CLINICAL LINGUISTICS & PHONETICS, 1996, VOL. 10, NO. 3, 169–187

Analysis of stuttering using CHILDES and CLAN

N. BERNSTEIN RATNER, B. ROONEY

University of Maryland at College Park

B. MACWHINNEY

Carnegie Mellon University

Abstract

In this article we present a method for coding fluency in conversational transcripts. This coding system was specifically designed to be compatible with CHAT, the convention used in the Child Language Data Exchange System (CHILDES). Further, we detail a short computational program compatible with the CHILDES CLAN programs which can compute tallies of normal and stuttered disfluencies and perform a breakdown of the types and loci of stuttered moments. The value of a standard transcription system for coding fluency and of archived data sets to the future of fluency research is discussed.

Keywords: fluency, transcription, computation.

The Child Language Data Exchange System (CHILDES): an overview

The CHILDES Project is a three-pronged initiative directed at improving the collection, transcription, dissemination and analysis of child language data. Its major components are an open-access archive of language records, an explicit set of language transcript coding conventions, and a package of computational tools for the computer-assisted analysis of language data. MacWhinney and Snow (1985, 1990) and MacWhinney (1991, 1995a,b) provide detailed discussion of the evolution of the CHILDES initiative, which we will briefly summarize here for the benefit of readers unfamiliar with CHILDES. In 1984, researchers in child language acquisition met to consider changes in the way spontaneous language data were typically coded and analysed. Although child language research historically has relied quite heavily on transcriptions of naturalistic data from children and from adult–child interaction, the field had not adopted standard transcription conventions, mechanisms for sharing data, or structured approaches to the computer-assisted analysis of transcript data. Each had the potential to greatly improve the quality of language acquisition research. The Child Language Data Exchange System (CHILDES) was designed to address these goals. Originally a project funded by the John D. and Catherine T. MacArthur Foundation, it has grown from an effort to collect existing computerized language transcripts from just over a dozen researchers to an initiative which has

Address correspondence to: Nan Bernstein Ratner, Department of Hearing and Speech Sciences, University of Maryland at College Park, College Park, MD 20742, USA.

amassed data from over 100 studies of language acquisition and disorder, has developed rich and detailed coding conventions for newly gathered data, and has compiled programs for rapid transcript analysis.

The system is divided into a data archive, transcription conventions and computing utilities. In the following sections we will outline the nature of these components, demonstrate how they have changed the nature of research in child language acquisition, and propose ways in which CHILDES can serve as a model for new ways of using language production data in fluency research.

The CHILDES database

The CHILDES database, or Child Language Archive, is an open-access repository of child language transcript data. International access is available through the Internet. Its current holdings exceed 160 million characters of text, and include records from almost 100 researchers and well over 1000 research subjects. The database contents are both historical and contemporary, and range from Brown and Bloom's original samples to newer corpora added on an ongoing basis. The archive also contains records from almost two dozen languages, and from bilingual and language-impaired children.

The conceptual importance of the archive to current trends in child language acquisition research is inestimable. The archive is without doubt the most cooperative venture ever undertaken in the field of language acquisition research, and some of its impacts are readily observable. First, the open-access nature of the database has encouraged the use of archived data to re-examine and reconceptualize basic concepts in child language acquisition. It has become an impressively utilized resource: in the past 5 years the CHILDES database and associated computational programs have been referenced in over 250 articles, chapters, presentations and dissertations in language acquisition, language use, and language disability. Its use to test predictions about the adequacy of differing theoretical approaches to describing child language development has been frequent, and has led to changes in the way evidence is used to support or refute linguistic argumentation. In particular, the use of archived data to suggest the superiority or equivalent adequacy of contrasting characterizations of the language learning process in sequential analyses of the nature of the same sample of child speech has been frequent and thought-provoking (e.g. Bloom, 1990; Hyams and Wexler, 1993; Ihns and Leonard, 1988; O'Grady, Peters and Masterson, 1989; Thomas, 1994).

Second, the archive was compiled with an emphasis on unified transcription conventions, which we will address in detail in a subsequent section of this paper. The importance of unified transcription conventions goes beyond standardizing the readability and interpretability of transcript data. When records are united by a common format, it is possible to perform group analyses on a scope unimagined earlier in the history of language research. For example, because the database records are searchable electronically, it is now possible to examine hundreds of spontaneous speech samples to find the proportionate occurrence of a particular structure in the speech of 3-year-old children, or the degree to which over-regularization of a grammatical rule is seen in 4-year-olds. Thus, the database permits the sample size used for linguistic analysis to greatly exceed that typically gathered by any single researcher, and increases the power of observations which can be made about the nature of children's language use. By way of example we will provide a few representative examples from the recent literature in child language. Gropen, Pinker, Hollander, Goldberg and Wilson (1989) were able to scan over 86 000 child utterances in the archive for instances of the dative alternation in English, a relatively rare form in early child language. Thomas (1994) was able to survey 200 000 child utterances from nine subjects and seven researchers in order to analyse the evolution of reflexives (another relatively rare phenomenon) in children's expressive language development. Sokolov and Moreton (1994) were able to pool data from 58 children followed by eight different researchers in their assessment of individual differences in children's tendencies to imitate maternal language models. Ely and McCabe (1993) were able to supplement their elicited observations of developmental changes in how children quote the speech of others with spontaneous examples of such behaviours in the archived records of 25 children from three research corpora. These and numerous other examples illustrate how archived data can be integrated and used beyond the scope of original research questions to expand our knowledge of child language acquisition.

