

A Clinician's Complete Guide to CLAN and PRAAT

Nan Bernstein Ratner
University of Maryland

Shelley B. Brundage
George Washington University



Updates by Davida Fromm, Carnegie Mellon University
Last update: October 2023

INTRODUCTION: WHY you want to learn how to use these programs

We have all been taught that a speech and language sample is the “gold standard” for evaluating our cases. But you know how tedious language sampling and fluency counts can be, from transcribing to calculating usable clinical scores. Probably, the last time most clinicians actually DID a language sample was in graduate school.

This manual will teach you to use CHAT and CLAN, which will:

- Speed up your accurate transcription of data by as much as five-fold.
- Link your transcript to the actual audio or video file you made in the clinic.
- For language analysis, automatically (!!!) compute clinical measures of interest, such as MLU, TTR, DSS, Brown’s morphemes (for children), and a host of other values (WITHOUT MAKING MATH ERRORS!).
- For fluency analysis, automatically analyze the frequency of different types of disfluency behaviors and compute the percent stuttered words.
- Allow you to link to free acoustics software DIRECTLY from the transcript, to calculate values such as speech rate and acoustical features of targeted words or utterances.
- Create your OWN targeted analyses of your client’s data.
- Grammatically tag and analyze data for more than a dozen languages other than English.
- Output to PHON for phonological analysis.

These programs are free, open-source software, with excellent support from a major US government-supported team that built this wonderful utility, and a large list of community users around the world. The three programs we describe in this manual are:

CHAT and CLAN

CHAT and CLAN are a part of CHILDES (CHILd Language Data Exchange System), which provides tools for studying conversational interactions, as well as serving as a repository for language corpora from around the world (MacWhinney, 2000). CHAT is a software program for transcribing sound files using a standard set of rules (commonly called “CHAT format”). CLAN is a data analysis program for analyzing transcriptions that were transcribed in CHAT format. (For more information, go to <https://childes.talkbank.org/> .)

PRAAT

Praat is a freely available program, developed by Dutch researchers, to perform acoustical analysis (see more further in this guide). It intersects directly with CLAN, via a command under MODE menu: send to sound analyzer drop down menu.


GETTING STARTED: COMPUTER SETUP

First, turn on your computer ☺ and be connected to the Internet. Second, be aware that a large set of short screencasts demonstrating much of what is described in this manual (and more!) is available online at <http://talkbank.org/screencasts/> .

DOWNLOAD CLAN:

If you already downloaded CLAN previously, then before downloading a new version, just be sure the CLAN program is not open (QUIT CLAN). To download CLAN, go to <http://dali.talkbank.org/clan/> and download either the Windows version or the Mac version.

This is what you should see (though the version dates, indicated under each link, will continue to change as the program is updated) -- the Windows version on the left and the Mac version in the middle:

TalkBank		CLAN
Windows	Mac OS X 10.15 or later	Unix
CLANWin	CLANc	UnixCLAN
V 28-Sep-2023 11:00	V 28-Sep-2023 20:00	V 28-Sep-2023 11:00

Official **user manuals** for CHAT and CLAN are at <https://talkbank.org/> (TalkBank webpage).

Manuals

[CHAT manual](#)

[CLAN Manual](#)

[SLP Manual](#)

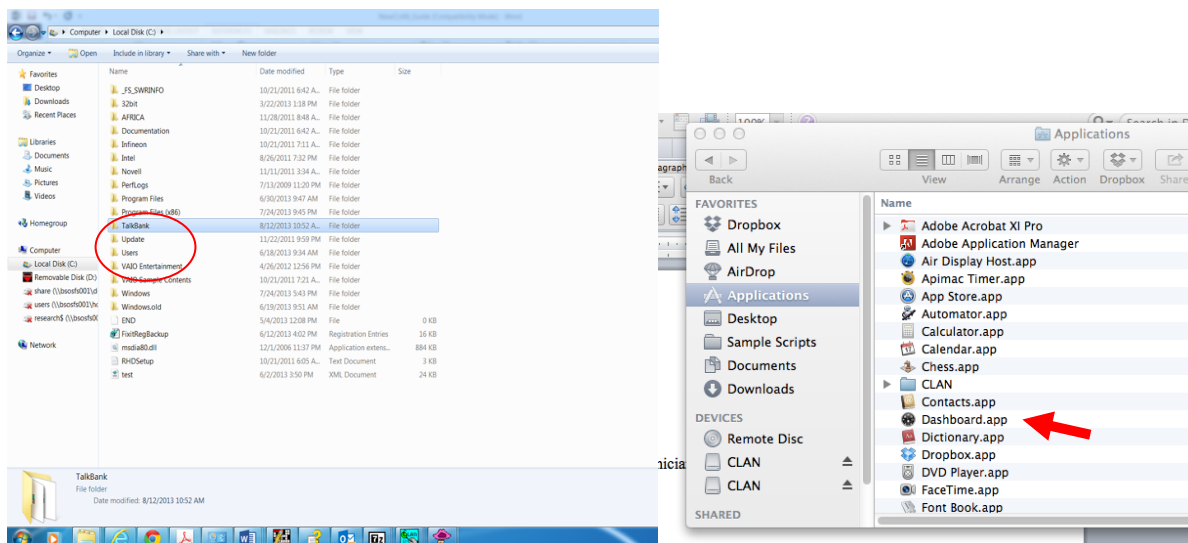
[Tutorial Screencasts](#)

These manuals are long and technical, so **don't print them**. Instead, we recommend clicking on the CHAT or CLAN manual link to read about topics as needed because they are updated regularly. Importantly, these manuals are searchable by keyword, which may be useful later.

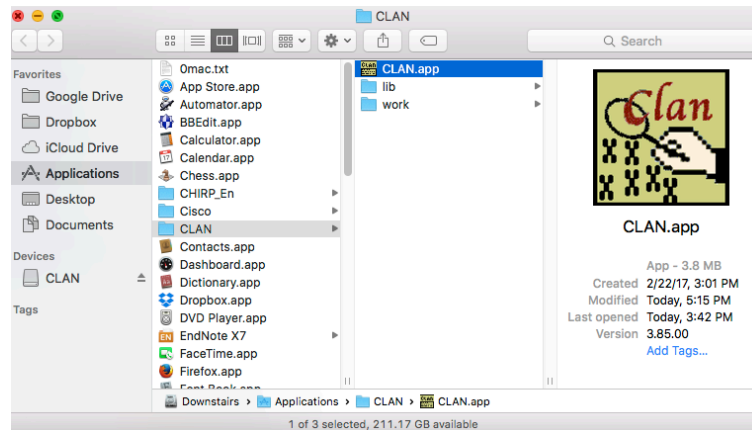
The manual you are reading right now (SLP Manual) should contain most of what you'll need to know, but if you have any questions, the manuals described above are much more comprehensive!

WHERE IS CLAN?

On a PC, the default directory that gets built when CLAN is installed is called c:TalkBank/CLAN. It should be located on your C: drive, perhaps like this:



On a Mac, you should see CLAN in your APPLICATIONS folder. Click the CLAN folder and you will see CLAN.app; click to launch. You should also drag that CLAN.app file to your dock for easy access. Below you see the CLAN folder on a Mac, showing the CLAN application and its subfolders.



The Talkbank/CLAN directory (PCs) and CLAN folder (Macs) have subfolders called LIB, and WORK. To make your life easier, you may want to put things you are working on in the subfolder called WORK. Store your audios and videos there, and the program will easily find them when you try to transcribe things. You can also store your audios and videos anywhere else you choose, but you will just need to make sure to tell the CLAN where they are (the working directory) when you want to run CLAN commands on those files.

AUDIO AND VIDEO FILES

CHAT (the transcription rule system/program assumed by CLAN) allows you to link a transcript directly to audio or video. This speeds up transcription a LOT, and improves accuracy. Also, as will be described later, the audio file can be exported to PRAAT for acoustical analysis of what you've recorded. Clinicians may want to use PRAAT to analyze features such as speech rate, articulation of individual words, prosodic contour or range for looking at prosody in clients, as just one example.

For fluency cases, for example, being able to review a transcript and see/hear what the client was actually doing is immensely helpful in being able to accurately count and locate disfluency behaviors. It also serves as a permanent record of what the client was doing and saying that can be maintained and compared over time by multiple listeners, SLPs, researchers, etc.

There is an amazing function of a linked transcript that we cannot demonstrate in this manual, but you will be able to see it when you follow all the instructions in this short guide. As you play a transcript in CLAN, it will show you the highlighted transcript lines and you will see/hear that segment of the sample at the same time. Because of the audio-video linkage, you can go directly to any given point in the transcript and listen to it.

The most commonly used audio and video formats that are supported by CLAN are .wav, .mov, .mp4, and .mp3. *Although CLAN supports most major audio and video formats, it does not recognize video files written to DVD.* If you wind up needing to transcribe a video from a DVD,

you will have to use a video conversion program such as VLC or MPEG StreamClip to convert the individual video files on the DVD to .mp4 H-264 format.

HOW TO TRANSCRIBE IN CHAT

All CHAT transcripts have certain characteristics: Headers, Main Lines, and Dependent Tiers. Each of these is discussed below.

HEADERS

Headers must be present at the top of every file before you can start transcribing. You actually need to write a tiny bit into a file BEFORE you can start transcribing. To open a new, empty CHAT file, double click the CLAN icon. When the program opens, you will see a command window; this window is used for analysis, not for transcription, so you can close it. Then go to the File menu and click on New. This will give you a blank CHAT file window (default name is newfile.cha) where you can begin transcription. **YOU MUST TRANSCRIBE IN CLAN, NOT IN MS-WORD!!!**

Type in the following headers that are needed to start a file:

@Begin

@Languages: eng (*for English, use other codes for other languages; see the full manual*)

@Participants: CHI Kiddo Child, SLP Casey student Therapist

(first part is the Speaker ID – CHI, SLP)

(second part is the ID, Name, or Specific Role of your speaker – Kiddo, Casey_student)

*(third is the “Standard Role” of your speaker which is one of a fixed set of roles in the CLAN program – e.g., Child, Mother, Therapist)**

(separate multiple speakers by a comma)

(see item #5 below for how to insert this line automatically)

@Media: filename, video (*or audio if the media file is audio; always leave off the filename extension; see item #6 below for how to insert this line automatically*)

@End

*Note: If you have older CHAT files with Standard Roles like Client or Examiner, you may get an error message because those Standard Roles have been eliminated. Below (item #4) we will provide a short list of recommended Speaker IDs and Standard Roles that should meet your purposes. (If you're really interested, you can see the full list of Standard Roles in the CLAN manual -- Section 7 *File Headers, @Participants*).

Save these header lines as a file, and be sure it saves with .cha as the file extension (last 3 letters after the period). Be sure there are NO spaces, periods, commas, non-Roman characters, or other punctuation in the filename before the .cha extension. Good examples are ARsession1.cha, Kid5age24mos.cha, JDbaselineflu.cha etc. It will typically store to whatever you mark as your working directory in CLAN (in the CLAN Commands window).

Basic Rules for Headers

1. Every line must end with a carriage return (oops, showing our ages here, the Enter/Return key; carriage returns are for typewriters; ask your parents what a typewriter is).

2. The first line in the file must be an @Begin header line.

3. The second line in the file must be an @Languages: header line, followed by the TAB key, followed by the 3-letter code for the language. Note: this can be inserted automatically – as explained in #5 below.

4. The third line must be the @Participants header line listing a three-letter code for each speaker, the speaker's name (or ID code), and the speaker's Standard Role (based on a list of roles in the CLAN program). The examples below show the codes and roles we recommend – especially the SLP-Therapist and CHI-CHILD combinations. Notice that in the middle section, where you give a name/ID for the speaker, you need to use an underscore if the name/ID is more than one word. **All of these can also be inserted automatically – as explained next.**

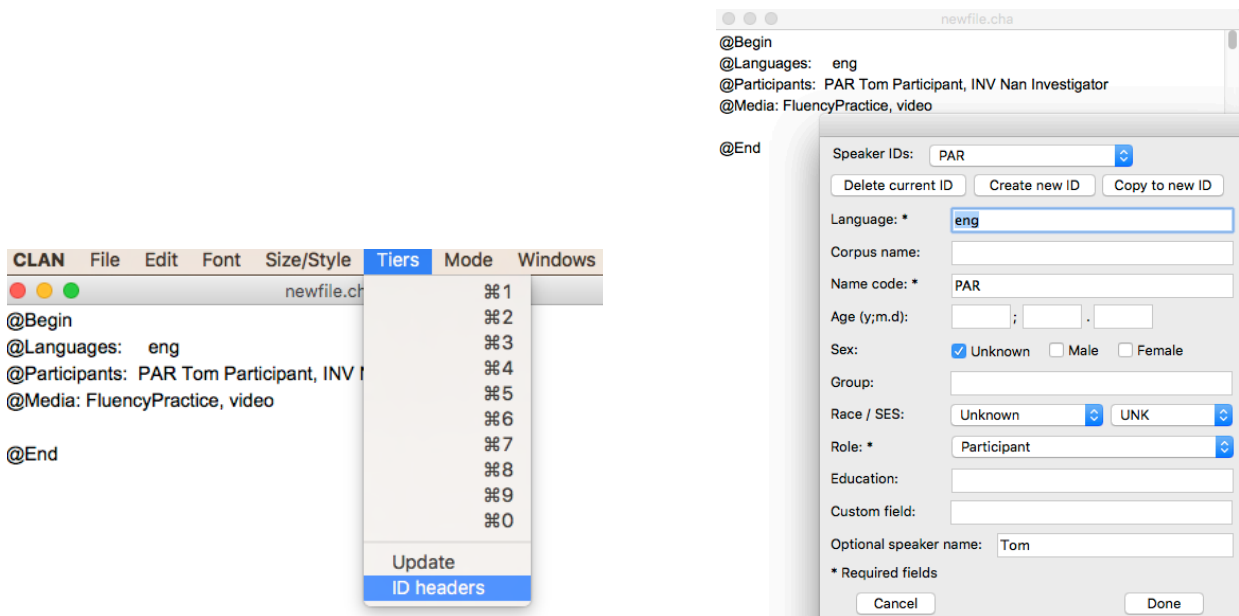
@Participants: CHI 01a_nf Child, MOT 01a_p Mother, SIB 01a_s Sibling

@Participants: ADU 23_esm Adult, INV Steve Investigator

@Participants: PAR umd01f Participant, SLP Oren Therapist

@Participants: BDF Bobby Teenager, RBF Robin Sister

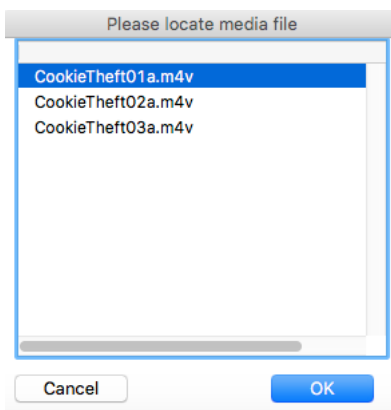
5. The next few lines are @ID headers providing further details for each speaker. YOU DO NOT NEED TO TYPE THESE IN. **These will be inserted automatically for you when you use the drop-down menu under Tiers (CLAN menu, top of screen) and select ID Headers (see screenshot on left below), after you have listed all the participants.** In this menu, you can provide information about the speakers in your file (see screenshot on right). When you click DONE, it automatically inserts these lines, and adds this information. If you have more than one speaker, you must update each speaker's information. You can also use this drop-down menu to add all the information to automatically insert the @Languages and @Participants tiers as well. Just use Create new ID button at the top and fill in all the relevant fields and drop-down menu items.



Note: You can pull this ID headers window down at any time, and it can act as a short set of case history information, such as age, gender, diagnosis, and other notes. When you do it, some of it will then reflect in file headers when you open your CHAT file.

Another note: If you keep track of many clients or participants, many of the CLAN programs will output results in a spreadsheet. If you have updated information in the ID headers, these will print out as column variables, very handy for tracking client progress, examining patterns on your caseload or for study statistics.

6. The @Media header tells the program which audio/video file you will be transcribing. They must match EXACTLY. If the title of the audio/video file does not exactly match the one that you write in the header, then the program will not be able to find the audio. You don't need to put the file extension in the name; CLAN knows that .mp3 or .wav is audio, .mov is video, etc. Your only options for media are audio or video. **To insert the @Media tier automatically**, be sure you saved the CHAT file as explained above and be sure you saved it in the same folder where your media file is located (or move your media file to the folder where you just saved the CHAT file). Then just put your cursor on the next blank line (after @Participants), go to "File" in the CLAN menu and select "Select media file". You should then see a box on your screen with your media file listed in it. Just select the media file you want to transcribe, click OK, and it will appear in the CHAT file automatically in exactly the right format!



7. The last line in the file (after all header tiers and the transcription) must be an @End line.

MAIN LINES

Lines beginning with * indicate what was actually said. These are called "main lines" or "main tiers". There are some basic rules for coding main lines.

1. Each main line should include one and only one utterance. When a speaker produces several utterances in a row, put each one on a new main line.
2. After the asterisk on the main line comes a three-letter code in upper case letters for the participant who was the speaker of the utterance being coded. The three-letter code is followed by a colon and then **a tab (not spaces)**. These can be inserted with shortcuts, which will be described below in the section on How to Transcribe.

3. What was actually said is transcribed starting in the ninth space on the line (using TAB in CLAN brings you automatically to this position). REMINDER: Use a tab, not space bar, after each colon. If you use the shortcuts described above, this will happen automatically.

4. CHAT recognizes a variety of abbreviations for people in the audio/video.

However, to make maximal use of comparison databases being built that will allow you to compare a child or an adult with speech or language disorder to typical performance, WE SUGGEST THAT YOU ONLY USE *CHI FOR CHILD CLIENTS/PARTICIPANTS, AND *PAR FOR ANY ADULT CLIENT/PATIENT/PARTICIPANT. You can choose other three letter codes, but you will lose a lot of the wonderful science you can refer to afterwards if you do that. So, try to listen to this advice!

5. Utterances should end with an “utterance delimiter”. The basic utterance terminators are the period, the exclamation mark, and the question mark. The other common ones are used to mark incomplete or trailing off utterances (see below). If you forget, you will find an error message later when you check your file that says, “missing utterance delimiter”.

6. Use upper case letters only for proper nouns and proper modifiers and the word “I.” CLAN will treat “We” and “we” as different words (unless you pick a CLAN option to ignore case) and MOR will parse “We” as a proper noun instead of a pronoun.

7. Unintelligible words with an unclear phonetic shape should be transcribed as xxx.

8. **IT IS A GOOD IDEA NOT TO GET TOO HUNG UP ON PRONUNCIATION.** CLAN is primarily for language analysis, and the MOR program will try to compute values based on attempts at known words in a language. Even PHON (a wonderful program to which CLAN exports) will require a target representation to automatically compute intended pronunciation for the target word. Thus, we suggest that you do NOT write things such as “dat” for “that”, or “runnin” for “running” – try to write the **intended** word as it would be in the target language (e.g., write “that” and “running”.) Otherwise, the MOR parser will try to look up a word such as “wabbit” (for *rabbit*) and won’t find it, which defeats most of the language analyses in CLAN. If you care about capturing these nuances, you can notate real obvious differences in pronunciation by putting the target word in square brackets following a colon and a space.

