Translation the Wiki Way

Alain Désilets
National Research Council
of Canada
Montreal Rd, Ottawa, On,
K1A 0R6, Canada
1-613-990-2813

alain.desilets@
nrc-cnrc.gc.ca

Lucas Gonzalez
Fluwikie
www.fluwikie.com
+34 667638350

lucas.gonzalez@
canarias.org

Sébastien Paquet
Socialtext, Inc.
655 High St
Palo Alto, CA 94301
+1-617-379-0818

sebpaquet+wikisym@
gmail.com

Marta Stojanovic
National Research Council
of Canada
Montreal Rd, Ottawa, On,
K1A 0R6, Canada
1-613-990-2813

marta.stojanovic@
nrc-cnrc.gc.ca

ABSTRACT

This paper discusses the design and implementation of processes and tools to support the collaborative creation and maintenance of multilingual wiki content. A wiki is a website where a large number of participants are allowed to create and modify content using their Web browser. This simple concept has revolutionized collaborative authoring on the web, enabling among others, the creation of Wikipedia, the world's largest online encyclopedia. On many of the largest and highest profile wiki sites, content needs to be provided in more than one language. Yet, current wiki engines do not support the efficient creation and maintenance of such content. Consequently, most wiki sites deal with the issue of multilingualism by spawning a separate and independent site for each language. This approach leads to much wasted effort since the same content must be researched, tracked and written from scratch for every language. In this paper, we investigate what features could be implemented in wiki engines in order to deal more effectively with multilingual content. We look at how multilingual content is currently managed in more traditional industrial contexts, and show how this approach is not appropriate in a wiki world. We then describe the results of a User-Centered Design exercise performed to explore what a multilingual wiki engine should look like from the point of view of its various end users. We describe a partial implementation of those requirements in our own wiki engine (LizzyWiki), to deal with the special case of bilingual sites. We also discuss how this simple implementation could be extended to provide even more sophisticated features, and in particular, to support the general case of a site with more than two languages. Finally, even though the paper focuses primarily on multilingual content in a wiki context, we argue that translating in this "Wiki Way", may also be useful in some traditional industrial settings, as a way of dealing better with the fast and ever-changing nature of our modern internet world.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: User-centered design, Interaction styles, Natural language, Ergonomics; H.5.4 [**Hypertext/Hypermedia**]: Navigation, User issues; H.5.3 [**Group and Organization Interfaces**]: Computer-supported cooperative work, Web-based interaction

General Terms

Design, Languages, Human Factors.

Keywords

Multilingual wiki, Translation workflow, Collaborative Web-Authoring, User-centered design, Groupware, Hypertext.

1. INTRODUCTION

This paper discusses the design and implementation of processes and tools to support the collaborative creation and maintenance of multilingual wiki content.

Wikis are simple to use, asynchronous, Web-based collaborative hypertext authoring systems. The original concept is due to programmer Ward Cunningham [19], whose prototype implementation has inspired many variants [22]. While a precise definition of wiki does not exist [18], the general consensus is that a wiki is a collective website where a large number of participants are allowed to modify or create pages using their Web browser.

Wiki introduced groundbreaking innovations at the level of technology for supporting collaborative web-authoring, but also at the level of the process, philosophy and even sociology of such authoring ([22], [3], [12], [14], [1], [31]). From the point of view of technology innovation, wiki introduced a new and simple way to edit web pages, and this at a time (1995) when the web was a read-only medium for all but the most technically adept users (i.e., webmasters). To edit a page on a wiki site, all a user needs to do is to click on the Edit button (or link) that appears on that page (**Figure 1**), modify the text that is then displayed in an editable field, and click on a Save button (**Figure 2**). In spite of the apparent crudeness of its interface (in particular, the non-WYSIWYG nature of its editor), non-technical users are able to use it with relative ease [10].

From the point of view of innovation in the process, philosophy and sociology of collaborative web authoring, wiki introduced a new way of thinking that favors:

- Democratic Peer Review over Editorial Control

- Ease of Access and Open Editing over Security and Control

- Incremental Growth over Upfront Design

- Free Form Content over Structured Content

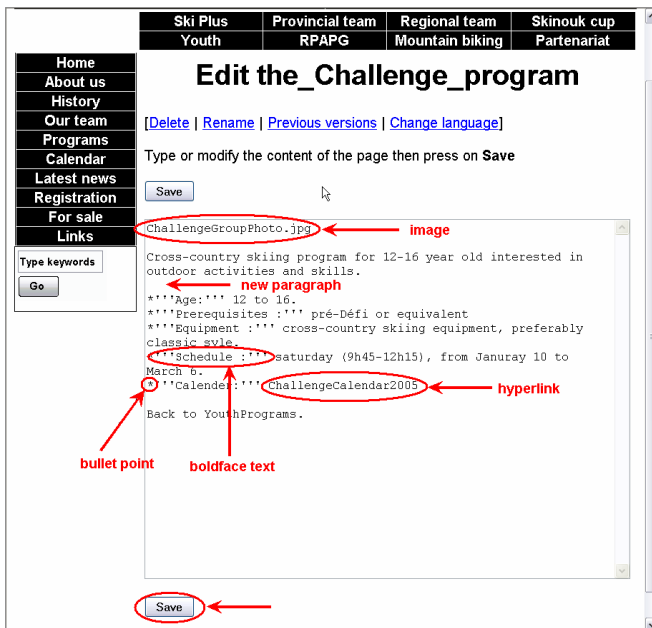**Figure 1: Viewing a wiki page and opening it for editing.**

Figure 2: Editing the content of a wiki page.

This particular philosophy is often referred to as *"The Wiki Way"* [19].

While the Wiki Way may at first look like a recipe for disaster except for small and obscure web sites, it turns out to be a reasonable strategy with attributes that enable a wide variety of applications. Wikis have been used for corporate intranets [2], software documentation (e.g. codex.wordpress.org), Frequently Asked Questions (FAQ), repositories (e.g. www.allmyfaqs.com), textbooks (e.g. www.wikibooks.org), travel guides (e.g. www.wikitravel.org), specialized knowledge bases (e.g.

www.fluwikie.com) and even collaborative story-telling by children [11]. The most high-profile wiki project is by far Wikipedia (www.wikipedia.org), "the free encyclopedia that anyone can edit". Started in 2001, Wikipedia is already the world's largest online encyclopedia. In spite of (or thanks to) its completely open nature, it provides high quality information in over 100 languages [15] and is increasingly being cited by mainstream media [20]. Various theories have been proposed to explain Wikipedia's unlikely success in terms of: a decrease in transaction costs for editing and correcting content [3], the ease of transition from information consumption to information production roles [1] and its use as a vehicle for enhancing contributors' reputation and influence [14].

Many of the popular wiki engines have been internationalized in order to support a variety of languages. For example, many engines allow sites to be configured so that interaction between the wiki engine and its users happens in a given language. Moreover, many provide full Unicode support, which allows authoring of content in any language. The fact that a wiki engine is internationalized does not however automatically mean that it is well-suited for multilingual content. Indeed, the vast majority of wiki engines (even internationalized ones) can only support one language at a time on any particular site, and do not deal well with situations where the same content needs to be published in several languages at the same time.

There are large notable exceptions to this, such as Wikipedia and Wikitravel, which provide content in multiple languages. However, these are not truly multilingual, because they do not provide the same content in different languages. These sites are in fact a collection of parallel communities that produce content about overlapping sets of topics in different languages, with little if any synergy across languages.

This **parallel authoring** approach has two major advantages. Firstly, because it does not aim at providing the same content in different languages, the task is fundamentally simpler and does not require any special tool. It can be supported even with the basic functionality provided by most wiki engines. Secondly, it ensures that the content in each language will be written from the ground up with that particular linguistic and cultural audience in mind.