CHAT

The coding conventions used in formating database entries are referred to as CHAT (Codes for the Human Analysis of Transcripts). A brief outline of CHAT conventions is provided by MacWhinney (1995a); full transcription conventions appear in MacWhinney (1991, 1995b). CHAT files adhere to conventions specifying the internal structure of files, how speakers' utterances and turns are coded, and how utterances can be multiply coded for varying levels of analysis. An example of a CHAT transcript appears in Table 1. It contains information regarding the interaction participants, what they said (provided on the 'main' or speaker tier), and any relevant subtiers coded for morphological, syntactic, phonological or pragmatic content. It is of interest to note that, because CHAT files are editable, an original transcript can be reannotated by its own author or other researchers to include levels of analysis not originally of interest. For example, in the example provided, the child's main tier was originally augmented only by a %PHO or phonological tier. When morphological analysis of the data appeared fruitful, additional tiers of information were added to the original record. As we will discuss below, some of these additional tiers can now be generated automatically through a command sequence. Researchers who have been using SALT coding for their clinical or research records can utilize the CLAN SALTIN utility to convert SALT files to CHAT format.

CLAN

Once transcripts adhere to a uniform structure and coding schema, it then becomes possible to develop analytical programs which can search the contents of single and multiple records for behaviours of interest. In CHILDES, the computational utilities are called CLAN (Computerized Language analysis). CLAN utilities are briefly summarized in MacWhinney (1995a), and in full detail in MacWhinney (1991, 1995b). CLAN programs rest in a subdirectory of the database and, like the data, are available on an open-access basis through the Internet. Full access instructions can be found in MacWhinney (1991, 1995a,b) as well as in Sokolov and Snow (1994). Among CLAN program capabilities are frequency counts for broadly and narrowly defined behaviours of interest (FREQ), for the utterance contexts in which

Table 1. *Format of a basic CHAT file annotated for fluency behaviours*

```
@Begin
@Participants: SUB Subject
@Date:     21-DEC-1993
@Coder:    Becky Rooney
@warning   This file contains fluency codes not included in MacWhinney (1995b). For full
           specification of the codes and their use, please refer to Bernstein Ratner, Rooney
           and MacWhinney, 'Analysis of stuttering using CHILDES and CLAN', Clinical
           Linguistics and Phonetics, 1996, p. 169–187. Users may need to edit their depfiles
           to permit these files to pass CHECK.
*SUB: um@fp that's possible yeah@fp uh@fp but you+know@fp I you+know@fp I/(2)
      (ha)ve gone through you+know@fp uh@fp all/(2) kinds of programs you+know@fp
      and you+know@fp I've read up on what I think that/(2) um@fp you+know@fp
      h(&2h)ow s:tut~tering can s:ometimes be cured through um@fp ps:ychological
      counselling.
*SUB: so you+know@fp when I read that it's like@fp#oh@fp yeah@fp [//] th(&2T)at's the
      new like@fp new@pr hope or whatever you+know@fp I'm you+know@fp [/]
      I'm@PR you+known@fp [/] (I'm@PR) just that way whatever@fp because o(&4o)f
      my uh@fp my+uh@PR past f:ailures.
*SUB: oh+yeah@fp/(2)#I know but it's just that I/(2) {tend to uh@fp you+know@fp
      I+tend+to@PR like@fp look f:orward you+know@fp/(2) rather than &s
      you+know@fp like@fp dwell in the uh@fp present you+know@fp which/(2) is~n't
      always good.
*SUB: well@fp uh@fp let's see#uh@fp I would say um@fp being like@fp uh@fp [/]
      being@PR where a {teacher would say uh@fp you+know@fp let's go in a c:ircle
      you+know@fp for like@fp group discussion you+know@fp and he s:tarts like@fp
      you+know@fp like@fp [/] he+starts@PR on the opposite s:ide of the room
      you+know@fp and I {just hap~pen to be like@fp the last one.
*SUB: that's like@fp uh@fp the worse case.
      @End
```

such behaviours are found (KWAL), for co-occurring linguistic behaviours (COMBO), and for interactional contingencies (CHIP). Sokolov and Snow (1994) have compiled a useful set of tutorial exercises to demonstrate the application of CLAN programs to the analysis of a wide range of language behaviours.

Linguistic analysis of stuttering: some limitations of our research framework

Fluency research has not yet adopted a collective approach to data analysis, although both Frank Wijnan and Susan Meyers Fosnot have contributed data sets to the CHILDES archive, and we are currently reformating both imitative and spontaneous language data from the children reported in Bernstein Ratner and Sih (1987) and Brundage and Bernstein Ratner (1989) for archive inclusion. Wijnan's data (the 'Utrecht' corpus, CHILDES archive non-English directory, MacWhinney, 1995b) and Fosnot's data (Fosnot corpus, CHILDES archive Language Impairments directory, MacWhinney, 1995b) each use their own system of fluency transcription, which impedes eventual consolidation of databases for more extensive analyses of fluency across multiple populations. Lack of a substantial body of archived data hampers both secondary analysis of published data, as well as examination of the generalizability of research findings. Most studies of fluency–language interactions in children and the nature of interactions between stuttering children and their parents have utilized small *n* designs. Further, differing approaches to the operational definition of language measures have made it difficult to draw parallels between the findings of different studies.

Research into the possible linguistic determinants of stuttering loci and frequency has historically utilized two major approaches to data collection: the analysis of spontaneously generated conversational speech, and the analysis of speech elicited through sentence imitation and modelling tasks (Bernstein Ratner, 1996).

The use of either approach in understanding how stuttering interacts with language acquisition and use has its limitations. The first of these is the unclear status of the developmental difficulty of certain speech tasks for children as a group, or for children as individuals (Bates, Dale and Thal, 1995). Establishment of *a-priori* taxonomies of task difficulty for imitation and modelling tasks (e.g. Bernstein Ratner and Sih, 1987; Gordon, 1991) represent, at best, only an approximation of the current psycholinguistic understanding of the probable ages at which the majority of young children spontaneously produce sentences of particular types. Use of *post-hoc* taxonomies involves similar concerns, and tends to force a macroanalytic, corpus-based approach to the differences between samples of fluent and stuttered speech (e.g. Gaines, Runyan and Meyers, 1991).

