Corpus Sharing Strategy for Descriptive Linguistics
Kazushi Ohya*

Abstract
This paper introduces the idea of data sharing strategy based on a conversion service, not on a sharing application, scheme, or ontology, that are dominant in proposals for language documentation. Although these three methods have been basic tactics for sharing corpora, they have a conceptual flaw in terms of descriptive linguistics. In this paper we report the results of a previous project - the LingDy project, and propose a basic concept for corpus sharing strategy to support personal diachronic data sharing. This paper is a revised version of a handout at JADH2012, so readers should be careful that this content is based on results at the time of 2012.

1 Introduction
This paper is a report of the three-year LingDy project (2008–2010) on the documentation of six endangered languages, and a progressive report of a subsequent four-year project (2011–2014). As a result of the last project, we confirmed that (1) a method of transforming language data created by individual linguists into data in a shared format such as is used in a global-scale archive is an important and fundamental research target of language documentation; (2) it is difficult to realize this mechanism by the three ways hitherto adopted: sharing of application, scheme, and ontology; and (3) the key of language documentation in terms of data management is demarcation change and sound data handling, which pose key problems in computer science and linguistics respectively. These are the bases of the current objectives for our ongoing (in 2012) four-year project. In the following sections, first, we report the previous project, and show drawbacks of a broadly used scheme proposed by many standards; second, we confirm a philosophy of descriptive linguistics and an idea of personal diachronic data sharing; and finally, we propose an idea for sharing data based on data conversion services.

2 Language Documentation in the LingDy Project
Since 2008, we have been experimenting using a computational environment for endangered language study, which in recent years has come to be called language

* Tsurumi University
documentation (Gippert 2006). In our understanding, language documentation is a process of recording language information using computers. This project, with support from the LingDy project1 at Tokyo University of Foreign Studies, had aimed (1) to improve the environments of individual documentation activities by new tools and learning the usage, and (2) to seek a framework of archive systems for multiple endangered languages which will be used for typology research and for sharing the data with the language communities. The targeted languages were Yukaghir, Alyutor, Itelmen, Hezhen, Xibe, and Tiddim Chin. Our project’s approach to language study is descriptive linguistics, unlike in natural language processing studies, which uses a prescriptive approach. This difference is a vital point for language documentation to overcome.

Through this project, in addition to basic field linguistic activities, we (1) made a shared list of data items to be stored in our database, which will be used for both individual language studies and inter-language studies, such as typology; (2) provided a chance to learn ToolBox, ELAN (Brugman et al. 2002 and Wittenburg et al. 2004), and Perl; (3) made a Java application for batch cutting-sound processing; (4) made a Java application to convert data from a ToolBox format into XML, which has functions for data validation and normalization, and (5) made an experimental database system with Berkeley DB XML as a database engine and XQuery as a query language (Ohya 2009).

We presupposed plain text data with tags produced by ToolBox as an input data for data transformation (fig. 1).

1 http://lingdy.aacore.jp/en/
2 Compiled by Iku Nagasaki.
3 Compiled by Yukari Nagayama.
4 Compiled by Chikako Ono.
5 Compiled by Linjing Li.
6 Compiled by Norikazu Kogura.
7 Compiled by Kosei Otsuka.
9 This software can be downloaded from the following site and used in free: https://sites.google.com/site/lingdytextarchive/software
Figure 1. Example of ToolBox Data

We chose ToolBox as a platform to create input data because (1) learning a new kind of tool puts a strain on linguists, (2) there are many who already use it and others who have an interest in learning ToolBox, and (3) ToolBox was preferable to GATE (Cunningham 2000), ELAN, oXygen or other XML editors, FLEx, and other input tools used by linguists at that time. As input data we used two ToolBox files: a corpus file and a metadata file of speakers in the corpus.

A list of shared items for this database had been defined without difficulties. However, translations such as sentences, morphemes, glosses, and annotations tend to increase in number. For example, English descriptions are needed for a linguistic study, Japanese for our understanding, Russian for field linguistics, and the local language for the communities of the recorded language. In this project, we restricted the number of languages that can be used in each data unit to six for the purposes of implementing the system.

