[Note:

This is a section of a larger document.

It is the as-built documentation I wrote following the instalation and customization of a major telecommunications company's voice recognition and response system.

The system is called Speech Navigator, a brand name.]

Table of Contents

1 SPEECH NAVIGATOR

1.1 DOCUMENTS REFERENCED

Speech Navigator Architectural design Production v0 7_.doc
Speech Navigator Detailed IVR Design FINAL (v1.5).doc
Speech Navigator Detailed Network Design Version 1.8.doc
Speech Navigator Detailed Softphone Design
TuVox <Company 1> Solution Architecture v1.08
Speech Navigator Technical Design ADDENDUM v 0 9 FINAL.doc

1.2 SPEECH NAVIGATOR: OVERVIEW

1.2.1 SYSTEM OVERVIEW

Speech Navigator is a <Company 1> suite for processing customer calls utilising speech recognition. The system comprises:

- a voice recognition engine,
- the TuVox suite of speech recognition management applications,
- the Genesys Voice Platform, and
- the existing IVR system.

A call is passed from the IVR system to the TuVox suite. TuVox runs on the Genesys Voice Platform. The caller talks to the machine; the machine collects the responses, uses OCS to interpret the speech, and talks back. After the machine-caller conversation has finished, the call returns to the existing IVR system.

As described in the 'Detailed Network Design' document, Speech Navigator has been divided into two halves.

core: the systems that provide the speech-recognition abilities. These components may be in less secure, non-<Company 1>, sites.

customer: systems that deliver calls to the Speech Navigator environment. These are components that are, and must remain, within the <Company 1> environment

Note that the terminology used in this document is Speech Navigator-centric, as opposed to <Company 1>-centric: the speech recognition applications are "core", the <Company 1> applications are the "customer".

core	customer
• systems that provide management or reporting • SIP / VoIP management • Speech processing	• Genesys • IVR / RS/6000 • TuVox
Possibly not at <Company 1>, e.g. at <Company 2>. May, in the future, be run by unrelated companies	Must be on the <Company 1> site, for security

Currently both the 'core' and the 'customer' are within the walls of <Company 1> though, from a design viewpoint, there is no reason for this to continue.

1.2.2 Hardware overview

As a brief overview, Speech Navigator involves the following machines:

- two infront-of-switch IBM RS/6000 boxes dedicated to providing the initial IVR routines for calls that have been selected for Speech Navigator treatment.
  [IE1861, IE1862]
  [Testing: HP1632]
- two SIP session managers. These manage and supply the details required for SIP initiated connections to Speech Navigator.
  [HP1777, HP1788.]
  [Backup: HP1778, HP1789]
  [Testing: HP1628]
- two IP Communications Servers (IPCS). These are the entry points to the Speech Navigator system, and the end points for the UDP/IP VoIP connection established by SIP.
  [HP1781, HP1791,
  [Back-up: HP1784, HP1794]
  [Testing: HP1629, HP1644]
- two Genesys Voice Web Managers.
  [HP1775, HP1776]
  [Backup: HP1787]
  [Testing: HP1626, HP1627]
- the TuVox Speech Application Server. This run the applications which implement and manage the Speech Navigator suite
  [HP1785]
  [Backup: HP1796]
  [Testing: HP1623, HP1645]
- eight ScanSoft OCS Servers. These provide the voice recognition capabilities of the Speech Navigator system.
  [HP1779, HP1780, HP1782, HP1783, HP1789, HP1790, HP1792, HP1793]
  [Testing: HP1630, HP1631, HP1622, HP1643]
- the TuVox Report Application Server.
  [HP1786]
  [Testing: HP1624]

These machines are split between <City 1> (Mayoral Drive Floor 5) and <City 2> (Caro Street Floor 5).

1.2.3 Software Overview

1.2.3.1 State Tables

State Table	Description
IStart.stb	all calls enter this state table, most being directed to ABCExpress.
ABCExpress.stb	<Company 1> Service Express. Sub-components of TSE.stb are used for many Speech Navigator tasks.
TSE_MainMenu.stb	In case of error the call may be re-directed here.
IVR2Speech.stb	the IVR interface to the Speech Navigator system. Uses IBMTromboneCall.stb to actually connect to Speech Navigator.
IBMTromboneCall.stb	handles a SIP trombone connection to an external system
TSESpeech.stb	The interface for transfer to a large number of other state tables.
other state tables	A caller may be re-directed to a specific, non-TSE, state table for particular tasks.

1.2.3.2 Speech Navigator.

Component	Description
Ant 1.6.2	
Apache 2.0.49	Web-Server
ASR Log Manager	
Bandwidth Manager	
EventC	
Genesys Queue Adapter	
Internet Information Server	
Web server,	
IIS Snap-in,	
ASP.NET	
IPCS Package v7.0 NE	A VoIP interface to Genesys.
JavaSDK 1.4.2.08	Java routines and libraries
Microsoft SQL Server 2000 Client Tools	Database server
MRCP TTS Client	
MSXML 4.0 SP2 Parser and SDK	
MySQL 4.1.11	Database
Outbound Resource Locator	
Policy Manager	
RealSpeak 4.0.x (Replaces Speechify)	Pronounces words from text.
Resource Manager	
Speechworks Media Server (SWMS)	
Speechworks OSR 3.0.4	Implements the voice recognition systems
SQL Server 2000 Client Software	Database client
Sun/iPlanet Directory Server 5.1 SP2	
Tivoli Data Protection client for MS	

Component	Description
SQL	
Tivoli TSM agent	
CA Unicenter Windows Agent	
Microsoft Operations Manager (MOM) agent	
Tomcat 5.0.28	Web-server
Voice Web Manager (VWM) Package	
Voice Web Provisioning System (VWPS)	Allocates VoIP resources on the IPCS
VWCM SIP Call Manager Software	
Windows FTP Server	an ftp server for updating Speech Navigator components.

1.2.3.3 Genesys

Component	Description
AIM: Advanced Interface Module	
GVP: Genesys Voice Platform	the Genesys IVR platform. Supports Speech Navigator.
IPCS: IP Communications Server	the VoIP interface to Genesys.
ISCC	Transfers connections between TServers
IServer	a TServer specialised for IVR operation, used by the GVP
Lightweight Directory Access Protocol	Supplies device addresses
SIP Resource Manager	two programs that manage the IPCS
TServer	Supplies the Genesys interface to non-Genesys devices
URS: Universal Routing Server	implements the Genesys call routing rules
Voice Web Call Manager Database	a database for Speech Navigator systems
Voice Web Provisioning System (VWPS)	a Speech Navigator configuration system.
VWM: Voice Web Manager	the Speech Navigator management application

1.2.4 NETWORK OVERVIEW

Like the IVR system, Speech Navigator is distributed between the <City 2> and <City 1> <Company 1> sites. These two sites are connected by <Company 1>'s Internal Data Network, and through an ATM link which encapsulates the Data (VLANFW-Data) and Management (VLANFW-MGMT) VLANS. Application specific shared VLANs run through the two sites.

The TuVox machines, existing IVR boxes, the SIP Session Mangers are on VLAN 15. Backup connections run through VLAN16. Between sites the VLAN15 production network is trunked (802.1q) over a fibre optic connection.

The <Company 2> Management Network, and the <City 1> and <City 2> Core Networks, with their internal multiply redundant internet work connections are managed by OSPF (a routing protocol). also connect onto VLAN15. These are obviously for CTI engineer use, not production, but are in addition 802.1q trunked over fibre optic connections.

All the Genesys boxes, the IPCS Servers, Voice Web Managers, and the eight ScanSoft OCS servers are connected at the two sites by the three Speech Navigator Production, Network, and Console networks, and through switches to each site's VLAN16. Note though that these each of these three networks is differently sub-netted between <City 2> and <City 1>.

1.3 SPEECH NAVIGATOR: CALL FLOW

1.3.1 THE DECISION TO SEND A CALL TO SPEECH NAVIGATOR

At the highest level, the decision to use speech is made by one or more of:

- Has the caller been randomly selected to receive a Speech Navigator treatment? As the Speech Navigator program is tested and implemented an increasing proportion of eligible calls will receive speech.
- Is there a speech treatment for this DNIS (and CLI) ?
- If the caller has dialled <Company 1> Service Express they will not receive speech, unless they request an operator, in which case they may receive speech.

Based on the dialled number, say *120, calls are directed to the IVR (the "WebSphere Voice Response system" WVR) by call plans in the <Company 1> Intelligent Network. A management device on the PSTN, ICM, then makes a decision as to whether to re-redirect the call to Speech Navigator. This is a random reallocation, e.g. prob=0.30. Upon success, the call is re-renumbered to an alternative DNIS number, from which the call is Speech Navigator. It is this new DNIS that signals to the existing IVR system that the call *could* be routed to the GVP for speech treatment.

