The Handbook of Child Language

Edited by

Paul Fletcher and
Brian MacWhinney

1995

5 Computational Analysis of Interactions

BRIAN MACWHINNEY

Child language research thrives on naturalistic data – data collected from spontaneous interactions in naturally occurring situations. However, the process of collecting, transcribing, and analyzing naturalistic data is extremely time consuming and often quite unreliable. One of the major methodological developments in the field of child language research over the past decade has been the introduction of computerized systems for dealing with the transcription, coding, and analysis of spontaneous production data. One such system – the Child Language Data Exchange System (CHILDES) – will be our focus in this chapter. We will begin by reviewing the background to the formation of the CHILDES system. Next, we will examine the basic CHILDES tools, and the ways these tools can be used to address particular research goals. The chapter will then explore a new range of analytic capabilities planned for the next few years.

1 Background

The dream of establishing an archive of child language transcript data has a long history, and there were several individual efforts along such lines early on. For example, Roger Brown's (1973) transcripts from the children called Adam, Eve, and Sarah were typed onto stencils from which multiple copies were duplicated. The extra copies have been lent to and analyzed by a wide variety of researchers – some of them (Moerk, 1983) attempting to disprove the conclusions drawn from those data by Brown himself! In addition, of course, to the copies lent out or given away for use by other researchers, a master copy – never lent and in principle never marked on – has been retained in Roger Brown's files as the ultimate historical archive. In this traditional model, everyone took his copy of the transcript home, developed his/her own coding scheme, marked codes and tallies directly on the transcript, wrote a

paper about the results and, if very polite, sent a copy to Roger. The original database remained untouched. The nature of each individual's coding scheme and the relationship among any set of different coding schemes could never be fully plumbed.

The dissemination of mimeographed and photocopied transcript data cast a spotlight on the weak underbelly of our analytic techniques in language acquisition research. As we began to compare handwritten and typewritten transcripts, problems in transcription methodology, coding schemes, and cross-investigator reliability became more apparent. But, just as these new problems were coming to light, a major technological opportunity was emerging in the shape of the powerful, affordable microcomputer. Microcomputer word-processing systems and database programs allowed researchers to enter transcript data into computer files which could then be easily duplicated, edited, and analyzed by standard data-processing techniques. The possibility of utilizing shared transcription formats, shared codes, and shared analysis programs shone at first like a faint glimmer on the horizon, against the fog and gloom of handwritten tallies, fuzzy dittoes, and idiosyncratic coding schemes. Slowly, against this backdrop, the idea of a computerized data exchange system for the study of child language development began to emerge.

In 1984 a meeting of 16 child language researchers formally launched the CHILDES system with Brian MacWhinney and Catherine Snow as co-directors. The initial focus on the CHILDES project was on the collection of a nonstandardized database of computerized corpora. Between 1984 and 1986, our work focused on the assembly of a large computerized database of transcripts. As the database grew, it soon became apparent that researchers needed more than a disparate set of corpora transcribed in a confusing diversity of styles. They needed a consistent set of standards both for the analysis of old data and for the collection and transcription of new corpora. During the period from 1986 to 1991, the CHILDES system addressed these needs by developing three separate, but integrated, tools. The first tool is the database itself, the second tool is the CHAT transcription and coding format, and the third tool is the CLAN package of analysis programs. These three tools are presented in detail in MacWhinney (1991a) and illustrated through practical examples in Sokolov and Snow (1994). Researchers who plan to make use of the CHILDES tools will want to consult both of these resources.

The three major components of the CHILDES system are the database, the CHAT transcription systems, and the CLAN programs. The next three sections describe these three basic tools.

2 The Database

The first major tool in the CHILDES workbench is the database itself. Researchers across the globe can now reach the CHILDES database through the

InterNet, retrieving huge amounts of consistently coded child language transcript data. In effect, researchers now have access to the results of nearly a hundred major research projects in over a dozen languages across the last 25 years. Using the CHILDES database, a researcher can directly test a vast range of empirical hypotheses against either this whole database or some logically defined subset.

The database includes a wide variety of language samples from a wide range of ages and situations. Learners include children with language impairments, adults with aphasia, second language learners, and bilingual children. Most importantly, almost all of the data represent real spontaneous interactions in natural contexts, rather than some simple list of sentences or test results. Although more than half of the data come from English speakers, there is also a significant component of non-English data. All of the major corpora have been formatted into the CHAT standard and have been checked for syntactic accuracy. The total size of the database is now approximately 160 million characters (160 MB). The corpora are divided into six major directories: English, non-English, narratives, books, language impairments, and bilingual acquisition. In addition to the basic texts on language acquisition, there is a database from the Communicative Development Inventory (Dale, 1990) and a bibliographic database for child language studies (Higginson and MacWhinney, 1990).

2.1 *Access to the database*

Membership in CHILDES is open. However, members are asked to abide by the rules of the system. In particular, users should not distribute copies of programs or files without permission; they should abide by the stated wishes of the contributors of the data; and they should acknowledge properly all uses of the data and the programs. Any article that uses the data from a particular corpus must cite a reference from the contributor of that corpus. The exact reference is given in a file called *00readme.doc* which is distributed along with each data set. On the top level of the database is the general 00readme.doc file that requests that users also cite MacWhinney (1991) and MacWhinney and Snow (1985) when using the programs and data in published work.

All of the CHILDES materials can be obtained without charge by anonymous FTP to poppy.psy.cmu.edu. InterNet connections that can reach poppy.psy.cmu.edu in Pittsburgh or atila-ftp.uia.be in Antwerp are now widely available at universities both in the United States and abroad. The procedure for transferring files depends on the type of machine you are using and the type of files you wish to retrieve. However, in all cases, you first need to follow certain basic rules for anonymous FTP connections:

1 Connect to poppy.psy.cmu.edu (128.2.248.42) using anonymous FTP. If you get an answer from poppy, then you know that you have InterNet access. If you do not get an answer, you may not have access or access may be temporarily broken.

2 When you receive the request for a username, enter "anonymous." Type in your name as a password.
3 If you want to retrieve data files, type "cd childes" to move to the /childes directory. If you want to retrieve the CLAN programs, type "cd clan" to move to the /clan directory.
4 Type "ls" to view the directory structure and use "cd" again as needed. It is easy to confuse directories with files. When in doubt type "cd filename." If that works, it was a directory. If not, it was a file.
5 Type "binary" to set the transfer type. Although some of the files you wish to retrieve may be text files, the binary mode will work across all file types and it is safest to use it as your default transfer type.
6 Use the "get" command to pull files onto your machine.
7 When you are finished, type "bye" or "quit" to close the connection.

In some cases the machine that is running FTP is not the final destination for the files. For example, some users run a program like Kermit from a DOS machine to connect to a UNIX machine and then use the UNIX machine to reach poppy.psy.cmu.edu. Although the FTP transfer is usually reliable, the second transfer through Kermit is more often error prone and slow. It is better if you can run FTP from the machine that will be the final destination for the files.

