_high_Page01_en_tourist.wav"/>
                    </ElementGroup>
…
                </ElementGroup>
                <ElementGroup xp:deliverytype="SCORM" xlink:label="L5425" xp:description="introduction_text_it" language="it">
                    <ElementGroup xp:deliverytype="SCORM" xlink:label="L5426" xp:description="Introduction_text_a" type="Tourist">
                        <Text xlink:label="L5427">Benvenuti. In questa sezione scoprirete …<br/>
                        </Text>
                        <audio src="L06_Madonna_Pomegranate_high/L06_Madonna_Pomegranate_high_Page01_it_tourist.wav"/>
                    </ElementGroup>
                </ElementGroup>
            </SubPage>
        </SubPages>
    </Page>
</LOMobile>
<!-- End of XML file generated by eXact Packager  -->
```

4.2 Content Indexing

The term content indexing could be referred and understood differently by different users depending on their background. It is therefore worth providing a basic introduction to clarify what is intended here, which are the related impacts and issues coming form the need to put the indexing operation in context with multi-lingual content.

In most cases, and regardless of the context, the indexing process begins with a thorough exam of the content to be indexed (either reading the text, checking details of picture, an audio or a video). The purpose of this initial step is to identify the selection of terms to be indexed. This process could be performed manually or automatically (at least for text) and will end up with a list of terms that are relevant and characterize the content under exam. It may be also needed to proceed to the creation of subentries, where appropriate.

In respect of books, articles and other printed/printable contents, indexing is usually done by specialized librarians, archivists or freelancers hired by publishers or book packagers to catalogue content or index it or create reference. Yet there are a number of sub-specialties in indexing, such as web indexing (the application of a back-of-book-style index to a website or intranet), database indexing (the application of a pre-defined controlled vocabulary such as MeSH to articles for inclusion in a database), periodical indexing (indexing of newspapers, journals, magazines).

From what just stated is apparent that the process could either be handled manually or thanks to purposely devised software systems. In the first case the process is highly human dependent and therefore could be “error-prone” in the sense that two different indexers will very much probably end up with different results when operating on the same content (especially if it is a complex one).

In the second case the software system has to rely on a set of tools (including but not limited to taxonomies) and therefore the quality of the result will be highly dependent on a combination of factors like the quality of the analyzed source, the quality and completeness of the used taxonomy, the accuracy and completeness of the analysis algorithm(s) implemented plus the richness of combined methods used to perform the job (so to ensure higher degree of reliability and overall consistency).

The same applies to metadata, and in particular to classification keywords. For example while certain items could be easily and univocally identified (author, title, publisher, date of production, copyright…) other may be highly dependent on / biased by the indexer/archivist choice (subject, theme…).

4.2.1 Taxonomies and multi-language

The term taxonomy originates form Greek and has been originally used to identify only the science of classifying living organisms (alpha taxonomy), but later was applied in a much wider sense, and therefore now may also refer to either a classification of things, or the principles underlying the classification. Frequently hierarchical in structure, with parent child relationships, might also be a simple organization of objects into groups, or even an alphabetical list. In current usage within "Knowledge Management" and "Computer Science", taxonomies are seen as slightly less broad than ontologies, which are data models that represent a domain and are used to operate on objects in that domain and the relations among them. Both taxonomies and ontologies are used in artificial intelligence, semantic web, software engineering and information architecture as a form of knowledge representation about the world (or some part of it) able to generally describe:

- ***Individuals***: the basic or "ground level" objects
- ***Classes***: sets, collections, or types of objects
- ***Attributes***: properties, features, characteristics, or parameters that objects can have and share
- ***Relations***: ways that objects can be related to one another

The most commonly used formats for describing taxonomies/ontologies are VDEX and RDF. The first is the acronym used to identify the IMS Vocabulary Definition Exchange (VDEX) specification; a grammar for controlled vocabularies. The second stands for Resource Description Framework (RDF), a W3C standard. Therefore is evident that RDF is suitable for a wider set of applications while VDEX is suitable for a more targeted usage; nevertheless (as both can be expressed in XML) it is feasible to apply criteria and tools for managing different languages at the same time as apparent from the following VDEX example.

