



IP Office - Job Aid

Using Databases within Voicemail Pro

Summary

Voicemail Pro can support 3rd party databases and manage database requirements. Only one database can be supported per session. The VmPro Client provides icons to allow database actions to be configured by a system administrator as part of a callflow.

Contents

VOICEMAIL PRO DATABASE USAGE	4
INTRODUCTION	4
DATABASE ACTIONS	5
<i>Database Open Action</i>	5
<i>Database Execute Action</i>	6
<i>SELECT...FROM SQL Function</i>	7
<i>INSERT...VALUES SQL Function</i>	8
<i>Examples</i>	8
<i>Database Get Data Action</i>	9
<i>Database Close Action</i>	9
DATABASE USAGE EXAMPLE.	10
SCENARIO	10
RETRIEVE DATA FROM A DATABASE	11
<i>Database Open Action</i>	12
<i>Database Execute Action</i>	13
<i>Database Get Data Action</i>	14
RETUNING DATA FROM THE DATABASE.	15
<i>Speak Book Title</i>	16
ENTER DETAILS INTO THE DATABASE	17
<i>Confirm Book Details</i>	18
<i>Collect Callers Details</i>	18
CLOSING THE DATABASE.....	19

Voicemail Pro Database Usage

Introduction

This document contains an example on the use of a database within a call flow together with detailed information how the database is accessed.

The call flow used in the exercise is an example only and not intended to reflect real-life customer auto-attendant and other voicemail functions.


The call flow example used in this document will be available on the IP Office Engineers toolkit 2.0. This can be imported into your Voicemail Pro if required.

Note: Importing a callflow database will overwrite any existing callflow.

Database Actions

 These actions relate to retrieving and adding data to a database.

Database Open Action

 This action provides the ability to link into a third party database. If connection to the database succeeds then **Success** is returned, otherwise **Failure**. If there is a connection to a database already then the current connection is closed and the new one requested will be opened.

Specific Tab:

The connection string to open the database can be entered directly into the field. For help on constructing the connection string use the browse button to open the Data Link Properties form.

- **Provider Tab:** Select the OLE DB Provider of the data that is to be connected to. Click Next to move to the Connection Tab.
- **Connection Tab:** Specific information relating to the Database provider needs to be completed. Help of the information required for each of the fields can be obtained by clicking the Help button. Fields available will be dependant on the type of provider. Test that the information entered will allow entry into the database, click the Test Connection button. A message Test Connection Succeeded will show if successfully connected to the database.
- **Advanced Tab:** Network Settings & other settings e.g. Access permissions. Fields showing will be dependant on the type of provider selected. Click help for specific information about any of the fields.
- **All:** The properties that have been selected on the previous tabs are shown in the All Tab. Amendments can be made as required by selecting the Name and click Edit Value,

Database Execute Action

 This action allows the administrator to enter a query on a database.

Specific Tab

When first opened the SQL Wizard appears which can be used to build the query. The query can also be entered in SQL directly into the space on the specific tab.

SQL Wizard Interface.


- **Select the relevant Database Open Icon.** A list of the Database Open actions that have been created are shown in the drop down menu. Select the required Database Open action if there is more than one. If only one is present the Database open action name will be automatically selected and path details shown.
- **Select Delimiters.** The String and Date delimiters need to be selected from the drop down lists. The default delimiters are String = " and Date = #. The delimiters may need to be changed depending on the type of database being used. When completed click **Next**.
- **SQL Function.** You are able to select information from the database or enter information into the database. Choose one of these options and click **Next** to view the SQL Wizard form. Depending on which SQL Function was selected controls which form is displayed.

SELECT...FROM SQL Function

The screenshot shows the SQL Wizard interface. At the top, the 'Function' is set to 'SELECT', 'Tables' is 'BookList', and 'Field' is 'Selected Fields'. Below this, a criteria row is defined with 'Author' in the Logical field, '=' in the Relational field, and an empty Data field. The 'AND' operator is selected. An 'ADD' button is present. A table below shows the criteria being added:

Logical	Field	Relational	Data
Author	Author	Contains	\$key

At the bottom of the dialog are '< Back', 'Finish', and 'Cancel' buttons.

- **Function.** The function SELECT is automatically entered.
- **Tables.** A drop down list shows all the tables contained within the database. Select the table to be searched from the listing.
- **Field** The drop down list shows all the fields contained within the selected table. Select the field that you want to retrieve information from when the input matches the criteria. When information from more than one field is needed, specify the fields by clicking  and select the required fields. Up to six fields from a database can be included within a search.

Criteria Section The lower half of the screen allows the criteria of the search to be defined. Details entered in the fields will be added to the table shown below the fields when the **ADD** button is selected.

