

## **ADAPT IT**

**BRUCE WATERS**

bruce\_waters@sil.org

The author works with SIL Australia and is a programmer with the Adapt It software team.

### **Leveraging bilingualism**

Work began on Adapt It in mid-November 1999. Released first in 2000 via email attachments to people requesting it, it quickly established itself as a helpful product in the Bible translation movement. Later releases have been from a variety of websites, the main site being at <http://adapt-it.org>, generously provided to the Adapt It team by Kahunapule Michael Johnson. Adapt It is open source software, distributed to anyone who wants to use it under typical open source licensing schemes, and is free of charge.

The intent, from the beginning, was that bilingual citizens anywhere in the world would be able to use the software without being intimidated by complexity. User experience, from the feedback over the last decade, has been very positive. Some have had a few difficulties, but overall, users find Adapt It easy to learn and easy to use. Support for user profiles and better features for collaboration with other Bible translation software, to come out in version 6, will make it even more so.

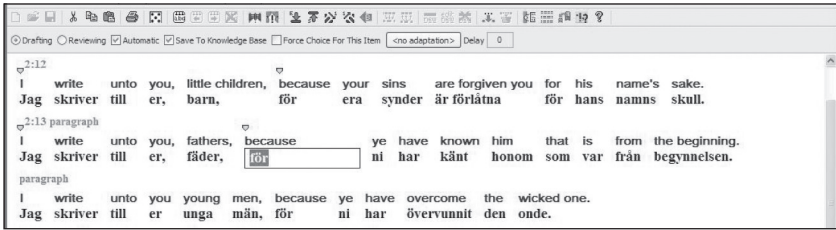
In many parts of the world bilingualism and multilingualism are common. Low cost effective software which leverages this fact did not become available until about a decade ago. People were not unaware of bilingualism, it was just that few people used it as a primary translation resource, and software support was lacking from within Bible agencies. When Adapt It first came on the scene, it was controversial—both within SIL circles and outside them. Many experienced translators and translation consultants felt that, at the very least, standards would drop and the expectation of quality would go unsatisfied. A leader from a major Bible agency, in a 2005 conference, even characterized adaptation technologies, using any software, as “unethical.”

Bilingualism is, however, an enormously important resource for Bible translation. Perceptions, thankfully, have begun to change. After a decade of use, there are now hundreds of projects where Adapt It is the adaptation engine of choice. Other software (for example, Paratext) can now do the same task and adaptation is becoming respectable. A rationale for why it works so well is discussed in a separate paper in this volume.

### **Adapt It—the general design**

Adapt It is an example of a “translation memory” machine. It contains a storage system, called the knowledge base (KB) for use by its adaptation engine. The KB memorizes the associations between an exemplar text word or phrase and

its adaptation equivalent word or phrase. It stores these equivalences in ten special kinds of lists (hash tables), based on the number of words of the exemplar text's phrase. It supports phrasal adaptations of up to ten words in length, but a practical limit for usefulness is about five to six words. It's helpful to keep phrasal adaptations short whenever possible, so that the equivalence has a chance of being used somewhere else when doing the adapting work. An exemplar text phrase of four words may occur hundreds of times in the New Testament, but a ten word one is likely to be unique and thus not reusable.



Adapt It presents to the user an interlinear layout in which to work, as pictured above.\* The exemplar text and the user's adaptations are laid out in "strips." The top line of a strip has the exemplar text, and underneath is a moving text box control, called the "phrase box," where the user may type the target language's adaptation word or phrase. Adapt It keeps the preceding and following context in view by automatically scrolling the display as the phrase box moves. The phrase box expands or contracts automatically as the user adds or removes text. In fact, the whole display is dynamic, responding instantly to what the user does, and keeping the exemplar text and the adaptations aligned vertically. A vertical scroll bar enables the user to scroll from end to end in the adaptation document. There is no drop in responsiveness when working on a large book.