1.3.1.1 Speech fields in the IVRDIP database table

While re-numbering of the call on the PSTN selects the call as targeted for speech, the final determination as to whether speech will be delivered is made by a value in the SPEECH_PROFILE field of the IVRDIP database table.

All IVR and speech routing data, indexed by DNIS and CLI, is held in he IVRDIP database. The indication that a speech treatment is available is that the SPEECH_PROFILE field of the IVRDIP database is filled with a reference into the SPEECH_PROFILES table, which holds the required Speech Navigator data. For a particular call, if a valid reference into the SPEECH_PROFILES table is listed in the IVRDIP table for that call's CLI and DNIS, then a speech treatment, based on the indexed record in the SPEEC_PROFILES table, will be applied to the call.

1.3.1.2 Speech Decision Steps

So, in summary, a call must satisfy each of the following steps, in turn, in order to be directed to Speech Navigator. Some of the stages are discussed in more detail in the following sections.

1. the call's DNIS has a Speech Navigator treatment available
2. the call's CLI qualifies for speech treatment.
3. the call is made during a time of day when Speech Navigator is active
4. the call is then randomly selected to be directed to speech recognition. The probabilities for assignment vary among DNISs, e.g. 128 calls may be 80%, while 123 calls are 10%.
5. The SPEECH_PROFILE field in the IVRDIP table is not null for this call's DNIS and CLI combination, and the value in the SPEECH_PROFILE field points to a valid speech treatment in the SPEECH_PROFILES table.
6. The customer does not press the key '9' before the call is transferred into the Speech Navigator environment.
7. In IVR2Speech state table a further check is made that it is appropriate for this call to receive speech treatment

8. The Speech Navigator system is active. This test is made in the IVR2Speech.stb state table.
9. Also in the IVR2Speech state table, the trombone transfer to Speech Navigator must be successful: there are available SIP managers, the call is handled properly on the IPCS, and Speech Navigator responds.

If any of these steps fail, the default IVR is used.

An alternative path to the use Speech Navigator is theoretically – though not practically – available. If, during a <Company 1> Service Express (TSE) IVR the caller pushes '0' , they should be (but aren't) directed to Speech Navigator.

1.3.2 Through the existing IVR system

1.3.2.1 IStart

All calls enter the IVR (and Speech Navigator) systems through the "IStart.stb" IVR State Table. Amongst its many other tasks, IStart notifies Genesys of a call by executing the NotifyCallStart( ) function of the d2is custom server, which wraps the Genesys-to-WebSphere interface driver. The NotifyCallStart( ) function also delivers the call's DNIS and CLI to Genesys. An just noted above, the DNIS and CLI are used by Genesys to finally decide the call is to be routed to Speech Navigator.

Now that Genesys has been informed of a new call, and completed some initial processing, the d2is function GetRequest( ) is used to gather a large amount of routing information for this call. The GenTools custom server interprets the data returned from Genesys, writing to the variable ivr_dnis the basic, non-Genesys, routing information needed for Speech Navigator. ivr_dnis is copied to the SV81 variable. It is the information in the SV81 that directs the call's path both to Speech Navigator, and for many calls, after the return from Speech Navigator.

Note that the large amount of information returned in the single call to d2is.GetRequest( ), which includes a long string of routing information, has saved a commensurately large number of d2is interactions with Genesys. Usage of the d2is custom server is the central constraint in the IVR system.

SV81 now holds a long string of carat ('^') delimited routing information. For example, SV81 may contain:

```
|IVR|DNIS|
17^PERCENT_SPEECH=50^CIRN_REQUIRED=YES^DEFAULT_IVR_NAME=ResMobile^DATA_REQUIR
ED= ACCT_BALANCE~OLC|LABEL|IVR2Speech|
```

IStart now calls the GenIVR state table, passing to it the label 'IVR2SPEECH' , the SV81, and some other useful information.

1.3.2.2 GenIVR

NB: no call from GenIVR, from TSE

GenIVR a switch-board function, responsible for directing execution-flow amongst the top-level destination IVRs; all IVR calls come here. GenIVR optionally plays a greeting (not for speech) and transfers the call to the appropriate IVR, in this case IVR2SPEECH.

1.3.2.3 IVR2Speech

Simply, IVR2Speech prepares both the routing and customer information for Speech Navigator, before transferring the call.

In more detail, IVR2Speech:

1. tests for the customer to press '9', in which case they are directed back to the IVR system;
2. again tests that this call should receive speech;
3. find and test a SIP server, from which the SIP Manger information will shortly be obtained. SIP is the protocol that sets up VoIP connections.

TNZ_TUX_2 is the customer server for interfacing to ICMS, <Company 1>'s customer information system.

4. TNZ_TUX_2 is used to both:
   a. get list of SIP managers, and other SIP related network information
   b. get a long string from ICMS of all of the caller's information
5. From the list of SIP Managers just obtained, find an appropriate one that is operational;
6. Write to Genesys the string of customer data obtained from ICMS. This will be collected back from Genesys by Speech Navigator;
7. Finally, if an appropriate SIP Manger was found, and thus a SIP address is available, run the IBMTromboneTransfer state table to transfer the call to Speech Navigator, and control to Genesys.

IVR2Speech receives control back from Speech Navigator when the later drops the call.

All IVR log records for any Speech call, regardless of subject or DNIS, are currently written to IVR2SPEECH log. This may change as speech handling becomes normalised.

1.3.3 ENTERING SPEECH NAVIGATOR

To connect to Speech Navigator the ISDN call is directed across the RS/6000's internal H.100 bus to the machine's DTeA card. This specialised VoIP card does both the A-law speech compression, and the ISDN to VoIP conversion. This produces G.711 frames of 80 bytes length, each taking 10ms of voice.

The network connection is over <Company 1>'s internal Gigabit fibre optic internal trunks, from the IVR machines to the Genesys IPCS VoiceXML Speech Navigator interfaces.

1.3.3.1 call passage IVR to Speech Navigator

< Diagram removed >

After the IVR2Speech state table trombones the call to the IPCS, from which a VoIP call to the GVP is set up (using SIP), there is now a direct connection from the caller, via the RS/6000 State Table system, via the GVP, to the TuVox Speech Navigator system. From here the TuVox VoiceXML scripts control the call.

The SIP session connection from the IVR is responded to by the IP Communication Server. The response from the IPCS is a trigger for the appropriate GVP application for the call. The appropriate application is Speech Navigator.

Once SIP has established the call, the VoIP conversation itself runs over UDP/IP.

< Diagram removed >

IVR2Speech, having collected the SIP information to set up a call, uses the IBMTromboneCall.stb to set up a SIP connection to the IPCS. From the Speech Navigator point of view this a “SIP invite” on the IPCS server.

1.3.3.2 Proxy server

The connection from the IVR is actually made through a SIP Proxy Server that implements load-balancing and subsequent routing of SIP connections. Once the proxy has checked that the connection is valid, the SIP Resource manager launches a server on the port to which the SIP connection is directed.

The SIP connection to the IPCS can now be made.

1.3.3.3 Session Initiation Protocol (SIP) call initialisation

SIP has become the de facto standard for setting up VoIP calls. As a very brief description, SIP sets up a call as per the following action list, where 'A' is the IVR system, and 'B' is the IPCS, connected to via a SIP proxy.

<u>**A**</u>	<u>**B**</u>
invite →	
	← Ringing (code 180)
	← Answers (code 200 = OK)
ACK →	

The VoIP call is now established, over a UDP / IP connection. As the connection is UDP, dropped packets are not re-sent or tracked, so that blank spaces will appear in the conversation. This is ameliorated by using an application-level RTP (Real-Time Transport) protocol, which makes short pieces of guess-noise to fill in the gaps.

VoIP conversation
VoIP conversation
VoIP conversation
:
:
VoIP conversation
VoIP conversation
VoIP conversation

Speech Navigator hangs up.

	← drops call (BYE)
code 200 = OK →	

1.3.3.4 IPCS

When the IPCS detects the SIP invite it uses the SIP session manager, which uses the Resource Manager, to find out which of it VoIP ports to use. Leveraging the consistency of bad design, the Resource Manager then uses the VoiceWebCall database to track the progression of the VoIP call on the IPCS. The IPCS itself is managed by the Genesys Voice Web Manager (VWM).

The IPCS makes a HTTP / IP connection to the TuVox application server from which it receives a stream of call-customised VoiceXML scripts. It is these scripts that present "Speech Navigator" to the caller.

