CHAPTER ELEVEN

The Comparative Siouan Dictionary Project

David S. Rood and John E. Koontz

1. INTRODUCTION. The project we describe is the preparation of a dictionary of Comparative Siouan, that is, a reconstruction and cognate list of the vocabulary and, necessarily, much of the morphology of the language that was the ancestor of the modern Siouan languages. (See Rood 1979 for a survey of Siouan studies as they stood before the Comparative Siouan Dictionary project.) The Comparative Siouan Dictionary (CSD) is now in an advanced state of preparation, except for introductory essays and a phase of final cleanup editorial work, though it has now been about twenty years since we first conceived of the idea. This chapter describes the history of the project, including its conceptual history, the use of computers, the funding history, and the evolution of methodology as we moved from paper to computers and learned to work together as a team. Some of the specific linguistic results of the work have been published by Rankin, Carter, and Jones (1998).

The Siouan family consists of fifteen to eighteen documented languages in three major subgroups (see fig. 11.1); the exact number of languages depends on whether one classifies some of them as dialects or as separate languages. They are (1) the Missouri River Siouan group, Crow and Hidatsa; (2) the Central Siouan group, subdivided into Mandan and a large grouping called Mississippi Valley Siouan, comprising Dakotan[1]

1. Dakotan is a neologism intended to serve as a term for the subgroup as a whole (Dakotan) in contrast to the Santee-Sisseton and Yankton-Yanktonais version of the native name for the ethnicity (Dakhota or Dakota), which is often applied to the Santee-Sisseton dialect specifically. For a discussion of Dakotan dialectology, see Parks and DeMallie 1992. Santee-Sisseton is commonly known as Santee or Dakhota or Dakota. Yankton-Yanktonais is traditionally grouped erroneously with Assiniboine and Stoney as Nakota or Nakoda. Teton is also known as Lakhota or Lakota.

Proto-Siouan
- Missouri River Siouan
  - Crow
  - Hidatsa
- Central Siouan
  - Mandan
  - Mississippi Valley Siouan
    - Dakotan
      - Santee-Sisseton
      - Yankton-Yanktonais
      - Teton
      - Assiniboine
      - Stoney
    - Hočąk-Chiwere
      - Winnebago (Hočąk)
      - Chiwere (Iowa-Oto-Missouria)
    - Dhegiha
      - Omaha-Ponca
      - Kansa (Kaw)
      - Osage
      - Quapaw (Arkansas)
- Southeastern Siouan
  - Biloxi
  - Ofo
  - Tutelo

Figure 11.1. Siouan Language Family.

(which includes the dialects Santee-Sisseton, Yankton-Yanktonais, and Teton and the languages—or dialects, depending on one's point of view—Stoney and Assiniboine), Winnebago-Chiwere (Winnebago or Hočąk and Chiwere or Ioway-Otoe-Missouria), and Dhegiha (Omaha-Ponca, Osage, Kansa, and

Quapaw); and (3) the Southeastern Siouan group (also called Ohio Valley), Biloxi, Ofo, and Tutelo.[2]

2. OVERVIEW OF THE PROJECT. Dictionaries are notoriously time-consuming projects, typically the work of lifetimes, or at least professional lifetimes, and this in a discipline where the collaborative work needed to pile up man-years is rare and unusual. Comparative dictionaries are particularly demanding, because they require expertise in more than one language and introduce their own elaborate machinery for correlating and glossing the forms compared. In the CSD project we have been fortunate in having to deal with only ten or twenty languages, but unlucky in having to face multiple orthographies for most of them, including ad hoc ones, and older phonetically based systems.

To reduce the task to manageability, the CSD project at the Center for the Study of the Native Languages of the Plains and Southwest (CeSNaLPS, or the Plains Center) adopted three expedients:

- First, there would be three principal editors—Robert L. Rankin, Richard T. Carter, and A. Wesley Jones—plus a project manager, David S. Rood.
- Second, interested graduate assistants—primarily Jule Gomez de Garcia and John E. Koontz—would be employed.
- Third, given Rood's interest in computer applications in linguistics, it was determined that computers should be used.[3]

The rest of this chapter describes the way in which we worked, grew, and changed. We begin with some additional background.

3. PRECURSORS. Like any major project, this one had a number of precursors.

3.1. THE SIOUAN LANGUAGES ARCHIVE. The first precursor was the Siouan Languages Archiving Project. In the early 1970s Rood predicted

2. Hočąk is usually known as Winnebago. The Wisconsin Winnebagos prefer the native name Hočąk, which is also spelled Hocak. Hochank, Hochunk, and Hochangara are other variants. Chiwere, used as a synonym or cover term for Iowa-Oto-Missouria, is a spelling variant of Jiwere or Jiwele, the native name of the Oto. Oto is also spelled Otoe; Iowa is also spelled Ioway. Dhegiha is also spelled Ȼegiha in the turn-of-the-century orthography of the Bureau of American Ethnology. It is an Omaha-Ponca word meaning 'local, one of this group'. It is used by linguists to refer to the whole group of Dhegiha languages. Kansa is also spelled Kansas, Kaw, Konze, or Kanze. The Quapaw are also known as the Arkansas.

3. In addition, Koontz had a background in computer science and Jones was an enthusiastic user of the new personal computers. Koontz was inspired by a course on the use of computers in linguistics that had been offered a few years previously by visiting professor Robert Hsu of the University of Hawaii.

that computers would be an important research tool in linguistics in the future and that the only way to capitalize on this for Siouan languages would be to have our resources, that is, grammars, dictionaries, and text collections, in a form that the computers could manipulate.

At that time computers were giant machines—at the University of Colorado a series of Control Data Corporation systems—housed in air-conditioned buildings and accessed by mysterious people in white coats, who took packages of punched cards from you at a window and, if you were lucky, returned a printout and your cards a few hours or even days later. If you made a programming or data encoding error, you then had to repeat the process, of course, but the iterative refinements of procedures that now take a few minutes took weeks.