Once the files are on your local machine, you must untar them. This means that before you can look at the data, you will need to get a copy of the TAR program. TAR is always available on UNIX systems. If you are running FTP from a Macintosh or a DOS machine, you can retrieve a copy of the TAR program from poppy.psy.cmu.edu. The DOS tar.exe program can be found in the /clan/msdos directory and the Macintosh tar.hqx file can be found in the /clan/macintosh directory. The Macintosh version of TAR must be decoded using binhex 4.0 before it can be used. Once the TAR application is debinhexed, it functions as a normal Macintosh application with menus that are fairly easy to follow. There is also a file called "tar.doc" describing TAR in greater detail.

Once you have retrieved and installed a copy of TAR, you are ready to untar the CHILDES files you have retrieved. The UNIX or DOS TAR command you need to issue is something like this:

```
tar -xvf eng.tar
```

Untarring the files will recreate the original directory structure. Each corpus has been placed into a separate tar file.

In addition to the tar files for the database, you can retrieve the CLAN programs and the CHILDES/BIB database from poppy.psy.cmu.edu also through anonymous FTP. If you are running FTP from a Macintosh, you can retrieve the most recent version of CLAN along with certain Macintosh utilities. You should connect to poppy.psy.cmu.edu and use cd clan/macintosh to

move into the directory with Macintosh programs and utilities. These files are all in binhexed format, as indicated by the .hqx extension. The basic CLAN program is clan.hqx. The file manual.hqx has the CHILDES manual in MS-Word format. The Macintosh tar program is in tar.hqx. Once the files are on your machine, use binhex 4.0 to decode them. Do not use binhex 5.0. When transferring CLAN, also remember to transfer the text files in /clan/lib. DOS users can also use FTP to retrieve the most recent version of CLAN from poppy.psy.cmu.edu. Get all the clan.tar file from the /clan/msdos directory and untar the file.

For users without access to the InterNet, as well as for those who want a convenient way of storing the database, we have published (MacWhinney, 1994) a CD-ROM in High Sierra format which can be read by Macintosh, UNIX, and MS-DOS machines which have a CD-ROM reader. The disk contains the whole database, the programs, and the CHILDES/BIB system. One directory contains the materials in Macintosh format and the other contains the materials in UNIX/DOS format. If you have been thinking about adding CD-ROM capabilities to your system, the availability of this CD-ROM will provide you with an excellent excuse to make the addition. There is no charge for the CD-ROM, but you will have to spend $400 or so for a CD-ROM reader. This unit fits directly into the SCSI port on the Mac. For the IBM-PC, it requires an adaptor card that will provide a SCSI port. Often these are sold along with the CD-ROM reader. CD-ROM access is relatively slow and you cannot write CLAN output files to the CD-ROM, so you may want to copy over particular CHILDES corpora to your hard drive. However, it is possible to run CLAN programs on files on the CD-ROM and to then direct the output to your hard disk. A major advantage of the CD-ROM is that the entire database can be stored on a single stable disk.

For further information on changes to the database and programs, researchers can subscribe to the info-childes@andrew.cmu.edu electronic bulletin board.

2.2 *Reformatting of the database*

All of the current corpora are in good CHAT format. This means that they can pass through the CLAN program called CHECK without producing any errors. However, only a few of the most recent corpora were entered directly into CHAT. The process of converting the older corpora into CHAT required years of careful work. Some corpora had to be scanned into computer files from typewritten sheets. Other corpora were already computerized, but in a wide variety of transcription systems. For each of these datasets, we had to translate the project-specific codes and formats into CHAT. Several sets of files were translated from SALT (Miller and Chapman, 1982–93) using the SALTIN program. For other corpora, special purpose reformatting programs had to be written. The fact that these transcripts now all pass through CHECK without error means that all of the files now have correct headers, correct listings

of participants, and correct matches of coding tiers to main lines. Each main line has only one utterance and every utterance ends with a legal terminator. There are no incorrect symbols in the middle of words and all paired delimiters are correctly matched. However, we cannot yet guarantee that the files are consistently coded on the level of individual words. This level of consistency checking requires processing through the MOR program which will be discussed later.

3 CHAT

The most conceptually difficult task we faced in developing the CHILDES system was the formulation of the CHAT transcription system. From 1984 to 1990, during the period which we now refer to as the period of protoCHAT, we explored a variety of transcription forms. In 1990 we began to finalize the shape of CHAT until it reached the more stable form published in the manual in 1991 (MacWhinney, 1991). Users have expressed happiness with the current status of CHAT and the fact that virtually no changes have been made to the basic conventions since 1990. The finalization of CHAT allowed us to sharpen the workings of the CLAN CHECK program so that it now constitutes a computational implementation of the whole CHAT system.

No coding or transcription system can ever fully satisfy all the needs of all researchers. Nor can any transcription system ever hope to fully capture the richness of interactional behavior. Despite its inevitable limitations, the availability of CHAT as a lingua franca for transcription both within the Program Project and within the general field of child language research has already led to solid improvements in data exchange, data analysis, and scientific precision.

3.1 *Key features of CHAT*

The CHAT system is designed to function on at least two levels. The simplest form of CHAT is called minCHAT. Use of minCHAT requires a minimum of coding decisions. This type of transcription looks very much like the intuitive types of transcription generally in use in child language and discourse analysis. A fragment of a file in minCHAT looks like this:

```
@Begin
@Participants:  ROS Ross Child BRI Brian Father
*ROS:           why isn't Mommy coming?
%com:           Mother usually picks Ross up around 4 PM.
*BRI:           don't worry.
*BRI:           she'll be here soon.
```

*ROS: good.
@End

There are several points to note about this fragment. First, all of the characters in this fragment are ASCII characters. The @Begin and @End lines are used to guarantee that the file was not destroyed or shortened during copying between systems. Each line begins with a three-letter speaker code, a colon, and then a tab. Each line has only one utterance. However, if the utterance is longer than one line, it may continue onto the next line. A new utterance must be given a new speaker code. Commentary lines and other coding lines are indicated by the % symbol.

Beyond the level of minCHAT, there are a variety of advanced options that allow the user to attain increasing levels of precision in transcription and coding. Some of the major specifications available in the full CHAT system are:

1 *File headers.* The system specifies standard file headers such as "Age of Child," "Birth of Child," "Participants," "Location," and "Date" that document a variety of facts about the participants and the recording.
2 *Word forms.* CHAT specifies particular ways of transcribing learner forms, unidentifiable material, and incomplete words. It also provides conventions for standardizing spellings of shortenings, assimilations, interactional markers, colloquial forms, baby talk, and certain dialectal variants.
3 *Morphemes.* There is a system for morphemicization of complex words. Without such morphemicization, mean length of utterance is computer based on words, as defined orthographically.
4 *Tone units.* There is a system for marking tone units, pauses, and contours.
5 *Terminators.* There are symbols for marking utterance terminations and conversational linkings.
6 *Scoping.* CHAT uses a scoping convention to indicate stretches of overlaps, metalinguistic reference, retracings, and other complex patterns.
7 *Dependent tiers.* There are definitions for 14 coding tiers. Coding for three of these dependent tiers has been worked out in detail:
   (a) *Phonological coding.* CHAT provides a single-character phonemic transcription system for English and several other languages called UNIBET. It also provides an ASCII translation for the extended IPA symbol set called PHONASCII.
   (b) *Error coding.* CHAT provides a full system for coding speech errors.