*CHI: baba [: baby]

Incomplete words can be written with the omitted material in parentheses – e.g., (be)cause, runnin(g), (a)bout. MOR will use the information in the square brackets and ignore the parentheses around the omitted letters to figure out the part-of-speech and grammar for the target words.

Finally, variations in spelling catenatives (whadaya, dontchya) or other non-lexical forms (hm, uhuh) can be DEADLY to type-token ratios, VOCD, and MOR. There are standard forms for these items later in the manual, where we discuss this potential problem area again.

10. Fluency codes are in APPENDIX 7 of this manual. Fluency is typically coded on the main line, though you can add fluency codes to any dependent tier. (Ask us if you have special needs, such as tagging part of speech that was stuttered.)

For fluency coding, we strongly advise both clinicians and researchers to transcribe the sample for content first and save it. Then, we suggest you **duplicate/copy it with a new filename**, (e.g., duplicate Tom.cha and name it Tomflu.cha) and do your fluency coding on that version. While all the fluency codes we provide pass the CHECK program in CHAT, some have been known to complicate the more sophisticated language analyses, or import/export to PHON or SALT. So, we suggest that you simply save the file with two different filenames, and use one for fluency analysis and the other for language analyses. They will both still be linked to the same video or audio – you don't need two copies of media files.

DEPENDENT TIERS

Dependent tiers are optional additions to the transcript. These tiers begin with the % symbol and can contain codes and commentary regarding what was said in the Main Tier directly above it. There are some basic rules for coding dependent tiers.

1. The % symbol is followed by a three-letter code in lowercase letters for the dependent tier type, such as “pho” for phonology; a colon; and then a tab. The text of the dependent tier begins after the tab.
2. There are a number of pre-suggested dependent tiers for various purposes; consult the CHAT manual for ideas. You can also insert your own, if you want. But this will require you to adjust a file to make the transcript pass CHECK, so this is an *advanced topic* (see the end of this manual for advanced topics). **Some programs insert tiers.** For instance, the MOR program that we will describe for **automatically** analyzing the grammar in your transcript will insert %mor and %gra dependent tiers when you run it.

Here are two samples that illustrate headers, main lines, dependent tiers and basic rules described above.

```
@Begin
@Languages: eng
@Participants: CHI Ross Child, FAT Brian Father
@ID: eng|macwhinney|CHI|2;10.10|||Child|||
@ID: eng|macwhinney|FAT|35;02.|||Child|||
@Media: Ross, audio
*CHI: why isn't Mommy coming?
%com: Mother usually picks Ross up around 4 PM.
*FAT: don't worry.
*FAT: she'll be here soon.
*CHI: good.
@End
```

(Please note that we didn't start each utterance with a capital letter.)

Here is an example from an adult aphasia patient from the Adler corpus in the AphasiaBank database. (It has some other codes in it, but disregard them for now.)

```
@Begin
```

@Languages: eng
@Participants: PAR adler02a Participant, INV mf Investigator
@ID: eng|Adler|PAR|69;09.|male|Conduction|adler02a|Participant||74.9|
@ID: eng|Adler|INV|||mf|Investigator|||
@Media: adler02a, video
@G: Speech
*INV: I am gonna be asking you to do some talking .
*INV: how do you think your speech is these days ?
*PAR: not good .
*PAR: &=laughs &=head:shake not good .
*PAR: &-uh <I'm not> [/] I'm not good at this .
@End

SAVING THE .cha FILE AND STARTING THE FULL TRANSCRIPT

You will not be able to start transcribing using linked audio or video until you have saved the file to your computer with a complete set of header tiers. You can use Command-S or Control-S from your keyboard, or you can go to File in the CLAN menu and select “Save”. Your file will have a .cha extension. Remember, the transcript needs to be saved in the same folder as the audio file.

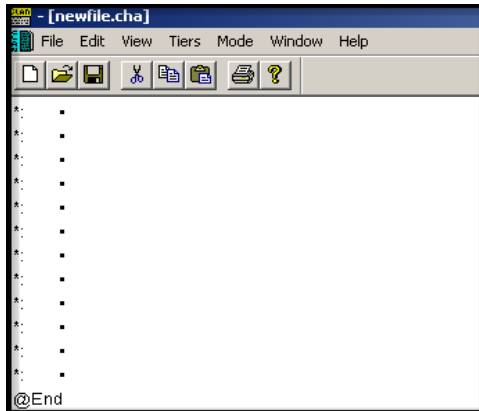
LINKING TEXT TO AUDIO OR VIDEO

This is the truly wonderful aspect of CHAT/CLAN (SALT cannot do this, sorry). Linking speeds up transcription (up to 3 or 4 times, easily, and sometimes even more), increases transcription accuracy, and allows an infinite number of reviews or analyses of your session. Now you are ready to do this. Read all the steps below before starting this part of the process.

- Open CLAN and open your “starter” CHAT file that you just prepared.
- Close the CLAN commands window (Click the X in the upper left of box).
- In the CLAN menu, go to MODE and select “Transcribe audio or video” OR press the F5 key to begin linking, but ...
- **BE PREPARED** BECAUSE THE RECORDING WILL START PLAYING AND YOU NEED TO BE READY TO PUSH THE SPACE BAR AT THE END OF EACH UTTERANCE. Pushing the space bar causes a “bullet” (it looks like a black dot) to appear, and the cursor will move to the next line. Try inserting 10 to 20 bullets to start -- each time you hear a natural break, push the space bar. To stop the process, just push any key or click the mouse.
- If you get an error that the program can't locate the file, make sure: 1) the media file name matches the name listed in the @Media header tier in your CHAT transcript exactly; 2) you have stored the media file and the CHAT transcript in the same folder; 3) the media file name does NOT have any spaces, or any non-English characters in the filename or in the pathway to it; or 4) the audio or video file is REALLY an mp3, .mov, etc. (you cannot make a file into a different file simply by adding .mov to the end of a .wmv file, any more than you can make a .doc into an .xls simply by renaming it!). If all the above steps are correct, consider taking these steps in this order: 1) restart CLAN; 2) restart your computer; 3) re-download CLAN; and 4) make sure Quicktime (Mac) or Windows Media Player (PC) is installed and working (try it on a non-CLAN file). Note that on PCs, file extensions are hidden by default. If the file extension is missing, then you might be tempted to add it. For example, file "Shaggydog.mov" will look on PC

Desktop like "Shaggydog" only, because extension ".mov" is hidden. If you try to correct this by adding extension ".mov" to "Shaggydog" the visible media file name will be "Shaggydog.mov", but the actual name will be "Shaggydog.mov.mov". CLAN will not be able to find this file.

- When you have come to the end of the recording, type @End.



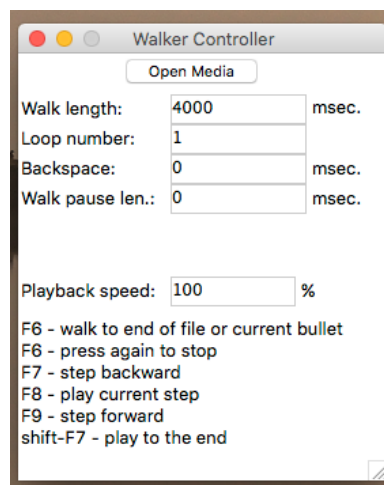
When you're finished linking the file, your transcript will look like this.



- SAVE your file!
- After you have done this, each time you place the cursor by a "bullet" and press F4, you will hear the segment of the recording linked to that line. Transcribe what you hear using the instructions found in this manual. In the section below, we will show you how to make this part even easier!

This looks very strange but is the beginning of something wonderful.

Next, go to the WINDOWS in the CLAN menu and select "Walker Controller" from the drop-down menu. This box will appear on your screen:



Many of the terms in this box may seem like gibberish, but the ones you care about are: *walk length*, *loop number*, and *playback speed*. *Walk length* controls how much you hear when playing back a bullet (if you don't hear everything you should when you play a bullet, your segment is longer than the default, so just increase this number until you seem to hear enough). The *loop number* is how many times the segment will repeat while you type – very critical – pick

the one that makes you happy! *Playback speed* controls how fast the playback occurs. 100% is normal, but if the interaction is going too fast, you can decrease rate (and pretend you are transcribing Darth Vader). Alternatively, if you are falling asleep because the person talks too slowly, you can actually make it go faster by setting playback speed to numbers greater than 100% (and transcribe Munchkins ☺). Keep the Walker Controller window open while you do the next steps below.

Click on the first bulleted line. Then click F6. (*If this doesn't work in Mac*, click on apple icon, then select "System Settings", then type "function keys" in the Search window on the top left, then click on "Function Keys Keyboard". That will take you to where you can change the function keys settings – click on the button next to "Use F1, F2, etc. keys as standard function keys" and click "Done".) You should hear your segment play as many times as you told it to. You are now ready to transcribe. Here is another cool thing: unlike most programs, you can type while having the Walker open – typing won't deactivate the Walker window.

Now skip down to how to transcribe. Unless you are a perfectionist, in which case read the remainder of this section. Don't worry if your "bullets" (that's what the dots are called) don't line up exactly with utterances. They are primarily there to make transcribing easier and more accurate. But the bullets **will** inform any analyses that involve timing information – e.g., words/minute. So, if you are very perfectionistic, you can easily change how much of an utterance is linked to which bullets. But you can even do this at some point later after you've gotten the main job done – the transcription, of course. When and if you choose to do this, follow these steps.

- Put your cursor next to the bullet you want to change. (If you want to add more to the beginning, place the cursor next to the bullet above the one you want to change).
- Press F5. The audio will begin to play where that bullet starts.
- AGAIN, BE READY TO PRESS THE SPACE BAR when you want to end that bullet and begin the next one.
- A highlighted line will show you which line is currently playing.
- When you've finished fixing the bullet(s), click the mouse anywhere and you will immediately stop recording bullets.
- Any bullet you did not change will just stay the way it was.
- Remember, you can always press the F4 function key on any bulleted line to check it.
- HOT TIP: You can also see the numbers inside the bullet by typing ESC-A (press the esc key, release it, and type A) or by going to the CLAN menu to Mode and selecting "Expand bullets". A section on Manual Bullet editing below has more information about tweaking bullets.

HOW TO TRANSCRIBE

There are rules for this, for a reason. The programs expect transcripts to look a certain way, or they won't run, or they will generate gibberish, which makes the whole exercise meaningless. Once you get the hang of it, it's no different from any other activity in which you are asked to follow a format, such as a web address or a telephone number.

Transcribing on Main Lines

Main Lines begin with a *, the 3-letter speaker ID, a colon, and a tab. A fast way to type in that information is to find the shortcuts assigned in the Tiers menu section in the CLAN menu.

Select Tiers in the CLAN menu (as you did above for the ID headers) and notice, for example, that the Speaker ID (e.g., CHI) has been assigned to CTL-1 (or CMD-1 for Mac). If you do not see the Speaker ID assigned to a shortcut, just select Update. Now you'll see that each speaker tier gets assigned to a shortcut and instead of typing *CHI: followed by a tab, you can just type CTL-1. Place your cursor between the asterisk and the colon in your transcript and use the assigned shortcut to insert the info automatically.

Important: How do I decide when one utterance ends and another begins? One approach is for each main line to contain only one utterance or C-unit (conversational unit). An utterance or C-unit is a string of words that:

1. is followed by a pause of 1 second or more;
2. ends with a terminal intonation contour; or
3. has a complete grammatical structure.

Having a complete grammatical structure is a sure mark of an utterance, but often an utterance may have an incomplete structure in short one-word answers. Also, although a following pause can mark an utterance, often another person will talk right away, cancelling the pause.

Reliable separation of a conversation into utterances or C-units is important because measures such as MLU depend on reliable application of these criteria. Chapter 7 of the CHAT manual provides these additional guidelines:

1. Word repetitions should be marked with [/] and distinguished from repeated requests which are separate utterances.
2. Clauses combined with coordinating conjunctions (because, since, etc.) should be distinguished from clauses that use "and" as a conversational continuer.
3. Preposed elements such as vocatives, *yes*, and *well* should be joined to the main utterances, as should postposed elements such as *right?*

Another approach to utterance termination decisions, often used in adult language samples, is based on the rules established by Saffran, Berndt, and Schwartz (1989) using a hierarchy of indices: syntax, intonation, pause, semantics.

After the *, Speaker ID, and tab are entered, type what the speaker says and end every line with some sort of punctuation, usually a period. Be sure to place a space between the end of the last word on a line and the punctuation mark (though if you don't do this, CLAN will add it later, otherwise the CLAN commands don't work correctly).

Here is what a transcript looks like (but pretend you don't see the old "Clinician" Standard Role because it no longer is legal in CLAN):

```

- [4697JK7mosAdult.cha]
File Edit View Tiers Mode Window Help
@Begin
@Languages: en
@Participants: MOT 4697JK7 Mother, CHI 4697JK7 Child, EXP Clinician
@ID: en|NSF|MOT|||||Mother||
@ID: en|NSF|CHI|0;7.||||Child||
@ID: en|NSF|EXP|||||Clinician||
@Coder: Anna_Synnestvedt
*EXP: okay it's been fifteen minutes .
*EXP: so that part is done .
*EXP: how did the toys taste ?
*EXP: good ?
*EXP: so then we just have a few follow up questions .
*EXP: I'm not going to write down the answers because I'll just let the
microphone pick it up .
*EXP: um the first question is which of the toys seemed to interest your
child the most ?
*MOT: uh the plastic # food .
*MOT: <related> [/] I mean you know this bin .
*EXP: okay .
*EXP: and which is a good toy for his personality ?
*MOT: um that is a weird question .
*EXP: yeah some of these are kind of not xxx questions .

```

Punctuation

There must be a delimiter (punctuation mark) at the end of every utterance. Most utterances will end in a period (.) question mark (?) or exclamation point (!). Be sure to put a space between the end of the last word and the punctuation mark.

No capital letters should be used except for “I” and proper nouns. There should be no punctuation inside of utterances (e.g., commas, semicolons, etc.) except for apostrophes. Some utterances will require different delimiters. These are described below.

Trailing Off +...

The trailing off or incompleteness marker is used for incomplete, but not interrupted, utterance. Trailing off occurs when speakers shift attention away from what they are saying, can't find the words to finish their thought, or sometimes even forgetting what they were going to say.

- *CHI: smells good enough for +...
- *MOT: what were you saying ?

If the speaker does not really get a chance to trail off before being interrupted by another speaker, then use the interruption marker +/. rather than the incompleteness symbol. Do not use the incompleteness marker to indicate simple pausing, repetition, or retracing.

Trailing Off of a Question +..?

If the utterance that is being trailed off has the shape of a question, then this symbol should be used.

*EXP: so do you have any of these toys at home or +..?

Interruption +/.

This symbol is used for an utterance that is incomplete because one speaker is interrupted by another speaker.

*EXP: what did you do +/.

*CHI: mommy .

*EXP: +, with your spoon .

Self Interruption +//.

This symbol is used for an utterance that is incomplete because a speaker interrupted him/herself.

*MOT: well we haven't started to +//.

*MOT: Alex put that down !

Retracing Without Correction [/] (also in fluency codes, Appendix 7)

The [/] symbol is used in those cases when a speaker begins to say something, stops and then repeats the earlier material without change. The material being retraced is enclosed in angle brackets. In a retracing without correction, it is necessarily the case that the material in angle brackets is the same as the material immediately following the [/] symbol. Here is an example of this:

*CHI: <I wanted> [/] I wanted to invite Margie .

If only one word is repeated, the angle brackets are not necessary. When the angle brackets are not used before the repeating symbol [/], CLAN assumes that only the last word was repeated.

If there are pauses (longer than 1 second) and fillers between the initial material and the retracing, they should be placed after the retracing symbol, as in:

*CHI: it's [/] (.) &-um (.) it's [/] it's (.) a &-um (.) dog .

When a word or group of words is repeated several times with no fillers, all of the repetitions except for the last are placed into a single retracing with angle brackets.

Retracing with Correction [//] (also in fluency codes -- Appendix 7)

This symbol is used when a speaker starts to say something, stops, repeats the basic phrase, but changes any part of the phrase. Usually, the correction moves closer to the standard form, but sometimes it moves away from it. The material being retraced is enclosed in angle brackets. In retracing with correction, it is necessarily true that the material in the angle brackets is different from what follows the retracing symbol. Here is an example of this:

*CHI: <I wanted> [//] &-uh I thought I wanted to invite Margie .

Retracing with correction can combine with retracing without correction, as in this example:

*CHI: <the fish is> [//] the [/] the fish are swimming .

Sometimes retracings can become quite complex and lengthy. This is particularly true in speakers with fluency and language disorders. It is important not to underestimate the extent to which retracing goes on in such transcripts.

For most CLAN programs, information associated with repetitions and corrections is EXCLUDED by default, though you can run commands with the option of including material from repetitions and revisions if you want. This will be discussed later. But MOST IMPORTANTLY, the MOR command that parses the utterance will ignore material that is repeated and retraced. In this way, MOR provides a morphosyntactic analysis of speaker's ultimate final version of the utterance (in the example above, "the fish are swimming").

CLAN allows you to code for anything you want – e.g., errors, main events, speech acts. See the CLAN manual if you want to code for other aspects. Once you learn how to code a few things, you will be able to fiddle around with it. CLAN also allows you to mark utterances that you may want to exclude from CLAN commands (see the heading below about excluding utterances).

Overlaps

Sometimes two people will speak at the same time. In most cases, you don't care, and sometimes, for clinical interactions, you don't really want to code one person just saying, "uh-huh" over and over again. Or you can put the other person on the next line, even if they were talking at the same time. But if you care, here are a few ways to code the overlap.

1. Interposed word &*

It is most convenient and advantageous to mark short verbal encouragers, such as "yeah" or "mhm", within the main speaker's utterance without breaking it up. This is marked using &* followed by the speaker's 3-letter ID, a colon, and then the interposed word.

*PAR: it was really difficult &*INV:mhm when all of that was happening.

2. Lazy overlap +<

The "lazy overlap" symbol shows that the one speaker (PAR) spoke while the the other speaker (INV) was speaking, but it doesn't show at what point in the Investigator's sentence the Participant started talking.

*INV: how did you communicate with her ?

*PAR: +< I just kept talking .

3. Overlap follows [>]

The "overlap follows" symbol indicates that the text enclosed in angle brackets is being said at the same time as the following speaker's bracketed speech. This code must be used in combination with the "overlap precedes" symbol, as in this example:

*INV: how did you communicate <with her> [>] ?

*PAR: <I just kept talking> [<] .

You can insert spaces to make a visual indication of where the overlap occurs. CLAN will ignore the spaces.

- *INV: how did you communicate <with her> [>] ?
- *PAR: <I just kept talking> [<] .

FLUENCY codes: *typical disfluency* (please see section on stuttering codes in Appendix 7 for specific codes and programs to transcribe and analyze stuttering)

Unfilled pauses

You can indicate short, medium, and long pauses by using (.), (..), and (...), respectively.

Filled pauses

When words are used as fillers, they should be marked with &- to show that they should not be included in lexical analyses (i.e., MLU or TTR analyses) *and SHOULD be counted in disfluency analyses*. Examples:

- &-like
- &-you_ know (note that we pulled the *you* and *know* together with an underscore so it gets counted as one lexical item)
- &-and, when it seems to be just a staller ...