Automated concordance generation in ToolBox seems to be an attractive function for linguists.

oXygen, http://www.oxygenxml.com/
FLEx, http://fieldworks.sil.org/flex/
Now we are changing to using FLEx instead of ToolBox because (1) this is the intention of the production institution of ToolBox and FLEx, (2) FLEx provides useful functions for lexicography which reduces our encoding task, and (3) FLEx provides functions to support XML data in part.

The items necessary for each sentence/utterance are ID, IPA transcriptions, English translations of morphemes or words (depending on languages), and English glosses. In addition to these, linguists can record a translation of sentences, translations for morphemes or words, annotations, a base form of morphemes or words, POSs, and keywords.
As converted or output data we used two kinds of XML data: a simple scheme with a similar data structure to ToolBox (fig. 2), and a scheme in a so-called stand-off style (Thompson and McElvie 1997) (fig. 3).

Instances of XML data in a simple scheme and in a stand-off–style scheme are in figures 4 and 5 respectively.

A stand-off–style data scheme consists of a teiHeader element as a metadata part, and a corpus part with primary data, location and annotation elements. In the annotation part, elements of Feature Structure (fs) proposed by TEI (TEI Consortium 2007) and ISO\textsuperscript{15} are adopted to record the main components of language data. This XML data in a stand-off–style scheme was used as a main data set for the XML database system.

The TEI header provides bibliographic information units sufficient for our linguistic data. However, we did not record a list of terminology such as POS (part of speech) in TEI header, because we could not see any benefit to doing so, and it needlessly contaminates XML data with link information.

Figure 4. A sample simple scheme

The database system we made with Berkeley XML DB provides a simple search form and interlinear formed data as the results (figure 6).

We adopted the feature structure (fs) because this scheme is suggested by TEI and ISO. However, as a result of this project, we have concluded that the fs is a useful mechanism but a scheme in a standoff style is not. XML data in a standoff style scheme (1) tends to become verbose, or big data, (2) makes it difficult to handle link-traverse processing, and (3) is difficult to transform into other XML formats compared with XML data in a simple tree-based scheme.

Through this experimental application, we confirmed that (1) a flexible data conversion service or function is expected at multiple stages in language documentation by linguists, and (2) it will help to eliminate difficulties not only in sharing data but also in handling demarcation change of language information and sound data with symbol data. More detail about this observation is described in the following sections. To realize this service or function, we started a new project in 2011.16

16 The project name is A Study of Digital Archive Environment and Language Documentation for Minority Languages in NorthEast Eurasia, which is supported by Grants-in-Aid for Scientific Research in Japan, grant number 23401025.
3 Terminology

Before observing problems discovered in our previous project, we will explain basic terms used in this paper.

In markup language research, a structure or a structural pattern of meta-descriptions has been called “text structure,” “document structure,” or “scheme” (ACH, ACL, and ALLC 1994). Recently, especially since the advent of XML Schema, “schema” has been used to indicate a definition pattern of data structure in XML documents or instances. In some cases, “scheme” and “schema” seem to be used interchangeably, but in this paper we distinguish between them. A scheme is an existing structure or a structural pattern of meta-description, and a schema is a pattern to define a structure in an XML instance. Therefore, there is a case where a scheme cannot be defined by a schema. For example, a DTD cannot define a pattern of structures such as consisting of two elements followed by some three elements other than the last two elements. We can say that denoted text domain D by scheme, D(scheme), and schema, D(schema) are in the relation D(scheme) ⊇ D(schema).

Provided some schema such as DTD, XML Schema, or Schematron, a part of actual XML data can be called an XML “instance,” the total of which is equivalent to a valid XML document. Instances of XML data postulate the existence of structure definitions such as a schema. On the other hand, there has not been an appropriate term to indicate parts of XML data without structural definition patterns, the total of which is equivalent to a well-formed XML document. In this paper we use “instantiations” in this sense, and we introduce the term “skeleton” for meta-description or tags without contents inside meta-description units with tags.

