A Generic Collaborative Platform for Multilingual Lexical Database Development

Gilles SÉRASSET
GETA-CLIPS, IMAG, Université Joseph Fourier
BP 53 – 38041 Grenoble cedex 9 – France
Gilles.Serasset@imag.fr

Abstract

The motivation of the Papillon project is to encourage the development of freely accessible Multilingual Lexical Resources by way of online collaborative work on the Internet. For this, we developed a generic community website originally dedicated to the diffusion and the development of a particular acception based multilingual lexical database.

The generic aspect of our platform allows its use for the development of other lexical databases. Adapting it to a new lexical database is a matter of description of its structures and interfaces by way of XML files. In this paper, we show how we already adapted it to other very different lexical databases. We also show what future developments should be done in order to gather several lexical databases developers in a common network.

1 Introduction

In order to cope with information available in many languages, modern information systems need large, high quality and multilingual lexical resources. Building such a resource is very expensive. To reduce these costs, we chose to use the "collaborative" development paradigm already used with LINUX and other open source developments.

In order to develop such a specific multilingual lexical database, we built a Web platform to gather an Internet community around lexical services (accessing many online dictionaries, contributing to a rich lexical database, validating contributions from others, sharing documents, ...). Initially built for the Papillon project, this platform is generic and allows for the collaborative development of other lexical resources (monolingual, bilingual or multilingual) provided that such resources are described to the platform.

After presenting the Papillon project and platform, we will show how we may give access to many existing dictionaries, using an unified interface. Then, we will present the edition service, and detail how it may be customised to handle other very different dictionaries.

2 The Papillon project

2.1 Motivations

Initially launched in 2000 by a French-Japanese consortium, the Papillon project[1] (Sérasset and Mangeot-Lerebours, 2001) rapidly extended its original goal — the development of a rich French Japanese lexical database — to its actual goal — the development of an Acception based Multilingual Lexical Database (currently tackling Chinese, English , French, German, Japanese, Lao, Malay, Thai and Vietnamese).

This evolution was motivated in order to:

- reuse many existing lexical resources even the ones that do not directly involve both initial languages,
- be reusable by many people on the Internet, hence raising the interest of others in its development,
- allow for external people (translator, native speakers, teachers...) to contribute to its development,

For this project, we chose to adopt as much as possible the development paradigm of LINUX and GNU software[2], as we believe that the lack of high level, rich and freely accessible multilingual lexical data is one of the most crucial obstacle for the development of a truly multilingual information society[3].

[1] http://www.papillon-dictionary.org/

[2] i.e. allowing and encouraging external users to *access and contribute to* the database.

[3] i.e. an Information Society with no linguistic domination and where everybody will be able to access any content in its own mother tongue.

2.2 Papillon acception based multilingual database

The Papillon multilingual database has been designed independently of its usage(s). It consists in several monolingual volumes linked by way of a single interlingual volume called the interlingual acception dictionary.

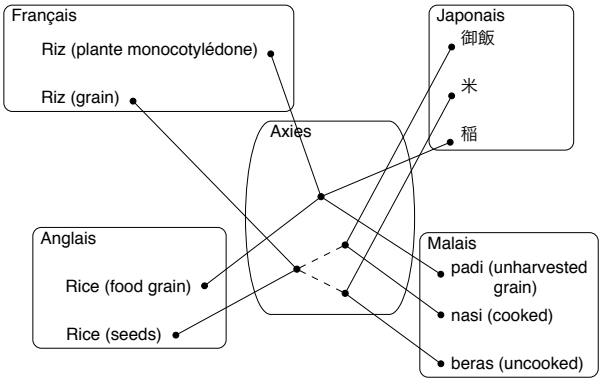

Figure 1: Macrostructure of the Papillon MLDB, showing the handling of contractive problems.

Each monolingual volume consists in a set of word senses (*lexies*), each lexie being described using a structure derived from the Explanatory and Combinatory Dictionary (Mel'čuk et al., 1995; Mel'čuk et al., 1984 1989 1995 1996).

The interlingual acception dictionary consists in a set of interlingual acceptions (*axies*) as defined in (Sérasset, 1994). An interlingual acception serves as a placeholder bearing links to lexies and links between axies[4]. This simple mechanism allows for the coding of translations. As an example, figure 1 shows how we can represent a quadrilingual database with contrastive problems (on the well known "rice" example).