NON-lexical fillers/disfluencies (um, uh, er, etc.) should be transcribed as &-um, &-uh, etc. Non-lexical fillers can also be transcribed with the ampersand only, but we recommend the ampersand-dash combination moving forward. There is a fine line between interjections (e.g., “oh”), communicative turn sounds, and these. Use the ampersand-dash combo (&-) to indicate what you think is a disfluency, as opposed to something that has a communicative function.

Quotation on Next Line +”/.

During story reading and similar activities, a great deal of talk may involve direct quotation. To mark off this material as quoted, special symbols are used.

- *CHI: and then the little bear said +”/.
- *CHI: +” please give me all of your honey .
- *CHI: +” if you do, I'll carry you on my back .

The use of the +”/.

 symbol is linked to the use of the +” symbol. Breaking up quoted material in this way allows us to maintain the rule that each separate utterance should be on a separate line. You should only do this if the client is reading or quoting. If the child makes up utterances for a character, it’s not all that important to mark it as quoted. You can just type what you hear. But some clinical and research questions like to notice when children or adults use another person’s voice, because that is more sophisticated than simply narrating.

Quotation Precedes +”

This symbol is used when the material being directly quoted precedes the main clause.

- *CHI: +” please give me all of your honey .

*CHI: the little bear said +”.

Single quoted words can use the @q marker, as in:

*CHI: and the boy said shh@q .

Multiple words that should hang together

Some words need to be treated as ONE word (e.g., “you know”, “merry go round”, “patty cake”, “kind of”, or a person’s name, such as Mister Spock, or Brian MacWhinney). When words link together to form a frozen phrase, link them with an underscore (you_know, patty_cake, Nan_Bernstein_Ratner). There are also complicated rules for using + to link generative compounds. If you get ? on a %mor tier parse, this could be the reason, and you can play with typing it differently (pulling it together, pull it apart with underscore or +) to see if the program is happier. There shouldn’t be too much of this in most files.

Other Coding Conventions: ERRORS!

Most of the time, people don’t speak in perfect sentences. Here are some coding conventions for some of the most common deviations.

If you are reading this, you are probably a clinician ☺. If you wind up with a file that doesn’t have some errors, we’d be amazed. That’s WHY you do language sampling! When you find an error in use, mark it in the following way:

CHI: he had two mouses [: mice] [] .

You can just use the [*] or you can code errors more finely (the CHAT manual has lots of examples), but the main point here is that you noted an error (which will be computed in KidEVAL and EVAL), took away a DSS sentence point, and you used the target replacement [: mice] to let the MOR program know what the person was trying to say.

Notes:

- 1) ERRORS are meant to reflect problems in the grammar or word use that are clearly unacceptable in the speaker's dialect. Please remember that ellipses (shortened conversational turns, such as short responses that are not full sentences) are NOT ungrammatical and occur frequently in normal interaction. Thus, do not use this error code [*] for utterances lacking a subject and verb, prepositional phrases, etc. because of ellipsis (INV: what do you want? CHI: cookies; INV: where are you going? CHI: home.)
- 2) However, if a speaker's elliptical response has an error, you would code it as such (e.g., CHI: two cookie [*]; CHI: to [*] home.)
- 3) Finally, all transcribers should be mindful that some utterances that appear to be ungrammatical in Mainstream English (or other languages), may be quite grammatical within the target speaker's dialect. Thus, a speaker of AAE who says: CHI: he going to the store would NOT receive an error code; the same utterance spoken by a child of mainstream American English would likely be flagged with [*].

Missing words: 0

You can code an omitted word or part of speech using the zero symbol as shown below. Caution is urged, though, as coding of word omissions is a difficult and unreliable process. Items marked as omitted are not included in the MLU count.

*CHI: 0does he like it? (this corresponds to the “gloss” -- what you heard – “he like it?”)

*PAR: 0mod he like it? (indicating that the modal “does” or “did” was omitted from “he like it?”)

Phonological Fragments &+fr

If there is a fragment of a word, use the &+ symbol and then type out the fragment.

*CHI: he had a &+fr friend.

*CHI: I really wanted to &+vi visit the zoo.

In the first example, the child began to say “friend” but stopped part way through the word. Notice that a lot of disfluencies are being tagged with the ampersand. The ampersand shows up in the Appendix under stuttering codes as well, for obvious reasons.

Unintelligible words xxx

If you cannot understand a word or phrase contained within an utterance, code it as xxx. Before doing so, try to listen a couple times to figure it out. If you think you can make out the sounds, you can use yyy, and then use a %pho tier to put these sounds in, but that’s beyond the scope of this manual.

Pauses (.)

If the speaker pauses for a short duration, use the (.) symbol. This symbol is not necessary if the pause occurs between utterances, even if the bullet makes it sound like it happens at the beginning of the utterance. Pauses that occur between utterances do not need to be coded. Medium-length or long pauses can be marked with (..) and (...), respectively.

Babbling and Jargon (from kids or patients)

Sometimes you get sounds, but you have no English word to “gloss”. You can import fonts and insert phonetic symbols (or characters from other languages, such as Chinese). See the CHILDES website for help with this if you need it. For many situations, it may be best to mark it as unintelligible using yyy (and trying to gloss what you heard) or xxx (no real best guess as to how to transcribe):

*CHI: yyy [=! dada] .

*CHI: xxx [=! vocalizes/laughs/whines, etc] .

*PAR: I probably got xxx and things like that .

Tables for Other Communicative Behaviors and Words

Now we have a bunch of Tables for various kinds of words. WHY? Because so many analyses will compare word types and tokens, having a system (and checking your spelling) is extremely

important, or you will wind up crediting the speakers with more words than they really say. Sloppiness hurts your end result. A lot.

Simple Events

Often you will want to mark some non-verbal activity that a child produces at a particular juncture. You can do this using the `&=text` format. Examples include `&=moans`, `&=laughs`, or `&=coughs`. This type of coding can be extended to mark imitations as in `&=imit:plane` if the child makes a plane noise, gestures as in `&=ges:frustration`, or other actions such as `&=moves:doll`, `&=shows:pictures`, `&=points:picture`, or `&=opens:mouth`. The full list of these codes is given in section 7.9 of the CHAT manual.

Exclamations, Interactional Markers and Other Funny Words

Conversational speech has hundreds of forms that are fairly standard, but not spelled in conventional ways. For example, should one spell the word for realizing something as *ahhah* or *ahah* or even *ahaah*? For the hundreds of forms like this, the English MOR has selected one of the forms as standard. For example, it allows *okay*, but not *ok*; use *uhhuh* for the common *yes* variation and *uhuh* for *no* (though several other spellings are accepted as well). To find out the right way to spell each of these forms, you can consult the file `0allwords.cdc` in the `ENG` folder (explained later) of the MOR lexicon. That file has all the words recognized by English MOR in alphabetical order. After a while, you will get used to these conventions and will only need to consult this file on occasion. If you want to focus specifically on learning the forms for exclamations and communicators, then you can look at the `co.cut` file in the `/eng/lex` folder.

Shortened Words

Sometimes the word will be shortened but still intelligible. You can put the deleted part of the word in parentheses. This will enable CLAN to do analyses on the word, and will help you later if you are trying to count syllables, or need to know exactly how the child said the word. The following table lists some examples.

Table 3: Shortenings

Examples of Shortenings			
(a)bout	don('t)	(h)is	(re)frigerator
an(d)	(e)nough	(h)issself	(re)member
(a)n(d)	(e)spress(o)	-in(g)	sec(ond)
(a)fraid	(e)spresso	nothin(g)	s(up)pose
(a)gain	(es)presso	(i)n	(th)e
(a)nother	(ex)cept	(in)stead	(th)em
(a)round	(ex)cuse	Jag(uar)	(th)emselves
ave(nue)	(ex)cused	lib(r)ary	(th)ere
(a)way	(e)xcuse	Mass(achusetts)	(th)ese
(be)cause	(e)xcused	micro(phone)	(th)ey
(be)fore	(h)e	(pa)jamas	(to)gether
(be)hind	(h)er	(o)k	(to)mato
b(e)long	(h)ere	o(v)er	(to)morrow
b(e)longs	(h)erself	(po)tato	(to)night

Cad(illac)	(h)im	prob(ab)ly	(un)til
doc(tor)	(h)imself	(re)corder	wan(t)

Assimilations

Sometimes two or more words will blend together and sound completely different from the individual words that created them. In such cases you can type these assimilations into the main tier to reflect this change:

Table 4: Assimilations

Nonstandard	Standard	Nonstandard	Standard
coulda(ve)	could have	mighta	might have
dunno	don't know	need(t)a	need to
dyou	do you	oughta	ought to
gimme	give me	posta	supposed to
gonna	going to	shoulda(ve)	should have
gotta	got to	sorta	sort of
hadta	had to	sorta	sort of
hasta	has to	wanna	want to
hafta	have to	wassup	what's up
kinda	kind of	whaddya	what did you
lemme	let me	whyntcha	why didn't you
lotsa	lots of		

We've added: gotchya (ns) got you (s)

Excluding utterances from analysis

Utterances: While transcribing, you may want to use the [+ exc] post-code to mark utterances that you would want to exclude from analysis. These may include questions or comments about the task, such as *"I can't think of the word"*, *"this is hard"*, *"okay, I'm done"*.

- *PAR: mom is washing dishes.
- *PAR: I think I saw this before. [+ exc]
- *PAR: he's gonna fall off the stool.

These CLAN commands will automatically ignore utterances with the [+ exc] post-codes: EVAL, KIDEVAL, C-NNLA, C-QPA, SCRIPT. For all other CLAN commands, you can add this to the command line to exclude utterances with the [+ exc] post-code: **-s"[+ exc]** .

Transcribing Dependent Tiers

Why would you want to add tiers to your transcript? Well, common reasons are to mark for speech act intents, comments about what is going on, whether the client managed to include a narrative component in his story, etc. Dependent tier lines begin with the % symbol. This symbol is followed by a three-letter code in lowercase letters for the dependent tier type, such as "nar" for narrative code or "com" for a comment. The three-letter code is followed by a colon and then a tab. The text of the dependent tier begins after the tab. A full list of pre-arranged, conventional tiers is in the CHAT manual. Consult it before making one up!

Comments %com

If you wish to note something odd or of interest in the recording, create a dependent tier for comments.

*CHI: Alpha had a frog .

*CHI: then when Asa wake up he could [//] looked for his frog everywhere .

%com: the character's name changes from Alpha to Asa and back again .

Just like with main lines, there should always be a colon after the three-letter code and a tab after the colon before the actual comment begins. The same rules of punctuation and capital letters apply to these lines.

Why would you do this? It often helps to annotate something you noticed during the session that you are likely to forget later, that provides useful information about the utterance or interaction.

Note: If a significant portion of the file is unanalyzable, add a warning header at the top of the file, rather than comments all over the place; likewise, if there is a generic observation that affects the whole file, put it up at the top as well.

@Warning: The first 15 minutes of this file had background noise.

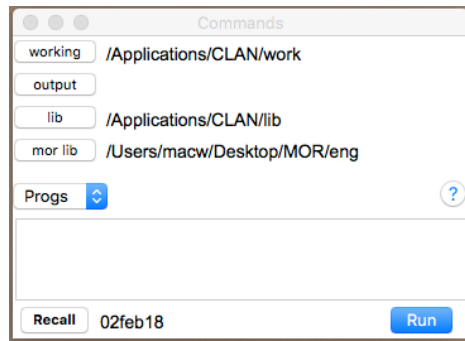
MOVING ON TO THE GOOD STUFF: CLAN ANALYSES

CHECK

Before you run any real analyses, it is important to make sure the transcripts are set up correctly or else the analyses won't run. To do this, you need to run a program called CHECK on the transcripts.

One easy way to run check is to use the **ESC-L function** while you're transcribing or once you've finished. With the transcript open, simply press the escape key, release it, then type L. You'll see a message at the bottom of the screen showing you where (by line number) you have an error and what the problem is. If you don't see line numbers in your transcript, go to Mode in the CLAN Menu and select "Show line numbers". Keep using ESC-L to check until you get the Success! message at the bottom of the page. Then SAVE your file!

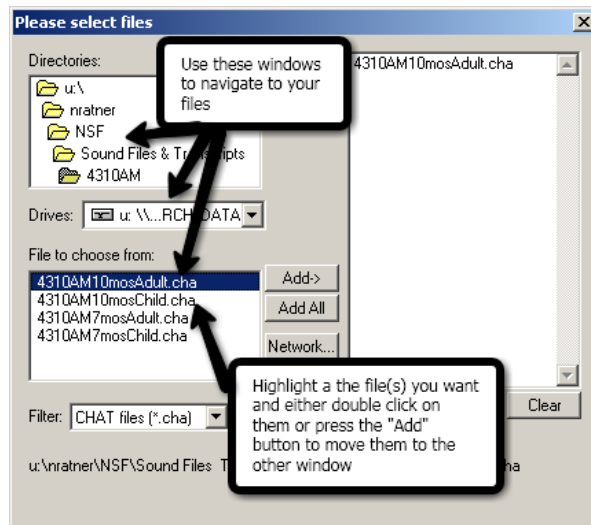
Another way to run CHECK, especially if you want to check multiple files, is by running a CLAN command. If you're in CLAN and you don't see the Command window, type CMD-D or CTL-D or go to the Windows in the CLAN menu and select "Commands". Otherwise, open CLAN and the Commands window should automatically open.



There are several different ways to run these commands, so the example here is just one way. If you have other ways that work (that you learned or figured out), great. Also, the screenshots in this manual are mostly from a PC and will look different than what you see on a Mac. The screencasts, on the other hand, use a Mac, and there's a 3-minute video called "check-1" (checking a transcript for errors) so you can have a look there for additional guidance: <https://talkbank.org/screencasts/>

In the commands window, type "check" followed by a space. Once you type in a command in the Commands window followed by a space, a 'File in' button will appear above the window.

To choose the transcripts on which you'd like to run the analysis, click the "File in" button. A new window will open, and from there you navigate to where your transcripts are saved.



You can run CHECK and any of the following analyses on more than one transcript at a time if you so choose. Once you have selected the file(s) you want to analyze and have moved them into the window on the right, press "Done." This will return you to the Commands window, and an "at" sign (@) will appear next to your command. Then press "Run" in the Commands window or simply hit Return on your keyboard and the analysis will run on its own.

Results of most CLAN analyses are sent to a CLAN Output page that pops up on the computer screen (except for a few analyses we will explain further). You can save the Output page using the File → Save from the CLAN menu, or you can cut and paste the output to a word file and

save it. The CHECK command will produce errors in formatting of the CHAT transcript. Here's an example of one of the types of errors you might see listed on your Output page:

```
*** File "/Users/macw/Desktop/CHATfiles/S002_intake copy.cha": line 10.  
*FAT:who's that •  
Missing either TAB or SPACE character.(3)
```

This gives you the line number, highlights where the error is, and explains the error. **HOT TIP:** If you triple click on the line with the 3 asterisks, you will be magically taken directly to the correct line in the transcript to fix your error! Fix and save. Keep going through your list of errors, fixing, and saving, and then run CHECK again.

Typical CHECK gripes:

Wrong format (spaces, not tabs). Confusing spaces and tabs between the Speaker ID and the line of dialogue is such a common mistake, there is actually a program to fix this. In the command window, type CHSTRING +q and then input the offending file. It will fix all the bad lines at once.

Missing headers. Typically, this problem occurs when you didn't format appropriately, or didn't include a speaker you then transcribed, or invented a role not in the approved list.

Missing "utterance delimiters". Sometimes it's easy to confuse a bullet for a period and forget to put some in. If you get a lot of these errors on a check run, there is a program to fix the situation: In the CLAN window, type DELIM and put in the offending file. DELIM will insert periods (but only periods) at the ends of utterances lacking final delimiters.

Forgetting to put @End at the end of the file ☹

CLAN actually runs two CHECKs when you run the CHECK command. Be sure to keep fixing the files and re-running CHECK until you get a CLAN Output page with no errors.

****Note: You can actually run most programs on files that CHECK is having some problems with; it may affect accuracy of output somewhat, but probably not in a clinically relevant way.** If a program won't run, it's because something is pretty wrong with your file, and you need to figure it out and fix it. *If you still have trouble, either contact chibolts (see listserv instructions) or e-mail one of us.*

CLINICAL ANALYSES

In the Fall of 2012, CLAN developed a super-useful utility called KIDEVAL (there is also EVAL for adult aphasia samples and now FLUCALC for computing fluency-related behaviors). These programs send output automatically to an Excel spreadsheet with a mother lode of analysis results, such as mean length of utterance in morphemes (MLU-M), mean length of utterance in words (MLU-W), type-token ratio (TTR), clause density, and number of major morphemes observed. It will even output multiple transcript analyses into the same Excel file.

There is only one minor thing you have to do first to use KIDEVAL and EVAL. You have to run a command called MOR on the transcript. MOR is short for "morphology" and will

automatically insert a %mor tier and a %gra tier, so that you don't have to figure out where morpheme boundaries are, or the syntactic construction of the sentence (e.g., parts-of-speech). Don't worry, your linguistics classes will still come in handy, but this saves a lot of work, and is much more accurate than doing stuff by hand.

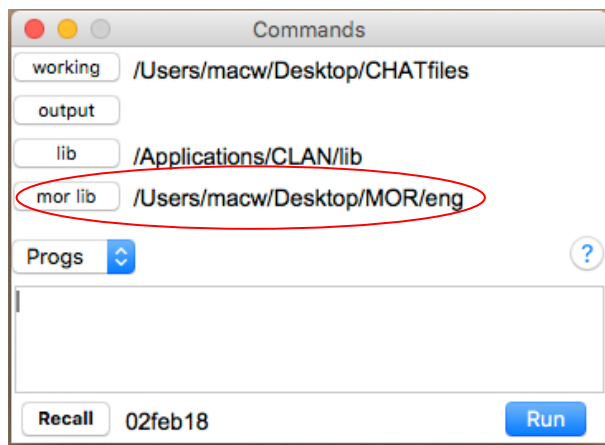
Preparing to run MOR

To run MOR for your language, you need to download the MOR grammar. It is super-easy to do this by selecting the option under "File" in the CLAN menu called "Get MOR grammar". For English data, select "eng". The MOR grammar will automatically download to the Desktop and the CLAN Command window will automatically set your MOR LIB directory to that location to find it. You can move the MOR folder somewhere else, but then be sure to click on the MOR LIB button in the CLAN Commands window and navigate to where the MOR folder is located.

You can also download MOR for 10 other languages. MOR has the best coverage and accuracy for English, but the others are not bad, and all are much better than trying to do morphological analysis by hand.

You can even analyze bilingual samples ☺; however, if you are going to analyze a single file with more than one language, that is an advanced topic, and you should contact Shelley Brundage [brundage@gwu.edu].