The input to Adapt It is plain text. We encourage the use of Unicode, but a Windows version of the software is also available which does not support Unicode and which can be used with early Windows versions going back to Windows 95. Adapt It is also available for Macintosh and Linux systems but only as a Unicode version. From the start, Adapt It supported standard format markup, and supports the current USFM standard (for which see <http://paratext.ubs-translations.org/about/usfm>). Importing a Paratext Scripture file into Adapt It is normal procedure. The USFM markers are, however, metalanguage information that is not directly relevant to the adaptation process; therefore the software automatically hides the markers from view. It does write their long names in the whiteboard area above each strip, at the location where the particular marker happens to be stored away from sight, as with chapter and verse numbers. For example, in the picture above, the "paragraph" marker would be `\p` in the file from which the document was created; chapter two would be represented by `\c 2`, and verse twelve by `\v 12`. When the adapted text is exported later on, Adapt It will automatically replace the markers at their correct locations in the exported target language text. If it

\* My thanks to Birgitta Häggbladh for graciously supplying the Swedish and German translations in the first two pictures.

doesn't know exactly where it should do so, a dialog will open allowing the user to click where the marker should be, and then click a button to have it placed there. Adapt It saves the KB and documents in Extensible Markup Language (XML) format.

Adapt It also manages punctuation, stripping it from words and phrases before they are stored in the KB, and restoring it automatically. The user can override the punctuation replacement mechanism at any time by typing different punctuation. The settings for punctuation characters in both the exemplar language and the target language are independent, and the user can set up appropriate correspondences between these possibly differing punctuation sets.

Adapt It supports up to four lines per strip. Besides the two mentioned above, there can optionally be a third line for “glosses” which can be in a third language. One use of this feature is to have words and phrases from a published translation in a regional or major language visible, so as to assist the adapter to get a handle on the correct meaning if his grasp of the exemplar text is less than that of a fully fluent bilingual speaker. The fourth line, again optional, is for a free translation of the adaptation text. The free translation can be in any language—typically it is done in the language of the translation consultant contracted to check the adaptation for accuracy. Free translations are entered in a special mode, and Adapt It auto-sections the text into bite-sized chunks using punctuation and the presence of hidden standard format markers to help it do so in a way that is beneficial, although the user does have control over the sections via buttons which are available in that special mode. The picture below shows glossing done in Melanesian Tok Pisin, and a free translation in German.

The screenshot shows the Adapt It software interface. At the top, there is a title bar with the text "Ich habe an euch geschrieben, Kinder,". Below the title bar, there are several buttons: "Shorten", "Lengthen", "Remove", "< Prev", "Next >", "Advance", and "Define Sections By:" with radio buttons for "Punctuation" and "Verse". The main text area contains the following text:

2:12  
 I write unto you, little children, because your  
 Jag skriver till er, barn, för era  
 Mi raitim long yupela liklik pikinini, long wanem? bilong yupela  
 Ich habe an euch geschrieben, Kinder, weil eure Sunden vergeben

sins are forgiven you for his name's sake.  
 synder är förlätna för hans namns skull.  
 ol sin [God] i lusim ol long bilong em nem i go antap moa.  
 sind, um seine Namen willen.

2:13 paragraph  
 I write unto you, fathers, because ye have known him  
 Jag skriver till er, fäder, för ni har känt honom  
 Mi raitim long yupela, ol papa, long wanem? yupela bin save long em  
 Ich habe an euch geschrieben Väter weil ihr ihn, der von anfang an war,

that is from the beginning.  
 som var från begynnelsen.  
 dispela i stap long longtaim bifo.  
 kennelernt habt.

The ability to add a free translation indicates that Adapt It is not just for doing translations between related languages. It can also support the Scripture checking process, and has been used often in that capacity. There are critics of

this, but such criticism is not well founded. Feedback from the field indicates that the KB lookups, which are done with context on either side fully visible, lead to tighter control of the back-translating, better concordance, and a much superior back-translation text overall.