RDF Example

```
<?xml version="1.0"?>
 <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
             xmlns:dc="http://purl.org/dc/elements/1.1/"
             xmlns:exterms="http://www.example.org/terms/">

   <rdf:Description rdf:about="http://www.example.org/index.html">
       <exterms:creation-date>August 16, 1999</exterms:creation-date>
   </rdf:Description>

   <rdf:Description rdf:about="http://www.example.org/index.html">
       <dc:language>en</dc:language>
   </rdf:Description>

   <rdf:Description rdf:about="http://www.example.org/index.html">
       <dc:creator rdf:resource="http://www.example.org/staffid/85740"/>
   </rdf:Description>

 </rdf:RDF>
```

VDEX Example

```
<?xml version="1.0" encoding="UTF-8" ?>
- <vdex orderSignificant="true" profileType="hierarchicalTokenTerms" language="en"
      xsi:schemaLocation="http://www.imsglobal.org/xsd/imsvdex_v1p0 imsvdex_v1p0.xsd"
      xmlns="http://www.imsglobal.org/xsd/imsvdex_v1p0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
      instance">
   - <vocabName>
       <langstring language="en">EDUCATIONAL Curriculum Map</langstring>
     </vocabName>
     <vocabIdentifier>http://www.giuntilabs.it/currmap.xml</vocabIdentifier>
   - <term>
       <termIdentifier>500</termIdentifier>
     - <caption>
         <langstring language="en">EDUCATION SYSTEM</langstring>
         <langstring language="it">SISTEMA EDUCATIVO</langstring>
       </caption>
     - <term>
         <termIdentifier>500.001</termIdentifier>
       - <caption>
           <langstring language="en">Primary Education</langstring>
           <langstring language="it">Educazione Elementare</langstring>
         </caption>
```

```
        - <term>
            <termIdentifier>500.001.001</termIdentifier>
          - <caption>
              <langstring language="en">English Language</langstring>
              <langstring language="it">Lingua Italiana</langstring>
            </caption>
          - <term>
              <termIdentifier>500.001.001.001</termIdentifier>
            - <caption>
                <langstring language="en">Grammar</langstring>
                <langstring language="it">Grammatica</langstring>
              </caption>
            </term>
          </term>
        ...
    </term>
  - <relationship>
      <sourceTerm>500.002.003.001</sourceTerm>
      <targetTerm>500.002.003.004</targetTerm>
      <relationshipType
          source="http://www.imsglobal.org/vocabularies/iso2788_relations.xml">RT</relationshipType>
    </relationship>
  - <relationship>
      <sourceTerm>500.002.004</sourceTerm>
      <targetTerm>500.002.006.001</targetTerm>
      <relationshipType
          source="http://www.imsglobal.org/vocabularies/iso2788_relations.xml">RT</relationshipType>
    </relationship>
</vdex>
```

4.3 Specifying languages

The language aspect is to be handled also in the metadata part of the content, luckily in that respect standards provide some support as they state that wherever it is necessary to specify a language such as in data element '***General.Language***' or in any language string the following coding can be used:

1. use a 2 letter code from ISO 639-1
2. use a 3 letter code from ISO 639-2 (see: http://www.loc.gov/standards/iso639-2/normtext.html, it does not matter between bibliographic & terminology since they only differ for languages that have 2-letter codes)
3. add the ISO Country code [ISO3166] when necessary, separated by a dash.