17 W3C (2001) "XML Schema Part 0-2"
18 And now, a schema has another meaning, i.e. a list of semantic units, which is coined in a community of the semantic web or ontology. In this paper we do not use schema in this sense.
19 Grammar or syntax of a language and an expression pattern or a kind of morphosyntactic rule are different rules. DTD can play a role of grammar to define the whole instance, but it does not necessarily define the whole structural patterns in the instance. This may also be a point that is made by difference of descriptive and prescriptive viewpoints.
20 This term comes from “skeletal outlines” in J.H.Coombs et.al. 1987.
Multi-Link Path Model

4.1 Overview

In the field of language documentation, an XML scheme in a stand-off-style has been recommended for archives and repositories (Witt and Metzing 2010) because this scheme can be reduced from any scheme of marked-up data, which means that it is possible to reproduce data not only in the original scheme but also in the other scheme. And the scheme in a stand-off-style can also be used to encode multi-layered or partial data with overlaps or gaps (Bird and Liberman 1999), which has been regarded as problematic in encoding humanities texts.

In a stand-off–style description, targeted or primary texts and additional or annotational texts are encoded separately in instances or instantiations, which means that a textual reference-referent relationship is indicated by links; that is to say, descriptions in a standoff style lead to the implication of a model with a link structure. The relationship between primary and annotation data is represented denotationally by link attributes or element content, and connotationally by link paths. On the link-structural model, the number of link paths between primary and annotation data is greater than one, and is dependent on the model we employ. For example, in the case of TEI (TEI Consortium 2007), the number of possible paths can be more than one (fig. 7).

```xml
<w xml:id="w001">Scorn</w>
<w xml:id="w002">not</w>
<w xml:id="w003">the</w>
<w xml:id="w004">sonnet</w>

<seg>
  <include xmlns="http://example.org/ns/nonTEI" href="." xpointer="range(element(w001),element(w004))"/>
</seg>
```

Figure 7. One link path in TEI

The Annotation Graph (AG) introduced by S. Bird and M. Liberman (1999), which has been a fundamental concept for defining corpus schemes, implies two paths in the model. CES/XCES (Ide and Suderman 2006) and TEI/FS also adopt a two-path model (fig. 8). Actual implementations such as GATE (Cunningham 2000) and ELAN (Brugman et al. 2002 and Wittenburg et al. 2004), which are used to assist in annotating,

---

also suppose two link paths in each model, which connect three description parts: primary data, defining unit, and annotation data.

In LAF/GrAF (Ide and Suderman 2007), the estimated number of link paths can be at least three, among four parts: primary data, sink elements with segmentation functions, edge elements with linkage functions, and annotations (fig. 9).

In terms of processing, this kind of link path corresponds to traversal steps: i.e., link paths in a model reflect processing order. The model expresses processing hierarchies for establishing a data format to process. Therefore, a scheme in a standoff style has difficulties with link-handling processes (Alink 2002). For example, if we implement processes to handle the model, we need to know in advance a link path or IDREF-ID direction, the number of the paths, and a descriptive structure of defining units (Ohya 2008). This becomes an obstacle to sharing corpora in repositories (Ohya 2009).

An element that carries link information as reference and, at the same time, play a role of referent for a link from the other link elements is called “a location address element” in HyTime. This element presupposes a multi-link path model, but has not been supported in actual implementations to date. Notations for a graph such as RDF and LAF/GrAF to represent a scheme based on a link-structural model also do not solve this difficulty. Relations between HyTime and related standards concerning corpora are illustrated in figure 10.

---

4.2 Drawbacks

In addition to the link-handling problem, a stand-off–style scheme has two other problems: a primary data constraint and difficulty in reusability of primary and annotation parts.

Suppose we use a scheme in a stand-off-style with two link paths in the model such as AG, GATE, and ELAN. If the target data has a multi-layered or concurrent data model, we have to set an object as primary data. For example, in the case of corpora, timelines are good candidates. Timelines are not actual data, but they can be regarded as working objects to realize this scheme, because they satisfy the constraint of primary data: primary data must be comprised of codes that are smaller than any parts of the defining units and annotation data. If not, annotation data cannot refer to any unit because there is no appropriate primary data to be referred from a unit. This constraint is very similar to that found in a relational database, that is, terminal data in a cell is a primitive unit that cannot be divided into sub-parts. Consequently, this constraint detracts from one of the flexibilities of XML data.23

If annotation data is encoded in a stand-off–style scheme, primary and annotation data are not commutative in that they are not enable to play the other role. In order to navigate any part of data in stand-off–style descriptions—traverse to upper, sibling, or overlapping parts—we need to use information from data that defines units (Alink et al. 2006, Durusau and O’Donnell 2002). This navigation supposes the nature of languages, or linearity, which provides an index of a location. Such navigation fails if primary data has a higher-order structure in XML data, because a structure represented by markup languages does not substantially correspond to the order that linearity of text provides24.