### **The adaptation process**

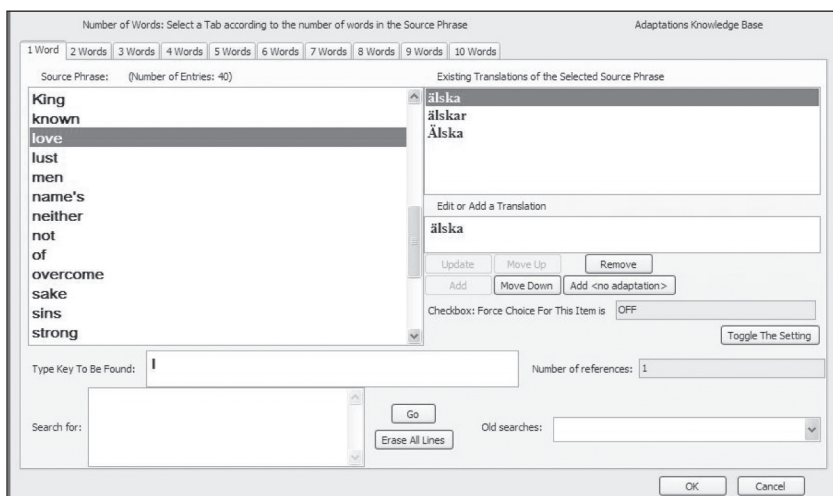
Suppose a person was doing an adaptation manually, without the help of software, between a closely related pair of languages. Lexical similarity is far less important than syntactic similarity; the work gets considerably harder if there are major syntactic differences. What work practices would be involved?

- a. There will be a lot of simple word-for-word lexical substitutions—the relatedness of the two languages leads to that being so.
- b. To some extent (depending on the grammatical closeness of the two languages) a group of words will need to be adapted as a whole unit to a different group of words having the same overall meaning in the target language. These are phrasal adaptations, and they are capable of “hiding” minor syntactic differences between the two languages. For example, one language’s verb complex may be more richly developed than the other’s. But if the verb complex in each language can be handled by a phrasal equivalence each time it occurs in a clause, then the two languages will appear to the adapter to be more similar than they actually are.
- c. There will be constant examination and comparison of the linear flow of the adapted text as it is being composed, comparing and contrasting it with the linear flow of the exemplar text.
- d. Distinctive features of the languages, that is, things present in one but not present in the other, will have an impact. If, for example, the exemplar language has no relative pronouns, but the target language does, then when the adapter comes to a relative clause, she will have to insert a relative pronoun in order to keep the adaptation grammatical and meaningful. Features that occur in the exemplar language but not in the target language will require adjustments. An honorific might simply not be adapted, but other features important for the meaning may require a periphrastic construction.
- e. Occasionally, even after great diligence and much thought, adapting the words and phrases simply won’t be enough to be natural. In this circumstance the manual adapter is likely to abandon following the wording of the difficult section in the exemplar text, and instead “retranslate” the meaning by a different, more natural syntax and more appropriate lexical choices for the target language. Once that difficult section is handled, then she can return to the simpler word and phrasal adaptations.
- f. Periodically, the adapter will want to halt and review what she has written. Does it make sense? Does it convey the same meaning as the exemplar text? Is it natural idiomatic language? Asking herself questions like this may then lead her to backtrack and edit what she has done to get a better result.

These are the kinds of work processes that would be required when doing manual adaptation, for example, working with a pencil and notebook from a printed Bible in a related language.

What Adapt It does is to provide support for these kinds of work procedures, while managing the data storage as well. In drafting mode, Adapt It considers text up to ten words ahead of where the phrase box currently is. It then quickly makes ten phrases and looks in the KB for the longest possible match. If it finds a match, and the adaptation in the KB is unique, it will insert it, and move the phrase box on to the next empty location, continuing this lookup and insert cycle until a halt is forced upon it.

The KB lookup and insert function is faster than the eye can follow, so checking of what was automatically inserted relies on halts happening. Fortunately, halts happen often. In addition, Adapt It has a “Delay” button which allows the user to set a delay of up to three seconds per insertion, if the user prefers a more sedate pace. Since every time a new equivalence is entered into the KB it becomes immediately available for lookup in the text which lies ahead of the phrase box, automatic insertions become more and more common as the KB progressively becomes more populated.



When a translator first starts using Adapt It, the KB is not populated at all, and the translation experience is little different from typing into a word processor. (Pre-populating the KB is possible if suitable data are available, but generally such data don't exist.) As the user works, the KB quickly becomes populated, and within a week, a significant number of words in each verse will be automatically inserted. Within a month, maybe an average of a quarter of each verse is automatically generated. Within several months, the majority of each verse will be auto-generated. When insertions are made by the software, they are given background highlighting so that the user can easily see which material has to be reviewed for accuracy before she attempts to adapt more text.

