The ecology of documentary and descriptive linguistics[1]

Jeff Good

University at Buffalo

1. Introduction

This paper will propose a model of the "ecology" of documentary and descriptive linguistic research. I use the term *ecology*, here, as a designation for the set of individuals, resources, tools, and actions that are involved in creating, archiving, and using documentary and descriptive resources.

The primary goal of developing this model will be to facilitate the characterization of tools and standards for digital linguistic resources with respect to the entire documentary and descriptive process in order to (i) help researchers avoid duplicating the work of others unnecessarily and (ii) ensure that linguistics, as a discipline, does not accidentally focus on particularly salient domains (e.g., interlinear glossed text curation) at the expense of others (e.g., transmission of resources to an archive) which are equally important to the overall health of the ecology. A secondary goal of developing this model will be to lay out in one place a number of the concepts that are crucial to understanding the state-of-the-art in digital linguistic resources and tools but which are sometimes difficult to obtain detailed information on.

The guiding philosophical principal of this enterprise is that ensuring that the resources linguists create are long-lasting and interoperable requires not only strategies for dealing with specific problems (e.g., how to input lexical data) but also recognition that solving individual problems without seeing how those solutions connect to the "big picture" may not be very different, in the long run, from not having solved the problem at all. A tool that produces a best-practice lexicon placed within a larger ecology where there is no way for that lexicon to be archived may produce an outstanding resource—but one which the future will never know about.

Furthermore, given that the resources for creating linguistic tools and resources are quite limited, it is imperative that we plan projects in ways which facilitate cooperation and avoid duplication of effort. If desiderata for a given linguistic tool or standard are not laid out in terms of how they relate to other tools and standards, there is the immediate danger that two projects will fail to realize they have implemented custom solutions to a more general problem. Such duplication of effort may not directly "damage" the ecology. However, to the extent that it diverts resources from areas also in need of tools and standards development, it can damage it indirectly.

[1] This paper was originally written as a position paper for the 2006 E-MELD Workshop on Digital Language Documentation: Tools and Standards: The State of the Art. I am grateful to Helen Dry, Heidi Johnson, and Nick Thieberger for comments on that version of the paper.

Of course, none of this is to say that it is “wrong” for people to pursue their own objectives without considering how their efforts fit into the larger ecology. Following Bird and Simons (2003), I take the foundation for decisions relating to the role of technology in linguistic research to lie outside the technological realm and, instead, to be grounded in the values of the communities with a stake in the products of documentary and descriptive research. To the extent that the narrow community of linguists and the larger community of consumers of linguistics resources value resource portability, in Bird and Simons’ sense of the term, then I believe they should value a healthy documentary and descriptive ecology. If someone else has strikingly different values in this domain, the points made here will probably be largely irrelevant to them.

The structure of this paper is as follows. Section 1 discusses the primary objects in the ecology breaking them down into three categories: individuals, resources, and tools. Section 2 discusses the role of different types of communities of individuals in the ecology. Section 3 discusses different types of actions performed by individuals and tools. Section 4 gives a sample model of a fragment of the ecology in order to illustrate some of the ways in which such models can facilitate work on the digital tools and standards for language documentation. Finally, section 5 offers a brief conclusion.

The intended audience for this paper is the technically-inclined ordinary working linguist, and the model of the ecology to be developed here reflects my own understanding of their conception of the ecology. Clearly other audiences may conceptualize it differently.

2. The “species”: Individuals, resources, tools

2.1. Introduction

We can conceive of the digital linguistic ecology as containing three primary types of “species”: individuals (including both linguists and non-linguists), resources (e.g., dictionaries, grammars, and texts), and tools. They have in common that they constitute relatively stable features of the ecology. I discuss each of these in turn.

2.2. Individuals

Individuals are the primary agents in the ecology, they create the resources in the ecology and use the tools designed to facilitate resource creation and exploitation. In general, at a given point in time, an individual can be put into one of the three heuristic classes below.

- **Archivist:** An individual in charge of ensuring a resource is properly archived.
- **Collector:** An individual who collects data and creates resources containing that data.

- **User:** An individual who makes use of resources created by collectors.[2]

At different times, a given individual may taken on different roles. A collector of data from one language is likely to be a user of data from another language. Someone who is a user of data one day (say, of texts in a particular language) may become a collector the next day when they "repackage" that data into a new resource (say, a text concordance). The purpose of putting individuals into these classes is not to assign them hard and fast roles. Rather, it is to help understand that the interactions among individuals, at a given point in time, will largely be determined by which of the three above categories the individuals can be understood as belonging to at that time.

Figure 1 describes the prototypical interactions among individuals by modeling the user as making requests to an archivist or to a collector for particular resources and by modeling the collector as transmitting resources to an archivist. Noteworthy here are asymmetries in the relationships holding between different types of individuals. The user receives a direct benefit in their prototypical interactions with archivists and collectors while the archivist and the collector do not receive any comparable benefits from the user. Similarly, the collector receives nothing directly in return for sending resources to the archivist.

