Computer Pidgin Language: A new language to talk to your computer?

Stephen Hinde, Guillaume Belrose
Internet Systems and Storage Laboratory
HP Laboratories Bristol
HPL-2001-182
July 30th , 2001*

E-mail: {Stephen_Hinde, Guillaume_Belrose}@hpl.hp.com

CPL, voice speech, telecoms

This paper explores a new concept called Computer Pidgin Language (CPL). This is a radical new approach to dealing with the problem of humans talking to computers. The new approach is to teach people a new language that is efficient for dialogues with computers - a sort of artificial spoken language. We see this as being analogous to how people learn scribble on a PDA. We explore in this paper the motivation for CPL from the appliance, e-service, and infrastructure perspective. We explore some early results from a proof of concept demo that we have built to test these ideas. We also explore some of the wider implications of CPL and longer-term research directions.

* Internal Accession Date Only Approved for External Publication

Computer Pidgin Language:
A new language to talk to your computer?

Stephen Hinde and Guillaume Belrose

13th July 2001

HP Laboratories

Filton Road, Stoke Gifford, Bristol, BS34 8QZ, UK.

{Stephen_Hinde, Guillaume_Belrose}@hpl.hp.com

Abstract

This paper explores a new concept called Computer Pidgin Language (CPL). This is a radical new approach to dealing with the problem of humans talking to computers. The new approach is to teach people a new language that is efficient for dialogues with computers – a sort of artificial spoken language. We see this as being analogous to how people learn scribble on a PDA. We explore in this paper the motivation for CPL from the appliance, e-service, and infrastructure perspective. We explore some early results from a proof of concept demo that we have built to test these ideas. We also explore some of the wider implications of CPL and longer-term research directions.

1. An Introduction to Computer Pidgin Languages

CPL or Computer Pidgin Language is a radical departure from the normal approach to Speech Recognition Systems. CPL is inspired by a frustration at a perceived lack of progress in Spoken Language Research over the last 20-30 years. The authors believe that systems that only understand people 85% of the time are hardly usable, so speech recognition is very much a last resort technology or a curiosity. This led us to reflect on what could we do which is radically different to improve this?

The radical departure we are proposing is that instead of training a computer system to recognize human speech, we could train the human to speak a new spoken language that is optimized to maximize the efficiency of Automatic Speech Recognition. We call a language of this type a Computer Pidgin language or CPL.

One of the inspirations for thinking about CPL as an approach to spoken language recognition is observing the evolution of handwriting recognition. There was little progress in the field of handwriting recognition over a period of 20-30 years, until the move to a technique called "Scribble Matching"[17]. "Scribble Matching" was first introduced on the Apple Newton in 1993 and was later reintroduced in a much better form on the Palm™. The innovation came with the realization that handwriting could be simplified to a series of standard strokes that could be taught to a human, and which a computer could then recognize easily. This turned handwriting recognition from a long-term research area to a commercial technology. Our question is if a similar step is possible in the field of speech recognition.

The authors of this paper have started some initial work in this direction to assess whether CPL languages do exist by performing some simple proof of concept experiments.

The simplest class of CPL would consist of a small vocabulary language for use on a mobile phone, a child's toy, an appliance or PDA. Essentially command and control languages. Its is this class of CPL language we have considered in our initial proof of concept

experiment, and which we will discuss later in this paper.

However we are aware that potentially there is a vast field of long-term research into the area of CPL, the more long-term questions raised by CPL concern thinking about deriving Artificial Languages with similar complexity to human languages. Linguists have been deriving Artificial Languages for many years, but as far as we are aware no one has worked on a language for talking to Computers. The motivation for doing this would be to create the spoken language of Cyber Space that human beings and machines could use to converse: the "Latin" of the cyber scholar.

The early slave traders derived simple "Pidgin" languages for talking to slaves, as they believed the slaves to lack the intelligence to learn Western languages. This inspired our name Computer Pidgin Language. The paradox is that these languages have evolved to form highly grammatically complex languages – the Creole languages. So maybe in the future there will be CCL – Computer Creole Languages.

Human languages are constantly evolving and have been an integral part of evolution. With the moves towards Genetic Algorithms and Programming it is interesting to speculate whether CPL can also undergo evolution and optimization as it goes. Our initial work in this area has used GA techniques to find CPL vocabularies. This GA approach to ASR has proved interesting.