---

23 This constraint would become effective in encoding concurrent information without timelines.

24 The data structure made up of marked-up parts is a semantic object, which means that a set of concatenated strings, letters, or characters is not the same as an order of data units.
That is, annotation data with a structure might not be possible to be primary data for other annotation data.

## 5 Ways of Data Sharing

Generally speaking, in order to share corpora encoded in a markup language, three methods have been adopted: sharing application, scheme, and semantics.

### 5.1 Application Sharing

Linguists engaged in descriptive linguistics have to pay attention to software implemented by SIL. ToolBox was such an application that linguists had to use or know. FLEX is a new application taking over from ToolBox, while ToolBox has been declared obsolete by SIL. FLEX is ambitious software that provides functions of real-time concordance building, ensuring consistency of items in a dictionary at all levels, providing multiple editing modes, and many other functions. FLEX has now been adopted by the RELISH project in which SIL, MPI, the University of Frankfurt, and Eastern Michigan University cooperate. We estimate that this project will become the mainstream of language documentation in the field linguistics.\(^{25}\)

However, even if FLEX becomes a common application among linguists, it is hard to image that it will satisfy all needs for descriptive linguistics. In fact, the present FLEX does not provide a function for handling sound data. ELAN, made by MPI, is used by many linguists as a tool to annotate sound with texts, which can be exported in an XML format or an EAF file. ELAN does not provide a way to analyze sound data in terms of frequencies like formant analysis. For sound analysis, there are many applications such as Praat.\(^{26}\) To rely on a single application for data sharing is unrealistic.\(^{27}\)

---

\(^{25}\) This is the then-current perspective at JADH2012. The present status and presence of the RELISH project is not the same as at that time. However, we believe that the significance of the project has been highly evaluated.

\(^{26}\) Praat, http://www.fon.hum.uva.nl/praat/.

\(^{27}\) As we see in the conclusion, the importance of handling sound information has not changed to the present.
5.2 Scheme Sharing
As we observed in section 4.2, the present standards proposed by ISO and TEI have difficulty to be pivotal data formats. As a solution to the problems, communities of HyTime and DSSSL had sought a shared abstract data model called Grove based on shared properties called Property Set. This model was in a graph structure, which is a target of node, or link, traversals. However, there was no definition for link handling or processing. If we stick to this line, we need a directive format to make a model resulting from applying some processes (Ohya 2009). This kind of directive plan, or model plan, makes a semantic model for data traversals, and each node in this model is a target mapped from a set of keywords such as ontology.

5.3 Semantic Sharing
Since the advent of the semantic web, many proposals such as RDF, ontology, schema, and linked data have appeared and been applied to many systems. However, an ontology-based semantic model does not fit in a system for descriptive linguistics. The reasons are as follows. (1) The present approaches in ontology adopt naive nominalism, which means that all concepts are indicated by nouns. However, this approach had already been rejected in linguistics because there is no surjection from a set of concepts to a set of semantic referents of nouns. For example, a sentential concept, which is called an event, cannot be referred to by a nominal only without an ad hoc name for the event. (2) There is no definition corresponding to the model plan shown in the previous section, which means that there is no way to indicate composite concepts. (3) The present set of words, or ontology, strongly depends on a specific language—English—which means that it is not appropriate for language studies.