1.3.4 PROCESSING A CALL IN SPEECH NAVIGATOR.

1.3.4.1 overview

Now that a VoIP connection has been made to the IPCS:

- VoiceXML scripts are sent from the TuVox server, to the IPCS, where they are executed.
- These scripts play voice segments to the caller, hold collected data, and do some call management
- The GVP sends the caller's "utterances" to the OSR speech recognition engine, which, with other Speech Navigator components, returns a "destination", a next-step the Speech Navigator system should take.
- VoiceXML scripts are generated, and played, to implement this next step.

This cycle continues until a "destination" is reached that requires handling outside Speech Navigator. At this point the information so far collected is passed back to the IVR system for final handling.

< Diagram removed >

1.3.5 THE VOICEXML INTERFACE: SPEECH NAVIGATOR.

Now that a connection to the IPCS has been made, VoiceXML scripts are played to the caller, and the caller's speech is collected, analysed, and used to specify further VoiceXML scripts. To the caller, these three components provide an interface. This interface is Speech Navigator.

Initially a single, customised "root" VoiceXML script runs, and remains active for the length of the call. This "VRUIntro" page holds menus, "forms", links, and exception handlers for the use of "leaf" VoiceXML scripts.

Next the TuVox server sends a "leaf" VoiceXML script to the IPCS. This script, which runs within the root VoiceXML script's environment, collects the long carat delimited string of routing, customer, and call information sent from the RS/6000 StateTable (IVR) environment as the "IVRDATA" Genesys Key-Value Pair. Once checks are passed, the input data is copied to the root VRUIntro Page's 'global' or 'public' variables, here called "ConceptTopics" ('CT's). This data is also written to the TuVox application server.

The initial set-up of the Speech Navigator environment takes about two seconds, the time for four phone rings.

1.3.6 PROCESSING CALL

A further VoiceXML script is sent to activate the first Speech Navigator voice statement:

> "*Welcome to <Company 1> Business. To help me assist you with your call, please tell me what you are calling about* ."

The caller is now relied upon to, in some way, reply to this prompt.

The GVP collects the caller's response, an " Utterance", and sends it to the ScanSoft OSR to be interpreted. After the caller's speech has been interpreted, the "Statistical Modelling Language" and the "Destination List" are used to determine the next action, a transfer or another VoiceXML script. This process is briefly described in the section "Caller Replies, Reply Interpreted" on page 16.

From the point of view of the caller, the main structures of Speech Navigator are the particular hierarchies and patterns of VoiceXML scripts that tend to be played in response to certain types of utterances. These main structures are grouped into six categories:

- Disambiguation Structure,
- Complex Structure,
- Customer Specific InfoBlast,
- quick-words short-cut,
- Vague Response,
- Guide Me Menu Structure

It is possible for Speech Navigator to direct a call to the correct destination after a single statement from the caller. However usually one or more hierarchy structures have to be used.

Note that, at each stage the process "Caller Replies, Reply Interpreted" is utilised.

1.3.6.1 Disambiguation structure

A hierarchy constructed so as to lead the caller to the one appropriate topic for the call. Within each step of a Disambiguation Structure the "Caller Replies, Reply Interpreted" stage is executed.

As an example, after a caller saying "Its about one of things you can get added to your phone", Speech Navigator determines the call is about "smart Phone" services. It then may determine the caller needs help with one of the services, and finally determines that the problem is with call_minder.

This disambiguation Structure may be as below:

stage	Speech Navigator "destination"(example)
start	smart_phone_services_d
disambiguation #1	smart_phone_services_help_d
disambiguation #2	(one of:) • call_minder • call_waiting • caller_display • call_restriction • call_diversion • smart_phone_services_other

Throughout the Disambiguation Structure a running dynamic assignment of sequence number is maintained. At the end of a Disambiguation Structure, Speech Navigator has gathered sufficient information to direct the call to its destination.

1.3.6.2 Complex Structures

Sets of specific voice prompts, structures, and information, pre-panned for more complex topics. While a Complex Structure is like a specialised Disambiguation Structure, it also delivers information to the caller, like a collection of small Customer Specific / InfoBlasts.

Within each step of a Complex Structure the "Caller Replies, Reply Interpreted" stage is executed.

Speech Navigator has 25 complex structures, for example taking a person through setting up call minder, or connecting to the internet. If necessary, from here a caller can be transferred to an appropriate technical or customer support.

1.3.6.3 Customer specific / InfoBlast

These are custom-generated blocks of information recited by Speech Navigator. There are two of these: Account information summary, and collections (debt). Recitations such as these are referred to as "InfoBlasts" in TuVox terminology.

1.3.6.4 quick-words short-cuts

If the caller's utterances are identified as one of any "quick words" the caller is immediately re-routed to the appropriate destination. Some possible quick words are:

> 'Off Peak Hours', 'Payment Options', 'Automatic Payment Form', 'Credit Card Payment', 'Direct Debit Form', 'Phonebook', 'World Clock'.

These are implementations of the OSR "Voice Enrolment" functionality.

1.3.6.5 Vague response

Vague Response (processes) are variations on: "can you be more precise ? here's a hint.". Within each step of a Vague Response structure the "Caller Replies, Reply Interpreted" stage is executed.

As an example,

caller:	"I suppose I might want to talk to someone in credit",
Speech Navigator:	"Do you have questions about an overdue account"
caller:	"Yes, I do"

Having resolved the Vague Response, Speech Navigator may return to a Disambiguation Structure to fully characterise the caller's needs. If a second Vague Response is needed, the caller is directed to a GuideMe menu structure.

1.3.6.6 Guide Me Menu Structure

Should Speech Navigator "realise" that the caller is having trouble with the Speech Navigator system itself, then control will be diverted to a "GuideMe" Menu Structure, informing the caller about Speech Navigator and how to use it.

The Guide Me Menu Structure also utilises the voice recognition described in "Caller Replies, Reply Interpreted".

1.3.7 CALLER REPLIES, REPLY INTERPRETED

Replies are interpreted by ScanSoft OCS (version 3.0).

1. A G711 audio stream is sent (from the caller via) GVP to the OCS Speech Detector. This stream is the caller's "Utterance".
2. The Speech Detector collects and passes short, isolated, blocks of pure speech to the OSR Recogniser.
3. The OSR recogniser matches the blocks of speech against libraries of "grammars" to determine their content.
4. Once words have been recognised, they are compared against a collection of important key-words called "slots". As the relationship of slots to meaning may not be straight-forward, a process called "Robust Parsing" is used to properly extract "slots" and meaning.

Key-words, "slots", have to be extracted from sentences, which may be confused, incomplete, uniquely ordered, and/or incorrect. This process is called "Robust Parsing", and the open-ended questions that necessitate Robust-parsing – like "Welcome to <Company 1>, what can I help you with today ?" – are called "mixed initiative dialogs".

5. the OSR Statistical Language Model (SLM) matches the gathered "slots" and related information to an 'interpreted response', an internal measure of what was said.
6. The 'interpreted response' is matched to a "Speech Navigator destination" which indicates the next Speech Navigator action. The 'interpreted response' to next 'Speech Navigator destination' matching was developed during the Semantic Mapping stage of Speech Navigator's development. The 'DestinationList' holds every Speech Navigator destination.

The next Speech Navigator destination has now been determined. This next destination is made available to the TuVox engine which now constructs the appropriate VoiceXML script.

1.3.8 DATA RETURNED FROM SPEECH NAVIGATOR.

At the end of the call's passage through a custom-generated collection of VoiceXML scripts, Speech Navigator has determined a 4-digit exit-code, called a "Speech-ID". This is converted by Genesys – actually by a URS routing strategy – into a string of data to be returned to the IVR system. For each Speech-ID, the URS will return a different string of data. This string of data is put into the value of a single Key-Value Pair: SPEECHDATA.

The Speech_to_IVR database table holds the data to be written to the SPEECHDATA value. The Speech-ID value returned from Speech Navigator is looked-up against the "ID" column key, and all the other fields of the located Speech_to_IVR record are then concatenated into a carat (‘^’) delimited string of data, resulting in a SPEECHDATA Key-Value pair resembling:

SPEECHDATA | HANDLING=STATE_TABLE^
SEQUENCE_NUMBER=5505^STATE_TABLE=TSE_Main^……

The table lists the return-codes

ask for CIRN	**ask if CIRN**	**ask if CIRN No PIN**	**CIRN known Ask PIN**	**CIRN known No PIN**	**No CIRN No PIN**	**Return Point**
1000	1001		1002			billing_accountsummary
1100	1101		1102			collections_extension
		1200		1201		anytime_info
	1300		1302		1301	anytime_signup
	1400		1402		1401	billing_account_number
	1500		1502		1501	billing_billcopy
	1600		1602		1601	billing_billdates
	1700		1702		1701	billing_last_payment_details
	1800		1802		1801	billing_unbilled_calls
	1900		1902		1901	collections_record_receipt_number
					2000	country_code
					2100	faults_busyline
					2200	flybuys
	2300			2301		mobile_cellular_secretary_brochure
			2400	2401		payments_ap_form
	2500		2502		2501	payments_creditcard
	2600			2601		payments_directdebit_form
		2700		2701		smarties_brochure
					2800	toll_national_specials
					2900	time_of_day
					3000	world_clock
					9999	test_speechnavigator

ID numbers in the range 5000 and up, but excluding 9999, are reserved for agent transfers and will be defined by TuVox.