2.3 Development methodology

The development of the Papillon multilingual dictionary gathers voluntary contributors and trusted language specialist involved in different tasks (as shown in figure 2).

- First, an automatic process creates a draft acception based multilingual lexical database from existing monolingual and bilingual lexical resources as shown in (Teeraparseree, 2003; Mangeot-Lerebours et al., 2003). This step is called the *bootstrapping* process.

[4]Note that these links are not interpreted semantically, but only reflect the fact that translation is possible

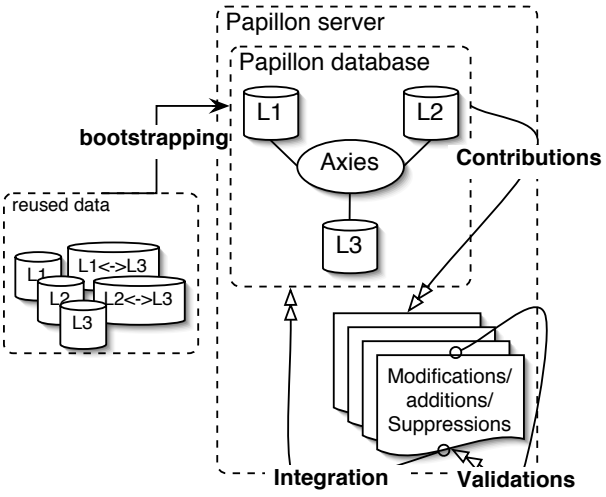

Figure 2: Methodology for the development of the Papillon database.

- Then, *contributions* may be performed by volunteers or trusted language specialists. A contribution is either the modification of an entry, its creation or its deletion. Each contribution is stored and immediately available to others.
- Volunteers or language specialist may *validate* these contributions by ranking them.
- Finally, trusted language specalists will *integrate* the contribution and apply them to the master MLDB. Rejected contributions won't be available anymore.

2.4 The Papillon Platform

The Papillon platform is a community web site specifically developed for this project. This platform is entirely written in Java using the "Enhydra[5]" web development Framework. All XML data is stored in a standard relational database (Postgres). This community web site proposes several services:

- a unified interface to simultaneously *access* the Papillon MLDB and several other monolingual and bilingual dictionaries;
- a specific edition interface to *contribute* to the Papillon MLDB,
- an open document repository where registered users may share writings related to the project; among these documents, one may find all the papers presented in the

[5]see http://www.enhydra.org/

different Papillon workshops organized each year by the project partners;

- a mailing list archive,

Sections 3 and 4 present the first and second services.

3 Unified access to existing dictionaries

3.1 Presentation

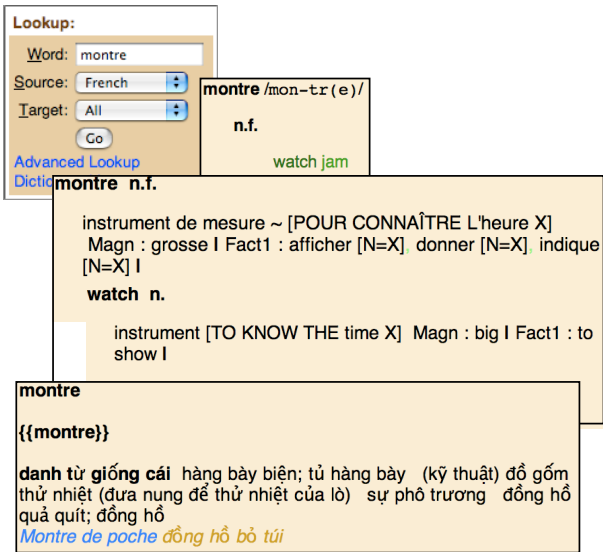

Figure 3: The unified access interface and results from three different dictionaries

To encourage volunteers, we think that it is important to give a real service to attract as many Internet users as possible. As a result, we began our development with a service to allow users to access to many dictionaries in a unified way. This service currently gives access to twelve (12) bilingual and monolingual dictionaries, totalizing a little less than 1 million entries, as detailled in table 1.

3.2 Strong points

The unified access interface allows the user to access simultaneously to several dictionaries with different structures. All available dictionary will be queried according to its own structure. Moreover, all results will be displayed in a form that fits its own structure.

Any monolingual, bilingual or multilingual dictionary may be added in this collection, provided that it is available in XML format.

