Dialog Designer
(DD)
Release 5.0.0.17.01
Release Letter

(Supersedes All Previous Releases)

From: Cynthia Hiatt Date: April 12, 2010

Table of Contents

1 Release Information 2
2 Features 2
2.1 New Features included in this delivery 2
2.2 Features/Requirements not included in this delivery 2
2.7 Features/Requirements with Limitations/Restrictions in this delivery 3
2.7 Features with Limitations/Restrictions Removed for This Release 3
2.5 Features with Changes in This Release 3
2.6 New Features in the 5.0 Release 4
2.7 Features with Dependencies on other Products 4
2.8 MRs included in this Release 4
3 Installation and Upgrades 5
3.1 New Installations 5
3.2 Upgrades – General 5
3.3 Upgrades – Upgrades from VP 5.0 to VP SP 1 – IVVR Application Impact 5
3.4 Upgrades with CTI 6
3.5 Upgrades from Dialog Designer 4.1 – CCXML 6
3.6 Upgrades from Dialog Designer 4.0 – no CCXML 6
3.7 Upgrades from Dialog Designer 4.0 – CCXML Applications 6
DTD Changes:* *6
Event Changes:* *7
Object Changes:* *7
3.8 Dialog Designer 4.0/4.1/5.0 Runtime 7
3.9 Co-residency 7
3.10 Additional Application Installation Instructions for Websphere 7
3.11 Additional Application Installation Instructions for WebLogic 8
3.12 Additional Application Installation Instructions for Tomcat 8
4 Known Issues 9
4.1 Limitations, Work Arounds, Notes: 9
5 Version Information 20
6 Hardware Prerequisites 20
7 Software Prerequisites (Supported Versions) 21
7.1 Design Environment 21
7.2 Runtime Environment 21

1 Release Information

This release note describes the content of Dialog Designer 5.0.17. This is the third service pack build for DD 5.0 release.

The primary features of the 5.0 release are:

- Extensibility
  - Custom Navigator
  - Pluggable Data Connector
  - Loquendo **(not currently available on VP)**
  - VRUSM GUI support
  - Created a common Avaya Preferences area and put Dialog Designer and Application Simulation under this common category so that other tools and products can add their preferences to a common area.
- Feature Support for Voice Portal 5.0
  - Variable Administration through the VPMS
  - CCXML basic http i/o processor
  - ASA/UUI Support
  - IVVR Support
- Currency with 3rd party products
  - Support for Tomcat 6
  - Support for Java 1.6
  - Eclipse 3.4
  - WebLogic 10
  - Sysdeo 3.2.1
  - Voice Portal 5.0

2 Features

This section describes the status of the features that were developed for this release.

2.1 New Features included in this delivery

- If an application uses subflows, then all project callflows should be re-generated to make sure that updated servlet identifiers for subflows are generated.
- A new JTAPI driver based on build 4.2.2, build 458 has been included in this Service Pack to pick up the latest fixes.
- A new version of the VP report library is included in this SP to resolve MR 370830. Items fixed are:
  - If the VPMS is version 4.x, then do NOT send out getRuntimeVariables web service calls.
  - Scale the vpAppLogClient. Within one minute of the timeframe, any request to the errored VPMS will be written to savereport.log instead of retry.
  - Add a try/catch block when parsing the duration while reading from savereport.log. If there is a parse error, put duration as 0. Also, do not delete really bad possible corrupted savereport.log; rename the file.

2.2 Features/Requirements not included in this delivery

N/A

2.7 Features/Requirements with Limitations/Restrictions in this delivery

N/A

2.7 Features with Limitations/Restrictions Removed for This Release

N/A

2.5 Features with Changes in This Release

- Upgraded applications that contain a CTI Dial operation will need a manual modification in the data node to add a value to the property: Caller Call Info Variable. Typically, this value is set to cticallinfo which is the default call info variable DD uses for caller CTI information.

- **VPMS Configuration for Application Logging:**

  Tomcat on VPMS

  If you are running a Dialog Designer application that does application logging, and your Voice Portal system is configured to handle more than 75 simultaneous calls, you must make the following configuration changes on your VPMS:

  1. In a text editor, open the *$CATALINA_HOME/conf/server.xml* file.
  2. Find the entry **maxThreads="500"** under **<Connector port="8009"**.
  3. Change the value from **500** to **1000**.
  4. Save and close the file.

  Tomcat on Linux application server

  If you are running a Dialog Designer application that does application logging, and your VPMS goes out of service, Dialog Designer temporarily stores the application log data on your application server while waiting for the VPMS to come back in service. If your application server is Tomcat running on Linux and the VPMS stays out of service for a long time, Tomcat might crash because the system runs out of file handles.

  If you encounter this problem, do one or both of the following on your application server.

  Increase the Linux file handle limit on your application server:

  1. In a text editor, open the */etc/init.d/tomcat* file.
  2. Find the **start()** procedure.
  3. Add the following line to the start procedure before Tomcat is started:

     ```
     ulimit –n 8192
     ```

  4. Save and close the file.

  Change Tomcat startup options to improve garbage collection on your application server:

  1. In a text editor, open the */etc/init.d/tomcat* file.
  2. Find the line that begins with **export JAVA_OPTS=**.

3. Change this line to read as follows.

Note: All of the text below must be entered on a single line.

```
export JAVA_OPTS="-server -Xmx1024m -XX:MaxNewSize=30m -X:+UseParNewGC -
XX:+UseConcMarkSweepGC -XX:CMSInitiatingOccupancyFraction=60 -
XX:ThreadStackSize=512"
```

4. Save and close the file.

2.6 New Features in the 5.0 Release

The following is a high level list of new features for this release. For a detailed list, please refer to the requirements document.

- Custom Navigator
- Pluggable Data Connector
- Loquendo **(not currently available on VP)**
- VRUSM/IC 7 support
- Variable Administration through the VPMS
- CCXML CreateSession support
- ASA/UUI Support
- IVVR Support
- Tuning Tools **(for internal CSI use only)**

2.7 Features with Dependencies on other Products

Some of the new features in Dialog Designer 5.0 require new versions of other products such as Voice Portal.

