Handbook of Research in Language Development Using CHILDES

Edited by

Jeffrey L. Sokolov
University of Nebraska at Omaha
Harvard Graduate School of Education

and

Catherine E. Snow
Harvard Graduate School of Education

LEA
1994 LAWRENCE ERLBAUM ASSOCIATES, PUBLISHERS
Hillsdale, New Jersey Hove, UK

13 New Horizons for CHILDES Research

Brian MacWhinney
Carnegie Mellon University

Conversational interactions between real parents and real children are the empirical bedrock of the study of child language acquisition. It is through these interactions that children are guided through the grammatical and interactional intricacies of their mother tongue. It is in these interactions that children demonstrate most clearly their successful control of the structure and functioning of their language, while also demonstrating gaps in their control of structures, functions, and interactional processes. Any satisfying theory of language acquisition, be it nativist or empiricist, must eventually be able to make contact with the actual shape of these real conversational interactions.

One of the most challenging aspects of the study of language learning is the incredible complexity of real interactional events. Whenever we try to capture this complexity within the confines of some sort of transcription or coding system, the richness of the real world inevitably bubbles out of our test tube. Given this, any overestimation of the veridicality of our transcriptions can lead us to misperceive significant aspects of the language learning situation. These fundamental problems in observational and measurement technique are common to all of the natural and physical sciences. Certainly, there is no reason to give up the study of child language acquisition because of imperfections in measurement. Instead, we should view these imperfections as opportunities for further achievements. Some of the greatest advances in science have involved the construction of tools that address basic issues in measurement and observation. Consider the impact of physical tools such as the microscope, the telescope, the computer, the spectrogram, the polymerase chain reaction, or the linear accelerator. Consider the impact of methodological tools such as the alphabet, the phoneme, the number zero, the calculus, LISP, or Linnaean taxonomy. Much can be said for the importance of physical devices and methodological tools in sustaining the entire edifice of scientific progress.

The CHILDES Project has as its goal the construction of tools that will facilitate both the veridical representation of conversational interactions and the smooth computational analysis of important properties of these interactions. The first major publication of the CHILDES Project was the CHILDES manual (MacWhinney, 1991). The manual presented three major tools for child language analysis: the CHAT transcription system, the CLAN data analysis programs, and the CHILDES database of files transcribed in the CHAT system. The emphasis in the manual was upon specification rather than explanation. Readers were given a set of specific guidelines and detailed descriptions of programs and codes, but the manual gave virtually no examples of the actual use of these tools in real research projects. At the time the manual was written, our plan was to supplement the specifications given in the **manual** with a series of detailed examples and tutorials in a separate **handbook.** This separate handbook – the book which you are now reading – provides hands-on, concrete, detailed examples of real analytic problems and the specific CLAN techniques that can be used to address these problems. The handbook fills the various gaps left in the manual in a way that allows the two books to form a coordinated whole.

At times, the detailed expositions in both the manual and the handbook may have seemed excessive or even tedious. True enough, the learning curve for CHAT and CLAN is unfortunately rather steep. But, once these tools are mastered, they are relatively easy to apply. Moreover, CHAT and CLAN provide a publicly available comprehensive antidote to inaccuracy and inefficiency in data analysis. How inaccurate and inefficient child language data analysis can be is something that many of us would like to forget. The new generation of child language researchers may tend to take precise computational tools for granted, but those of us who have spent years laboring with hand-written transcripts understand how far we have come from the days of diary notebooks, hand-compiled concordances, and blurred mimeographed copies. With the new tools presented in this book, child language researchers no longer need to spend hours poring over transcripts looking for a single use of a word. We no longer need to mark tallies of word occurrences in the margins of our printed transcripts and then turn through our notebooks, page by page, adding up these tallies by hand, only to realize after hours of work that we have been ignoring some crucial dimension and that the whole analysis has to be started again from scratch. Theoreticians no longer have to base major theoretical claims on a few scattered examples from handwritten transcripts photocopied from colleagues. Above all, the entire field now has direct, on-line access to the core set of actual empirical observations upon which our empirical generalizations

regarding language learning have been based. This direct access allows us a more direct understanding of both the strengths and the weaknesses of this empirical base.

Yes, we have come a long way, but there is still a long way to go. There is still too much tedium in transcript analysis; our computer programs are still too difficult and clunky; the link between the transcript and the actual visual and auditory events is still far too tenuous and inexact; and there are still many languages and types of children for which child language data are not available. Although there seems to be a broad consensus regarding the basic distinctions that must be captured in transcripts, the actual use of these categories by real transcribers has not yet been fully tested for reliability and consistency. The CHAT system presents a fairly complete framework for the coding of syntax, morphology, phonology, speech acts, and speech errors, but the coding of large amounts of child language data using these systems has only now begun. Until large sets of data have been coded with these specialized systems, we will not be able to fully appreciate the possibilities for computerized analysis of child language corpora. There is still an enormous amount of work to be done before we can say that the computer has reached its full potential as a way of allowing us to better understand language development.

Although we may not yet have arrived in the Promised Land, neither are we wandering adrift in the Desert. The CHILDES tools provide us not only with a solid basis for current work, but also a platform upon which we can stand to reach higher and to see further. Using this platform, we can begin to see what transcript analysis will look like at the beginning of the 21st century. This chapter uses the current CHILDES platform as a way of gaining that broader vision. From where we now stand, there are three horizons across which we can gaze. The first and nearest horizon involves the work of the last two years that has extended, perfected, and enriched the three basic tools outlined in the CHILDES manual. The second, intermediate horizon involves the construction of a blueprint for several new tools that will reshape the way in which we interact with our datasets. The third horizon is the one that extends into the next century. It is the distant horizon with its unclear profile that allows us to dream most wildly about powerful computers, enormously rich datasets, intuitive user interfaces, multimedia exploratory reality, and major new conceptual challenges. Let us begin our quest by making a careful inspection of the recent past. In this chapter, I assume that the reader has now completed an overview of both the manual and the handbook, is basically

comfortable with use of the three current CHILDES tools, and is now interested in looking at future directions in the CHILDES Project.

13.1. The Recent Past

The nearest horizon is the one bounded by the publication of the CHILDES manual in 1991 and the present. The publication of the manual in 1991 marked the end of the period which we now refer to as "proto-CHAT." From 1986 to 1990, several versions of the CHAT/CLAN system were circulated in photocopied form. The constant evolution of the system during that period was exciting, but it also often proved frustrating as researchers found they had to continually update both their transcribed files and their understanding of this changing system. In 1991 we decided that the time had come to stabilize the core systems for both CHAT and CLAN. The publication of the manual in 1991 marks the end of the period of proto-CHAT. The stabilized system of CHAT and CLAN published in the manual is the one that is being used by scores of research groups internationally. Since the publication of the manual, the core of both CHAT and CLAN have remained stable, and we intend to continue to maintain the stability of this core. At the same time, we have extended the core of CLAN by adding new programs, new facilities, new options, and new systems for coding.

The next two sections provide an overview of recent changes to CHAT and CLAN. They do not have the tutorial quality of the rest of this handbook. Rather, they are intended to provide an introduction to some new features that will be further documented in the next edition of the manual and in future editions of this handbook. The version of CHAT and CLAN in the first edition of the manual was CHAT 1.0 and CLAN 1.0. The versions I discuss here are CHAT 2.0 and CLAN 2.0.