   (c) *Morphemic coding.* CHAT provides a system for morphemic and syntactic coding or interlinear glossing.

The full CHAT system is covered in MacWhinney (1991).

3.2 *How much CHAT does a user need to know?*

CHILDES users fall into two groups. One group of researchers wants to examine corpora, but has little interest in collecting and transcribing new data. Another group of researchers wants to collect and transcribe new data and may be only marginally interested in analyzing old data. Typically, linguists and computer scientists fall into the first category and developmental psychologists and students of language disorders fall into the second category. Researchers in the second group who are using CHAT to transcribe new data soon come to realize that they need to learn all of the core CHAT conventions. Although these users may begin by using minCHAT, they will eventually gain familiarity with all of the conventions used on the main line, as well as with those dependent tier codes relevant to their particular research goals.

Users who are focusing on the analysis of old data may think that they do not need to master all of CHAT. For example, if a researcher wants to track the development of personal pronouns in the Brown corpora, they may think it sufficient to simply look for strings such as "he" and "it." However, this type of casual use of the database is potentially dangerous. For example, users need to understand that, in the Brown corpora, the forms "dem" and "dese" are often used as spelling variants for "them" and "these." Failure to track such variants could lead to underestimates of early pronoun usage. There are dozens of correspondences of this type that researchers need to understand if they are to make accurate use of the database.

When users start to use more detailed analysis programs such as MLU or DSS, the need to understand symbols for omitted elements and repetitions becomes increasingly important. For example, users need to understand that some corpora have been morphemicized in accord with the standards of chapter 6 and that others have not. Users also need to understand how symbols for missing elements or retracings can affect both lexical and syntactic analyses. It is possible that, in many cases, users could reach correct conclusions without a full understanding of the core features of CHAT. However, it is impossible to guarantee that this will happen. It is clear that the best recommendation to CHILDES users is to try to learn as much of CHAT as possible. If a researcher makes erroneous use of the database, these errors cannot be attributed to the CHILDES system, but only to the researcher who has failed to fully learn the system. Reviewers of articles based on the use of CHILDES data need to make sure that the researcher fully understood the shape of the database and the inevitable limitations of any empirical dataset.

3.3 *The importance of dry runs and* CHECK

Before a new user of CHAT and the CLAN programs spends hundreds of hours transcribing data, it is important to spend a few hours conducting small analytic "dry runs." This involves using your favorite editor to create a simple test file with some of the target forms that you plan to track in your larger analysis. Then you should make sure that this file passes cleanly through the CHECK program. It is difficult to overemphasize the importance of learning to use CHECK.

After running CHECK, you should apply those CLAN programs that will track the forms you wish to analyze. If you can demonstrate to yourself that the entire process of data entry and analysis will go through successfully from start to finish for some small sample file, then you can be relatively certain that expansion of the database and addition of further research questions will go smoothly. Failures to follow this simply advice have led to hundreds of wasted hours. *Verbum sapientibus sat.*

3.4 *Beyond CHAT*

The basic desiderata motivating all transcription systems include: readability, computational consistency, high retrievability, category expressivity, and precision. Unfortunately, these goals are often incompatible. In particular, it is often the case that codes which are important for some analysis are ones that we don't even want to look at for another analysis. Moreover, the need to retrieve categories will often require us to transcribe utterances in terms of standard words, phonemes, speech acts, or syntactic categories, despite the fact that often the standard categories do not apply.

These conflicting pressures on transcription systems can only be relieved by allowing the analyst a closer contact to the reality underlying the transcript. One way of providing this is to construct a complete phonological transcription, together with prosodic and intonational markings, as suggested by Peters, Fahn, Glover, Harley, Sawyer, and Shimura (1990). However, the work involved in producing a %pho line is so enormous that only two of the nearly 70 current corpora have full phonological transcripts. A major current goal of the CHILDES project is the creation of a linkage between the transcript and the underlying audiovisual reality that will minimize reliance on particular coding decisions and maximize the analyst's ability to explore the reality of the interaction.

4 CLAN

The third major tool in the CHILDES workbench is the CLAN package of analysis programs. The CLAN (Child Language Analysis) programs were written in the C programming language by Leonid Spektor at Carnegie Mellon University. The programs benefitted from work done by Jeffrey Sokolov, Bill Tuthill, and Mitzi Morris, as well as from the SALT systematization developed by Jon Miller and Robin Chapman (Miller and Chapman, 1982–93).

The CLAN programs can be compiled to run under MS-DOS, UNIX, VMS, XENIX, or Macintosh operating systems. The Center at Carnegie Mellon provides contributing members with executable versions of CLAN on floppies, technical assistance, and a manual for the programs. Researchers who are not planning on contributing to the database can purchase these materials from Lawrence Erlbaum Associates. Most users install the programs on a hard disk along with CHAT files either from their own research projects or from the CHILDES database.

CLAN commands include the program name, as set of options, and the names of the files being analyzed. For example the command

```
freq + f*.cha
```

runs the FREQ program on all the files in a given directory with the ".cha" extension. The "+f" switch indicates that the output of each analysis should be written to a file on the disk. Unless specifically given a file extension name, the FREQ program will figure out names for the new files. Many of the programs have quite a few possible options. Each option is explained in detail in the manual.

The programs have been designed to support five basic types of linguistic analysis (Crystal, 1982; Crystal, Fletcher, and Garman, 1976/89); lexical analysis, morphological analysis, syntactic analysis, discourse analysis, and phonological analysis. Let us look at how CLAN can be used to test hypotheses in each of these four areas.

4.1 *CLAN for lexical analysis*

The easiest types of CLAN analyses are those which look at the frequencies and distributions of particular word forms. for example, it is a simple matter to trace the use of a word like "under" or a group of words such as the locative prepositions. The analysis can be done on either a single file or a group of files. For example, let us suppose that we want to trace the use of personal pronouns in the three children studied by Roger Brown. We would construct a file including all of the personal pronouns with one pronoun on each line and call this file "pronouns." We would then use the FREQ command to count the occurrences of the pronouns in a file with a command like this:

```
freq + spronouns + t*ADA adam01.cha
```

The switch +t*ADA is included in order to limit the tally to only the utterances spoken by the child. If we also want the frequencies of the words spoken by the mother, we would use this command:

```
freq + spronouns + t*MOT adam01.cha
```

If we want to extend our analysis to all of the files in the directory, we can use the wild card:

```
freq + spronouns + t*ADA adam*.cha
```

If we want the collection of files to be treated as a single large file, we can add another switch:

```
freq + spronouns + t*ADA + u adam*.cha
```

The FREQ command is powerful and quite flexible, permitting a large number of possible analyses. The outputs of these analyses can be sent to either the screen or to files. The names of the output files can be controlled. For example, one might want to maintain a group of output files with the extension ".mot" for the frequencies of the mother's speech. These can be kept in a separate directory for further analysis.