If all has worked well, you should see the following in your mor lib(rary): ENG or the language of your choice should be the last element in the path:



Okay, you are ready to run MOR. You can use the drop-down menu in the "Progs" box to select "mor" from the list of CLAN commands. Or you can type MOR followed by a space in the command window. Then you can click on the "File in" button that will appear and select the file(s) you want to analyze. Or you can type the filename (e.g., flu01.cha) or a list of filenames into the command line:

```
mor flu01.cha ctl01.cha
```

OR

```
mor *.cha (if you want to run MOR on all the files in the working directory folder)
```

Note: The working directory button is the first one in the CLAN Command window. BE SURE that the working directory is set to the folder where your files are located. Again, just click on the button and navigate to your folder.

The MOR command actually runs 5 programs sequentially to insert morphology into the transcripts. In the first step, every word in your transcript is looked up in a huuuuuuge dictionary that MOR uses. Then, because many words in a language are ambiguous, a second program, POST, uses very sophisticated algorithms to assign the best possible parse to your transcript. These parsers are very accurate (particularly when compared to harried researchers, students and clinicians ☺). Then it runs POSTMORTEM, which cleans up some further details on the %mor line. POSTMORTEM is particularly important for German (deu). Fourth, it runs MEGRASP which adds a %gra tier to the transcript showing grammatical relations. Last, it runs a command to check if all the words in your file can be identified by the MOR command (for experienced users, this is the mor +xb command). If your transcript has one or more unrecognized lexical items (e.g., "staet" for "start"), you will see a message in red at the bottom of the CLAN Output page telling you how many of the total words in your transcript(s) were not recognized. The message includes clear information about where those words are listed and what you should do about it. HOT TIP – below the message, it the name of an Output file with a .cex extension. If you triple click on that line, the .cex file with the list of unrecognized words will open. And HOT TIP #2, within that .cex file, if you triple click on any line that begins with "File", you will magically be taken to the spot in the transcript that needs an edit. Fix the typo or add the target word right after an actual error production, as in stissers [: sisters], SAVE YOUR FILE again, and RE-RUN the MOR command so your file gets fully parsed properly. If you find a lot of unrecognized words that you know ARE real words, let the folks at CMU know and they will add them to the lexicon. If your transcript has no errors, you will simply not see anything about unrecognized words after MOR finishes running all the programs.

Note: If you want to get rid of the %gra tier that appears automatically after running the MOR command, you can run this command to remove it from your transcripts: `trim -t%gra +1 filename.cha` (or `*.cha` for a group of files, or use the *File In* button).

Finally, it is good to CHECK the file again. If there's a problem, you really should go back and make a fix. This is typically that POST didn't disambiguate a line and there is still a ^ separating grammatical options. If there are only a few of these, you can use the escape-2 command to choose the right options by hand.



KIDEVAL: The answer to the SLP's dreams for child language sampling.

Once you have your file automatically MOR-ed and you checked to make sure there aren't lots of ? marks under perfectly good words, just type KIDEVAL in the CLAN Command window followed by a space (or select "kideval" from the "Progs" drop down menu under the "mor lib" button. Next, click the Option button and select which file(s) you want to analyze (navigate as needed, use the Add button, and click Done when finished).

Now you have to decide whether you want to compare your file(s) to the database or not. If you have already downloaded the KIDEVAL database, you have the option to either compare the child's sample to this database by age and gender, or not compare. Currently, we should not compare anything other than adult-child play with a child under the age of 6 who speaks a mainstream dialect of English to the database. Note: Clinicians are advised to use KidEval reference values in conjunction with other language performance indicators (e.g., standardized test scores). A child's performance can also be benchmarked by prior research findings for individual LSA measures (such as MLU, NDW, etc.) summarized in the SLP Guide to CLAN Appendices for further information.

If you choose “do not compare”, you then need to choose which speaker you are interested in analyzing and what language. Press OK and you will see the elements of the command automatically written in the CLAN Command window. Press “Run” and VOILA! (Note: If you want to learn more about the “compare to database option”, have a look at the KIDEVAL-1 screencast at <https://talkbank.org/screencasts/> .

The output from KIDEVAL and EVAL is DIFFERENT from what you typically get by running CLAN commands. It does not put the data in a CLAN Output file on your screen. It produces an Excel spreadsheet that you open in Excel (the last line on the CLAN Output page tells you the name of the .xls file, but again, you can triple click on this “Output file” line and go directly to the Excel spreadsheet. NOTE: If your Excel is prior to version 15.3, the spreadsheet will not look right if you do the triple click technique. You will need to open Excel, go to the menu to “File”, select “Open”, navigate to the .xls file you want to open and click “Get Data”. Then, in the Text Import Wizard window, choose “Delimited” and click “Next”, (UNclick “Tab” if it's selected), select “Comma” and click “Next” and then “Finish”.

Finally, you should see a lovely spreadsheet, with all sorts of info that used to take hours to compute (if people could figure it out at all, or do the linguistics or math right).

This is what an output from KIDEVAL looks like:

D3	CHI																												
	L	N	O	S	T	U	V	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU					
1	Code	Total Utts	MLU	Worc	MLU	Morc	types	tokens	TTR	Clause/Ut	DSS	VOCD	D_c	PROG	in	on	7	33	8	"poss "	cop "	det	"-PAST	"-3S	"&3S	aux "	"cop "	"aux "	
2	MOT	236	3.449	3.958	221	815	0.271	0.564	7.65	92.74	14	10	3	7	33	8	0	49	0	4	40	16	0	35					
3	CHI	134	2.235	2.485	86	325	0.265	0.493	4.74	24.13	2	3	0	16	4	0	0	22	0	11	20	20	0	1					
4																													
5																													
6																													
7																													
8																													

This is a big display, so here is a fast guide to what you are looking at (we have actually hidden some columns you would get if ran the program, just to fit the most clinically useful stuff into the screen shot.)

TO SAVE THIS SPREADSHEET AS A REAL EXCEL SPREADSHEET, you need to go to the menu to FILE, SAVE as, give it a name, and then scroll up to select the xlsx format, decide

where you want it saved, and press save. Otherwise, it's just in text format, even though it looks like a legitimate spreadsheet.

Each person's data goes on a single row (there was both a Mother and Child in this file, and we looked at both of them, just for fun). Going across the top, you see these measures, in order.

- Total number of utterances the speaker produced.
- Mean Length of Utterance (MLU) in words (this is strictly not MLU, which is measured in morphemes, but is very good for looking at other languages).
- MLU in morphemes (the traditional measure).
- MLU and MLU-W have recently updated “norms” for children across childhood (2;6-8;11). See Rice et al., (2010) Mean Length of Utterance levels in 6-month intervals for children 3 to 9 years with and without language impairments. *Journal of Speech, Language, and Hearing Research*, 53, 333–349. (See **Appendix 1**.)
- Types: the number of different words used in the sample (there are options in CLAN for lumping word roots and inflections, but this one takes each word version separately).
- Tokens -- total number of all words in the sample.
- TTR -- Type-Token ratio, a measure of lexical diversity. This number goes from .001 to 1.0; a low value indicates a lot of repetition (as if someone said “yada yada” 100 times and nothing else); a high value means each word in the sample was different. You actually can't get numbers too close to 1.0 and still have grammar – think about it.
- There are a number of references showing typical TTR values for children across early development. One (Miller et al., 2005) is included in **Appendix 2**.
- Number of different words in 100 words (NDW): this value has evolving norms. Stay tuned; Some are provided in **Appendix 3**.
- Clause/utt -- clausal density, the average number of clauses per utterance.
- DSS: Developmental Sentence Score (Lee, 1974). This normed value is a classic in clinical child language sample analysis. The manual refers you to source material and norms (we have pdf files of the scoring procedures and norms if you'd like them; due to copyright, we provide them separately). REMINDER: DSS gives a sentence point to anything you didn't place an error code on. If you didn't mark utterances as ungrammatical when you saw them, this value will be somewhat higher than it should be when comparing norms. See some reference norms in **Appendix 4**.
- IPSyn: Index of Productive Syntax (Scarborough, 1990). See sources and reference values for IPSYN in **Appendix 5**. Several of the items in IPSyn are computed on the basis of the %gra line and this means that your input transcripts must have a %gra line, if you want to compute IPSyn in KIDEVAL. However, when you run MOR, this also runs

MEGRASP automatically to compute a %gra line, so you should be able to do all of this automatically.

- VocD: Vocabulary diversity. This is a twist on TTR, but supposedly more reliable when looking at files that vary substantially in length. The CLAN manual has lots of information about this measure. It is used by many researchers instead of TTR. *For fluency researchers, we have serious concerns about all other vocabulary diversity measures, because PWS tend to talk so much less in most sampling sessions.* Some reference values for VocD are provided in **Appendix 6**.
- *-PROG through ~aux|* : These columns provide “Brown’s morphemes”, in developmental order, as they were seen in the sample (and are listed below). These numbers are either RAW COUNTS, or percentages, depending upon which *option* for KIDEVAL you pick¹ and add to the command line.²
 - -PRESP the present participle *-ing*, as in *swimming*.
 - in the preposition *in*, as in *the cheese is in the bag*.
 - on the preposition *on*, as in *put it on*.
 - -PL the regular plural, as in *dogs*.
 - &PAST the irregular past, as in *fell*.
 - ~poss the possessive clitic, as in *John’s*
 - cop the uncontractible copula as in *Is Meg nice? Meg is*.
 - det:art the determiner, as in *the ball*.
 - -PAST the regular past, as in *jumped*.
 - -3S the regular third person singular present, as in *runs*.
 - &3S the irregular third person singular present, as in *does* or *has*.
 - aux the uncontractible auxiliary, as in *Is John running? Yes he is*.
 - ~cop the contracted cliticized copula, as in *Meg’s tall*.
 - ~aux the contracted cliticized auxiliary, as in *John’s going*.

DSS and IPSYN

After running the KidEval utility, a clinician may wish to know WHICH utterance components are present or absent in a child's sample. Even if a sample is not long enough to generate an IPSyn or DSS total, it is possible to run DSS or IPSyn as **separate commands** to see scoring details, which can be very informative in therapy goal setting. Because DSS and the IPSyn were developed for English grammatical/syntactic categories, they now have dedicated names that reflect this³. For instance, to run DSS on a sample, simply type DSSeng +t*CHI and then select your file(s). The program will generate a file containing a table such as that shown in Appendix 8. It shows which points were awarded by utterance for various DSS elements. Similarly, the command IPSYNeng +t*CHI will generate a file containing an IPSYN table (see Appendix 9) showing how the various phrase structure examples were identified in the child's sample. This can provide information about how the clinician might want to go about setting goals to expand utterance constituents.

¹ To get raw counts for this program, add the command +o4.

² To see what options exist for ANY program, go to the command window and type the program name or select it from the Progs drop down menu, then click RUN. The screen output will give you the proper syntax and any unique options that exist for that program.

³ This replaces an earlier requirement to add +leng (language is English) to the command line.

EVAL

EVAL is for clinicians who want to know about language abilities of ADULTS with and without aphasia. When you type (or choose) EVAL in the command window and follow it with a space, it will ask you to pick a file, then pick among options. If you like, it can compare your client to a database of people in AphasiaBank, a large cooperative endeavor funded by TalkBank (Brian MacWhinney and Audrey Holland, PIs). Basically, if you can run KIDEVAL, you can run EVAL. It requires the same %mor line, so be sure you run the MOR command first. The data reported in EVAL are not exactly the same as the data from KIDEVAL – the variables are more appropriate and relevant to an adult sample and to aphasia behaviors (e.g., MLU in words instead of morphemes, no Brown stages).

If, under options, you pick “Update database”, and you pick controls for your comparison group, the resulting Excel spreadsheet will flag (with asterisks), categories in which your patient differed by more than a standard deviation from the control adults from the AphasiaBank database. We think this is pretty cool, frankly [actually, one reviewer of this manual used lots stronger “linguistic enthusiasm”].

ANALYZING WRITTEN (Typed) LANGUAGE SAMPLES IN CLAN

You can analyze written language using EVAL and/or KIDEVAL. To do this, remove any information you don’t want to analyze (e.g., student identifiers, headers specific to the assignment), and RE-SAVE the file as plain text (.txt). This is important, as it will remove irrelevant information (such as font codes, margins, etc. that tend to foil CLAN). Then use the command TEXTIN in the commands window and input your new file. This will produce a CHAT file (e.g., myfile.textin.cha). When you open this file, you will find that it has created a set of headers in CHAT format and replaced speaker tiers with the generic code *TXT, placing each sentence on a main line by using punctuation markers. You will need to do some editing. You can use global search and replace to change *TXT to *CHI, for example. You will need to change header wording manually by using the update tiers option you learned to use earlier. You may need to do additional small cleanup to delete sentence internal punctuation, change “smart” curly quotes and apostrophes from MS-Word to straight quote and apostrophes, etc. But your file should quickly be ready to MOR and then analyze using EVAL and/or KIDEVAL, using the same steps outlined earlier in this manual.

FLUENCY ANALYSES IN CLAN

You can run fluency AND language analyses on the same file. To produce a list of disfluent words, separated by type of disfluency, you can use the FLUCALC command. Unlike KIDEVAL and EVAL, this program does not have any pre-set options, so you need to specify which speaker you want to analyze. In this case, it is the child.

`flucalc +t*CHI filename.cha (or *.cha for a group of files, or use the File In function)`

FLUCALC will perform a fluency analysis of a language sample, in both raw counts and percentages of intended words. It will also provide a “beta” weighted disfluency value over words, based on the formula proposed by the Illinois Stuttering Project (Yairi & Ambrose, 1999) for computations made on syllable counts.

You must use fluency codes specified in this manual (and in the CHAT manual in “Disfluency Transcription). As noted, values are currently word- (rather than syllable-) based. Plans are underway to permit a syllable-based option to be used in interpreting scores against commonly reported measures such as the *SSI-4*.

The same speech/language sample can be used for both language sample analysis (KIDEVAL, EVAL) and fluency appraisal via FLUCALC.

Again, the output will be as it was for KIDEVAL. Please see the explanation in that section above about the .xls file that gets created, how to find it, how to open it, and how to save it.

Columns specify the following values (beyond those provided by header information, such as gender, age, corpus, etc.):

- Total utterances in the sample
- Total mor words: these are the total intended words, as identified by application of MOR to the sample
- # Prolongation: raw count of sound prolongations
- % Prolongation (over mor words)
- # Broken word
- % Broken word
- # Block
- % Block
- # PWR (Part-word repetition)
- % PWR
- # PWR RU (Repetition units): these are sometimes called iterations, or the number of excess repetitions in a part-word repetition. This column totals all RUs seen in the sample, for use in the weighted disfluency score
- % PWR RU
- # Phonological fragment – these are best viewed as abandoned word attempts, e.g. & +*fr-tadpole*, where the speaker appears to change word choices; this code was original to CLAN programs.
- % Phonological fragment
- # WWR (whole word repetition)
- % WWR
- # WWR RU (repetition units; please see PWR above)
- % WWR RU
- # Phrase repetitions
- % Phrase repetitions
- # Word revisions
- % Word revisions
- # Phrase revisions
- % Phrase revisions
- # Pauses

- % Pauses
- # Pause duration (if specified by coder)
- % Pause duration
- # Filled pauses
- % Filled pauses
- # SLD (stutter-like disfluency); this is done by summing over categories in columns Prolongations through whole-word repetitions (WWR), with the exception of columns reporting repetition units (RUs)
- % SLD: proportion of stutter-like disfluencies over total intended words
- # TD (typical disfluencies): this is done by summing categories in columns Phrase repetitions through filled pauses)
- % TD
- # Total (SLD+TD): this sums all forms of disfluency, both stutter-like and typical, seen in the sample
- % Total (SLD+TD)
- Weighted SLD. This is an adapted version of the SLD formula for distinguishing between typical disfluency and stuttering profiles in young children. It was originated by Yairi & Ambrose (1999) and originally referenced against a standard sample of 100 syllables, rather than words, as here.
 - This formula multiplies the SUM of part-word and whole-word repetitions by the MEAN of the observed repetition units in the sample; it then adds this value to TWICE the sum of prolongations and blocks. This formula penalizes the severity of the segment repetition profile as well as the presence of prolonged sounds and blocks, which are virtually absent in any sample of typically fluent speech.
 - Please note that currently, CLAN is not able to isolate whole word repetitions (the only ones counted in Yairi & Ambrose's formula) from repetitions of polysyllabic word targets). We are currently using contributed corpora to the FluencyBank in order to provide guidance on how to adjust interpretation of the Weighted SLD for clinical purposes and hope to provide such guidance in the near future.

The output will look like this, with column headings for each of the output variables across the first row, and filenames for each file analyzed down the first column:

A	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV																																																												
File	mor	Uts	mor	Word#	Prolonga	%	Prolonga	#	Broken	v	%	Broken	i	%	Block		#	PWR	%	PWR		#	PWR	RU	%	PWR	RU	#	Phonoq	%	Phonoq	%	WWR		#	WWR		#	WWR	RU	%	WWR	RU	%	Phrase	r	#	Phrase	r	#	Word	r	%	Word	r	%	Phrase	r	#	Phrase	r	#	Pauses	%	Pauses	#	Pause	du	#	Pause	du	#	Filled	pa	%	Filled	pa	%	SLO		%	SLO		#	TD		%	TD		#	Total	(S)	%	Total	(S)	Weighted	(
S001_3mo	107	368	2	0.005	0	0	0	0	5	0.014	15	0.041	0	0	7	0.019	13	0.035	0	0	0	0	0	0	0	0	0	0	1	0.003	14	0.038	1	0.003	15	0.041	8.696																																																												
S001_intar	112	339	2	0.006	1	0.003	1	0.003	4	0.012	9	0.027	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0.003	0	0	0	0	5	0.015	8	0.024	6	0.018	14	0.041	4.425																																																									
S002_9mo	351	1432	14	0.01	1	0.001	3	0.002	81	0.057	107	0.075	24	0.017	47	0.033	52	0.036	21	0.015	6	0.004	16	0.011	2	0.001	0	0	91	0.064	170	0.119	136	0.095	306	0.214	13	0.047																																																											
S002_intar	374	895	11	0.012	0	0	0	0	30	0.034	73	0.082	0	0	5	0.006	19	0.021	4	0.004	0	0	1	0.001	6	0.007	0	0	6	0.007	46	0.051	17	0.019	63	0.07	12	0.037																																																											
S003_intar	71	199	0	0	0	0	1	0.005	5	0.025	7	0.035	0	0	0	0	0	0	0	0	1	0.005	0	0	1	0.005	0	0	2	0.01	6	0.03	4	0.02	10	0.05	4.523																																																												
S005_3mo	316	842	5	0.006	1	0.001	7	0.008	23	0.027	30	0.036	3	0.004	22	0.026	28	0.033	2	0.002	1	0.001	7	0.008	2	0.002	0	0	24	0.029	61	0.072	36	0.043	97	0.115	9	0.039																																																											
S006_6mo	91	237	0	0	0	0	0	0	0	0	0	0	0	5	0.021	1	0.004	3	0.013	1	0.004	0	0	0	0	0	0	0	1	0.004	6	0.025	2	0.008	8	0.034	1.266																																																												
S006_9mo	255	854	0	0	1	0.001	0	0	4	0.005	4	0.005	0	0	1	0.001	7	0.008	2	0.002	0	0	4	0.005	0	0	0	0	2	0.002	6	0.007	8	0.009	14	0.016	1.288																																																												
S006_intar	214	529	0	0	2	0.004	2	0.004	17	0.032	25	0.047	5	0.009	21	0.04	31	0.059	0	0	0	0	0	0	0	0	0	0	6	0.011	47	0.089	6	0.011	53	0.1	11.342																																																												
S007_12m	127	346	3	0.009	1	0.003	2	0.006	21	0.061	33	0.095	3	0.009	7	0.02	14	0.04	6	0.017	0	0	2	0.006	0	0	0	0	15	0.043	37	0.107	23	0.066	60	0.173	16.474																																																												
S007_intar	159	618	6	0.01	1	0.002	0	0	32	0.052	58	0.094	0	0	27	0.044	40	0.065	0	0	0	0	2	0.003	0	0	0	0	10	0.016	66	0.107	12	0.019	78	0.126	17.799																																																												
S008_3mo	146	430	11	0.026	0	0	1	0.002	22	0.051	38	0.088	11	0.026	9	0.021	14	0.033	4	0.009	1	0.002	1	0.002	0	0	0	0	4	0.009	54	0.126	10	0.023	64	0.149	17.674																																																												
S008_intar	179	415	20	0.049	0	0	0	0	35	0.084	62	0.149	23	0.055	16	0.039	26	0.063	1	0.002	0	0	0	0	0	0	0	0	12	0.029	94	0.227	13	0.031	107	0.258	30.843																																																												
S008_intar	151	267	6	0.022	0	0	0	0	12	0.045	18	0.067	8	0.03	6	0.022	9	0.034	0	0	0	0	1	0.004	0	0	0	0	7	0.026	32	0.12	8	0.03	40	0.15	14.607																																																												
S010_intar	63	129	0	0	3	0.023	1	0.008	3	0.023	5	0.039	1	0.008	2	0.016	2	0.016	0	0	0	0	1	0.008	8	0.062	0	0	6	0.047	10	0.078	15	0.116	25	0.194	6.977																																																												
S010_intar	125	282	2	0.007	0	0	3	0.011	3	0.011	4	0.014	0	0	4	0.014	5	0.018	1	0.004	1	0.004	0	0	0	0	0	0	8	0.028	12	0.043	10	0.035	22	0.078	6.738																																																												
S010_intar	173	344	0	0	0	0	0	0	0	0	0	0	0	4	0.012	5	0.015	4	0.012	1	0.003	0	0	0	0	0	0	7	0.02	4	0.012	12	0.035	16	0.047	1.453																																																													
S011_12m	207	494	8	0.016	0	0	0	0	7	0.014	8	0.016	1	0.002	6	0.012	7	0.014	7	0.014	2	0.004	2	0.004	0	0	0	0	16	0.032	22	0.045	27	0.055	49	0.099	6.275																																																												
S011_intar	419	953	2	0.002	0	0	0	0	38	0.04	52	0.055	0	0	52	0.055	80	0.084	12	0.013	2	0.002	3	0.003	0	0	0	0	139	0.146	92	0.097	156	0.164	248	0.26	14.271																																																												
S012_9mo	295	550	0	0	0	0	0	0	2	0.004	4	0.007	51	0.093	3	0.005	3	0.005	3	0.005	0	0	0	0	0	0	0	0	8	0.015	56	0.102	11	0.02	67	0.122	1.273																																																												
S013_intar	453	1324	0	0	0	0	0	0	6	0.005	14	0.011	0	0	12	0.009	14	0.011	3	0.002	1	0.001	2	0.002	0	0	0	0	18	0.014	6	0.005	24	0.018	2.115																																																														
S015_3mo	243	353	0	0	0	0	0	6	0.017	15	0.042	38	0.108	0	0	1	0.003	1	0.003	0	0	0	0	0	0	0	0	6	0.017	22	0.062	6	0.017	28	0.079	14.448																																																													
S015_intar	43	96	0	0	0	0	0	0	0	0	0	0	0	0	1	0.01	2	0.021	0	0	0	0	0	0	0	0	0	0	1	0.01	0	0	1	0.01	2.083																																																														
S017_3mo	143	466	0	0	0	0	0	0	1	0.002	1	0.002	1	0.002	3	0.006	4	0.009	2	0.004	0	0	1	0.002	0	0	0	0	5	0.011	3	0.006	8	0.017	1.073																																																														
S017_intar	112	425	0	0	0	0	0	0	6	0.014	13	0.031	0	0	11	0.026	19	0.045	3	0.007	0	0	2	0.005	0	0	0	0	1	0.002	17	0.04	6	0.014	23	0.054	7.529																																																												
S018_intar	164	367	0	0	0	0	0	0	3	0.008	5	0.014	0	0	9	0.025	9	0.025	2	0.005	0	0	2	0.005	3	0.008	0	0	3	0.008	12	0.033	10	0.027	22	0.06	3.815																																																												
S019_intar	188	564	0	0	1	0.002	10	0.018	8	0.014	13	0.023	8	0.014	4	0.007	4	0.007	0	0	1	0.002	7	0.012	1	0.002	0	0	13	0.023	31	0.055	22	0.039	53	0.094	6.56																																																												
S020_intar	131	254	3	0.012	0	0	3	0.012	29	0.114	59	0.232	2	0.008	5	0.02	8	0.031	0	0	0	0	0	0	0	0	0	2	0.008	42	0.165	2	0.008	44	0.173	31.102																																																													
S021_intar	259	641	3	0.005	4	0.006	2	0.003	45	0.07	104	0.162	28	0.044	22	0.034	42	0.066	10	0.016	0	0	1	0.002	0	0	0	100	0.156	104	0.162	111	0.173	215	0.335	24.337																																																													
S023_intar	280	832	12	0.014	2	0.002	0	0	29	0.035	62	0.075	0	0	20	0.024	23	0.028	7	0.008	0	0	1	0.001	0	0	0	0	18	0.022	63	0.076	26	0.031	89	0.107	13.101																																																												
S024_intar	264	557	0	0	0	0	0	0	3	0.005	6	0.011	0	0	2	0.004	2	0.004	1	0.002	1	0.002	1	0.002	1	0.002	0	0	1	0.002	5	0.009	5	0.009	10	0.018	1.436																																																												
S025_intar	181	554	0	0	0	0	4	0.007	6	0.011	9	0.016	0	0	25	0.045	32	0.058	3	0.005	0	0	2	0.004	0	0	0	0	7	0.013	35	0.063	12	0.022	47	0.085	8.845																																																												



YOU ARE NOW DONE WITH EVERYTHING YOU NEED TO KNOW ABOUT RUNNING CLINICAL ANALYSES USING CLAN.

YOU CAN STILL ASK OTHER CLINICAL AND/OR RESEARCH QUESTIONS USING THE SAME TRANSCRIPT, SO HERE ARE SOME MORE PROGRAMS TO LOOK AT. YOU CAN ALSO GO TO THE CLAN MANUAL AND BE AMAZED AT HOW MANY THINGS YOU CAN DO WITH A SINGLE TRANSCRIPT.
HAVE FUN!

Remember, the screencasts -- <https://talkbank.org/screencasts/> -- provide short demonstrations of most everything that's covered in this manual.

IF YOU WANT TO LEARN HOW YOU CAN USE YOUR TRANSCRIPT TO DO PHONOLOGICAL ANALYSIS, GO TO <https://www.phon.ca/phontrac>



JOINING CHI-BOLTS

We predict you will really like CLAN after the rather steep initial learning curve; if you think you will use it after being MADE to use it in any class assignments, we suggest you do something pretty simple to stay up to date and figure out “glitches” or new ways to use the system. Go back up to the CHILDES webpage -- <https://chilDES.talkbank.org/> -- and look under “Contact” for a link to join membership lists. CHI-Bolts is the non-intrusive but helpful listserv that helps people who get frustrated, who seem to have located problems (bugs), who wish Brian MacWhinney and Leonid Spektor would write a new program or option [sometimes, their wish is fulfilled within hours], or who want to lurk and see how people get creative. You won't get more than one note a week, typically.

If you think you have found a “bug”, please do the following: send the list your sample file, the command string you used, and the error message CLAN sent you instead of the output that you hoped for.

CONVERTING SALT FILES

Are you a SALT user? If we have won you over to the extent that you wish you could analyze OLD files using CHAT/CLAN, we can still help you. If you have a SALT file, the SALT2CHAT command will turn the file to CHAT and then, typically with few fixes, you can start from the

point at which you run MOR, POST, etc. to get EVAL or KIDEVAL output to compare. If you have totally unformatted text, the TEXT2CHAT command takes files, no matter how messy (one sentence on a line, or whole paragraphs), and generates a rough CHAT transcript that you can clean up and work with using the information in this guide.

OTHER COMMANDS AND PROGRAMS PEOPLE USE A LOT:

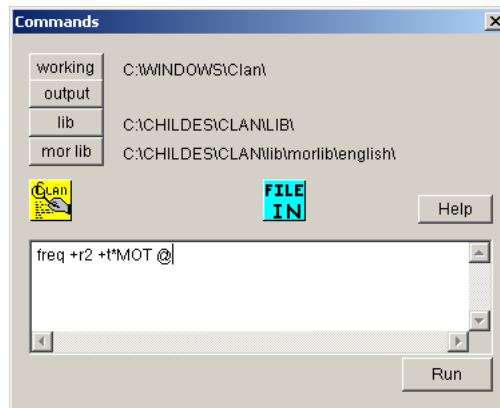
For all CLAN commands, if you type the command into the command window or select it from the “Progs” drop down menu and press “Run”, you will get a page of information about the command line and the many options for tailoring the command to your needs. Also, the screencasts page has some Commands examples you can watch.

FREQ

FREQ is a command that lists all of the words used in that transcript and counts how many times each word is used. BECAUSE IT CAN BE CUSTOMIZED, FREQ IS THE MOST POPULAR AND USEFUL PROGRAM IN CLAN. It’s normally used to compute the Type-Token Ratio (TTR), but can be used for much more.

One use of the FREQ command is to see a list of the words in your transcript for purposes of spell-checking. (Note: If you ran the mor +xb command, you should not need to do this.) To run FREQ for spell-checking, simply type “freq +r2” into the Commands window. If you want to target a specific speaker (let’s say the Mother’s utterances), add +t*MOT (or whatever the 3 letter speaker ID of interest is) to the command. “FREQ” is the command. The t*MOT tells the program to only run on lines beginning with *MOT. When the analysis is run with no speaker tier identification, it will automatically run on all main lines.

Once you have typed in your command, click the “file in” button and choose the file(s) on which you wish to run FREQ in the same way you chose files for CHECK. (Or you can type the filename or *.cha for all files in the working directory folder.) Then click Run on the Commands window or press Return/Enter on your keyboard to run the analysis. An output file will open with a list of all words used in the transcripts.



If you are spell-checking, read through the words and identify any misspellings or miscodings. Common things to look for are: words connected to punctuation marks, which will happen if spaces are not inserted between the end of the utterance and the punctuation or if an error has been made in the coding; typing xx when it should be xxx for unintelligible words or phrases;

and misspellings of words such as “spahgetti.” (see KWAL below for how to fix this kind of nasty mistake).

The manual for CLAN shows the many things you can count using **FREQ**, which are virtually **unlimited** (which is why most researchers spend most of their lives using this single option). A widely used option is to make a text file with a class of words you’d like to track (such as fluency codes, question words, pronouns, etc.) and then **FREQ** (yep, it can be a verb!) your language sample using that file as input. That can be a way to check which pronouns the child has or doesn’t yet have, etc. It works just like **freq**, except that you use the option **+s@<filename>** in the command line, where “+s” means ‘search for’ what is described in the file specified. So, if you made a file with question words, one per line, and saved it as **whwords.cut** in the CLAN program **lib** directory (in **Applications**), you could run this command:

```
freq +t*CHI +s@whwords.cut *.cha
```

The output would list how many times the child used “what”, “who”, “where”, “when”, “why” and “how”. Examples of how to use **FREQ** to search common kinds of things that clinicians or researchers might want are listed in great detail in the “real” CLAN manual. Note: If you make your own **.cut** files, you do it in **CHAT** (go to “File” in the menu, select “New”) but then go to “Mode” in the menu and unselect **CHAT** mode (so there’s no check mark by it) and you’re in text mode. And remember, save it as **.cut**, not **.cha**. Also, **HOT TIP** -- <https://talkbank.org/examples.zip> has examples of **.cut** files in a folder called “pos”. You can grab the **whwords.cut** file and stick it in the CLAN **lib** folder or in the folder where your files are that you want to analyze and you can run it and see the output for yourself.

There is also a “fluency” folder inside that **examples.zip** folder that contains a utility file with all the stuttering codes. It is used to calculate fluency rates, and can be used for other analyses as well (if you are a researcher). For instance, you could insert the stutter codes from the main tier on dependent tiers, such as the **%mor** line. Then you could calculate the rate of stuttering on various parts of speech⁴.

KWAL (for KeyWordAndLine)

KWAL is another analysis run by CLAN using the **Commands** window. This analysis takes a word and finds the lines on which that word occurs in each transcript. This analysis is necessary to find out which lines the target word is on and in what position in the utterance the target is located. So, if you wanted to search for the Mother’s use of the target word “book”, you would type this command (where the target word is entered immediately after the “s” which means “search for”):

```
kwal +t*MOT +sbook filename.cha (or *.cha for a group of files, or use the File In function)
```

(If the target you are looking up contains an apostrophe, enclose the target in quotation marks.

For example: `kwal +t*MOT +s"he's"`)

⁴ For an example, see Tsai, Peitzu, Brundage, Shelley, Lim, Valerie and Bernstein Ratner, Nan (2010). Linguistic analysis of bilingual stuttering: concepts and methods. In P. Howell & J. Van Borsel (eds.). *Fluency disorders and language diversity*. Taylor & Francis.

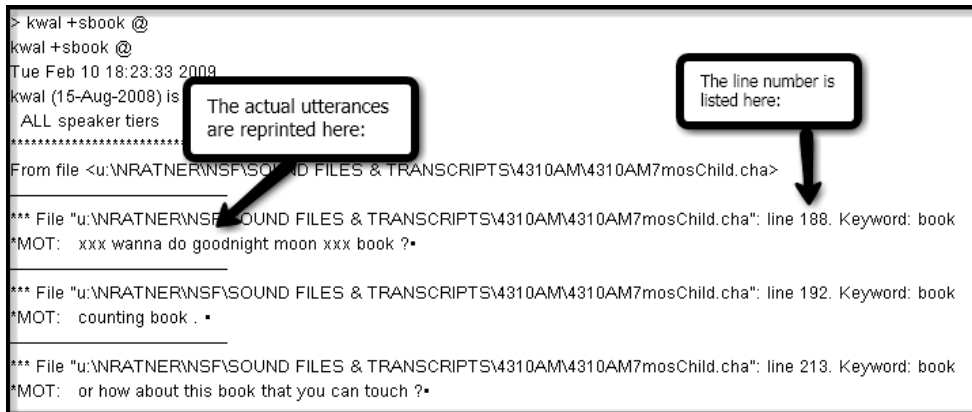
The output will show an entry for each instance of that word used in the transcript and it will tell you on what lines the target occurred.

```
> kwal +sbook @
kwal +sbook @
Tue Feb 10 18:23:33 2009
kwal (15-Aug-2008) is
ALL speaker tiers
*****
From file <u:\NRATNER\NSF\SOUND FILES & TRANSCRIPTS\4310AM\4310AM7mosChild.cha>

*** File "u:\NRATNER\NSF\SOUND FILES & TRANSCRIPTS\4310AM\4310AM7mosChild.cha": line 188. Keyword: book
*MOT: xxx wanna do goodnight moon xxx book ?*

*** File "u:\NRATNER\NSF\SOUND FILES & TRANSCRIPTS\4310AM\4310AM7mosChild.cha": line 192. Keyword: book
*MOT: counting book . *

*** File "u:\NRATNER\NSF\SOUND FILES & TRANSCRIPTS\4310AM\4310AM7mosChild.cha": line 213. Keyword: book
*MOT: or how about this book that you can touch ?*
```



If you triple click on the line that has the triple asterisks, it will take you right to the line in the transcript. This is great for fixing mistakes if you find them at some point. KWAL the mistake, and the program will provide output that you can click to go right to the error and fix it.

COMBO

COMBO finds combinations of things that you might want to count or track; for example, if you wanted to know how often a child said “Once upon a time”, you would use COMBO because this is a multi-word string. Likewise, if you wanted to see how often a child used the aux with a participle in the present progressive on the %mor tier, you would use a combo command, like this:

```
combo +t%mor +s"aux|^part|^*-PROG" filename.cha
```

MORTABLE

Mortable will generate an Excel spreadsheet that computes the proportion of particular syntactic elements (not raw counts), in a sample, quite an excellent clinical assessment or tracking ability. The command for mortable is:

```
mortable +t*CHI +leng filename.cha
```

Like KIDEVAL, this output will be an Excel spread sheet. Here is a sample output, from some aphasia data (it used +u to include all the files in a directory of clients)