1. The new CCXML features require Voice Portal 5.0.
2. The IVVR feature requires Voice Portal 5.0.
3. Loquendo support requires Voice Portal 5.0 or IR 3.0 or 4.0 releases.
4. The Variable Administration feature is only applicable to Voice Portal 5.0 release.
5. The Tuning Tools require a new interface in Voice Portal 5.0.
6. GUI support for VRUSM

2.8 MRs included in this Release

id	severity	component	summary
wi00371096	Enhance	CTI Connector	Update JTAPI driver
wi00369516	High	General	HTTPS on Weblogic Does Not Work
wi00364047	High	Runtime Framework	CFD: Event handler can cause subflow return problems: next servlet not provided in exit point map for return node …
wi00366388	Medium	Call Flow Editor	Symbolic Node Reference IDs get set incorrectly sometimes during a copy/paste operation
wi00354434	Medium	Code Generation	Module code generation not generating setNeedsDefaultDisconnectHandler()
wi00368349	Medium	CTI Connector	add a Provider startup delay
wi00358780	Medium	Deployment	Module is reported as red in validation page.

wi00339886	Medium	General	app name is double added to path when using avaya.dd.tempdir
wi00349318	Medium	IVVR	No warning when doing a copy/paste of media items from a VXML 3.0 project to a NON-VXML 3.0 project.
wi00354455	Medium	Runtime Framework	CFD:Exit delay timer can preempt PDC cleanup and exit code generation
wi00366579	Medium	Runtime Framework	On nomatch / noinput leaving a form hidden attribute not reset
wi00362286	Low	Localization Packages	0049.wav in Thai language Bundle plays "9" instead of "19"
wi00365097	Low	Modules	NDM validation in validate.jsp checks for framework version -- not applicable

3 Installation and Upgrades

3.1 New Installations

With the first time (new user) installation of Dialog Designer 5.0, the pre-requisite components such as Eclipse, Tomcat and Java will be included on the media with their own installers for convenience, but the user must install them separately before installing Dialog Designer.

For instructions to install Dialog Designer 5.0, please refer to the documentation by Information Solutions included in the on-line documentation. A stand alone pre-requisites document is available for download prior to installing DD.

3.2 Upgrades – General

There are framework updates in this build that may result in Java compilation errors after upgrading. To correct this, regenerate projects reporting errors.

Upgrades assume the user is starting from Dialog Designer 4.0 or 4.1. If you are using a 3.0 or 3.1 version of Dialog Designer, please upgrade to 4.1 before moving to 5.0.

3.3 Upgrades – Upgrades from VP 5.0 to VP SP 1 – IVVR Application Impact

Using IVVR applications developed with DD 5.0 with VP 5.0 Service Pack 1:

IVVR applications developed with DD 5.0 will not work correctly with VP 5.0 Service Pack 1. When video is used in parallel with audio on VP 5.0, the last video frame would remain on the screen till the user provided some input, and then it would move to the next prompt. This behavior is no longer seen with VP 5.0 Service Pack 1. With VP 5.0 Service Pack 1, the video will go away as soon as its media tag finishes. You can set the duration of the video longer than the actual length of the video to get the last frame of the video to freeze.

Behavior	Modification
White screen between prompts	Use coordination parallel prompt elements to play audio/TTS in parallel with video. For each prompt, make sure that there is a video playing from the beginning of the prompt to the end. If the video is shorter that the audio/TTS set the

	duration attribute for the videos media tag.
White screen between reprompts	Specify small prompt timeout values to minimize the white screen between reprompts and make sure the duration of the video is set long enough.
Synchronization problems with audio/video	Make use of coordination parallel and sequence to get the desired behavior. It is recommended to set the "endsync" attribute for the coordination parallel.
Application not working for a particular screen size	Use different prompts for different screen sizes or use a single prompt and add if/else statements to it.
Non bargable prompts looses video when getting to input state	Insert a silent bargable audio file after the last non-bargable audio prompt and increase the duration of the video.

3.4 Upgrades with CTI

If you are upgrading a 4.x application that uses CTI conferencing (add/remove conference party), you must manually edit the callflow file and explicitly select the active call ID attribute.

3.5 Upgrades from Dialog Designer 4.1 – CCXML

If you are using web services or database access with CCXML 5.0, you need to change the constructor:

```
public getSayings( com.avaya.sce.runtimecommon.SCESession mySession ) {
```

To

```
public getSayings( com.avaya.sce.runtimecommon.IRuntimeSession mySession ) {
```

3.6 Upgrades from Dialog Designer 4.0 – no CCXML

There are no unique upgrade steps when upgrading the development environment from Dialog Designer 4.0 to 5.0 for applications that do not use CCXML. It is recommended that you save backup copies of your 4.0 applications before the upgrade. Also, it is recommended that projects are copied (using Windows Explorer) into the 5.0 workspace, then imported into DD 5.0.

3.7 Upgrades from Dialog Designer 4.0 – CCXML Applications

Dialog Designer 4.1/5.0 (and VP 4.1/5.0) now uses the latest revision of the CCXML specification. The latest version introduces some backward compatibility issues that require changes to your CCXML applications built with 4.0. This section details the changes that impact applications.

DTD Changes:

send	Data attribute changed to name.
dialogprepare & dialogstart	Added parameters attribute.

	Parameters go to the vxml dialog and namelist goes to the application server.
	Review every namelist in dialogprepare and dialogstart. Recommend changing namelist to parameters or adding parameters.
transition	Removed the name attribute, events are referenced as event$
	Change evt to event$
	Remove name=”evt”
	Change (ext) to (event$)

Event Changes:

Removed	error.dialog.wrongstate
	error.merge
Added	connection.merge.fail
	connection.redirect.fail
	connection.accept.fail
	connection.reject.fail

Object Changes:

Dialog Object	Added input, outputs (affects join/union)
	Removed duplex, endpoint
Conference Object	Added conferencename
Connection Object	Added inputs, outputs (affects join/union)
	Removed endpoint, duplex

3.8 Dialog Designer 4.0/4.1/5.0 Runtime

A new wizard (Avaya Speech Development -> Export Runtime Support Files) has been added and will produce the runtimeconfig web application and the runtime support ZIP files for the selected application server. The options to create these files no longer exist on the project export wizard pages.

3.9 Co-residency

Co-residency of differing versions of Dialog Designer applications is not supported with 5.0. A new application server must be setup for Dialog Designer 5.0 applications. Earlier versions or applications must be upgraded by running them through the DD 5.0 development environment and redeploying them on a DD 5.0 application server.

3.10 Additional Application Installation Instructions for Websphere

- Extract the runtimesupportWebsphere.zip files to your classpath. We recommend copying them to your lib\ext directory.