There are however many interesting anthropological, social, linguistic, cultural and psychological questions that would require answering if CPL were to progress towards wide spread acceptance. Many of these questions would be better answered by the academic community rather than by our small research group so our intention is to stimulate other communities to think about the CPL concept.

2. A review of Appliance, Infrastructure, E-Service integration

There is much speculation by Mobile Service Providers of the rich new possibilities brought about by the intersection of Appliances, E-services and Computer Systems linked by "the always on infrastructure". In this section we are going to argue that the interface to the human is in danger of being the "weakest-link" in this world.

In the PC world, the "Windows" human computer interface is now common currency amongst the computer literate community – but the interface is far from intuitive and excludes a large section of the population.

In the world of small mobile appliances there is no effective interface for accessing the Internet, WAP, DTMF phone and small PDA interfaces relatively new interfaces and have their limitations. For many years Spoken Language System (SLS) researchers have advocated speech as a rich and natural method of interaction with small devices. However 30-40 years of Spoken Language Systems research has still has only lead to inadequate and unsatisfactory results.

The new driver of mobility and appliance computing is creating a strong business pull for an efficient human computer interfaces – however there is this strong tension between the humans ability to communicate and the computers ability to understand – figure 1.This represents where we believe innovation is required breaking this tension.

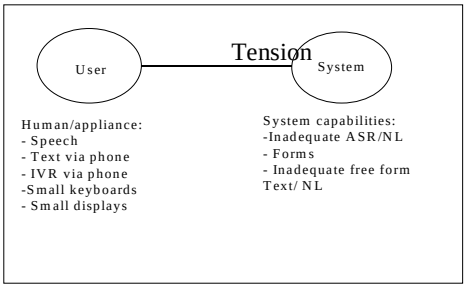

Figure 1: Computer dialog system inefficiency measured against efficient appliance based human interfaces.

The current largest application of Speech technology is to the Interactive Voice Response (IVR) market. The high cost of human operators in the call center has motivated call center operators to deploy IVR systems. However the interfaces remain very simple, the set up costs are high, and usability is less than optimum.

With the convergence of Internet and Telecom technologies there has been a webifying of the IVR story and IVR systems have been renamed VoiceWeb

systems[11] or WebIVR. These new systems offer improved interfaces to the Web back end technologies and a standard Markup language for authoring IVR applications. However the basic speech technologies remain the same so that efficiency of the human interface remains poor.

Science fiction has led popular imagination to expect us to talk to computers like in Star Trek or other science fiction – figure 2. There seems no evidence in current research that we are even close to making this happen.

Captain Kirk: "Computer…."

Computer Voice: "Yes Captain"

Captain: "What can you tell me about the Phaos system?"

Computer Voice; "B-type planet, can support human life, 3 moons"..

Figure 2: Star Trek computer interface.

The MIT Galaxy system represents the State-of-the-Art in terms of advanced research systems. These systems move speech systems into the realm of free conversation or "mixed initiative". They feature domains of discourse such as talking about weather or travel[1][2].

The most advanced SLS in the best research groups in the world such as the DARPA Communicator system and the MIT Galaxy system should give us a glimpse of what will be possible commercial systems in say 10 years. However these systems still suffer inadequacies and problems, in terms of high set up cost, errors, and limited domains of discourse[4].

3. Artificial languages

All language is man-made, but artificial languages are made systematically for some particular purpose. They take many forms, from mere adaptations of an existing writing system (numerals), through completely new notations (sign language), to fully expressive systems of speech devised for fun (Tolkien) [8], or secrecy (Poto and Cabenga) or learnability (Esperanto)[9]. There have been artificial languages produced of no value at all such as Dilingo [10] and artificial language toolkits[5].

Another artificial language, was invented by Dr. Ludwig L. Zamenhof of Poland, and was first presented to the public in 1887. It has enjoyed some recognition as an international language, being used, for example, at international meetings and conferences. The vocabulary of Esperanto is formed by adding various affixes to individual roots and is derived chiefly from Latin, Greek, the Romance languages, and the Germanic languages. The grammar is based on that of European languages but is greatly simplified and regular. Esperanto has a phonetic spelling. It uses the symbols of the Roman alphabet, each one standing for only one sound. A simplified revision of Esperanto is Ido, short for Esperandido. The French philosopher Louis Couturat introduced Ido in 1907, but it failed to replace Esperanto.