13.1.1. CHAT 2.0

Since the end of 1990, changes to the CHAT transcription and coding system have been extremely few. The chief changes involved the addition of these coding symbols:

1. +//. In addition to the symbol +/. which codes interruption by another speaker (page 43), CHAT now allows the symbol +//. for self-interruption.

Some researchers want to distinguish incompletions involving a trailing off and then a restart from incompletions involving an abrupt self-interruption. When an incompletion is not followed by further material from the same speaker, the +... symbol should always be selected. However, when the speaker breaks off an utterance and starts up another, the +//. symbol can be used. There is no hard and fast way of distinguishing cases of trailing off from self-interruption. For this reason, some researchers prefer to avoid making the distinction. Researchers who wish to avoid making the distinction, should use only the +... symbol.

2. **+/?** The +/. symbol (page 43) coded only for the interruption of declarative utterances. However, it is important to be able to note that the utterance being interrupted is a question. The symbol +/? is now available for use at the end of an interrupted question.

3. **[/-]** The [//] code is used for coding false starts with retracings (page 52). However, sometimes a speaker makes a false start, but does no retracing. This can be coded with the [/-] symbol. It is important to avoid confusing false starts without retracings from simple self-interruptions. False starts without retracings continue the same basic subject material, whereas self-interruptions involve dropping the current subject and starting up with something entirely new.

4. **$=text(text)** The previous version of CHAT used the form $=text{text} to mark the locus of errors on the **%err** line (page 84). However, the curly braces did not interact correctly with core features of the CLAN programs, so we were forced to change them to parentheses.

5. **[Ø text]** Omitted words can be coded as Øword (page 26). However, some readers tend to miss the initial Ø and think that the word is really being produced. To mark the omission more clearly, transcribers can include the omitted form in square brackets. For the purposes of the CLAN programs, these two forms of coding are equivalent.

For examples of passages in which these codes could be used, please refer to the page citations given.

Other changes to the CHAT documentation include corrections to the IPA consonant table on page 73, correction of "ASCII" to "UNIBET" in the UNIBET tables for English and Italian, addition of UNIBET tables for German and Brazilian Portuguese, and several changes to the tables for Dutch and French. In chapter 14, new tables have been added for English part-of-speech codes, affixes and clitics, and sample morphological codings. The use of the colon within part-of-speech codes has been clarified and the placement of the # sign on prefixes has been moved. Researchers who are doing extensive coding on the **%mor** should obtain a complete copy of the revised version of chapter 14.

The fact that so few changes were made to basic CHAT codes during the years of 1990-1993 is a good sign. It indicates that the core system captures the most important distinctions that researchers typically make in transcribing child language data.

13.1.2. CLAN 2.0

Ideally, changes to the transcription system should be few and far between. Changes to the CLAN programs also need to preserve the core functions that users have learned to rely on. However, it is easy to add new programs and new options without interfering with use of the old programs and old options. In fact, one of the nicest aspects of computerized transcription is that it lays the groundwork for a potentially infinite development of data analysis programs without requiring major changes in the underlying transcription scheme. For central CLAN programs such as FREQ, GEM, and KWAL, CLAN 2.0 only repairs a few bugs and adds a few features here and there. However, the new version of CLAN has greatly expanded analytic capabilities in three major areas: data display, data coding, and morphosyntactic analysis. The major new programs in these areas are LINES, COLUMNS, CED, MOR, DSS, and COOCCUR. Let us take a look at each of these new programs.

13.1.2.1. Data Display – COLUMNS, SLIDE, and LINES.

There is much 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 parent's 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 older CLAN programs base their data display modification capabilities on the notion of "limiting." By combined use of the **+t**, **+s**, and **+z** options, programs such as KWAL, COMBO, and FREQ can either include or exclude particular speakers, dependent tiers, headers, regions, or lines matching particular search strings. The files derived from these analyses can either be simply collections of lines or legal CHAT files with certain material excluded. These new files can then either be subjected to further analysis or simply stored as alternative versions of the main data set. The chapters in this book have provided dozens of examples of uses of limiting in the CLAN programs.

Our newer programs attempt to provide more extreme modifications of the basic CHAT format. One fairly minimal modification of the standard CHAT format is achieved by the LINES program, which inserts line numbers in front of each main tier line. When researchers are working on the computer with CLAN, it is easy enough to find line numbers by editor search commands. However, sometimes larger laboratories find it necessary to freeze a transcript into a particular hardcopy form for comparison between researchers and for temporary annotation. For this type of work with hard copy away from the computer, it is often important to have line numbers actually printed on the transcript. Unlike files filtered through COMBO or KWAL, files with line numbers inserted are no longer legal CHAT files and cannot be used with the CLAN programs. For that reason, users should be careful never to do coding or further transcription on files with line numbers added.

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).

13.1.2.2. CHECK

The CHECK program was available in CLAN 1.0, but it has been extensively revised in CLAN 2.0. CHECK now does a better job of finding errors in tier identifiers and delimiters. In the older version, it was possible to repress overwhelming numbers of CHECK errors by using the **-t%** option. This allowed you to clean up the main line before beginning work on the dependent tiers. However, it is often the case that the main line itself may have the largest number of CHECK problems. To help with this, we have added a **+d1** option, which reduces the output given by CHECK to only one of each error type. For example, if you have been using WordPerfect and have not yet converted spaces to tabs after the speaker identifications, you can use this option and you will only receive one complaint about missing tabs, rather than hundreds. When using the **+d1** option,

it is helpful to know that there are 46 different error messages produced by CHECK. What the **+d1** option guarantees is that you will only get one complaint for each of these types of errors: missing line beginnings, missing tabs, missing colons, missing @Begin, missing @End, missing @Participants line, nonstandard participant roles, missing roles, incorrect tier names, duplicate speaker declarations, missing speaker identifications, delimiters in words, unmatched paired delimiters, missing main tiers, undeclared codes, illegal date entries, illegal time entries, multiple utterances per line, undeclared prefixes, undeclared suffixes, duplicate coding tiers, missing terminators, extra terminators, and incorrect pairings of @Bg and @Eg markers.

Ideally, CHECK only needs to rely on the standard **depfile.** CHECK uses the codes in the **depfile** as its guide to understanding which CHAT codes should be permitted on which tiers. The **depfile** we distribute is, in effect, a summary of the CHAT system given in the manual. Sometimes users have good reasons for making exceptions to CHAT conventions. In order to override the definitions given in the **depfile** without having to tinker with that file, we have added the capacity to create a **ØØdepadd** file. This file then also provides an overt record of additions or modifications to CHAT required for particular corpora. For example, if you need to allow for equals signs on the @Comment line and for words with suffixes on the @Bgd line, you could create a **ØØdepadd** file with these two lines:

@Bgd: ***_***
@Comment: =

If the **depfile** has a code that is too permissive, such as $*, you will want to remove this before entering the more specific codes in your **ØØdepadd** file. In general, it is still best to focus on using CHECK early in the process of transcription, before you begin to accumulate errors. Whenever possible, it is best to use only the standard **depfile**, but sometimes there will be reasons for extending CHAT by using a **ØØdepadd** file.