- **Field:** Select field to be searched from the dropdown list
- **Relational:** Select the operator that is required from the list that contains "=", "<>", "<", ">", "<=", ">=", "Contains" and "Starts with".
- **Data:** Enter the data value or variable to be used. For example \$key would be entered to return the last DTMF sequence i.e. the callers input from their telephone.
- **Logical:** If more than one criteria is required use the operators AND or OR then enter the next criteria.

Following this action you may need to use the **Database Get Data** action. This will allow the user to navigate through the records that match the search criteria.

INSERT...VALUES SQL Function

Field	Values
ContactTelephone	\$CP4
Cost	\$DBD[1]
CreditCardExpiry	\$CP3
CreditCardNumber	\$CP2
ISBN	\$DBD[2]

- **Function.** The function INSERT is automatically inserted.
- **Tables.** A list of tables contained within the database are shown in the drop down list. Select the table that will be updated.
- **Field.** All the fields within the selected tables will be listed in alphabetical order.
- **Values.** The System or user defined variables that need to be inserted into the database.

Examples

\$DBD[1] would return the value contained within the second column of a virtual database. The virtual database is created when a Select From SQL action is used. When any input matches the specified criteria, information from the selected fields are stored in memory. The field names that are selected are arranged in alphabetical order.

For example, the Select from SQL function could be used to return the values contained within the fields Reference, Author and Title from a table called booklist. If the selection criteria is matched the information contained within the fields would be saved in the following order


Column 1 - Author
 Column 2 - Reference
 Column 3 - Title.

If details from the Title column are to be returned then the value \$DBD[2] would need to be entered.

\$DBD[0] would return details from Column 1 - Author
 \$DBD[1] would return details from Column 2 - Reference
 \$DBD[2] would return details from Column 3 - Title

Note: Fields are counted from 0 through to 5 enabling up to 6 fields to be included

Database Get Data Action

 Once a query has been made against a database then either a single or a set of results are returned. This action will allow access to the data items that are returned.

Specific Tab


There are four options on how the data can be retrieved.

- **Retrieve the next item in the list.** Allows the call flow to facilitate the stepping through of a list or results returned by the Database Execute action.
- **Retrieve the previous item in the list.** Allows the call flow to facilitate the stepping through of a list or results returned by the Database Execute action.
- **Retrieve the first item in the list.** Allows the call flow to facilitate jumping to the start of the list.
- **Retrieve the last item in the list.** Allows the call flow to facilitate jumping to the end of the list.

The Database Get Data action has the following possible outcomes:

- **Success:** The current record has successfully been assigned to the \$DBData variable(s).
- **At End:** You have reached the end of the list.
- **Empty:** The execute method returned no data.
- **Failure:** There was a problem trying to retrieve the next data record.

Database Close Action

 This action will close the current database connection. If the database is open when a call terminates then the Database Close action will be automatically run.

Database Usage Example.

Scenario

In this callflow example an auto-attendant has been created to allow callers to order books. The book details are held within a Microsoft Access database, other databases can be used. Callers will be able to enter either the ISBN or Author's name. The title and cost of the item will be looked up allowing the caller to purchase the item if they wish to. If the caller purchases the book they will be able to enter their credit card details and a contact number.

The call flow example used in this document will be available on the IP Office Engineers toolkit 2.0. This can be imported into your Voicemail Pro if required though importing a callflow database will overwrite any existing callflow.

Note: It is assumed that users are experienced in creating callflows. Only the new actions relating to Databases are explained in detail.

The parts of the callflow explained are:

- Retrieving Data from a database on page 11.
- Using data retrieved from the database on page 15.
- Entering details into the database on page 17.

