

## **Electronic dictionaries for language reclamation.**

**James McElvenny and Aidan Wilson, University of Sydney.**

### **1. Introduction**

This talk presents *Wunderkammer*, a program for storing and displaying multimedia electronic dictionaries on mobile phones. This software has been developed within the Project for Free Electronic Dictionaries (<http://www.pfed.info/>) at the University of Sydney, a project primarily aimed at making electronic dictionaries of Australian Aboriginal languages available to their speakers.

We start in section 2 with a brief look at the rationale for making electronic dictionaries available on mobile phones. Section 3 presents a view of *Wunderkammer* from the user's perspective. Section 4 looks at the *Wunderkammer* dictionary file format and *wkimport*, a program for importing existing dictionaries into *Wunderkammer*.

### **2. The rationale for *Wunderkammer***

Through informal observation we noticed that mobile phones are very widespread in Aboriginal communities in Australia. In many communities almost everyone has a mobile phone that they keep with them most of the time, and most people, especially the young, are very familiar with their phones and comfortable with operating them. Most of the phones support the installation of third-party software that runs on the Java ME (Micro Edition) platform, so it is possible for us to write our own software and install it on the phones. We realised that mobile phones were the perfect hosts for electronic dictionaries – all we had to do was write the software.

### **3. A tour of *Wunderkammer***

The dictionary shown in the following tour through the functions of the *Wunderkammer* software is an electronic dictionary of the Kaurna language (Adelaide, South Australia). A demonstration version of the Kaurna electronic dictionary can be downloaded from the *Wunderkammer* website (<http://www.pfed.info/wksite>). The content of the Kaurna demonstration dictionary is

made available with the permission of Kurna Warra Pintyandi (<http://www.adelaide.edu.au/kwp/>), the Kurna language group.

The entry point to *Wunderkammer* is the menu system, which allows users to navigate through a dictionary to find entries. There can be any number of menus, which can be embedded within each other to an indefinite depth. Menus can be produced on any field in an entry. In the *Wunderkammer* version of the Kurna dictionary, for example, we included a menu based on headwords, a menu based on glosses for a reverse index and a menu of semantic domains. These can be seen in Figure 3 below.



Figure 3. The main menu of the Kurna dictionary in *Wunderkammer*

A search box is provided for menus that contain a large number of items. The user can enter text into the search box to limit the number of items shown in the menu. Only menu items that contain the search text will be shown. In the example shown in Figure 4 below only items that contain the text 'ta' are shown.



Figure 4. Search function in *Wunderkammer*

Once the user reaches the bottom level of the menu system, they are taken to an entry. In the entry view all the fields that belong to an entry are shown. The user can scroll up and down through the contents of the entry. Each entry can contain one sound and any number of images. Entries can also contain links to menus or other entries. The user can jump to the menu or entry that the link points to by clicking on the link. The layout of a typical entry in *Wunderkammer* can be seen in Figure 5 below.



Figure 5. Entry view in *Wunderkammer*

#### 4. The *Wunderkammer* file format and importing dictionaries

In order to minimise the space required for storing dictionary content and the processing overhead for reading it, we used a custom binary format for storing dictionary data in *Wunderkammer*. To make it easy to convert existing dictionary files into this custom format we developed the program *wkimport*, which currently supports the conversion of files in XML and FOSF (the backslash-coded format used by *Shoebox/Toolbox* from SIL). It has been designed in such a way that it should be simple for Java programmers to write their own extension modules to convert files from other formats.

Note that we introduced the custom format used in *Wunderkammer* only as a means of getting the best performance out of the program on the very limited Java ME platform. The *Wunderkammer* format is not a suitable archival format. It does not implement any widely adopted open standard and it allows only a very simple dictionary entry structure that is tied to the implementation of *Wunderkammer*. In all the dictionaries that have been converted for use with *Wunderkammer* we maintain a master dictionary file as the main form of storage for the dictionaries and convert these master files for use with *Wunderkammer*. The conversion process is made simple by *wkimport*.

The structure of dictionary entries in the *Wunderkammer* file format has been made maximally simple to minimise storage requirements and to keep the design of the user interface as simple as possible. The possibilities for user interfaces on mobile phones are somewhat more limited than those on computers. Mobile phone screens are much smaller than computer screens and mobile phone keypads can be difficult to use. In order to keep the design of the user interface simple, we found it necessary to limit the number of sounds that can be associated with an entry to one and to remove any internal hierarchical structure from entries.

With only one sound per entry, it is possible to have a single button to press to play the entry sound, whereas if there were multiple sounds per entry a more complex user interface would have to be implemented to allow the user to select the sound to play. Of all the types of data that can be stored in a *Wunderkammer* dictionary, sounds tend to be the largest. Limiting the number of sounds to one per entry also helps to

prevent the problem of dictionary files growing too big and exceeding the storage capacity of phones.

We found it difficult to develop a user interface that could display hierarchical structure within entries on the small mobile phone screens. Eliminating hierarchical structure within entries also reduces, if only slightly, the size of dictionary files. Note that even though we rejected hierarchical structure within entries for *Wunderkammer*, it would be better to be able to include this information. Hopefully as the mobile phone upgrade cycle continues its inevitable course, newer phones with greater storage capacity and facilities for better user interfaces will become available. This should make it possible to implement entries with more complex structure in future versions of *Wunderkammer*.

As indicated above, the limited storage capacity of most current models of mobile phones presents a particular problem for the inclusion of multimedia content in *Wunderkammer* dictionaries. Multimedia content tends to take up a lot of space and dictionaries that contain many sounds and images may exceed the storage capacity of lower end mobile phones. Dictionaries with little or no multimedia content do not normally exceed the capacity of even the smallest mobile phones, however. The largest dictionary ported to *Wunderkammer* so far is a dictionary of the language Dalabon (Arnhem Land, Northern Territory), with 3,634 entries and no multimedia content. This dictionary, when packaged with the *Wunderkammer* program files, takes up only 878 KB, which runs even on phones with the lowest specifications.

As with future improvements to the *Wunderkammer* user interface and file format, restrictions on the amount of multimedia content that can be included in dictionaries should ease as newer models of mobile phones become available. At the moment, however, dictionary creators need to consider what mobile phones the potential users of a dictionary have and to make sure that the dictionary does not exceed the storage capacity of the lower end phones. In communities where there is a mix of models, it would be possible to produce more than one version of a dictionary. Some versions could have all multimedia content, some only part, and others no multimedia.

## 5. Conclusion

This talk has given an overview of the *Wunderkammer* software for storing and displaying electronic dictionaries on mobile phones, both from a user's perspective and from the perspective of a dictionary creator. The *Wunderkammer* software, *wkimport* dictionary importing tool and detailed documentation for these programs are available from the *Wunderkammer* website: <http://www.pfed.info/wksite>