The second problem inherent in analysis of elicited and spontaneous speech samples from stuttering children is the fact that speech production is a layered phenomenon, in which output can be analysed using any number of discretely defined parameters. For example, researchers can ask whether stuttering varies as a function of the length of utterance in syllables, words or morphemes (Brundage and Bernstein Ratner, 1989), as a function of the length of utterance in clausal units (Wall, 1980), the lexical composition of sentences (Hubbard and Prins, 1994), the sentence structures and types used (Bernstein Ratner and Sih, 1987; Gordon, 1991), or, finally, their pragmatic functions (Weiss and Zebrowski, 1991, 1992). Though one can operationally define a linguistic unit of interest for analysis, it is inescapable that a spoken sentence represents the integration of an entire linguistic hierarchy of elements into a coherent and functional whole.

These problems are not unique to stuttering research. For years, child language researchers have endeavoured to ascertain whether observations made about language acquisition using small samples of children (such as Brown's Adam, Eve and Sarah) can be validated using larger subject samples. It has always been difficult to track the development and use of optional and complex structures in small language corpora. Finally, the difficulty of examining cross-domain relationships in children's output (for example, relationships between phonology and the lexicon, the lexicon and syntax, syntax and pragmatics) was a significant problem for researchers who wanted to understand how limitations and abilities in one domain might affect performance in another.

Applications of CHILDES to fluency research

CHILDES offers a number of advantages to researchers in fluency. Conceptually it has paved the way to an understanding among language researchers that shared and open access to data can increase the power of linguistic analyses, and enable reanalysis and secondary analysis of data. For a field such as stuttering, where subject populations are usually quite small, the potential for pooled analyses of transcript data offers the opportunity for more informative cross-sectional group analyses, as well as estimation of the degree of individual variation.

In the linguistic analysis of stuttered speech, differences in operational definitions from study to study have mitigated the impact of study findings. No two studies utilize the same definitions of syntactic complexity, and few studies examine the relationship between stuttering and more than one linguistic variable concurrently. Researchers continue to note that past estimates of the effect that syntactic complexity has on the frequency of stuttering and normal disfluency are typically hampered by the unknown contributions of utterance length, lexical composition and discourse function. CLAN has the potential to address this problem, because a single transcript can be analysed using varying measures of language use.

An interesting opportunity is afforded to fluency researchers by the emphasis in both CHILDES and CLAN on dyadic interaction analysis. There are a number of corpora in CHILDES which focus on mother–child interaction at varying stages of child language development, and the computational capacity of CLAN permits analysis of parental behaviours as well as child and adult turn-taking behaviours which appear to be contingently related (Sokolov and Moreton, 1994). As we have noted elsewhere, research on interactions between stuttering children and their parents probably needs to move beyond measures of speech rate entrainment and turn-taking latencies to examine those lexical and syntactic behaviours in parental speech which appear to scaffold fluency in stuttering children (Bernstein Ratner, 1996). Such analyses are facilitated by CLAN.

Coding fluency behaviours in CHAT

For fluency researchers to utilize the full potential of CHILDES, it was necessary to develop fluency coding conventions which encompass the range of behaviours seen in the speech of stuttering children. This was not an entirely trivial issue. The original CHAT coding conventions (MacWhinney, 1991) anticipated a primary interest in transcription of normally fluent speech. Thus, while a few isolated codes were provided for phonological fragments, broken words and phrase repetitions, the original CHAT codes could not represent the full range of disfluencies of interest to fluency researchers. Further, up until this point, codes for normal and stuttered disfluencies were spread across main and dependent tier systems and could not easily be used to indicate the location and specific nature of disfluent segments (MacWhinney, 1995b: 97).

In the following sections we present a system for coding fluency. We prefer to use the term ‘fluency’ rather than ‘stuttering’ because we view our system as a means for characterizing the flow of any speaker’s output. MacWhinney (1995b) presents codes for other primary aspects of language form and use, such as phonetic codes, morphosyntactic codes and speech act codes, and we view fluency within this larger framework of characterizing spoken language. In developing fluency codes we had four primary goals. The first was *transparency*. We would define this goal as one of readability; it should be possible for readers of the transcript to readily and easily reconstruct the actual nature of the fluency breakdown from the codes we use.

The second goal was one of *transcription ease*. We tried to develop codes which used a minimal number of keystrokes and avoided combination keystrokes. The third goal was compatibility with existing CHAT conventions. We sought to avoid using current CHAT symbols for multiple purposes, or in ways which would create ambiguity in reading CHAT transcripts. For example, we preserved the use of the ampersand (&) to reflect the notion of word fragments, and built upon this symbol. We avoided post-codes on words which used hyphens (e.g. *well-fp* to indicate a filled pause) because hyphens are used to mark morphological inflections in CHAT.

The fourth goal was the most ambitious, but the most important, in our opinion. The development of transcription coding conventions is relatively easy when compared to the problem of developing *computationally useful* codes. Because CHAT interfaces with CLAN, it was important to develop codes which would not interfere with transcript analysis using current CLAN programs, and which would permit the development of new CLAN-compatible programs which could compute tallies of behaviours of interest to fluency researchers. In the next sections we will review the development of the fluency codes and the development of the new computational programs.