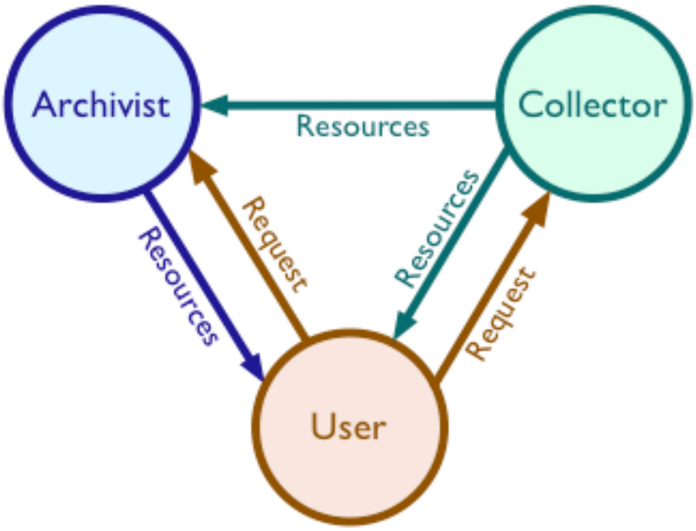

Figure 1: Prototypical individual interactions

There is one more individual worth mentioning here who plays an important role in the digital linguistic ecology: the tool developer. For the "ordinary working linguist", the tool developer would not seem to play a direct role in the ecology because such individuals are generally conceptualized more like gods (sometimes benevolent, sometimes malevolent) than partners in the resource-creation process. This may change in the future, of course. But, at

[2] The three-way distinction among individuals given here does not distinguish between users/collectors who are speakers of a language being documented and users/collectors who are not. As pointed out by Nick Thieberger, it may actually be important to include speakers as a distinct set of individuals in the ecology from the other types due to their ciritical role in a project's success or failure.

present, the role of the tool developer in the model being developed here is peripheral (in the sense of "to the side" not in the sense of "unimportant").

2.3. Resources

The second important types of object in the digital linguistic ecology are the resources containing collected data. These resources can be of many different forms, but, here, I will assume that all linguistic resources can be broadly categorized as primary texts, lexicons, language descriptions, or some combination of the three.[3] Primary texts are taken to include recordings, and associated transcriptions, of speech events of any kind—narratives, conversations, elicitations, letters, etc. The category of language descriptions should be construed broadly to include, among other things, both descriptive grammars and theoretical work.

Resources are the most stable part of the documentary and descriptive linguistic ecology. Indeed, it is hoped that standards already proposed and under development will allow researchers to produce resources which will last for eternity. They are also one of the primary vehicles through which individuals interact with each other since an important method of collaboration is through resource exchange and sharing. For example, a linguist may work on the analysis of a text collaboratively with a native speaker either by sending a partially-analyzed text to the speaker for verification or by working on a copy which is available to both individuals for examination and editing at once.

In recent years, it has become fashionable to talk about resource *interoperability*. So, it would be useful, here, to briefly discuss what this means, in the context of the larger ecology. Informally, we can understand resource interoperability to refer to the ability of the information contained in different resources to be usefully combined and put to new uses. However, the vagueness of this definition leaves out a number of complications, and it is important to distinguish between different types of resource interoperability, some of which are given below.

- **Content interoperability:** The ability of the substantive content of two resources to be easily compared and joined together. A prototypical case of high content interoperability is the ability to compare two analyses of similar data from two different languages within the same theoretical framework (e.g., two HPSG analyses of passivization).
- **Terminological interoperability:** The ability of the terminological content of two resources to be easily compared and joined together. A prototypical case of high terminological interoperability would result when the terminology of two resources was associated to a common ontology (e.g., the GOLD ontology).

[3] This assumption is based on the fact that these are three highest-level linguistic data types proposed by the OLAC Working Group on Linguistic Data Types. See: http://www.language-archives.org/REC/type-20060406.html.

- **Structural interoperability:** The ability for the content of two resources to be easily compared because they share a common structure. A prototypical case of structural interoperability would result from two documents of a similar type (e.g., lexicons) adopting the same model for their data (e.g., the same model of the structure of a lexical entry).
- **Markup interoperability:** The ability for the content of two resources to be easily compared because they share a common markup system for their data. A prototypical case of markup interoperability is the ability for two XML documents to be easily read and manipulated by the same tools.
- **Format interoperability:** The ability for the content of two resources to be easily compared by the use of the same tool because they share a basic format readable by the tool. A prototypical case of high format interoperability is the ability for two text documents to be opened by any tool that can read text files.
- **Encoding interoperability:** The ability for the characters encoded in two resources to be easily compared and joined together. The prototypical case of high encoding interoperability results when two documents both use Unicode encoding.

In the ideal world, interoperability would be achieved at all levels which were not linguistically “interesting”—that is, levels which do not constitute active areas of linguistic research. This would seem to include most of the levels listed above except for content interoperability and, in some cases, structural interoperability. While, clearly, some degree of content interoperability is desirable, total content interoperability would amount to a complete lack of theoretical debate. All too often, however, it is difficult to locate substantive cases of disagreement in the content of two resources because of lack of interoperability at other levels, representing a linguistically-uninteresting barrier to research.