The second major tool for conducting lexical analyses is the KWAL program which outputs not merely the frequencies of matching items, but also the full context of the item. For example, the KWAL command that searches for the word "chalk" in the sample.cha file will produce this type of output:

```
kwal + schalk sample.cha
kwal is conducting analyses on:
ALL speaker tiers
********************************************
From file <sample.cha>
----------
***File sample.cha. Line 39. Keyword: chalk
*MOT: is there any delicious chalk?
```

It is possible to include still further previous and following context using additional switches.

Frequency analyses

With tools like FREQ and KWAL, one can easily construct frequency analyses for individual children at specified ages. Many such counts have been produced. However, it is more difficult to move up to the next level of generalization on which a frequency count is constructed across children and ages. First, one would want to tabulate frequency data for the speech of children separately from the speech of adults. And one would not want to automatically combine data from children of different ages. Moreover, we would not want to merge

data from children with language disorders together with data from normally developing children. Differences in social class, gender, and educational level may lead one to make further separations. And it is important to distinguish language used in different situational contexts. When one finishes looking at all the distinctions that could potentially be made, it becomes clear that one needs to think of the construction of a lexical database in very dynamic terms.

Such a database could be constructed using FREQ and other CLAN tools, but this work would be fairly tedious and slow. What we plan to do to address this problem is to build a file with every lexical item in the entire database and attach to each item a set of pointers to the position of the item in every file in which it occurs. These key files and pointer files will be stored along with the database on a CD-ROM. Using the pointers from the master word list to the individual occurrences of words, the user can construct specific probes of this database configured both on facts about the child and facts about the words being searched. The program that matches these searches to the pointer file will be called LEX. Using LEX, it will be possible, for example, to track the frequency of a group of "evaluative" words contained in a separate file in two-year-olds separated into males and females. And the same search can also yield the frequency values for these words in the adult input. Although we may want to publish hard-copy frequency counts based on some searches through this database, the definitive form of the lexical frequency analysis will be contained in the program itself.

Once the LEX tool is completed, the path will be open to the construction of three additional tools. The first of these is a simple extension of the current KWAL program. Currently, researchers who want to track down the exact occurrences of particular words must rely on the use of the +d option in FREQ or must make repeated analyses using KWAL and keep separate track of line numbers. With the new LEX system, instead of running through files sequentially, KWAL will be able to rely on the pointers in the master file to make direct access to items in the database.

Lexical field analyses

A second type of lexical research focuses attention not on the entire lexicon, but on particular lexical fields. Using the +s@file switch with FREQ and KWAL, such analyses can already be computed with the current version of CLAN. Completion of the LEX facility will further facilitate the analysis of lexical fields. For example, using the lexical database, we will be able to examine the development of selected lexical fields in the style of the PRISM analysis of Crystal (1982) and Crystal, Fletcher, and Garman (1976/89). This analysis tracks the child's developing use of content words in 239 lexical subfields. Examples of these fields include farm tools, units of weight measurement, and musical instruments. These 239 fields can be merged into a set of 61 categories which

can, in turn, be merged into nine high level fields. Bodin and Snow (1993) show how analyses of this type can be conducted on the CHLDES database. Likely candidates for intensive examination include mental verbs, morality words, temporal adverbs, subordinating conjunctions, and complex verbs.

Other important semantic fields include closed-class items such as pronouns, determiners, quantifiers, and modals. As Brown (1973), Lahey (1988) and many others have noted, these high frequency closed-class items each express important semantic and pragmatic functions that provide us with separate information about the state of the child's language and cognitive functioning. For example, Antinucci and Miller (1976), Cromer (1991), Slobin (1986), and Weist et al. (1984) argue that tense markings and temporal adverbs are not controlled until the child first masters the relevant conceptual categories.

It is also possible to track basic semantic relations (Bloom, 1975; Lahey, 1988; Leonard, 1976; Retherford, Schwartz, and Chapman, 1981) by studying the closed-class lexical items that mark these relations. In particular, one can follow these correspondences between semantic relations and lexical expressions:

Relation	*Lexical expressions*
Locative	in, on, under, through, by, at
Negation	can't, no, not, won't, none
Demonstrative	this, that
Recurrence	more, again, another
Possession	possessive suffix, of, mine, hers, her
Adverbial	-ly
Quantifier	one, two, more, some
Recipient	to
Beneficiary	for
Comitative	with
Instrument	with, by

Lexical rarity index

A third measure that can be developed through use of the LEX facility is the Lexical Rarity Index or LRI. Currently, the major index of lexical diversity is the type–token ratio (TTR) of Templin (1957). A more interesting measure would focus on the relative dispersion in a transcript of words that are generally rare in some comparison dataset. The more than a child uses "rare" words, the higher the Lexical Rarity Index. If most of the words are common and frequent, the LRI will be low. In order to compute various forms of this index, the LRI program would rely on values provided by LEX.

Another easy way of tracking the emergence of "long" words is to use the WDLEN program which provides a simple histogram of word lengths in a file along with the location of the longest words.

4.2 *CLAN for morphological analysis*

Many of the most important questions in child language require the detailed study of specific morphosyntactic constructions. For example, the debate on the role of connectionist simulations of language learning (MacWhinney and Leinbach, 1991; MacWhinney, Leinbach, Taraban, and McDonald, 1989; Marcus et al., 1991; Pinker and Prince, 1988; Plunkett and Sinha, 1992) has focused attention on early uses and overregularizations of the regular and irregular past tense markings in English. A full resolution of this debate will require intensive study of the acquisition of these markings not just in English, but in a wide variety of languages at a wide sampling of ages. Similarly, the testing of hypotheses about parameter setting within G-B theory (Hyams, 1986; Pizzuto and Caselli, 1993; Valian, 1991; Wexler, 1986) often depends upon a careful study of pronominal markings, reflexives, and wh-words.

During the earliest stages of language learning, the most obvious developments are those involving the acquisition of particular grammatical markings. The study of the acquisition of grammatical markers in English has been heavily shaped by Brown's (1973) intensive study of the acquisition of 14 grammatical morphemes in Adam, Eve, and Sarah and the cross-sectional follow-up by de Villiers and de Villiers (1973a). Since Brown's original analysis, there have been scores of studies tracking these same morphemes in second language learners, normally developing children, and children with language disorders. There have also been many studies of comparable sets of morphemes in other languages.