The CHAT fluency codes

Codes for stuttering

As mentioned above, we built upon existing CHAT convention which uses the ampersand to represent a word fragment. Further, because lexical analysis of a language transcript must be able to ignore variations in pronunciation or fluency, it was necessary for word-level disfluencies to be coded within parentheses. CLAN programs can be instructed to ignore material in parentheses when computing frequencies for lexical items, computing TTR, MLU, DSS, etc. With these concepts in mind, we developed the following codes:

> *Sound/syllable repetitions*: the repeated element or sequence is marked in parentheses, with the segment and number of iterations noted, as in the following examples:

ba(&1be)by	gloss: *ba-ba-by*
th(&3T)at	gloss: *th-th-th-that*

These examples require us to further explain the desirability of noting disfluent segments using phonetic symbols. As we will note later, it will be possible to compute phonetic profiles of disfluency should this be of interest to researchers or clinicians. This is possible only in a language such as English, however, if phonetic transcription, rather than orthography, is used to annotate the disfluency. CHAT uses a computer-keyboard compatible phonetic transcription system known as UNIBET (Bernstein Ratner, 1994). One of its major features is the use of single characters to replace IPA digraph elements, as well as its replacement of non-keyboard symbols with standard keyboard characters. See Appendix 1 for a listing of UNIBET symbols.

> *Whole word repetitions*: We code these by use of a slash, the standard CHAT retrace code, followed by the number of repetitions, all within parentheses. Example:

that's(/2) interesting	gloss: *that's that's* *that's interesting.*

> *Prolongations*: We use the standing CHAT code for prolonged segments, the colon, placed after the prolonged element. Example:

s:omething	gloss: *ssssomething*
so:mething	gloss: *soooomething*

Blocks: Blocks are indicated by the symbol ˆ immediately prior to the blocked segment (no intervening spaces). Example:
ˆI tend to have blocks early in sentences.

Broken words: We use the existing CHAT code, the tilde (~) for broken words. Example:
Is that a rhin ~ ocerous?

Using these codes as a guideline it is also possible for any researcher to develop additional codes for disfluency behaviours not included above. Such codes would need to be explained in the file header for the speech sample, using the CHAT @Warning header.

Codes for normal disfluencies

Hesitations (unfilled pauses): We use the existing CHAT code of the hash sign (#), separated from surrounding words by a space. Example:

I think I can meet you in # twenty minutes.

Filled pauses: Filled pauses present an interesting problem. CHAT did not have standing conventions for coding FPs. We first explored the possibility of constructing a file of typical English FPs such as *well*, *you know*, *um*, etc. and then programming CLAN searches to count or ignore them as the researcher desired. However, in English as in most other languages, FPs are often well-formed, meaningful words (e.g. *Do you feel well?*, *Tell me what you know*) which one would not want to misclassify during computations. Thus, we prefer to suffix filled pauses with the sequence *@fp*. In order for multi-word filled pauses to count as a single event, we use the CHAT convention of using the + symbol to link compound words. Example:

Well@fp you+know@fp how I hate spinach.

Phrase repetitions
CHAT contains standing codes for retraces or repetitions of phrases. However, for the ease of computing the degree to which a subject engages in phrase repetition and postponement activities, we developed an alternative code which follows the format of filled pause coding. In this alternative coding, elements within a phrase repetition are linked by + and the sequence is suffixed by the *@pr* code. Example:
I'm going to I'm+going+to@pr study later.

A sample CHAT file, showing basic CHAT format and the use of the fluency codes, is provided in Table 1. A more elaborate transcript, with multiple dependent tiers, is shown in Table 2. While files containing multiple tiers can be time-consuming to construct, basic fluency-annotated CHAT files do not take any more time to compile than any other system of transcription which annotates the location, type and severity of stutters and disfluencies. Further, the construction of multiple tiers for

Table 2. *A more elaborate CHAT file with additional dependent tiers*

Sample mother–child dyad chat file

Guide to tier codes:

*CHI =Main tier corrected for articulation and (missing words)

%PHO =Phonetic transcription

%MOD =Model (correct target) adult articulation

%GLS =Best estimate of child's intended utterance (gloss)

(Note: all data represent real events, with the *exception of disfluencies*, which were added to this file for illustrative purposes)

@Begin	
@Participants:	CHI Adami Child, MOT Nan Mother, SIS Jamie Sister, FAT Bob Father
@Age of child	4;3
@Date:	10-Apr-1991
@Coder:	Rachel Brown
@Coding:	CHAT 1.0
@warning:	This file contains fluency codes not included in MacWhinney (1995b). For full specification of the codes and their use, please refer to Bernstein Ratner, Rooney and MacWhinney. 'Analysis of stuttering using CHILDES and CLAN', *Clinical Linguistics and Phonetics*, 1996, p. 169–187. Users may need to edit their depfiles to permit these files to pass CHECK.
*MOT:	that's a very old teddy bear.
*MOT:	that was mommy's when I was a little girl.
*CHI:	w(&3w)ant to ke ~ ep it?
%pho:	wa tu kip It?
%mod:	want tu kip It?
%gls:	do you want to keep it?
*CHI:	w(&3w)ant to # keep?
%pho:	wa tu kip?
%mod:	want tu kip?
%gls:	do you want to keep it?
*MOT:	do I want to deep it?
*MOT:	well # don't you love it?
*MOT:	don't you love him?
*CHI:	ˆI love(/2) him.
%pho:	ai nAv hIm.
%mod:	ai IAv hIm.
%gls:	I love him.
*MOT:	well you can you can have him then.
*MOT:	I will let you have him.
*MOT:	his name is Timothy.
*MOT:	Timothy bear.
*CHI:	and Bu ~ ddy (is) n(&n3)ame-ed Jack.
%pho:	En bAdi nemd dZ&k.
%mod:	&nd bAdi nemd dZ&k.
%gls:	and Buddy is named Jack.
*MOT:	Buddy's name is Jack?
*MOT:	I thought Buddy's name was Jack # uh@fp I thought Buddy's name was Buddy.
*MOT:	hmm.