In the context of the larger linguistic ecology, it is important to always be explicit about what kinds of interoperability are desired between two resources and what kinds of interoperability are facilitated by the use of a given tool. It is never enough, for example, to say that two resources are interoperable because they share a common “format”. Is it a common structural format or a common machine format? If the former, how much structure do they share? Being unclear on the exact nature of the interoperability achieved by the use of a shared “format” may be little better, in the long run, than not considering interoperability issues at all.

2.4. Tools

2.4.1. Custom and non-custom tools

At present, tools are the most problematic and unsystematized aspect of the ecology. We can broadly understood tools as objects which facilitate the creation and exchange of resources. While such a broad definition would include such traditional items as pen and paper, here I am primarily interested in tools coming out of the digital realm.

Nevertheless, it is important that we do not limit the discussion to tools designed specifically for linguistic purposes. When looking at the present-day ecology of language documentation and description, it is clear that Microsoft Word, for example, has a critical role as the tool of choice for most linguists when creating resources containing long stretches of prose (e.g., descriptive grammars or theoretical papers). And, while this program may be deprecated for certain linguistic uses for which there are more suitable tools (e.g., the creation of lexicons), it is unlikely it will completely disappear from the ecology of language documentation and description in the near future.

Similarly, while the ubiquity and general utility of basic communication tools like e-mail programs or messaging clients renders them almost invisible, we should recognize that they are among the most successfully-employed tools in the ecology. E-mail programs are particularly noteworthy for facilitating two distinct processes: individual-to-individual communication and individual-to-individual resource exchange (in the form of attachments).

It seems useful, therefore, to distinguish between two kinds of tools in the ecology, *custom* and *non-custom*. The former will refer to tools specifically designed to facilitate a documentary or a descriptive task. The latter will refer to tools designed for other tasks (perhaps quite general ones), which, in one way or another, have been employed for descriptive and documentary functionality. I give some important examples of each type of tool below.

- **Custom:** AGTK, Elan, Transcriber, FIELD, IMDI Metadata Editor, OLAC Repository Editor, Shoebox/Toolbox, Praat
- **Non-custom:** E-mail software, web browsers, Microsoft Word, Microsoft Excel, FileMaker Pro, iTunes

An important near-term goal of documentary and descriptive linguistics should be to determine where non-custom, widely-used tools are suitable for the accomplishment of certain tasks relating to language resources and where only custom-built tools will suffice. It is abundantly clear that linguists do not need to reinvent e-mail's facilitation of basic communication between people, for example. However, it is much less clear if using e-mail as a means of resource exchange is appropriate—among other concerns, e-mail attachments do not offer a ready means to associate relevant metadata to the resources being transferred.

Distinguishing between custom and non-custom tools—and understanding when a custom tool is needed—can be quite complex. This is because the notion of "tool" is, itself complex. Almost certainly, the most complex tool, from the perspective of modeling the ecology, is the web browser.[4] While the web browser originally had one primary function, rendering HTML web pages and allowing a user to navigate between different web pages, it can now replicate—from the user's perspective—many of the functions of more specialized tools. Among other things, it can serve as an e-mail client (e.g., Hotmail or GMail), a lexical

[4] A less complex case, but still noteworthy in the present context, is FileMaker which, in the hands of a skilled developer, can perform a number of tricks for which it was never intended, for example, automated interlinear glossing.

database browsing and entry system (e.g., the FIELD tool[5]), or a collaborative document editor (e.g., Wikipedia). In reality, the browser itself only serves as an interface to such tools—much of the "real" work done by other tools, either locally on the user's own machine or on remote servers. From the perspective of the average user, however, it is as though the browser is the Swiss-Army knife of digital tools.

If a browser is running a linguistic tool like E-MELD's FIELD lexicon editor, is it a custom or a non-custom tool? The best way to deal with this question is probably not to conceptualize the browser as a tool at all but, rather, as a *platform* on which other tools are built. The distinction between a platform and a tool is not absolute but relative. From the perspective of the user of FIELD, the browser is a platform.[6] From the perspective of (perhaps the same) user visiting a web site, the browser is a tool.

Making the platform/tool distinction allows us to take a more nuanced view of the custom/non-custom tool distinction than would otherwise be possible. At a foundational level, there are essentially no custom digital linguistics tools—nearly all of the hardware we use was not designed specifically for linguists.[7] Our hardware serves as the platform on which operating systems are built, and these operating systems, in turn, serve as the platform on which all other programs run. Other platforms can be built on top of those operating systems, etc. At some point in this picture, the tools we use for our day-to-day linguistics work emerge.

We should, thus, ask ourselves, what the relationship of our tools should be to different platforms, in addition to what kinds of tools we need. At one extreme, we could propose that all tools should be self-contained units on top of an operating system. At another extreme, we could propose that all tools should be built on some convenient non-custom higher-level platform, like a browser. Or, we could propose that linguists require some linguistic-specific platform on which all future tools should be built.

Of course, the ideal solution is likely to be some mix of all of these. What is important here is that we should not only ask ourselves whether or not we need a custom "tool", we should also ask ourselves what kind of platform we want our tool to be developed on. Non-custom platforms, especially web browsers, have the advantage that users are often already familiar with many aspects of their user interface. However, it may make sense to give up this familiarity if there were a custom platform better suited for rich data exchange among tools specifically designed for working with linguistic resources. For example, a specialized linguistic platform could have built-in functionality for "smart" cut-and-paste of sentences from texts and lexical entries from an electronic dictionary into prose documents—i.e., cut and paste utilities that automatically formatted these data types in a way appropriate for the

[5] http://emeld.org/tools/fieldinput.cfm

[6] I should underscore here that I am dealing with conceptualizations localized to specific users. From the tool designer's perspective, a browser will typically only be the platform for one part of a tool: the user interface. For the typical user, however, the user interface *is* the tool, making the distinction irrelevant.