The main disadvantage of this approach is the lack of synergy and content re-use across the various languages. In order to produce high-quality content, authors typically need to spend a lot of time researching their topic, writing it up in a way that reads well for a wide audience, and tracking the topic for new information as it becomes available. With the parallel-authoring approach, each linguistic community must do this work from scratch every time.

For a community like Wikipedia, which is able to leverage a very large population of authors in all languages, this is not a big issue. But there are sites where the number of domain experts in each target language is too small to support parallel authoring, or where parallel authoring is not desirable for other reasons. Consider for example Fluwikie (www.fluwikie.com), a wiki site that collects and disseminates information for the prevention of pandemic influenza. At the moment, all content on Fluwikie is available in English only, with only a relatively low number of pages having been translated into French, Spanish and Turkish. This is a problem because many of the countries where outbreaks

of pandemic flu might initially occur have a population that predominantly does not speak English (ex: Indonesia, China). Because pandemic flu is a highly specialized topic, one cannot expect to create sub-communities of domain experts for each and every language. However, given the importance of the topic, it might still be possible to spawn a community of volunteers willing to **translate** such content. This translation would have to be widely cooperative given the nature of the issue: speed in sharing important scientific findings, news, and ideas would hopefully lead to more appropriate action in the relatively urgent context of a pandemic.

Even a site like Wikipedia, which seems amenable to a parallel authoring approach, could benefit from a translation approach. Indeed, given that Wikipedia advocates the use of a Neutral Point of View (NPOV) [1] that represents views fairly and without bias, it seems that much of the content **could** be reused across languages, which might save a significant amount of effort. In such a context, it would be helpful to have features that would allow each linguistic community to at least keep abreast of what the other communities are writing on a particular subject.

In short, there are many circumstances where a translation approach would make more sense than parallel-authoring. Unfortunately, there currently exists very little in the way of wiki processes and tools for a translation approach. On the few sites where a translation approach has been attempted (ex: Lizzy[2], MetaWiki[3], Fluwikie[4]) the burden of ensuring the correspondence between different versions of the same content is almost entirely on the end user, and there is little if any automation of even the most basic operations. While the idea of tools for multilingual wiki content have been discussed and proposed (ex: [23], [24]), very little has been done to actually implement them (except for [4] and [26]), and they are certainly not commonly used on wiki sites.

In this paper, we take a **first systematic look** at what processes and tools are needed to support collaborative translation of wiki content by a group of loosely coordinated volunteers. In other words, we are trying to define the basics principles of *"Translation the Wiki Way"*.

It is worth noting that this concept of *"Translation the Wiki Way"* may have applications outside the world of wikis. Indeed, in many small to medium-sized organizations, authoring and translation of content may be a more lightweight, almost wiki-like affair than in larger organizations like automobile manufacturers. Even in the context of large industrial organizations, Schubert [28] notes a recent trend towards what seems to us like a more wiki way of approaching multilingual content:

> *"[...] the distinctions between the professions of translators, technical writers, documentation engineers, information managers etc., are becoming increasingly blurred [...] jobs and task profiles diversify. Technical writers engage in writing, updating and maintaining documentation in several*

---

[1] http://en.wikipedia.org/wiki/NPOV

[2] http://lizzy.iit.nrc.ca/

[3] http://meta.wikimedia.org/wiki/Meta:Interlanguage_links

[4] http://www.fluwikie.com/

languages in parallel, which takes them quite far afield from the classical profile of a monolingual text producer."

Given this, it could be that lightweight tools and processes designed for the wiki world can positively influence more traditional tools and processes, and suggests ways in which they could be adapted to better deal with the fast and ever changing nature of our modern internet world.

The remainder of the paper is organized as follows. In Section 2, we look at processes and tools that are currently being used for creating and maintaining multilingual content in more traditional industrial contexts, and discuss why they are not appropriate in a wiki context. In Section 3, we discuss the results of a User-Centered Design exercise carried out to explore what processes and tools for *"Translation the Wiki Way"* should ideally look and feel like, from the perspective of an end user. In Section 4, we discuss our experience developing basic processes and tools for the special case of bilingual content translation by fully bilingual authors. In Section 5, we discuss how this simple implementation could be extended to provide even more sophisticated features, and in particular, support the general case of a site with more than two languages. Finally, Section 6 provides conclusions and a list of directions for future work.

2. TRADITIONAL TRANSLATION PRACTICES ARE NOT APPROPRIATE FOR THE WIKI WORLD

Although processes and tools to support content translation are practically nonexistent for wiki sites, such processes and tools exist and are being used for translating content in more traditional industrial environments. In this section, we take a look at those and discuss why they are not appropriate in a wiki environment. Note that since a tool always assumes a particular process, our discussion focuses on limitations of the processes, as opposed to the tools that support them.

Currently, there are three predominant processes used in translation, which we call:

- Sequential translation
- Parallel authoring
- Incremental just-in-time translation

Sequential translation is popular with very large industrial organizations such as automobile and computer manufacturers, who need to publish large amounts of product documentation in several languages. This approach is described as follows in Shutz and Nubel [30]:

> *"Today, technical documentation is a sequential process performed over several stages with very restricted communication channels between the different stages. The main stage in this process is authoring which is concerned with the actual composing and writing of service information and repair instructions. [...] The very last stage in the documentation process is translation which in most cases is done by an external translation agency or translation companies."*

and also in Hartley and Paris [16]:

> *"The prevailing scenario in such applications is to write a text in a given language and translate it into the other languages required once the final version of the text has been agreed. Translation can take place only when a final document has been produced, and the product launch can take place only when the translation, in its turn, has been completed."*

The problem with this process is that it is optimized for translating a **frozen** and **finalized** document. It does not deal well with situations where the source needs to be modified after translation has started (or been completed). Yet, no matter how much care is taken to review the source document beforehand, errors are almost always found after translation has begun. Even if the source document does not contain errors, it is bound to evolve over time to reflect changes in the reality that it is documenting. The reliance of sequential translation on a stable source document has two negative consequences. First, there is a tendency to spend an inordinate amount of time revising the source document before starting translation, which increases time to market for the product (or time to publication in the case of documents that are not tied to manufactured products). Second, there is a tendency against modifying the source once translation has started, unless the changes are deemed really important. This results in less agility and a reduced ability to deal quickly with change.

Hartley and Paris [16] also describe an alternative approach used by large manufacturers, which we call **parallel authoring**, where:

> *"Writers of different native languages are briefed at the same time, and write the documents more or less independently in the different languages, conferring with each other when necessary. This is designed to avoid delays inherent in the translation scenario, and also to ensure that from the outset all documents are biased to the expectations of their respective readership."*

A similar approach is used by many large corporations that need to provide web sites in multiple languages (Esselink [13], p.37). As pointed out in the above quote, the main advantage of parallel authoring is that it accelerates time to publication and ensures that each language version is optimized for its linguistic community. However, as we discussed in our introduction, the downside is a lack of synergy and content reuse across the various language versions of the content.

With the advent of the Web, a third model has emerged, which we call the **incremental just-in-time translation** process. This process is optimized to deal with the volatile and ever-changing nature of the Web, where the life expectancy of a page is often counted in terms of days, not months or years. With the incremental just-in-time translation process, changes to web pages are tracked in real-time, and requests for translations are issued as soon as such a change occurs. In some cases, the requests are even sent automatically to translation services to ensure a speedy translation. In a way, incremental just-in-time translation provides the best of both worlds. It combines the fast time to publication of parallel authoring while supporting synergy and content reuse across languages as in sequential translation. Because of the growing importance of the web, a large number of products (generally referred to as Globalization Management Systems)

have emerged to support this process (Esselink [13], p. 22). One disadvantage of such an incremental approach is that it tends to encourage (sometimes even enforce) literal, sentence by sentence translation. But good translation which is idiomatic and culturally appropriate often requires that the translator deviate from the structure of the source text. In particular, whole sentences may be left altogether untranslated if they are not culturally appropriate in the target language.