  Example:
  WS_HOME\AppServer\lib\ext

- Once the app is running, to navigate to your index page for your speech application via a browser, you should include index.html within the URL as Websphere does not automatically default to this path.

  Example:
  http://myserver:port/mySpeechApp/index.html

- When deploying DDADMIN (a new component with 4.0) on WebSphere 6.1, you need to manually set jsp compilation to version 1.5 to get it to compile. At WebSphere Integrated Solution Console, do the following to install the DDADMIN ear file – runtimeconfig.ear

1. Application > Install Application. At “Specify the EAR, WAR, JAR, or SAR module to upload and install,” window, Select “Local file System.” Browse to runtimeconfig.ear. Select “Show me all installation options and parameters.” Click Next.
2. At “Choose to generate default bindings and mappings” window, click Next.
3. At “Specifies the resulting security warnings from an analysis of this application” window, click “Continue.”
4. At “Select installation options” window, select “Precompile JavaServer Pages files” and keep the other default values. Click Next.
5. At “Map modules to servers”, select the runtimeconfig.ear. Click Next.
6. At “Provide options to compile JSPs” window, select “runtimeconfig.ear, WEB-INF/web.xml”, write “15” under “JDK Source Level.”
7. At “Provide JSP reloading options for Web modules” window, click Next.
8. At “Map shared libraries” window, click Next.
9. At “Map virtual hosts for Web modules” window, select “runtimeconfig.ear”, click Next.
10. At “Map context roots for Web modules” window, click Next
11. At “Summary” window, click Finish.
12. Make sure to click Save after installation. You can access DDADMIN at *http://myserver:port/runtimeconfig*

3.11 Additional Application Installation Instructions for WebLogic

- Extract the runtimesupportWebLogic.zip files to your classpath. We recommend copying them to your domain server lib directory.

  Example:
  WL_HOME\samples\domains\wl_server\lib

- Using DD export to obtain a war file of your application, such as mySpeechApp.war, explode mySpeechApp.war to a folder which ends with ".war". For example, you can name this folder mySpeechApp.war also. Upload mySpeechApp.war folder to WebLogic, and choose "I will make the deployment accessible from the following location", where the location references the path to the mySpeechApp.war folder.

- Once the application is running, you can navigate to your index page for your speech application via a browser.

  Example:
  http://myserver:port/mySpeechApp

- To deploy your application war file non-exploded (not recommended) see Known Issues item #20.

3.12 Additional Application Installation Instructions for Tomcat

Tomcat 5.x:

- Extract the runtimesupportTomcat.zip files to your classpath. We recommend copying them to the Tomcat common directory.

  Example:
  TOMCAT_HOME\common

- Once the application is running, to navigate to your index page for your speech application via a browser, you can click on your application in the Tomcat manager site.
- A default “log4j.properties” file is placed in the TOMCAT_HOME/common/classes directory. You should examine this file to determine if it meets your logging needs. By default, the logging is set to “INFO” which will generate a lot of log data in the TOMCAT_HOME/logs/Catalina.out file, which may grow quite large. You may consider changing the log level to WARN to reduce the amount of log output.

Example:
log4j.rootCategory=WARN, CONSOLE

Tomcat 6:

- Extract the runtimesupportTomcat6.zip files to your classpath. We recommend copying them to the Tomcat root directory.
  Example:
  TOMCAT_HOME

- Once the application is running, to navigate to your index page for your speech application via a browser, you can click on your application in the Tomcat manager site.
- A default "log4j.properties" file is placed in the TOMCAT_HOME/lib directory. You should examine this file to determine if it meets your logging needs. By default, the logging is set to "INFO" which will generate a lot of log data in the TOMCAT_HOME/logs/Catalina.out file, which may grow quite large. You may consider changing the log level to WARN to reduce the amount of log output.
  Example:
  log4j.rootCategory=WARN, CONSOLE

4 Known Issues

The following section describes any known issues, workarounds, words of caution, etc, and a table of outstanding bug MRs for DD.

4.1 Limitations, Work Arounds, Notes:

CCXML:

1. In simulation, since there is only one input (microphone) and one output (speakers) for multiple call appearances, dialogs must be "serialized" to use the resources. Once a dialog starts a form, it must complete before the next form with a call appearance may use the resources. This generally provides a reasonable simulation of an application, however, in a CCXML application you have multiple call appearances and if one has input, a subsequent call appearance playing an informational message will be blocked until the input is completed and exists. This can occur in the ConferencePredefined application when a conference attendee hangs up. Since the incoming call has a dialog running, listening for an add or drop command, the message "XXX left the conference" will not be heard until the input on the incoming call completes. This does not happen on the platform.
2. When running certain applications (E.g. ConferencePredefined sample application), if the Dialog Designer->Avaya Application Simulator->Enable display of the Avaya Voice Browser output in a console window is enabled, the application can behave irregularly. Please disable this if you encounter this behavior.
3. Applications that use the launchCCXML VP web service with DD/VP 4.0 will not work on VP 4.1 without modification. You now must include a send tag. For placement, one possibility is in the ccxml.loaded transition. The syntax for the addition is:

   ```
   <var name="status"expr="event$.reason"/>
           <send name="'avaya.launchresponse'"
   targettype="'avaya_platform'"target="session.id"namelist="status"/>
   ```

4. If the incoming call classification is "LIVE VOICE" then you need to go to the params tab in the simulation profile and select a new call classification then select "live voice".
5. When working with the Sample Applications, InvokeDatabase and FindMe, there is a limitation when using a MSAccess database and multi-byte data. Multi-byte data will not be stored/retrieved properly. See http://forum.java.sun.com/thread.jspa?threadID=231861&messageID=3968823 for more details, but essentially, the default ODBC-JDBC driver does not handle UTF-8/16 correctly.

6. If there are any leading \r\n in a document before the ?,xml… the parse will fail. The work around in the JSP page is to have:

   <?xml version="1.0" encoding="UTF-8"?>
   <!DOCTYPE ccxml PUBLIC "http://www.w3.org/TR/ccxml" "ccxml.dtd">
   <ccxml xmlns="http://www.w3.org/2002/09/ccxml" version="1.0">

   for the first 3 lines before <%import...