With the Papillon platform, giving access to a new, unknown, dictionary is a matter of writing 2 XML files: a dictionary description and an XSL stylesheet. For currently available dictionaries, this took an average of about one hour per dictionary.

Dictionary	Languages	Nb of Entries
Armament[a]	fra eng	1116
Cedict[b]	zho eng	215424
Ding[c]	deu eng	124413
Engdict[d]	eng kor	214127
FeM[e]	fra eng msa	19247
Homerica[f]	fra	441
JMDict[g]	jp en fr de	96264
KanjiDict[h]	jpn eng	6355
Papillon	multi	1323
ThaiDict[i]	tha	10295
VietDict[j]	fra vie	41029
WaDokuJiTen[k]	jpn deu	214274

[a] Japanese French dictionary of armament from the French Embassy in Japan
[b] Chinese English from Mandel Shi (Xiamen univ.)
[c] (Richter, 1999)
[d] (Paik and Bond, 2003)
[e] (Gut et al., 1996)
[f] University Stendhal, Grenoble III
[g] (Breen, 2004a)
[h] (Breen, 2004b)
[i] Thai Dictionary of Kasetsart University
[j] (Duc, 1998)
[k] (Apel, 2004)

Table 1: Dictionaries available through the unified access interface

3.3 Implementation

It is possible to give access to any XML dictionary, regardless of its structure. For this, you have to identify a minimum set of information in the dictionary's XML structure.

The Papillon platform defines a standard structure of an abstract dictionary containing the most frequent subset of information found in most dictionaries. This abstract structure is called the Common Dictionary Markup (Mangeot-Lerebours and Sérasset, 2002). To describe a new dictionary, one has to write an XML file that associate CDM element to pointers in the original dictionary structure.

As an example, the French English Malay FeM dictionary (Gut et al., 1996) has a specific structure, illustrated by figure 4.

Figure 5 gives the XML code associating elements of the FeM dictionary with elements of the CDM.

Along with this description, one has to de-

```
<HFEM xmlns:xml="http://www.w3.org/.../namespace">
  <HW-FRE>montre</HW-FRE>
  <HOM/>
  <PRNC>mon-tr(e)</PRNC>
  <AUX/>
  <BODY>
    <SENSE-STAR>
      <SENSE>
        <CAT-STAR>n.f.</CAT-STAR>
        <SENSE1-STAR>
          <SENSE1>
            <TRANS-STAR>
              <TRANS>
                <ENG-STAR>watch</ENG-STAR>
                <MAL-STAR>jam</MAL-STAR>
              </TRANS>
            </TRANS-STAR>
            <EXPL-STAR/>
          </SENSE1>
        </SENSE1-STAR>
      </SENSE>
    </SENSE-STAR>
  </BODY>
</HFEM>
```

Figure 4: A simplified example entry from the French English Malay FeM dictionary.

```
<cdm-elements>
  <cdm-volume element="volume"/>
  <cdm-entry element="HFEM"/>
  <cdm-headword element="HW-FRE"/>
  <cdm-pronunciation element="PRNC"/>
  <cdm-pos element="CAT-STAR"/>
  <cdm-definition element="FRE"/>
  <cdm-translation d:lang="eng"
                   element="ENG-STAR"/>
  <cdm-translation d:lang="msa"
                   element="MAL-STAR"/>
  <cdm-example d:lang="fra" element="FRE"/>
  <cdm-example d:lang="eng" element="ENG"/>
  <cdm-example d:lang="msa" element="MAL"/>
  <cdm-key1 element="HOM"/>
</cdm-elements>
```

Figure 5: Associations between elements of the FeM dictionary and elements of the CDM.

fine an XSL style sheet that will be applied on requested dictionary elements to produce the HTML code that defines the final form of the result. If such a style sheet is not provided, the Papillon platform will itself transform the dictionary structure into a CDM structure (using the aforementioned description) and apply a generic style sheet on this structure.

4 Editing dictionaries entries

4.1 Presentation

As the main purpose of the Papillon platform is to gather a community around the *development* of a dictionary, we also developed a service for the edition of dictionary entries.