As far as we are aware there hasn't been an artificial spoken language produced for its ability to talk to computers.

4. E-Inclusion and Computer Dialogues

E-Inclusion is a current term that is being used to talk about how to give people of varying cultural diversity and social backgrounds access to E-Services. We observe that currently there are some 60,000 plus active languages in the world. In fact linguists have no idea as to what is the exact number of active languages. The current approach to E-Inclusion is "Localization" or to force people to learn English or some other widely spoken language. One approach to Spoken Language E-Inclusion is to advocate an international language, which would be universally available around the world to talk to devices. If this spoken language was more efficient for talking to computers than the existing language people might be motivated to learn it.

5. Human languages and protocols

Humans have an innate ability to learn complex languages. This is one of the prime differentiators of the human species over other species – their ability to use complex language structures this ability is markedly strong in children and at later ages varies according to environment and individual capability. It is interesting to speculate how languages evolved. We can obviously see the origins of language in more primitive species but also we could think of languages as being a particular form of protocol, and the all species have a very strong ability to learn and work with protocols – systems of operating outside their immediate bodies. Our hypothesis is that the human being is a highly evolved protocol engine, where as the computer is a new boy on the block in terms of

evolution of protocols. Interestingly with the move towards genetic algorithms (GA) and genetic programming (GP) we see computers starting to use evolutionary methods to work with protocols. We have started to use GA's in our design of CPL.

6. Creation of a CPL vocabulary

This part describes a method that was put in place to create small size CPL vocabularies. Such a vocabulary can be used to control a device with simple commands. It consists of a set of words that is designed to optimize the efficiency of automatic speech recognizer.

6.1. Data representation

A CPL word is represented as a sequence of *p* consonant vowel units represented with the ARPAbet notation[18]. We use a subset of the British English phone set that does not contain the following phones: "ia" as in peer, "ea" as in pair, "oh" as in pot and "ua" as in poor. These phones are not used in American English.

ARPAbet	Example	ARPAbet	Example
B	But	Iy	Bean
P	Put	Ih	Pit
D	Den	Ae	Pat
T	Ten	Aa	Barn
G	Game	Ah	Putt
K	Can	Ao	Born
F	Full	Ay	Buy
V	Very	Ax	About
S	Some	Ey	Bay
Z	Zeal	Eh	Pet
Dh	Then	Er	Burn
Sh	Ship	Ow	No
L	Like	Aw	Now
R	Run	Oy	Boy
Y	Yes	Uh	Good
W	Went	Uw	Boon
Hh	Hat	Ng	Long
M	Man	Ch	Chain
N	Not	Jh	Jane

Figure 3 Phonetic alphabet.

6.2. The optimization problem

We use a genetic algorithm ([15][16]) exploring the space of phonemes to find a vocabulary with the lowest confusability. The GA manipulates a population of *N* individuals. Each individual contains a DNA string coding a set of *k* words (i.e. a vocabulary V). The DNA string is a sequence of *k*p*2* phonemes, for example "f'aa'dh'er k'aa's'ay" (the symbol ' is used to separate two phonemes).

We randomly create an initial seed of *N* individuals. Each individual is evaluated by a fitness function. After evaluation the *N* fittest individuals are selected to form the next generation of the population. We use a ranking selection algorithm whereby the probability of selecting an individual is related to its rank within the population. The crossover exchanges for two individuals fragments of DNA cut around a cross over point selected randomly. The mutation operator replaces a consonant by a consonant, and a vowel by a vowel. The frequencies of cross over and mutation are controlled by probabilities.

6.3. The fitness function

The fitness function measures the confusion of the vocabulary created by an individual. The goal of the GA is to minimize this function.

We dispose of a phoneme confusion matrix A generated from recognizing a training set of British English utterances with the ABBOT speech recognizer. This recognizer is a hybrid RNN/HMM large vocabulary continuous speech recognizer that was developed by Cambridge University and University of Sheffield.