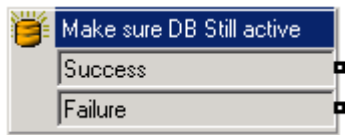
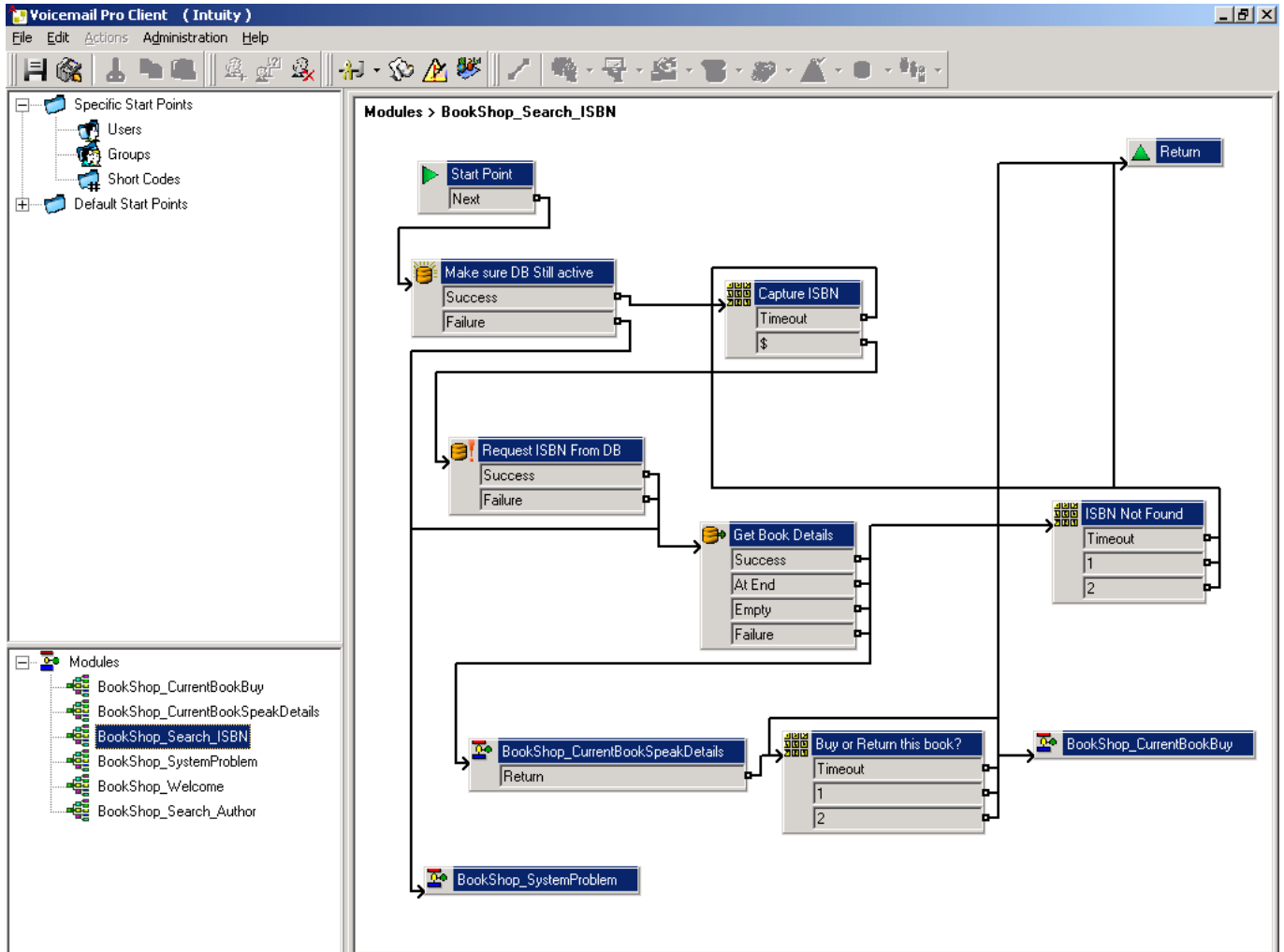
Example of the database that is used in the callflow.

BookList : Table				
	ISBN	Author	Title	Cost
	1001	James Herbert	The Fog	£4.30
	1002	James Herbert	The Rats	£2.59
	1003	James Herbert	The Lair	£2.65
	1004	James Herbert	Domain	£2.67
	1100	Frank Herbert	Dune	£9.99
	1101	Frank Herbert	Dune Messiah	£7.50
	1102	Frank Herbert	Children of the Dune	£9.95
	1111	Mary Shelley	Frankenstein	£4.40
	1212	Bram Stoker	Dracula	£3.60
	2222	J R Tolkien	The Hobbit	£5.50
	2323	Jane Austen	Pride and Prejudice	£4.80
	3333	J R Tolkien	Lord of the Rings	£6.50
	3434	William Golding	Lord of the Flies	£2.30
	4444	George Orwell	1984	£2.70
	4545	Mark Twain	Huckleberry Finn	£3.40
	5555	George Orwell	Animal Farm	£2.90
	5656	H G Wells	The Time Machine	£4.40
	6666	J D Salinger	The Catcher in the Rye	£3.10
	6767	J K Rowling	The Harry Potter Books	£9.50
	7777	A A Milne	Winnie the Poo	£2.60
	8888	Rudyard Kipling	The Jungle Books	£3.40
	9999	Jonathan Swift	Gulliver's Travels	£2.30
				£0.00

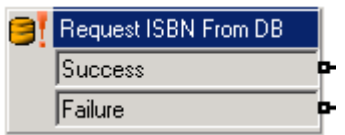
Record: 23 of 23

Retrieve Data from a Database

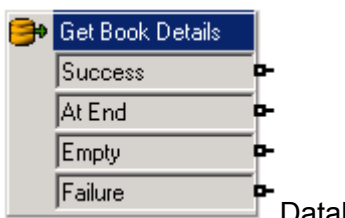
The Bookshop_Welcome module allows callers to choose to search the database by either ISBN or author's name. The screen below shows the callflow module used when a search by ISBN is selected. The database actions that have been used are shown below the callflow diagram, with details on the following pages.