The 14 morphemes studied by Brown include the progressive, the plural, the regular past, the irregular past, *in, on,* the regular third person singular, the irregular third person singular, articles, the uncontracted copula, the contracted copula, the possessive, the contracted auxiliary, and the uncontracted auxiliary. Brown's framework for morpheme analysis has been extended in systems such as the LARSP procedure by Crystal, Fletcher, and Garman (1976), the DSS procedure by Lee (1974), the ASS procedure of Miller (1981), and the IPSyn procedure of Scarborough et al. (1991). Other markers tracked in LARSP, ASS, and DSS include the superlative, the comparative, the adverbial ending -ly, the uncontracted negative, the contracted negative, the regular past participle, the irregular past participle, and various nominalizing suffixes. Within the CHILDES framework, de Acedo (de Acedo, 1993) shows how CHAT and CLAN can be used to study the Spanish child's learning of grammatical markers. In the next two sections, we discuss several CLAN tools for the study of morphological structure.

Morphological analysis from the main line

Chapter 6 of the CHILDES manual describes a system for coding the presence of grammatical markers on the main line. For example, the word "jumped" can

be coded as "jump-ed." Words that cannot be analyzed into simple combinations of morphemes can be represented using the replacement option as in:

went [: go-ed]

These two forms of coding allow users to search for all instances of the past "-ed." The basic lexical tools of FREQ and KWAL can be used to do this. In a more complex language, such as Italian or Hungarian, main line coding of morphemes tends to become cumbersome and hard to read. To address this problem, we have written a program for the automatic extraction of codes on a secondary %mor tier.

MOR – *Automatic morphological anlaysis*

The more extensive coding needed for many projects requires a complete construction of a part-of-speech analysis for each word on the main line. This analysis is placed on a separate tier called the %mor tier. The coding of the %mor tier is done in accord with the guidelines specified in chapter 14 of the CHILDES manual. Once a complete %mor tier is available, a vast range of morphological and syntactic analyses become possible. However, hand coding of a %mor tier for the entire CHILDES database would require perhaps 20 years of work and would be extremely error-prone and noncorrectable. If the standards for morphological coding changed in the middle of this project, the coder would have to start over again from the beginning. It would be difficult to imagine a more tedious and frustrating task – the hand coder's equivalent of Sisyphus and his stone.

The alternative to hand coding is automatic coding. Over the last three years, we have worked on the construction of an automatic coding program for CHAT files, called MOR. Although the MOR system is designed to be transportable to all languages, it is currently only fully elaborated for English and German. The language-independent part of MOR is the core processing engine. All of the language-specific aspects of the systems are built into files which can be modified by the user. In the remarks that follow, we will first focus on ways in which a user can apply the MOR system for English.

How to run MOR

The MOR program takes a CHAT main line and automatically inserts a %mor line together with the appropriate morphological codes for each word on the main line. The basic MOR command is much like the other commands in CLAN. For example, you can run MOR in its default configuration with this type of command:

mor sample.cha

However, MOR is unlike the other CLAN programs in one crucial regard. Although you can run MOR on any CLAN file, in order to get a well-formed

%mor line, you often need to engage in significant *extra work*. We have tried to minimize the additional work you need to do when working with MOR, but it would be misleading for us to suggest that no additional work is required. In particular, users of MOR will often need to spend a great deal of time engaging in the processes of (1) lexicon building and (2) ambiguity resolution.

Lexicon building. In order to determine whether MOR correctly recognizes all of the words in your transcripts, you can first run MOR on all of your files and then run this KWAL command on the .mor files you have produced:

kwal + t%mor + s"?|*" *.mor

If KWAL finds no question marks on the %mor line, then you know that all the words have been recognized by MOR. If there are question marks in you *.mor output files, you will probably want to correct this problem by running MOR in the interactive update mode. You can then either add the new words to you main lexicon or else create a secondary, corpus-specific lexicon with the missing words needed for this corpus.

Ambiguity resolution. MOR automatically generates a %mor tier of the type described in chapter 14. As stipulated in chapter 14, retraced material, comments, and excluded words are not coded on the %mor line produced by MOR. Words are labeled by their syntactic category, followed by the separator "|," followed by the word itself, broken down into its constituent morphemes.

*CHI: the people are making cakes.
%mor: det|the n|people v:aux|be&PRES v|make-ING n|cake-PL.

In this particular example, none of the words have ambiguous forms. However, it is often the case that some of the basic words in English have two or more part-of-speech readings. For example, the word "back" can be a noun, a verb, a preposition, an adjective, or an adverb. The "^" character denotes the alternative readings for each word on the main tier:

*CHI: I want to go back.
%mor: pro|I v|want inf|to^prep|to
v|go adv|back^n|back^v|back.

The entries in the eng.cld file maintain these ambiguities. However, open-class words in the eng.lex file are only coded in their most common part-of-speech form. The problem of noun–verb ambiguity will eventually be addressed through use of the PARS program, which is currently under development. Those ambiguities which remain in a MOR transcript after the drules and the PARS program has operated can be removed by using MOR in its ambiguity resolution mode. The program locates each of the various ambiguous words one by one and asks the user to select one of the possible meanings.

MOR *for other languages*

In order to maximize the portability of the MOR system to other languages, we have developed a general scheme for representing allomorphic rules and combination rules. This means that a researcher can adapt MOR for a new language without doing any programming at all. However, the researcher/linguist needs to construct: (1) a list of the stems of the language with their parts-of-speech, (2) a set of "arules" for allomorphic variations in spelling, and (3) a set of "crules" for possible combinations of stems with affixes. Building these files will require a major one-time dedication of effort from at least one researcher for every language. Once the basic work of constructing the rules files and the core lexicon files is done, then further work with MOR in that language will be no more difficult that it currently is for English. However, construction of new rules files is an extremely complex process. And construction of a closed-class and open-class lexicon will also take a great deal of time. Although no programming is required, the linguist building these files must have a thorough understanding of the MOR program and the morphology of the language involved. Complete documentation for the construction of the rules files is available from Carnegie Mellon and will also be included in the next edition of the CHILDES manual.

4.3 *CLAN for syntactic analysis*

Once a %mor line has been constructed, either through use of MOR or through hand coding, a variety of additional morphosyntactic analyses are then available. Instead of analyzing the lexical items on the main line, programs can now analyze the fuller morphosyntactic representation on the %mor line. The simpler forms of analysis can still be done using FREQ and KWAL. However, several other programs add additional power for morphosyntactic analysis. The MLU program can compute the basic Mean Length of Utterance index in a variety of ways (Rollins, 1993). COMBO extends the power of FREQ and KWAL by permitting more complete Boolean string matching. For example, if the user wants to search for all instances of a relative pronoun followed eventually by an auxiliary verb, it is possible to compose this search string in COMBO. Certain types of matching between the main line and the %mor line can be achieved using the +s switch in the MODREP program. In addition, the COOCCUR program can be used to tabulate sequences of syntactic structures appearing on the %mor line.