< Diagram removed >

1.3.9 RETURN TO IVR2SPEECH

In the initial design it was considered having Genesys handle all the post Speech Navigator routing. At this stage it would be too much work to implement this change.

After the call exits the Speech Navigator system:

- the trombone connection being run by the IBM_Trombone_Custom_Server is terminated
- the IBMTromboneCall.stb state table, a simple wrapper, completes
- control returns to the IVR2Speech state table
- the IVR2Speech state table continues

If the traversal of Speech Navigator was successful the first thing IVR2Speech does is retrieve the SPEECHDATA Key-Value pair that, as just described, contain the call information derived from Speech Navigator. The TNZ_TUX_2 function 'TUX_Speech_ParseGvpData' is used to parse the string into its component values.

1.3.10 POST SPEECH NAVIGATOR OPTIONS

There are, let's say, ten targets for a post speech call:

1. a part of TSE (ABCExpress) for a particular task.
2. a return to the location in TSE from which the caller was diverted to Speech Navigator.
3. a state table that does one specific task, and <u>does</u> require variables to be set. e.g. CALNextBillChrg that tells the caller the state of their next bill
4. a state table that does one specific task, and <u>does not</u> require variables to be set. e.g. the World Clock
5. Table driven state table.
6. TSE_MainMenu, if an error or confusion occurred.
7. CSR
8. Transfer to a non-<Company 1> system. Not currently implemented.
9. call-termination: the caller and <Company 1> have both given and received all the information required
10. call-termination: the caller has hung up the phone.

1.3.10.1 ABCExpress (TSE)

The State-table "ABCExpress" , and the value of a label within ABCExpress, are returned from Speech Navigator. Control passes from IVR2Speech to this location with-in ABCExpress.

Any update value of SV248 controls which part of TSE is run.

1.3.10.2 return to where the caller left TSE

If this is required, then IVR2Speech returns control to TSE, from where the caller came, and then forces an '8' key-press, returning the caller to the previous menu within TSE.

1.3.10.3 state table via TSESpeech.stb (variable set)

If a caller needs to access a state table, or part of a state table, preparation of variables may be required. If this is the case, control first passes from IVR2Speech to TSESpeech in order to prepare the variables required for the transfer. Each part of TSESpeech prepares for transfer to a different state table. Control is then passed to the appropriate state table.

The state tables to which control may pass are listed on page 29.

1.3.10.4 State table directly (no variables set)

If a subsequent state table can be called without parameters being set, then it is called directly by IVR2Speech.

For example a caller may have dialled 126, gone to Speech Navigator, then said they wanted the "World Clock". Speech Navigator would set the Speech Navigator exit code (the "Speech-ID") to 3000. Next the URS would convert this

1.3.10.5 Table driven IVR

Control may pass to a table driven state table after Speech Navigator. This is handled as a parameter-less (no variables set) transfer from IVR2Speech.

1.3.10.6 TSE_MainMenu – if an error occurred

If an error or confusion occurred control is passed to TSE_MainMenu, which uses the SV248 value returned from Speech Navigator to direct the call to the correct next state table.

1.3.10.7 Transfer to a CSR

If the caller has to be transferred, logging is done and the exit code is set to 9. The remainder of calls need another IVR to be run.

1.3.10.8 External transfer

Blair Thompson, one of the two developers of the IVR – Speech Navigator system, has, on the subject of transfers to a non-<Company 1> external system after Speech Navigator, said, that transfers to any external number is of such great ease – just dial the number – that there is no need for any support to be coded. There are no current examples, but they with great ease appear.

1.3.10.9 Caller Termination: interaction completed.

Standard logging is done and the call is terminated.

1.3.10.10 Caller hang-up

If the caller has hung up the phone, some logging is done and the state table exits with code 4 (hung-up).

1.4 SPEECH NAVIGATOR: COMPONENTS

1.4.1 NETWORKS

VLAN	Network	Subnet Mask	Comments
	146.171.254.0	255.255.255.192	TAG External DMZ
	64.221.252.192	255.255.255.192	TuVox External Address
4	146.171.22.0	255.255.255.128	External
15	192.168.67.160	255.255.255.224	IVR_DEV
16	192.168.67.192	255.255.255.224	IVR_BAK
20	192.213.139.0	255.255.255.0	G-IDN-Networks
21	192.168.52.64	255.255.255.224	Model Core Components
22	192.168.52.96	255.255.255.224	Model Customer Components
23	192.168.52.128	255.255.255.224	Model Core/Customer Components
23	192.168.52.128	255.255.255.224	staging admin network
30	10.48.72.0	255.255.255.0	ATC L3 Data Network (via DHCP)

The network connection to Speech Navigator, from the existing <Company 1> IVR system, is implemented in IP. Voice traffic (VOIP) is carried using Real-Time Protocol via UDP/IP and all other connections (data and signalling) use TCP/IP.

In this architecture there are VOIP and Speech resources in each WVR site (<City 1> and <City 2>). This has been done to avoid any potential issues arising from carrying voice over a wide area network.

The new staging environment located between MDR and ATC is connected via Gigabit Ethernet.

1.4.1.1 (Session Initiation Protocol) SIP System

SIP is the protocol used to establish VoIP calls, analogous to the D-Channel signalling in the initialisation of ISDN calls.

When conferencing a call to Speech Navigator , the StateTable IVR system:

- contacts a SIP manager
- gets a SIP address from the SIP_Manager.
- the IBM supplied IBMTromboneOut custom server to make a connection to the SIP address

The SIP address is one of the four Genesys IPCS servers, the entry point to the TuVox part of Speech Navigator. Currently only local IPCS devices are used, though eventually each site's backup manager may provide a non-local SIP Manager if no local SIP Manager isn't available. in the initial stages concerns regarding latency over networks for VoIP limit this.

The SIP address is in the form of **route_point@GVP_switch** say “sip:ray@99.99.55.55:5060” , where

- sip the standard prefix | route point
- ray the address, like the name on an email address |
- 99.99.55.55 an IP address. | GVP Switch
- 5060 port

1.4.1.2 Network Cards and lines

Device	supplies	lines	
E1 trunk	30 ISND lines in.	30 ISDN lines in	
DTTA card DTeA card	4 x E1 trunks,	4 x 30 =	120 ISDN lines in 120 VoIP lines out
IVR box	2 x DTTA telephony cards, 1 x DTeA VoIP card	2 x 120 = 1 x 120 =	240 incoming lines 120 VoIP lines out
2 x IVR boxes	4 DTTA cards, 2 DTeA cards	2 x 240 = 2 x 120 =	480 incoming lines 240 VoIP lines out
IPCS server	licensing limit	120 VoIP calls in	
all IPCS server	licensing limit	200 VoIP calls in 160 max at any one site	
Max lines	**Speech Navigator max calls**	**200**	

1.4.1.3 Constraints on Speech Navigator call numbers

Constraint	Limitation
Licensing	max 200 concurrent VoIP calls max 160 concurrent VoIP calls per site
design	The Speech Navigator system has been designed with a maximum of 5 calls per second per IPCS server
hardware	There are three appropriate PCI slots in a 520 Series RS/6000 machine, for a maximum of 2 DTTA (ISDN) cards and one DTeA. <Company 1> are limited to 240 incoming, and 120 Speech Navigator calls per box.
hardware	(No constraint) IVR machine power . Once the call is established there is no CPU overhead, as the call comes in one card, over an internal bus (the H.100 bus) and out the other card.
ISDN lines	The DTTA cards used by <Company 1> are no longer produced, leading to an absolute limit on the number of ISDN lines supported.
design	The Speech Navigator system has been designed to handle a maximum of 8500 calls (peak hour), lasting an average of 65 seconds. 190 lines have been calculated to be adequate to handle this load.

from the “<Company 1> – Service Architecture” TuVox document, the calculation to determine 190 lines are required during the peak hour:

Constraint	Limitation
Call Handling Time	65 seconds (average)
peak traffic	8500 calls per hour
erlangs required	154
5.0% call blockage (peak hour)	158 lines required
1.0% call blockage (peak hour)	174 lines required
0.1% call blockage (peak hour)	188 lines required (190 lines)
Lines available	240 lines available
peak hour support under machine failure	80%
lines available under machine failure	160
supported future lines	1000 lines: possible future

Notes:

Beyond Speech Navigator’s capacity calls will return to the existing IVR system.