We (Rood and colleague Allan R. Taylor) did not know that if we had waited a few years it would have gotten so much easier, so we began to transfer to punched cards the information in the books we wanted to search and to carry box after heavy box of cards down four flights of stairs in Woodbury Hall for transport to the University Computer Center and back (about a six-mile round trip), over and over again.

The results were functional but ugly. The sixty-four-character proprietary Control Data Corporation character set contained only uppercase Roman letters, digits, and a few punctuation marks. This imposed the need to represent many characters with digraphs, trigraphs, and so on, including such simple things as uppercase characters. The standard English lowercase Roman characters were represented by the uppercase characters of the set, the uppercase characters by characters preceded by a plus, acute accents by an asterisk following the character, raised *n* by an *'N* following a character, and so on, as in (1). Each character and diacritic encountered in encoding the documents had its own representation. The system employed is defined by Rood (1981).

(1) +DAK'HO*TA = Dak'o'ta, representing Dakhóta
+UMA'N*HA'N = Uma^{n}'ha^{n}, representing Umą́hą

Still, by the end of the project we had all the material for the extinct languages and much of the material on the other languages in machine-searchable form (Rood 1981). Now it was time to put it to use.

3.2. THE WORKSHOP ON COMPARATIVE SIOUAN. A second preliminary step was taken in summer 1984. With support from the National Science Foundation (NSF) and the National Endowment for the Humanities (NEH) we—here primarily Rood—gathered together a group of people

who had been working on various Siouan languages[4] and asked them to think about ways we could advance Siouan studies cooperatively, without duplicating efforts. We concluded (among other things) that a comparative project was in order, and we learned that Carter and Rankin were already working independently on comparative Siouan databases (using handwritten slip files), each with the idea that he would someday compile a comparative dictionary. At the workshop they agreed to merge their files and transfer the information to "cognate set sheets," single-page compilations from various languages of forms that looked like they might constitute a cognate set. This work was undertaken at the workshop. The resulting sheets were passed from person to person, with the idea of filling in more possible forms from various additional sources not yet consulted by Rankin or Carter.

At the end of the workshop the participants divided themselves into teams to execute various follow-up projects. The first of those was to be the comparative dictionary, based on the workshop slips, and the editors designated were Carter, Rankin, and Jones. Rood was given the task of overseeing the project and finding funding for it, which kept him busy writing grant proposals for the next couple of years. None of the other projects we planned at that time—a revised bibliography, a collection of grammatical sketches, and a collection of papers—have yet been started, the dictionary having taken all our attention until now.[5]

4. THE FIRST FEW YEARS. During the next three years, the cognate set sheets were passed from editor to editor. First Carter took them home and sorted through them, dividing some sets into two or three or merging sets that seemed to be duplicates. Then Rankin, spending a sabbatical year in Boulder, did the same thing. Finally Jones, in Bismarck, got a turn at adding Hidatsa and Crow material and scrutinizing the others' work. During this period, Graczyk and Koontz added various slips, which were sent to the dictionary editors, or commented further on existing slips. In summer 1989, after one failed attempt, we received funding from the NEH for the project and began the long series of summer meetings and winter tasks that are summarized in table 11.1 below. The story hereafter involves parallel developments in computational technology, our

4. Carter (University of Manitoba), Jones (University of Mary), Rankin (University of Kansas), Rood (organizer) (University of Colorado), Patricia A. Shaw (University of British Columbia), and Paul Voorhis (Brandon University), with Allan R. Taylor (University of Colorado) and Josephine White Eagle (MIT) part of the time, and (then) graduate students Randolph Graczyk (University of Chicago), John E. Koontz (University of Colorado), and Willem de Reuse (University of Kansas). Ray Gordon (SIL) also visited briefly.

5. Paul Voorhis did complete a manuscript sketch grammar of Catawba for the projected volume of grammatical sketches of Siouan languages.

TABLE 11.1. Chronology of the Dictionary

Date	*Funding*	*Methodology*
1984–85	Workshop NEH, NSF[1]	Slip files to cognate sheets.
1986–89	none	Copying cognate sheets (paper shuffling).
July 1989	NEH[2]	First use of computer database program.
1990–91	NEH, APS, CU	Continue as above.
July 1991	Extension; new grant denied	Everyone on a computer; daily file merging; everyone doing different languages.
Summer 1992	New NEH grant[3]	Printouts, everyone on same page, data entry later.
Summer 1993	NEH-2 continued & supplemented	Working together on one computer; "finished" about 25% of database.
Summer 1994	Supplement from NEH	All on one computer; about 84% of database acceptable to all.
Summer 1995	Squeezed supplement	Finished last 16% and went through all of it again.
Fall 1995ff	none	Database exchanged for proofing and cleanup; formatting and indexing programs
1999ff	none	Copy editing using formatted printed copies of the database.
200?	none	Final printed product?

NOTES: [1]NEH: RD-20477-84; NSF: BNS 8406236.
[2]NEH: RT-21062-84.
[3]NEH: RT-21238-91.

understanding of Proto-Siouan, and our evolution of various techniques for cooperative editing.

4.1. THE FIRST COMPUTERIZATION. We originally thought we could do the entire job in three years, although the NEH would agree only to support us for two. Skeptical about meeting their expectations, we nevertheless plunged gleefully into the work. It is hard to remember now what computer technology and tools were like in those days, but as the computationally supported part of the project began in 1989, there were only a couple of database programs available to us that seemed appropriate, and neither of them was quite what we needed (see §6.3 below).

In spite of this handicap, we finally chose the program that seemed to be the lesser of two evils (askSam) and hired Jule Gomez de Garcia (at the University of Colorado) to start entering the cognate set sheets, using

a brand-new IBM PC/AT with a huge 20-megabyte hard drive, a major advance over the computer center visits and the roomful of cabinets of punch cards of a few years earlier.[6]

The first two years, the editorial team assembled annually in Boulder for several weeks each summer. During those first summer meetings, we continued to use the cognate set sheets as our working tools, since interactive computing was still not feasible. The editors sat around a big table and discussed the contents of the sheets one by one. They appointed a scribe to write down their results, so that Gomez de Garcia could then transfer the new decisions to the database.