CTI:

7. For CTIC deployment, tsapi.pro needs to be in common/lib on WebLogic.
8. For CTIC deployment, tsapi.pro needs 2 additional properties for talking to AES 4.2 through secure connection. The avayaprca,jks comes with cticonnector.war. You just need to change the path to match your deployment. For example:

   trustStoreLocation=/opt/apps/cticonnector/WEB-INF/lib/avayaprca.jks
   trustStorePassword=password
9. You can now disable the application from gathering call information automatically by changing the flag sage.cti.autoinvoke.callinfo to false in your web descriptor. If you make this change, then you must use a CallInfo item in your data node and set the property: Initial Call to true before doing any CTI Call operations.
10. Starting with the JTAPI 3.1 driver, the DD CTI connector, can use the UCID (Universal Call ID) to keep track of call ID's. To use this feature, you should enable it on your switch (consult your switch documentation). It is not mandatory to enable it, but it will guarantee no call ID's are recycled which have been problematic in the past with high call volumes.
11. When passing data using the UUI, the maximum number of bytes you can use to write is 96. You can use a complex variable as the data for UUI. The properties allow entry of a variable/field in the dial operation.
12. When placing a call to a line on the same switch and it is in the busy state, you may get back a "failed" instead of a "busy" state indication. You should not check specifically for the "busy" state.
13. CTI does not support the "Numbered Services" configuration on the IR platform.
14. You can hear the status of the call if you are doing a blind call or a consult call with transfer on ring set. Note that you cannot hear busy or invalid line (fast busy) because the switch disallows a transfer to a busy line.
15. If you have CTI nodes in your application, you can't run the application in HTML mode. The application will stop when it hits the CTI nodes.
16. The CTI Connector is configured to startup when the application server is started. You can disable this functionality if you do not have the CTI Connector configured yet or if you do not want it to run by editing the web.xml file found in your CTIConnector/WEB-INF directory on your appserver. Look for the value: <load-on-startup>1</load-on-startup> and remove or comment out this tag. **NOTE:** Setting it to 0 does not turn it off. You will need to manually start the CTIConnector if this flag has been removed or commented out.
17. The CTI Connector has a setting in the web.xml called providerstartupdelay (defaults to 4000 ms) that allows you to adjust a maximum delay value (in ms) to allow the provider to come in service. If the provider does not come into service before the timer expires, the connection is lost and will be retried.

Connectors:

18. It is not recommended to use the "Refresh" button on your HTML browser when an application relies on a connector (CTI, IC, etc.). Due to the behavior of some HTML browsers, the refresh action causes the connectors to initialize a second time for the session which is not allowed. The recommended action is to close the HTML browser and then start up a new session.

Design Constraints:

19. The Dialog Designer framework has constraints on the number iterations through call flow nodes that do not generate VXML. Those nodes are: Data, Servlet and Tracking. DD uses a servlet request forward to move between consecutive data nodes that do not generate VXML. This is faster and more efficient than redirecting the HTTP request. However, if you need to iterate more than 100 times on consecutive data nodes, you may run out of stack space. The recommendation if you need to iterate more than 100 times, is to move your iteration into Java code.

Deployment:

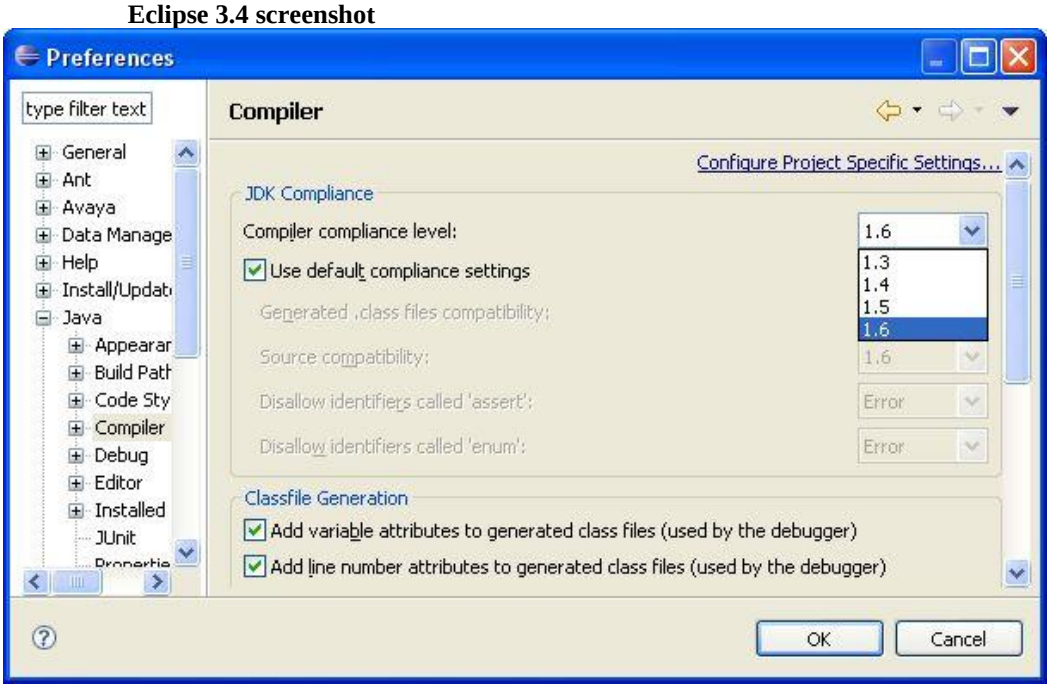
20. To use packed war files you must set the system property avaya.dd.temp.dir:
    -Davaya.dd.tempdir=c:\temp or some other value.

    DD assumes that it can create directories and files in that location. If directory create is disabled, then you must create a subdirectory for each DD web application deployed with sub directories temp and data/log. When recording, temp recording files will be placed in temp and if tracing is enabled the trace.log and report.log will be in data/log. If you have a deployed application named Simple, under c:\temp you would see: c:\temp\Simple\temp and \data\log.