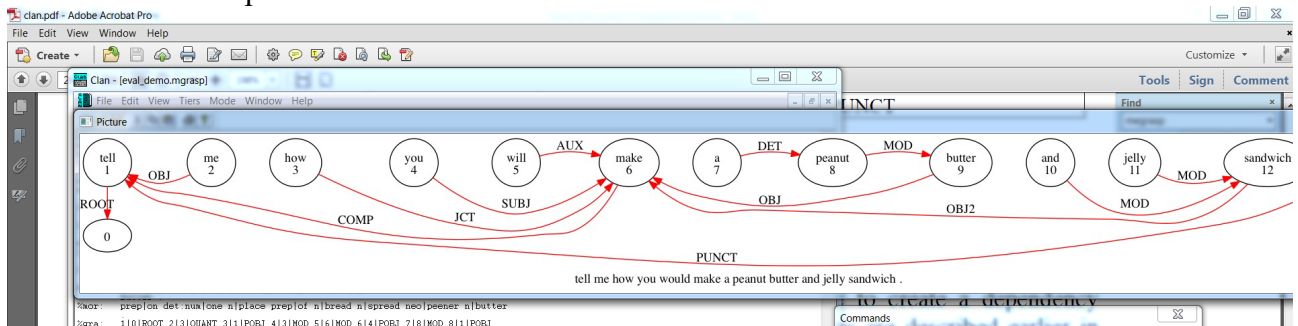
	A	B	D	E	F	G	H	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
1	File	Language	Code	Age	Sex	Group	SES	Education	Custom_fi	# words	% *wh	% adj	% adv	% aux	% conj	% coord	% det	% inf	% modal	% n,n*	% neg	% part	% PE
2	adler01a	eng	PAR	58;11	male	Anomic	adler01a	86.8	1447	0.968	3.041	5.943	2.211	2.626	8.017	8.569	1.313	1.382	16.793	2.419	0.207	1.866	
3	adler02a	eng	PAR	69;9	male	Conductio	adler02a	74.9	1877	0.852	5.807	9.164	1.119	2.397	8.258	11.827	0.213	1.225	14.544	2.504	0.107	0.906	
4	adler03a	eng	PAR	78;5	female	NotAphasi	adler03a	95.8	1304	0.92	2.531	4.371	2.684	2.147	6.212	7.822	2.607	0.767	15.184	2.531	1.074	1.15	
5	adler04a	eng	PAR	75;6	female	TransMotc	adler04a	72.6	626	0.319	5.431	4.633	2.556	2.396	11.182	9.425	0.799	0	18.69	1.438	0.958	1.278	
6	adler05a	eng	PAR	68;2	female	Conductio	adler05a	65.5	1121	0.803	2.498	4.996	4.193	1.606	8.385	9.634	1.517	1.338	14.095	1.784	0.981	2.052	
7	adler06a	eng	PAR	70;7	male	Wernicke	adler06a	28.2	504	1.19	4.563	4.563	3.175	1.984	6.151	8.73	0.595	0.992	12.302	0.992	0.992	1.786	
8	adler07a	eng	PAR	80;8	female	NotAphasi	adler07a	95.6	3190	1.285	3.73	7.774	3.354	2.508	7.147	6.113	1.379	1.317	12.194	2.1	0.909	2.351	
9	adler08a	eng	PAR	56;9	male	Anomic	adler08a	78.5	735	0.68	4.354	6.939	1.224	3.946	7.211	8.435	0.272	2.041	11.973	1.905	0.408	0	
10	adler09a	eng	PAR	41;8	female	Anomic	adler09a	92.8	1493	1.072	2.545	10.918	1.875	2.076	6.363	7.033	2.009	1.942	15.941	1.34	0.737	0.737	
11	adler10a	eng	PAR	44;8	male	Broca	adler10a	51.2	244	2.049	3.279	2.049	0	0.41	7.787	8.197	1.639	0	16.803	1.639	0.41	0.82	
12	adler11a	eng	PAR	80;11	male	Broca	adler11a	17	194	0	0	1.546	0	1.031	0	0	0	0	3.608	0.515	0	0	
13	adler12a	eng	PAR	40;7	female	Anomic	adler12a	87.2	1126	0.799	3.908	8.348	2.043	3.197	5.329	8.526	2.043	1.599	14.654	2.575	0.888	1.243	
14	adler13a	eng	PAR	52;4	male	Broca	adler13a	55.8	646	2.477	2.012	1.084	0.31	0	8.824	1.238	0	0	14.396	4.18	0.31	0.619	
15	adler14a	eng	PAR	71;4	male	Conductio	adler14a	83	1390	0.719	2.374	5.18	2.014	1.079	3.885	7.914	2.158	0.288	19.568	1.655	0.36	1.727	
16	adler15a	eng	PAR	78;11	male	Anomic	adler15a	73.5	2365	1.015	4.059	7.146	2.664	1.86	7.653	6.85	1.142	1.099	13.192	1.268	0.254	1.353	
17	adler16a	eng	PAR	63;6	male	Broca	adler16a	57.2	514	0.389	4.28	6.031	2.918	1.167	9.728	16.732	0	0	20.233	0.389	1.946	0	
18	adler17a	eng	PAR	85;2	male	Anomic	adler17a	86.8	2158	0.788	5.051	7.97	0.741	1.993	9.87	4.124	0.695	0.232	21.038	1.576	0.093	0.51	
19	adler18a	eng	PAR	71;6	male	TransMotc	adler18a	59.8	392	0.255	11.99	11.99	1.531	1.531	11.735	5.357	1.02	0.765	14.286	1.531	0.255	1.276	

If you run MORTABLE on two different files, for instance your client’s baseline file and then another file some sessions later, it would allow you to compare how much more (hopefully) your client uses a target morpheme under treatment.

MEGRASP

Currently, MEGRASP isn’t used by any clinical programs, although it was developed for IPSYN, another powerful child language sampling procedure with norms that KIDEVAL measures. It runs automatically when you run MOR and creates a %gra tier. If you triple click on that tier in your transcript, you will actually see a tree diagram pop up for your utterance. *This might be a fun party trick if a linguistics get-together needs something to liven it up. Or you want to help your kid with burdensome English homework 😊* We have included it here for its WOW value. Stay tuned for more developments. CLAN develops at the speed of light. And it never costs you anything to upgrade.

Here is the WOW picture:



More information for people who want to know more/too much:

OPTIONAL TRANSCRIPTION OPTIONS

Manually adjusting bullet length:

First, go to Mode menu and choose expand bullets (or use the ESC-A option, typing the “esc” key, releasing it, then typing “A”). Now you’ll see actual numbers (in msec) inside a begin and end bullet. For PC users, the CONTROL key (for the start bullet) and SHIFT key (for the end bullet) can adjust bullet lengths up and down slightly as needed. For Mac users, it’s the CONTROL and COMMAND keys. Push the key with the right or left arrow to go up or down

by 25msec intervals. This is an option as opposed to re-bulleted a finished transcript to get the bullets to align more perfectly. You can also manually change the numbers in the bullets but you have to be very careful with that. If you accidentally delete a bullet (the dot) from either end of the open bullet, just copy and paste any other dot back in. (Again, try to suppress spending a lot of time to make them perfectly align because the bullets for most clinical work (and even research) is more to help speed transcription; it does not enter into most analyses.

“Lazy GEMS”:

Use this command to mark different activities in one interaction. Suppose you have different activities in one session, and you don’t want KIDEVAL or another program to include some of them. You can use lazy gems to mark the beginnings of different activities, e.g., @G: talking, @G: reading (which you don’t want, because it is not spontaneous language), and run your programs using the option to do only “talking” gem sections. See the CHAT manual or the 3:46 minute screencast on Lazy Gems if you want more info.

We think this is more than enough to get you started using CHAT media-linked transcription and CLAN.

We hope you like it, will use it, and will provide us with feedback.

We are also here to help.

Enjoy!⁵

Acoustic Analysis with PRAAT is explained in the next few pages, followed by the Appendices referred to throughout the manual.



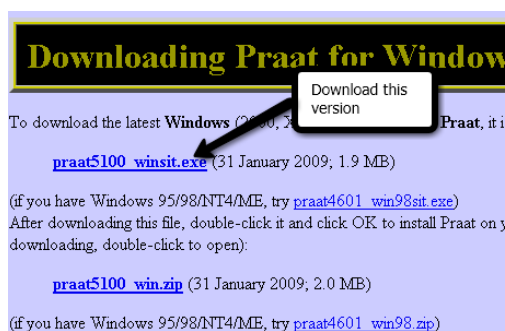
⁵ With thanks to the US NIH, for its support of CHILDES, Aphasia Bank and Phon and NSF for its support of TalkBank.

ACOUSTIC ANALYSIS

One of the more amazing talents that CLAN has is the ability to go straight from a linked transcript (audio or video) to acoustical analysis that many clinicians would probably gladly do if it were easy enough (to look at properties of sound segments, intonation contour of utterances, rate calculations, etc. It is a completely seamless interface. This next section shows how to do this.

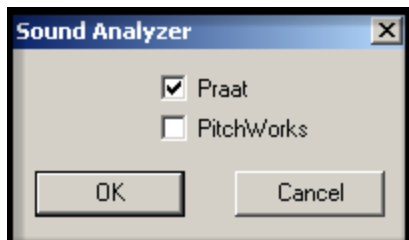
PRAAT

To run the acoustic analysis, you will need to download PRAAT. This can be done at <http://www.fon.hum.uva.nl/praat/>. Click “Windows” (or Mac) in the upper right-hand corner and then click on the first download option on the next page. Instructions for how to install are listed on that page. There is also a PRAAT tutorial for CLAN in the tutorial series at the CHILDES web site (file 13; PRAAT pdf); you should probably read that as well.

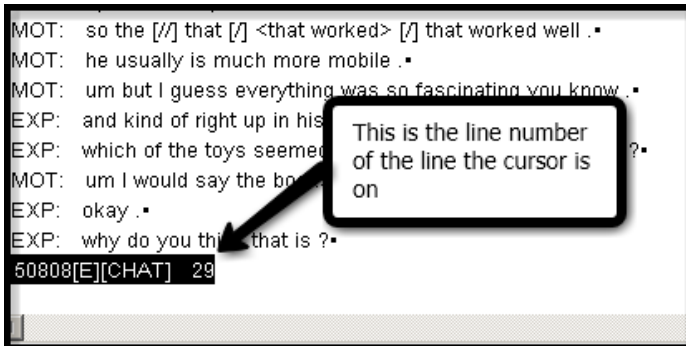


From CLAN to PRAAT

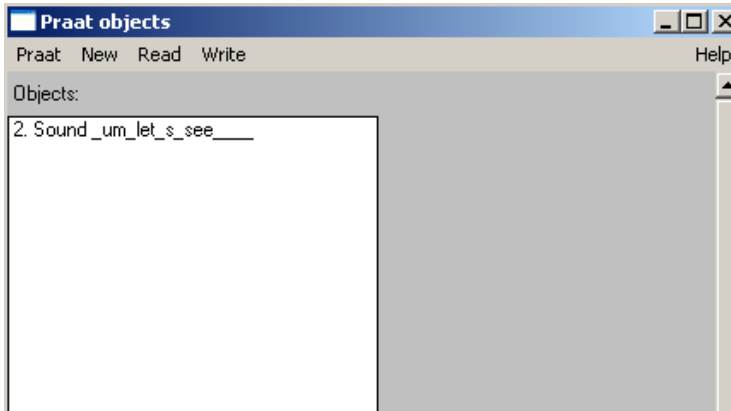
The bullets in CLAN can be sent directly over to PRAAT, but you’ll have to set it up to do this. First, open both programs. PRAAT MUST be open first in order to use it with CLAN. When you open PRAAT, you can close the window labeled PRAAT Picture and leave open the window labeled PRAAT Objects. In CLAN, you can close the Commands window. Then, in CLAN, go to Edit → Select Sound Analyzer. In the new window, click the box next to PRAAT and make sure the other box is unchecked. Once you have set this, the setting will remain this unless you change it. You don’t need to do this step every time.



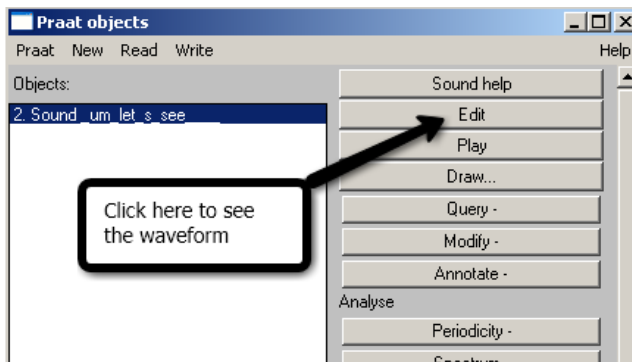
Next, open the transcript you are working on. Using the spreadsheet, look up each matched target. The spreadsheet will tell you in which transcript and on what line the target is located. The line numbers in CLAN show up at the bottom of the screen.



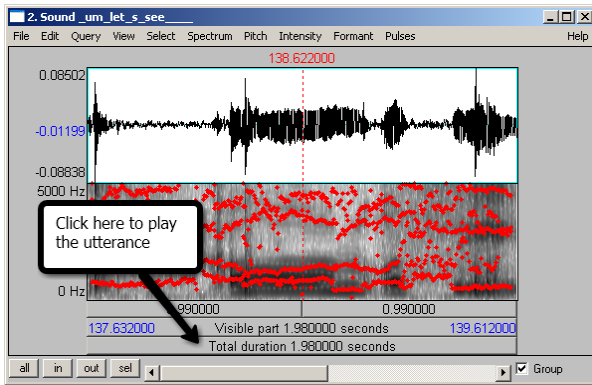
Locate the line containing the target and click your cursor on that line. Then, go to Mode → Send to Sound Analyzer. For this to work, PRAAT must already be open. Once you do this, the text from that line in the transcript will show up in the PRAAT Objects window.



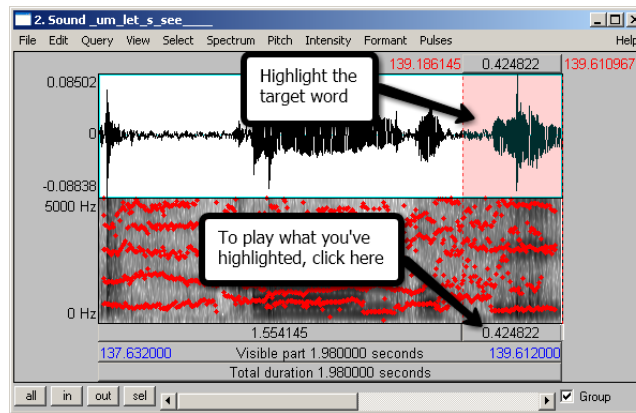
Click on the object you are analyzing and then options will show up on the right side of the window. Click Edit to see the waveform and spectrogram.



Another window will open, and in that window you will see two version of the same thing. The top one is the waveform and the bottom one is the spectrogram. If you click on the bar on the bottom, the whole utterance should play.

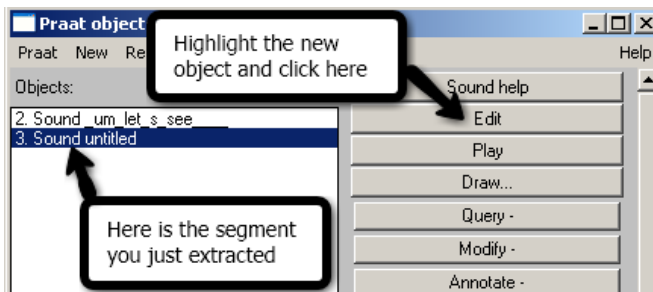


After listening to the utterance, find the section of the waveform that contains the target word and highlight it by clicking at the beginning of the word and dragging your mouse over to the end of the word. Play what you have selected by pressing on the top bar. Adjust your highlighted portion until you are satisfied that you have the whole word and nothing else. This will take some practice.

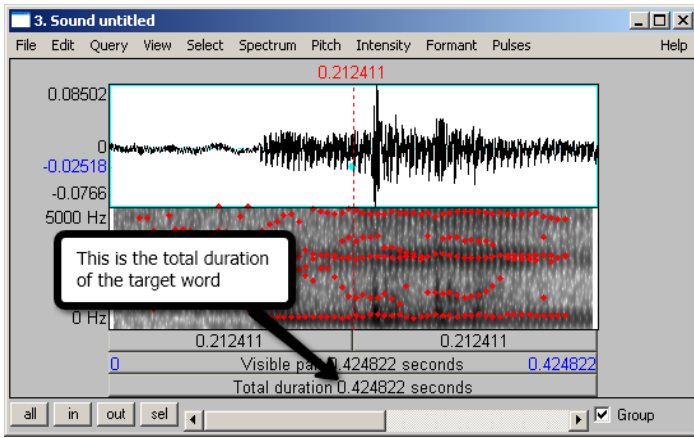


Word (or utterance) duration

Once you have isolated the target word, go to File → Extract Selected Sound (Time from 0) and close the window. This will create a new object in the Objects window. To look at a close-up version of the target word, click on the new object and click Edit.



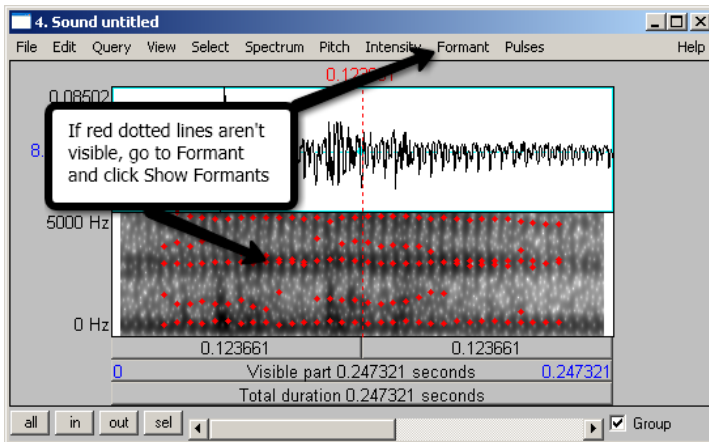
When the new waveform window opens, record the total duration of the word in seconds. You can find this value at the bottom of the window where it says Total Duration.



To do a much more precise measurement of speaking rate than you can with a stopwatch (☹), you can mark the beginning and end of an utterance, get its timing across the bottom window, count the number of words/syllables in the utterance (you already have them written out!), and there you go!

Formants

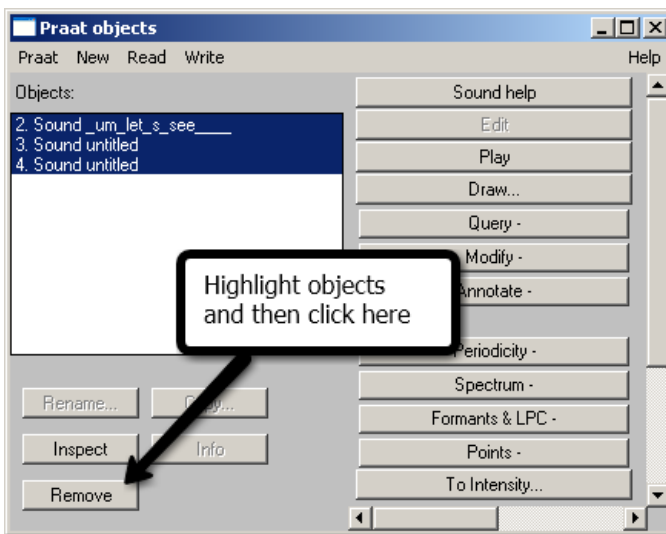
To find the first and second formants, look for red dotted lines in the spectrogram of the isolated vowel. If they are not there, go to Formant → Show Formants. This will make the dotted lines show up. Then, find a place where the formants look as stable as possible. This means they should make a straight line across a section of the vowel. Click in the middle of the most stable part.



Then go to Formant → Formant Listing. A new window will open. In this window, the formants are listed. Record the first and second formants in the appropriate columns in the spreadsheet.

Time_s	F1_Hz	F2_Hz	F3_Hz	F4_Hz
0.092807	487.620744	2866.452755	2991.374778	4694.548560

Once the spreadsheet is filled in for that target, you can move on to the next one. In order to keep things uncluttered to avoid any confusion, you should remove the objects you are finished with from the Objects window before starting to work on a new utterance. To do this, highlight the unwanted objects and then click Remove at the bottom of the window.



Other typical PRAAT options are to show Intonation Contour, Intensity, etc. There are a number of PRAAT tutorials on the web for you to explore. It is possible to write scripts for some acoustic analyses that will automate the process. This is an advanced topic, but might be worth learning if you have lots of files that you wish to evaluate.

APPENDIX 1: Reference values for children's MLU

MLU_w and MLU_m (From Rice et al., 2010)

Table 4. Means and standard deviations for mean length of utterance in words (MLU_w) and morphemes (MLU_m) per group.