The phrase box stops its automatic lookups and insertions whenever (a) it comes to an exemplar text word or phrase not in the KB already, or (b) when the KB has two or more adaptations stored for the one exemplar text word or phrase. In the case of (b), this halt is valuable, because Adapt It automatically opens a dialog which lists all of the possible adaptations of that word or phrase already “seen” and stored from the user’s earlier work. The user can then click to insert one of the listed entries, or type a new one which will be inserted into the document at that point and also stored in the KB. The user’s choice for what word or phrase is appropriate is informed by the preceding and following context, which is fully visible on the screen. Adapt It promotes a linear workflow style which requires the user to make explicit, contextually defined decisions. This particular feature, incidentally, is not shared by Paratext. Paratext analyses a whole verse at once, and only gives a “best guess” adaptation even when there are contextual variants possible. While contextual variants can be viewed and selected by clicking on the words of the text, within Paratext it is up to the user to notice and change it if the wording is not appropriate for the context.

How then does Adapt It support the kinds of work procedures for a manual adaptation listed above?

1. The work flow within Adapt It generally proceeds in linear fashion—the same way that all language is ultimately expressed, regardless of its hierarchical internal structure.
2. The user can locate the phrase box anywhere in the document, at will. She simply clicks underneath the exemplar word or phrase where she wants the box to be, whether there is adaptation text there or not. The box immediately goes there and the display adjusts. This feature makes it simple to edit mistakes or choose a different word or phrase if review of the meaning suggests that something else would make better sense. (Editing done this way automatically produces a corresponding update in the KB contents, without the user doing anything explicitly to make it happen. The KB can, however, be inspected and edited independently of the document contents at any time.)
3. The user presses the Enter key (or the Tab key) on the keyboard to “sign off” on an adaptation she has entered into the phrase box. She is in no way restricted in the number of words she may enter. The exemplar text at that point may just be a single word, but she can type an adaptation which is a phrase. Similarly, if she designates a group of words in the exemplar text as a phrase (Adapt It calls these a “merger,” because several words are being merged together to form a group), she can adapt it as a single word, or as a phrase with fewer or more words than the exemplar text phrase. When she signs off by pressing the Enter key, a number of things happen. First, the exemplar-to-target phrasal equivalence is entered into the KB (which consists most often of just simple “single-word-for-single-word” equivalences). Second, Adapt It looks ahead for an empty location, and generates all the exemplar phrase possibilities there, and looks them up in the KB, and inserts the longest possible translation equivalence if there is a unique one, and moves on, repeating this process until there is a halt. Otherwise, if no unique renderings are found, Adapt It either shows the list of possible adaptations available, or if there are none, it halts

and waits for the user to type something. Though this process takes many sentences to describe, for the user it is very simple and transparent. It is the most common work procedure and is continuously repeated during adaptation. It corresponds to points a, b, and c above. The before and after pictures below illustrate a two-word merger: first select the words, then type the adaptation. Adapt It groups the selection into a phrase automatically.

2:15

and	ye	have	overcome	the	wicked	one.	Love	not
och	ni	har	överbunnit	den	wicked			

2:15

and	ye	have	overcome	the	wicked one.	Love	not
och	ni	har	överbunnit	den	onde		

4. Differences between the exemplar language and the target language sometimes require more than 1, 2, and 3 above. For example, doing a back-translation from Melanesian Tok Pisin, which lacks relative pronouns, into English, which uses them, would be a problem without the Adapt It “placeholder” feature. The ability to insert a placeholder ... (three dots) into the exemplar text anywhere it is needed allows the user to add the additional word or phrase that the target text needs in order to be grammatical.

long	dispela	man	...	i stap	bipo tru	na
about	this	man	who	was	in the	beginning and

5. Another feature which helps the user handle thorny language differences is the “Retranslation” feature. The user simply selects the group of words of the exemplar text which are particularly difficult to translate adequately by adapting the individual words and phrases therein. Then she presses a command button and a dialog opens which allows her to type whatever target language words are required in order to convey the meaning accurately. She can type fewer, the same number, or more words than she selected. If she types more than she selected, and “signs off” with a click of the OK button, Adapt It will automatically insert extra placeholders at the end of the selected exemplar text words, and lay out her typed words underneath them. Retranslations are not entered into the KB, as they tend to be too long to be reusable anywhere else.
6. Finally, to help the user see what Adapt It has done, anything it automatically inserts is given background highlighting in a light mauve colour. (The colour can be easily changed to the user’s preferred colour.)