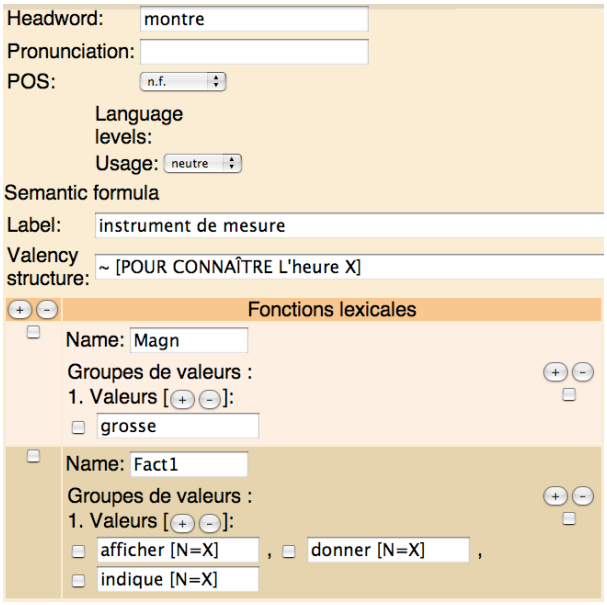

Figure 6: The edition interface is a standard HTML interface

Any user, who is registered and logged in to the Papillon web site, may contribute to the Papillon dictionary[6] by creating or editing[7] an entry. Moreover, when a user asks for an unknown word, he is encouraged to contribute it to the dictionary.

Contribution is made through a standard HTML interface (see figure 6). This interface is rather crude and raises several problems. For instance, there is no way to copy/paste part of an existing entry into the edition window. Moreover, editing has to be done on-line[8]. However, as the interface uses only standard HTML elements with minimal javascript functionality, it may be used with any Internet browser on any platform (provided that the browser/platform correctly handles unicode forms).

4.2 Strong points

From the beginning, we wanted this interface to be fully customizable by Papillon members

[6] And, for now, only to this particular dictionary.

[7] Removal of an entry is not yet implemented.

[8] In fact, entries may be edited off-line and uploaded on the server, but there is currently no specialized interface for off-line edition, meaning that users will have to use standard text/XML editor for this.

without relying on the availability of a computer science specialist. our reasons are:

- the fact that we wanted the structure of the Papillon dictionary to be adaptable along with the evolution of the project, without implying a full revisit of the web site implementation;
- the fact that each language may slightly adapt the Papillon structure to fit its own needs (specific set of part of speech, language levels, etc.), hence adding a new dictionary implies adding a new custom interface;

Hence, we chose to develop a system capable of generating a usable interface from a) a description of the dictionary structure (an XML Schema) and b) a description of the mapping between element of the XML structure and standard HTML inputs.

For this, we used the ARTStudio tool described by (Calvary et al., 2001). Using a tool that allows for the development of plastic user interfaces allows us to generate not only one, but several interfaces on different devices. Hence, as we are now able to generate an HTML interface usable with any standard web browser supporting Unicode, we may, in the future, generate interfaces for Java applications (that can be used offline) or interfaces for portable devices like pocket PCs or Palm computers.

4.3 Implementation

4.3.1 Definition of the dictionary structure

To provide an edition interface, the Papillon platform needs to know the exact dictionary structure. The structure has to be defined as a standard XML schema. We chose to use XML schema because it allows for a finer description compared to DTDs (for instance, we may define the set of valid values of the textual content of an XML element). Moreover XML schemata provides a simple inheritance mechanism that is useful for the definition of a dictionary. For instance, we defined a general structure for the Papillon dictionary (figure 7) and used the inheritance mechanism to refine this general structure for each language (as in figure 8).

4.3.2 Description of the interface

Describing the interface is currently the most delicate required operation. The first step is to define the set of elements that will appear in the interface and their relation with the dictionary structure. Each such element is given a unique ID. This step defines an abstract interface where all elements are known, but not their layout, nor their kind.

This step allows for the definition of several different tasks for the edition of a single dictionary.

The second step is to define the concrete realization and the position of all these elements.

```
<element name="lexie">
 <complexType>
  <sequence>
   <element ref="d:headword" minOccurs="1"
                             macOccurs="1" />
   <element ref="d:writing" ... />
   <element ref="d:reading" ... />
   <element ref="d:pronunciation" ... />
   <element ref="d:pos" ... />
   <element ref="d:language-levels" ... />
   <element ref="d:semantic-formula" ... />
   <element ref="d:government-pattern" .../>
   <element ref="d:lexical-functions" ... />
   <element ref="d:examples" ... />
   <element ref="d:full-idioms" ... />
   <element ref="d:more-info" ... />
  </sequence>
  <attribute ref="d:id" use="required" />
 </complexType>
</element>
...
<element name="pos" type="d:posType" />
<simpleType name="posType">
  <restriction base="string" />
</simpleType>
...
```