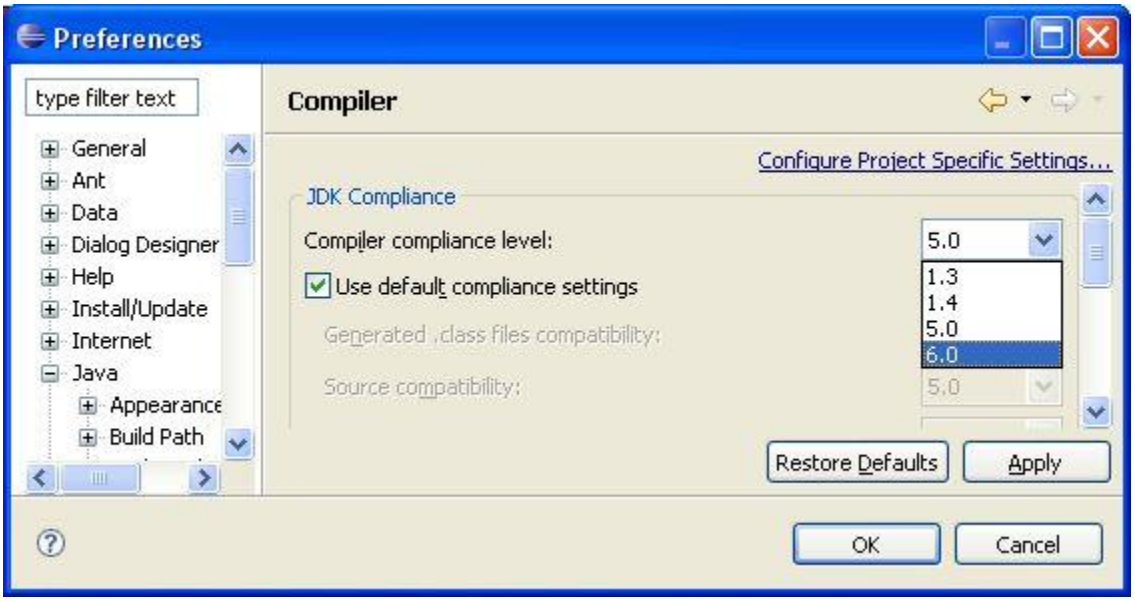
    If the avaya.dd.tempdir is not writable then recording will not work, nor will tracing and local reporting. The avaya.dd.tempdir can also be used with non-packed war files as well, thus providing a mechanism to move the temp and log files out of the web application to a separate area.

21. A DD 5.0 application will fail to run if it is compiled with a Java version that is different from the one on the Application Server. To fix this, select the Java Compiler compliance level that matches what is installed on the Application Server located at Window->Preferences->Java->Compiler in eclipse:

Eclipse 3.4 screenshot



eclipse 3.2 screenshot **NOTE THE DIFFERENCE IN THE DROPDOWN BOXES!**



Failure to match the java version of your target Application Server will result in this error when attempting to run the DD app in HTML mode:

HTTP Status 500 -

---

type Exception report

message

description The server encountered an internal error () that prevented it from fulfilling this request.

exception

```
javax.servlet.ServletException: Error allocating a servlet instance
      org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReport
Valve.java:117)
      org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdap
ter.java:151)
      org.apache.coyote.http11.Http11Processor.process(Http11Process
or.java:870)
      org.apache.coyote.http11.Http11BaseProtocol$Http11ConnectionHa
ndler.processConnection(Http11BaseProtocol.java:665)
      org.apache.tomcat.util.net.PoolTcpEndpoint.processSocket(PoolT
cpEndpoint.java:528)
      org.apache.tomcat.util.net.LeaderFollowerWorkerThread.runIt(Le
aderFollowerWorkerThread.java:81)
      org.apache.tomcat.util.threads.ThreadPool$ControlRunnable.run(
ThreadPool.java:685)
      java.lang.Thread.run(Thread.java:595)
```

root cause

```
java.lang.UnsupportedClassVersionError: Bad version
number in .class file
      java.lang.ClassLoader.defineClass1(Native Method)
      java.lang.ClassLoader.defineClass(ClassLoader.java:620)
      java.security.SecureClassLoader.defineClass(SecureClassLoader.
java:124)
```

```
      org.apache.catalina.loader.WebappClassLoader.findClassInternal
(WebappClassLoader.java:1847)
      org.apache.catalina.loader.WebappClassLoader.findClass(WebappC
lassLoader.java:873)
      org.apache.catalina.loader.WebappClassLoader.loadClass(WebappC
lassLoader.java:1326)
      org.apache.catalina.loader.WebappClassLoader.loadClass(WebappC
lassLoader.java:1205)
      org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReport
Valve.java:117)
      org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdap
ter.java:151)
      org.apache.coyote.http11.Http11Processor.process(Http11Process
or.java:870)
      org.apache.coyote.http11.Http11BaseProtocol$Http11ConnectionHa
ndler.processConnection(Http11BaseProtocol.java:665)
      org.apache.tomcat.util.net.PoolTcpEndpoint.processSocket(PoolT
cpEndpoint.java:528)
      org.apache.tomcat.util.net.LeaderFollowerWorkerThread.runIt(Le
aderFollowerWorkerThread.java:81)
      org.apache.tomcat.util.threads.ThreadPool$ControlRunnable.run(
ThreadPool.java:685)
      java.lang.Thread.run(Thread.java:595)
```

note The full stack trace of the root cause is available in the Apache Tomcat/5.5.23 logs.

---

22. For logging to work with Weblogic on Windows when using packed war files, you need to extract the ddlog4j.properties file from the war file into the target directory. For example if avaya.dd.tempdir is set to "c:\temp" and you are running application "MyApp" then you need to extract the ddlog4j.properties file from the MyApp.war and place it in c:\temp\MyApp\data. This will be resolved in the next service pack and in 5.1.
23. To optimize performance on the Application Server, we recommend adding the following:

    For Windows: Create a system level environment variable:
    JAVA_OPTS = "-server -Xmx1024M -XX:+UseConcMarkSweepGC -XX:+UseParNewGC -XX:ThreadStackSize=512

    ulimit -n 8192

    For Linux: when Tomcat is started as a service, add the flowing to the beginning of your /etc/init.d/tomcat file:
    JAVA_OPTS="-server -Xmx1024M -XX:+UseConcMarkSweepGC -XX:+UseParNewGC -XX:ThreadStackSize=512"

    ulimit -n 8192

    export JAVA_OPTS

    NOTE: The Xmx1024M parameter we try to set to somewhere around half the size of the RAM configured on the box.
    Restart the machine after setting the JAVA_OPTS, and verify that the new setting are in effect by doing an: ps -ef | grep java

ECMA Script Expressions:

24. Single quotes that are part of the value of an ECMAScript expression will generate a semantic error in the Voice Browser if not properly formatted with an escape character (\). For instance:

<var name="Output" expr="'John's return value'"/> incorrect expression

<var name="Output" expr="'John\'s return value'"/> correct expression

For Dialog Designer to correctly generate the VXML output you need to add another escape character when building the expression in your callflow, otherwise Java will interpret it as an escape character itself.
Within your call flow you would create the expression as follows:
John\\'s return value

This will insure the escape character is properly generated in the VXML output. Other characters to watch out for include ('gt;' for '>', '<' for '<', and '&' for '&').