More complex analyses of syntactic development require us to deal with structures defined in terms of traditional syntactic categories such as Subject, Object, and Main Verb. Among the most important syntactic structures examined by procedures such as LARSP, ASS, IPSyn, and DSS are these:

Structure	*Example*
Art + N	the dog
Adj + N	good boy
Adj + Adj + N	my new car
Art + Adj + N	the new car
Adj/Art + N + V	my bike fall
V + Adj/Art + N	want more cookie
N + poss + N	John's wallet
Adv + Adj	too hot
Prep + NP	at the school
N + Cop + PredAdj	we are nice
N + Cop + PredN	we are monsters
Aux + V	is coming
Aux + Aux + V	will be coming
Mod + V	can come
Q + V	who ate it?
Q + Aux + V	who is coming?
tag	isn't it?
aux + N	are you going?
S + V	baby fall
V + O	drink coffee
S + V + O	you play this
X + conj + X	boy and girl, red and blue
V + to + V	want to swim
let/help + V	let's play
V + Comp	I know you want it
Sent + Conj + Sent	I'll push and you row
V + I + O	read me the book
N + SRel	the one you have in the bag
N + ORel	the one that eats corn
S + Rel + V	the one I like best is the monster
passive	he is kicked by the raccoon
Neg + N	no dog
Neg + V	can't come
PP + PP	under the bridge by the river
comparative	better than Bill

Several of these structures also define some of the semantic relations that have been emphasized in previous literature. These include recipient (direct object), agent (subject in actives), verb, and object.

The discussion in this section has focused on the construction of indicators for development in English. However, these same tools can also be usefully applied to basic issues in crosslinguistic analysis. Once we have collected a

large database of transcripts in other languages and created a full %mor tier encoding, we can ask some of the basic questions in crosslinguistic analyses. Are there underlying similarities in the distribution of semantic relations and grammatical markings used by children at the beginning of language learning? Exactly which markings show the greatest language-specific divergences from the general pattern? How are grammatical relations marked as ergative in one language handled in another language? Under what circumstances do children tend to omit subject pronouns, articles, and other grammatical markers?

4.4 *CLAN for discourse and interactional analyses*

Many researchers want to track the ways in which discourse influences the expression of topic, anaphora, tense, mood, narrative voice, ellipsis, embedding, and word order (Halliday and Hasan, 1976; MacWhinney, 1985b). To do this, researchers need to track shifts in narrative voice, transitions between discourse blocks, and foreground–background relations in discourse. They are also interested in the ways in which particular speech acts from one participant give rise to responsive or nonresponsive speech acts in the other participant. CLAN provides several powerful tools for examining the structures of interactions and narrations.

Coder's Editor

The most important CLAN tool for data coding is the CED Coder's Editor, which is a new program in CLAN 2.0. CED can lead to remarkable improvements in the accuracy, reliability, and efficiency of transcript coding. If you have ever spent a significant amount of time coding transcripts or if you plan to do such coding in the future, you should definitely consider using CED. More importantly, CED is at the core of our plans for an integrated exploratory workbench, to be discussed in the next section.

CED provides the user with not only a complete text editor, but also a systematic way of entering user-determined codes into dependent tiers in CHAT files. The program works in two modes: coder mode and editor mode. Initially, you are in editor mode, and you can stay in this mode until you learn the basic editing commands. The basic commands have been configured so that both Word Perfect and EMACS keystroke equivalents are available. If you prefer some other set of keystrokes, the commands can be rebound.

In the coding mode, CED relies on a codes.lst file created by the user to set up a hierarchical coding menu. It then moves through the file line by line asking the coder to select a set of codes for each utterance. For example, a codes.lst list such as


```
$MOT
:POS
    :Que
    :Res
:NEG
$CHI
```

would be a shorter way of specifying the following codes:

```
$MOT:POS:Que
$MOT:POS:Res
$MOT:NEG:Que
$MOT:NEG:Res
$CHI:POS:Que
$CHI:POS:Res
$CHI:NEG:Que
$CHI:NEG:Res
```

This coding system would require the coder to make three quick cursor movements for each utterance in order to compose a code such as $CHI:NEG:Res.

Chains and sequences

Once a file has been fully coded in CED, a variety of additional analyses become possible. The standard tools of FREQ, KWAL, and COMBO can be used to trace frequencies of particular codes. However, it is also possible to use the CHAINS, DIST, and KEYMAP programs to track sequences of particular codes. For example, KEYMAP will create a contingency table for all the types of codes that follow some specified code or group of codes. It can be used, for example, to trace the extent to which a mother's question is followed by an answer from the child, as opposed to some irrelevant utterance or no response at all. DIST lists the average distances between words or codes. CHAINS looks at sequences of codes across utterance. Typically, the chains being tracked are between and within speaker sequences of speech acts, reference types, or topics. The output is a table which maps, for example, chains in which there is no shift of topic and places where the topic shifts. Wolf, Moreton, and Camp (1993) apply CHAINS to transcripts that have been coded for discourse units. Yet another perspective on the shape of the discourse can be computed by using the MLT program which computes the mean length of the turn for each speaker.

Recasts

Currently there is only one CLAN program that focuses on the lexical and syntactic match between successive utterances. The is the CHIP program

developed by Jeffrey Sokolov and Leonid Spektor. CHIP is useful for tracking the extent to which one speaker repeats, corrects, or expands upon the speech of the previous speaker. Sokolov and Moreton (1993) and Post (1993) have used it successfully to demonstrate the finetuning of instructional feedback to the language learning child.

Discourse display

There is more to a transcript than a series of codes and symbols. The superficial form of a transcript can also lead us to adopt a particular perspective on an interaction and to entertain particular hypotheses regarding developments in communicative strategies. For example, if we code our data in columns with the child on the left, we come to think of the child as driving or directing the conversation. If we decide instead to place the parents' utterances in the left column, we then tend to view the child as more reactive or scaffolded. Ochs (1979) noted that such apparently simple decisions as the placement of a speaker into a particular column can both reflect and shape the nature of our theories of language development. Because it is important for the analyst to be able to see a single transcript in many different ways, we have written three new CLAN programs that provide alternative views onto the data. The basic principle underlying these data display programs is the motto of "different files for different styles." The display programs – COLUMNS, LINES, and SLIDE – are designed to facilitate the alternative ways of viewing turns and overlaps.

The COLUMNS program produces CHAT files in a multicolumn form that is useful for explorations of turn-taking, scaffolding, and sequencing. COLUMNS allows the user to break up the one-column format of standard CHAT into several smaller columns. For example, the standard 80 character column could be broken up into four columns of 20 characters each. One column could be used for the child, one for the parent, one for situational descriptions, and one for coding. The user has control over the assignment of tiers to columns, the placement of the columns, and the width of each separate column. As in the case of files produced by SLIDE, files produced by COLUMNS are useful for exploratory purposes, but are no longer legal CHAT files and cannot be reliably used with the CLAN programs.