4. use IANA registered language tags, prefixed with **i-**
5. use SIL Ethnologue 3-letter codes, prefixed with **x-E-**
6. make up a name for token languages prefixed with **x-T-**
7. make up a name, prefixed with **x-** for user defined languages

All of the above are acceptable but partners should prefer 1, 2, or 3; in the following table some examples are provided:

Coding	Description	Type
nl	Dutch	ISO 639-1
aus	Australian Languages	ISO 639-2
IT	Italian	ISO 3166
i-xxxxx	IANA registered xxxxx	IANA
x-E-pcd	Picard	SIL Ethnologue
x-T-ELR	The ELR Token Language	SIL Ethnologue
x-none	Not possible to identify a language	SIL Ethnologue

It is worth mentioning that in several environments is currently being used the ***xml:lang*** attribute based on [IETF RFC 3066] (http://www.ietf.org/rfc/rfc3066.txt)

4.4 Licenses and rights

Even though this may seem a non-crucial problem, yet it is a very important issue to be tackled. Users will inevitably rely on localised versions of the license both for the linguistic aspect of the text expressing the rights and constraints implied and for the compliancy, respect and enforcement of local rules and laws.
We should differentiate here between contracts and licenses. The same would be needed for visual representation and storage format. Contracts have to be expressed in the language of the party offering the service,

content, etc. but licenses have to be expressed using standard Rights Expression Languages (like MPEG-21 REL or ODRL), that are related to rights data dictionaries mainly expressed in English terms, although other languages could be possible, and even relationships between terms in different languages. This latter point is due to the fact that licenses have to be handled also by applications.
What just stated could be basically summed up as follows: a localized version of a license should comprise the expression of rights and conditions available for that county, but using the standard rights expression language. On the other hand, the visual representation of the license could be translated, but only at GUI level, as the license format will be the same for any country and language.

5 Metadata & multi-language management

This document builds upon what already reported in the previous version and tries to provide a more practical approach to addressed issues while the original document was more focused on giving an overall picture. We have mentioned several times the relevance of metadata as far as multi-lingual issue management is concerned. So far we have primarily focused on content related aspects, now our attention will be moved to metadata itself. Therefore in the present section we will focus on underlying principles and major issues to be taken into account to ensure an effective metadata management in a multi-lingual environment regardless of the content nature, type and media.

5.1 Principles

Given the fact that MPEG21 object format is so flexible to accommodate a wide set of possible contents, both in terms of nature and formats, also related metadata could be defined with a high degree of freedom. To this extent AXMEDIS object structure has been defined so to accommodate a set of metadata that will be suitable to cover all possible fields of application of AXMEDIS framework. Therefore an AXMEDIS object will have, by default, its own metadata (specific to object management within the designed framework) as well as Dublincore ones; but it will also allow to host generic metadata loaded from several sources like XML files, MPEG21 Identifier(s) or Related identifier(s) or Type(s).
Given the fact that each kind of content (in terms of target market and audience) will be characterized by a different set of metadata and that in some markets it is common to have a specific set of metadata (LOM in education, Dublincore in museums and art related fields) it is necessary to see where superposition allows automatic replication/transferring of data and where it will be necessary to specifically insert relevant metadata. In practice we have:

- **Data common between AXInfo and LOM** – this data will be possibly automatically filled in at creation time.
- **Data common between Dublincore and LOM** – this data will be possibly automatically filled in at creation time.
- **Data specific to AXInfo** – this data will be filled in at authoring time.
- **Data specific to LOM** – this data should have been possibly filled in at authoring time.
- **Data specific to Dublincore** – this data will be possibly filled in at authoring time for classification and management purposes.

There are two types of element subsets to be taken into account: the elements that should be filled in every metadata instance (mandatory elements) and the elements that would be very useful to be filled (recommended elements).
In general the information model for metadata is based on element whose actual content is called a value. Values can be entered as free text, inserted in predefined format or selected from set lists, which are called vocabularies. Usually there are five data types in metadata information model, briefly:

- **CharacterString**: text can be entered in the element directly.
- **LangString**: the text must identify its language and there can be one or more character strings in the element.
- **DateTime**: the element contains date and time information and there can also be textual information about this point in time.
- **Duration**: the element contains information about an interval in time and there can also be textual information about the duration.
- **Vocabulary**: the element contains source and value where source is a reference to publicly sourced and maintained value set and value is a value from that set.