4.2. THE COMPUTER ENVIRONMENT.

4.2.1. DOS AND THE CHARACTER SETS. The computers employed in the project from the very first PC/AT on have all been independent (non-networked) DOS systems. The ones in service currently run in Microsoft Windows 95, but the project continues to operate primarily in DOS windows.

The main impact of using DOS has been that DOS supports as its only native character set the 256-character IBM extended ASCII set, while the project employs well over 256 characters, many of them not included in this set. This has never been a real limitation because throughout the time of our work the technology to stretch this DOS limitation has been available. Still, we have spent considerable time setting up this technology and maintaining it, and we have always been limited within this technology to seeing a single character set at a time on the screen, though we could print as many as we needed on the same page.

Under DOS, the character set problem must be resolved for each combination of display device and application. The two display devices in question have been the EGA/VGA[7] series of graphics cards (and suitable attached monitors) and the Hewlett-Packard LaserJet series PCL[8] (non PostScript LaserJet) printers and compatibles, mainly the Plains Center's[9] now quite venerable HP LaserJet II.

The details of solving the character problem under DOS need not detain us here, because they are obsolete under modern versions of

6. Simultaneously, Rood acquired an IBM PC/AT with a 30-megabyte hard drive for use as an editorial tool with the *International Journal of American Linguistics*. The Department of Linguistics at the University of Colorado already had a pair of similar machines for its own use.

7. Enhanced Graphics Adapter/Video Graphics Array.

8. Printer Control Language.

9. The Plains Center is the name by which the University of Colorado's Center for the Study of the Native Languages of the Plains and Southwest (CeSNaLPS) is commonly known.

Microsoft Windows, which support TrueType fonts for display and printing. Suffice it to say that we use the Duke Language Toolkit to create the EGA/VGA screen fonts, and the Summer Institute of Linguistics (SIL) program Keyswap to redefine keyboards. We use the SIL Premier Fonts and now the SIL Legacy Fonts packages to create printer fonts, and we use the SoftCraft Font Solution Pack (primarily the Laser Fonts package) to generate Microsoft Word printer drivers for the HP LaserJet II and supplement the features of the SIL font tools. The SIL packages can make Microsoft Word printer drivers, but the Laser Fonts package is (or was) much more adept at this.

Given the evolution of Microsoft Windows in its various versions and the development of TrueType fonts for use with it, a more satisfactory solution today would be to use the SIL Encore Fonts package to create Windows and printer fonts and the SIL-promoted TavulteSoft Keyboard Manager tool to define keyboards. In the Microsoft Windows environment it is possible to see more than one font at a time on the screen as well as on paper, in any application that supports the use of Windows fonts and printing.

4.2.2. INFORMATION AND DATABASE COORDINATION. For the bulk of the project, some members did not have access to e-mail, and none of the project computers at the University of Colorado were networked in any significant sense before late 1997. The lack of e-mail access for some of the members made collaboration during the academic year difficult, but the teaching loads and other research of the editors prevented most academic year CSD activity during nonsabbatical years anyway.

A more serious problem has been the lack of file-sharing facilities. All exchanges of files took place by diskette, though it sometimes took some ingenuity to fit the database onto a diskette. Various file compression and archiving tools have been used in this capacity. All these problems have been rendered obsolete now by the general availability of Internet-based e-mail, ftp file transfer, the Web, and larger removable media.

5. FURTHER EVOLUTION. As the work progressed, we found that we needed more and more time, and, of course, we kept running out of money, too. The first grant was supplemented by the American Philosophical Society and by the University of Colorado and given a time extension, but in 1991 we were forced to apply for a second grant. Our first application was denied, because we were not doing "salvage" linguistics, which seemed to demand all the available funding for Native American projects at the time. But the second time we successfully argued that we had all done our part in that arena and were now ready to make use of some of what we had helped to salvage. The second grant, supplemented in 1993 and extended twice, saw us through to where we are now.

We had to make concessions to get the supplement, such as accepting reduced indirect cost reimbursements and doing without honoraria for the summer workshops for the last two years.

During this period, there was a distinct evolution in the way we processed the growing and maturing database. At first the cognate set sheets contained essentially guesses, and to a considerable extent the early effort involved splitting, lumping, and rearranging sets. Then came a period when we concentrated on expanding the contents of the sheets, at which point each editor undertook to work with subgroups of languages, for example, Jones with Crow and Hidatsa, Carter with Mandan, Dakotan, and some of the Southeastern languages, and Rankin with the others. John Koontz also contributed to this effort for the Winnebago and Chiwere groups.

During our summer meetings in this period, each of us had his own computer and the three editors worked independently on the database, though the files were merged at the end of each day so that everyone had a newly updated version the next day. Those sessions involved long periods of silent clicking and page flipping, interspersed with questions and discussion about what was being discovered.

Off and on during this period we were able to add new chunks of data, too; these included new forms from Chiwere elicited by Louanna Furbee and her colleagues, from Osage by Carolyn Quintero, and additions from continuing fieldwork on Crow by Randolph Graczyk and on Mandan by Carter. We also made use of archive searching of Winnebago and Chiwere material by Gomez de Garcia and of Omaha-Ponca material by Koontz.

Gradually, in fact imperceptibly at the time, we switched away from finding new forms to discovering new sound correspondences, and it was during this period, too, that the editors discovered big differences of opinion on how to represent certain features of the protolanguage. Eventually they agreed on some compromise representations, but not without intense (and often repeated) argument. There were times when we wondered whether the team would survive the controversy, but the editors' loyalty to the project ultimately overcame their loyalty to their own preferences. Naturally, the potential for this kind of disagreement becoming fatal is a danger for this kind of project; in our case it has proved surmountable.