Age range	n	MLU _w			MLU _m		
		M	SD	Cohen's d effect size	M	SD	Cohen's d effect size
				Affected			
2;6-2;11	6	2.37	0.32	0.93	2.59	0.39	0.90
3;0-3;5	15	2.84	0.38	0.97	3.07	0.48	1.07
3;6-3;11	24	3.10	0.75	1.04	3.36	0.80	1.09
4;0-4;5	54	3.31	0.70	1.22	3.64	0.80	1.22
4;6-4;11	72	3.60	0.62	0.95	3.95	0.70	1.01
5;0-5;5	84	3.72	0.61	1.05	4.09	0.70	1.10
5;6-5;11	97	3.95	0.60	0.85	4.34	0.67	0.89
6;0-6;5	108	3.98	0.70	0.90	4.38	0.75	0.92
6;6-6;11	94	4.18	0.71	0.79	4.63	0.79	0.84
7;0-7;5	103	4.15	0.62	0.69	4.56	0.69	0.72
7;6-7;11	100	4.33	0.88	0.57	4.77	0.96	0.60
8;0-8;5	94	4.36	0.75	0.86	4.80	0.83	0.89
8;6-8;11	61	4.49	0.86	0.72	4.97	0.93	0.68
				Unaffected			
2;6-2;11	17	2.91	0.58		3.23	0.71	
3;0-3;5	29	3.43	0.61		3.81	0.69	
3;6-3;11	38	3.71	0.58		4.09	0.67	
4;0-4;5	49	4.10	0.65		4.57	0.76	
4;6-4;11	74	4.28	0.72		4.75	0.79	
5;0-5;5	78	4.38	0.63		4.88	0.72	
5;6-5;11	77	4.47	0.61		4.96	0.70	
6;0-6;5	70	4.57	0.66		5.07	0.75	
6;6-6;11	63	4.70	0.66		5.22	0.71	
7;0-7;5	51	4.72	0.83		5.22	0.91	
7;6-7;11	47	4.92	1.03		5.45	1.13	
8;0-8;5	41	5.08	0.84		5.67	0.97	
8;6-8;11	18	4.99	0.71		5.51	0.79	

APPENDIX 2: Norms for TTR

Type-Token Norms

Total Number Of Words And Different Words In T-Unit Sample Based On 50 Utterances					
Age	Words	Conversation		Narrative	
		Mean	SD	Mean	SD
3	Total	160.53	33.91	194.26	49.97
	Different	76.35	16.86	83.41	17.84
4	Total	195.78	60.68	257.19	48.17
	Different	92.07	24.41	108.26	17.75
5	Total	254.48	45.79	277.82	54.75
	Different	114.91	18.97	110.12	15.47
6	Total	251.83	59.93	325.31	80.25
	Different	112.95	20.53	121.44	21.11
7	Total	287.05	62.04	379.46	78.64
	Different	124.89	17.94	132.97	20.53
9	Total	311.70	66.61	419.41	73.43
	Different	133.48	21.48	133.26	25.11
11	Total	342.41	99.39	465.44	84.24
	Different	141.70	29.12	168.70	24.69
13	Total	330.15	78.92	447.96	72.02
	Different	139.00	24.92	159.11	21.17

Miller, J., Long, S., McKinley, N., Thormann, S., Jones, M.A., and Nockerts, A. (2005). *Language sample analysis II: The Wisconsin guide (pp. 21-22)*. Madison, WI: Wisconsin Department of Public Instruction.

APPENDIX 3: Reference values for NDW

TABLE 4. NDW in 100- and 200-token samples.

	Group			<i>F</i> (2,72)
	SLI	LE	AE	
NDW 100-tokens	53 _a (6)	53 _a (8)	60 _b (8)	6.74*
NDW 200-tokens	82 _a (9)	84 _a (11)	93 _b (8)	10.24*

Note. Group means are reported, with standard deviations in (.). Means with different subscripts differ significantly using Tukey HSD contrast.

* $p < .05$

Comparison values for NDW (100 word sample); note: LE = language-age peers for SLI sample; AE is age-matched peers.

Age in months (sd): SLI 59.12 (4.96) LE 38.84 (3.01) AE 59.08 (5.04)

(from Watkins, R. V., Kelly, D. J., Harbers, H. M., & Hollis, W. (1995). Measuring Children's Lexical Diversity: Differentiating Typical and Impaired Language Learners. *Journal of Speech, Language, and Hearing Research*, 38(6), 1349-1355.)

APPENDIX 4: Reference values for DSS

From Lee, L. L. (1974). *Developmental sentence analysis: A grammatical assessment procedure for speech and language clinicians*. Northwestern University Press. Page 230.

TABLE 7. DSS MEANS AND STANDARD DEVIATIONS OF 200 SUBJECTS BY ONE-YEAR AGE GROUPS

Age Group	N	Mean DSS	Standard Deviation
2-0 to 2-11	40	3.73	1.28
3-0 to 3-11	40	6.64	1.00
4-0 to 4-11	40	8.04	1.64
5-0 to 5-11	40	9.19	1.90
6-0 to 6-11	40	10.94	2.26

Table 8 shows the ranges and percentiles of the average DSS sentence score for the two hundred children by one-year age groups. The 10th-, 25th-, 50th-, 75th- and 90th-percentile values for the over-all DSS within each age group were computed (Friedman 1971). Comparison of children's clinical DSS measures with percentile rankings for children of the same chronological or mental age is discussed in Chapter IV.

TABLE 8. RANGE AND PERCENTILES OF DSS SCORES FOR 200 SUBJECTS BY ONE-YEAR AGE GROUPS

Age Group	N	Range	Percentiles				
			10th	25th	50th	75th	90th
2-0 to 2-11	40	1.50-6.70	2.30	2.98	3.73	4.48	5.16
3-0 to 3-11	40	4.60-8.60	5.38	5.98	6.64	7.30	7.89
4-0 to 4-11	40	4.86-12.94	6.01	6.97	8.04	9.10	10.06
5-0 to 5-11	40	6.04-13.40	6.72	7.89	9.19	10.49	11.66
6-0 to 6-11	40	6.64-15.84	8.11	9.43	10.94	12.43	13.78

APPENDIX 5: Interpretation of IPSYN

- The following table is from Oetting, J. B., Newkirk, B. L., Hartfield, L. R., Wynn, C. G., Pruitt, S. L., & Garrity, A. W. (2010). Index of Productive Syntax for Children who speak African American English. *Language, Speech, and Hearing Services in Schools, 41*(3), 328-339.

Table 5. Across-study comparison of IPSyn totals by sample size and child age.

<i>Sample size and study</i>	<i>2½ years</i>		<i>3½ years</i>		<i>4 years</i>		<i>6 years</i>	
	M	SD	M	SD	M	SD	M	SD
75 utterances								
African American English, Horton-Ikard et al. (2005)	52.27	5.16	70.19	8.52	—	—	—	—
Mainstream English, Scarborough (1990) ^a	52.73	10.13	72.20	7.23	80.53	5.48	—	—
100 utterances								
African American English, Current study	—	—	—	—	83.75	5.39	81.91	9.04
Mainstream English, Scarborough (1990) ^a	58.80	10.67	79.53	6.37	85.80	4.21	—	—
100+ utterances								
African American English, Current study	—	—	—	—	91.10	5.24	87.69	10.96
Southern White English, Oetting et al. (1999)	—	—	—	—	84.00	5.67	89.63	5.23
African American English, Oetting (2005)	—	—	—	—	85.92	9.49	91.75	11.69

^aThis study reflects a longitudinal design; others listed are cross-sectional.

- This next table shows values for Kindergarteners with and without Specific Language Impairment. Hewitt, L., Hammer, C., Yont, K. & Tomblin, J.B. (2005) Language sampling for children with and without SLI. *Journal of Communication Disorders, 38*, 197-213.

Table 1

Syntactic and semantic language measures for kindergarten children with and without SLI

	SLI			ND		
	Mean	S.D.	Range	Mean	S.D.	Range
Total number of utterances	98	30.3	52–152	105	33.9	63–182
IPSyn nouns Score	20.11	1.18	17–22	20.51	0.89	18–22
IPSyn verbs Score	27.33	3.86	19–33	29.00	2.56	23–33
IPSyn sentences score	26.77	3.93	19–34	29.40	3.55	21–37
IPSyn total score	74.48	7.64	62–88	79.14	5.08	69–87
MLU-m	5.82	1.32	4.24–8.96	6.86	0.74	5.26–8.13
NDW-50	122.7	20.1	78–162	137.8	15.2	97–164

Note: IPSyn: index of productive syntax (Scarborough, 1990); MLU-m: mean length of utterance in morphemes; NDW-50: number of different word roots in 50 utterances.

- Yang et al. (2021) demonstrated that IPSYN works equally well with 50 utterances compared with 100. Therefore, the command in CLAN requires only 50 utterances. See the CLAN manual for information on how to run it with 100 utterances if desired. The 3 tables below show reference values for revised IPSyn based on 50 utterance sample. Yang, J. S., MacWhinney, B., & Bernstein Ratner, N. (2021). *The Index of Productive Syntax:*

Appendix (p. 1 of 3)

Revised Items and Statistical Properties of the Proposed IPSyn-C

Table A1. Item parameter estimates and standard errors for overall scale items.

Item	Description	Overall scale calibration									
		100-utt sample (N = 388)						50-utt sample (N = 639)			
		a (SE)		c ₁ (SE)		c ₂ (SE)		a (SE)		c ₁ (SE)	c ₂ (SE)
N1	Noun			*					*		
N2	Pronoun			*			3.8	(1.5)	11.8	(3.8)	8.9 (3)
N3	Modifier			*					*		
N4	Two-word NP			*					**		
N5	Det + N			**			1.6	(0.3)	4.4	(0.4)	3.2 (0.3)
N6	V + 2-wd NP	2.3	(1.8)	6.9	(3.8)	5.9 (2.5)	3.1	(0.5)	5.6	(0.6)	4.0 (0.5)
N7	N plural	1.0	(0.4)	4.0	(0.6)	2.9 (0.4)	0.7	(0.1)	2.4	(0.2)	1.1 (0.1)
N8	2-wd NP + V	1.5	(0.3)	1.7	(0.3)	0.5 (0.2)	1.4	(0.1)	0.3	(0.1)	-1.1 (0.1)
N9	3-wd NP	1.6	(0.3)	2.9	(0.4)	1.4 (0.3)	1.5	(0.1)	0.9	(0.1)	-0.7 (0.1)
N10	NP Adv	1.0	(0.2)	0.4	(0.2)	-1.3 (0.2)	0.8	(0.1)	-0.7	(0.1)	-2.6 (0.2)
N11	Bound morpheme	0.4	(0.2)	0.4	(0.2)	-0.1 (0.2)	0.4	(0.1)	-0.6	(0.1)	-1.1 (0.1)
V1	Verb			*					*		
V2	V part or prep			*			2.5	(0.5)	5.4	(1.5)	4.8 (1.7)
V3	Prep phr	2.1	(1.6)	7.1	(4.1)	5.2 (2.1)	2.3	(0.4)	4.5	(1.4)	2.7 (0.4)
V4	N + cop + N	1.9	(1.3)	4.7	(1.4)	4.2 (1.4)	1.5	(0.2)	2.6	(0.2)	2.1 (0.2)
V5	Catenative			**					**		
V6	Aux BE, DO, HAVE	2.6	(0.7)	4.7	(0.8)	4.1 (0.7)	1.9	(0.2)	1.9	(0.2)	1.3 (0.2)
V7	Prog -ing	1.0	(0.3)	3.3	(0.4)	1.8 (0.2)	0.7	(0.1)	1.4	(0.1)	0.0 (0.1)
V8	Adverb			**			2.4	(1.7)	-7.9	(3.2)	NA
V9	Modal + V	1.7	(0.3)	2.4	(0.3)	1.0 (0.2)	1.4	(0.2)	0.6	(0.1)	-0.7 (0.1)
V10	3rd-pers sing pres	1.3	(0.2)	2.2	(0.3)	0.7 (0.2)	1.2	(0.1)	0.6	(0.1)	-1.1 (0.1)
V11	Past tns modal	1.1	(0.2)	-0.3	(0.2)	-1.5 (0.2)	1.1	(0.2)	-1.5	(0.2)	-2.8 (0.2)
V12	Reg past	1.1	(0.2)	0.4	(0.2)	-1.0 (0.2)	1.2	(0.2)	-1.0	(0.1)	-2.7 (0.2)
V13	Past aux	1.8	(0.3)	0.4	(0.2)	-1.8 (0.3)	1.2	(0.2)	-1.0	(0.1)	-3.1 (0.2)
V14	Medial Adv	1.9	(0.4)	3.4	(0.4)	1.8 (0.3)	1.6	(0.2)	1.2	(0.2)	-0.3 (0.1)
V15	Ellipsis			#					#		
V16	Past cop	1.9	(0.3)	-1.0	(0.3)	-2.1 (0.3)	1.9	(0.3)	-2.5	(0.3)	-3.7 (0.4)
V17	Bound morpheme	1.1	(0.3)	-2.2	(0.3)	-4.1 (0.6)	1.1	(0.2)	-3.0	(0.3)	-4.6 (0.5)
Q1	Intonation			**					**		
Q2	Routine			**					**		
Q3	Simple Neg	1.7	(0.5)	4.8	(0.7)	2.7 (0.4)	1.4	(0.2)	2.3	(0.2)	1.0 (0.1)
Q4	Wh-Q + V			**					**		
Q5	S + neg + V			**					**		
Q6	Wh-Q with S-Aux inversion	4.2	(5.8)	10.9	(11.8)	9.8 (11.8)			**		
Q7	Neg cop, modal or aux	1.3	(0.3)	2.1	(0.3)	0.8 (0.2)			**		
Q8	Y/N Q with S-Aux inversion	0.3	(0.2)	0.4	(0.1)	-0.7 (0.2)	0.2	(0.1)	-0.6	(0.1)	-2.0 (0.1)
Q9	Why, when, which, whose	0.4	(0.3)	-1.5	(0.2)	-4.1 (0.5)	0.4	(0.3)	-2.4	(0.2)	-5.9 (0.9)
Q10	Tag Q			*					*		
Q11	Neg Q with S-Aux inv.	0.5	(0.4)	-2.5	(0.3)	-4.0 (0.6)	0.3	(0.4)	-3.4	(0.3)	-5.8 (0.8)
S1	Two words			*					*		
S2	Subj + V			**					**		
S3	V + Object	2.4	(1.9)	7.3	(2.9)	5.8 (2.4)	1.2	(0.3)	3.1	(0.4)	1.9 (0.3)
S4	S + V + O	3.3	(1.3)	7.2	(2.4)	5.6 (1.7)	1.4	(0.3)	2.6	(0.3)	1.4 (0.2)
S5	Any conjunction	2.9	(1.1)	5.3	(1.1)	4.2 (0.9)	2.5	(0.3)	2.4	(0.2)	1.2 (0.2)
S6	Any 2 V's	3.6	(1.5)	8.1	(2.3)	6.6 (1.9)	2.5	(0.4)	3.4	(0.3)	2.3 (0.3)
S7	Conjoined phrases	1.5	(0.3)	2.0	(0.3)	0.7 (0.2)	1.7	(0.2)	0.1	(0.2)	-1.2 (0.2)
S8	Infinitive	1.9	(0.3)	1.1	(0.3)	0.0 (0.3)	1.4	(0.2)	-0.1	(0.1)	-1.3 (0.2)
S9	Let/Make/Help/Watch	-0.1	(0.2)	-0.1	(0.1)	-1.5 (0.2)	0.1	(0.1)	-1.1	(0.1)	-2.7 (0.2)
S10	Subordinating conj.	3.7	(0.7)	1.6	(0.4)	0.1 (0.4)	3.3	(0.4)	-0.7	(0.3)	-2.3 (0.3)
S11	Mental state V	1.2	(0.2)	-0.9	(0.2)	-2.6 (0.3)	1.4	(0.2)	-2.0	(0.2)	-4.0 (0.3)
S12	Conjoined clauses	2.9	(0.4)	1.0	(0.3)	-0.7 (0.3)	3.7	(0.4)	-1.8	(0.3)	-3.6 (0.4)
S13	If or wh-clause	2.3	(0.4)	0.3	(0.3)	-1.2 (0.3)	2.2	(0.3)	-1.4	(0.3)	-3.2 (0.3)

Appendix A (p. 2 of 3)

Revised Items and Statistical Properties of the Proposed IPSyn-C

Table A1. (Continued).

		Overall scale calibration											
		100-utt sample (N = 388)					50-utt sample (N = 639)						
Item	Description	a (SE)		c ₁ (SE)		c ₂ (SE)		a (SE)		c ₁ (SE)		c ₂ (SE)	
S14	Bitransitive pred.	0.5	(0.2)	-1.4	(0.2)	-3.3	(0.4)	0.5	(0.2)	-2.3	(0.2)	-4.4	(0.4)
S15	3 or more V's	1.6	(0.2)	0.0	(0.2)	-1.6	(0.2)	2.1	(0.2)	-1.8	(0.2)	-3.7	(0.3)
S16	Relative clause	1.9	(0.3)	0.3	(0.2)	-1.2	(0.3)	1.8	(0.2)	-1.2	(0.2)	-2.8	(0.2)
S17	Infinitival clause	0.8	(0.3)	-1.4	(0.2)	-3.2	(0.3)	0.9	(0.2)	-2.3	(0.2)	-5.5	(0.6)
S18	Gerund	0.3	(0.2)	-1.2	(0.2)	-3.6	(0.4)	0.4	(0.2)	-2.3	(0.2)	-4.4	(0.4)
S19	Left or center-embed clause	1.8	(0.4)	-1.6	(0.3)	-3.4	(0.5)	1.9	(0.3)	-2.9	(0.3)	-5.3	(0.5)
S20	Passive or tag	2.48	(2.6)	-7.96	(5.2)	NA		2.36	(1.7)	-7.86	(3.2)	NA	

Note. For more detailed individual item descriptions, please see Altenberg et al. (2018). **Bolded** items were removed after statistical analysis. Table data designators: * = item dropped due to little variability in item responses; ** = item dropped due to unstable model estimation due to quasi-complete separation; # = item dropped due to insignificant slope; NA = items that have only two-category responses. All dropped items for 50-utterance samples are also **bolded**. utt = utterance; a = slope; c₁ = Intercept 1; c₂ = Intercept 2; NP = Noun Phrase; Det + N = determiner + noun; V = verb; 2-wd = two-word; 3-wd = three-word; Adv = adverb; V part or prep = verb particle or preposition; Prep phr = prepositional phrase; cop = copula; Aux/aux = auxiliary; Prog = progressive; 3rd pers sing pres = third-person singular present; Past tns modal = past tense modal; Reg past = regular past tense; Past aux = past tense auxiliary; Neg/neg = negation; Wh-Q = wh-question; S = subject; S-Aux = subject-auxiliary; Y/N Q = yes/no question; Tag Q = tag question; Neg Q = question with negation; inv. = inversion; Subj = subject; O = object; conj. = conjunction; pred. = predicate.

Appendix A (p. 3 of 3)

Revised Items and Statistical Properties of the Proposed IPSyn-C

Table A2. A summed score conversion table for the proposed IPSyn-C in yearly intervals for ages 2–6 years (see Table A1).