### Other features

Adapt It has a Notes feature. Each note can be up to 30 Kb in size. The notes are automatically hidden, but can be shown on demand with the click of a small icon above the word in the strip where the note is located.

Adapt It exports the adaptation as plain text, with USFM markers automatically restored to their proper places in the target language text. It can also export the adaptation in Rich Text Format (RTF), for which it uses a Scripture styles template, to achieve a publication-like output. It can also export



the interlinear display in automatically-sized tables. The latter type of export has options for how to handle free translations, notes, and glosses—usually as footnotes, but some types of information can also be exported as marginal comments. The next release, version 6, will also support Lexicon Interchange Format (LIFT) for export and import of the KB data, and Open XML for Editing Scripture (OXES) export of the adaptation along with a free translation and Adapt It notes, if the latter are present in the document. More export formats will be supported in future versions, for example, to support e-publishing.

Adapt It typically stores its data in the default data folder for the host machine's operating system, in a folder called Adapt It Unicode Work. Recent versions allow this location to be elsewhere. Commands on the Administrator menu permit the work folder to be located anywhere and "locked" to that location, including on an encrypted drive within the machine or attached externally.

A recent trend in software for field use is to divide tasks between those that a person without a lot of previous computing experience can do and those that a highly experienced person can do. Examples of software that supports this distinction include WeSay, a lexicography application, and OurWord, a translation tool. Adapt It will, from version 6, support this distinction in two ways. First, the administrator will be able to give a command to restrict Adapt It to looking for files to be loaded for adaptation in a single folder called "Source Data," within the current adaptation project's folder. Thereafter, the user will not be able to change where files are obtained from, nor where they are stored. This will protect new users from the stress of apparent loss of work, when all that has happened is that the files have become lost in the hierarchical file system of the operating system. Second, there will be "User Profiles"—the ability to hide certain command options, with predetermined profiles labelled Novice, Experienced, Skilled, and Custom. Each of these can be adjusted to show or hide additional menu commands besides those included in each profile by default. Using none of these profiles results in all menus being available. The more complex Adapt It becomes, the more important it will be to hide from view functionality not needed for the task at hand, thus keeping the application from becoming intimidating. Training can then be given in stages, with more of the application's features being turned on at each progressive stage.

From the beginning, Adapt It has been able to preprocess a source text using up to four consistent change tables in series. This process is optional, but can be a very powerful tool for a dialect adaptation situation where there are highly predictable sound or orthographic changes and/or consistent lexical changes between the exemplar text and the target text. A few years ago this preprocessing capability was extended to use the SIL Encoding Converters functionality where desired. The forthcoming version 6 will add to this preprocessing a statistical guessing capability for target forms.

Since Adapt It version 4 was released in 2008, we have also released Linux and Macintosh versions of Adapt It which run as efficient native applications on those platforms, positioning Adapt It as a very responsive tool for low power computing environments designed for use in small rural solar-powered programs.



## Future plans

Once the present line of development is completed, around mid 2011, the Adapt It team plans to support various collaboration scenarios. Prediction is risky, but we want to support as soon as possible some or all of the following:

- Allow multiple machines to run Adapt It, with all sharing their KBs across the local area network.
- Provide the same functionality via the web—with a server somewhere “in the cloud” storing the KB for multiple users within the one project who may be anywhere in the world.
- Move both the software and the KB storage for language pairs to the web, so that anyone with a simple computer can work in a web browser to do adaptation work.
- Make similar functionality available on the increasingly capable smart phones, tablets, and other hand-held devices.
- We also want to support Send/Receive—similar to what Paratext and other tools do. And e-publishing has not escaped our notice either.

Adapt It development is not an official software project of SIL International. The team members are a small group of volunteers, mainly from within SIL. Two are full-time, a third full-time member will commence at a future date when circumstances allow. There are three part-time developers and a retired Wycliffe Associates person who helps with training and the production of the online helps. Despite our low profile, we are dedicated to providing this software to those who need it, indefinitely into the future, as the Lord permits.