The TuVox architecture, a central repository servicing a number of application servers, lends itself to easy expansion.

1.4.1.4 bandwidth / capacity

All voice calls will be G711. The quality of these calls must be guaranteed. Allow 100kb per call. At 190 peak calls, the required bandwidth is 19000 kb, roughly 2Mb during the peak hour. The connection between the IVR and the IPCS is a 1000 Mb fibre optic, 802.1q trunked line.

Additionally each call will generate approximately:

- 3kb between Genesys Server 1 (GVP I-svr) and VWM1 MD AK (GQA).
- 10kb between IPCS MD AK and TUVOX APP MD AK
- 1kb between WVR (DTEA) and SIP SM MD AK

1.4.2 IPCS Servers

IPCS servers are

- entry points to the GVP, and thus the Speech Navigator system
- there are two at each site, together able to supporting 160 ports together, or up to 120 calls under failure conditions.
- are a "voice browser" which retrieves and runs the VoiceXML scripts dynamically generated by the TuVox application server.

These are obviously used to manage the call on the Speech Navigator platform.

The four IPCS servers are HP1781, HP 1784, HP 1791, and HP 1795. They each run:

- MRCP TTS Client,
- MRCP ASR Client,
- IPCS Package v7.0 NE

1.4.3 Software components

From the document Speech Navigator Support- incl Machines-v0.6 2006-09-08.doc

1.4.3.1 Production

Server	Software	Machine
Genesys IPCS Server	MRCP TTS Client, MRCP ASR Client, IPCS Package v7.0 NE	HP1781 HP 1784 HP 1791 HP 1795
Genesys SIP Session Manager	VWCM SIP Call Manager Software, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP1777 HP 1778 HP 1788 HP 1792
Genesys Voice Web Manager	Voice Web Manager (VWM) Package V7.0 NE Components, Policy Manager, Bandwidth Manager, EventC, ASR Log Manager, Resource Manager, Genesys Queue Adapter, Outbound Resource Locator, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP1775 HP1776 HP 1787
Genesys Voice Web Provisioning	Internet Information Server (Web server, IIS Snap-in, ASP.NET), MSXML 4.0 SP2 Parser and SDK, Sun/iPlanet Directory Server 5.1 SP2 , SQL Server 2000 Client Software with SP 3a, Voice Web Provisioning System (VWPS) - Package V7.0 NE, Microsoft SQL Server 2000 Client Tools with SP3a, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent , Tivoli Data Protection client for MS SQL	HP1774
ScanSoft OSR Server	Speechworks OSR 3.0.4 , RealSpeak 4.0.x (Replaces Speechify), Speechworks Media Server (SWMS) 3.1.4, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP 1779 HP1780 HP 1782 HP 1783 HP 1789 HP 1790 HP 1793 HP 1794
TuVox Report Application Server	Tomcat 5.0.28, Apache 2.0.49, MySQL 4.1.11, Ant 1.6.2, JavaSDK 1.4.2.08 , Windows FTP Server, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP 1786
TuVox Speech Application Server	Tomcat 5.0.28, Apache 2.0.49, MySQL 4.1.11, Ant 1.6.2, JavaSDK 1.4.2.08, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP 1785 HP 1796

1.4.3.2 Testing

Server	Software	Machine
Genesys IPCS Server	MRCP TTS Client, MRCP ASR Client, IPCS Package v7.0 NE	HP 1629 HP 1644
Genesys SIP Session Manager	VWCM SIP Call Manager Software, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP 1628
Genesys Voice Web Manager	Voice Web Manager (VWM) Package V7.0 NE Components, Policy Manager, Bandwidth Manager, EventC, ASR Log Manager, Resource Manager, Genesys Queue Adapter, Outbound Resource Locator, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP 1626 HP 1627
Genesys Voice Web Provisioning	Internet Information Server (Web server, IIS Snap-in, ASP.NET), MSXML 4.0 SP2 Parser and SDK, Sun/iPlanet Directory Server 5.1 SP2 , SQL Server 2000 Client Software with SP 3a, Voice Web Provisioning System (VWPS) - Package V7.0 NE, Microsoft SQL Server 2000 Client Tools with SP3a, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent , Tivoli Data Protection client for MS SQL	HP 1625
ScanSoft OSR Server -- and -- Genesys SIP Session Manager	Speechworks OSR 3.0.4 , RealSpeak 4.0.x (Replaces Speechify), Speechworks Media Server (SWMS) 3.1.4, VWCM SIP Call Manager Software, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP 1643
TuVox Report Application Server	Tomcat 5.0.28, Apache 2.0.49, MySQL 4.1.11, Ant 1.6.2, JavaSDK 1.4.2.08 , Windows FTP Server, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP 1624
TuVox Speech Application Server	Tomcat 5.0.28, Apache 2.0.49, MySQL 4.1.11, Ant 1.6.2, JavaSDK 1.4.2.08, Tivoli TSM agent, CA Unicenter Windows Agent, Microsoft Operations Manager (MOM) agent	HP 1623 HP 1645
Genesys	Genesys Framework v 7.0	HP 1620 HP 1621

1.4.4 State Tables

The state tables involved in Speech Navigator are:

State Table	Description
IStart.stb	all calls enter this state table, most being directed to ABCExpress.
ABCExpress.stb	<Company 1> Service Express. Sub-components of TSE.stb are used for some Speech Navigator tasks. In case of error the call may be re-directed here.
TSE_MainMenu.stb	In case of error the call may be re-directed here.
IVR2Speech.stb	the IVR interface to the Speech Navigator system. Uses IBMTromboneCall.stb to actually connect to Speech Navigator.
IBMTromboneCall.stb	handles a SIP trombone connection to an external system
TSESpeech.stb	The interface for transfer to a large number of other state tables.
other state tables	A caller may be re-directed to a specific, non-TSE, state table for particular tasks.

1.4.4.1 IStart State Table

All calls entering <Company 1>'s IVR system pass through either the IStart (or GenStart) state table; whether heading towards Speech Navigator, TSE, a customer service representative, or an ordinary IVR.

The tasks accomplished by the IVR2Speech state table are:

- Interacts with Genesys, sending call information and receiving routing information
- Interacts with back-end systems, notably ICMS via the TNZ_TUX custom server
- Transfers most calls to GenIVR.
- Many of the calls to GenIVR are then directed to ABCExpress, which in turn sends them to the actual state table implementing the requested functionality.
- Transfers other calls to
  - "Destination" – transfer to a CSR via ITransfer
  - "Transfer" – loops for another treatment, either another IVR, or to a CSR
  - "music" – will play music via GenMusic
  - "Ringback" – will play a ringing sound via GenRingBack
  - "Ran" – will play a voice, e.g. "We value your call …."
  - "Silence" – 1 second silence
  - "Routetype Default" – an error, send to a default ACD on a call centre PABX

1.4.4.2 ABCExpress State Table

ABCExpress is the main entry point for many of <Company 1>'s IVR services. Many different dialled numbers (DNISs) are directed here.

The tasks accomplished by the ABCExpress state table are:

- Play an initial, and any custom, greeting
- Transfer the caller to the appropriate sub-State Table, dependent on the value of SV248, which in turn depends on the number the caller dialled to get here.
-

The state tables run from ABCExpress are:

> AutoAttendant, BillingExpress3, BSS, BusAccntTolls, BusinessTolls, cdma_otaf_main2, CellSecretry, CellSecretry41, CIC, ColTestHarness, Contract, CorpPilot, CreditManageCol, CreditManagemnt, FaultsRedir, FreeMins, GO_FIVE, TS_AnytimeP, TS_AnytimePNeg, TS_NatPlanBuy, TS_NatPlans, TS_NatPlanSugg, TS_Residents, TSE_MainMenu, TWO_ON_1, WrapUpCall

Transfer to Speech Navigator. If a caller presses '0' while in TSE, they will, if possible, be transferred to Speech Navigator.

1.4.4.3 TSE_MainMenu State Table

TSE_MainMenu is a default entry point, called by ABCExpress, TSESpeech, and GenIVR in the absence of routing information. Properly directed calls are directed to ABCExpress.