[7] The only cases of digital hardware designed specifically for linguistic use I am aware of are certain measurement tools used in phonetic research.

document they are copied into, while, at the same time, retaining full provenance information for the data "hidden" within the document.[8]

A final point about the tool/platform distinction worth marking is that one of the most popular digital "tools" in the history of linguistic fieldwork, Shoebox, might be better conceptualized as a tool*kit* consisting of a set of tools (for example, a lexicon tool and a text tool) all built on a common platform. One reason for the popularity of Shoebox, clearly, is the common platform on which these different tools were built leads to a similar user experience across tools and a very high degree of interoperability across different resources created with those tools. When designing the next generation of tools, it seems worthwhile to keep in mind these ingredients of Shoebox's success.

2.4.2. Issues in tool design

In general, the most pressing questions on linguists' minds with respect to tools tend to take forms like: When will we have a tool that finally does X right? or Is it OK to use this tool for this problem? While such task-oriented questions are perfectly reasonable in the context of work on a particular documentary and descriptive project, they are problematic in the context of ensuring the "health" of the larger ecology. Purely task-oriented tool use and conceptualization runs into two important problems: (i) a given task is accomplished without accompanying plans for longevity or usability of created resources and (ii) features of the tool being designed/used unnecessarily duplicate functionality found elsewhere.

With respect to the second problem, it is worth mentioning that there are two possible outcomes, one more problematic than the other. The less problematic outcome is that time is simply wasted. The more problematic one is that, while the "duplicated" solution may have more or less the same functional coverage as a previously existing solution, the two may differ in ways which, while trivial from the perspective of linguistic research, could be quite important from the point of view of interoperability.

For example, at present, there exist several annotation formats which can be usefully employed to create time-aligned transcriptions of audio recordings. Some noteworthy ones in the context of linguistic work are the Praat annotation format[9], the Transcriber annotation format[10] the Elan annotation format[11], and the TASX annotation format[12]. There is a good deal of overlap in the content these formats are meant to encode. However, the formats are

[8] It is useful, when thinking about the future of tool design, to envision the ideal set of tool interactions, rather than limit ourselves to what would be easy with today's technology, since it helps ensure that we design our tools in the present to be flexible enough to accommodate functionalities we will want at some point in the future.

[9] http://www.praat.org

[10] http://trans.sourceforge.net/

[11] http://www.mpi.nl/tools/elan.html

[12] http://medien.informatik.fh-fulda.de/tasxforce

sufficiently distinct to inhibit interoperability of content encoded in them without the development tools to convert between them. Praat uses a different markup system from the other three formats, which are all encoded using XML, presenting a low-level barrier to interoperability (markup interoperability). The Elan annotation format and the Transcriber annotation format both use an XML encoding. However, they each make use of a different conceptual model for the structure of an annotation, creating a high-level barrier to interoperability (structural interoperability).

In some cases, of course, there may be very good reasons for two tools to implement different solutions to similar problems—this is not a problem in and of itself. In fact, it might be sign of healthy competition among competing ideas. However, this situation should be avoided when not truly necessary, and an important step to assure this does not happen is to see how a particular tool's functionality is located within the overall documentary and descriptive ecology and not to focus only on the narrow task at hand.

2.4.3. Tool interoperability

Having discussed some important aspects of how tools fit into the ecology of language documentation and description, it is worth revisiting the issue of interoperability to see what role tools may have in facilitating interoperation among resources. As with resource interoperability, we can distinguish a number of different ways in which tools can interoperate with each other, some of which are given below.

- **Tool interoperability:** Two tools may be designed to interoperate with each other (in the limiting case, this could be two instances of the same tool on two different machines), allowing for a high-degree of interoperability for resources created by these tools. A prototypical case would be interoperation between an e-mail program and an address book program where each can send information encoded in its resources directly to the other.
- **Format interoperability:** Two tools may both use a common resource format, allowing each to read files produced by the other. A prototypical case are the multitudes of tools which can read plain text files.
- **Exchange interoperability:** Two tools may use distinct working formats but allow their resources to be exported into a common exchange format, thereby facilitating interoperability. A prototypical case is the ability of spreadsheet and database tools to export to tab-delimited text formats, which can be read and imported by other spreadsheet and database tools.

There appears to be consensus that the linguistics community values exchange interoperability, not least because such interoperability can facilitate archiving—assuming that a given tool can produce an exchange format which is the same as an accepted archival format. Interoperability of the other two types could clearly be beneficial to the ordinary working linguist, requiring them to use and master fewer tools and formats. However, this would be at the expense of requiring tool developers to cooperate at relatively deep levels of tool design, which will not always be practical.

3. Communities

While individuals are conceptualized as basic objects in the conceptual model being developed here, it is important to recognize that individuals will group together into communities of various sizes and kinds (and, of course a particular individual may be a member of many communities).