Yet another form of CLAN display provides a focus on overlaps and cross-tier correspondences. Using SLIDE, a CHAT file can be displayed as a single unbroken stretch of speech across an "infinite" left-to-right time line. Whereas standard CHAT files use carriage returns to break up files into lines, a file displayed in SLIDE has all carriage returns removed. The SLIDE program converts a CHAT file into a set of single long lines for each speaker. These lines can be scrolled across the computer screen from left to right. At any point in time, only 80 columns are displayed, but the user can rapidly scroll to any other point in this single left–right line by using the cursor keys. When two speakers overlap in a conversation, SLIDE displays the overlapped portions on top of each other. SLIDE can also be used to display accurate placement of material otherwise indicated by <aft> and <bef> and to provide correct display of the match between morphemes on a %mor line with corresponding words on the main line as required in many systems of interlinear morphemicization. This form of display provides far better time–space iconicity than any previous form of display. Of course, this display cannot be captured on the printed page; it is only available on the computer screen because of its capacity to scroll almost limitlessly left to right. An earlier noncomputerized prototype for SLIDE can be found in Ervin-Tripp (1979).

Finally, CED itself is now capable of varying the way in which transcripts are viewed by allowing the user to suppress display of particular tiers or even data from particular speakers right in the CED window while coding and editing the transcript.

4.5 *CLAN for phonological analyses*

Despite all the care that has gone into the formulation of CHAT, transcription of child language data remains a fairly imprecise business. No matter how carefully one tries to capture the child's utterances in a standardized transcription system, something is always missing. The CHAT main line induces the transcriber to view utterances in terms of standard lexical items. However, this emphasis tends to force the interpretation of nonstandard child-based forms in terms of standard adult lexical items. This morphemic emphasis on the main line can be counterbalanced by including a rich phonological transcript on the %pho line. As Peters, Fahn, Glover, Harley, Sawyer, and Shimura (1990) have argued, the inclusion of a complete CHAT %pho line is the best way to convey the actual content of the child's utterances, particularly at the youngest ages. CHAT provides two systems for transcribing utterances on the %pho tier. For coarse transcription, researchers can use the UNIBET systems that have been developed for English and several other languages. For a finer level of phonetic transcription, researchers can use the PHONASCII rendition of IPA notation.

Analysis of the %pho line

The construction of a complete %pho tier for even a few hours of data is a formidable task. Verification of the reliability of that transcription is an even bigger problem. However, once this tier is produced, CLAN provides several programs to facilitate analysis. The two programs that are most adapted to this analysis are PHONFREQ, which computes the frequencies of various segments, separating out consonants and vowels by their various syllable positions, and MODREP, which matches %pho tier symbols with the corresponding main line text. For more precise control of MODREP, it is possible to create a separate %mod line in which each segment on the %pho corresponds to exactly one segment on the %mod line.

Linking to a digitized record

Although inclusion of a complete %pho line is a powerful tool, even this form of two-tier transcription misrepresents the full dynamics of the actual audio record. If the original audiotapes are still in good condition, one can use them to continue to verify utterances. But there is no way to quickly access a particular point on an audiotape for a particular utterance. Instead, one has to either listen through a whole tape from beginning to end or else try to use tape markings and fastforward buttons to track down an utterance. The same situation arises when the interaction is on videotape.

Computer technology now provides us with a dramatic new way of creating a direct, immediately accessible link between the audio recording and the CHAT transcript. The system we have developed at Carnegie Mellon, called Talking Transcripts, uses fast optical erasable disks, a 16-bit digitizer board, and the Macintosh operating system to forge these direct links. Once a large sound file has been written to disk, the transcriber can use CED to control access to the file during the coding of a new transcript. Although this process requires some additional time setting up the basic digitization, this investment pays for itself in facilitating high-quality transcription. Each utterance can be played back exactly and immediately without having to use a reverse button or foot pedal.

As a user of this new system, I have found that having the actual audio record directly available gave me a much enhanced sense of an immediate relation between the transcript and the actual interaction. The impact on the transcriber is quite dramatic. Having the actual sound directly available does not diminish the importance of accurate transcription, because the CLAN programs must still continue to rely on the CHAT transcript. However, the immediate availability of the sound frees the transcriber from the fear of making irrevocable mistakes, since the ongoing availability of the audio record means that codes can always be rechecked for reliability.

Phonological analysis with a digitized record

Using this new link between the CHAT %pho line and digitized speech, the way is now open for us to design an entire Phonologist's Workbench grounded on the immediate availability of actual sound. The new programs for phonological analysis that we now plan to write include:

1. *Inventory analysis.* We will extend the PHONFREQ program, so that it can compute the numbers of uses of a segment across either types or tokens of strings on the %pho line. The program will also be structured so that the inventories can be grouped by distinctive features such as place or manner or articulation or by groups such as consonants versus vowels. The ratio of consonants to vowels will be computed. Summary statistics will include raw frequencies and percentage frequency of occurrence for individual segments. Nonoccurrences in a transcript of any of the standard segments of English will be flagged.
2. *Length.* The MLU program will be used to compute mean length of utterance in syllables. This can be done from the %pho line, using syllable boundaries as delimiters.
3. *Variability.* The MODREP program will be made to compute the types and tokens of the various phonetic realizations for a single target word, a single target phoneme, or a single target cluster. For example, for all the target words with the segment /p/, the program will list the corresponding child forms. Conversely, the researcher can look at all the child forms containing a /p/ and find the target forms from which they derive.
4. *Homonymy.* Homonymy refers to a child's use of a single phonetic string to refer to a large number of target words. For example, the child may say "bo" for *bow, boat, boy, bone,* etc. The MODREP program will calculate the degree of homonymy observed by comparing the child's string types coded on the %pho tier with the corresponding target forms coded on the %mod tier.
5. *Correctness.* In order to determine correctness, the child pronunciation (%pho line) must be compared with the target (%mod line). The MODREP program will be modified to compute the number of correct productions of the adult target word, segment, or cluster. For example, the percentage consonants correct (PCC) will be computed in this way.
6. *Phonetic product per utterance.* This index (Bauer, 1988; Nelson and Bauer, 1991) will be computed by a new CLAN program called PHOPRO. The index computes the phonetic complexity of the utterance as a function of the number of place of articulation contrasts realized. This index is low if everything is at one place of articulation; it is high if all points of articulation are used.
7. *Phonological process analysis.* Phonological process analyses search for systematic patterns of sound omission, substitution, and word formation that children make in their simplified productions of adult speech. Thus, such processes refer to classes of sounds rather than individual sounds. Process analysis must be based upon the comparison of the %pho and %mod tiers. The Clan Analysis of Phonology, or CAP, will examine rates of consonant deletion, voicing changes, gliding, stopping, cluster simplification, and syllable deletion. In addition, nondevelopmental error will be identified an calculated (Shriberg, 1990).
8. *PHONASCII and UNIBET code modifications.* PHONASCII and UNIBET codes will be modified and/or elaborated to enable cross-tier analysis.
9. *Automatic phonetic transcription of high-frequency words.* To facilitate phonetic and phonological transcription of corpora, we will develop an on-line users reference to provide automatic phonological coding

of the 2,000 most frequently used words in the English language to facilitate phonetic transcription of naturalistic speech data (e.g. words such as "and" and "the" will not have to be redundantly transcribed each time they occur).