The tasks accomplished by the TSE_MainMenu state table are:

- Collect the caller's key-press and transfer appropriately to
  - BillingExpress
  - Tolls_Express
  - ProductExpress
  - Help_Express
  - ParseShortCode
- On ' 0 ' transfer to Speech Navigator, or a CSR

Note that TSE_MainMenu is only used in fault, or problem, situations.

1.4.4.4 IVR2Speech StateTable

The tasks accomplished by the IVR2Speech state table are:

- test for an override of the Speech Navigator system, by listening for a key-press of '9'
- uses the TNZ_TUX_NZ customer server to gather customer and routing information. This is listed on page XX .
- From the routing information just gathered find an available SIP manager, via which the IVR system connects to a Genesys IPCS server, thus presenting the Speech Navigator interface
- Initialise a the connection to the Genesys Voice Platform (GVP), via which the callers "utterances" are sent to the speech recognition system.
- Use the IBMTromboneCall StateTable is used to trombone the call to Speech Navigator.

1.4.4.5 IVRTrombone.stb

IVRTromboneCall is used to trombone-transfer the call to the Speech Navigator system.

The tasks accomplished by the TSE_MainMenu state table are:

- Opens and connects to the Trombone Transfer custom server
- Uses the Trombone Transfer custom server to transfer the call.

1.4.4.6 TSESpeech State table

TSESpeech is run after a caller has used Speech Navigator, and now is returning to the IVR system. TSESpeech prepares parameters and variables for transfer to particular IVRs, each section of TSESpeech prepares for transfer to a different state table.

The tasks accomplished by the TSESpeech state table are:

For each section, i.e. for each transfer:

- Prepares the parameters and variables for the transfer
- Call the appropriate state table.

The state tables called by TSESpeech are:

State table	where (label)	task
CALBillCpy	the start	Recite the caller’s account number
CALBillCpy	the start	Have a copy of their bill sent
CALActDetails	the start	What are the dates on the current & next bill
BillExprMain	the start	A summary of the payments made.
Authenticate	the start	Required for any PIN authorisation
CALBillPmts	SpeechAPForm	Have an A/P form sent out
CALBillPmts	SpeechCreditPmt	Start the credit-card payment process
CALBillPmts	SpeechDDForm	Have a Direct Debit form sent out
CALNextBillChrg	the start	A summary of current charges on bill
PayExtension	the start	Extend the date till payment due
PayExtension	APartyEnquiry	Extend the date till payment due
PayExtension	OtherNoEnquiry	Extend the date till payment due
PrepareFBData	the start	To fly-buys, register or points
FlushLogBuffer	the start	internal use
CALActDetails	the start	Details of the last payment made
BillExprMain	the start	else run BillExprMain to get payment details
TSE_MainMenu	the start	else transfer to the start of TSE
Mailing	the start	Have a Mobile Secretary form sent out
RecDetailsCol	the start	To record a payment as being paid
Mailing	the start	Have a “smarties” form sent out

1.4.5 Speech Navigator data from IVR2Speech

After querying ICMS, the IVR2Speech state table has gathered all the information needed by Speech Navigator. This is SIP connection data, used to make a VoIP connection to the Speech Navigator system, a SIP switch ID, a SIP route point and a list of SIP managers.

Customer data:

Key	Input	Comment
CLI	CLI	start with'0', max 10 digits
CUST_KNOWN	Customer known	"FALSE', the other variables blank
ACCT_BALANCE	Account balance	e.g. 23.76
OVERDUE_AMT	Overdue amount	e.g. 23.76
SUBTYPE	Customer subtype	RS=residential, HB=home business.
SV248	DNIS indicator	A code indicating the type of call.
OLC	Own Line Challenge	code: 0,1,2 or 3: ask for a PIN ?
DUE_DATE	Due date	YYYYMMDD. When bill due
NEXT_STMT_DATE	Next statement date	YYYYMMDD. When next statement
ICMS_INTERACTIVE	ICMS interactive flag	True/False: Can ICMS be written to ?
CC_OPEN	Call Centre open	True/False: Is a Call Centre open
CIRN	Calling In Regard Num	What is the phone number this call is about

The customer data is concatenated into a carat ('^') delimited string of Key-Value Pairs

```
key1=value1^key2=value2^key3=value3^key4=value4
```

and passed to Genesys in the value of a single Key-Value Pair, with the key "IVRDATA", arranged as:

IVRDATA | CLI=091234567^ACCT_BALANCE=54.65^…..

Speech Navigator retrieves this KVP from Genesys and writes the component values into variables on the root "VRUIntroPage" VoiceXML page. These variables are known as "ConceptTopics", (CTs).

Concept Topics match to the IVR data as listed in the following table:

Source Key	ConceptTopic (CT) name
CLI	CTCLI
CUST_KNOWN	CTDataExists
ACCT_BALANCE	CTAccountBal
OVERDUE_AMT	CTOverdueAmt
SUBTYPE	CTSubType
SV248	CTDialCode
OLC	CTOLCSetting
DUE_DATE	CTDueDate
NEXT_STMT_DATE	CTStatementDate
ICMS_INTERACTIVE	CTICMSStatus
CC_OPEN	CTAgentsAvailable
CIRN	CTCIRN
SEQUENCE_NUMBER	CTFromTSE

1.4.6 IVRDIP Database tables (Genesys)

IVRDIP is the Oracle database used in the IVR system. The five IVRDIP database tables used in the Speech Navigator process are:

Table	Description
Route_start_1	The existing IVR database
Speech_Profile	For each type of speech treatment (profile) a list of data fields required, and connection (SIP) information.
Speech_to_IVR	For each type of returned call, a list of default information
Seq_Num_Lookup	Information for Genesys skill-based routing to CSRs
Welcome_Msg	Customised Welcome messages.

1.4.6.1 Route_start_1

This is the existing IVR database table. However the column SPEECH_PROFILE has been added to link to the Speech data in the following tables.

Column Name	Bytes	Comments
DNIS (key)	10	Number the caller dialled
CLI	32	Number called from
DNIS_GROUP	30	
SV248	10	basic routing code, type
IVR_NAME	20	IVR to run
STRATEGY_NAME	30	
IVR_PLATFORM		
SPEECH_PROFILE	12	Link to Speech_Profiles table

For Speech, the main points are

IVR_NAME=IVR2Speech,

SPEECH_PROFILE field, which is the name of a Speech_Profile, as described in the next table.

SV248, passes data from the SPEECH_PROFILES table back to the IVR system

1.4.6.2 Speech_Profiles

This is a new table made for the Speech Navigator implementation. The main data it holds is:

DATA_REQUIRED – what information to extract from ICMS

DEFAULT_IVR_NAME – the IVR to run if Speech Navigator is not working

SIP information – used to establish the VoIP connection to Speech Navigator.

Column Name	Bytes	Comments
PROFILE	12	Profile name, from Route_Start_1
PERCENT_SPEECH		usually 100%. Speech Decision made elsewhere
CIRN_REQUIRED		True/false
DATA_REQUIRED	128	A list of the field names of data required for this profile
DEFAULT_IVR_NAME	20	State table to run if Speech Navigator not operational
SIP_SWITCH_NAME	16	
SIP_ROUTE_POINT	8	SIP information, to connect to Speech Navigator
SIP_MANAGERS	128	
CC_OPEN_TIME	8	
CC_CLOSE_TIME	8	The times that the call centres are open
CC_OPEN_WEEKENDS		

1.4.6.3 Speech_To_IVR

The Speech_To_IVR table provides a holding place for data sent back from Speech Navigator to the IVR.

- ID is the Speech Navigator exit code ("SPEECH_ID"), indicating the post-speech handling, plus the extent of authorisation performed. The values of these IDs are listed in the return code table on page 17.
- The state-table and table label indicate what IVR section is to play after speech
- The SV248 may be a corrected value: the caller dialled 126, but wanted the world clock.

Column Name	Bytes	Comments
ID	6	Speech_ID. See the table on page 17.
HANDLING	16	
SEQUENCE_NUMBER	5	True/false
STATE_TABLE	20	
TABLE_LABEL	20	
MAIN_STATE_TABLE	20	Used for backup scenarios
SV248	5	Numeric
CIR_REQUIRED	1	Y/N
CIR_ASKED	1	Y/N
CIR_KNOWN	1	Y/N
AUTH_REQUIRED	1	Y/N

1.4.6.4 Seq_No_Lookup

The data in this table is used for Genesys skill based routing. URS uses it in ascertaining the appropriate group of agents, to which the call will be forwarded. This table supplies the skill-level required (SEQ_SKILL), which is compared to the registered skill-levels of the agents currently working.