The three most important communities for present purposes are the community of archivists, the community of collectors, and the community of users. There are a number of reasons why we need to recognize the presence of these in any model of the ecology of language documentation and description, some of the most important of which are given below.

- **Self-interest:** The self-interests of members of a given community will tend to be overlapping and may be opposed to the self-interest of members of other communities. For example, it is in the interests of the community of archivists for there to be good tools for metadata creation and for data collectors to use those tools. Data collectors, however, may see little need for these tools in their day-to-day work and, in fact, might consider time spent using them to be wasted.
- **Communication:** The types of communication typical within a community will be different from the types of communication between members of different communities. For example, two data collectors on the same project may need to debate how to annotate a particular piece of data. However, a data user is likely to be requesting data from a collector or asking how to interpret an annotation.
- **Exchange:** The ways in which resources are exchanged among members of the same community is likely to be distinct from how resources are exchanged between members of different communities. For example, data collectors will often need to exchange "drafts" of resources with each other, while data users will typically only receive "public" versions of resources. Archivists, may need to exchange resources with each other for backup purposes but, unlike other users, not be particularly interested in the content of the resources.
- **Content:** Different communities will have different desires for what content is included in resources. A user might need much more detailed grammatical information about the data in a resource than the collector who creates the resource.

The three communities listed above can, of course, be subdivided into relevant subcommunities. For example, the community of users could probably usefully divided into an academic subcommunity, subcommunities of speakers, and the general public. In examining the role an individual, resource, or tool has in the documentary and descriptive linguistic ecology, it is important to always have a clear sense of what communities an individual is a part of and what communities a given resource or tool is intended to serve.

Furthermore, it is absolutely crucial to recognize that different communities may have conflicting interests and that these conflicts are an integral part of the ecology itself.

Ideally, the tools in the ecology will someday be sufficiently advanced that they will mitigate the inherent tensions among different communities to the point of irrelevance. However, some of these tensions, particularly in regard to resource content, may never be amenable to a purely technical solution. Nevertheless, it is clear that an interesting, and complicated, area of future research will involve the development of standards and tools to facilitate community "interoperation".

4. Getting things done: Tasks and transactions

4.1. Tasks

So far, the focus has been on "objects" within the ecology. It is also necessary to consider "actions". It seems useful to distinguish between two kinds of actions *tasks* and *transactions*. The prototypical task will involve an individual editing a resource or obtaining data from a resource. The prototypical transaction will involve a resource being transmitted from one individual to another. Some possible tasks would be:

- Adding an entry to a lexicon (a prototypical task for a collector).
- Editing the metadata for a resource (a prototypical task for a collector or an archivist).
- Querying a lexicon for information about a particular lexical item (a prototypical task for a collector or a user).

In the digital realm, all tasks require the mediation of tools—whether these are custom or non-custom tools. Finding the entry for a lexical item for example could be achieved by opening up a text file (if the lexicon is stored as text) and doing a simple text search with a text editor's "find" command. Or, it could be done using a tool custom designed for task of finding lexical items within a lexicon file.

While most linguists' prototypical sense of a task will involve a human user acting on a resource, we should not confine our general conceptualization of tasks to human users alone. It could be the case that some tasks will be performed by tools with the mediation of other tools. To pick one possible example, a tool designed to assist in the interlinearization of texts may need to interact with a tool designed to access information from a lexicon. One could, of course, integrate the capability to access a lexicon directly into a interlinearizing tool itself. But, given the fact that other different tasks will require similar access to lexical data (for example, the task of a users looking up a word), it could be desirable to create one general purpose lexical access tool that can be used both by users *and* by other tools.

User- and tool-initiated tasks are schematized in figure 2 where a collector and a interlinearizer are both shown as interacting with a lexicon with the mediation of a tool specifically designed to facilitate editing of and access to the data in a lexicon.

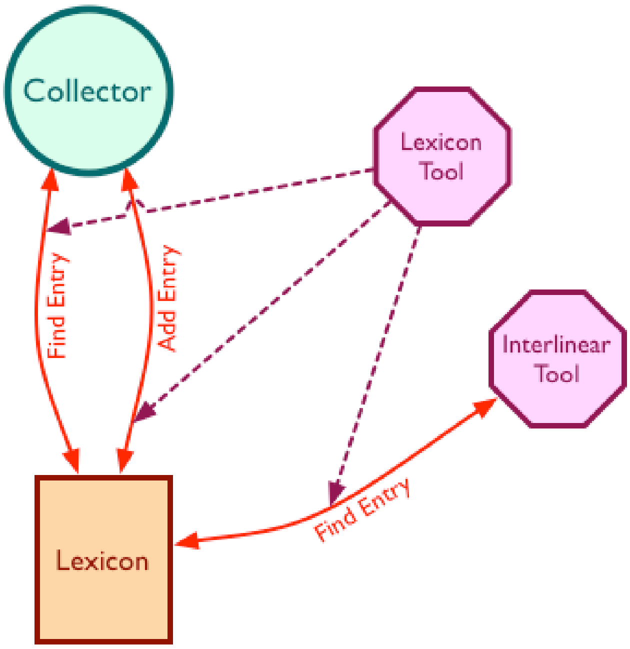

Figure 2: User tasks and tool tasks

4.2. Transactions

In addition to tasks, the other major kind of actions in the ecology are transactions—actions that involve interaction between multiple users. Two kinds of transactions are:

- **Resource exchange:** The transmission of a resource between two users, either of the same subcommunity or across different subcommunities.
- **Message exchange:** The transmission of a message between two users, either of the same subcommunity or across different subcommunities.