It is worth noting that using the **LangString** data type is possible to identify the language of the related metadata/or content, which implicitly opens up a clear scenario for multi-language management of metadata. Furthermore ISO-639 is included in the MPEG21 standards for: original primary spoken language and primary language of text element. Therefore it is wise to suggest that it should be applied the principle stated from W3C in respect to accessibility best practice in assigning language tags (within html) to the domain of MPEG21 objects, i.e. to identify the 'primary language' of spoken and written content wherever possible within an object. Furthermore, whenever recommended best practice is to use a value from a controlled vocabulary, an valuable suggestion for multi-language management is to localize the related vocabularies so to be able to localized also the management of fields referring to those vocabularies.
In the previous version of the document we reported the full list of metadata fields belonging to the mandatory and optional category both for Dublincore and IEEE-LOM. In the following table they are simply recalled. It is worth taking into account that we have considered (as far as Dublincore is concerned) as mandatory those fields that will allow an effective and efficient content characterization in terms of metadata spanning onto a wide set of possible usages, contexts and content domain.

Dublincore		**IEEE-LOM**	
Mandatory	**Optional**	**Mandatory**	**Optional**
contributor ***coverage*** ***creator*** ***date*** ***description*** ***format*** ***identifier*** ***language*** ***publisher*** ***relation*** ***rights*** ***source*** ***subject*** ***title*** ***type***	***abstract*** ***accessRights*** ***accrualMethod*** ***accrualPeriodicity*** ***accrualPolicy*** ***alternative*** ***audience*** ***available*** ***bibliographicCitation*** ***conformsTo*** ***created*** ***dateAccepted*** ***dateCopyrighted*** ***dateSubmitted*** ***educationLevel*** ***extent*** ***hasFormat.*** ***hasPart*** ***hasVersion*** ***instructionalMethod*** ***isFormatOf*** ***isPartOf*** ***isReferencedBy*** ***isReplacedBy*** ***isRequiredBy*** ***issued*** ***isVersionOf*** ***license*** ***mediator*** ***medium*** ***modified*** ***provenance*** ***references*** ***replaces*** ***requires*** ***rightsHolder*** ***spatial*** ***tableOfContents*** ***temporal*** ***valid***	***General.Identifier.*** ***General.Title.*** ***General.Language.*** ***General.Description.*** ***Technical.Location.*** ***Educational.Intended End User Role.*** ***Educational.Typical Age Range.*** ***Rights.Copyright and Other Restrictions.*** ***Rights.Description.*** ***Classification.Keyword.***	***General.Keyword***. ***General.Structure***. ***Life Cycle.Contribute***. ***Life Cycle.Contribute.Role***. ***Life Cycle.Contribute.Entity***. ***Life Cycle.Contribute.Date***. ***Meta-Metadata***. ***Meta-Metadata.Contribute.Role***. ***Meta-Metadata.Contribute.Entity***. ***Meta-Metadata.Date***. ***Meta-Metadata.Language***. ***Technical.Format***. ***Technical.Size***. ***Technical.Facet***. ***Technical.Facet.Value***. ***Educational.Learning Resource Type***. ***Educational.Learning Context***. ***Educational.Typical Learning Time***. ***Educational.Description***.

5.2 IPR related fields

Information about who can access the resource or an indication of its security status. Access Rights may include information regarding access or restrictions based on privacy, security or other regulations. Recommended best practice is to identify the license using a URI, the same applies to elements used to indicate the entity. Examples of such licenses can be found at http://creativecommons.org/licenses/. Given the specific

context is worth taking into account that MPEG21 offers a specific solution and support for this isse as detailed hereafter.

5.3 DRM related fields (MPEG21 set)

DRM information in the AXMEDIS project will be expressed by means of rights expressions (licenses), described with MPEG-21 Rights Expression Language (REL). Other RELs will be considered during the development of the project. The vocabulary used in these licenses for the description of actions is described in MPEG-21 Rights Data Dictionary (RDD). Terms on MPEG-21 RDD are written in English, but the content of the fields expressed inside licenses are not bounded to any specific language. For instance, the title of the license could be written in Spanish or Italian and it will still be the title element, as expressed in the MPEG-21 REL XML Schema.