CHECK only examines files for their compliance to the syntactic specifications of CHAT. An important second type of checking can be achieved by using FREQ to create a unified frequency count for an entire corpus. This is best done with this command:

freq +u +f *.cha

This command will produce a single file with all the words you used on the main lines of all your files. You can then go over these words to check for spelling errors and other inconsistencies. A useful clue in looking for spelling errors is to search for words with a frequency of 1. If you use FREQ with the **+o** option, you can immediately find all the words with a frequency of 1 together at the end of the printout. Once your preliminary cleanup is done, you may want to repeat the same analysis using the **+t%** and **–t*** options, so you can check for errors in the codes on the dependent tiers. Alternatively, you can provide CHECK with a complete listing of your codes by creating a **ØØdepadd** file.

13.1.2.3. CED – 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 truly 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.

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 WordPerfect and EMACS keystroke equivalents are available. If you prefer some other set of keystrokes, the commands can be re-bound.

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 the following:

```
$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.

CED has been successfully used by researchers at Harvard, CMU, and elsewhere to enter codes for speech acts and for narrative structures into CHAT files. Complete documentation of all of the editor commands for CED can be obtained from CMU and will be included in the next edition of the CHILDES manual.

13.1.2.4. MOR – 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 & Leinbach, 1991; MacWhinney, Leinbach, Taraban, & McDonald, 1989; Marcus, Ullman, Pinker, Hollander, Rosen, & Xu, 1991; Pinker & Prince, 1988; Plunkett & Sinha, 1992) has focused attention on early uses and overregularizations of the regular and irregular past tense markings in English. The testing of hypotheses about parameter-setting within G-B theory (Hyams, 1986; Wexler, 1986) often depends upon a careful study of pronominal markings, reflexives, and wh-words. Although some of these phenomena can be detected by simple searches for words like "what" or "he," most of them require a more complete characterization in terms of a full part-of-speech coding for entire corpora. This coding could be achieved by hand application of the codes specified in chapter 14 of the CHILDES Manual. However, hand-coding of the entire CHILDES database would require perhaps 20 years of work and would be extremely error-prone and non-correctable. 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 3 years, we have worked on the construction of an automatic coding program for CHAT files. This program, called MOR, was first developed in LISP by Roland Hausser (1990), modified for C by Carolyn Ellis, and then completely rewritten by Mitzi Morris with assistance from Leonid Spektor. 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 that can be modified by the user. In the remarks that follow, I first focus on ways in which a user can apply the 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 easily run it on any CHAT file, for you to get a well-formed **%mor** line, you often need to engage in *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 (a) lexicon building and (b) ambiguity resolution.

Files Used by MOR

Before I examine ways of dealing with lexicon building and ambiguity resolution, let us take a quick look at the files that support a MOR analysis. For MOR to run successfully, there must be four files present in either the library directory or the current working directory. Although you do not need to have a detailed understanding of the functioning of these files, it will help you to have a

view of the shape of these basic building blocks. The default names for these files are **eng.ar**, **eng.cr**, **eng.lex**, and **eng.clo**. These four files and a fifth optional file contain the following information:

1. **Allomorphic rules.** The eng.ar file lists the ways in which morphemes vary in shape. The rules that describe these variations are called "**arules**."

2. **Concatenation rules.** The eng.cr file lists the ways in which morphemes can combine. The rules that describe allowable concatenations are called "**crules**."

3. **Closed class items.** The eng.clo file contains the closed class words and suffixes of English. Because this group forms such a tight closed set, the user will seldom have to modify this file.

4. **Open class items.** The default name of the open class lexicon for English is eng.lex. This file is what we call the "disk lexicon." Words in the disk lexicon are listed in their canonical form, along with category information. The version of eng.lex we distribute contains about 2,000 words. We are also distributing a larger lexicon called big.lex with 24,000 words. However, only machines with very large memories, such as UNIX workstations or high-end Macintoshes, will be able to run this bigger lexicon. We are currently developing tools that will also allow MS-DOS programs access to larger amounts of memory.

5. **Disambiguation rules**. We also distribute a set of local context disambiguations rules or "**drules**" in the eng.dr file. These rules can resolve a large proportion of the part-of-speech ambiguity in English on the basis of local co-occurrence information. In order to use the drules file, you need to use the **+b** option when you use the MOR command.

MOR uses the eng.ar, eng.lex, and eng.clo files to produce a *run-time lexicon* that is significantly more complete than the eng.lex alone. When analyzing input files, MOR uses the run-time lexicon together with the eng.cr file. As a user, you do not need to concern yourself with the actual shape of the run-time lexicon. And you will usually not have to touch the arules, crules, or drules. Your main concern will be with the process of adding or removing entries from the main open class lexicon

file. If all of the words in your files can be located in the eng.lex file, running of MOR is totally trivial. You simply run:

mor filename

But matters are seldom this simple, because most files will have many words that are not found in eng.lex and you will need to refine the eng.lex file until all missing words are inserted. Therefore, the main task involved for most users of MOR is the building of the lexicon file.

Lexicon Building – Finding Missing Words

In order to see 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 your *.mor output files, you will probably want to correct this problem by running MOR in the interactive update mode. If you know from the outset that your file includes many words that will not be found in eng.lex, you can directly begin your analysis by running this FREQ command:

freq +d1 +u +k *.cha > output.frq

This produces a simple list of all the words in your transcript in a form useful for interactive MOR lexicon building. The **+d1** option outputs words without frequencies. The **+k** option is needed to distinguish between "Bill" and "bill." The redirection arrow sends the output to a file we have called "output.frq."

Next you can run MOR again using the **+s** option with a filename added, as in this example:

mor +x1 output.frq

MOR will use to use eng.lex to attempt to analyze each word in the output.frq file. If it cannot analyze the word, it will enter it in a output file of lexical entry templates with the name output.ulx. Then you need to look at the words in the output.ulx file, using an editor. Some may be misspellings and will have to be corrected in the original file. Others will be new words for which you will have to enter a part-of-speech. When you are finished, you should rename the output.ulx file to output.lex. Then you can run MOR again in this form:

mor +loutput *.cha

If all has gone smoothly, this time MOR will be able to enter a part-of-speech characterization for every word in the transcript.

The Structure of eng.lex

Users of MOR may want to understand the way in which entries in the disk lexicon (eng.lex) are structured. The disk lexicon contains truly irregular forms of a word, as well as citation forms. For example, the verb "go" is stored in eng.lex, along with the past tense "went", since this form is a suppletive form, and is not subject to regular rules. The disk lexicon contains any number of lexical entries, stored at most one entry per line. A lexical entry may be broken across several lines by placing the continuation character backslash, (\), at the end of the line. The lexicon may be annotated with comments, which will not be processed. A comment begins with the percent sign, %, and ends with a new line.

A lexical entry consists of the surface form of the word, followed by category information about the word expressed as a set of feature-value pairs. Each feature-value pair is enclosed in square brackets, and the full set of feature-value pairs is enclosed in curly braces. All entries must contain a feature-value pair that identifies the syntactic category to which the word belongs, consisting of the feature "scat" with an appropriate value. Words that belong to several categories will be followed by several sets of feature structures, each separated by a backslash. Optionally following the category information is information about the stem. If the surface form of the word is not the citation form of the word, then the citation form, surrounded by quotes, should follow the category information. If the word contains fused morphemes, these should be given as well, using the & symbol as the morpheme separator. The following are examples of lexical entries:

can **{[scat v:aux]} **
{[scat n}

a **{[scat det]}**

an **{[scat det]}** **"a"**

go **{[scat v] [ir +]}**

went **{[scat v] [tense past]} "go&PAST"**

When adding new entries to eng.lex it is usually sufficient to enter the citation form of the word, along with the syntactic category information.

Ambiguity Resolution

MOR automatically generates a **%mor** tier of the type described in chapter 14 of MacWhinney (1991). As stipulated there, 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.clo file maintain these ambiguities. However, open class words in the eng.lex file are only coded in their most common part-of-speech form. If you use the **+b** option, some of these alternatives will be pruned, but some will still remain. The problem of noun-verb ambiguity will eventually be addressed

through use of the PARS program, which is currently being developed. A primitive disambiguation facility is currently available in the disambiguation rules in the eng.dr file. Over the next years, we intend to continually improve the abilities of PARS to eliminate ambiguities.