As the database matured, the effort to bring forms from the individual languages into established and growing cognate sets changed to one of examining the sets from top to bottom for consistency in sound correspondence. At first when this point was reached, the editors found themselves most comfortable with multiple copies of printouts of the computer database. These they discussed at length and scribbled on, after which the scribbles were converted to database amendments by Jule Gomez de Garcia. Then, in summer 1993, they discovered that they could work directly on the computer by clustering around a single screen and discussing what

they saw, changing and fixing it then and there. We thus witnessed a very fine-grained evolution from pure paper to machine copies of paper to separate copies of machine entries to single, interactive entries. From here, that looks impressively logical and systematic, but in fact it was just a way of growing, and unplanned growing at that.

Summer 1995 saw us using up the very last pennies of our grant money, but at the same time we found we had a nearly finished dictionary. The painfully slow process of looking at each set carefully together, which had covered 25 percent of the database one year (1993), an additional 60 percent the next (1994), and the last 15 percent in the third year, accelerated so fast that in the third year the editors finished their examination of the database and also went through the whole collection a second time. Now they were happy with their results and Koontz was able to write computer programs for formatting and indexing that could be rerun on the database at will. This has freed us to edit from formatted material, a definite psychological plus, without fearing that we were creating new problems for the final product. Eight years of cooperation (counting the workshop), instead of the originally envisioned two or three, have finally come near to paying off. Unfortunately, the close examination of the database entailed in formatting it for printing has revealed the presence of numerous minor inconsistencies in form.

6. THE DATABASE.

6.1. CONSTRUCTION. As stated in section 3.2, initially the CSD database was based on the combined slip files of Carter and Rankin, with consideration directed to the published reconstructions of Matthews and others, including to some extent the early work of Wolf, though this work is difficult to collate with more recent work.[10] We supplemented this material with

- observations from specialists in particular languages, for example, Randolph Graczyk for Crow and Josephine White Eagle for Winnebago;
- extensive examination of some newly rediscovered correspondences like *R ("funny *r*") (cf. Dorsey 1885);
- attempts to find sets involving unusual segmental sequences found in some of the languages (like Dakotan *gw* or Ioway-Otoe *dw*); and
- searches for culturally and ethnotaxonomically relevant vocabulary.

10. We were also given access to the unpublished work of Terrence Kaufman but decided not to consult it because of the various philological and other problems of working with the unpublished notes of a living individual without his direct involvement.

These are all standard techniques and for the most part were carried out manually.

We did not attempt the technique of back-construction, in which, using known or postulated sound changes, all possible predecessors of forms in various languages are constructed and collated mechanically, looking for matches that will then be evaluated under human intervention. This would best be done at the root level in languages with the kind of morphological structure that Siouan languages have, and we lacked extensive root lists to which to apply this technique.

We did, however, have access to the computer files from the Siouan Language Archiving Project. These included a large number of texts in certain of the languages and several dictionaries. We were able to use these in several ways.

First, and most simply, we were able to do computerized searches of the materials for particular languages to fill in gaps in our data. For example, Koontz has searched the Dorsey (1890, 1891) texts extensively to fill in the gaps in the Omaha-Ponca data. We have also searched for unusual segmental sequences or interesting morphological structures.

Second, we needed to find a way to identify the verb roots that were hiding behind any of several instrumental prefixes, since the roots are often cognate but do not occur with the same prefixes from language to language. For example, take the PSi form **-xuxe* 'to break brittle things'. The cognates in the daughter languages occur with various instrumental prefixes, for example, Hidatsa *núxuxxe* 'break by hand', Lakhota *naxúye* 'break by stepping on', Winnebago *booxúxux* 'break something brittle by blowing'. So we culled from several large, representative dictionaries in the Siouan Archives all the instrumental derivative entries that we could identify automatically by shape. These were first folded at the root initial to facilitate sorting by root initials and then manually collated by Jones (see Jones 1991), leading to his discovery that a large portion of the underlying roots of instrumental stems fall into families of related forms such that stems of the structure $C_1C_2VC_3$ seem to display a very old complex structure in which C_1 and/or C_3 may have constituted separate morphemes added to the C_2V root. We refer to the nonroot morphemes as root extensions and in the database (not in the final dictionary) use the purported root and extensions to keep apart sets with similar glosses.

6.2. FORM. We maintain our dictionary as a form of database.[11] In effect, the database is a computerized slip file. It does not look like a

11. Much of Koontz's thinking on this can be traced to the work of Hsu, though some of the same principles are reiterated in the SIL Shoebox manual. Hsu's work has been particularly influential in the lexicography of Pacific and Native American languages.

printed dictionary in this format, but it is easy to find one's way around in and edit. We derive printed reports, including the final printed dictionary, from this form using various software tools. Part of the database entry for 'dog' is given below.[12]

GLOSS[dog
GRAMCAT[N
SEMCAT[Anml
ENTHIST[GdeG 11-29-89
CHGHIST[done 06-20-90
CHGHIST[. . .
\PSI[*wi-šų́ke
OTHREC[. . .
PCH[. . .
PMV[*šų́kE
PDA[*šų́kA
LA[šų́ka | 'dog' C
DA[†šų́ka | "suŋ´-ka" | 'dog' R-450a
ST[sų́ga | 'dog' PAS
PWC[*šų́ų́ke
. . .
CH[šų́ŋe | 'horse' Marsh
WI[šų́ų́k | 'dog, horse' KM-3005
WI[šų́ų́gnį́k | 'puppy' KM-3002
PDH[. . .
PSE[. . .
COM[This ancient term has been adapted in historical times . . .