One can see readily that none of the processes described above deal well with all the realities of the wiki world. Sequential translation is clearly inappropriate since wiki content is grown in an organic ever-changing fashion, and often never reaches a stable and final state. Parallel authoring is somewhat more appropriate, and in fact, this is the model being used by the few existing multilingual wiki sites (ex: Wikipedia and Wikitravel). But as pointed out in the introduction, it is not appropriate for all types of wiki sites, and even sites that currently use this approach could benefit from a closer integration of at least some of the content in the different languages. Of the three processes, incremental just-in-time translation seems best suited to wiki. However, it still has some important limitations which we describe below.

First, although incremental just-in-time translation supports a more flexible workflow where changes can be made after translation has started, it still assumes that there is a master language (usually English) and that all changes will be first made to that version of a page. But in a wiki context, it is not reasonable to ask all contributors to write content and make changes in English first, because not all of them will be fluent enough to write in that language. Certainly, the majority of contributors to a wiki site can express their thoughts more clearly in their native language. Therefore, in a wiki context, the process must allow authors to write in their own native language, and make it easy to later propagate their contributions to the remaining languages.

Second, incremental just-in-time translation assumes that timely translation of content can be ensured through contractual agreements or some form of centrally exercised control over the translation effort. But this does not hold in the context of wikis because they are typically maintained by a loosely coordinated group of volunteer contributors. Therefore the amount of central control that can be exercised to ensure timely translation into the various languages is limited or inexistent. A consequence of this is that the process must allow the team of translators to self-organize, for example, by allowing translators to independently judge which translation tasks have the highest priority at any point in time. Another implication is that one may have to publish changes or new content written in one language before they have been translated to all other languages. The reason for this is that it may take some time before this complete translation is done. In the meantime, the system must provide ways for visitors to navigate the site and get up-to-date information even though some of its content may not have been translated yet into their native language. For example, the system might allow visitors to see machine translation of untranslated sections. The fact that translators on wikis are not monetarily compensated for their work also means that the process must include effective ways of recruiting volunteer translators. For example, it must provide many opportunities for site visitors to progressively migrate from a pure role of consuming content towards a more active role of
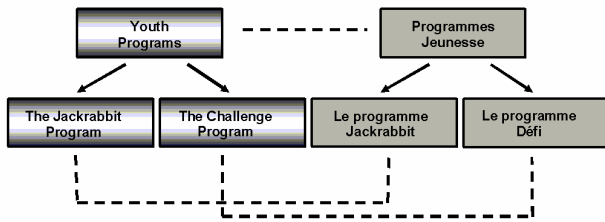
Figure 3: Link structure of a bilingual site.

producing/translating content. As noted by Bryant et al [1], most core contributors to Wikipedia started as simple consumers of content, and were gradually pulled into contributing to it by specific characteristics of the system and the community built around it.

Third, incremental just-in-time translation assumes a well-compartmentalized workflow, where different people have different responsibilities and privileges for different parts of the process, such as authoring, translating, revising and publishing. But in the world of wikis, the same people often hold all those responsibilities and privileges, even if they only exercise some of them at any given point. This turns out to be an important attribute of wikis, one which contributes to the success of sites like Wikipedia. Indeed, the failure of Nupedia (the precursor of Wikipedia, which was started by the same people) is largely attributed to the fact that it used a traditional heavily compartmentalized workflow ([3], [27]).

Fourth, incremental just-in-time translation assumes that translation will be done by professionally trained translators. This does not hold in the world of wikis because they tend to be maintained by domain experts as opposed to professional writers and translators. In particular, one cannot assume that these contributors are fluent in any but their native language. An implication of this is that there is a need for tools that allow domain experts to translate content from a source language they might not know into their native language. Also, those same domain experts tend to act as both authors and translators at the same time. Therefore the process must allow them to easily switch between translation and authoring modes. For example, if a domain expert is translating a page from English to Spanish, and in the process he thinks of new content to be added, he should be able to immediately switch to an author role and add this new content directly to the Spanish page, and later translate it back to English.

Fifth, incremental just-in-time translation does little to help translators preserve the structure of intra-language links in the different languages. **Figure 3** shows an example of a three-page bilingual site, with inter-language links (dashed lines) and intra-language links (full lines). Inter-language links are links between corresponding pages in the different languages. Most tools that support incremental just-in-time translation automate the generation and maintenance of such inter-language links. However, they provide little if any help when it comes to maintaining the correspondence of intra-language links (i.e. links between different pages of a same language). For example, a

translator may want to make sure that the intra-language links that appear on an English page **Youth Programs** and the intra-language links that appear on its French translation **Programmes Jeunesse** are themselves respective translations of each other. Unfortunately, such a check must be performed manually with most current tools.

3. DESIGNING A CROSS-LANGUAGE WIKI FROM THE END-USER PERSPECTIVE

Having established the need for special processes and tools to support production and translation of multilingual wiki content, we now explore what they should provide from the perspective of an end user. More precisely, we report on a lightweight User-Centered Design (UCD) exercise carried out to capture and describe the needs of different kinds of users for a multilingual wiki system. This is work still in progress and the full and final detail will be published soon [9]. We can however provide preliminary high-level findings based on our work to date.

This work is being done by two of the authors (Désilets and Gonzalez) who have had some experience in translating wiki sites. In addition, Désilets has developed basic wiki tools to help with bilingual content translation (these tools will be described in Section 4). In this design exercise we employ modeling techniques from several methodologies: Usage-Centered Design [5], Interaction Design [6] and Agile Usability [25].

Up to now, we have defined three high-level **Business Goals** [25], that is, criteria by which success of a multilingual wiki might be judged. These are:

Information Available in any Language: Site visitors can get up-to-date information in a timely fashion, as long as they can read one of the languages that the site supports.

Content Authoring in any Language: Content creators are able to write contributions as long as they are able to write in one of the languages supported by the site. Their contributions become available in other languages within a reasonable time frame.

Thriving Community of Translators: The site fosters the emergence of a self-organizing and vibrant community of translators which ensures that all important information is available in all important languages in a timely fashion.

Note that it may look like the third goal is more a means to an end than an actual end goal, but in fact this is not really the case. This third goal essentially captures the fact that we are interested in a "Wiki Way" of meeting the first two. Indeed, a system could meet the first two goals and still use a traditional workflow that is heavily compartmentalized and centrally controlled. Such a system could hardly be described as a multilingual **wiki** system.

We have also identified a clear set of **User Roles** [5] that capture the various types of intentions users may have when using a multilingual wiki. These are:

Site Visitor: Interested in consuming information on the site (in some language she is comfortable reading) without creating or modifying content.

Content Author: Interested in creating or modifying content (in some language she is comfortable writing in), without worrying about translation.

Content Translator: Interested in translating content without contributing original content at this point.

Sub-domain Curator: Interested in ensuring the quality of content on a set of pages (a sub-domain) she is interested in and knowledgeable about. This may include content that was written in a language that she cannot read.

Language Curator: Interested in making sure that the portion of the site in a particular language is up to date and well written. This may involve translation from languages that she cannot read.

Abuse Preventer: Interested in limiting abusive behavior on the site, in particular spamming. This may be limited to abusive behavior exhibited on pages in languages they know, or may include pages in languages foreign to them.

Linguistic Resources Maintainer: Interested in maintaining site-specific linguistic resources like lexicons, and making sure that they continue to be useful to the community of authors and translators.

Site and Community Leader: Acts as an authority figure on the site, and makes sure that the site and community as a whole are "doing OK".

Among those, we have labeled the first three (**Site Visitor**, **Content Author** and **Content Translator**) as focal roles. A focal role is one that is frequent, very important or both, and which therefore needs to be very well supported by the system. Note that a list of roles does not imply a compartmentalized process with responsibilities clearly delineated between people. As we said earlier, User Roles are meant to capture different kinds of user intentions, not responsibilities. Indeed, it is quite common (especially in a wiki context) for a same user to switch roles several times in a single session on a system.