Those ambiguities that remain in a MOR transcript after the drules and the PARS program have 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 arules and crules. 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: (a) a list of the stems of the language with their parts-of-speech, (b) a set of arules for allomorphic variations in spelling, and (c) 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. 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.

13.1.2.5. DSS – Developmental Sentence Score

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. Some of these analyses were discussed in the CHILDES manual and others were mentioned in other chapters in this book. One particularly elaborate system for morphosyntactic analysis that has been widely used in research on language

disorders is the Developmental Sentence Score (DSS; Lee, 1974). Lee's DSS tracks the uses of eight major morphosyntactic types: indefinite pronouns or noun modifiers, personal pronouns, main verbs, secondary verbs, negatives, conjunctions, interrogative reversal in questions, and wh-questions. The CLAN DSS program can automatically compute DSS scores for individual samples. In computing these scores, specific lexical markers and syntactic patterns for each of the eight grammatical categories are broken into developmental stages, and developmental scores are assigned to usages of structures or items at each level. Earlier-occurring constructions receive fewer points, whereas later-occurring constructions receive more points. An additional sentence point is given to each sentence if it meets all adult grammatical standards.

DSS scores are based upon analysis of a corpus of 50 sentences. The DSS program is designed to extract a set of 50 sentences from a language sample using Lee's six inclusion criteria:

1. The corpus should contain 50 complete sentences.
2. The speech sample must be a block of consecutive sentences.
3. All sentences in the language sample must be different.
4. Unintelligible sentences should be excluded from the corpus.
5. Echoed sentences should be excluded from the corpus.
6. Incomplete sentences should be excluded.

DSS can rely on CHAT codes such as xxx for unintelligible sentences and +... for incomplete sentences to compute the 50-sentence corpus automatically. However, there is one additional criterion in the original DSS framework that the DSS program cannot automatically compute. This is the criterion that DSS analysis should be used if and only if at least 50% of the utterances are complete sentences as defined by Lee. If fewer than 50% of the sentences are complete sentences, then the Developmental Sentence Type analysis (DST) is appropriate instead. A warning message will be included on the DSS printout if this criterion has not been met.

Once all 50 sentences have been assigned sentence points, the DSS program automatically generates a table. Each sentence is displayed in the left-hand column of the table with the corresponding point values. The Developmental Sentence Score is calculated by dividing the sum of the total values for each sentence by the number of sentences in the analysis. Here is a sample output file:

Sentence	IP	PP	PV	SV	NG	CNJ	IR	WHQ	S	TOT
I like this.	1	1	1						1	4
I like that.	1	1	1						1	4
I want hot dog.		1	1						0	2
I like it.	1	1	1						1	4
what this say.	1		–					–	0	3

Developmental Sentence Score: 4.2

The table has been specifically designed for users to determine "at a glance" areas of strength and weakness for the individual child for these eight grammatical categories. The low point values for both the indefinite and personal pronoun (IP, PP) categories in the table indicate that this child used earlier developing forms exclusively. In addition, the attempt markers for the primary verb (PV) and interrogative reversal (IR) categories suggest possible difficulties in question formulation.

13.1.2.6. COOCCUR – Co-occurrence Analysis

The COOCCUR program produces a complete list and count of pairs, triplets, or longer strings of words. The analysis of syntactic clusters produced by COOCCUR could be a major building block in an empirical analysis of the child's construction of syntax from lower-order co-occurrences (Braine, 1963, 1976, 1987; Elman, 1991; Ingram, 1972, 1975; MacWhinney, 1974, 1975; Maratsos & Chalkley, 1980). By default, the cluster length is two words, but you can reset this value just by inserting any integer up to 20 immediately after the **+n** option. The second word of the initial cluster will become the first word of the following cluster, and so on.

cooccur +t*MOT +n3 +f sample.cha

The **+t*MOT** option tells the program to select only the *MOT main speaker tiers. The header and dependent code tiers are excluded by default. The **+n3** option tells

the program to combine three words into a word cluster. The program will then go through all of the mother's (*MOT) main speaker tiers in the **sample.cha** file, three words at a time. When COOCCUR reaches the end of an utterance, it marks the end of a cluster, so that no clusters are broken across speakers or across utterances.

Co-ocurrences of codes on the **%mor** line can be searched using commands such as this example:

cooccur +t%mor –t* +s*def sample2.cha

This command would allow one to track the types of morphosyntactic constructions in which particular categories co-occur with definite articles.

13.1.2.7. Modifications to CLAN 1.0 Facilities

In addition to the new programs included in CLAN 2.0, we have made several modifications to the older programs and documentation. One major change involved the dropping of the manuals facility, because it proved difficult to maintain this facility properly across all the computer platforms on which CLAN is being used. A number of program-specific options were added, including the **+d3** and **+d4** options for output formatting in FREQ, the **+o** option for separate control of limiting for the output in KWAL and COMBO, the **+a** option for alphabetization in KWAL, the **+b** option for tier naming in PHONFREQ, and the **+b** option for delimiter control in MLU. Several problems relating to the automatic exclusion of strings such as &*, *-Ø*, or *+Ø* in MLU and FREQ were fixed, and the various MLT documentation errors were fixed. Finally, the BIBFIND, MODREP, and PHONFREQ programs were totally rewritten to work more efficiently and more in accord with the published documentation.

13.1.3. Development of the Database

One of the major goals of the CHILDES Project involved the reformatting of the many different corpora in the database into the single lingua franca of CHAT. Using a wide variety of computational techniques to perform the translation, and relying upon the CHECK program as a filter for syntactic accuracy, we completed this reformatting in August of 1992. There are six corpora that have not been shifted to CHAT format. These are the corpora from: Fawcett; the ESF project; Stern and Stern; Sulzby; Hayes; MacWhinney and Bates. The ESF database and the

Fawcett data are no longer included in the database because reformatting of these data into CHAT would involve too many technical difficulties. The data from the diary notebooks of the Sterns will not be reformatted, because it is of greater historical interest in its present form. The transcripts from the other three projects will eventually be either reformatted or dropped from the database.

In most cases, reformatting required us to take data that was not yet in CHAT format and translate the project-specific codes and formats into CHAT. The fact that these transcripts now all pass 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. At this point, we cannot yet guarantee that the files are consistently coded on the level of individual words. However, this level of consistency checking remains a goal for our future work with the database.

13.1.3.1. New Corpora

New corpora that have been added to the database since the publication of the Manual include:

1. **Deuchar**. A case study of bilingual acquisition of Spanish and English by Margaret Deuchar of the University of Cambridge (Deuchar & Clark, 1992).

2. **Hayashi**. A case study of bilingual acquisition of Danish and Japanese, along with some English, by Mariko Hayashi of Århus University.

3. **Peters/Wilson**: A case study of the acquisition of English by a blind child donated by Ann Peters of the University of Hawaii and Bob Wilson of the Foreign Service Institute (Peters, 1987; Wilson & Peters, 1988).

4. **Van Kleeck**: Cross-sectional data from 37 three-year-olds collected in a laboratory setting by Anne Van Kleeck of the University of Texas, Austin.

5. **Rondal**. A case study of French language acquisition by Jean Rondal of the University of Liége (Rondal, 1985).

A variety of additional corpora are currently in preparation in areas such as autism, specific language impairment, focal lesions in childhood, second language acquisition, and the acquisition of Spanish and Swedish. A new edition of the electronic version of the CHILDES/BIB database (Higginson & MacWhinney, 1990) was released in October 1992. This new version includes hundreds of new entries and corrects various errors in the first edition.

13.1.3.2. FTP Access to the Database and Programs

All of the CHILDES materials can be obtained without charge by using anonymous FTP to poppy.psy.cmu.edu. InterNet connections that can reach poppy.psy.cmu.edu are now widely available at universities both in the United States and abroad. The procedure for transferring files varies depending 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 the basic rules for 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.

6. Use the "get" command to pull files onto your machine.

7. When you are finished, type "bye" or "quit" to close the connection.

Once the files are on your local machine, you must untar them. 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 UNIX or DOS tar command you need to issue is something like:

tar –xvf eng.tar

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

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 any BinHex utility to decode them. When transferring CLAN, also remember to transfer the text files into /clan/lib.

DOS users can also use FTP to retrieve the most recent version of CLAN from poppy.psy.cmu.edu. Get all the files in the /msdos and /lib directories under the /clan directory. Try to maintain the directory structure. The files in the /msdos directory are executable programs that should run immediately on your machine, once you have set the path. Note that the TAR.EXE program will be included along with the other CLAN programs.

13.1.3.3. CD-ROM Access to the Database and Programs

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, 1992) a CD-ROM in High Sierra format, which can be read by Macintosh, UNIX, and MS-DOS machines that have a CD-ROM reader. The single 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 $500 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. To request a subscription to the bulletin board, send your request to info-childes-request@andrew.cmu.edu.

13.2. The Immediate Future