6 New Strategy in Language Documentation
From the previous project we learned that (1) many systems proposed for language documentation is presupposed to be used in natural language processing research; (2) the systems are difficult to use in language documentation for descriptive linguistics; (3) when using these systems in descriptive linguistics, multiple kinds of data conversion

---

28 This is in a sense of shared semantics newly coined at schema.org.
29 This way is possible to apply to prescriptive linguistics.
30 There is no one-to-one relation between a language such as English and any another language at a level of word. Using loanwords function well in introducing new cultures or concept, but does not in describing concept that exists before introducing the loanwords.
are required, because in a field study data manipulation is a core activity of the study ranging from the level of a field note to that of computational data; (4) a service or function of multiple data conversion is expected not only in the case of individual research activities, but also in the case of archive systems, because the more such services are provided in the repositories, the more data will be offered by linguists who are not willing to convert their own data to fit into the specific format required from the repositories; and (5) within the data manipulation, a sound unit is an old but new object \(^{31}\) and we need a robust mechanism for handling sound data as a primary unit in language documentation or a language study.

6.1 Non-Prescriptive Linguistics

Many projects are proposing systems to support encoding and archiving language resources. However, most of them are difficult to use for linguists engaged in descriptive linguistics because the systems are made based on prescriptive linguistics concepts. \(^{32}\) Prescriptive linguistics taken in research of natural language processing (NLP) requires something single to promote social synchronic data sharing. For example, global-scale repositories require us to prepare corpora in a definite format. \(^{33}\) On the other hand, linguists want to use language resources in their own ways. To discover data structures or formats is one of the tasks of their studies. For such researchers, a system to support trial and error in defining data formats is desired. For example, in descriptive linguistics, a work to demarcate language units such as morphemes is a research activity itself. A part of speech (POS) is the typical example, a set of which changes depending on what kind of syntax we adopt. Therefore, it is difficult to prepare a list of concepts, units, or ontologies before the XML data is stored.

\(^{31}\) In a history of linguistics, sound in a language is the first and primary target linguists observe, then after the advent of comparative linguistics sings play an important role as a primary target of linguistic study. And now, in the digital age, we can handle sound as a primary data the same as sings on computers. This is a background that a research of language documentation started.

\(^{32}\) Descriptive linguistics is a language study to record and analyze evidences of language actually used, which provides resources for general linguistics, and, on the other hand, prescriptive linguistics is a language study to make a dominant rule, or a doctrine used in language education, dictionaries, language policy, etc.

\(^{33}\) Such a definite or dominant rule is required to handle language data in NLP in order to control the data. This stance is the same adopted in prescriptive linguistics.
In order to use the present systems in language documentation, at least, we need to prepare a data conversion service or function for supporting demarcation change and multiple categories.

6.2 Personal Diachronic Data Sharing
As we confirmed, sharing language data is important not only in language societies but also for individual linguists. As data analysis proceeds, the data units become more and more fine-grained, which means they have been incrementally changing throughout the research. This phenomenon can be seen both in research communities and in personal research activities. Ideally, linguists would enter their data into their system as permanent data that could be modified when necessary. Linguists want to use their data consistently in their research lives. Inter-annual data variation is an important subject in a study of language documentation, as is social data variation. In this sense, data conversion plays a more important role than scheme because data variation means scheme variation and this change is continual. A system without a data conversion service or function would not be useful to linguists.

6.3 Sound Data Handling
Language documentation makes linguists change their ways or processes of recording language information at every stage. It includes, for example, recording speech, segmenting, encoding, correcting, making concordances or dictionaries, changing data categories, printing them in an interlinear format, retrieving, converting, and preserving. In language documentation, the whole process is expected to be configured to be used on computers or in computational environments. Ideally, linguists would enter sound data into their system as a primary data unit with signs such as IPA. Any signs, including IPA, are essentially annotations of sound. However, for a long time linguists have treated phonetic signs as primary data because of the limitations of recording media. Of course, there is no guarantee that rawer information plays a more important role than abstract information such as phonetic signs. However, it can be said that

---

34 It can be said that making a system can be an objective of activities in prescriptive linguistics, but not in descriptive linguistics. This basic statement is important for people in Digital Humanities.

35 Social data variation means multiple data formats are devised in projects or organizations. Inter-annual data variation means diachronic variation in people and organizations, which can be regarded as a problem of data preservation.
handling sound data as a primitive data unit is a new challenge in language
documentation and also in linguistics itself.