An important reason for distinguishing tasks from transactions is the fact that transactions will often take place across communities. Transaction tools, therefore, may need to be designed to be compatible with the interests of multiple communities in mind. Another practical reason to distinguish transactions from tasks is specific to resource exchange. This should always involve exchange both of a resource and of its metadata, and any tool designed for resource exchange will need to facilitate this.

A prototypical set of transactions is schematized in figure 3.

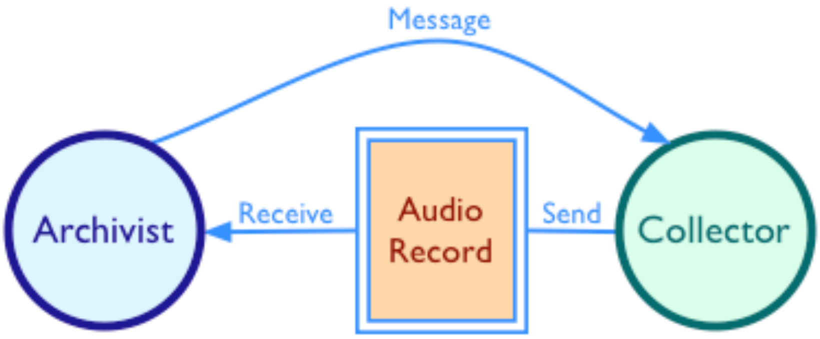

Figure 3: Transactions between a collector and an archivist

Figure 3 schematizes two transactions: the exchange of a resource from a collector to an archivist and the exchange of a message from the archivist to the collector.

4.3. Modeling tasks and transactions

When modeling tasks and transactions, it is important that one keep in mind the granularity "problem". A given task, for example, may itself be comprised of various subtasks—which themselves are comprised of another set of subtasks. This is already alluded to in figure 2 where the broad task of "working with a lexicon" is broken down into two subtasks of "adding an entry" and "finding an entry". (This, of course, leaves out the important subtask of "editing an entry".) A subtask like "finding an entry" could in turn be broken down into tasks like "entering search criteria", "applying search criteria to the lexicon", and "transmitting the results to the user". (This back-and-forth aspect to such a tasks is, in fact, why they are represented with two-headed arrows.) Determining the appropriate granularity at which to model a task or transaction is far from obvious and is dependent on which part of a project is being worked on at a given time. In general, the more fine-grained the model, the more technical expertise is required in developing it. Since, in the end, tool designs are built (explicitly or implicitly) on a number of interdependent abstract (coarse granularity) and concrete (fine granularity) models of a given task or transaction, it is important to think about issues relating to granularity at all stages of tool development.

There is, however, unfortunately, no easy solution to the granularity problem. That is, it is rarely obvious what level of granularity is required to deal with a given problem, or even how to determine what subtasks should be considered to be at the same level of granularity. The best one advice one can probably give on this issue at present is, unfortunately, negative rather than positive. Two specific *don't*'s come immediately to mind: (i) don't get caught up in details of implementation before larger conceptual problems are worked out and (ii) don't assume a technician can ever really understand the linguist's needs early on since the technician's skills involve manipulating technology at a very low-level of granularity and most linguists live at a high-level of granularity.

A common example, these days, of where the first *don't* would apply is when a "website" is invoked, at early stages, as a critical element of a project which requires a general system of resource dissemination. Without a doubt, websites will often turn out to be the best available option for resource dissemination. But, there is no need to tie a project to any particular dissemination technology until it is completely clear what needs to be disseminated. For small-scale documentation projects, for example, whose resources will all be sent to a publicly accessible OLAC archive, it is not obvious if a special project web site will serve the needs of users any better than simply directing users to the archive housing resources produced by the project.[13]

Issues surrounding the second *don't* are why, so often, the tools resulting from collaborations between linguists and programmers are rather dissatisfactory. The route between vague statements like, "I need a lexicon tool", and a working piece of technology is a perilous one, filled with hundreds of small decisions, from nitpicky details of the user-interface to fundamental choices about resource formats. Tool design requires a good deal of basic research and needs to be pursued with the same mindset as analyzing a particular complex piece of linguistic data. Early analyses of technical problems will often be wrong—just as initial linguistic analyses are often wrong. One's understanding of the problem will evolve as technologies evolve—just as advances in linguistic theory cause our understanding of linguistic data to evolve. And, above all, like any *research* project, it will involve a lot of research, ranging from an examination of prior relevant work (in the form of existing related tools and standards) to discovering what kinds of collaborations are required to fill in important gaps of expertise. Bridging the granularity gap between the linguist and the technician is a lot of work—but the alternative is a tool which simply won't be adequate for the task it was designed for.