The success criteria and user roles have pretty much stabilized at this point of our work. Based on these roles, we are still in the process of defining **User Tasks**. The purpose of these is to describe at a high level the various tasks which a user acting in a particular role needs to carry out in order to achieve a particular goal [5]. For example, a User Task for the role **Content Translator** might be **Find which Parts of a Page Need to be Retranslated**. We are also threading the User Roles and User Tasks into **Scenarios** [6], which are detail rich narratives describing how a particular user might employ the system for a particular purpose on a particular occasion. Finally, we are also generating a **Span Plan** [25], which prioritizes and organizes features to help developers decide what functionality to implement first, so that the system provides primitive but valuable end-to-end functionality even at the earliest stages of development.

Once this UCD exercise is completed, we believe it will provide a valuable blueprint that wiki developers can use to implement progressively sophisticated support for multilingual content into any wiki engine. At the moment, however, we can only offer broad preliminary insights that we have gained through this partially completed exercise.

First, multilingualism seems to be a broad-reaching aspect that affects all parts of a wiki engine. It affects all users, whether they are mainly consumers, creators, translators or organizers of information. We noticed that many User Tasks which on the surface seemed unaffected by multilingualism, turned out on closer inspection to have a language dependency. For example, it seemed at first that the task of reading the content of a page (a **Site Visitor** task) would not be influenced by multilingualism. But on closer inspection, we realized that visitors might need to know when a page in their native language is out of date with the most recent version in other languages, and be offered options to deal with the situation (ex: ask for someone to translate it, or view a machine translation of the more recent changes made in another language). The impact of multilingualism also seems to span all phases of a wiki site's lifecycle, from initial seeding of content, to growth, to steady-state maintenance.

Second, multilingualism could potentially increase the complexity of tasks that are currently simple to do in monolingual wiki sites. This must be avoided as much as possible. For example, making a contribution to a wiki site is currently very easy, and this accounts for much of the success of sites like Wikipedia [3]. But consider the case of an Italian user who, in the midst of reading an Italian page on a multilingual site, decides he can contribute to it. If the site were to implement the traditional workflow used in most industrial translation contexts, this user would have to first make his contribution in English, and then reproduce it in Italian. This alone might stop him from contributing. In the context of a wiki site, it would therefore be better to allow the contributor to write directly on the Italian page, and then have him or another person reproduce those changes on the English page.

Third, the system needs to support fluid and seamless transitions between most roles. This is a characteristic of current monolingual wikis in general, and it must be preserved in a multilingual context. For example, users need to be able to easily move from visiting the site, to authoring some content, to translating, back to authoring, and then back to visiting, etc... This is needed in a wiki context because users usually hold a wide range of privileges and responsibilities. Another reason for supporting this smooth transition is that it encourages people to gradually migrate from acting solely as a **Site Visitor** to becoming more active and acting as a **Content Author**, a **Content Translator**, or both. This is essential in meeting our third Business Goal of a **Thriving Community of Translators**.

Fourth, motivating people to translate wiki content may have to be done in a very different way than motivating them to write original content. For many contributors, the task of translating content may be perceived as less glamorous than writing original content, and this might make it a less attractive task. But at the same time, it requires much less time and commitment, and it is much more clearly defined than authoring. Thus one way to incite volunteer contributors into doing translation work might be to make it easy for them to find translation tasks that fit their current time availability (ex: *"I only have an hour, which is too short for authoring anything original. What small translation tasks could I complete in that time instead?"*), or that they are likely to accept to do (see for example [7] for an evaluation of different strategies for suggesting tasks to contributors).

Finally, while most of the User Tasks and Scenarios can be supported through relatively simple technologies, there are some
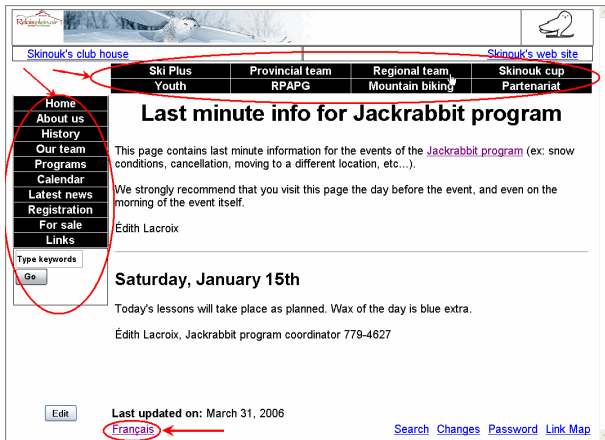
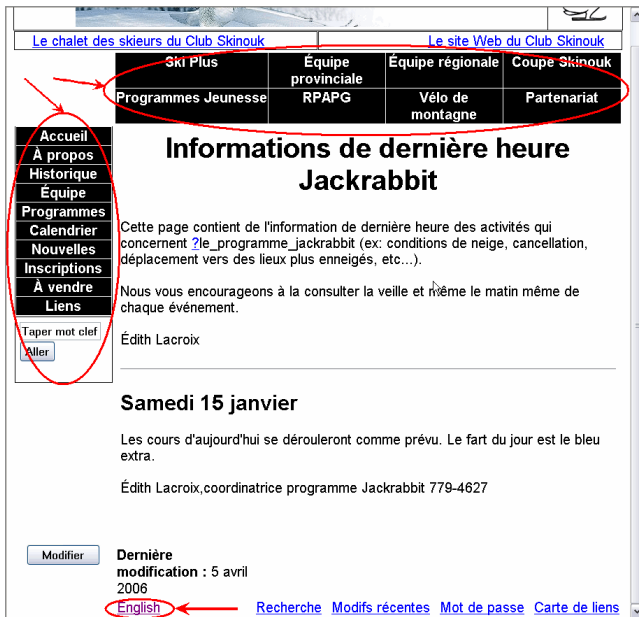**Figure 4: An English page on a bilingual LizzyWiki site.**

Figure 5: The corresponding page in French.

key ones where more advanced technologies like Machine Translation may be needed. We will provide several examples of those in Section 5.

4. SUPPORTING BILINGUAL SITES IN LIZZY WIKI

Having painted a broad picture of what a multilingual wiki system should look like from the point of view of an end user, we now describe features we implemented in our own wiki engine, called LizzyWiki [21], to support part of that vision. As a first step, we focused on the special case of bilingual content being maintained by bilingual contributors, with many of the visitors being also bilingual. In other words, we assumed the following.

- The wiki site only has two languages (i.e. bilingual only).

- All authors and translators are able to read both languages, and able to write in at least one of them.

- Most (but not all) visitors are able to read in either language, although they may be more comfortable reading in one of the two.

Note that those assumptions are perfectly reasonable in the context of a bilingual country like Canada where LizzyWiki was developed. In addition to these assumptions, we focused only on the three focal roles: **Site Visitor**, **Content Author** and **Content Translator**. Besides those simplifying assumptions, we stayed true to the broad principles captured by our UCD exercise; in particular, we wanted to make sure that:

- Multilingual features interfere as little as possible with the usual processes of visiting and authoring content on the wiki.

- The system allows users to switch fluidly between **Site Visitor**, **Content Author** and **Content Translator** roles.

We now explain exactly how this is supported in LizzyWiki, using a series of examples based on an actual site, used by a small cross-country skiing club in the Ottawa area[5]. We describe features for each of the focal roles in turn.