Table 2. *Continued.*

*CHI:	that(/3) is h:is # name.
%pho:	d&t Iz hIz nem.
%mod:	T&t Iz hIz nem.
%gls:	that is his name.
*CHI:	that is his name we(/3) ˆcall him.
%pho:	d&t Iz hIz nem ni kO hlm.
%mod:	T&t Iz hIz nem wi kol hlm.
%gls:	that is the name that we call him.
*CHI:	his name ˆis Jack.
%pho:	hIz nem iz dZ&k.
%mod:	hIz nem iZ dZ&k.
%gls:	his name is Jack.
*MOT:	you're going to change Buddy's name to Jack?
*MOT:	that will be his new name?
*CHI:	yeah.
%pho:	j&.
%mod:	j&.
%gls:	yeah.
@End	

phonetic, syntactic or pragmatic analysis simply replaces series of discrete analyses of the same data when researchers or clinicians wish to perform multi-faceted analyses of a speech sample. Any minimal amount of extra transcription time involved in constructing files within CHAT format is more than offset by the rapid ability to analyse multiple aspects of conversational speech using a single transcript—for example, fluency counts and multiple types of grammatical and conversational analyses can both be computed from a single file without reformatting or recoding.

Warning headers

The fluency transcription system we propose here does use some CHAT symbols in ways that were not intended for the general transcription of child language data. As with other files which may make alternative or unconventional use of CHAT symbols or coding conventions, users are strongly encouraged to include an @warning header in files which include the fluency codes provided in this article. A suggested wording for the header is shown below:

> *@begin*
> *@participants: CHI John child*
> *@date: 21-DEC-94*
> *@coder: Mary Smith*
> *@warning: This file contains fluency codes not included in MacWhinney (1995b). For full specification of the codes and their use, refer to Bernstein Ratner, Rooney and MacWhinney, 'Analysis of stuttering using CHILDES and CLAN', Clinical Linguistics and Phonetics, 1996, p. 169–187. Users may need to edit their depfiles to permit these files to pass CHECK.*

'Talking transcripts'

Even excellent transcription has its limitations. In fluency research there are significant concerns about the reliability of the transcription of fluency behaviours, particularly their loci (Cordes and Ingham, 1994). Further, acoustic analysis of the speech of fluent and stuttering speakers faces the cumbersome obstacle of locating samples which can be matched for parameters of interest to the researcher. Analysis must proceed through comparison of either audio- or video-records, and the activity transcript.

Computer technology offers a new opportunity to link transcripts with the audio-signal. The Talking Transcripts initiative, under development at Carnegie Mellon (MacWhinney, 1995a,b), uses optical erasable disks, a 16-bit digitizer board, and the Macintosh operating system to link digitized speech samples and CHAT transcripts. Once the audio signal has been digitized and linked to portions of the CHAT transcript, the user can click on a section of the transcript and hear the actual speech record. As MacWhinney (1995a,b) notes, linkages between audio- and video-recordings and transcripts will be of particular interest to researchers interested in examining conversational synchrony in dyadic interaction, as well as researchers interested in the acoustic specification of transcript entries.

Development of the FLUCALC utility

Once codes were developed for the primary disfluency behaviours of interest to researchers and clinicians, we turned our attention to the development of computational tools which would run basic fluency calculations on a sample of transcribed speech. As indicated previously, we relied on existing CLAN utilities, such as FREQ, PHONFREQ and MLT to perform these analyses.† Normally, CLAN commands run one analysis at a time. For example, the FREQ command would do a frequency tally of the lexical entries in a speech sample, or of a class of behaviours defined by the researcher. Many fluency analyses, however, require a string of counts which are fairly predictable. For example, calculating mean stuttered words will require one to count the number of intended words in a sample, excluding fillers and repeated elements, and then the number of stuttered and normally disfluent episodes. Because these three sequences would always be required in order to compute the incidence of disfluency in a sample, we decided to create batch files which compiled a series of separate CLAN command sequences into a single command. We call this utility FLUCALC. Currently, FLUCALC is not included in the CLAN programs distributed through the archive. It must be added to CLAN by the user, following instructions we supply below. The user must also create four additional files and store them in the \CLAN\bin subdirectory. These files are:

flucodes, which lists the symbols used to indicate stuttered disfluencies;
normdis, which lists the symbols used to indicate normal disfluencies;
newpunct.txt, which teaches CLAN to strip fluency codes from words when it does lexical and syntactic analyses; and
alphabet, which tells CLAN to treat multi-letter sequences as one, when appropriate for certain analyses. *Note*: the *alphabet* file used for fluency counts differs

†To obtain copies of free CLAN software, consult either MacWhinney (1995a) pp. 154–156, or MacWhinney (1995b) 280–283.

from the one which is required to do phonetic counts (Bernstein Ratner, 1994). If both fluency and phonetic counts are being run, they must be run separately and the alphabet file edited for each purpose.)

Once the FLUCALC.bat file and support files are created, they will enable the CLAN user to count the number of intended words in a speech sample, excluding normal disfluencies, retraces and repetitions; the number of stuttered words; the number of normal disfluencies; a frequency profile of stuttering behaviours; and mean length of speaker turn. FLUCALC.bat is easily edited to include any other CLAN commands which a researcher routinely would like done on a transcript or corpora of transcripts. The full specifications for the FLUCALC.bat file and the contents of the *flucodes*, *normdis*, *newpunct.txt* and *alphabet* files are provided for users in Appendix 2. They need only be typed into ASCII files and inserted into the CLAN subdirectory.