Key to fieldnames (not all exemplified): GLOSS gloss; GRAMCAT grammatical category; SEMCAT semantic category; ENTHIST entry (keying) history; CHGHIST change history; PSC Proto-Siouan-Catawban; PSI Proto-Siouan reconstruction; OTHREC other reconstructions; PCH Proto-Crow-Hidatsa; CR Crow; HI Hidatsa; PMA Pre-Mandan; MA Mandan; PMV Proto-Mississippi Valley; PDA Proto-Dakotan; LA Lakota (Teton); Da Dakota (Santee); SV Sioux Valley; YA Yankton; YS Yanktonais; AS Assiniboine; ST Stoney; PWC Proto-Winnebago-Chiwere; CH Chiwere (Ioway-Otoe); IO Ioway; OT Otoe; MO Missouria; WI Winnebago; PDH Proto-Dhegiha; OP Omaha-Ponca; PO Ponca; OM Omaha; KS Kansa; OS Osage; QU Quapaw; PSE Proto-Southeastern; TU Tutelo; SP Sapony; PBO Proto-Biloxi-Ofo; Bl Biloxi; OF Ofo; PCA Proto-Catawban; CA Catawba; WO Woccon; OTHLGS Other language (families); COM Comment. Key to source abbreviations (partial): C Richard Carter; R Robert Rankin; PAS Patricia Shaw; (Gordon) Marsh; KM Kenneth Miner.

12. For this chapter, the figures and lists were recoded using our three current Microsoft Windows ANSI-based character sets, which we call Standard Siouan, James Dorsey, and Dakotanist. These are implemented as TrueType fonts. In the actual database we use a single modified DOS Enhanced ASCII character set we call Siouan Dictionary. Siouan

It might seem simpler to maintain the data in the form of a final document, that is, as a word processor or desktop publisher file, or in some form of application-independent markup, for example, SGML or HTML, but that approach commits one to at least a particular final structure and generally also to a particular formatting scheme, depending on the factorability of formatting schemes in the application chosen. It may be possible, but difficult, to convert from this structure and formatting scheme to others, whether for primary use or merely to obtain some auxiliary report. Moreover, conversion between formatting schemes often results in the loss of some or all of the formatting. Thus maintaining the database in final report form commits one in some degree to a particular structure and a particular formatting of it, which can be awkward in a ten- to twenty-year project. Apart from this, report formats are seldom optimal for computerized data retrieval.

The form of database we selected for our efforts is called informally a textbase (see askSam Systems 1991: 2). Textbases are an extension of the standard tabular conception of a database to more freely formatted textual data. Although they have been applied to such tasks as organizing legal briefs, contact notes, and recipes, they are particularly useful for dictionaries and other linguistic work.

As in a standard database, the basic unit of data is a record, which is subdivided into fields. In the illustration for 'dog' above, for example, the record represents the cognate set 'dog'. Most of the fields represent cognate forms. The fields represent reconstructions, comments, notes on the editing process, and so on. Thus the LA field is a cognate Lakota (Teton) form for 'dog', and the PDA field is the Proto-Dakotan reconstruction based on the various Dakotan forms, while COM is a general comment by the editors. The key field is GLOSS, the English gloss.

In spite of these similarities, there are some differences between databases and textbases. Databases require the existence of a unique key or indexing field. Textbases do not. Textbase keys are typically ordered, but they need not be unique. This permits, for example, two records representing descriptions of two homophones, or, in our case, multiple records for reconstructions with the same gloss. In some extreme cases of textbases, there are no keys and even the record structure may be missing. The extreme cases usually arise with treatment of running texts as databases.

Dictionary is implemented both as an EGA/VGA screen font and as a Hewlett-Packard LaserJet printer bitmap font.

The Siouan Dictionary character set lacks some characters that we need. In the database these are represented with digraphs, like *a.* for *ą*. In printing, both the Siouan Dictionary character set and the digraphs are mapped to several different modified ASCII bitmap fonts, including the Siouan Dictionary bitmap font mentioned above. Some additional fonts are required to achieve the typographic requirements of the dictionary apparatus.

Another difference is that standard databases permit only a fixed set of fields in a record, while textbases permit an arbitrary number of fields in each record. Usually the set is similar from record to record, but a field can be omitted if there is no information to include in it, or a new field added, if something new turns up. In principle each record may have a unique set of fields. In the context of a dictionary, fields relevant to a particular kind of lexical entry can be included in that kind of entry but omitted in others. A Siouanist can include reflexive derivatives with verbs but omit them with nouns. By extension, fields for which the data are missing can also be omitted. Thus, in a comparative dictionary, there need be no reflex field for a language that does not participate in that cognate set. If a given language has no cognate for **wi-šų́ke,* no field for that language appears.

Fields may also be repeated. This allows multiple definitions or multiple examples. Because of these two extensions, the name of the field must typically be included with each instance of the field, to identify the type of field. Thus the illustration for 'dog' has two WI fields for two (related) Winnebago stems including cognates for **wi-šų́ke.*

Standard databases usually require fixed-length, fixed-form fields, whereas textbases permit arbitrary-length, variable-form fields. Thus there is no arbitrary limit on the length of a definition or an example. Cognate citations can be as long as needed, as can comments.

Textbase data fields can have internal structuring called subfields. Our citation fields consist in principle of the following:

- a standard orthography phonological form, in many cases necessarily deduced from a subphonological source recording;
- the source recording itself in the original orthography, if it differs from the standard form;
- the gloss from the source; and
- a reference to the source.

In support of the subfield structure, remarks on a given citation should be placed in an accompanying field, not intermingled arbitrarily with the material just listed. Unfortunately, we arrived at this last principle after the fact, and we have always had a great deal of difficulty adhering to the other stipulations. The subfields are nicely divided with | characters, but in our original scheme we relied on the dagger marking deduced standard forms and the quotation marks around source forms and glosses to impose the structure. It would probably have been a good idea to write a program to critique and/or heuristically correct the format of citation fields and to have encouraged the editors to run it at intervals. In fact, Koontz runs something like this as part of the formatting process, but this is too late in the processing to be optimal.

Returning to the illustration of 'dog', the first DA (Dakota) field there represents a deduced standard form *šų́ka,* recorded "šuŋ'-ka," meaning 'dog', in the Riggs dictionary (1890: 450, col. a). In practice we might early on have recorded only the deduced standard form, omitting the source form and/or the gloss and/or some or all of the reference.