This matrix provides the conditional probability $a_{ji} = pr(y = p_i \mid x = p_j)$ of recognizing a phoneme as p_j when it is actually p_i.

We created a confusion function $conf(p_i, p_j)$ with the following properties:

$$conf(p_i, p_i) = 1$$

$$conf(p_i, p_j) = a_{ji}$$

$$if\ (a_{ji} = 0) conf(p_i, p_j) = \varepsilon$$

For some entries in the matrix, the confusion between two phonemes can be null. However, in practice, there still is a probability of these phonemes to be confused. We set this probability to a small value ε ($\varepsilon = 0.0001$).

We propose two methods to evaluate the confusion between two words from the vocabulary. Given two words $Wi\{p_{i1}, p_{i2},...p_{i2p}\}$ and $Wj\{p_{j1}, p_{j2},...p_{j2p}\}$, their confusion can be the sum or the product of the confusion of the phonemes composing these words.

$$(A) conf(W_{i,}W_j) = \prod_{t=1}^{2*p} conf(p_{it}, p_jt)$$

$$(B) conf(W_i,W_j) = \sum_{t=1}^{2*p} conf(p_{it}, p_{jt})$$

We proposed three methods to evaluate the overall confusion of a set of words. For a given vocabulary V, its confusion can be the average confusion of the words composing this vocabulary, or the total confusion of all the words from the vocabulary, or the worst (i.e. highest) confusion.

$$(C) conf(V) = \sum_{i=1}^{k} \sum_{j \neq i} conf(Wi,Wj)$$

$$(D) conf(V) = \frac{1}{k(k-1)} \sum_{i=1}^{k} \sum_{j \neq i} conf(Wi,Wj)$$

$$(E) conf(V) = \max(conf(w_i, w_j)), i \neq j$$

6.4. Constraining the evolution

While generating words, we noticed that the GA produced words that were not easy to pronounce. We added arbitrary constraints to the structure of the words in order to tackle this problem.

- Only one diphthong is allowed per word. Diphthongs are: ‘ey’, ‘ay’, ‘oy’, ‘ow’, ‘aw’.
- Short vowels are not allowed at the end of the word. Short vowels are ‘aa’,’ao’,’ih’,’eh’,’ae’,’ah’,’uh’ and ‘aw’.
- The phones zh and th are not used
- The phone ng is not a valid starting consonant.
- The phones r, y and w are not allowed either side of a diphthong.
- The phone hh is not allowed as a second consonant.

If an individual does not respect one of these constraints, it receives a penalty that increases its fitness and reduces its chance of surviving the selection process.

6.5. Results of the evolution

We created different sets of 26 words with various parameters and confusion matrices. In all cases, the algorithm quickly converges towards a stable solution.

The graph below shows the evolution of the best individual fitness for a population of 5000 individuals that evolved during 300 iterations. The GA converges to a solution where the fitness of the best individual is 0.011036.

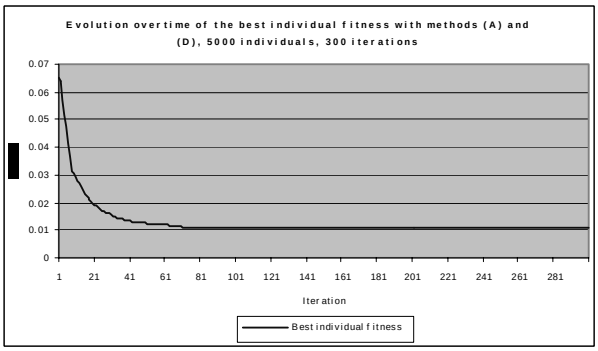

Figure 4: Evolution overtime of the fitness of the best individual in the population.

We devised a batch mode program that runs a certain number of times the GA with the same parameters. We use it to determine whether or not the GA converges towards similar solutions. The batch mode ran 90 times with a population of 5000 individuals evolving during 500 iterations with the methods (A) and (C). The graph indicates that the results of the GA are consistent overtime.