```
<?xml version="1.0" encoding="UTF-8"?>
<r:license xmlns:r="urn:mpeg:mpeg21:2003:01-REL-R-NS" xmlns:sx="urn:mpeg:mpeg21:2003:01-REL-SX-NS"
xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS" xmlns:dsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2003:01-REL-R-NS
../schemas/rel-r.xsd
                    urn:mpeg:mpeg21:2003:01-REL-SX-NS ../schemas/rel-sx.xsd
                    urn:mpeg:mpeg21:2003:01-REL-MX-NS ../schemas/rel-mx.xsd">
    <r:grantGroup>
        <r:grant>
            <r:keyHolder>
                <r:info>
                    <dsig:KeyValue>
                        <dsig:RSAKeyValue>
                            <dsig:Modulus>KtdToQQyzA==</dsig:Modulus>
                            <dsig:Exponent>AQABAA==</dsig:Exponent>
                        </dsig:RSAKeyValue>
                    </dsig:KeyValue>
                </r:info>
            </r:keyHolder>
            <mx:play/>
            <r:digitalResource>
                <r:nonSecureIndirect URI="http://www.onlinemusic.com/mySong.mp3"/>
            </r:digitalResource>
            <sx:exerciseLimit>
                <r:serviceReference>
                    <sx:uddi>
                        <sx:serviceKey>
                            <sx:uuid>ee1398c0-8abe-11d7-a735-b8a03c50a862</sx:uuid>
                        </sx:serviceKey>
                    </sx:uddi>
                </r:serviceReference>
            </sx:exerciseLimit>
        </r:grant>
    </r:grantGroup>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyValue>
                    <dsig:RSAKeyValue>
                        <dsig:Modulus>X0j9q99yzA==</dsig:Modulus>
                        <dsig:Exponent>AQABAA==</dsig:Exponent>
                    </dsig:RSAKeyValue>
                </dsig:KeyValue>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license>
```

Example of a license expressed in MPEG-21 REL

As it can be seen in the previous sample license, XML elements are expressed in English language, like grant, ***exerciseLimit*** or play. Elements coming from other XML schemas, like the ones related to digital signatures (those starting with "***dsig***:") are also written in English.

The content of the elements can be expressed in any language, but there is no way to indicate which language is being used. For most of the cases, this does not represent a problem, as we deal with URIs or identifiers. For the case of information whose visual representation may change from country to country, like dates

or numbers, the application in charge of showing DRM information should solve this problem, not the rights expression language.

6 References & Bibliography (ALL)

In this sections are reported the more relevant reference and standards that have to be taken into account when selecting content either for re-editing / publishing or distribution purposes. The reader will find here quick reference information and links to more in dept info.