Grammars:

25. Renaming a grammar column is now in the properties view. When the column is selected, the properties view will display the column name and column repeat. This was done to facilitate column repeats, and to better follow the Eclipse paradigm.
26. If you want to use SRGS/SISR with Nuance 9 and DD/VP 5.0, select SRGS-SISR as the grammar format.
27. For external grammar deployment, the application web.xml must be edited to change localhost to the actual ip address of the application server. This can be done before or after deployment. As an alternative, you can use relative urls for external grammars MR 5381, or store the host and port in a DD variable and reference the DD varaible in the url
%DDVAR_myextgrammarsloc%/Myapp/somedir/agrammar.grxml MR 5312
28. Built in speech grammars are provided by the ASR vendor. While most of them say they are supported, the specific grammars that are supported in any given language and the quality of the results when using them can vary. You should consult the speech vendor for specifics on the support for the languages you would like to use.
29. The quality of the Japanese and Chinese ASR provided by Microsoft (used in DD simulation) for built in grammars is not very high. When testing built-in grammars using Japanese or Chinese, it is recommended to use the “Send ASR” feature.
30. The following illustrates how the runtime can be used to generate large external grammars. The example is simple, but can be extended to build the grammar from an external data source. The reason to do this is for performance at runtime.

```
import com.avaya.sce.runtime.DynamicGrammar;
import com.avaya.sce.runtime.GrammarItem;
import com.avaya.sce.runtime.GrammarRule;
import com.avaya.sce.runtimecommon.SCESession;

public class MyGrammar extends DynamicGrammar {

  public MyGrammar() {
  }

  public void buildGrammar(SCESession mySession) {
    GrammarRule rule;

    rule = new GrammarRule();
    rule.add(new GrammarItem("dog", "DOGTAG"));
    rule.add(new GrammarItem("cat", "CATTAG"));
    add(rule);
    rule = new GrammarRule();
    rule.add(new GrammarItem("blue", "BLUETAG"));
    rule.add(new GrammarItem("red", "REDTAG"));
```

```
            add(rule);
        }

        public static void main(String[] args) {
  try {
      MyGrammarg = new MyGrammar();
      g.buildGrammar(null);
     String result = g.toString("en-us", SCERT.ASR_TYPES_MAP.get(SCERT.ASR_DISPLAY_LOGUENDO),
SCERT.VOICE_MODE);
            XmlWriter writer = new XmlWriter();
            Document doc = XML.getDocumentFromXMLString(result);
            writer.write(doc, System.out);
    } catch (Exception e) {
            e.printStackTrace();
    }
  }
```

Help:

31. If you encounter the error "Runtime error" when using the on-line help, you may need to change you Internet Explorer settings to work properly with Sun's JavaDoc utility. While in your IE browser, go to Preferences, Tools, Internet Options. Select the Advanced tab. Make sure that "Disable Script Debugging (Internet Explorer)" and "Disable Script Debugging (other)" are checked. Click Ok. Close all IE windows and Eclipse. Then restart Eclipse.

IBM Grammars:

32. On the IR platform only, dynamic grammars with IBM ASR do not work properly. This is IR platform bug which requires a patch to the IR platform.

33. External Grammars get cached for 24 hours on the WVS. If you make a change to an external grammar, it will not take effect until the next day. This default can be modified as follows:

- Browse to /opt/WebSphere/AppServer/bin
- Run wsadmin.sh
- After that run the following set of commands that display and set com.ibm.voice.server.rr.fetchexpires (which determines the cache expiration time in milsec) to desired value:

set tmpCfg [$AdminControl queryNames *:*,type=WVSAdminConfig] (will retrieve system name and information)
set cachetime [$AdminControl invoke $tmpCfg get com.ibm.voice.server.rr.fetchexpires] (will retrieve the current setting for cache expiration period)
$AdminControl invoke $tmpCfg put {com.ibm.voice.server.rr.fetchexpires <cache expiration in msec}

IC:

34. You cannot run an ICC enabled module in the simulator standalone as it will not invoke the ICC newcall operation. You must invoke it using a parent application that is also IC enabled.
35. If you implement a prompt to play while transferring a call to IC, the transfer can sometimes occur faster than the prompt is played. The flush prompt feature should be used for this scenario.
36. Some variables require a type. Specifically, variables used in a web operation might have to be converted to a specific type (int, long, etc). Fields on the VDU variable and VDU cache variable do not perform this conversion as they deal strictly with Strings. If you need to use a VDU field within your web operation that requires a type to be set, you should copy the VDU (or VDU cache) field into another local variable that does perform the conversion.
37. If you are running an IC-enabled application in the simulator, you must check to make sure there is a VOX or VRUSM/HTTPVox simulator configured in Window……IC Simulator.

IVVR:

38. Although Voice Portal will play back 4 channel png files from your media page, the Ambulant player does not support them and will give you an error if you attempt to play them. You must use 3 channel png files in simulation
39. Some users might see this message from Ambulant, due to the preferences setting.



The solution is to change the setting, when the Ambulant player comes up, goto, file->preferences-> Media tab, and make sure you disable the checkbox for "Use ffmpeg". And replay the app in the simulator.

40. When using the simulator with IVVR applications, you need to press the stop button before DD continues with callflow simulation, then the start button to get to the next video/text.
41. If you have media items that have no intrinsic duration (text and images) you need to make sure they have an explicit duration set on their media properties. This is especially important during simulation which can cause unexpected results in the Ambulant Player and sometimes crashes.
42. If you are combining media items under a SMIL PAR element, pay attention to the end sync property and the durations (either implicit or explicit) of each media item to get your expected results. Likewise this applies to the VXML PAR when combining media pages with speech/phrases.
43. When using video images in parallel with other items, set endsync to “first” or specify a duration for the video image, otherwise the application may hang as it waits indefinitely for the image to stop playing.

Localization Packages/Certificate:

44. The certificates used to sign DD 5.0 were updated because they expired in August; consequently, localization packages that are installed with the previous certificate will produce warnings. New bundles with the current certificate will be made available with the GA of DD 5.0 and with subsequent service packs for 3.1, 4.0 and 4.1.