Database Open Action on page 12



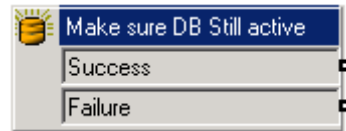
Database Execute Action on page 13



Database Get Data Action on page 14

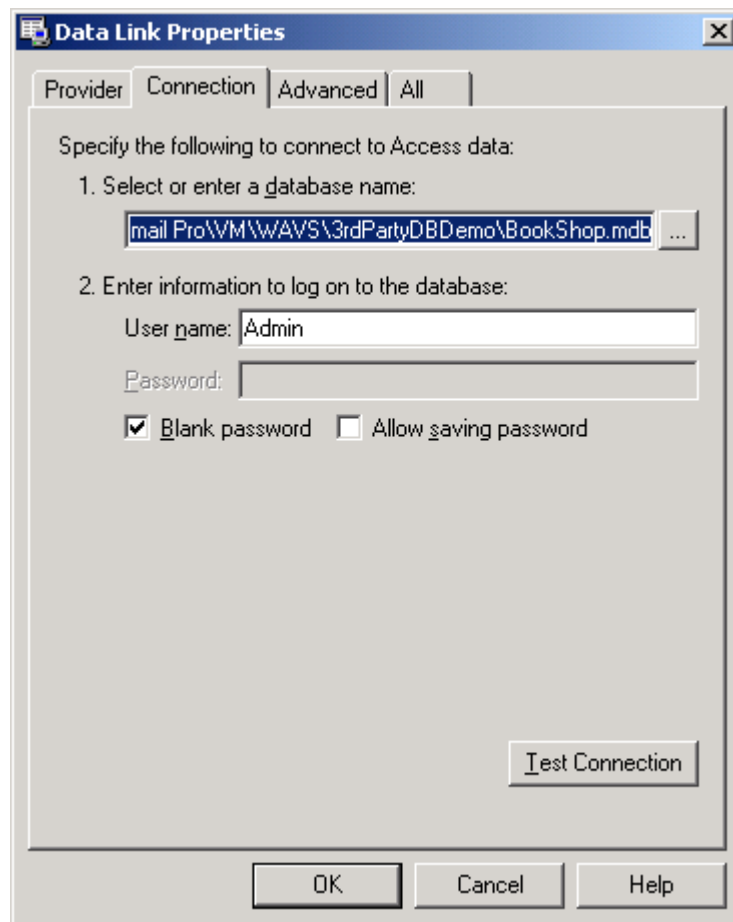
Database Open Action

The Database Open Action is used to link to the bookshop database.



The specific tab of the action contains the location of the database. Click the browse button to view the Data Link Properties dialog. The details entered into these screens will depend upon the type of database used. This example uses a Microsoft Access Database.

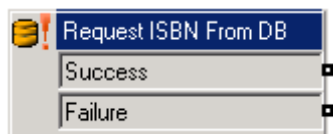
The example shown below shows the connection to the database.



If the database is available the callers move through the callflow to a menu action that will capture the ISBN number entered.

Database Execute Action

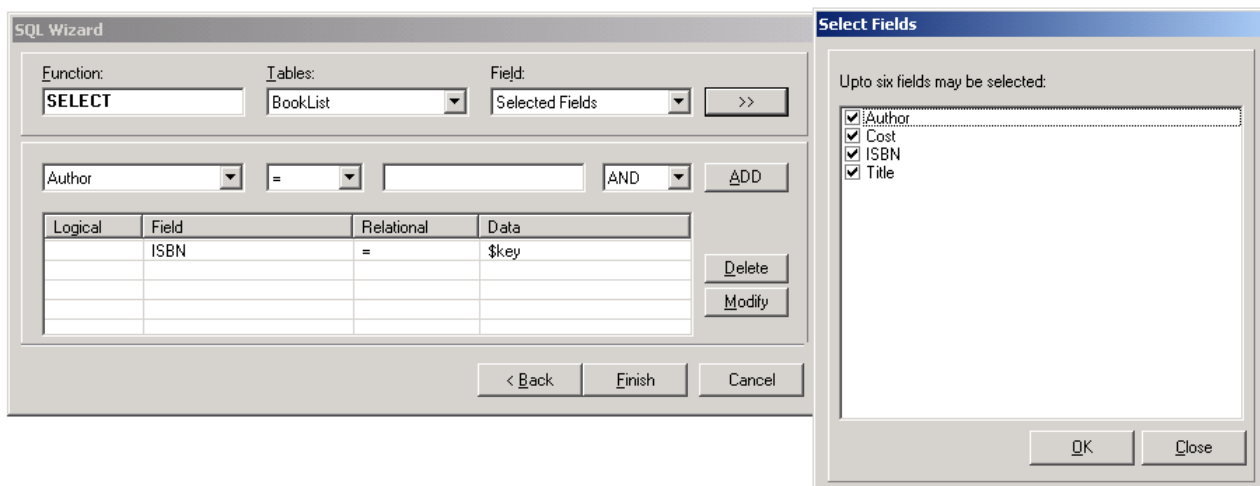
The Database Execute Action contains a query against the open database, in this example it concerns the ISBN captured in the previous menu action.