[W3C] Guidelines on Language Tags usage: http://www.w3.org/International/articles/language-tags/
[TGN] Thesaurus of Geographic Names http://www.getty.edu/research/tools/vocabulary/tgn/index.html
[W3CDTF] ISO 8601 date encoding http://www.w3.org/TR/NOTE-datetime
[MIME] Internet Media Types http://www.iana.org/assignments/media-types/
[RFC3066] Language specifications http://www.ietf.org/rfc/rfc3066.txt
[ISO639] Language specifications http://www.loc.gov/standards/iso639-2/
[DCMITYPE] DCMI Type Vocabulary http://dublincore.org/documents/dcmi-type-vocabulary/
[XML MAG] schema (v2.0) http://www.iccu.sbn.it/MAG/MAG_2.0/MAG_sito_Schema/mag_2_0_ec.html

[1] Adeline Yen Mah, “A thousand pieces of gold – A memoir of China’s past through its proverbs”, Chapter 16, pp 293 Harper Collins Publishers 2003
[2] “Language Review & Guidelines”, APLAWS project website, 2002, www.aplaws.org.uk/download_content/documents/translation_is_not_enough.pdf
[3] “Accessibility and Usability Review & Guidelines”, APLAWS project website, 2002, www.aplaws.org.uk/download_content/documents/usability_study.pdf
[4] “APLAWS Pathfinder Final Report”, APLAWS project website, 2002, http://www.aplaws.org.uk/download_content/documents/Final_APLAWS_Pathfinder_Report.doc
[5] “Centralizing The Localization Process”, Transware White papers, 2005, www.transware.com/services/
[6] “Developing Localization Friendly E-Learning”, TRANSWARE White papers, 2004, www.transware.com/services/
[7] “Multilingual E-Learning – Getting It Right From The Start”, TRANSWARE White papers, 2004, www.transware.com/services/
[8] “TRANSWARE’s Solutions For Audio Localization”, TRANSWARE White papers, 2004, www.transware.com/services/
[9] “Internationalization”, TRANSWARE White papers, 2005, www.transware.com/services/
[10] “Web Globalization Best Practice”, TRANSWARE White papers, 2005, www.transware.com/services/
[11] John Harris and Ryan McCormack, “Translation is not enough. Considerations for global Internet development”, Sapient Corporation, 2000, www.sapient.com
[12] Robin Lloyd, “Three Approaches to Multilingual Content Management - Using a Content Management System (CMS)”, KM World (Special Supplement), 2002
[13] Willem Stoeller, “The Hub of the Wheel”, PROJECT MANAGEMENT - The Guide from MultiLingual Computing & Technology - #63 Supplement April/May 2004
[14] Claire Ingram, “Working With the Client to Ensure a Successful Project”, PROJECT MANAGEMENT - The Guide from MultiLingual Computing & Technology - #63 Supplement April/May 2004
[15] Thomas Waßmer, “Four Translation Project Management Tools”, PROJECT MANAGEMENT - The Guide from MultiLingual Computing & Technology - #63 Supplement April/May 2004
[16] Kenneth McKethan, Jr., “Project Churn: Counting the Cost”, PROJECT MANAGEMENT - The Guide from MultiLingual Computing & Technology - #63 Supplement April/May 2004
[17] Jeff Allen, “Project Management Checklist”, PROJECT MANAGEMENT - The Guide from MultiLingual Computing & Technology - #63 Supplement April/May 2004
[18] “LPM Best Practices Survey”, PROJECT MANAGEMENT - The Guide from MultiLingual Computing & Technology - #63 Supplement April/May 2004
[19] MLC Research, “Best Practices Build Solid Projects”, PROJECT MANAGEMENT - The Guide from MultiLingual Computing & Technology - #63 Supplement April/May 2004

[20] Andreas Eisele (Saarland University & DFKI) , “Exploiting Multilingual Corpora for Machine Translation”, JRC Enlargement and Integration Workshop - Exploiting parallel corpora in up to 20 languages, Arona, September 2005

7 Terminology (ALL)

Term	Explanation (including source if available)
CMS	Content Management Systems
DCMIType	DCMI Type Vocabulary. A list of types used to categorize the nature or genre of the content of the resource
DDC	Dewey Decimal Classification. See: http://www.oclc.org/dewey/index.htm
IETF	Tags for the Identification of Languages see RFC 3066
IMT	The Internet media type of the resource. See: http://www.iana.org/assignments/media-types/
ISO3166	ISO 3166 Codes for the representation of names of countries. See: http://www.iso.org/iso/en/prods-services/iso3166ma/02iso-3166-code-lists/list-en1.html
ISO-639	International Standard naming system for languages. ISO-639 codes now include subtags to identify dialects such as lang=’en-GB’ indicates British-English content rather than US-English.