Column Name	Byte	Comments
SEQ_NO (key)	5	Primary key
DEBT_AGE	10	
OVERRIDE_RULE	6	
VIRTUAL_QUEUE_NAME	50	
STRATEGY_NAME1	50	
SEQ_SKILL	25	
WELCOME_ID		New column, defaults to 0
SELF_SERVICE_INFO	36	New column

1.4.7 GENESYS

Genesys is the CTI system that

- manages the routing throughout the IVR system,
- provides the environment within which Speech Navigator runs: GVP.

The components of the Speech Navigator Genesys system are listed in this section.

1.4.7.1 TServer

A TServer is a Genesys connection to an external device, usually a PABX.

1.4.7.2 IServer

The IServer functions as in interface between a computerised IVR system, and Genesys. To the IVR it appears as a network interface, while to Genesys it appears as a T-Server. In fact it *is* both of these.

1.4.7.3 ISCC

ISCC is a separately licensed Genesys product, used to allow transfers between T-Servers, in this case the IVR – RS/6000 and the GVP. Two ISCC are needed, one for each site. It is assumed that <Company 1> already has these, as they are required for the existing the Symposium-Link / Meridian T-Server

1.4.7.4 URS: Universal Routing Server

The Universal Routing Server supplies routing information to the remainder of Genesys. This is a core Genesys product.

1.4.7.5 GVP: Genesys Voice Platform

GVP provides IVR-like services using the Genesys environment. In the <Company 1> environment GVP is used as a platform for running the TuVox Speech Navigator system. The GVP platform will accept up to 200 concurrent calls across the two sites (constrained to a maximum of 160 on any single site OSR Servers/licenses). This is a licensing decision.

GVP:NE consists of one or more IP Communications Servers and a SIP Session Manager, which contains a SIP Proxy and a Resource Manager The IP Communications Server (IPCS) and Voice Communications Server (VCS) communicate with the Voice Web Manager (provisioning, reporting and policy). GVP: NE – Genesys Voice Platform Network Edition (v 7.0.x) is used in the Speech Navigator implementation.

1.4.7.6 IPCS: GVP IP Communications Server

IP Communications Servers terminate and process VoIP (Voice over IP) calls using SIP. The IPCS will execute the Speech Navigator application by parsing and executing the tags contained in the VoiceXML documents produced by TuVox. GVP IPCS will also accept touch-tone input and verbal commands through embedded speech recognition technology, present pre-recorded voice files or synthesized Text-to-Speech, and allow network-level queuing. The Voice Web Manager centrally manages distributed IP Communications Servers..

1.4.7.7 AIM: Advanced Interface Module

The AIM or Advanced Interface Module runs the Genesys Queue Adapter which enables GVP:NE to integrate with the Genesys Routing Solution via the Genesys I-Server.

1.4.7.8 LDAP: Lightweight Directory Access Protocol

The LDAP server runs the Sun One LDAP server and is the repository for the GVP: NE configuration information. This server and database supplies Genesys applications with detailed location and access information.

1.4.7.9 VWM: Voice Web Manager

Voice Web Manager is centralized management software allowing the multi-tenanted configuration and management of the Voice Web Central Office complexes. . The Voice Web Manager centrally manages the distributed IP Communications Servers.

1.4.7.10 SIP Resource Manager

The SIP Resource Manager maintains resource state for IP Communications Server resources and consists of

- the SIP Session Manager and
- the Resource Manager.

The SIP Session Manager is a SIP Proxy that participates in call control transactions, and works in conjunction with the Resource Manager. The SIP Session Manager is a component of the Voice Web Call Manager.

The Resource Manager manages the IPCS/VoiceXML resources. It understands the characteristics of a set of VoiceXML ports, and so manages pools of resources for the SIP Session Manager. Based on application provisioning parameters, the SIP Session Manager uses the Resource Manager to identify the IP Communications Server end point for a given inbound call.

1.4.7.11 Voice Web Call Manager Database

The Voice Web Call Manager Database is a standalone SQL database server required to store the call state and resource information. The Resource Manager uses this database to supply information to the SIP Session manager.

1.4.7.12 Voice Web Provisioning System (VWPS)

The Voice Web Provisioning System is an application that permits the multi-tenanted configuration of the Voice Web Central Office (VWCO) complexes. It also allows administrators to create and configure applications, assign URLs to dialled numbers, assign port thresholds per application, and enforce other customer and application specific policies.

1.4.8 TuVox

TuVox integrates and presents the components implementing Speech Navigator. It is fundamentally a web-server system generating custom VoiceXML pages.

- delivers the voice user interface of Speech Navigator, by generating and presenting customised VoiceXML pages.
- maps stated caller intent to end destinations,
- provides a connection to the OSR speech recognition system
- passes route instructions back to the IVR upon call completion

The three main components of

< Diagram removed >

1.4.8.1 TuVox Application Server

The application server contains the runtime content of the Speech Navigator application. The application server contains

- the web server component for serving up the VoiceXML pages to the Genesys GVP IPCS voice web server.
- a Tomcat web server.
- a MySQL database.

1.4.8.2 TuVox Reports Server

Provides web based application level reporting that are used for both technical tuning and business management. Data from the speech application is saved to a data warehouse for generating standard and custom reports.

1.4.8.3 JavaScript add-ins

The TuVox environment accommodates and encourages the development of JavaScript additions to the existing VoiceXML scripts. These are added in the "TuVox Producer" for simple, custom tasks.

1.4.9 OCS VOICE RECOGNITION

The voice-recognition is accomplished by the ScanSoft Open Speech Recogniser (v3.0.x), along with Speakfreely, Realspeak (v4.0.x).

There are seven Speech Navigator structures through which a caller can navigate:

- Disambiguation Structure,
- Complex Structure,
- Customer Specific InfoBlast,
- quick-words short-cut,
- Vague Response,
- Guide Me Menu Structure
- destinations

1.4.9.1 Disambiguation Structures

Disambiguation structures are the large number of pre-determined paths that a call may proceed down, in the process of finding the correct solution. These determine the main constructs of Speech Navigator. Listing these is beyond the scope of this document.

1.4.9.2 complex structures

Complex Structures are sets of specific voice prompts, structures, and information, pre-panned for more complex topics. There are 25 Complex Structures in Speech Navigator:

Anytime Info, Broadband Connection Problem, Broadband Email, Broadband Helpsetup, Call Diversion, Call Minder, Call Restriction, Call Waiting, Caller Display, Calling Cards, Directory Assistance, Faults Crossed Lines, Faults Generic Do Unplug Test, Faults Mobile, Faults Unable To Connect, Mobile Cellular Secretary, Mobile Roaming, Mobile Text Messaging, Nuisance

Calls Text, Off Peak Hours, Payment Options , Phonebook Request, Toll International Australia, Toll International Cap, Toll National Specials,

1.4.9.3 Customer specific / InfoBlast

These are custom-generated blocks of information recited by Speech Navigator. There are two of these: Account information summary, and collections (debt). Recitations such as these are referred to as "InfoBlasts" in TuVox terminology.

1.4.9.4 Guide Me Menu Structure

Should Speech Navigator "realise" that the caller is having trouble with the Speech Navigator system itself, then control will be diverted to a "GuideMe" Menu Structure, informing the caller about Speech Navigator and how to use it.

Guide Me structures are supplied by TuVox, and are not able to be described here.

1.4.9.5 destinations

The "destinations" are the internal routing points within Speech Navigator. After each part of a disambiguation structure, complex structure, and at the end of a caller's path through the TuVox system, the caller will end up at one of these "destinations". It is on the basis of the current destination that Speech Navigator decides what – creates custom VoiceXML scripts – to do next.

1.4.9.6 quick-words short-cuts

Quick-words are implementations of the OSR “Voice Enrolment” functionality. If the caller’s utterances are identified as one of any “quick words” the caller is immediately re-routed to the appropriate destination.

quick-word	Destination
Account Number	billing_account_number
Automatic Payment Form	payments_ap_form
Busy Line Check	faults_busyline
Call Diversion	call_diversion
Call Minder	call_minder
Call Restriction	call_restriction
Call Waiting	call_waiting
Caller Display	caller_display
Calling Australia	toll_international_australia
Calling Cards	calling_cards
Cellular Secretary	mobile_cellular_secretary
Check Bill Dates	billing_billdates
Collect Calls	collect_call
Country Codes	country_code
Credit Card Payment	payments_creditcard
Direct Debit Form	payments_directdebit_form
Fly Buys	flybuys
Get Bill Copy	billing_billcopy
Get My Account Details/Account Balance	billing_accountsummary
International Specials	toll_international_cap
Last Payment Summary	billing_last_payment_details
National Toll Plans	toll_national_specials
Off Peak Hours	off_peak_hours
Payment Options	payment_options
Phonebook	phonebook_request
Prepaid Top up	mobile_prepaid_topup
Record Receipt Number	collections_record_receipt_number
Roaming	mobile_roaming
Setup Broadband	broadband_helpsetup
Text Messages	mobile_text_messaging
Troubleshoot Broadband	broadband_technicalsupport
Unbilled Calls	billing_unbilled_calls
World Clock	time_of_day
World Clock	world_clock

1.4.9.7 Vague response

Vague Response (processes) are variations on: “can you be more precise ? here’s a hint.”. Within each step of a Vague Response structure the “Caller Replies, Reply Interpreted” stage is executed.