5. The ecology

While understanding the entirety of the ecology of language documentation and description is an undertaking well beyond the scope of this paper, it would be useful here to schematize a fragment of the ecology so that we can obtain a general idea of its shape and understand the utility of seeing how the different objects of the ecology fit together. Such a schematization is given in figure 4. Even though this only gives a small fragment of a possible ecology, it is still quite complex.

[13] To make use of an analogy from theoretical linguistics, jumping to decisions of implementation before larger issues have been worked through is like characterizing data theoretically without first getting a clear sense of the descriptive facts. In the end, it might turn out that *ONSET is the constraint you need, but it's a good idea to make an inventory of attested syllable structures before saying you've found evidence for it.

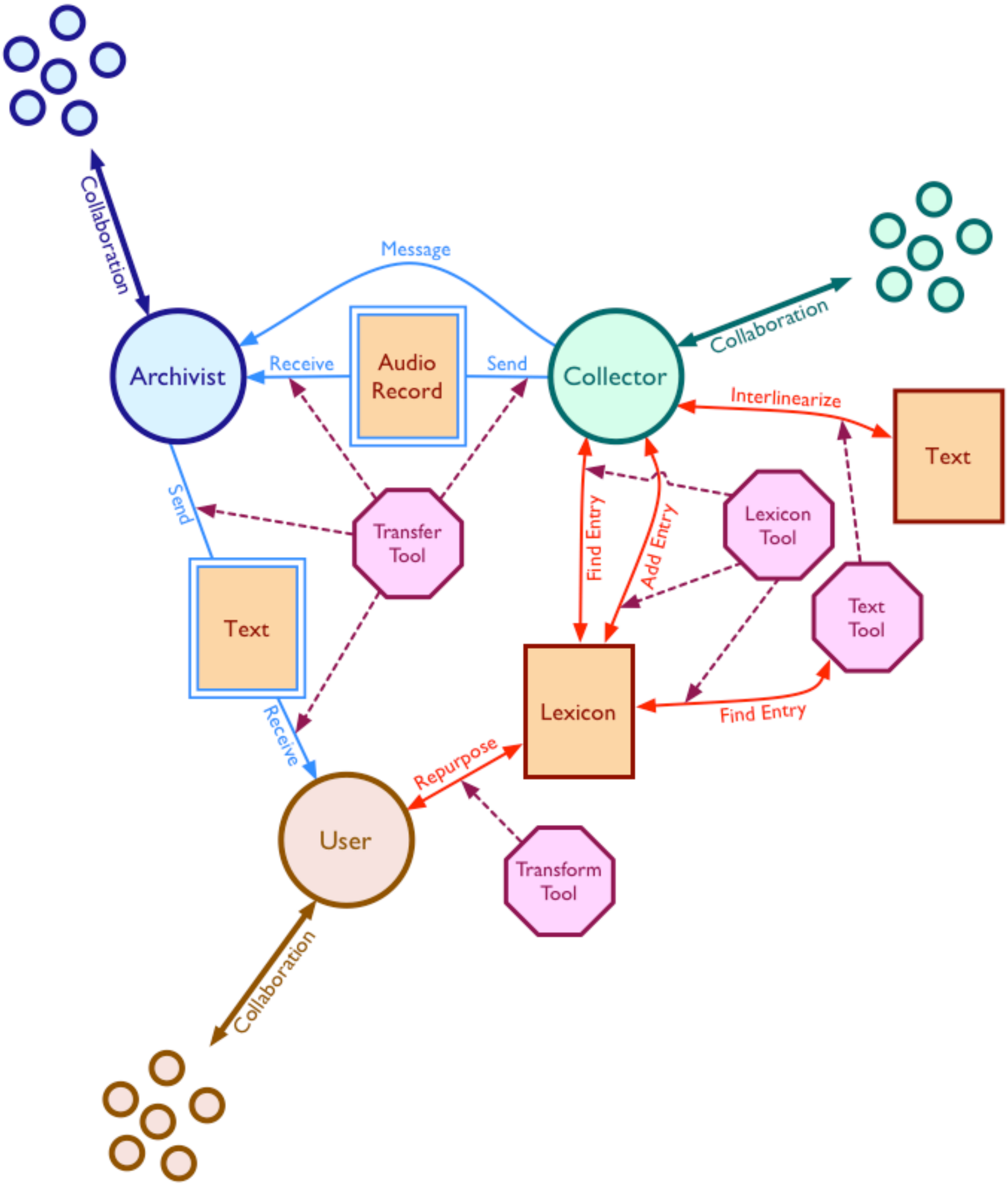

Figure 4: The ecology of language documentation and description

Figure 4 contains three individuals, an archivist, a collector, and a user, each of which is schematized as collaborating within their own community. It also contains three kinds of resources: texts, audio recordings, and a lexicon (the choice of these was largely arbitrary). Finally, it contains four abstracted tools: a lexicon tool, a text tool, a transformation tool (for changing the format of a resource into one customized for a given purpose, e.g., web-based display), and a tool facilitating resource exchange. (No tool facilitating simple message transfer is schematized, but, of course, various such tools exist in the ecology.) Tasks are indicated with red arrows and transactions with light blue arrows. A number of user-initiated tasks and transactions are given, and one tool-initiated task is given (the “find entry” task initiated by the text tool).