Executing FLUCALC

The basic form of the FLUCALC command is:

```
flucalc ⟨speaker tier code⟩ ⟨filename⟩ ⟨output filename⟩
```

The speaker tier code refers to the three-letter sequence which identifies the speaker whose data are to be analysed. The default code for transcribing a child in CHAT is *CHI, with the child's full identity specified in a file header. This convention enables unified group analyses across any number of transcripts and subjects. The filename is the name given to the transcript file. CHAT files are conventionally suffixed by *.cha* to signal that they are in CHAT format. The output filename is simply the user-defined destination for the analysis. Example:

```
flucalc chi johnny.cha johnny.out
```

This command tells CLAN to perform a fluency analysis on the child's tier (as opposed to his mother's or the examiner's output) in the file *johnny.cha* and to send the results to the file *johnny.out*, where it can later be retrieved and read. Sample output from this procedure is supplied in Table 3.

Future directions for fluency analysis

One of the greatest opportunities which CHAT and CLAN affords the researcher is the possibility of multiple analyses on the same set of transcribed data. CLAN supports Mean Length of Utterance, Mean Length of Turn, and Type–Token Ratio analyses. Developmental Sentence Scoring (Lee, 1974) and IPSYN (Scarborough, 1990) are under development. CLAN supports a full morphological parser (MOR) which will automatically generate a morphologically coded tier from the main speaker tier, for use in morphological and syntactic analysis. The CLAN FREQ utility can be scoped using search files to look for the presence of specific types of words or word classes. The CHIP utility examines the degree to which a speaker's output incorporates elements from the preceding speaker's utterance, a measure of imitativeness or conversational scaffolding which can be tied to the relative fluency of a child's output. The COMBO utility will search for co-occurring behaviours of interest to a particular researcher (i.e. the number of occasions on which normal disfluencies and stutterings occurred as a run).

Table 3. *Sample output from the FLUCALC utility*

```
FREQ.EXE +r3 +r4 +t*chi +pnewpunc.txt -s@normdis adampost
FREQ.EXE is conducting analyses on:
 ONLY speaker main tiers matching: *CHI;
From file <adampost>
 2 a              2 him          5 my           3 see
 1 all            8 his          1 myself       1 sing
 4 and            1 hmm          4 name         1 sit
 2 big            3 hold         1 name-ed      4 snake
 2 book-s        16 i            1 name-s       1 steven
 1 books          1 i'm          2 need         1 stick-ing
 1 buddy          1 in           1 new          3 tail
 1 butt           5 is           1 next         3 that
 1 call           5 it           4 no           1 there
 1 cause          2 jack         5 not          6 this
 1 chang-ing      1 jamie        1 nothing      5 to
 2 dinosaur       2 keep         1 now          1 turn
 3 do             2 know         1 ok           3 two
 1 do-'nt         2 like         7 one          1 tyranasaurus
 1 doll-s         3 lissa        1 or           4 want
 1 down           3 listen       1 out          1 we
 3 first          1 love         1 paul         1 whale
 1 get            3 maybe        5 read         1 yeah
 1 having         3 me           1 rex          6 yes
 1 he             1 mean         1 right        1 you
 1 help           3 mom          5 said         1 your
 1 here
────────────
 85   Total number of different word types used
200   Total number of words (tokens)
  0.425   Type/Token ratio

FREQ.EXE +r2 +t*chi +s@normdis adampost
FREQ.EXE is conducting analyses on:
 ONLY speaker main tiers matching: *CHI;
*************************************
From file <adampost>
5 #              1 ^keep-ing@pr     6 um@fp
────────────
  3   Total number of different word types used
 12   Total number of words (tokens)
  0.250   Type/Token ratio

FREQ.EXE +t*chi +s@flucodes adampost
FREQ.EXE is conducting analyses on:
 ONLY speaker main tiers matching: *CHI;
*************************************
From file <adampost>
1 ^call            1 do~ll-s      1 l:isten        1 n:ame-s
1 ^i               2 f:irst       1 listen/3       1 s:ing
1 ^s               1 h:is         1 love/2         1 t&t3urn
1 ^keep-ing@pr     2 i/2          3 m:aybe         1 that/3
1 ^one             1 ja~mie       1 n&n2ew         2 w&3want
1 bu~ddy           1 ke~ep        1 n&n3ame-ed     1 we/3
1 di~nosaur        1 l:ike
────────────
 26   Total number of different word types used
 31   Total number of words (tokens)
  0.839   Type/Token ratio
```

Table 3. *Continued*

```
PHONFREQ.EXE +b*CHI +s@flucodes adampost
1PHONFREQ.EXE is conducting analyses on:
 ONLY speaker main tiers matching: *CHI;
****************************************
From file <adampost>
2     &3w    initial=0,  final=0,   other=2
2     &n     initial=0,  final=0,   other=2
1     &t     initial=0,  final=0,   other=1
3     /2     initial=0,  final=3,   other=0
3     /3     initial=0,  final=3,   other=0
5     ^      initial=5,  final=0,   other=0
1     a~m    initial=0,  final=0,   other=1
1     e~e    initial=0,  final=0,   other=1
2     f:     initial=2,  final=0,   other=0
1     h:     initial=1,  final=0,   other=0
1     i~n    initial=0,  final=0,   other=1
2     l:     initial=2,  final=0,   other=0
3     m:     initial=3,  final=0,   other=0
1     n:     initial=1,  final=0,   other=0
1     o~l    initial=0,  final=0,   other=1
1     s:     initial=1,  final=0,   other=0
1     u~d    initial=0,  final=0,   other=1
MLT.EXE +r3 +r4 +t*chi +pnewpunc.txt -s@normdis adampost
MLT.EXE is conducting analyses on:
 ONLY speaker main tiers matching: *CHI;
****************************************
From file <adampost>
MLT (xxx and yyy are INCLUDED in the utterance and morpheme counts):
  Number of: utterances=59, turns=52, words=213
    Ratio of words over turns=4.096
    Ratio of utterances over turns=1.135
    Ratio of words over utterances=3.610

MLU.EXE +r3 +r4 +t*chi +pnewpunc.txt -s@normdis adampost
MLU.EXE is conducting analyses on:
 ONLY speaker main tiers matching: *CHI;
*****************************************************
From file <adampost>
MLU for Speaker: *CHI:
MLU (xxx and yyy are EXCLUDED from the utterance and morpheme counts):
  Number of: utterances=59, morphemes=208
    Ratio of morphemes over utterances=3.525
    Standard deviation=2.339

MLT.EXE +r3 +r4 +t*MOT +pnewpunc.txt -s@normdis adampost
MLT.EXE is conducting analyses on:
 ONLY speaker main tiers matching: *MOT;
*****************************************************
From file <adampost>
MLT (xxx and yyy are INCLUDED in the utterance and morpheme counts):
Number of: utterances=107, turns=53, words=614
    Ratio of words over turns=11.585
    Ratio of utterances over turns=2.019
    Ratio of words over utterances=5.738
```