Although we have completed a great deal of work in the past six years, there is still an enormous amount to be done. Our plans for the immediate future focus on these goals:

1. **CHAT**. We hope to keep CHAT relatively stable and free of changes. Whatever changes we anticipate making will probably be mostly in coding systems on dependent tiers, rather than in codings on the main line.

2. **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. Over the next few years, we expect the database to grow beyond the current focus on first language acquisition by normal 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. We have already begun 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).

3. **CLAN**. The most exciting prospects for future developments in CHILDES lie in the area of new developments in CLAN. We expect to construct a large variety of new CLAN programs to automate the process of phonological, morphosyntactic, lexical, and discourse analysis. We also plan several new forms of data display. Each of these new initiatives will be discussed separately.

13.2.1. Developmental and Clinical Profiling

Underlying each of these technical initiatives are broader long-term theoretical goals. On the one hand, researchers want to use the CHILDES system to illuminate crosslinguistic regularities in the ways that children learn languages. On the other hand, they want to evaluate individual differences in language development. The search for both differences and commonalities in language development is the fundamental engine that drives developments in our understanding of language acquisition. We can use measures computed from the CHILDES database to fuel and oil this engine. The CHILDES Project cannot supply theoretical interpretations, but it can supply an ongoing stream of data that will motivate the development of these theoretical interpretations.

The CHILDES Project can develop a wide variety of measures that can be subjected to a process of indicator validation. In particular, we need to validate the individual indicators that have been used in previous measurement instruments along three dimensions:

1. **Developmental validity**. Each indicator must accurately predict developmental stage, at least within a given developmental range. For example, during the first stages of language learning, MLU is a good predictor of developmental stage. However, later on, its developmental validity weakens. All indicators have floors and ceilings. For an indicator to display developmental validity, it only needs to correlate with developmental stage for part of the growth curve. MLU is one of the few measures for which developmental validity has been studied (Miller, 1981).

2. **Clinical validity**. Each indicator must also prove useful in allowing us to classify children as belonging to particular groups or populations. For example, we might expect that indicators focusing on the use of personal pronouns or topic maintenance structures would help us identify a child as autistic. Similarly, retracings, word repetitions, and hesitations would help us to classify a child as having problems with sentence planning (Wijnen, 1990). In some cases, we already have reason to classify a child in a particular clinical group, and we are then interested in the correlation between particular measures and our assignment of children to that group. In other cases, we are using our prior experience with the measure as the basis of the classification.

3. **Processing validity**. Each indicator must be conceptualized in a way that allows us to see how it relates to particular cognitive or social processes. This is a form of content validity.

In order for an indicator to be judged as "useful" it needs to display some combination of these three types of validity along with easy computability. Some indicators may have clinical validity without developmental validity. For example, "initial segment repetition" may be a clear characteristic of children with developmental disfluency, but one that undergoes few changes over time. Because it undergoes few changes over time, it has good clinical validity, but low developmental validity. Although indicators without developmental validity will not tell us much about development, they can be quite useful in making clinical assessments. It is also possible, although somewhat unlikely, that some measures will have developmental validity but no clinical validity.

We want to make sure that the indicators we will construct should be computable from CHAT transcripts. What this means is that there must be programs that can take CHAT files as input and produce tabulations for each indicator as output. There is no one single program that can compute every indicator. Instead, we will need to rely on a variety of programs, including FREQ, STATFREQ, MLU, COMBO, CHIP, MOR, PARS, and CHAINS. These various analyses can be grouped together into "scripts" or "batch files" to further automate the examination of large sets of corpora.

There are three reasons for requiring that all indicators be computable from CHAT files. The first reason is that, when faced with the prospect of tracking hundreds of indicators across as an enormous quantity of transcript data, the use of indicators that require hand calculation becomes simply impossible. The second reason is that the computation of indicators by machine is a fully **reliable** process. Once a program has been given a definition of an indicator, it will always compute this indicator in the same way. Of course, if the original transcript has errors, these

will not be corrected. However, errors in transcription would also adversely affect the computation of indicators by human calculation. Moreover, it is often possible to pick up transcription errors by computational filters built into the CLAN programs. Finally, by requiring that each indicator be computable, we are also guaranteeing that the calculation of the indicator be completely operationalized. We then know exactly how the indicator was computed and exactly how it is defined.

Having determined the developmental and clinical validity of our indicators, and having separated out those with low validity, we now have a set of indicators that will be useful for studying developmental profiles in both normally-developing and clinical populations. In the normally-developing child, we expect to find that, for a given part of the developmental curve, many measures will be highly intercorrelated. For example, we would not be surprised to find that increases in the vocabulary for names of plants and animals are highly correlated with increases in the vocabulary for mental states. This **developmental cohesion** in normally-developing children can be explained in a variety of ways. The most basic approach views all developments as driven by a single underlying motor. We can call this motor "experience" or we can call it "maturation." In either case, we are simply recognizing the obvious fact that children gain competence as they get older. Even in normally-developing children, we expect to see points at which developmental cohesion between measures begins to break down. Sometimes this is due to ceiling or floor effects. The **unbraiding** (Tager-Flusberg, 1988) of indicators that occurs in the normally-developing child must be assessed and understood in detail before we can properly understand the more extreme unbraiding that occurs in clinical populations.

Having completed the analysis of developmental cohesion with normally-developing children, we can then begin to characterize the additional patterns of unbraiding that occur in clinical populations. We take our map for normally-developing children and impose upon it our map of the same indicators in a particular clinical population. To do this, we will try to match across age so that the overall fit of the clinical population to the normal map is maximized. Given this, we can then focus on those indicators that are markedly above the normal level and those that are markedly below the normal level. We can speak of these mismatches to the normal profile as **asynchronies**. For example, we might expect to find the control of complex syntax and the more complex mental state verbs to be particularly low in children with Down syndrome across a variety of

developmental stages. We might also expect to see the use of particular speech acts depressed in autistic children across a variety of time periods. The exact characterization of the developmental course of these **asynchronies** is one of the major goals that will surely carry us well into the next century.

We cannot realistically assume that the goal of reconstituting the foundations of language assessment will be achieved in the immediate future. However, what we can hope to construct in the immediate future are the computational tools upon which this analysis can be based. Doing this will require new tools for the analysis of lexicon, morphosyntax, phonology, and discourse. Let us now look at each of these groups of new tools.

13.2.2. The Lexical Initiative

What are the most frequent words used by English-speaking children? The most useful account of actual spoken English usage from children is provided by Hall, Nagy, and Linn (1984). That tabulation, while extremely useful for certain comparisons, is based on samples taken from a fairly narrow age range. The other available count is Rinsland (1945). That count was based on a small set of data and is now badly out of date. Construction of a fuller frequency count for English-speaking children would seem to be a simple and important goal.

At first blush, it would seem that one could easily answer this question using the FREQ program and the CHILDES database. It is an easy enough matter to run FREQ on collections of files using the **+u** option. Using FREQ, one could simply compute the numbers of occurrences for every word in the database. The output would look much like the frequency analysis of the Brown University Corpus developed by Francis and Kucera (1982). However, there are conceptual problems involved in the construction of such a simple summary frequency count. Some of these problems were directly confronted by Hall, Nagy, and Linn. 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 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.

What we plan to do to address this problem is to apply the computational techniques that were developed in building the CHILDES/BIB retrieval system to the lexical analysis of the whole CHILDES database. Each file will be provided with a new header giving a detailed set of codes for the key participants at various age levels. In one large computation-intensive job, we will extract 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. Once these pointers are computed, no changes can be made in the database. Storage of the data on CD-ROM in this form would be ideal, since CD-ROM files cannot be altered. Once the pointers from the master wordlist to the individual occurrences of words are computed, the user can then 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 2-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.