For each of the examples, we also discuss how multilingual support in LizzyWiki differs from multilingual support in MediaWiki [24] (the engine used by Wikipedia) and PmWiki [26] (a wiki engine with fairly advanced multilingual features). One central difference is that LizzyWiki supports the concept of a page being up-to-date or not with its counterpart in the other language. Neither MediaWiki nor PmWiki support this because they lack the basic infrastructure for tagging a page as being up-to-date with its language counterparts. In turn, this fundamental difference explains why LizzyWiki is able to support many special features for identifying when a page is out of date and for helping the user remedy the situation. Another basic difference is that LizzyWiki automates many menial tasks involved in translation, which MediaWiki and PmWiki require the user to carry out manually.

We start with **Site Visitors**. To them, a page on a multilingual LizzyWiki site looks like **Figure 4**. Essentially, it looks like a normal wiki page, except that there is a cross-language link at the bottom of it. Clicking on this link takes the user to the version of the current page in the other language (**Figure 5**). Note how the system uses a different container to display the pages in different languages. When viewing an English page, all the buttons on the left and on the top are displayed in English, but when viewing a French page, they are displayed in French. All dialogs between the system and a visitor are also carried out in the language of the page from which this dialog was invoked. For example, if the user does a keyword search from a page in French, the search results dialog will be carried out in French. This means that the user never needs to explicitly say which language she wants the system to speak to her. This is determined implicitly based on the language of the pages she is reading.

The features we have mentioned above (cross-language links, language-sensitive containers and language-sensitive dialogs) are

---

[5] Skinouk Ski Club (www.skinouk.ca)
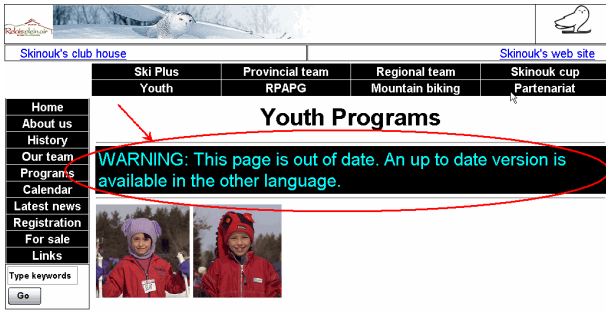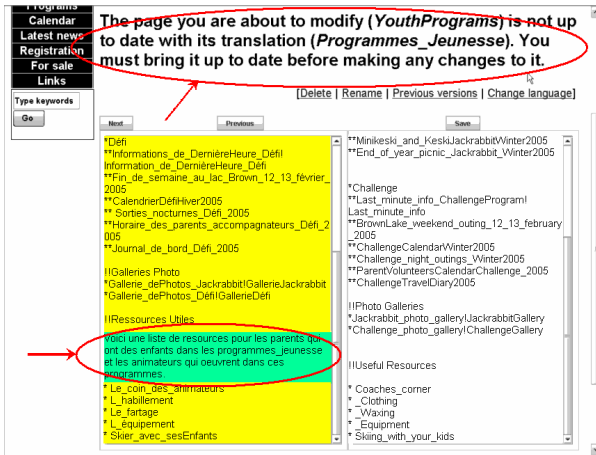
Figure 6: An out-of-date page.

Figure 7: Editing an out-of-date page.

also supported by MediaWiki and PmWiki. However, when a **Site Visitor** gets to a page that is out of date with its counterpart in the other language, she starts seeing features that are unique to LizzyWiki. For example, **Figure 6** shows a case where a user visits an out-of-date English page and sees a large warning telling her that she is not reading the most recent content. She may then choose to either (a) continue reading that page but treat the information it contains with caution, (b) read the French translation instead (if she is fluent in that language) or (c) switch to **Content Translator** role and translate the changes. Case (c) is particularly interesting since it invites users to progressively transition from being pure visitors, to becoming contributors (in the form of translators) on the site.

We now move to describing features for **Content Authors**. When a **Content Author** wishes to make a contribution to an existing page, he can do so directly in that page, no matter what language it is written in. In other words, he never has to first make the contribution in a "master" language (ex: English) and then later

translate that contribution to his native language. Note that in MediaWiki and PmWiki also, a **Content Author** can make his changes in any language. But this is only because those systems do not support the concept of up to dateness and therefore allow language versions to evolve independently of one another without any coordination at all.

Once a **Content Author** modifies a page, its counterpart in the other language automatically starts being displayed with the warning message illustrated in **Figure 6**. This alerts **Site Visitors** and **Content Translators** to the fact that the page is out of date. This message will remain until someone in a **Content Translator** role brings the two pages back in sync.

If a **Content Author** tries to modify a page that is out of date with its other language counterpart, she is first asked to bring it up
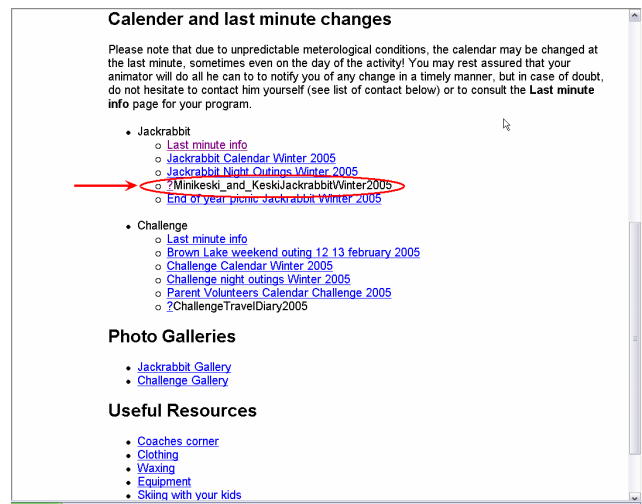

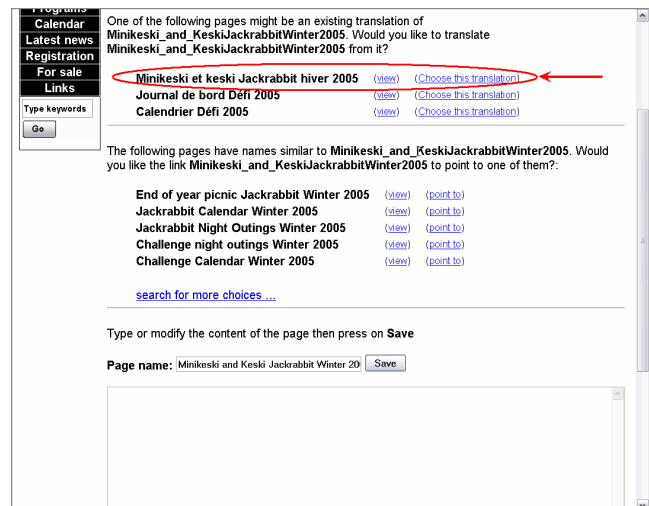Figure 8: Clicking on a question-mark link to create a new page.

Figure 9: Suggested pages to translate new page fron.

to date. This is in order to avoid situations where both versions of the page are out of date, and are lacking some changes that have been made in the other version. The dialog used to bring a page up to date with its counterpart is displayed in **Figure 7**, and it will be explained in more detail later when we discuss support for the **Content Translator** role. Note that in MediaWiki and PmWiki, **Content Authors** are not required to bring a page up to date with its translation before modifying it. Again, this is only because those systems do not support the concept of up-to-dateness and co-ordination of pages at all.

When a **Content Author** creates a new wiki page, the system helps him decide whether that new page should be created from scratch or translated from an existing page in the other language. For example, in **Figure 8**, the user clicks on a question-mark link (which, on several wikis, indicates a yet-to-be-created page) to create a new English page called **Minikeski and Keski Jackrabbit 2005**. This brings him to a dialog that looks somewhat like the usual wiki dialog for creating a new page (**Figure 9**). The difference in LizzyWiki is that it provides, at the top, a list of French pages that "look" like they might be a translation of the page the user is trying to create. These are French pages that do not have a translation yet, and whose location in the French intra-language links graph is "similar" to the location of **Minikeski and Keski Jackrabbit 2005** in the English intra-language links graph. More precisely, the system looks for pages that have at least one "common parent" with the page being created. For example, in **Figure 3** French page **Programmes Jeunesse** is a common parent between French page **Le Programme Jackrabbit** and English page **The Jackrabbit program**, because (i) it points to **Le Programme Jackrabbit** and (ii) its translation **Youth Programs** points to **The Jackrabbit program**.