Figure 7: General structure shared by all volumes of the Papillon dictionary; showing the part of speech element `pos` defined as a textual element.

```
<simpleType name="posType">
  <restriction base="d:posType">
    <enumeration value="n.m." />
    <enumeration value="n.m. inv." />
    <enumeration value="n.m. pl." />
    <enumeration value="n.m., f." />
    <enumeration value="n.f." />
    <enumeration value="n.f. pl." />
    ...
  </restriction>
</simpleType>
```

Figure 8: Redefinition of the type of the part of speech `pos` element in the Papillon French definition.

For instance, in this step, we specify the POS element to be rendered as a menu. Several kind of widgets are defined by ARTStudio. Among them, we find simple HTML inputs like text boxes, menus, check-boxs, radio buttons, labels..., but we also find several high level elements like generic lists of complex elements.

As an simple example, we will see how the `pos` (part of speech) element is rendered in the Papillon interface. First, there will be an interface element (called S.364) related to the `pos` element (figure 9). Second, this element will be realized in our interface as a comboBox (figure 10).

```
<Instance type="element" id="S.364">
 <InstanceKind value="static"/>
 <InstanceBuildKind value="regular"/>
 <Name value="pos"/>
 <ClassNameSpace value=""/>
 <ClassName value="posType"/>
 <TaskOwnerID value="S.360"/>
 <TaskRangeID list="S.360"/>
</Instance>
```

Figure 9: Definition of the abstract interface element associated to the `pos` element. This element will display/edit value of type `posType` defined in the aforementioned schema.

```
<Interactor type="element"
            class="GraphicInteractor" id="i2008">
 <Type value="presentation"/>
 <TaskID value="S.363"/>
 <InteractorID value="ComboBox"/>
 <InstanceID value="S.364"/>
 <Width value="10"/>
 <Height value="20"/>
</Interactor>
```

Figure 10: Definition of the effective widget for the `pos` element.

Using this technique is rather tricky as there is currently no simple interface to generate these rather complex descriptions. However, using these separate description allows the definition of several edition tasks (depending on the user profile) and also allows, for a single task, to generate several concrete interfaces, depending on the device that will be used for edition (size of the screen, methods of interactions, etc.).

4.3.3 Interface generation

Using the describe structure of the dictionary, we are able to generate an empty dictionary entry containing all mandatory elements. Then, we walk this structure and instantiate all associated widgets (in our case HTML input elements), as defined in the interface description. This way, we are able to generate the corresponding HTML form.

When the user validates a modification, values of the HTML input elements are associated to the corresponding parts of the edited dictionary structure (this is also the case if the user asks for the addition/suppression of an element in the structure). Then, we are able to regenerate the interface for the modified structure. We iterate this step until the user saves the modified structure.

5 Conclusions

The Papillon platform is still under development. However, it already proves useful for the *diffusion* of a little less than 1 million entries from 12 very different dictionaries. This is possible as, from the very beginning, we designed the platform to be as a generic as possible.

This genericity also allows for its use for the *on-line development* of the Papillon database. It is also used for the development of the Estonian French GDEF dictionary, managed by Antoine Chalvin from INALCO, Paris. Moreover, we developed an interface for the japanese German WadokujiTen (Apel, 2004). This proves that our platform may be useful in a general context.

Our future activities will follow 3 axis:

- improving the definition of edition interfaces; currently, we have no tool to simplify this definition and its complexity makes it difficult for a linguist to use it without help from computer science specialists;
- generating different interfaces from the same descriptions; currently, we only generate on-line HTML interfaces, but the tools we use allows for the development of interfaces in other contexts; hence with the same approach, we will develop java applets or java applications to be used either on-line or off-line;
- developing network cooperation modules between several instances of the Papillon platform; this will allow the deployment of the platform on several sites; we will address two aspects of such a deployment; first, duplication of identical instances providing access and edition services on the same dictionaries; second the deployment

of several instances providing access and edition services on different dictionaries (where dictionaries edited on a site may be accessed on another site).

6 Acknowledgements

Developments on the Papillon project could not have taken place without support from CNRS (France) and JSPS (Japan). We would like to warmly thank François Brown de Colstoun who supports this project since its very beginning. Developments of the platform and especially the editing part has been mainly done by Mathieu Mangeot and David Thevenin during their Post Doctoral fellowship at NII (National Institue of Informatics), Tokyo. Finally the Papillon platform would not be useful without partners who agreed to give free access to their superb dictionaries.