Modeling the ecology of language documentation and description has at least three important functions: (i) it allows us to map out the “state-of-the-art” in digital tools for linguistic work, (ii) it allows us to see where similar functionality may be required in distinct parts of the ecology, and (iii) it allows us to see interdependencies between individuals, resources, and tools which might otherwise be ignored. I briefly illustrate how the figure in 4 helps us deal with these three areas.

We can map out the state-of-the-art in digital tools by seeing how the abstract tools in figure 4 correspond to existing tools. This discussion does not exhaust all tools used for these functions and is meant to be exemplary not definitive.

- **Lexicon tool:** Two tools commonly used to create lexicons are FileMaker and Shoebox. Both have good functionality for adding entries to a lexicon and finding entries in a lexicon. Neither has functionality for interacting with an arbitrary text tool. However, Shoebox combines lexical functionality and text-analysis functionality, making it a better fit in the ecology schematized in 4.
- **Text tool:** Two tools commonly used to create annotated texts (ignoring time-aligned texts at the moment) are Shoebox and Word. Of these, only Shoebox offers any kind of integration with a lexicon tool, making it a better fit in the ecology schematized in 4.
- **Transfer tool:** The primary tool used for resource transfer at present is e-mail. E-mail has no built-in support for the packaging of metadata with the resource. It is, therefore, a problematic tool for this function. However, the ease of its us should not be ignored, and it could be used for sending metadata, as well as resources, given behavioral modifications on the parts of individuals.
- **Transform tool:** No general purpose transform tool for converting resources from one format to another is in use. Instead, users rely on the export functionality of a variety of different programs. This seems a reasonable solution for audio and video resources. However, it is more problematic for text-based resources where the formatting needs of linguists are quite particular. XSLT is a good solution for the transformation of XML resources for individuals with the requisite technical expertise.[14]

Taking up the second use of figures like the one in 4mentioned above, it shows us various ways in which similar functionality is required in distinct parts of the ecology. The most striking of these is in resource exchange. As schematized in the diagram, for many purposes, a single resource exchange tool (e.g., e-mail) will work for exchange between a range of different users (e.g., collector and archivist or archivist and user). The figure also illustrates how a single tool (e.g., a lexicon tool) may have functionality which could be of use both to an individual and to another tool.

[14] For an introduction to XSLT see: http://emeld.org/school/classroom/stylesheet/xsl-help3.html.

Finally, the figure in 4 allows us to see interdependencies in the ecology at multiple levels. At the highest level, it indicates the different roles of archivists, collectors, and users in the ecology, and shows how a well-functioning ecology needs to facilitate interactions among these three groups. At lower levels, it shows how the functionality of one tool (here the text tool) may depend on functionality present in another tool (here the lexicon tool). It also shows how worthwhile resource exchange may require more than simply managing the exchange itself. Resources will generally need to be transformed to suit the needs of different individuals. The figure in 4 shows a user applying a transformation to a given resource (a lexicon). However, such transformation could take place elsewhere: for example, perhaps an archivist could transform a resource for a user before transmitting it. At present, it is not clear at whose responsibility such transformation should be—and the figure gives only one possible solution.

The discussion here is intended to primarily be an illustrative exercise—one could repeat it for many other aspects of documentation and description. What is important is not so much the specific content of the conclusions as its illustration of how one can conceptualize individuals, resources, and tools not solely in the context of a particular problem but also as part of a wider ecology.

There is a critical element to the ecology that is not schematized in figure 4 but which needs to be discussed, at least briefly, here: standards. It is not straightforward to schematize how standards fit into the ecology because their role is so pervasive. They facilitate individual-individual interaction, tool-tool interaction, and individual-tool interaction by ensuring that resources exchanged among users and tools can be interpreted by all participants in the exchange. Every arrow in the schematized ecology needs to be associated with some standard, whether it be an ad hoc standard (e.g., choosing English as the language through which messages between a collector and an archive will be exchanged), a very general digital standard (e.g., the standards governing the exchange of e-mail between computers), or a standard designed specifically for linguistic resources (e.g., the XML schema proposed in Bow, Hughes, and Bird (2003) for interlinear glossed text).[15]

6. Conclusion

A paper like this one can only give a rough sketch of anything as complex as the ecology of language documentation and description. Details, both of the present state of the ecology and its ideal future state, need to be filled in. And, it is quite likely that there are important fundamental features of the ecology that have simply been missed here. Perhaps, for example, the classification of actions as either being tools or transactions is too gross, and a central category has been missed. Or, perhaps, standards should have been given a more central role in the ecology's structure. Furthermore, this document was written with a technically-sophisticated "ordinary working linguist" in mind. Different conceptualizations of the

[15] Future work on the ecology of language documentation and description, perhaps, could attempt to ground standards more directly in the ecology's model.

ecology, for non-linguist technicians and speaker communities, for example, would also be valuable.

References

Bird, Steven, and Gary Simons. 2003. Seven dimensions of portability for language documentation and description. *Language* 79:557–82.

Bow, C., Hughes, B., and Bird, S. (2003). Towards a general model for interlinear text. In *Proceedings of E-MELD Workshop 2003: Digitizing and annotating texts and field recordings, East Lansing, Michigan, July 11-13*. (http://emeld.org/workshop/2003/bowbadenbird-paper.html)