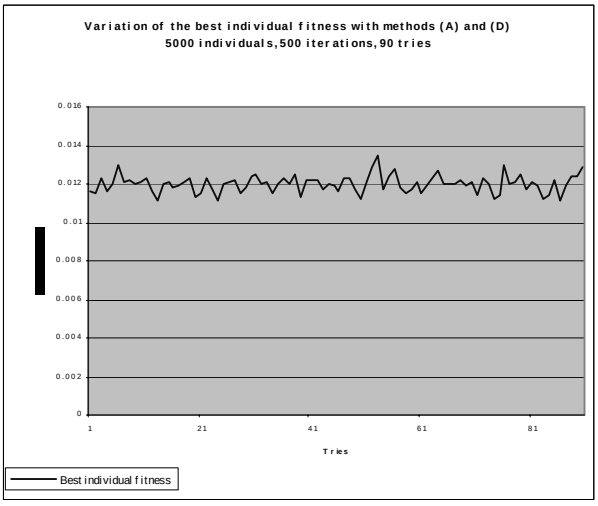

Figure 5: Comparison of the fitness of the best individual for 90 tries.

6.6. Experiments

For the proof of concept, we selected a CPL set created with the methods (A) and (C) and compared its efficiency against the International Spelling Alphabet.

CPL	CPL	Spelling alphabet	Spelling alphabet
Zeejoy	Failu	Alpha	November
Highma	Seree	Bravo	Oscar
Wooper	Farngoy	Charlie	Papa
Gowka	Poingy	Delta	Quebec
Zappay	Wass-eye	Echo	Romeo
Neefa	Kozer	Foxtrot	Sierra
Yeffoy	Persha	Golf	Tango
Pooboy	Loicher	Hotel	Uniform
Ggeamay	Showshu	India	Victor
Jower	Hoochay	Juliet	Whiskey
Juicy	Shoiby	Kilo	X-ray
Mgu	Saanga	Lima	Yankee
Shuki	Norshoy	Mike	Zulu

Figure 6: CPL set and International Spelling alphabet.

We recorded six speakers reading all the words from the CPL vocabulary and the international spelling alphabet. Among the six speakers, five were British English speakers, and one was a native French speaker. Among the British English speakers, one was a female speaker. The waveforms were passed to the ABBOT speech recognizer for analysis.

Speakers	Spelling Alphabet	CPL set
Male Speaker 1	2	2
Female Speaker	1	3
MS2	2	0
MS3	1	1
MS4	4	1
MS5	3	0
Total errors	13	7

Figure 7: Recognition results

These early results show that the CPL set performs better than the English equivalent, with a number of errors reduced by half. Nevertheless, if these early results sound promising, further experiments are required to prove the efficiency of CPL over English.

7. CPL Applications

7.1. Overview.

CPL can be used to control and interact with simple speech enabled devices. With a phone for instance, the user could say funny sounding sentences to send SMS messages to their friends (like “juicy failu”). We thought of speech-enabled toys, like virtual dogs that can could walk, bark, execute tricks, move and communicate with their masters using CPL.

We strongly believe that there is a place for CPL in the children and teenagers market, where the users are actually motivated to learn new ways of communicating to enjoy fun experiences.

7.2. The CPL phone

Figure 8: Graphical User Interface of the CPL Phone

As a proof of concept, we devised a software simulation of a simple hands-free phone using the Microsoft synthesizer and recognizer. This phone allows the user to perform actions such as answering or ending a call, dialing a number or sending pre defined SMS messages to recipients stored in a directory. In total, a vocabulary of 24 words was used to control all the functionalities.

The CPL phone demo provides an English and a CPL version (this set of commands is a cut down version of the CPL set). The authors of the demo made an arbitrary mapping between the CPL words and the corresponding commands. The user is free to switch between the two languages at any time and can access help in the course of the dialogue.

He can also take part in an interactive tutorial to learn the language. This tutorial involves two animated cartoon characters called Mrs. and Mr. Mike, who teach the user the CPL words by saying (via Text to Speech) and displaying them on a screen.

8. Further development

As it stands today, the CPL concept is still at an early stage. The word generation methods put in place promise interesting results. However, further experiments are required to demonstrate the whole potential of CPL.

8.1. Further experiments

First of all, we need to put in place more tests with users in order to be more confident on how CPL is better than English in terms of recognition accuracy.

By setting up user studies, we could collect answers to questions such as “How easy is it easy to use and learn CPL?”, or “Would you like to use it in your everyday life?” and tackle the problem of usability of such artificial languages.