1.4.9.7.1 Core Destinations

Seq Num	Destination	Conditions	Sample Transcription(s)
0		Successful disconnect event.	
1-100	Speech Navigator errors	Error: Not yet allocated	
5000	agent_request	DEFAULT CODE direct agent transfer destination	I want to talk to someone, can I talk to an operator

1.4.9.7.2 All Destinations

Seq Num	Destination	Conditions	Sample Transcription(s)
0		Reserved for successful disconnect event.	
1-100	Speech Navigator errors	Not yet allocated	
5000	agent_request	DEFAULT CODE direct agent transfer destination	I want to talk to someone, can I talk to an operator
1000	billing_accountsummary	InfoBlast/TSE IVR needs to ask for the CIRN	my account details please, my account balance
1001	billing_accountsummary	InfoBlast/TSE IVR needs to ask if CLI=CIRN	my account details please, my account balance
1002	billing_accountsummary	InfoBlast/TSE CIRN is known, IVR needs to ask for PIN authentication	my account details please, my account balance
1100	collections_extension	InfoBlast IVR needs to ask for the CIRN	i want an extension on my time to pay my phone bill, trying to find out of i can get extended time on my phone bill
1101	collections_extension	InfoBlast IVR needs to ask if CLI=CIRN	i want an extension on my time to pay my phone bill, trying to find out of i can get extended time on my phone bill
1102	collections_extension	InfoBlast CIRN is known, IVR needs to ask for PIN authentication	i want an extension on my time to pay my phone bill, trying to find out of i can get extended time on my phone bill
1200		InfoBlast/TSE IVR needs to ask if CLI=CIRN, PIN is not required	
1201		InfoBlast/TSE	

Seq Num	Destination	Conditions	Sample Transcription(s)
		CIRN is known, PIN not required	
1300		TSE Only IVR needs to ask if CLI=CIRN	
1301		TSE Only CIRN=CLI and PIN is not required	
1302		TSE Only CIRN is known, IVR needs to ask for PIN authentication	
1400	billing_account_number	TSE Only IVR needs to ask if CLI=CIRN	what's my account number, I can't find my account number
1401	billing_account_number	TSE Only CIRN=CLI and PIN is not required	what's my account number, I can't find my account number
1402	billing_account_number	TSE Only CIRN is known, IVR needs to ask for PIN authentication	what's my account number, I can't find my account number
1500	billing_billcopy	TSE Only IVR needs to ask if CLI=CIRN	I need a copy of my bill, I need a statement
1501	billing_billcopy	TSE Only CIRN=CLI and PIN is not required	I need a copy of my bill, I need a statement
1502	billing_billcopy	TSE Only CIRN is known, IVR needs to ask for PIN authentication	I need a copy of my bill, I need a statement
1600	billing_billdates	TSE Only IVR needs to ask if CLI=CIRN	i want to know when my next bill is due, ohh i'm actually just wondering if i can check some bill dates
1601	billing_billdates	TSE Only CIRN=CLI and PIN is not required	i want to know when my next bill is due, ohh i'm actually just wondering if i can check some bill dates
1602	billing_billdates	TSE Only CIRN is known, IVR needs to ask for PIN authentication	i want to know when my next bill is due, ohh i'm actually just wondering if i can check some bill dates
1700	billing_last_payment_details	TSE Only IVR needs to ask if CLI=CIRN	I want to know if I've paid an account, please
1701	billing_last_payment_details	TSE Only CIRN=CLI and PIN is not required	I want to know if I've paid an account, please
1702	billing_last_payment_details	TSE Only CIRN is known, IVR needs to ask for PIN authentication	I want to know if I've paid an account, please

< eleven pages of data removed >

1.4.10 VOICE XML COMPONENTS

VoiceXML scripts provide the caller's interface to Speech Navigator, as far as the caller is concerned they are Speech Navigator.

1.4.10.1 Speech Navigator VoiceXML structure

As detailed in the next section, one root VoiceXML script and usually one leaf VoiceXML script are active; the leaf script running within the root script's environment. The root script holds the customer and call data sent from the IVR system, holding these in "ConceptTopics", and making them available to the leaf scripts.

The VoiceXML scripts are supplied by TuVox, being generated by the internal TuVox Java applications and made available through the TuVox's Tomcat webserver. The current "destination" *q.v.* is the major input to the VoiceXML script generation process.

1.4.10.2 Root document function and description

Because the application is a set of VoiceXML documents sharing the same application root document, whenever the user interacts with a particular VoiceXML document, its application root document is also loaded. The application root document remains loaded while the user is transitioning between other documents in the application. While it is loaded, the application root document's variables are available to the other documents as application variables, and its grammars remain active for the duration of the application. Thus one of the following two conditions always holds during interpretation:

- The application root document is loaded and the user is executing in it, and there is no leaf document.
- The application root document and a single leaf document are both loaded and the user is executing in the leaf document.

When a leaf document load causes a root document load, none of the dialogs in the root document are executed. Execution begins in the leaf document.

- The root document's variables are available for use by the leaf documents, so that information can be shared and retained.
- Root document <property> elements specify default values for properties used in the leaf documents.
- Common ECMAScript code can be defined in root document <script> elements and used in the leaf documents.
- Root document <catch> elements define default event handling for the leaf documents.
- Document-scoped grammars in the root document are active when the user is in a leaf document, so that the user is able to interact with forms, links, and menus in the root document.

1.4.11 Voice Recognition components

1.4.11.1 Utterances

Utterances are the G711 audio streams sent from the caller via the GVP to the OCS Speech Detector.

1.4.11.2 Speech Detector

The Speech Detector collects short, isolated, blocks of pure speech from the utterance.

1.4.11.3 OSR: Open Speech Recognizer

The Open Speech Recognizer performs the conversion of spoken words into computer text. Speech is first digitized and then matched against a dictionary of coded waveforms, “grammars”. The matches are converted into text as if the words were typed on the keyboard.

1.4.11.4 Slots

“Slots” are the key words that may be found in a recognised utterance. Examples would be “money” “cell phone” “disconnect”.

1.4.11.5 OSR Statistical Language Model (SLM)

Matches the gathered “slots” and related information to an ‘interpreted response’, an internal measure of what was said.

1.4.11.6 Semantic Mapping and Destinations

The Semantic Mapping connects the interpreted responses to speech destinations.

1.5 OTHER CONNECTIONS TO SPEECH

1.5.1 Adjusting Speech Navigator global (ASR) parameters

As a general rule, most applications are implemented with standard default values of ASR parameters. There are a large number of parameters that are never changed in any implementation, but some common ASR parameters that tend to be modified most often include:

Parameter	Description
bargein	Set to ‘true’ allows a caller to interrupt a prompt with a response, otherwise set to ‘false’ to prevent a caller interrupting a prompt. Set to ‘true’ by default.
incompletetimeout	Controls the length of a period of silence after callers have spoken to conclude that they finished. This timer is also known as the “end of speech” timeout (or more correctly, the after end of speech timeout).
timeout	Controls how long the speech detector should wait after the end of the prompt for the caller to speak. When this duration has expired, the <noinput> event is triggered.
sensitivity	Controls sensitivity of the speech detector when looking for speech.

If parameters need to be changed for an implementation, or even for a particular dialog state, this would be done via the VoiceXML <property> tag, accessible directly in Producer.

1.5.2 From external organisations

IPSEC connections remotely from TuVox applications into the <Company 1> Access Gateway (AKLtag01FW) then to AKLpaiFW, then to a switch

1.5.3 To External Organisations

This subject was discussed with Blair Thompson. Noting the absence of code or structures enabling transfer outside the Speech Navigator environment, though the requirements for such transfers in the Speech Navigator specifications, it was demonstrated that such transfers, to any external system, are of such little difficulty – effectively dial the phone number – that designing for these is wasted effort.

In effect, transfer to an external organisations is available, though there is currently no explicit structures actualising such a transfers.

1.5.4 Software updates from TuVox

As shown in the diagram from TuVox, updates are sent to the Testing environment, from where they are uploaded to the production environment.

< Diagram removed >