Apart from failing to adhere to our standards, we overlooked one further refinement and, since thinking of it, have debated its merits. We should perhaps have devised some sort of scheme to delimit the parts of forms being compared. To some extent this is obvious, but not marking it explicitly may cover some imprecision in our thinking in some sets and may pose difficulties for readers less familiar with Siouan morphology. Its absence also makes it difficult to use programs to extract tables of sound correspondences. On the other hand, it would be particularly awkward in forms where cognate pieces are discontinuous or where syncope or cluster simplification has distorted the picture of a particular language. Likewise problematic, the obvious schemes for delimiting this sort of thing present difficulties for simple searching programs aimed at forms not instrumented in this way (see §7.1.2 for more details).

The issue of what to use as a key is a very real one in a comparative dictionary database. For a long time we made do with brief glosses, the problem being that the same set or an overlapping one was often lurking somewhere else under a different gloss. It was also sometimes difficult to keep sets pure in terms of phonological correspondences. This latter problem was solved by Jones, who, in the course of his root extension work (see details in §6.1), devised a scheme for representing the phonology of the root in terms of its core and extensions. This helped considerably in sorting out phonologically similar sets, but probably the only way to remove overlapping sets systematically is to have a complete set of indices of where the cited forms occur and refer to these continuously. In fact, some sort of automated identification of problem sets ought to be possible, given the indices.

6.3. THE SOFTWARE. After considering several textbase systems on the market in the late 1980s and one devised with considerable effort by Jones, called SiouxAnn, we selected one called askSam, mostly because it was the only one that seemed relatively friendly to user-defined character sets. It turned out to have several egregious faults, including a strange and (for us) unusable scripting and report-generating tool and a peculiar two-level system of records that interacted with fixed lengths for the lower level of record. Nevertheless, it did permit searches restricted to fields; it did permit the use of user-defined character sets; and although we have long since abandoned it, its traces still remain. In particular, we label fields with labels of the form "label[". The trailing "]" following the field was optional, and we have always omitted it.

We have since replaced askSam with several text editors. This is the approach recommended for use with Lexware by Hsu (1985). All these editors are "programmer's editors," characterized mainly by such features as

- being willing to edit any size file that can be stored on the system;
- ability to edit more than one file at a time;
- availability of pattern-based searching; and
- some sort of scripting language for encapsulating editing procedures.

We have used mainly products called Brief and the Sage Programmer's Editor. Some of us prefer one and some the other.

It should be noted that in the meantime the SIL has introduced a system called Shoebox, now available in a Windows version.[13] This is a textbase system for linguists, and had it been available when we began we would certainly have used it. It labels fields with labels of the form \label, following the conventions of the SIL's Standard Format for textbases, and has a host of features of use to a linguist. In fact, Koontz has been covertly using it in one way or another since it first came out, always converting the textbase to Standard Format first before doing any work with it:

\gl dog
\gc N
\sc Anml
\enh GdeG 11-29-89
\chh done 06-20-90
\chh . . .
\psi |SS *wi-šúke
\or . . .
\pch . . .
\pmv |SS *šúkE
\pda |SS *šúkA
\la |SS šúka |GL dog |SC C
\da |SS ³LB{z}šúka |BA šuŋ´-ka |GL dog |SC R-450a
\st |SS šúga |GL dog |SC PAS
\pwc |SS *šúùke
. . .
\ch |SS šúŋe |GL horse |SC Marsh
\wi |SS šúu̯k |GL dog, horse |SC KM-3005
\wi |SS šu̯u̯gń̩k |GL puppy |SC KM-3002
\pdh . . .

13. In spite of the availability of Shoebox, we suspect that text editors will remain an important tool in projects such as the CSD.

\pse . . .
\com This ancient term has been adapted in historical times . . .
. . .

SIL Standard Format does not have a particular practice for marking subfields, but several of the tools SIL provides use notations like |xx{. . .} to label and delimit use of a zone in which the character format named xx is to be used. As subfields generally have their own distinctive character formatting, this notation is pressed into service in the CSD to represent subfields. However, to save keystrokes and simplify searching subfields with tools not aware of Standard Format conventions, subfields are generally marked |xx . . . in the CSD textbase. Certain single-character subfields use a variant of the more restrictive notation; for example, ^LB{z} represents |LB{z}, which selects the dagger character out of the Lexbats (special lexicographical symbols) character set. The use of "^" in lieu of "|" is a trick to prevent these notations from being processed at the wrong time.

6.4. PRINTING THE DATABASE. To print a simple verbatim copy of the database in our DOS environment requires either the use of a word processor or manual installation of fonts in the printer. None of the textbase systems we have used do printing using user-defined fonts. Most do no printing at all. Because the SIL has been making it easy to use Microsoft Word for DOS as a linguistic tool throughout this period, this has been our printing tool of choice, though other expedients have been used from time to time. We started out with version 4.0, and are currently using versions 5.5 and 6.0. Printing is achieved by exporting the textbase to a text file or by appropriating the existing textbase file, if that is stored in text form. This file is then imported into Microsoft Word, formatted using the Siouan Dictionary font, and printed.

More elaborate reports have always required, unfortunately, the intervention of a programmer. This intervention permits extracting, modifying, and rearranging fields and selecting particular fonts for use with individual subfields or individual characters. The approach we have used has been to

- export the textbase from the textbase program;
- convert the textbase to an intermediate format with a script written in the Mortice-Kern Systems (MKS) Toolkit version of AWK for DOS. AWK is a Unix-derived scripting language, strong on pattern matching and character processing (see Aho, Kernighan, and Weinberger 1988);
- convert the intermediate format into a Microsoft Word for DOS file with one of a series of minor SIL tools, currently CTW; and
- print the resulting file with Microsoft Word for DOS.

This omits several steps required by perverse behavior on the part of one or the other of the applications, and the whole process is too complex to go into in detail in this context, but some general observations on the conversion process and the intermediate format are in order.