Nevertheless, if a demo such as the CPL phone proves to give good results and to be a fun and novel experience, we are pretty confident that the user would be motivated to learn and use the language.

8.2. New research areas

During the development of the proof of concept, we came across interesting research problems.

8.2.1. Cool words

We noticed while running the word generation experiment that some words produced by the GA sounded quite familiar, “cool” or funny. Words such as “juicy”, “wooper” or “wass eye” fall in this category, and we actually found out that they are quite easy to pronounce and remember. On the other hand, the CPL vocabulary contains words like “farngoy” or “poingy” which are less attractive to the ear and therefore more difficult to use.

The idea of “cool sounding” words is quite difficult to grasp and formalize in some criteria that can be used by a computer program. However, what we propose is for the user to set a list of favorite words he likes to listen to (like “hey man”, “what’s up”, “see ya”). The fitness function of the GA can be modified to create words that are similar sounding to those from the favorite list (for example using phone correlation techniques [19]).

8.2.2. Spoken scribble English

Our approach to CPL was to create a completely new language that people would have to learn and practice.

We are aware that many people would not have the motivation or the time require to learn yet another language. We could take a similar approach to scribble as it used on PDA. Scribble does not force the user to learn a new written language. It just encourages the user to modify the way to write characters so they can be recognized more easily. We could use a similar approach for CPL by creating a simple set of rules, which, when applied, produce slight variations to the English language to make it easier to understand.

9. Acknowledgements

We would like to thank the few people who encouraged us in pursuing this outrageous line of enquiry including John Manley and Steve Wright. Roger Tucker played a major role, giving us advice and assistance in the experimental results. Michael Mc Ternan worked with us producing the CPL phone demo. We would also like to thank the members of the Voice Web team Marianne Hickey, Paul Brittan, and Lawrence Wilcock who created a supportive community where ideas could be born.

10. References

[1] Polifroni J. and Seneff S., "GALAXY-II as an Architecture for Spoken Dialogue Evaluation" Proc. 2nd Int. Conf. on Language Resources and Evaluation (LREC), Athens, Greece, May 31-June 2, 2000.

[2] Zue V. et al., "JUPITER: A Telephone-Based Conversational Interface for Weather Information," IEEE Trans. on Speech and Audio Proc., V.8, N.1, Jan 2000.

[3] Taylor P.A., Black A. and Caley R., "The Architecture of the Festival Speech Synthesis System", Third ESCA Workshop in Speech Synthesis, 1998, pp. 147-151.

[4] Lessons from the Development of a Conversational Interface. M. Hickey, P. Brittan, TR, HP Labs Bristol.

[5] The Artificial Language Construction Kit Web Chapter **http://zompist.com/kit.html**

[6] Science Fiction and Society: Artificial Languages By Christopher B. Jones

[7] http://www.i5ive.com/linkcategory.cfm/6513/10597

[8] Ardalambion: Of the Tongues of Arda, the invented world of J.R.R. Tolkien **http://www.uib.no/people/hnohf/**

[9] Esperanto and Science Fiction: Jules Verne, article October 29,1999 **http://www.i5ive.com/article.cfm/1146/27066**

[10] Dilingo 2000 – an artificial language of no particular use to anyone. **http://www.dilingo.com**

[11] VoiceWeb, Nuance overview. http://www.nuance.com/partners/voiceweb.html

[12] Where do Languages come from? Merritt Ruhlen. **http://www.exploratorium.edu/exploring/language/index.html**

[13] Constructed grammar FAQ **http://personalweb.sierra.net/~spynx/FAQ/index.html**

[14] Constructed Human Languages **http://www.quetzal.com/conlang.html**

[15] Genetic Programming, Bradford Books, Dec. 1992, John P. Koza

[16] Genetic Algorithms, Springer-Verlag, 1992, Zbigniew Michalewicz

[17] Scribble Matching, Hull, Richard; Reynolds, Dave; Gupta, Dipankar , HP labs external technical report HPL-94-61 July 14, 1994

[18] Arpabet http://www.billnet.org/phon/arpabet.php

[19] "A high level approach to confidence estimation in speech recognition", Stephen Cox, Srinandan Dasmahapatra, University of East Anglia

[20] Microsoft Speech API http://www.microsoft.com/speech/.