7 Targets of New Project
We started a new project in 2011 with the objectives of (1) making a regional-scale
repository, which provides data validation and conversion services, to connect a global-
scale repository and individual linguists, and (2) supporting individual language
documentation activities in the whole process of building language data (Ohya 2011).
To achieve the objectives, we set two concrete targets: (1) flexible data conversion
based on API-sharing, and (2) an environment to handle sound data with annotation data.

We set requirements for a data conversion function as follows: (1) A converter uses a
declarative format to indicate data mapping. (2) This direction can be used as a
reversible format if condition permits. It means that if we run a converter twice with the
same direction format, we get the original input data back. (3) The direction can indicate
data units on a skeleton level of XML or similar markup languages. The challenge is to
devise a way to correlate units on skeletons in a simple form. Engineers and computer
scientists may consider this approach—sharing language data by data conversions—to
be tedious. However, in language documentation, engineers and computer scientists are
needed to support the humanities, and to recognize that application, scheme, and
semantic sharing do not work in this field. Data conversion is our solution.

Sound data is created by using many applications such as ELAN, Praat, Audacity,36
and Sound Forge.37 However, there is no reliable and consistent way to correlate sound
data and annotation or a series of phonetic signs. The key point to resolve this issue is
the handling of time point information. In the LingDy project, we created an application
that handles partial sound data by using time information generated by Sony Sound
Forge, Audacity, Praat, or ELAN. Based on this experimentation, we are seeking a
system to correlate annotation and sound data with time-point information, which will
be stored in an independent data unit that may be used to wrap variations of time data
format and to output a partial instantiation that works with ELAN data to play sound
with annotation data. For now, we are making an HTML5 environment to play both
ELAN and FLEx data with annotation data without additional information; providing
these applications, linguists can encode annotation on ELAN, store it on FLEx, revise it,

and play sound of the final annotation data with time information in the original data of ELAN.\textsuperscript{38}

8 Conclusion
A global-scale repository can be facilitated by techniques of semantic handling such as linked data and ontology. However, as we have observed, it is doubtful that the techniques can be applied in the regional-scale repository that we have introduced, because those who engage in descriptive linguistics expect personal diachronic data sharing. Our present project is a step along the way to establishing a regional-scale repository to connect a global-scale repository with individual linguists. Data conversion will be a key service or function for the regional-scale repository system. These days, some linguists who have been involved in language documentation seem to be relinquishing their plans or contributions for creating an ideal environment, and changing their focus from a system for creating data to a system for publishing data. This seems to be a retreat from essential problems in language documentation. We know that our goals are very difficult to reach. In fact some of the problems have not been solved for a long time. We will continue to devote our energy to the problems even if it takes more time than the estimated project length.\textsuperscript{39}

\textsuperscript{38} For now, there is no shared architecture or application for time information of sound, and each application provides original way for it. As the result, each process of data management with different application is disconnected each other. Using FLEx and ELAN in each advantageous field is difficult or almost impossible without additional data revision.

\textsuperscript{39} This paper is a revised handout distributed at JADH2012, and the actual status of the present project is, unfortunately, slightly different from the statement in this conclusion. Admittedly, in a sense, our estimation was optimistic. Due to human resources or our abilities, we got stuck in implementing a data converter because before coding this software we have to define a hierarchy of data description including implicit processing instructions such as entities and link handling. And we need a steady mechanism to correlate sound data or real data with encoded data or abstract data in order to bridge corpora and sound data in a system without ad-hoc ID handling. We will explain the details of these problems and report the result of the present project in the near future. As a result of this second project we made an HTML5 environment to play sound with data made by ELAN and applications to convert FLEx data into data suitable to be played in this environment. Based on the results, we started the third three-year project from 2014 to 2016, A Study of Documentation on Theories and Practices on Minority Languages in Siberia, supported by Grants-in-Aid for Scientific Research in Japan, grant number 26370512.
References


This project is a fundamental study for a data format to connect time information to encoded language resources by using languages in Siberia as test resources. A partial report of this project is introduced in the paper titled “A General Format for Time Information to the first-Class Data of General Linguistics” at the site of ICLDC4 2015, University of Hawaii.