What the SIL conversion tools do is convert a Standard Format textbase into a Microsoft Word for DOS file. Unfortunately, for simplicity, the textbases to be converted are assumed to be of a special kind configured to facilitate the specification of the Microsoft Word report format. These format-specifying textbases are not laid out on anything like the data-organizing principles used in laying out the CSD textbase. In the CSD textbase, records represent a set of compared forms, with each field providing a citation for one of the forms, or a comment on it, or a reconstruction. In a format-specifying textbase, the fields correspond instead to paragraphs of the report, and certain specially delimited strings within the fields correspond to stretches of text that are printed with special typefaces and type variants. The conversion script or program that we supply converts the field structure of the CSD textbase into the field structure of the format-specifying textbase and preserves or adds delimiters for parts of fields that need special formatting. See figure 11.2 for an example of the 'dog' record restructured as a format-specifying textbase.

In the form shown in figure 11.2, the fields are \DG (dictionary gloss), \DH (dictionary header), \DA (dictionary article), and \DC (dictionary

\DG |RG{286|RG{.}}{tab}|HW{dog} |GI{N}|RG{,} |GI{Anml}

\DH Original GdeG 11-29-89|RG{.' changed done 06-20-90|RG{;} . . .

\DA |LG{psi}{#}|SS{*wi-šúke}

\DA |FW{cf|RG{.}} . . . |RG{;} . . . |LG{pch} . . . |RG{ụ} |LG{pmv}{#}|SS{*šúkE} |RG{ụ}
|LG{pda}{#}|SS{*šúkA} |RG{ụ} |LG{la}{#}|SS{šúka} |GL{dog} |SC{rtc} |RG{ụ}
|LG{da}{#}|SS{|LB{z}šúka} |BA{śuŋ'-ka} |GL{dog} |SC{R-450a} |RG{ụ}
|LG{st}{#}|SS{šúga} |GL{dog} |SC{PAS} |RGụ |LG{pwc}{#}|SS{*š{ú,}ụke} |RG{ụ} . . .
|LG{ch}{#}|SS{šúŋe} |GL{horse} |SC{Marsh} |RG{ụ} |LG{wi}{#}|SS{šúụk}
|GL{dog|RG{,} horse} |SC{KM-3005}|RG{;} |LG{wi}{#}|SS{šụụgnḭ́k} |GL{puppy}
|SC{KM-3002} |RG{ụ} |LG{pdh} . . . |LG{pse} . . .

\DC This ancient term has been adapted in historical times . . .

Figure 11.2. 'Dog' Record Restructured as a Format-Specifying Textbase.

comment), instead of the individual citations of previous illustrations. In fact, the citations are run together in a single \DA field, representing a single paragraph. The citation fields of the textbase become subfields of this paragraph, each in the |xx{. . .} notation, and they are separated from each other within the paragraph with bullets (|RG{ụ}) introduced by the formatting program. The |RG{. . .} character formatting notation selects the "regular" DOS ASCII character set, while |SS{. . .} selects the Standard Siouan character set, |BA{. . .} selects the BAE (or Dorsey) character set, and so on.

The AWK scripts that convert the CSD textbase to the format-specifying version also introduce standard abbreviations. For example, C as an abbreviation for (Richard T.) Carter is replaced by rtc.

The {#} notations in this form of the database are used in connection with the generation of citation indices. Figure 11.3 is the final report, albeit still at a draft stage of development.

286. **DOG** N, Anml

Original GdeG 11-29-89; changed done 06-20-90; . . .

PSI *wi-šų́ke

cf. . . .· PCH . . . · PMV *šų́kE · PDA *šų́kA · LA šų́ka *dog* RTC · DA †šų́ka śuŋ′-ka *dog* R-450A ·

ST sų́ga *dog* PAS · PWC *šų́ụke · . . . · CH šų́ŋe *horse* MARSH · WI šų́ụk *dog, horse* KM-3005;

WI šụụgnį́k *puppy* KM-3002 · PDH . . . · PSE . . .

This ancient term has been adapted in historical times . . .

Figure 11.3. Final Report.

An essential component of this formatting process is the Microsoft Word style sheet, a file describing a set of section, paragraph, and character formats that the word processor permits users to apply to a word processor file. Each section, paragraph, and specially formatted character string within a paragraph is annotated with the name of the style that applies to it, and changing the style's definition in the style sheet file changes every piece of text annotated with that style name. In the SIL scheme of format-specifying textbases, the names of the fields match the names of the desired paragraph styles in a style sheet. For example, \DG fields are printed with the DG, or Dictionary Gloss, style.

Strings within a paragraph needing special formatting are delimited with sequences like |*xx*{*yyy*}, where *yyy* is the string and *xx* is the name of a character format in the style sheet; for example, in the \DG field the |HW{dog} sequence refers to the HW, or Headword, character style, which involves small bold capitals in the SIL Sophia sans serif font (cf. earlier illustrations).

More recently, SIL has turned its attention to this problem in the Windows context, and we now have two tools for converting from data textbases to Microsoft Word for Windows files, or, actually, to application-independent interchange files in Microsoft Rich Text Format (RTF), which can then be conveyed into Microsoft Word for Windows or any other program that accepts RTF files. Shoebox for Windows can even do a certain amount of printing itself now. The two formatting tools are called Multi-Dictionary Formatter and SF Converter. The former is restricted by the particular set of data textbase field labels it insists on. Its scheme of labels simply does not work for synchronic analyses of many American languages, let alone for comparative dictionaries. The latter, however, is probably flexible enough to replace many features of the conversion scripts that the CSD project has been employing, though not all.

Note that a number of new script writing tools are also now available for the PC, including freeware versions of Perl (Siever, Spainhour, and Patwardhan 1999) and Tcl/Tk (Ousterhout 1994). For some time there has also been a freeware version of AWK from the GNU software project.

7. CONCLUSION.

7.1. SOLVED AND CONTINUING COMPUTER PROBLEMS. Some of the problems we encountered have been solved—greatly reduced or essentially eliminated—by improvements in the general computing environment, though we have not yet adopted all of these. This is particularly true of our character set problems and the twin issues of personal communications and file access (see §4.2). Some of the problems we encountered have known solutions that have not been materially facilitated by developments since the project began but are still within reason with a little work (see §§6.3, 6.4). There are, however, some computing problems we encountered that are only beginning to be solvable or have not yet been solved. We will try to summarize them here.