Summed			Percentile rank						Summed			Percentile rank					
Score	T score	SE	24–72 mos.	24 mos.	36 mos.	48 mos.	60 mos.	72 mos.	Score	T score	SE	24–72 mos.	24 mos.	36 mos.	48 mos.	60 mos.	72 mos.
0	14.8	4.3	0	0	0	0	0	0	43	54.1	2.7	66	100	73	33	7	1
1	17.0	4.2	0	0	0	0	0	0	44	54.8	2.7	68	100	77	38	9	1
2	18.7	4.1	0	0	0	0	0	0	45	55.5	2.7	71	100	80	43	11	1
3	20.3	3.9	0	0	0	0	0	0	46	56.2	2.7	73	100	84	48	14	2
4	21.8	3.7	0	0	0	0	0	0	47	56.9	2.7	75	100	86	52	16	2
5	23.1	3.5	0	1	0	0	0	0	48	57.7	2.7	78	100	89	58	20	3
6	24.3	3.4	1	1	0	0	0	0	49	58.4	2.7	80	100	91	63	24	4
7	25.5	3.3	1	2	0	0	0	0	50	59.1	2.7	82	100	93	67	28	5
8	26.6	3.2	1	3	0	0	0	0	51	59.8	2.8	84	100	95	71	32	6
9	27.6	3.1	1	5	0	0	0	0	52	60.6	2.8	86	100	96	76	37	8
10	28.6	3.1	2	7	0	0	0	0	53	61.3	2.8	87	100	97	79	41	10
11	29.5	3.0	2	9	0	0	0	0	54	62.1	2.9	89	100	98	83	47	13
12	30.4	3.0	2	12	0	0	0	0	55	62.8	2.9	90	100	98	86	52	16
13	31.3	2.9	3	15	0	0	0	0	56	63.6	3.0	91	100	99	89	57	20
14	32.2	2.9	4	19	0	0	0	0	57	64.4	3.0	93	100	99	91	63	24
15	33.0	2.9	4	23	0	0	0	0	58	65.3	3.1	94	100	99	94	68	29
16	33.8	2.8	5	28	0	0	0	0	59	66.1	3.1	95	100	100	95	73	34
17	34.6	2.8	6	32	0	0	0	0	60	67.0	3.2	96	100	100	96	78	40
18	35.4	2.8	7	37	0	0	0	0	61	67.9	3.3	96	100	100	98	82	46
19	36.2	2.8	8	43	1	0	0	0	62	68.8	3.4	97	100	100	98	86	52
20	37.0	2.8	10	48	<u>1</u>	0	0	0	63	69.7	3.5	98	100	100	99	89	58
21	37.8	2.8	11	54	1	0	0	0	64	70.7	3.6	98	100	100	99	92	65
22	38.5	2.8	13	59	2	0	0	0	65	71.7	3.7	98	100	100	100	94	71
23	39.3	2.8	14	64	3	0	0	0	66	72.8	3.8	99	100	100	100	96	77
24	40.1	2.8	16	68	3	0	0	0	67	73.8	3.9	99	100	100	100	97	82
25	40.8	2.8	18	73	5	0	0	0	68	75.0	4.0	99	100	100	100	98	87
26	41.6	2.8	20	78	6	0	0	0	69	76.1	4.1	100	100	100	100	99	91
27	42.3	2.8	22	81	8	1	0	0	70	77.3	4.2	100	100	100	100	99	94
28	43.1	2.8	25	85	10	1	0	0	71	78.6	4.3	100	100	100	100	100	96
29	43.9	2.8	27	87	12	1	0	0	72	79.8	4.5	100	100	100	100	100	97
30	44.6	2.8	29	90	15	2	0	0	73	81.1	4.6	100	100	100	100	100	99
31	45.4	2.8	32	92	18	3	0	0	74	82.4	4.7	100	100	100	100	100	99
32	46.1	2.8	35	94	22	3	0	0	75	83.7	4.8	100	100	100	100	100	100
33	46.8	2.8	37	95	26	5	0	0	76	84.9	4.9	100	100	100	100	100	100
34	47.6	2.8	41	96	30	6	0	0	77	86.1	5.0	100	100	100	100	100	100
35	48.3	2.8	43	97	35	8	1	0	78	87.3	5.1	100	100	100	100	100	100
36	49.1	2.8	46	98	39	9	1	0	79	88.3	5.2	100	100	100	100	100	100
37	49.8	2.8	49	99	45	12	1	0	80	89.3	5.2	100	100	100	100	100	100
38	50.5	2.8	52	99	50	15	2	0	81	90.3	5.3	100	100	100	100	100	100
39	51.2	2.8	55	99	54	18	2	0	82	91.2	5.4	100	100	100	100	100	100
40	52.0	2.7	58	99	60	21	3	0	83	92.1	5.5	100	100	100	100	100	100
41	52.7	2.7	61	100	64	25	4	0	84	92.9	5.6	100	100	100	100	100	100
42	53.4	2.7	63	100	69	29	6	0	85	93.7	5.6	100	100	100	100	100	100
									86	94.4	5.7	100	100	100	100	100	100

Note. The T score is a standardized score with a mean of 50 and an SD of 10 that is transformed from item response theory scaled scores. If a child received a summed score of 20 (see the bold number line in the table), the corresponding T score is 37.0, and the standard error of the T score is 2.8, which means the 95% confidence interval for the child's score would be $[37 - 1.96 \times 2.8, 37 + 1.96 \times 2.8]$. With respect to the population of typically developing children between 24 and 72 months of age, this child's score is at the 10th percentile. If the child is 24 months old, the percentile rank is 48. If this child is 36 months old, the score falls at the first percentile, and the child can be flagged as a late talker with only 2% of likely error. The bold and underlined scores in the 36th and 48th month columns indicate that the children with those scores are highly likely to be late talkers with only 2% false positive rate. If a score is bold, the child can be flagged as atypical with 80% correct probability, but also with a larger probability (40%–50%) of false diagnosis. mos. = months.

APPENDIX 6: Interpretation of VocD

From Duran, P., Malvern, D., Richards, B., & Chipere, N. (2004). Developmental Trends in Lexical Diversity. *Applied Linguistics*, 25(2), 220-242.

Table 2: Descriptive statistics for D (stem forms) at each age

Age (months)	N	Mean	Std. Dev.	Median	Min.	Max.
18	18	14.80	10.31	13.60	1.48	36.99
21	20	21.49	16.70	19.09	2.60	67.24
24	28	27.44	20.52	25.41	2.50	84.64
27	29	34.77	17.70	31.16	7.48	65.76
30	29	41.53	16.93	45.59	4.05	69.67
33	29	43.67	15.45	45.47	10.38	73.88
36	29	47.83	13.97	47.14	13.26	69.95
39	30	49.48	15.41	49.08	11.22	80.78
42	29	53.12	13.55	53.80	10.57	73.54
60	15	64.02	8.46	63.48	50.83	83.30

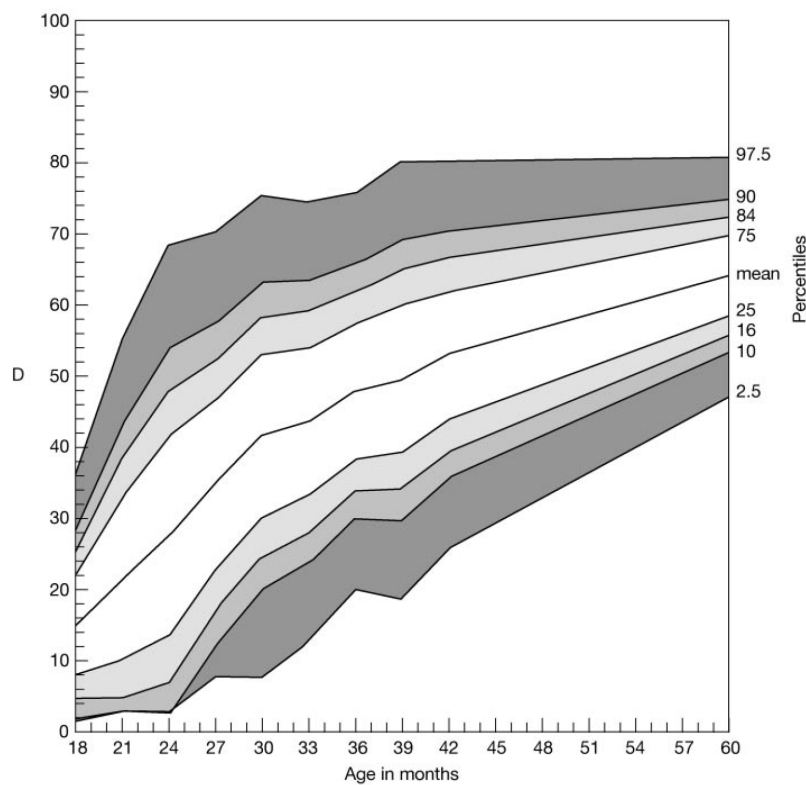


Figure 2: Values of D against age for selected percentiles derived from the Bristol cohort

APPENDIX 7: Coding Stuttering and Other Fluency Behaviors

A new set of codes has recently been developed for stuttering behaviors and will be included in all new issues of CLAN. The two new symbols are in Unicode and have shortcut combinations that work when you are transcribing in CLAN⁶. The complete list of stuttering/disfluency codes are:

Stuttering-like disfluencies (SLDs)	Code	Example	Notes
prolongation	:	s:paghetti	Place after prolonged segment
broken word	^	spa^ghetti	Pause within word
blocking	≠	≠butter	A block before word onset
repeated segment	←P	←Pr-r-r←Prabbit like←Pike-ike←P	The ←P brackets the repetition; hyphens mark iterations
lengthened repeated segment	←P and doubling	←Prr-r-r←Prabbit	The doubling of "r" indicates lengthening of the "r" segment
word repetition	[/]	dog [/] dog	Further distinctions are done inside FluCalc
Typical Disfluencies (TDs)	Code	Example	Notes
phrase repetition	<> [/]	<that is a> [/] that is a dog.	< > is used to mark repeated material
word revision	[//]	a dog [//] beast	Revision counts once
phrase revision	<> [//]	<what did you> [//] [//] how can you see it ?	Revision counts once
phonological fragment	&+	&+sn dog	Changes from "snake" to "dog"
pause	(.) or (..) or (...)	(.)	Counts the number of short, medium, long pauses
pause duration	(2.4)	(2.4)	Adds up the time values, if marked
filled pause	&-	&-um &-you know	Fillers with underscore count as one word

The ≠ character to mark blocking is entered by typing F2 and =

The ←P character to mark segment repetition is entered by typing F2 and /

Blocking of filled pauses is indicated in this way: &-≠you know

Blocking of repeated segments is indicated in this way: ←P≠in←Pinto

For both of these, the ≠ symbol must be placed in this specified order

⁶ In CLAN, to see short cuts for any phonetic, prosodic or fluency codes (in case you lose this ☺), go to WINDOW → Special Characters, and a drop-down menu will appear that you can scroll through to find the short cut. Please note that these key stroke combinations work inside CLAN and are **not** the key combos for typing in Word or other programs.

Tallying stuttering behaviors

Using the above codes ensures that when you transcribe, and analyze your transcript, neither **FREQ** nor **MOR** (the syntactic program) will include filled pauses as words (stuttered words will be counted as intended words, which is correct).

***SEE the “FLUENCY ANALYSES IN CLAN” section above
for the current command to tally disfluencies.***

To get %stuttered words, **FREQ** your file, and then the **output of the fluency tally** can be put over the **token total** to give you %blocks, %prolongations, etc.

To tally fluency behaviors, **CLAN** uses a file that looks like this [fluency-sep.cut], in the **LIB** folder, inside its **Fluency** folder, with these symbols, one per line⁷:

&-*

←ρ

≠

^

.

[/*] or </*> depending on whether you are counting occurrences of repetitions or words themselves.

[x *]

The only reason why we are reprinting it here is that some people want to either track or **NOT** track certain behaviors. You can add a behavior, such as hesitations, or fillers, by simply adding lines to this file and saving it. You can also remove a behavior if you don't think it is relevant to your fluency work.

NOTE: Remember that you can insert the symbol codes from the **Windows** (in the **CLAN** menu bar) drop down option list using “special characters”; if you are typing or inserting and the block or repetition code enters as something other than what you want (for example, a box, rather than the curly arrow), you need to set your default **CLAN** font so it accesses **Unicode**. Go to **View**, and set font, and click a specific font. The best option is **Arial Unicode** or **CA** font, but if you don't see those, any font followed by **Unicode** (**Lucida**) will do.

⁷ The asterisk is very powerful, and is a wild card that allows the program to find the code regardless of what letter combination it is next to. **CLAN** uses a lot of asterisks, as do many computer programs, and search routines, such as those in **PubMed**, **EBSCO**, etc. Try it if you haven't ever done this.

APPENDIX 8: CLAN DSS Command Output

```
dss +t*CHI +leng
Tue Jan 14 19:02:41 2020
dss (13-Jan-2020) is conducting analyses on:
  ONLY speaker main tiers matching: *CHI;
  and those speakers' ONLY dependent tiers matching: %MOR;;
*****
From file </Users/courtneyoverton/Desktop/50DDA.cha>
```

Developmental Sentence Analysis

Sentence	IP	PP	MV	SV	NG	CNJ	IR	WHQ	S	TOT
<unhunh> [//] her can't fit her clothes .		2 2	1		4				1	10
<a> [//] I don't know .		1	4 1		4				1	11
is he holding <&-uh his> [//] his pan ?		2 2	1						1	6
he die .		2	1						1	4
move the boat .			1						1	2
putched [: pushed] him .		2	2						1	5
pushed him on the boat .		2	2						1	5
I can't do it .		1	1		4				1	7
I'm a little kid .		1	2						1	4
no I don't know .		1	4 1		4				1	11
then spread it through there .			1						1	2
&-uh make a peanut butter .			1						1	2
put it in the peanut butter .			1						1	2
put it on that too .	1		1						1	3
yeah put it in jelly .			1						1	2
and then &-uh spread it on there .			1			3			1	5
put it on top of the other .			1						1	2
I'm gonna make it good .		1	2 1 2						1	7
<&-uh a> [//] get a boon [: balloon] .			1						1	2
cause he gon(na) lost it .		2	2 2						1	7
<cause> [//] (be)cause he's gon(na) lost it .		2	2 2			6			1	13
<&-uh he he> [//] he steal his boon [: balloon] .		2 2	1						1	6
he hit the boon [: balloon] .		2	1						1	4
ah, he lost .		2	2						1	5
then he had went home .		2	2 2						1	7
<cause cause cause > [//] (be)cause he mean .		2	1			6			1	10
<where &-uh> [//] where &-uh we'll color ?		3	4 1						1	9
(be)cause I'm being bad .		1	2			6			1	10
daddy whoop me .		1	1						1	3
<unhunh> [//] it is Derek .			1						1	2
<ain't got no> [//] ain't got no wed [: red] .	3		2 2		7				1	15
find red .			1						1	2
color the door red ?			1						1	2
it is gwee [: green] .			1						1	3
there's the red .			1						1	2
can I take my paper home ?		1 1	1						1	10
color him .		2	1						1	4
color the grass .			1						1	2
<I I I> [//] I writed it .		1	2						1	4
color the bike .			1						1	2
I brought it blue .		1	2						1	4
color it in pwant [: plant] .			1						1	2

I don't got no crayons .
 I get hurt .
 I <hurt> [/] hurt on my knee .
 we're gonna color this ?
 fly the airplane .
 fly an airplane .
 make pizza .
 make a cake .
 TOTAL

3	1	4	2	4				1	15
	1	7						1	9
	1	1	1					1	4
1	3	2	1	2				1	10
		1						1	2
		1						1	2
		1						1	2
		1						1	2
	8	53	92	8	27	21	7	0	266

Developmental sentence score: 5.32

APPENDIX 9: CLAN IPSYN Command Output

Ipsyneng +t*CHI
Mon Mar 14 14:15:19 2022
ipsyn (10-Feb-2022) is conducting analyses on:
ONLY speaker main tiers matching: *CHI;
and those speakers' ONLY dependent tiers matching: %MOR;; %GRA;;

From file <c:\talkbank\clan\work\Ratner\intake\S002_intake.cha>

*** Speaker: *CHI:
S002_intake.cha eng|Ratner|CHI|2;08.|male|Intake||Target_Child||

Rule: N2
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 132.
Point1: pro:int|where
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 152.
Point2: pro:per|it
Score: 2

Rule: N5
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 83.
Point1: det:art|a n|+n|birth+n|day
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 152.
Point2: det:art|a n|back
Score: 2

Rule: N6
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 152.
Point1: v|have det:art|a n|back
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 186.
Point2: v|make det:art|a n|window
Score: 2

Rule: N7
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 312.
Point1: n|+adj|french+n|fry-PL
Score: 1

Rule: N8
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 83.
Point1: det:art|a n|+n|birth+n|day v|party
Score: 1

Rule: N9
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 386.
Point1: qn|all det:art|the n|+adj|french+n|fry-PL
Score: 1

Rule: N10
Score: 0

Rule: N11
Score: 0

Rule: V2
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 186.
Point1: prep|for
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 256.
Point2: prep|over
ADD: score: 2 from V3
Score: 2

Rule: V3
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 186.
Point1: prep|for n:prop|Lucy_Lane
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 280.
Point2: prep|on det:art|a n|picnic
Score: 2

Rule: V4
Score: 0

Rule: V6
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 378.
Point1: aux|be&3S part|go-PRESP
Score: 1

Rule: V7
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 378.
Point1: part|go-PRESP
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 627.
Point2: part|drive-PRESP
Score: 2

Rule: V8
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 72.
Point1: adv|right
ADD: score: 1 from V8
Score: 2

Rule: V9
Score: 0

Rule: V10
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 378.

Point1: v|+adj|french+n|fry-3S
Score: 1

Rule: V11
Score: 0

Rule: V12
Score: 0

Rule: V13
Score: 0

Rule: V14
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 617.
Point1: v|put&ZERO adv|right
Score: 1

Rule: V151
Score: 0

Rule: V152
Score: 0

Rule: V153
Score: 0

Rule: V16
Score: 0

Rule: V17
Score: 0

Rule: Q3
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 152.
Point1: neg|not
ADD: score: 1 from Q5
Score: 2

Rule: Q5
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 152.
Point1: pro:per|it neg|not v|have
ADD: score: 1 from Q7
Score: 2

Rule: Q7
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 152.
Point1: mod|do&3S neg|not v|have
Score: 1

Rule: Q8
Score: 0

Rule: Q9
Score: 0

Rule: Q11
Score: 0

Rule: S3
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 453.
Point1: v|yum n|yum
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 453.
Point2: v|yum n|yum
ADD: score: 2 from S4
Score: 2

Rule: S4
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 413.
Point1: pro:sub|we v|eat det:art|a n|tomato
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 453.
Point2: n|yum v|yum n|yum
Score: 2

Rule: S5
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 183.
Point1: coord|and
Score: 1

Rule: S6
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 378.
Point1: v|eat v|+adj|french+n|fry-3S
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 453.
Point2: v|yum v|yum
ADD: score: 3 from S8 S9
Score: 2

Rule: S7
Score: 0

Rule: S8
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 378.
Point1: aux|be&3S inf|to v|eat
File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 599.
Point2: aux|be&1S inf|to v|put&ZERO
Score: 2

Rule: S9

File "c:\talkbank\clan\work\Ratner\intake\S002_intake.cha": line 186.

Point1: v|make det:art|a n|window

Score: 1

Rule: S10

Score: 0

Rule: S11

Score: 0

Rule: S12

Score: 0

Rule: S13

Score: 0

Rule: S14

Score: 0

Rule: S15

Score: 0

Rule: S16

Score: 0

Rule: S17

Score: 0

Rule: S18

Score: 0

Rule: S19

Score: 0

Rule: S20

Score: 0

N = 9

V = 11

Q = 5

S = 10

Total = 35