A second simple use of the LEX facility will permit automatic analyses of words in particular lexical fields. For example, using this lexical database, we will be able to examine the development of selected lexical fields in the style of the PRISM analysis of Crystal (1982). The goal here is to find particular lexical domains that serve to characterize or classify children by age and clinical subgroup.

Likely candidates for intensive examination include: mental verbs, morality words, temporal adverbs, subordinating conjunctions, and complex verbs.

A third tool that can be developed through use of the LEX facility is the Lexical Rarity Index of LRI. Currently, the major measure 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 data set. The more that 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.

13.2.3. The Morphosyntactic Initiative

The completion of the coding of the **%mor** line for the database will allow us to construct indicators based on: (a) morpheme inventories, (b) semantic relations, (c) phrasal and utterance complexity, and (d) specific syntactic structures.

13.2.3.1. Morpheme Inventories

The study of the acquisition of particular 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 (1973). 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. Other markers tracked in LARSP, ASS, IPSYN, 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. All of these markers are given discrete categories on the **%mor** line, and it will be easy to use KWAL to search out their occurrence, FREQ to tabulate their frequencies, and COOCCUR and COMBO to study patterns in which they co-occur.

In addition to these basic grammatical markers, researchers have been interested in tracking 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 (1984) argued that tense markings and temporal adverbs are not controlled until the child first masters the relevant conceptual categories.

13.2.3.2. Semantic Relations

Many basic semantic relations that have been discussed extensively in the literature (Bloom, 1975; Lahey, 1988; Leonard, 1976; Retherford, Schwartz, & Chapman, 1981) can be tracked by simply studying a few closed-class lexical items. In particular, we 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, etc.
Adverbial	–ly
Quantifier	one, two, more, some
Recipient	to
Beneficiary	for
Comitative	with
Instrument	with, by

13.2.3.3. Beyond MLU

Beyond the study of individual morphemes, we can construct indicators based on the emergence of syntactic structures. The simplest form of syntactic analysis looks only at the development of sentence length. For example, Templin (1957) used a measure based simply on mean utterance length in words. Brown (1973) refined the analysis of mean length of utterance by treating each morpheme as a separate item. A related measure that can also be computed is MLP or mean length of phrase, which is essentially the measure studied by Loban (1976). The various

types of mean length of utterance measures can be automatically computed by the CLAN MLU program. There is a wide variety of options that can be used in the computation of MLU. For example, one may exclude all imitative utterances, all hesitations, all word repetitions, and so on. The MLU program allows the user full control over exactly how these computations should be performed. A good discussion of procedures for computing MLU can be found in Miller (1981), as well as in chapters by Pan (this volume) and Rollins (this volume). Miller and Chapman (1981) report a strong correlation between MLU and age in a sample of 123 children. However, in the higher ranges of the MLU indicator, this correlation begins to decrease.

Miller (1981) emphasized the utility of studying sentence frequency distributions. In these distributions, one simply computes the number of utterances with one word, two words, three words, and so on. This simple computation can be performed by the CLAN MAXWD program.

13.2.3.4. Syntactic Structures

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 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.

13.2.3.5. Scales

Although it would be possible to track each of these morphological and syntactic structures as separate indicators of language development, there is good reason to expect to find an enormous overlap between these separate indicators in terms of their prediction of developmental stage and clinical subgroup. Given this, it is likely that we will eventually want to merge groups of these structures into composite indicators. However, the exact shape of these composites will have to be inferred statistically.

An alternative way of merging information is to form a scale. In several parts of the grammar, it appears that a smaller set of indicators can possibly be grouped together into a larger scale. For example, DSS, ASS, IPSYN, and LARSP all claim that children move through these stages in the elaboration of the verb phrase:

1. uninflected verb
2. copula or contracted copula
3. is + verb + ing
4. addition of –s or –ed affixes to the verb
5. control of additional forms of the copula
6. modal + verb
7. do + verb
8. past tense modals
9. "get" passive
10. modal + cop + verb + ing
11. have + verb + en
12. modal + have + verb + en

Do these developments really scale in this way? Can one safely say that a child who is using the "get" passive has already controlled the addition of the modal, *do*, and the other suffixes lower in the sequence? If this is true, then we can say that this is a real scale. If there are many reversals of the predicted order, than we will not treat this as a scale.

If we can indeed construct a variety of scales of this type, we can then use the child's level on each scale as a single indicator. For example, a child who controls the "get" passive and nothing higher on the scale would have a score of 9 on the verb phrase expansion scale. Similar scales will be tested for interrogative words, conjunctions, negatives, secondary verbs, and pronouns.

13.2.3.6. Crosslinguistic Applications

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?

13.2.4. The Phonological Initiative

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. 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) 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. Although inclusion of a complete **%pho** line is a powerful tool, even this form of two-tier transcription tends to miss 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 fast-forward 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 transcription. The system we have developed at Carnegie Mellon, called Talking Transcripts, uses digitized speech, mass storage technology, and the Macintosh operating system to forge these direct links. For each transcript, these steps must be followed:

1. Using the SoundDesigner program and the AudioMedia II digitizer board from Digidesign, a 30-minute audio segment is recorded to computer disk. The recording is done at 22KHz with 16-bit digitization. The audio quality at this sampling rate is excellent. Once the recording is started the tape recorder and computer can be left unattended. Although the process of digitizing a complete set of tapes will take many hours, there is no need to continually monitor the activity.

2. The resulting size of the digitized file for the 30-minute segment is about 80 megabytes. Because of the large size of these files, we write them directly to optical erasable cartridges. Optical erasable provides limitless, fast, erasable, but stable storage. Currently, the PMO-650 drive from Pinnacle is the fastest of these drives, but standards here are changing rapidly. Prices for this technology are still fairly high, but they are continuing to drop.

3. Once a complete 30-minute segment has been digitized, we can begin to transcribe the data in CHAT within a CED window.

4. Using a special new feature in the CED interface, we can automatically insert a mark in the transcript that corresponds to the next segment of the digitized file.

5. Later, when we want to play back a particular utterance, we simply click on the mark in the transcript in the CED window and the program directly plays the correct segment in the SoundDesigner file.

Although this process requires some additional time setting up the basic digitization and creating the playlist, this investment pays for itself in facilitating 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 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. It is difficult to describe verbally the immediacy of this link, but 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 tends to make the transcriber more confident regarding the process of creating full phonological analyses that can be verified for reliability later.

The immediate availability of digitized sound has strong positive consequences for the process of phonological transcription. It will now be possible 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 of 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. Non-occurrences 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, by 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 & Bauer, 1991) will be computed by a new CLAN program called PHOP. 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 refers to classes of sounds rather than to 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, non-developmental errors will be identified and 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 play back 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.

13.2.5. The Discourse Initiative

Many researchers want to track the ways in which discourse influences ways of expressing topic, anaphora, tense, mood, narrative voice, ellipsis, embedding, and word order (Halliday & Hasan, 1976; MacWhinney, 1985). They want 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.

The complex process of discourse analysis benefits from the use of virtually every aspect of the CLAN programs. The display programs – COLUMNS, LINES, and SLIDE – are designed to facilitate the viewing of turns and overlaps. The CED editor allows discourse analysts to enrich transcripts with a variety of codes in a fully hierarchical scheme. Sequential analysis programs such as CHAINS, CHIP, KEYMAP, and MLT can then analyze sequences, pairs, and links between these codes. Specific types of interactions can be marked with GEM for further analysis.

The time-consuming nature of speech act coding has led us to place particular emphasis on the development and refinement of the CED program (Bodin & Snow, this volume). However, as richly coded transcripts become more available, we will begin to place an emphasis on extensions to programs such as CHAINS or KEYMAP.

The initial uses of the CHIP program focused on superficial lexical matches between consecutive utterances. With the completion of the morphological coding