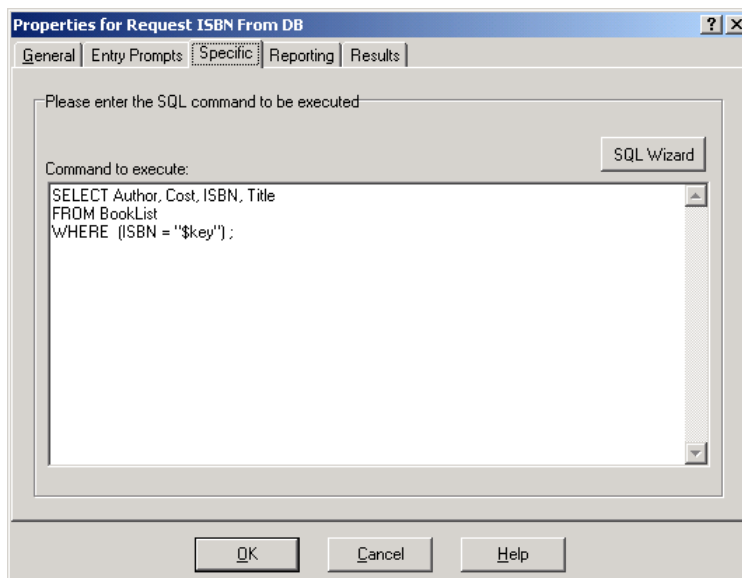
If the sequence of numbers entered by the caller matches an ISBN entry in the database, then the Author's name, cost, ISBN and book title details are captured. This query is entered into the Database Execute Action via the specific tab.

When entering information into the specific tab for the first time you will taken through a series of steps.

1. Select the Database Open Icon required. In this example the 'Make sure Database still active' icon was selected.
2. At the SQL Function window the option to 'Select ...From' was chosen as information from the database is required.
3. Details are then entered into the SQL Wizard, as shown below.

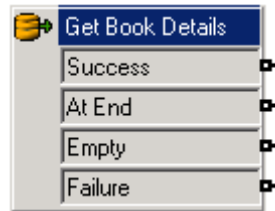


4. When the query has been entered the SQL wizard is closed. The specific tab of the action will contain the entered query, see example shown below.

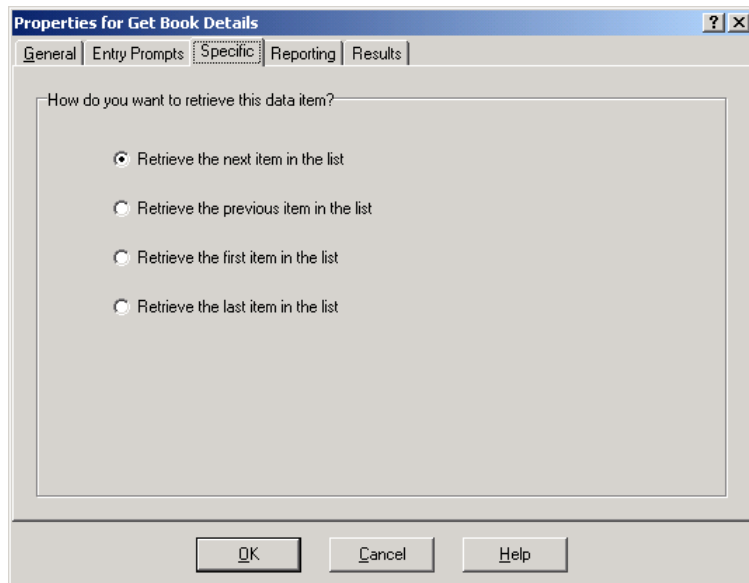


Database Get Data Action

The Database Get Data Action is used to return details of any matching entries following a search against a database.



