**CDP++.parser FAQ (slightly updated from CDP++)**

**Files**

All files should be in the zip file. You need to keep the “EnglishRes” directory.

**Running the Model**

The model runs in the same way as CDP+. Just click on the .exe.

You need the following files in the same directory for it to work:

EnglishOnsets.txt, EnglishCodas.txt, EnglishVowels.txt, SplitNetCrossW.29.txt, EngTrigrams.txt, NewLex4.3.2.txt, FinalNetCross.39.txt, 2009.great.txt

**Getting a word to run**

Go to RUN in the menu, type the word you want to run and press okay. It has to be UPPER CASE, otherwise the program will crash. It will also crash if you put in symbols that are not the 26 letters of the English Alphabet.

There is currently a button called “Fast” which is automatically selected. If you use this button, the results will be similar although not perfectly identical to the full model. It will also run vastly faster. Basically, this button restricts the lexicon to words that have an identical orthography to the input.

**Running a batch file**

This is the same as CDP+. First you need to write in a name for where you want the results. To do this, go to SAVING->QuickSave, and type in a name (e.g., ConradResults.txt”). Then, go to RUN->RunBatchNaming, and choose the batchfile you want (an example of the setup you need is given in the file DRPaap.txt). A screen will then appear asking you to type in a name of a word. You have to type in something here as if you were running the word online, but it won’t make any difference to anything. Now just wait… and wait…

The batch file must first have “naming” as the first word, and the maximum number of cycles words can run for before processing is terminated.

naming

BURY 300

CASTE 300

COMB 300

**Running a batch priming file**

Masked priming can only be done in batch form. It works in the same way as normal batch priming. You need to set up your file first with the prime word, next with the number of cycles you want it to remain for, next the number 0 (yes, this needs to be in there in the current version), and finally the maximum possible number of cycles the word will run for.

naming

TAKE 15 0 PEAR 150

**Finding out about interesting stuff that is going on.**

If you want to find out about the nitty-gritty details about what is going on, use “SAVING->Interest File” and type in a base name (e.g., dog). You will then get lots of little files with activation values in them that are over 0.02. Note that this will slow things down a lot. Also note that stress will appear in Phoneme 15 as either “{“ = 1st position stress (trochaic) or “\_“ = 2nd position stress (iambic).

**Using Graphs**

You can try using the graphs by clicking up the graphs you want to see and clicking the “Save Graph Info” button on the running screen. Note that the stress activation will appear in phoneme 15. They will also crash now and then. Don’t run words for more than 250 cycles or use parameters where lots of neighbors get activated. Running graphs will make things even slower.

**Other Stuff**

Q1. Is it really this slow?

A1. Yes

Q2. Is there anyway to run it faster?

A2. Use the fast button for everything. The main cases where the results differ in important ways are with simulations that require competitors (e.g., dyslexia).

Q3. Can I change the database or take words out

A3. **NO**. If you take words out, the program may crash or give you results that are not correct. Use the “fast” button if you want to make things go faster

Q4. The frequencies look a bit different to CELEX for some words

A4. All the summing of the homophone frequencies was done in the database, not online.

Q4. What information is presented in the running window when I run a single word?

A.

Cycles Taken to run the Word,

The phonology,

Stress Output Node 1 activation,

Stress Output Node 2 activation,

which node had maximum stress,

stress activation at the first stress spot of the TLA network,

stress activation at the second stress spot of the TLA network,

Average activation of the two possible outcomes for ambiguous cases of the letter –e (numbers closer to 1 equals –e being used as a coda, not a vowel).

Whether the sequence was parsable (1 = yes),

whether a dead-node was hit.

Q5. What information is presented in the running window when I run a single word?

81 ABBEY+++++++++++ {bI XXXXXX 0.917 0.139 1 0.474 0.000 1 0 1

199 IRSABE++++++++++ 3s1b XXXXXX 0.091 0.842 2 0.001 0.473 1 0 1

A.

Cycles taken to run the word,

the word presented,

the phonology generated,

an unused spot,

the stress value of the first stress node in the TLA network,

the value of the second stress node in the TLA network,

the stress node in the output layer with the maximum activation,

the actual value of the first stress node at the output layer,

the actual value of the second stress node at the output layer,

whether the word was successfully parsed,

whether a dead node was hit,

whether the word had finished processing by the maximum number of cycles given,

The output may also be a list of X’s which means that the model could not process the word for some reason or another (e.g., bad characters).

**What are all of these other files?**

It is very hard to work out exactly what is causing what and why if it is not clear what the graphemes always are. Therefore, the model will print out the list of grahpemes, their number, and what category they are in (onset, vowel and coda).

There are also some files the model will print out if a dead node is encountered.

**Where are the stress nodes to be found?**

These currently appear in the final phoneme slot.