7.1.1. FILE SHARING PROBLEMS. The project involved five or more workers simultaneously, at four or more sites during the academic year. While the technology of editing shared files is well understood and used in many commercial applications, our lack of networking technology has consistently prevented us from simultaneous editing of a single, centralized set of files. Although file sharing networks with record-based locking have existed throughout the period in which we have been working on the CSD, they are only now beginning to be a standard part of Windows computing and to appear in academic computing outside of computer department experiments. As far as we know, none of the linguistic textbase software

mentioned above supports such file sharing anyway. In the absence of file sharing at this level, we shipped diskettes by mail and divided the database into zones[14] that only one person was allowed to edit. In practical terms we have never managed to get a zone through more than one person during an academic year. During summer meetings, the editors mostly worked on the entire file as a committee.

Another sort of collaboration that better networking might have fostered would be some sort of automated conformance checking of the databases during off-hours by a conformance authority.

Note that even with record-based file locking, there are some procedural questions that must be addressed. There is no point in carefully locking others out of a record while one edits it, if one of these others will be deleting the now modified and unlocked record tomorrow. And what should be done if two editors wish to lock overlapping sets of records for separate purposes? A certain coherence and noncollision of agendas must be maintained, and this can probably only be achieved on a spiritual plane more or less separate from computing. The best computing itself can manage in this sphere is to keep an automated change log and provide facilities for backing out of conflicting changes.

7.1.2. PATTERN MATCHING PROBLEMS. It has already been mentioned that there is a potential for mechanisms indicating what parts of forms are being compared to interfere with searching. It is hard to search for *wasabe* if there are brackets ensconced around the *sabe: wa[sabe]* or separators between the components: *w.a.s.a.b.e* or *wa-sab-e.* A similar problem can occur as a result of interference from subfield or character formatting codes like *|xx{yyy}*. And the same problem occurs with respect to diacritics, which cannot be easily omitted. So searching for *wasabe* will not work if the form in the database is *wasábe* or *wasú̧be.* In a more general fashion, even differences in segments can get in the way. Why shouldn't *wasabe* match *wasape,* and so on? In fact, the truth is that linguists can easily find uses for much more powerful searching tools than currently exist. Shoebox has taken some steps in this direction, by allowing the definition of sets of characters, but not enough, and specialized solutions for linguists are less useful than they might be if they are restricted to particular applications and not available in all applications on a system.

7.2. EVALUATING THE EXPERIENCE. Before we conclude, we want to comment on two more topics: the concept of team editing and the barriers

14. It almost goes without saying that we have also not used any of the standard programming project tools for version control.

to final printing and dissemination of this magnificent product. As most readers probably know, the construction of a dictionary by a team, while normal for such projects as the standard reference dictionaries for European languages, is unusual for "exotic" languages. Naturally, the images of all the disasters of committee-created monsters spring readily to mind. However, in this case there seemed to be no other way to follow up on the 1984 workshop to ensure that we would some day get a dictionary, since both Rankin and Carter, the initiators of the work, had the reputation of waiting until everything was perfect before sharing anything with the rest of us.

In the CSD project, Rankin and Carter are the main lexicographers. Jones was added because of his background in historical linguistics and his crucial knowledge of the northern languages, which were unknown to the rest of us. Koontz was added to the project because of his long-standing interest in the reconstruction of the morphology of the family and because of his computer expertise.

Rood's role is pretty exclusively that of administrator and overseer—fund-raiser, red-tape cutter, and, once in a while, umpire. The personalities of the team members make or break the ability of the team to succeed, and the project manager will always be grateful for the willingness of this team to compromise for the sake of the project. Jones has never complained to Rood about anything, and although both Carter and Rankin have strong opinions about how to reconstruct certain phenomena, they have been able to come to compromises that allow the project to move ahead. Whenever there is a disagreement that does not permit compromise, both sides of the story are told at the relevant entry in the dictionary. Amazingly, there are not very many of those. We should probably emphasize that we are reporting results here; Rood was not usually in the same room in which the discussions that led to the compromises took place, but he never saw any blood, though there were sometimes some scowls.

A major strength of the team concept is the depth of the knowledge that informs the final decisions. Without Jones's specialized understanding of Hidatsa and Crow and his interest in working on their relationships with each other and the rest of the family, we would have had little hope of giving those languages their proper place in the reconstructions. Rankin knows Dhegiha equally deeply, and Carter's specialized knowledge of Mandan and Dakotan fills in details on that end. Both Carter and Rankin have worked extensively with the Southeastern languages to the extent that their documentation permits, and Koontz added not only Omaha-Ponca but also some understanding of Winnebago to the group. Thus we covered the family; no one working alone could have done this much in this length of time. So, if we were asked whether this approach is wise and workable, we would have to answer a resounding "Yes, if . . ."

It is certain that the results of this project are here sooner and stronger because of the multiple contributions from the team.

Naturally, there have been other contributions, mentioned above, but the integration of these contributions remains the achievement of the editors. Finally and importantly, we need to emphasize that Jule Gomez de Garcia has played a major role at various points in this project, although she was nominally hired for data entry. For at least two years, she continued her work without any pay whatsoever when the funding was low, and in many cases her skill in following the garbled directions of the editors saved us lots of time and backtracking. We have saved this mention of Jule's special contributions until last so it will be remembered.

So why hasn't the Comparative Siouan Dictionary been published yet? We have mentioned the perfectionist traits of the editors, and these tendencies will require a fair amount of further editing. Rood still needs to write an introduction and sketch of Siouan structure. The presentations at various Siouan and Caddoan conferences over the past two decades, though, constitute a start toward those introductory chapters, and the polishing and cleaning continue on the plains of Colorado, Kansas, Nebraska, and North Dakota. We are fairly confident that "Carter, Jones, Rankin et al." will be available for your perusal early in the new millennium.