Discussion

This paper is not intended to function as a full tutorial for potential users of the CHILDES archive or its resources; users should consult MacWhinney (1995b) for explicit instructions on archive access, CHAT coding conventions and the exhaustive listing of CLAN programs and their command structure. Rather, our intent was to acquaint researchers influency with the conceptual utility both of archived resources and the computational analysis of language transcripts. Both have substantially altered approaches to the field of child language acquisition research and have the potential to shape future investigation in fluency in similar fashion.

One of the more interesting sequelae to the CHILDES Project has been the active interchange of CHILDES system users on an electronic bulletin board (info-childes@andrew.cmu.edu). The bulletin board has enabled users to query one another's data and computational approaches, and to 'troubleshoot' both coding and analysis schemas.

Research in a discipline progresses through a variety of mechanisms. While some changes in research orientation are enabled through technological innovation (i.e. development of brain-imaging techniques, or computational advances), others result from philosophical changes in researchers' orientation towards data collection, use and interpretation. The CHILDES Project offers researchers in fluency the opportunity to exploit new technological approaches to data management, as well as an opportunity to work collectively towards more maximal use of data gathered in the study of fluency and its disorders.

Acknowledgements

Work on the CHILDES Project has been supported by grants from the John D. and Catherine T. MacArthur Foundation, the National Science Foundation, and the National Institutes of Health. CLAN programs were developed by Leonid Spektor at Carnegie Mellon University. The numerous researchers who have contributed to the archive, to the development of coding conventions and to the specifics of individual CLAN programs are acknowledged in MacWhinney (1991, 1995b). The fluency coding conventions and FLUCALC utility were originally presented at the 1994 American Speech–Language Hearing Association annual convention, New Orleans.

References

Bates, E., Dale, P. and Thal, D. (1995) Individual differences and their implications for theories of language development. In P. Fletcher and B. MacWhinney (Eds), *The Handbook of Child Language*. (Cambridge, MA: Blackwell).

Bernstein Ratner, N. (1994) Phonological analysis. In J. Sokolov and C. Snow (Eds), *Handbook of Research in Language Development using CHILDES* (Hillsdale, NJ: Erlbaum).

Bernstein Ratner, N. (1996) Stuttering: a psycholinguistic perspective. In R. Curlee and G. Siegel (Eds), *Nature and Treatment of Stuttering: new directions*, 2nd edn. (Boston: Allyn & Bacon) (In press).

Bernstein Ratner, N. and Sih, C. C. (1987) Effects of gradual increases in sentence length and complexity on children's dysfluency. *Journal of Speech and Hearing Research*, **52**, 278–287.

Bloom, P. (1990) Syntactic distinctions in child language. *Journal of Child Language*, **17**, 343–355.

Brundage, S. and Bernstein Ratner, N. (1989) Measurement of stuttering frequency in children's speech. *Journal of Fluency Disorders*, **14**, 351–358.

Cordes, A. and Ingham, R. (1994) Time-interval measurement of stuttering: effects of training with highly agreed or poorly agreed exemplars. *Journal of Speech and Hearing Research*, **37**, 1295–1307.

Ely, R. and McCabe, A. (1993) Remembered voices. *Journal of Child Language*, **20**, 671–696.

Gaines, N., Runyan, C. and Meyers, S. (1991) A comparison of young stutterers' fluent versus stuttered utterances in measures of length and complexity. *Journal of Speech and Hearing Research*, **34**, 37–42.

Gordon, P. (1991) Language task effects: a comparison of stuttering and nonstuttering children. *Journal of Fluency Disorders*, **16**, 275–287.

Gropen, J., Pinker, S., Hollander, M., Goldberg, R. and Wilson, R. (1989) The learnability and acquisition of the dative alternation in English. *Language*, **65**, 203–257.

Hubbard, C. and Prins, D. (1994) Word familiarity, syllabic stress pattern and stuttering. *Journal of Speech and Hearing Research*, **37**, 564–571.

Hyams, N. and Wexler, K. (1993) On the grammatical basis of null subjects in child language. *Linguistic Inquiry*, **24**, 421–459.

Ihns, M. and Leonard, L. (1988) Syntactic categories in early child language: some additional data. *Journal of Child Language*, **15**, 673–678.

Lee, L. (1974) *Developmental Sentence Analysis.* (Evanston, IL: Northwestern University Press).

MacWhinney, B. (1991) *The CHILDES Project: computational tools for analyzing talk* (Hillsdale, NJ: Erlbaum).