Memory Issues:

45. With the addition of the Eclipse WTP (Web Tool Project), memory issues have become more common. More information about the issues and workarounds can be found at the following links:

    Links that describe the issue:

    https://bugs.eclipse.org/bugs/show_bug.cgi?id=129490
    https://bugs.eclipse.org/bugs/show_bug.cgi?id=92250

    Similar threads on the Eclipse newsgroups:
    http://www.eclipse.org/search/search.cgi?cmd=Search!&form=extended&np=1&q=PermGen&tmptag=5&ul=/newslists/&wf=574a74

    Possible workaround:
    http://tassos.blogentis.net/2006/06/08/eclipse-and-permgen-space

Eclipse 3.4 appears to be even more memory intensive. Setting the permgen space to be 1024M seems to be a more stable setting.

Multi-Lingual:

46. Node (templates), Audio file names, grammar names, project variables, folders and project names can not contain double byte or extended ascii characters (E.g. Mandarin/Korean). This is because DD uses these names in urls/uris which must be composed of a subset of US ASCII characters. DD prevents the entry of unsupported characters.

OSDMs:

47. Pre OSDM 2.0.6 versions will not work with the new Nuance 9.0 (Quantum release). The new OSDM versions have not as yet been validated with DD 4.0.
48. When using OSDMs with WebLogic 9.2, you will need to use the web.xml file to fix CT NSRD00019567 on OSDM-Core 2.0.4.2006062211. The patch should be obtained from the Nuance support site. To install the web.xml file on each WebLogic 9.2 platform:

    a. Copy your original osdm2-core.war (version 2.0.4) to osdm2-core.zip (for instance, in a shell window, type the following command: "cp osdm2-core.war osdm2-core.zip")
    b. Extract osdm2-core.zip into a folder named osdm2-core
    c. Extract the OSDM_Core_2.0.4_TS39109_CT19567_Patch.zip file to the osdm2-core folder. The file WEB-INF/web.xml will be updated.
    d. Log on to the WebLogic Server Administration console.
    e. From the "Domain Structure", click on "Deployments"
    f. Check that no osdm2-core application is already deployed, otherwise delete it.
    g. From the "Change Center", click on "Lock & Edit"
    h. The "Install" button from the "Deployments" tab should now be activated and selectable
    i. Click on "Install" in order to start deploying osdm2-core webapp
    j. Select the location of the osdm2-core folder just created in step 2 and click "Next"
    k. Select option "Install this deployment as an application" and click "Next"
    l. Click on "Finish" to finalize the deployment
    m. From the "Change Center", click on "Activate Changes"
    n. You should now see that the osdm2-core web application has been deployed and is in the "Prepared" state
    o. Check the box to the left of the title osdm2-core, click "Start" and click "Yes" to confirm. The osdm2-core state should now show "Active".

    If you want your OSDM to work with OSR 3.0, you need to edit the web.xml file to substitute osr 2.0 for osr 3.0 in the OSDM war (e.g. osdm2-core.war) to get the application to run successfully.
49. When using OSDMs with WebSphere 6.1, bad fetch errors occur with the core. There is a work around required. From Nuance:

    1) Log on to the WebSphere6 Administrative console
    2) Expand the applications option and click on "Enterprise Applications"
    3) You should be able to see the osdm2-core application already installed. Select the check box to the left of the title osdm2-core and then click "Stop" in the header menu. The screen should refresh and you should now see a red 'x' next to it.
    4) Log off from the WebSphere6 Administrative console
    5) Using a windows explorer, go to where the osdm2-core web application has been "installed" under the WebSphere6 application server folder such as the following path:

    'IBM\WebSphere\AppServer\profiles\<AppServerName>\installedApps\
    <MachineNodeCellName>\osdm2-core.ear\osdm2-core.war'

    6) Under this folder open/edit the following file: WEB-INF/web.xml

7) In the web.xml file, search for the following parameter name:

« com.speechworks.osdm.cookieVersion » (with no guillemets)

8) Change the value of this parameter (see tag <param-value>) to the following value:

« netscape » (with no guillemets)

9) Save and close the file
10) Using a windows explorer, go to where the osdm2-core web application has been "deployed" under the WebSphere6 application server folder such as the following path:

'IBM\WebSphere\AppServer\profiles\<AppServerName>\config\cells\ <MachineNodeCellName>\applications\osdm2-core.ear\deployments\osdm2-core\osdm2-core.war'

11) Under this folder open/edit the following file: WEB-INF/web.xml
12) In the web.xml file, search for the following parameter name:

« com.speechworks.osdm.cookieVersion » (with no guillemets)

13) Change the value of this parameter (see tag <param-value>) to the following value:

« netscape » (with no guillemets)

14) Save and close the file
15) Log on to the WebSphere6 Administrative console
16) Expand the applications option and click on "Enterprise Applications"
17) Check box to the left of the title osdm2-core and then click "Start" in the header menu. The screen should refresh and you should now see a green 'x' next to it.
18) Try the OSDM example "osdm2-core/en.us/examples/menu.jsp" and check that no more bad-fetch errors are reported in the VXML browser when fetching a page from OSDM core 2.x.

Simulation:

50. When simulating using the VRUSM simulator, it is recommended that you use the DD Runtime configuration servlet and set the ping timeout = 0 secs. This will effectively disable pings which aren't necessary for simulation. A future release/service pack will allow you to do this automatically via the simulation page in DD.
51. When using the VRUSM simulator, make sure the ICC is configured to talk to the VRUSM and not the HTTPVox.
52. If you are experiencing slowness in simulation or unexpected delays, modify the garbage collection settings by adding either:

    -XX:+UseConcMarkSweepGC –XX:+UseParNewGC or –Xincgc to your eclipse startup.

    Example: Create a shortcut to eclipse and in the properties target:

    C:/SageRel-4.1\eclipse\eclipse.exe –vmargs –XX:+UseConcMarkSweepGC –XX:+UseParNewGC

    Additionally, add 2 entries in the Dialog Designer->Speech->Simulators table, one for vox->ic and one for ic->vox.

SSL:

53. When using SSL and Scansoft/Nuance ASR, grammars should be built as dynamic, even if they don't need to be, so they will be processed, in-line. The ASR engine has a problem processing regular grammars if they exceed 4 elements.