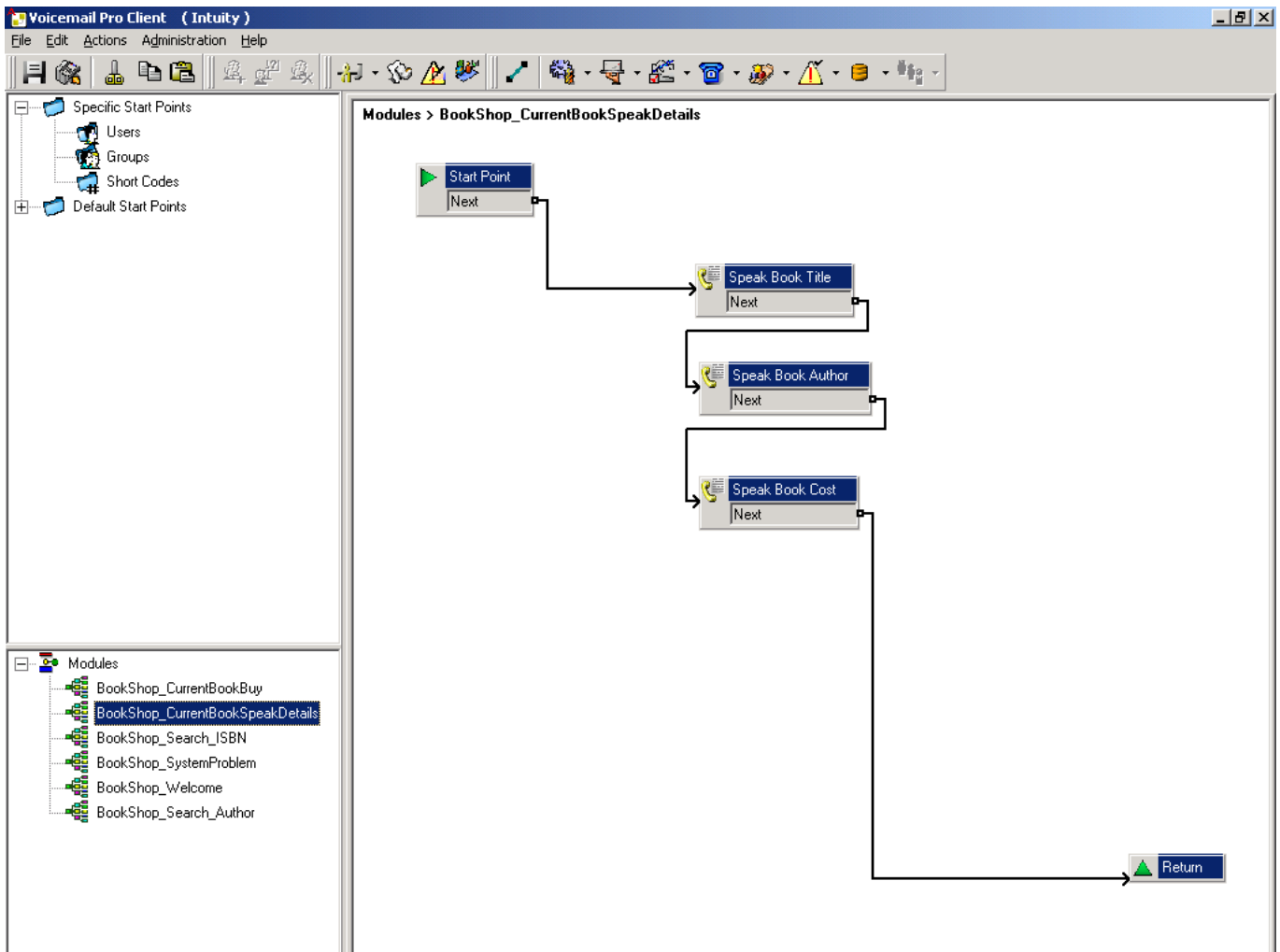
To retrieve the results an option is selected on the specific tab to select how the data is retrieved from the database. In this example the option 'retrieve the next item in the list' was selected to allow the caller to step through the results, if more than one match ISBN occurred.



If a matching ISBN has been found the callflow is routed to another module called 'Bookshop_CurrentBookSpeakDetails'.

Retuning data from the database.

The 'Bookshop_CurrentBookSpeakDetails' module will tell the caller the Book title, the Author's name and the cost of the book with the ISBN that they entered .

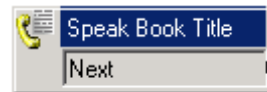


The information from the database is conveyed to the caller using the 'Speak Text Action'.

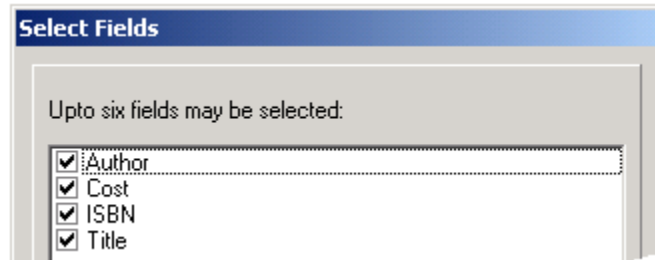
Note To use the Speak Action the system needs to be licensed for Text to Speech.

Speak Book Title

The Speak Book Title action is used to tell the caller the book title associated with the ISBN that was entered.



The fields selected in the 'Request ISBN from DB' action contain the information retrieved from the bookshop database. The fields selected were Author, Cost, ISBN and Title.



Any fields selected in a query will appear in alphabetical order.