Note that the **Content Author** may ignore these suggestions and, as he would do in a regular wiki, can input the content of the new page directly into the empty text box provided for that purpose. In this case, though, it turns out that the first suggestion, **Minikeski et Keski Jackrabbit Hiver 2005**, is indeed the French equivalent of **Minikeski and Keski Jackrabbit Winter 2005**. By clicking on the *"Choose this translation"* link beside it, the user is brought to a dialog that allows him to create the English page by translating it from the French version. An example of this dialog is displayed in **Figure 11**, and it will be explained in more detail when we discuss support for **Content Translators**.

Notice how this list of potential translations helps the **Content Author** and **Content Translators** maintain the correspondence between the intra-language link structures in both languages. In particular, this avoids situations where, for example, a new English page is accidentally created and edited independently from its existing French counterpart, and when this is discovered, the content of the two versions later needs to be merged back together. This suggested translations feature is unique to LizzyWiki. For example, if you create a new English page with MediaWiki or PmWiki, finding whether this new page has an existing French counterpart from which to translate is difficult. One essentially has to search the French site using keywords that might be contained in the name of the French equivalent if it exists.
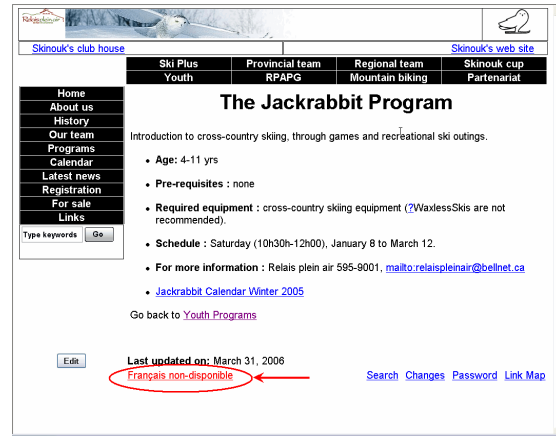
Figure 10: French translation is not available.
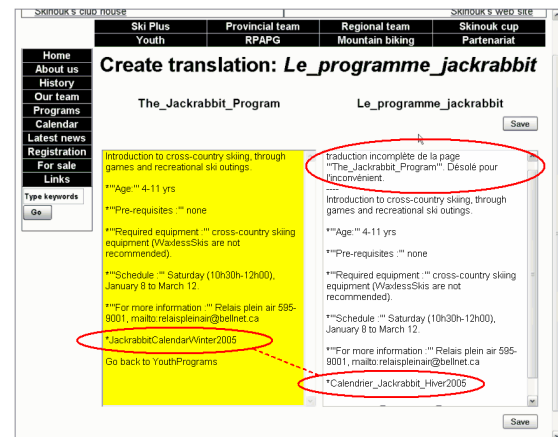
Figure 11: Creating a new French translation.

LizzyWiki also includes features to assist **Content Translators** in their job. For example, the out-of-date warning message shown in **Figure 6** allows **Content Translators** to see when a page needs some translation work. They can then bring such pages up to date by clicking on the Edit button. This brings them to the synchronization dialog shown in **Figure 7**. This dialog displays the most up-to-date version (French in this case) on the left, and the out-of-date version (English in this case) on the right. Changes that have happened in the French version since the last time both were in sync with each other are highlighted. The system also has an automatic scrolling feature, whereby it can automatically scroll the English version to a region that approximately corresponds to a particular change in the French version. This avoids the translator having to manually locate and move to the location in the English version where he needs to make a particular change. This automatic scrolling feature uses a simple heuristic to determine the equivalent location of a change in the target text (right hand side). Basically, it computes the relative location of the change in the source text (left hand side) as a proportion of the

whole text and assumes that the equivalent location will be approximately at the same relative position in the target text. Once the translator has finished bringing the English page up-to-date with its French counterpart, he signals this to the system by pressing a button on the save dialog (not shown here). From then on, the page will be displayed in either language without a warning sign.

Note that this page synchronization dialog is unique to LizzyWiki, and saves the user much work. Doing a similar side by side comparison in MediaWiki or PmWiki requires a lot of manual steps in order to: display both pages for editing in non-overlapping browser windows, identify the differences between the two, and figure out on which side each change must be reproduced. The last point is particular hard to do manually, because it is not always obvious whether a change corresponds to say, an insertion on the English side that must be reproduced on the French side, or a deletion on the French side that needs to be reproduced on the English side. Only by looking through the revision history of both sides can the user determine this with certainty.

Note also that while this page synchronization dialog somewhat encourages literal sentence by sentence translation, it does not actually enforce it. The translator is ultimately responsible for deciding how to best re-express a change in the target language, and he is the one who tells the system when he deems the two pages to be in sync. For example, the translator may decide to translate a single sentence using two or more sentences in the target language in order to make the source text more idiomatic. Or, he may decide to not translate a particular source sentence at all, because it is not culturally relevant in the target language. A consequence of this is that the page synchronization dialog could be used even in a parallel authoring context. This feature might prove to be a useful way for the various linguistic communities of a site to keep abreast of what they are writing about on any particular subject, without necessarily having to neither adhere to the same structure nor cover the exact same content.

Besides alerting **Content Translators** of pages that are out of date with their counterpart, LizzyWiki alerts them when a page does not yet have a translation. For example, **Figure 10** shows an English page that does not have a French translation yet. In such a situation, the inter-language link is displayed in red and it specifies *non-disponible* (which means *not available* in French). A **Content Translator** can easily remedy this situation by clicking on this *not available* link, at which point the system prompts him for the name of the French translation. Once this is done, the user sees a dialog like the one shown in **Figure 11**, which allows him to translate the new page based on the content of its existing counterpart in the other language. In this example, the system shows the content of the English page **The Jackrabbit Program** on the left, and its newly created French translation called **Le programme Jackrabbit** on the right. The English content of **The Jackrabbit Program** was automatically pasted into the French translation, and a warning sign was automatically added at the top, to tell **Site Visitors** that this page is still under translation. The **Content Translator** can then start replacing the English content with its French translation in the right-hand side, and save when he is done (possibly erasing the warning that was inserted at the top, if this is the final save of the translation).

Again, this kind of one-click creation of translations makes it very easy for people to move from a **Site Visitor** role to a **Site Translator** role. This in turn may result in more pages being translated. In contrast, to do the same thing requires no less than 10 steps in MediaWiki and 6 steps in PmWiki. Forcing the user to carry out such an involved series of actions creates a definite barrier to participation in the translation of the site. Moreover, it increases the cha,nces of human errors and of the user not being able to complete the task.

We note furthermore that when a **Content Translator** creates a new translation, LizzyWiki helps him by automatically translating links that have an existing counterpart in the other language. For example, **Figure 11** shows a situation where the French page **Calendrier Jackrabbit Hiver 2005** has previously been created and translated from English page **Jackrabbit Calendar Winter 2005** (or possibly the other way around). Therefore, when the system pasted the content of English page **The Jackrabbit Program** into the newly created translation **Le programme Jackrabbit**, it automatically changed the link **Jackrabbit Calendar Winter 2005** to **Calendrier Jackrabbit Hiver 2005** in the wiki markup. Like the suggested translations shown in **Figure 9**, this automatic link translation assists the **Content Translator** in maintaining the parallel structure of the intra-language links in both languages.