10 *Phonologist's reference.* To help beginning phonologists and to stabilize reliability for trained phonologists, we will have available a complete set of digitized speech samples for each phonological symbol used in either UNIBET or PHONASCII.

11 *Transcription playback.* The same phonological database used by the Phonologist's reference can also be used to playback the sounds of candidate transcriptions.

Alongside the development of programs to support these analyses, we will also be working to broaden the CHILDES database of phonological transcripts. There are very few computerized transcripts currently available, so we can reasonably start from scratch in this area. Because we are starting from scratch, we can require that all transcripts in the CHILDES phonological database be accompanied by good quality tape recordings which will be digitized at CMU and then distributed through CD-ROM.

4.6 *Utilities*

CLAN also includes a variety of features to make life easier when manipulating files and transcripts. The CHSTRING program can be used to make simple string replacements across collections of files. The CAPWD program prints all capitalized words. The GEM program is designed to allow the user to place important passages into a file for later analysis. Using a text editor, the user marks the passages to be stored. GEM then uses these marks to determine what should be excised and placed in the "gems" file. A good example of the use of GEM with FREQ can be found in Post (1993). For users working with files from SALT, the SALTIN program helps in the conversion to CHAT. The LINES program allows users to mark their CHAT files with line numbers. DATES can be used to compute a child's age, given the current date and the child's birthdate. The BIBFIND program is used to access the CHILDES/BIB database. In addition to these utility programs, the CLAN interface includes a keystroke editing function, some help facilities, and a program for displaying text files called PAGE. The Macintosh version of CLAN also has pulldown menus that can be used to construct CLAN commands.

5 The Future

If the CHILDES database is to continue to grow, we must continue to receive extensive cooperation from individual scientists. Researchers who use the

CHILDES tools to collect new data have the responsibility to contribute these new data to the database. In particular, researchers whose work has benefitted from government support have an obligation to contribute to scientific progress by adding their data to the database. In fields such as the sequencing of proteins in DNA, researchers, journals, and the government have set the requirement that only data which are publicly available in the Human Genome database can be published. A similar policy for language development studies would ensure the stable and continued development of the CHILDES database. Until such a policy is developed, voluntary acceptance of these responsibilities will guarantee continued growth of the database.

We expect that new additions to the CHILDES database will no longer require reformatting, since they will be transcribed in CHAT from the start. Already, we are starting to receive most new data files in CHAT format. Soon we expect this to become the norm. The database will also continue to grow beyond its original scope. The first corpora included in the database were on first language acquisition by normal English-speaking children. In the future, the database will grow to include large components of second language acquisition data, adult interactional data, and a variety of data from children with language disorders. The numbers of languages represented will continue to grow. As the database grows, it will be important to distinguish between the CHILDES system, the Aphasia Language Data Exchange System (ALDES), the Second Language Acquisition Data Exchange System (SLADES), and the overall Language Data Exchange System (LANDES).

As multimedia computational resources become increasingly available, and as fractal video compression methodology becomes more widely distributed, the CHILDES database will shift from its current concentration on ASCII transcripts to a focus on transcripts accompanied by digitized audio and video. Using the CED editor, links between events in the audio and video records will be tied to an increasingly rich set of links in the transcript. These links will be increasingly dynamic, allowing the user to move around through the audio and video records using the transcript as the navigational map. The full digitization of the interaction will allow the observer to enter into the interaction as an explorer. This is not the virtual reality of video adventures. The scientist is not seeking to change reality or to interact with reality. Instead, the goal is to explore reality by viewing an interaction repeatedly from many different perspectives. These new ways of viewing a transcript will be important for phonological and grammatical analyses, but their most important impact will be on the analysis of interactional structure and discourse. Having full video and audio immediately available from the transcript will draw increased attention to codes for marking synchronies between intonational patterns, gestural markings, and lexical expressions in ongoing interactional relations.

The construction of this new multimedia transcript world would allow us to begin work on the successor to the CHILDES Project. This is the Human Speech Genome Project. One of the first goals of the Speech Genome Project would be the collection, digitization, transcription, parsing, and coding of

complete speech records for all the verbal interaction of a set of perhaps a dozen young children from differing language backgrounds. They might include, for example, a child learning ASL, a child with early focal lesions, a child growing up bilingual, and children with varying family situations. The multimedia records will allow us to fully characterize and explore all of the linguistic input to these children during the crucial years for language learning. We will then be in a position to know exactly what happens during the normal course of language acquisition. We can examine exactly how differences in the input to the child lead to differences in the patterns of language development. We will have precise data on the first uses of forms and how those first uses blend into regular control. We will be able to track all types of errors and first usages with great precision.

Alongside this rich new observational database, the increased power of computational simulations will allow us to construct computational models of the language learning process that embody a variety of theoretical ideas. By testing these models against the facts of language learning embodied in the Speech Genome, we can both refine the models and guide the search for new empirical data to be included in the multimedia database of the future.

ACKNOWLEDGMENTS

This work was supported from 1984 to 1988 by grants from the John D. and Catherine T. MacArthur Foundation, the National Science Foundation, and the National Institutes of Health. Since 1987, the CHILDES Project has been supported by grants from the National Insitutes of Health (NICHHD). For full acknowledgments and thanks to the dozens of researchers who have helped on this project, please consult pages viii and ix of the manual (MacWhinney, 1991). The CLAN programs were developed by Leonid Spektor. Mitzi Morris wrote the DSS and MOR programs with extensive help from Julia Evans, Leonid Spektor, Roland Hausser, Carolyn Ellis, and Kim Plunkett. Nan Bernstein-Ratner collaborated in the development of guidelines for the creation of a phonological analysis system. Ideas regarding the Talking Transcripts project came from Helmut Feldweg and Sven Strömqvist. Joy Moreton, Catherine Snow, Barbara Pan, and Lowry Hemphill helped test and design the CHAINS and CED programs. Important suggestions for modifications of CHAT coding came from Judi Fenson, Frank Wijnen, Giuseppe Cappelli, Mary MacWhinney, Shanley Allen, and Julia Evans. Roy Higginson was the chief compiler of the CHILDES/BIB system. Steve Pinker suggested a name for the Human Speech Genome Project. George Allen designed the UNIBET and PHONASCII systems.

Social and Contextual Influences