\$DBD[0] would return details from the field Author

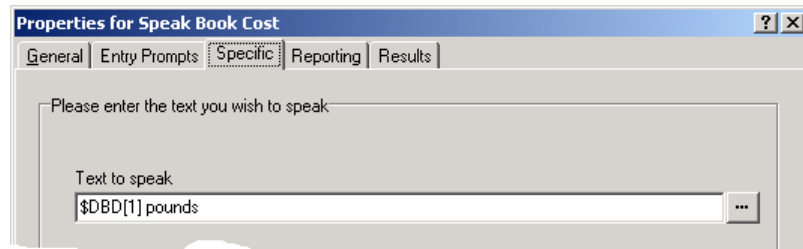
\$DBD[1] would return details from the field Cost

\$DBD[2] would return details from the field ISBN

\$DBD[3] would return details from the field Title.

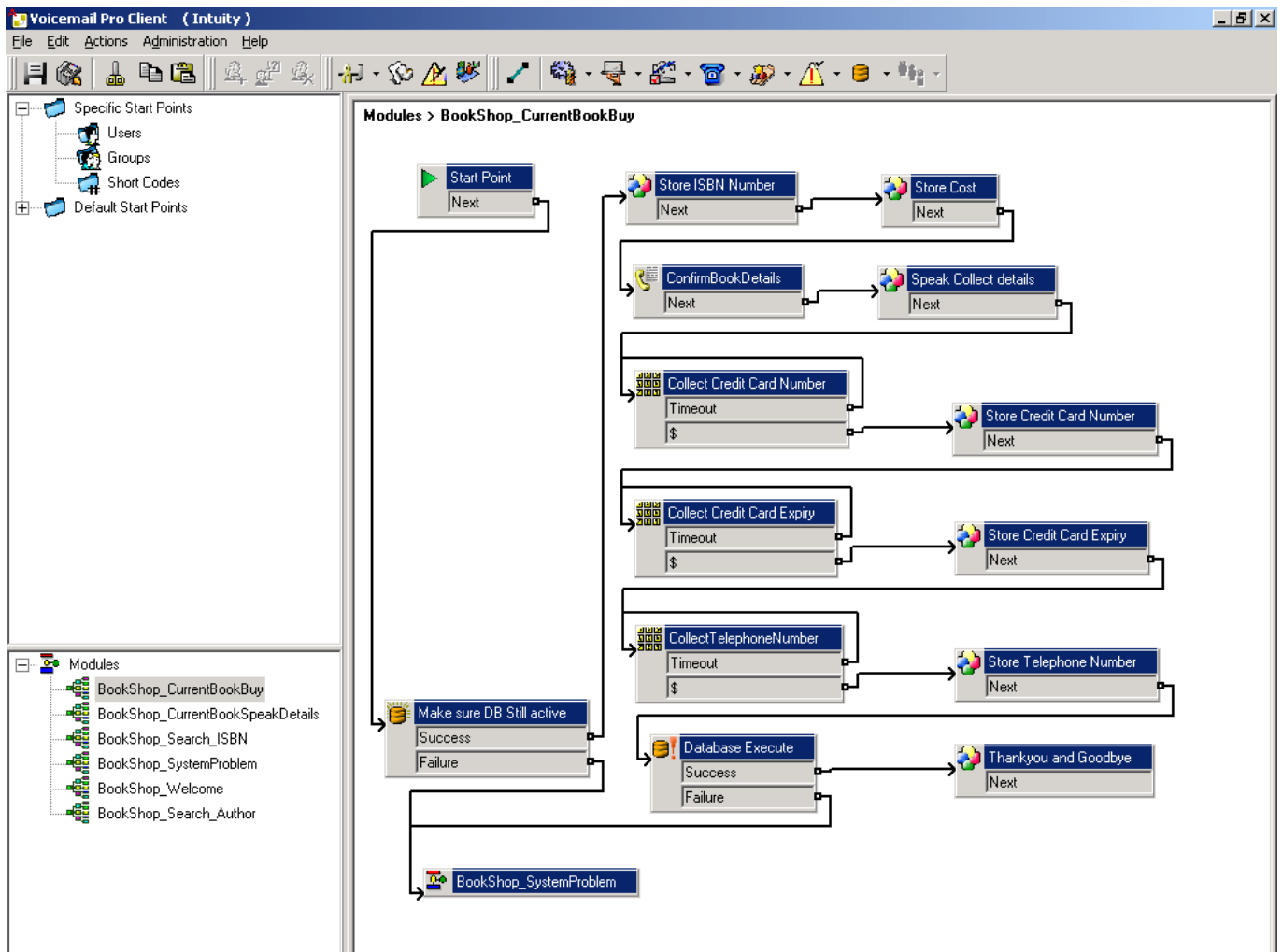
Each speak text action in the callflow returns the values from a different field selected within the database query.

The 'Speak Book Cost' action has additional text added so that the currency can be spoken, in this example pounds are used. See below.