References

Ulrich Apel. 2004. WaDokuJT - A Japanese-German Dictionary Database. In *Papillon 2002 Workshop on Multilingual Lexical Databases*, NII, Tokyo, Japan, 6-18 July.

Jim W. Breen. 2004a. JMdict: a Japanese-Multilingual Dictionary. In Gilles Sérasset, Susan Armstrong, Christian Boitet, Andrei Pospescu-Belis, and Dan Tufis, editors, *post COLING Wokshop on Multilingual Linguistic Resources*, Geneva, Switzerland, 28th august. International Committee on Computational Linguistics.

Jim W. Breen. 2004b. Multiple Indexing in an Electronic Kanji Dictionary. In Michael Zock and Patrick St Dizier, editors, *post COLING workshop on Enhancing and Using Electronic Dictionaries*, Geneva, Switzerland, 29th august. International Committee on Computational Linguistics.

Gaëlle Calvary, Joëlle Coutaz, and David Thevenin. 2001. A unifying reference framework for the development of plastic user interfaces. In M. Reed Little and L. Nigay, editors, *Engineering for Human-Computer Interaction: 8th IFIP International Conference, EHCI 2001*, volume 2254 / 2001 of *Lecture Notes in Computer Science*, page 173. Springer-Verlag Heidelberg, Toronto, Canada, May.

Ho Ngoc Duc, 1998. *Vietnamese French Online Dictionary*. http://www.informatik.uni-leipzig.de/~duc/Dict/.

Yvan Gut, Puteri Rashida Megat Ramli, Zaharin Yusoff, Kim Choy Chuah, Salina A. Samat, Christian Boitet, Nicolai Nedobejkine, Mathieu Lafourcade, Jean Gaschler, and Dorian Levenbach. 1996. *Kamus Perancis-Melayu Dewan, Dictionnaire francais-malais*. Dewan Bahasa dan Pustaka, Kuala Lumpur.

Mathieu Mangeot-Lerebours and Gilles Sérasset. 2002. Frameworks, implementation and open problems for the collaborative building of a multilingual lexical database. In Grace Ngai, Pascale Fung, and Kenneth W. Church, editors, *Proc. of SEMANET Workshop, Post COLING 2002 Workshop*, pages 9–15, Taipei, Taiwan, 31 August.

Mathieu Mangeot-Lerebours, Gilles Sérasset, and Mathieu Lafourcade. 2003. Construction collaborative dune base lexicale multilingue, le projet Papillon. *TAL*, 44(2):151–176.

Igor Mel'čuk, Nadia Arbatchewsky-Jumarie, Louise Dagenais, Léo Eltnisky, Lidija Iordanskaja, Marie-Noëlle Lefebvre, Adèle Lessard, Alain Polguère, and Suzanne Mantha. 1984, 1989, 1995, 1996. *Dictionnaire Explicatif et Combinatoire du français contemporain, recherches lexico-sémantiques, volumes I, II, III et IV*. Presses de l'Université de Montréal, Montréal(Quebec), Canada.

Igor Mel'čuk, Andre Clas, and Alain Polguère. 1995. *Introduction à la lexicologie explicative et combinatoire*. Universites francophones et champs linguistiques. AUPELF-UREF et Duculot, Louvain-la Neuve.

Kyonghee Paik and Francis Bond. 2003. Enhancing an English/Korean Dictionary. In *Papillon 2003 Workshop on Multilingual Lexical Databases*, Sapporo, Japan, 3-5 July.

Franck Richter, 1999. *Ding: a Dictionary Lookup Program*. http://www-user.tu-chemnitz.de/~fri/ding/.

Gilles Sérasset and Mathieu Mangeot-Lerebours. 2001. Papillon lexical database project: Monolingual dictionaries and interlingual links. In *NLPRS-2001*, pages 119–125, Tokyo, 27-30 November.

Gilles Sérasset. 1994. Interlingual lexical organisation for multilingual lexical databases in nadia. In Makoto Nagao, editor, *COLING-94*, volume 1, pages 278–282, August.

Aree Teeraparseree. 2003. Jeminie: A flexible system for the automatic creation of interlingual databases. In *Papillon 2003 Workshop on Multilingual Lexical Databases*, Sapporo, Japan, 3-5 July.