Looking back at the Business Goals we defined in Section 3, we can see that all in all, LizzyWiki does a reasonable job at meeting the needs of users in the special case of bilingual content created and translated by bilingual contributors and read mostly by bilingual visitors. Regarding **Information Available in Any Language**, we make it possible for visitors to browse the site in their native language, and when they hit a page that is out of date, they are notified of it. They can then choose to still read that out-of-date page in their native language, or they can chose to read an up-to-date version in their second language. Regarding **Content Authoring In Any Language**, we make it possible for authors to write content in their native language whatever that might be, and the system will later on help them (or other contributors to the site) reproduce that content in their second language. Regarding **Thriving Community of Translators**, we provide tools that invite users to do some translation work and takes care of a lot of the menial tasks involved. Also, the *"translation not available", "page out of date"* and *"still under translation"* warnings help the community of translators to self-organize and keep abreast of what pages need translation work done on them.

5. TOWARDS FULL SUPPORT FOR MULTILINGUAL SITES

Although the simple features of LizzyWiki described in the previous section deal well with the bilingual case, the tool still presents important limitations. Firstly, it cannot deal with sites that have more than two languages. Secondly, the requirement that **Content Authors** first bring a page up to date before modifying it tends to break their flow in a major way. Thirdly, features allowing the community of translators to self-organize are fairly limited. We now discuss how current LizzyWiki capabilities could be extended to deal with those issues.
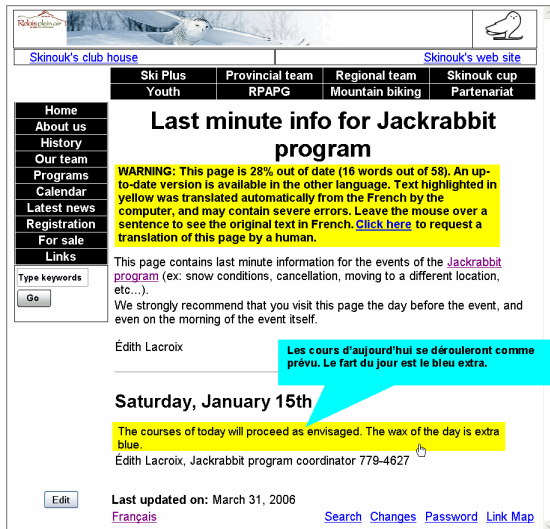
Figure 12: Making partially out-of-date pages still useful.

5.1 Dealing with more than two languages

On the surface it may seem that extending to more than two languages at a time is a trivial matter of putting more than one cross-language links on the pages. But it is much more complicated than that. The main issue is that for a site that has more than two languages, one simply cannot assume that all its contributors and many of its visitors will be fluent in all of the site's supported languages. This creates many complications.

For example, suppose a Spanish **Site Visitor** gets to a Spanish page, and finds that changes have been made to the Turkish version. If he does not read Turkish, there is currently no way for him to get the latest information that was posted on the Turkish page. Similarly, suppose a Spanish user has adopted some pages in Spanish (a **Sub-domain Curator)** and she finds that the Turkish version of one of those pages has been modified. Ideally, she would want to assess the correctness of this new content (in particular, making sure that it is not spam) and reproduce it in the Spanish version. But how is she to do this, if she is not fluent in Turkish?

Various tools could be developed to address this situation, such as:

- Machine Translation tools to help the visitors and contributors get the gist of the changes that were made in languages they cannot read.

- Features allowing the visitors and contributors to request a human translation of the changes, and to be notified when this happens.

- Tools to help visitors assess how much out-of-date a page is (ex: how many words have been changed, and what that represents as a percentage of page length). This would help

them decide how to deal with the out of dateness of the page (ex: read the out-of-date page in native language, read the up-to-date version in a second language, ask for a translation, etc.)

These features are illustrated by the mockup in **Figure 12.** Regarding the first point, note that while the idea of such real-time translation has been suggested in research literature ([30], [8]), this is not a common practice in the industry. Yet, studies have shown that even though current Machine Translation technology is not good enough to provide final professional-quality translation, it is sufficiently good to support gisting ([17], [8]). In the present situation, the task might even be easier because, in many cases, the machine will only be translating small changes to a page. Thus, most of what users would see would be human-translated text, complemented with a machine translation of small changes that have been done in another language version. Because the automatic translation of those changes is embedded in the context of well-translated text, it might be easier for the user to understand it, even if it is of poor quality.

Another complication when a site has many different languages is the possibility of compounded distortion. For example, suppose a change made to an English page is translated from English to French, then from French to German, then from German to Spanish and finally from Spanish to Turkish. In a situation like this, chances are that the message of the original English change will have been significantly distorted by the time it reaches Turkish[6]. One way to deal with this would be to use English as a pivot language. In other words, whenever changes in one language need to be propagated to the other languages, the system would impose the constraint that translation first pass through English. Thus, the English version could be used as a stable reference point by other languages. Note that this does not mean that English would become a master language. Indeed, **Content Authors** would still be able to first write contributions in the language of their choice, and English would only play a role of an intermediary in the translation process.

5.2 Allowing page modification without preliminary synchronization

As pointed out in Section 4, when a **Content Author** tries to modify an out-of-date page, LizzyWiki requires him to first bring it up to date with its counterpart in the other language. But suppose for example that a French author wants to add one sentence to a French page, but the system insists that he should first replicate 20 sentences that have been added recently to the English page. This would stop the author dead in his tracks and would probably result in him giving up.

It would be better if the system allowed the French and English pages to evolve independently of one another, at least for a while. The system would let the author do the merging of both pages in his own time (i.e. when he is ready to switch to a **Content**

---

[6] For example, see "Translation Relay" on Wikipedia:

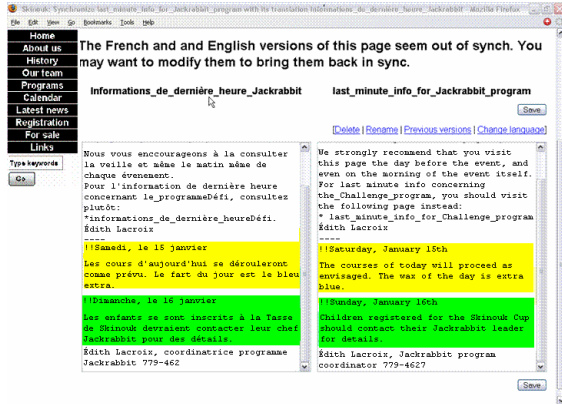http://en.wikipedia.org/wiki/Translation_relay

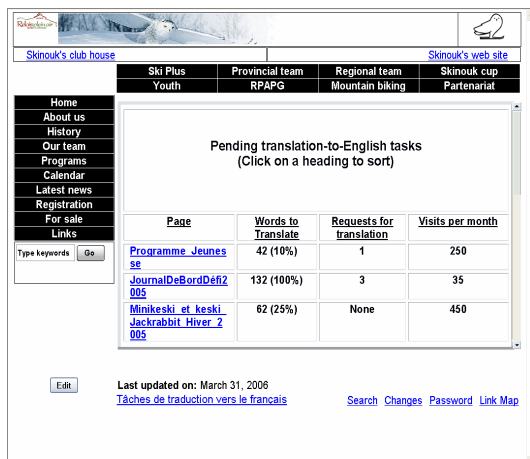**Figure 13: Two-way merging of translations.**

Figure 14: Global view of pending translation tasks.

Translator role), or it could support another person in carrying out this task later. This would necessitate an interface for facilitating a two-way merging of changes in both pages, as in the mockup shown in **Figure 13**. In this example, changes that have happened in the French and English versions are highlighted in different colors, which makes it easier for the user to see what needs to be done in what language. In order to make the pages look more similar to one another and make them even easier to compare visually, the changes have also been replicated automatically on the other side using machine translation. The user can then correct those machine translations on both side and once he is satisfied with the merge, he saves and signals to the system that the two versions are now in in sync.