MacWhinney, B. (1995a) Computational analysis of interactions. In P. Fletcher and B. MacWhinney (Eds), *The Handbook of Child Language.* (Cambridge, MA: Blackwell).

MacWhinney, B. (1995b) *The CHILDES Project: computational tools for analyzing talk*, 2nd edn (Hillsdale, NJ: Erlbaum).

MacWhinney, B. and Snow, C. (1985) The Child Language Data Exchange System. *Journal of Child Language*, **12**, 271–296.

MacWhinney, B. and Snow, C. (1990). The Child Language Data Exchange System: an update. *Journal of Child Language*, **17**, 457–472.

O'Grady, W., Peters, A. and Masterson, D. (1989) The transition from optional to required subjects. *Journal of Child Language*, **16**, 513–530.

Scarborough, H. (1990). Index of productive syntax. *Applied Psycholinguistics*, **11**, 1–22.

Sokolov, J. and Moreton, J. (1994) Individual differences in linguistic imitativeness. In J. Sokolov and C. Snow (Eds), *Handbook of Research in Language Development Using CHILDES* (Hillsdale, NJ: Erlbaum).

Sokolov, J. and Snow, C. (Eds) (1994) *Handbook of Research in Language Development using CHILDES* (Hillsdale, NJ: Erlbaum).

Thomas, M. (1994) Young children's hypotheses about English reflexives. In J. Sokolov and C. Snow (Eds), *Handbook of Research in Language Development Using CHILDES* (Hillsdale, NJ: Erlbaum).

Wall, M. (1980). A comparison of syntax in young stutterers and nonstutterers. *Journal of Fluency Disorders*, **6**, 283–298.

Weiss, A. and Zebrowski, P. (1991) Patterns of assertiveness and responsiveness in parental interactions with stuttering and fluent children. *Journal of Fluency Disorders*, **16**, 125–141.

Weiss, A. and Zebrowski, P. (1992) Disfluencies in the conversations of children who stutter: some answers about questions. *Journal of Speech and Hearing Research*, **35**, 1230–1238.

Appendix 1: UNIBET Symbols

IPA→UNIBET translations for English

UNIBET	IPA symbol	IPA name	Example word(s)
Consonants			
p	p	p	**p**it
b	b	b	**b**it
m	m	m	**m**itt
t	t	t	**t**ip
d	d	d	**d**ip
n	n	n	**n**ip
k	k	k	pi**ck**
g	ɡ	g	pi**g**
N	ŋ	eng	pi**ng**
f	f	f	**f**ew
v	v	v	**v**iew
T	θ	theta	e**th**er
D	ð	eth	ei**th**er
s	s	s	**s**ue
z	z	z	**z**oo
S	ʃ	esh	**sh**oe
Z	ʒ	yogh	plea**s**ure
tS	tʃ	t-esh	ca**tch**
dZ	ʤ	d-yogh	ju**dg**e
h	h	h	**h**op
w	w	w	**w**itch
W	ʍ	inverted w	**wh**ich
r	r	r	**r**ip
l	l	l	**l**ip
j	j	j	**y**ip
i	i	i	h**ee**d, b**ea**t
I	ɩ	iota	h**i**d, b**i**t
e	e	e	h**a**ved, b**ai**t
E	ɛ	epsilon	h**ea**d, b**e**t
&	æ	ash	h**a**d, b**a**t
u	u	u	wh**o**'d, b**oo**t
U	ʊ	closed omega	h**oo**d, f**oo**t
o	o	o	h**oe**d, b**oa**t (GA)
O	ɔ	open o	h**aw**ed, b**ough**t (GA) h**oar**d (RP)
A	ʌ	inverted v	b**u**d, b**u**t
a	a	a	m**a**, h**o**d, h**o**t (GA) h**ar**d (RP)
3	ɜ	reversed epsilon	h**er**d (RP)
6	ə	schwa	**a**bove
Q	ɒ	—	h**o**d (RP)

UNIBET	IPA symbols	Example word(s)
Diphthongs		
ai	ai	hide, bite
au	au	howdy, bout
oi	oi	ahoy, boy
6U	əo	hoed (RP)
R Sounds		
ir	ir	here (GA)
er	ɛr	hare (GA)
ar	ar	hard (GA)
or	or	haord (GA)
ur	ur	moor (GA)
3r	ɜr	herd, hurt (GA)
i6	iə	here (RP)
e6	eə	hare (RP)
u6	uə	moor (RP)
Suprasegmentals, etc.		
$		syllable boundary
#		morpheme boundary
##		word boundary
|		rhythmic juncture
|H		hesitation
7	?	glottal stop
”		pitch accent (primary)
’		heavy (secondary)
!		emphatic
.		falling terminal
?		rising terminal
-		continuation terminal
:	:	long, geminate

Appendix 2: Files required to support FLUCALC

Contents of the FLUCODES FILE

```
*&*
* = *
* ~ *
*:*
*^*
*/*
```

Contents of the NORMDIS file

```
*@fp
*@PR
#
```

Contents of the NEWPUNC.TXT file

```
=|^.,;?![]<>1234567890
```

Contents of the ALPHABET file

```
*:
*~*
^*
&1*
&2*
&3*
&4*
&5*
&*
tS
dZ
au
ai
ou
oi
```

Contents of the FLUCALC.bat file for Batching the fluency analyses

```
freq +r3 +r4 +t*%1 +pnewpunc.txt -s@normdis %2 >> %3
freq +r2 +t*%1 +s@normdis %2 >> %3
freq +t*%1 +s@flucodes %2 >> %3
phonfreq +b*%1 +s@flucodes %2 >> %3
mlt +r3 +r4 +t*%1 +pnewpunc.txt -s@normdis %2 >> %3
```

(other CLAN commands may be inserted as desired, to carry out other analyses)