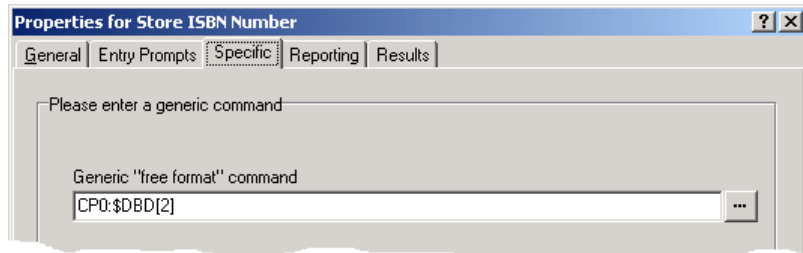
Enter Details into the Database

The caller is given an option to buy the book. If they select to buy the book the callflow module Bookshop_CurrentBookBuy operates. The callflow immediately checks that access to the bookshop database is still available via a Database Open action.

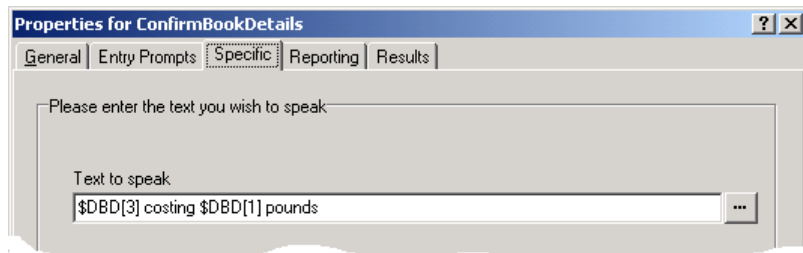


Confirm Book Details

The Generic Action is used to Store ISBN number and the cost. The example below shown how the ISBN number is stored, in the system variable CP0.



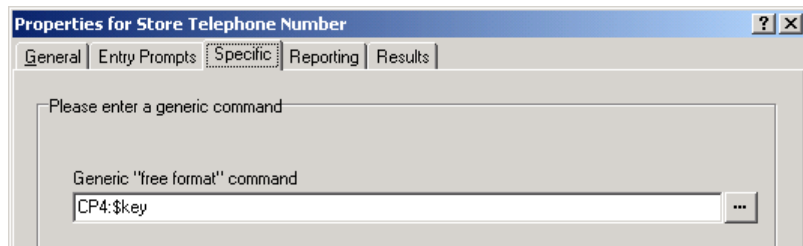
When the details have been stored the book title and cost are spoken to the caller. See the example below.



Collect Callers Details

Details can be entered into a database by a caller. In this example we collect the callers credit card number, expiry date and telephone number. All these details are collected and then the database is updated.

The example below shows the Specific tab entry used to collect the callers telephone number and assign it to the system variable CP4.



When all the details have been collected, the database needs to be updated. The database Execute Action is used. . When entering information into the specific tab for the first time you will taken through a series of steps.

1. Select the Database Open Icon required. In this example the 'Make sure Database still active' icon was selected.
2. At the SQL Function window the option to 'Insert ...values' was chosen as information needs to be added to the database..
3. Details are then entered into the SQL Wizard, as shown below. When the Database table is selected, the list of fields contained within the table is inserted.

Field	Values
ContactTelephone	\$CP4
Cost	\$DBD[1]
CreditCardExpiry	\$CP3
CreditCardNumber	\$CP2
ISBN	\$DBD[2]

When the details have been entered the SQL wizard is closed. The specific tab of the action will contain the command to execute, see example shown below.

```

INSERT INTO OrderDetails
(ContactTelephone, Cost, CreditCardExpiry, CreditCardNumber, ISBN )
VALUES ('$CP4', '$DBD[1]', '$CP3', '$CP2', '$DBD[2]');

```

Closing the Database.

At the end of the callflow the database needs to be closed. If the database is open when a call terminates, then the Database Close action will be automatically run.

Performance figures and data quoted in this document are typical, and must be specifically confirmed in writing by Avaya before they become applicable to any particular order or contract. The company reserves the right to make alterations or amendments to the detailed specifications at its discretion. The publication of information in this document does not imply freedom from patent or other protective rights of Avaya, or others.

Intellectual property related to this product (including trademarks) and registered to Lucent Technologies has been transferred or licensed to Avaya.

All trademarks identified by ® or TM are registered marks or trademarks, respectively, of Avaya Inc. All other trademarks are the property of their respective owners.

This document contains propriety information of Avaya and is not to be disclosed or used except in accordance with applicable agreements.

Any comments or suggestions regarding this document should be sent to "wgctechpubs@avaya.com".

© 2003 Avaya Inc. All rights reserved

Avaya
Sterling Court
15 - 21 Mundells
Welwyn Garden City
Hertfordshire
AL7 1LZ
England

Tel: +44 (0) 1707 392200
Fax: +44 (0) 1707 376933

Email: contact@avaya.com
Web: <http://www.avaya.com>