ISO639-2	ISO 639-2: Codes for the representation of names of languages. See: http://lcweb.loc.gov/standards/iso639-2/langhome.html
KMS	Knowledge Management Systems
LCC	Library of Congress Classification. See: http://lcweb.loc.gov/catdir/cpso/lcco/lcco.html
LCSH	Library of Congress Subject Headings
LCMS	Learning Content Management Systems
MESH	Medical Subject Headings. See: http://www.nlm.nih.gov/mesh/meshhome.html
NLM	National Library of Medicine Classification: See: http://wwwcf.nlm.nih.gov/class/
ODL	Open and Distance Learning
RFC1766	Internet RFC 1766 'Tags for the identification of Language' specifies a two letter code taken from ISO 639, followed optionally by a two letter country code taken from ISO 3166. See: http://www.ietf.org/rfc/rfc1766.txt
RFC3066	Internet RFC 3066 'Tags for the Identification of Languages' specifies a primary subtag which is a two-letter code taken from ISO 639 part 1 or a three-letter code taken from ISO 639 part 2, followed optionally by a two-letter country code taken from ISO 3166. When a language in ISO 639 has both a two-letter and three-letter code, use the two-letter code; when it has only a three-letter code, use the three-letter code. This RFC replaces RFC 1766. See: http://www.ietf.org/rfc/rfc3066.txt
TGN	The Getty Thesaurus of Geographic Names. See: http://www.getty.edu/research/tools/vocabulary/tgn/index.html
UDC	Universal Decimal Classification. See: http://www.udcc.org/
URI	A URI Uniform Resource Identifier. See: http://www.ietf.org/rfc/rfc2396.txt
W3CDTF	W3C Encoding rules for dates and times - a profile based on ISO 8601. See: http://www.w3.org/TR/NOTE-datetime

7.1 The DCMI Type Vocabulary

Term	Explanation (including source if available)
Collection	A collection is an aggregation of items. The term collection means that the resource is described as a group; its parts may be separately described and navigated.
Dataset	A dataset is information encoded in a defined structure (for example, lists, tables, and databases), intended to be useful for direct machine processing.

Term	Explanation (including source if available)
Event	An event is a non-persistent, time-based occurrence. Metadata for an event provides descriptive information that is the basis for discovery of the purpose, location, duration, responsible agents, and links to related events and resources. The resource of type event may not be retrievable if the described instantiation has expired or is yet to occur. Examples - exhibition, web-cast, conference, workshop, open-day, performance, battle, trial, wedding, tea-party, conflagration.
Image	An image is a primarily symbolic visual representation other than text. For example - images and photographs of physical objects, paintings, prints, drawings, other images and graphics, animations and moving pictures, film, diagrams, maps, musical notation. Note that image may include both electronic and physical representations. (Broader than: StillImage/MovingImage)
InteractiveResource	An interactive resource is a resource which requires interaction from the user to be understood, executed, or experienced. For example - forms on web pages, applets, multimedia learning objects, chat services, virtual reality.
MovingImage	A series of visual representations that, when shown in succession, impart an impression of motion. Examples of moving images are: animations, movies, television programs, videos, zoetropes, or visual output from a simulation. Instances of the type "Moving Image" must also be describable as instances of the broader type "Image".
PhysicalObject	An inanimate, three-dimensional object or substance. For example -- a computer, the great pyramid, a sculpture. Note that digital representations of, or surrogates for, these things should use Image, Text or one of the other types.
Service	A service is a system that provides one or more functions of value to the end-user. Examples include: a photocopying service, a banking service, an authentication service, interlibrary loans, a Z39.50 or Web server.
Software	Software is a computer program in source or compiled form which may be available for installation non-transiently on another machine. For software which exists only to create an interactive environment, use interactive instead.
Sound	A sound is a resource whose content is primarily intended to be rendered as audio. For example - a music playback file format, an audio compact disc, and recorded speech or sounds.
StillImage	A static visual representation. Examples of still images are: paintings, drawings, graphic designs, plans and maps. Recommended best practice is to assign the type "text" to images of textual materials. Instances of the type "Still Image" must also be describable as instances of the broader type "Image".
Text	A text is a resource whose content is primarily words for reading. For example - books, letters, dissertations, poems, newspapers, articles, archives of mailing lists. Note that facsimiles or images of texts are still of the genre text.