Note that while this approach allows **Content Authors** to modify pages without breaking their flow, it can introduce some additional complications. Indeed, if changes in the various language versions of a page are not merged rapidly, one could easily end up with a situation where none of the versions is completely up to date (i.e. each language version is lacking at least one change that was made in another language). This kind of chaotic situation could be minimized through the use of a pivot

language, as suggested in Section 5.1. In this case, the pivot language version of a page would act as an integrator of changes made in all other versions, and would therefore always be fairly up to date.

5.3 Supporting better self-organization of the translator community

While the features currently implemented in LizzyWiki allow **Content Translators** to see when an individual page needs translation work, they have no way of getting a global view of what translation work is needed on the site as a whole, and which translation jobs have the highest priority. Features to deal with this issue could easily be implemented such as:

- Tools allowing **Language Curators** and **Content Translators** of one language to get a sense of what pages need translation work done.

- Tools for assessing the priority or a particular translation task (ex: frequency of visits, number of explicit requests for translation).

- Tools for assessing the effort required for a particular translation task (ex: how many words need to be translated).

Figure 14 shows a mockup of a dialog that could support such functionality.

Another issue related to community organization is how to best encourage users to contribute some translations to the site. For example, one could develop features similar to those proposed by Cosley et al. [7], to automatically suggest translation tasks which either (i) are easy to do for a particular user, (ii) are about topics the user cares about, or (iii) are tasks that are highly relevant for the community.

6. CONCLUSION

In this paper, we have taken a first systematic look at processes and tools for supporting multilingual content in a wiki context. We believe that this work provides a fairly extensive and detailed blueprint for how to implement this. It is our hope that it will inspire developers of different wiki engines to implement functionality along those lines, and that this will result in the production of more multilingual wiki content.

We also believe that our description of *"Translation the Wiki Way"* may prove useful to developers of tools for managing multilingual web content in more traditional industrial contexts. We hope it will also inspire them to experiment with more lightweight tools and processes that might allow traditional industrial sites to grow in a more agile and organic way, and to deal better with the rapid ever-changing nature of our modern internet world.

In spite of the progress we made in this paper, much work remains to be done. The principles and functionality we have described remain largely untested as of this writing. In the coming months, we plan to deploy and evaluate them on a real community of use, namely the Fluwikie site (www.fluwikie.com).

Deploying our multilingual tools on Fluwikie will also give us a better glimpse of the social side of the equation. Indeed, the success of sites like Wikipedia is due only in part to technology. Much of the magic comes from the continuous "social

engineering" that goes on inside the community. Therefore, we need to better understand the social rules and procedures that work best in a multilingual wiki.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] Bryant et al. *Becoming Wikipedians: Transformation of Participation in a Collaborative Online Encyclopedia*. Proc. of GROUP International Conference on Supporting Group Work, Sanibel Island, FL, 2005, pp. 1-10.

[2] Buffa, M. *Intranet wikis*. Intrawebs Workshop 06, Edinburgh, May 2006

[3] Ciffolilli, A. *Phantom authority, self–selective recruitment and retention of members in virtual communities: The case of Wikipedia*. First Monday (Peer Reviewed Journal of the Internet), 8, 12, 2003.

[4] Community Wiki Collective. *Multilingual Experiment.* http://www.communitywiki.org/en/MultilingualExperiment (visited on April 2006).

[5] Constantine, L. *Software For Use: A Practical Guide To The Models And Methods Of Usage-Centered Design*. Addison-Wesley, 1999.

[6] Cooper, A. *About Face: The Essentials of User Interface Design*. John Wiley & Sons, 1995

[7] Cosley, D. et al. Using Intelligent Task Routing and Contribution Review to Help Communities Build Artifacts of Lasting Value. Proc. CHI 2006. Montreal, Québec, Canada, April 22-27, 2006.

[8] den Os, E. *Multilingual Web Sites: Best Practice, Guildelines and Architectures*. EURESCOM Report, EDIN 0205-0923

[9] Désilets, A., Gonzalez L. *Designing a Cross-Lingual Wiki from the End User's Perspective*. NRC Report, 2006 (to be published).

[10] Désilets, A., Paquet, S., Vinson, N.G. *Are Wikis Usable?* The 2005 International Symposium on Wikis. San Diego, California, USA. October 17-18, 2005.

[11] Désilets, A. and Paquet, S., *Wiki as a tool for Web-based collaborative story telling in primary school: A case study*. In Proceedings of Ed-Media 2005, (Montreal, Canada, June 27-July 2, 2005).

[12] Emigh, W. and Herring, S. *Collaborative authoring on the web: A genre analysis of online encyclopedias*. Proceedings of Hawai'i International Conference on System Sciences, 2005.

[13] Esselink, B. *A Practical Guide to Localization*. John Benjamins, 2000.

[14] Forte and Bruckman. *Why Do People Write for Wikipedia? Incentives to Contribute to Open-Content Publishing?* Proceedings of GROUP 05: International Conference on Supporting Group Work.

[15] Giles, J. *Internet encyclopaedias go head to head*. Nature, December 2005.

[16] Harley, A., Paris, C. *Multilingual Document Production: from Support for Translating to Support for Authoring*. Machine Translation 12 (1-2), pp. 109-29, 1997

[17] Jones, D. et al. *Measuring Human Readability of Machine Generated Text: Three Case Studies in Speech Recognition and Machine Translation*. 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASP 2005). March 18-23, 2005, Philadelphia, PA, USA.

[18] Lamb, B. Wide Open Spaces: *Wikis, Ready or Not*. EDUCAUSE Review, 39 (2004) 5, 36-48.

[19] Leuf, B. and Cunningham, W. *The Wiki Way: Collaboration and Sharing on the Internet*. Addison-Wesley, 2001.

[20] Lih, A. *Wikipedia as Participatory Journalism: Reliable Source? Metrics for evaluating collaborative media as a news resource*. 5th International Symposium on Online Jurnalism, U. of Texas at Austin, April 16-17, 2004.

[21] *Lizzy Wiki.* http://lizzy.iit.nrc.ca/.

[22] Mattison, D. *QuickiWiki, SWiki, TWiki, ZWiki and the Plone Wars: Wiki as a PIM and Collaborative Content Tool*. Searcher: The Magazine for Database Professionals, 23 (2003).

[23] Meatball Collective. Search for pages called: *Meatball Wiki* and *"MultilingualWiki"*. http://www.usemod.com/cgi-bin/mb.pl, (visited on April 2006).

[24] MetaWiki Collective. Search for pages called: *"Multilingual MediaWiki", "Multilingualism", "InterlanguageLinks"* and *"Enhanced Translation.* http://meta.wikimedia.org/ (visited in April 2006)

[25] Patton, J. *Finding the Forest in the Trees*. OOPSLA'05, October 16-20, San Diego, California, USA.

[26] PmWiki Collective. Search for page: *"MultiLanguage"*. http://www.pmwiki.org/ (visited on April 2006)

[27] Sanger, L. *The early history of Nupedia and Wikipedia: a memoir*. In DiBona, C., Stone, M. and Cooper, D., editors, Open Sources 2.0, O'Reilly Press: Sebastopol, CA, 2005.

[28] Schubert, K. *Resource and Workflow Management Support in Teletranslation*. Translating and the Computer 21. Proceedings of the Twenty First International Conference on Translating and the Computer. London, November 1999.

[29] Schubert, K. *Remote-Access Translation Services: Software Design with the User in Focus*. Translating and the Computer 19, ASLIB, London, 1997.

[30] Schutz, J., Nubel, R. *Evaluating Language Technologies: the Mujltidoc Appproach to Taming the Knowledge Soup*. Proceedings of AMTA '98. Langhorne, Pennsylvania, USA, October 28-31, 1998.

[31] Shah, S. *The Internet is Jain: How Gunslingin' Technolibertarianism Leads to Lotus Petals*. In Proceedings of New Forms Festival: Technography, (Vancouver, B.C.2004).