of corpora such as the Brown corpus, it is now possible to use CHIP to study the repetition and expansion of utterances not just in terms of superficial lexical match, but also in terms of underlying grammatical category.

13.3. The Distant Horizon

As we gaze across the immediate horizon into the distant future, we peer into a world where computational power continues to grow and the structure of the database becomes tighter and richer. We can assume that, within the next 20 years, personal computers will have access to virtually limitless amounts of digitized speech and video. These data will be accessible through fiber-optic networks, erasable CD-ROM disks, and powerful memory chips. Formats for data compression and transfer will be standardized across computer systems, further facilitating the transfer of multimedia data.

As these resources become increasingly available, the CHILDES database will shift from its current concentration on ASCII transcripts to a focus on transcripts accompanied by digitized audio and video. Links between events in the audio and video records will be tied to an increasingly rich set of links in the transcript. These "hot" 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.

Although the core distinctions of CHAT will continue to be important, the underlying computer representation of CHAT may shift to a form more like that of the Standard Generalized Markup Language (SGML; Sperberg-McQueen & Burnard, 1990) or of some successor to SGML. This underlying form will not be displayed to the user; rather, it will be stored in the CHILDES database and transformed when

users interact with the database. Movement to this more structured underlying representation will be facilitated by the development of transducers from CHAT to SGML and from SGML to CHAT. Once a full abstract representation of the database has been constructed, additional output filters can be constructed to deal with national orthographies, full IPA markings, and symbols for prosodic contours.

The construction of this new multimedia transcript world will 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 will 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 in total precision curves for overregularizations, item frequencies, and error types.

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 Institutes 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). For the recent work reported here, thanks should go to Leonid Spektor for construction of the BIBFIND, CED, COLUMNS, COOCCUR, DATES, LINES, and SLIDE programs and for further elaborations of earlier programs, including CHECK. Mitzi Morris developed 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. Helmut Feldweg also created a prototype version of the COLUMNS program, helped construct the German UNIBET, and supervised the solidification of the transcript database for German. Steven Gillis corrected errors in the Dutch UNIBET table and Christian Champaud corrected errors in the French UNIBET table. 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. Thanks also to Julia Evans, Jeff Sokolov, and Catherine Snow for their comments on this chapter.

References

Antinucci, F., & Miller, R. (1976). How children talk about what happened. *Journal of Child Language, 3*, 167-189.

Bauer, H. (1988). The ethologic model of phonetic development: I. Phonetic contrast estimators. *Clinical Linguistics and Phonetics, 2*, 347-380.

Bloom, L. (1975). Language development. In F. Horowitz (Ed.), *Review of child development research*. Chicago: University of Chicago Press.

Braine, M. D. S. (1963). The ontogeny of English structure: The first phase. *Language, 39*, 1-13.

Braine, M. D. S. (1976). Children's first word combinations. *Monographs of the Society for Research in Child Development, 41* (Whole No. 1).

Braine, M. D. S. (1987). What is learned in acquiring word classes: A step toward an acquisition theory. In B. MacWhinney (Ed.), *Mechanisms of language acquisition*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Brown, R. (1973). *A first language: The early stages*. Cambridge, MA: Harvard.

Cromer, R. (1991). *Language and thought in normal and handicapped children*. Oxford: Blackwell.

Crystal, D. (1982). *Profiling linguistic disability*. London: Edward Arnold.

de Villiers, J., & de Villiers, P. (1973). A cross-sectional study of the acquisition of grammatical morphemes in child speech. *Journal of Psycholinguistic Research, 2*, 267-278.

Deuchar, M., & Clark, A. (1992). *Bilingual acquisition of the voicing contrast in word-initial stop consonants in English and Spanish* (Cognitive Science Research Report No. 213). Unpublished manuscript, University of Sussex.

Elman, J. (1991). *Incremental learning, or the importance of starting small* (TR #9101). Unpublished manuscript, University of California, San Diego.

Ervin-Tripp, S. (1979). Children's verbal turn-taking. In E. Ochs & B. Schieffelin (Eds.), *Developmental pragmatics*. New York: Academic Press.

Francis, W., & Kucera, H. (1982). *Frequency analysis of English usage: Lexicon and grammar*. Boston: Houghton Mifflin.

Hall, W. S., Nagy, W. E., & Linn, R. (1984). *Spoken words: Effects of situation and social group on oral word usage and frequency*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Halliday, M., & Hasan, R. (1976). *Cohesion in English*. London: Longman.

Hausser, R. (1990). Principles of computational morphology. *Computational Linguistics*, 47.

Higginson, R., & MacWhinney, B. (1990). *CHILDES/BIB: An annotated bibliography of child language and language disorders*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Hyams, N. (1986). *Language acquisition and the theory of parameters*. Dordrecht: D. Reidel.

Ingram, D. (1972). The development of phrase structure rules. *Language Learning, 22*, 65-77.

Ingram, D. (1975). If and when transformations are acquired by children. In D. P. Dato (Ed.), *Developmental psycholinguistics: Theory and applications*. Washington, DC: Georgetown University Press.

Lahey, M. (1988). *Language disorders and language development*. New York: Macmillan.

Lee, L. (1974). *Developmental sentence analysis*. Evanston, IL: Northwestern University Press.

Leonard, L. (1976). *Meaning in child language*. New York: Grune & Stratton.

Loban, W. (1976). *Language development*. Champaign, IL: National Council of Teachers of English.

MacWhinney, B. (1974). *How Hungarian children learn to speak*. Unpublished doctoral dissertation, University of California, Berkeley.

MacWhinney, B. (1975). Pragmatic patterns in child syntax. *Stanford Papers and Reports on Child Language Development, 10*, 153-165.

MacWhinney, B. (1985). Grammatical devices for sharing points. In R. Schiefelbusch (Ed.), *Communicative competence: Acquisition and intervention*. Baltimore: University Park Press.

MacWhinney, B. (1991). *The CHILDES project: Tools for analyzing talk*. Hillsdale, NJ: Lawrence Erlbaum Associates.

MacWhinney, B. (1992). *The CHILDES database*. Dublin, OH: Discovery Systems.

MacWhinney, B., & Leinbach, J. (1991). Implementations are not conceptualizations: Revising the verb learning model. *Cognition, 29*, 121-157.

MacWhinney, B., Leinbach, J., Taraban, R., & McDonald, J. (1989). Language learning: Cues or rules? *Journal of Memory and Language, 28*, 255-277.

Maratsos, M. & Chalkley, M. (1980). The internal language of children's syntax: The ontogenesis and representation of syntactic categories. In K. Nelson (Ed.), *Children's language (Vol. 2)*. New York: Gardner.

Marcus, G. F., Pinker, S., Ullman, M., Hollander, M., Rosen, T. J., Xu, F. (1992). Overregularization in language acquisition. *Monographs of the Society for Research in Child Development, 57:4*.

Miller, J. (1981). *Assessing language production in children: Experimental procedures*. Baltimore: University Park Press.

Miller, J., & Chapman, R. (1981). Research note: The relation between age and mean length of utterance in morphemes. *Journal of Speech and Hearing Research, 24*, 154-161.

Nelson, L., & Bauer, H. (1991). Speech and language production at age 2: Evidence for tradeoffs between linguistic and phonetic processing. *Journal of Speech and Hearing Research, 34*, 879-892.

Ochs, E. (1979). Transcription as theory. In E. Ochs & B. Schieffelin (Eds.), *Developmental pragmatics*. New York: Academic Press.

Peters, A. (1987). The role of imitation in the developing syntax of a blind child. *Text, 7*, 289-311.

Peters, A., Fahn, R., Glover, G., Harley, H., Sawyer, M., & Shimura, A. (1990). *Keeping close to the data: A two-tier computer-coding schema for the analysis of morphological development*. Unpublished manuscript, University of Hawaii, Honolulu.

Pinker, S., & Prince, A. (1988). On language and connectionism: Analysis of a Parallel Distributed Processing Model of language acquisition. *Cognition, 29*, 73-193.

Plunkett, K., & Sinha, C. (1992). Connectionism and developmental theory. *British Journal of Development Psychology, 10*, 209-254.

Retherford, K., Schwartz, B., & Chapman, R. (1981). Semantic roles and residual grammatical categories in mother and child speech: Who tunes in to whom? *Journal of Child Language, 8*, 583-608.

Rinsland, H. (1945). *A basic vocabulary of elementary school children*. New York: Macmillan.

Rondal, J. A. (1985). *Adult-child interaction and the process of language understanding*. New York: Praeger.

Shriberg, L. (1990). *Programs to examine phonetic and phonologic evaluation records*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Slobin, D. (1986). Crosslinguistic evidence for the language-making capacity. In D. Slobin (Ed.), *The crosslinguistic study of language acquisition: Volume 2. Theoretical issues*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Sperberg-McQueen, C. M., & Burnard, L. (1990). *Guidelines for the encoding and interchange of machine-readable texts*. Chicago: Association for Computational Linguistics.

Tager-Flusberg, H. (1988). On the nature of a language acquisition disorder: The example of autism. In F. S. Kessel (Ed.), *The development of language and language researchers*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Templin, M. (1957). *Certain language skills in children*. Minneapolis, MN: University of Minnesota Press.

Weist, R., Wysocka, H., Witkowska-Stadnik, K., Buczowska, E., & Konieczna, E. (1984). The defective tense hypothesis: On the emergence of tense and aspect in child Polish. *Journal of Child Language, 11*, 347-374.

Wexler, K. (1986). Parameter-setting in language acquisition. In B. MacWhinney (Ed.), *Mechanisms of language acquisition*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Wijnen, F. (1990). The development of sentence planning. *Journal of Child Language, 17*, 550-562.

Wilson, B., & Peters, A. M. (1988). What are you cookin' on a hot?: Movement constraints in the speech of a three-year-old blind child. *Language*, 249-273.

Appendix

Answers to the Exercises

Linda Beaudin and Jeffrey L. Sokolov
University of Nebraska at Omaha

Chapter 2: Basic Measures of Child Language

2.4.1: We used the following command to answer this question:

```
maxwd +t*CHI +g1 +c1 +d1 *.cha | mlu
maxwd +t*CHI +g1 +c5 +d1 *.cha | mlu
```

A wildcard (*) is used to speed the process by searching through all of the files ending in **.cha**. Note that if we do not include the **+c1** option, the MAXWD program will default to finding the single longest utterance. Thus the following command would have achieved the same result as the first command above:

```
maxwd +t*CHI +g1 +d1 *.cha | mlu
```

2.4.2: We used the following command to answer this question:

```
freq +t*CHI +s"*-%%" +z50w *.cha
freq +t*CHI +s"*-%%" +z51w-100w *.cha
```

We would continue to compute the TTR for successive 50-word bands by modifying the **+z** option as needed. Notice that the **w** in the **+z** option denotes *words*, in contrast to the **u** in Command Box 2.3, which denotes *utterances*.

2.4.3: This analysis must be performed for all the mothers when their children are 20 months old and again at 30 months. This will require first using the data in the 20mos directory and then changing to the 30mos directory to perform the same analysis. **Hint**: The files for the complete New England corpus are marked with either an **a** (for 14 months), a **b** (for 20 months) or a **c** (for 30 months). Thus, the filename **kid068b.cha** denotes a 20-month-old child. Of course, we are analyzing