Tomcat:

54. Tomcat will reload the context of your speech application when it detects a change. This will occur pretty much anytime you change your project and save it while Tomcat creates a new current one for your sevlet and reloads the dependencies. DD does not have a way to indicate to Tomcat that a submodule is a dependency of its parent application. Therefore, when the parent's context is loaded into the new class loader there is a conflict with the class loader that was used to load the submodule. The solution is two fold:
    a. If your context file points to your speech application, you can simply reload the context of the submodule right in DD using the Tomcat context option "Reload this context."
    b. If you have deployed your submodule into Tomcat and Tomcat is running it out of the webapps directory, the simplest solution is to restart Tomcat or you can use the Tomcat Manager application. Refer to the Tomcat documentation for more information on how to do this using the Tomcat manager.
55. The console window may display warning messages about invalid log4j configuration when starting Tomcat from Eclipse. To fix this add a log4j.properties file to the Tomcat classpath (i.e. to <tomcat home>/common/classes). The properties file should contain these lines:
    log4j.rootLogger=info, stdout
    log4j.appender.stdout=org.apache.log4j.ConsoleAppender
    log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
    log4j.appender.stdout.layout.ConversionPattern=%d{dd/MM/yyyy HH:mm:ss} %5p - %m%n

Transfers:

56. Bridged transfers require CM version 3.0.1 or above, with an H323 configurations on IR due to a Configuration Manager defect in earlier releases.

TTS:

57. WVS 5.1.3 GA version of IBM TTS has a bug in it that impacts the sound quality of TTS prompts. A fix has been posted to the IBM support site: PMR# 05274,756,000.
58. Using symbols in TTS (i.e. &, <, >, etc.) in a prompt may result in runtime errors. These symbols generate invalid VXML code. **Workaround**:
    a. Spell out the word. For example: '&'→"and", '<'→"less than", etc.

Validate Menu Option:

59. Running the validate menu option on any DD 5.0 project causes a Warning to show up in the Problems view. The Resource is web.xml and the Location is line 8 which is: -><! DOCTYPE web-app PUBLIC"-//Sun Microsystems, Inc//DTD Web Application 2.3//EN" http://java.sun.com/dtd/web-app 2 3.dtd> There is currently no work-around to get rid of this warning once it occurs.

WebLogic:

60. Make sure the design environment is set to Java 1.5 when deploying to Weblogic because it does not yet support Java 1.6.

Web Services:

61. The graphical Web Services tool is intended for use with simpler web services. I.e. Web services that return base java types (string, int). If your Web service returns an object that has nested objects, at a minimum, you will need to select "Use Java Obj" on the Web Service wizard. This will instruct the runtime code to take the returned value and place it into the Dialog Designer variable as an Object. You will need to write some Java code to examine your returned Object and store values into Dialog Designer variables. This is necessary as Dialog Designer only supports a simple variable system of either a variable or a grouping of variables.

62. When using Web Services, operation name overloading in a wsdl:portType is disallowed by the Profile. A wsdl:portType in a description must have operations with distinct values for their name attributes. Note that this requirement applies only to the wsdl:operations within a given wsdl_portType. A wsdl:portType may have wsdl:operations with names that are the same as those found in other wsdl:portTypes.

WebSphere Application Server:

63. If the application executes the number of database operations that exceeds the number of the connection pool specified for the data source, the trace log will say javax.servlet.ServletException: Connection not available, Timed out waiting for 180001. The database operation will hang for a while before the error is thrown. To fix this, the configuration should be changed as follows:
    - Add the following node to the web.xml of the application
        - <resource-ref>
        - <!— replace DS with the real data source name in DD ->
            - <res-ref-name>jdbc/DS</res-ref-name>
            - <res-type>javax.sql.DataSource</res-type>
            - <res-auth>Container</res-auth>
            - <res-sharing-scope>Unshareable</res-sharing-scope>
        - </resource-ref>
    - Export the app for deployment; make sure this is done after the web.xml has been changed. Otherwise, you have to re-deploy the app.
    - In the Websphere admin console, when you install the application, there will be a new step call Map resource references to resources for mapping the data source name in web.xml to the data source you normally define in Websphere admin console.
    - On the screen for this step, all the way at the bottom, find the Browser button, click on it, and select the data source that you've defined for the intended database.
    - Finish the deployment.

64. Make sure the design environment is set to Java 1.5 if deploying to Websphere because they do not yet support Java 1.6.
65. See the Database section for more information on database configuration for WAS.
66. If you have trouble deploying the Tsapi.pro file or WAS install scripts on a Linux machine, try using the utility dos2unix on the files.
67. Some database drivers work fine in Dialog Designer when designing the speech application, but do not work in some versions of Websphere. In that case, replace the driver in Websphere with an updated or different version depending on your database type and configure the settings appropriately referring to the WebSphere documentation.

68. For the Help Topic: Installing project files on WebSphere servers, Item 4, should read:

    Set this variable to point to your …\Websphere\appserver**bin** directory.

WebLM:

69. If you are using a Solaris machine for your WebLM server, you must use WebLM version 4.1, or later or the host id provided in the RFA generated license will not work.

5 Version Information

To view the Dialog Designer version in the product, select Help, then About Eclipse, select Feature Details and select the Dialog Designer entry.

6 Hardware Prerequisites

- CPU speed 1GHz (min) 2GHz (recommended)
- RAM 512 MB (min) 1GB (recommended)
- Hard Disk drive 40 GB (min)

7 Software Prerequisites (Supported Versions)

7.1 Design Environment

- Windows XP
- Vista Ultimate
- Vista Business
- Eclipse 3.3 or 3.4
- GEF 3.3 or 3.4
- Tomcat version 6.0 or 5.5, **Note**: **Design & Runtime must use the same version of Tomcat or there will be issues with database connections.**
- J2SE version 1.6.x or J2SE version 1.5.x
- Microsoft Speech SDK 5.1 or 6.0
- Apache Axis 1.4
- DTP 1.5 (with Eclipse 3.3) or DTP 1.6 (Eclipse 3.4)
- EMF-SDO-XSD 2.3 (with Eclipse 3.3) or EMF-SDO-XSD 2.4 (with Eclipse 3.4)
- WTP 2.0 (with Eclipse 3.3) or WTP 4.0 (with Eclipse 3.4).

7.2 Runtime Environment

Not all of the components are required for the runtime (E.g. the application server will be either Tomcat; WebSphere or Oracle WebLogic, the OS will be Windows, Solaris or Linux; the platform will be either IR or VP, etc.). Supported versions are listed, but the vendors supported list of dependencies and combinations should be consulted when setting up the environment). The tested configurations are as listed in the requirements document.

- Windows Server 2003 standard
- Linux ES 4.0
- Tomcat 6.0 or 5.5.x , **Note: The design and runtime version of Tomcat must match or there will be issues with database connections.**
- J2SE version 1.6.x or J2SE version 1.5.x
- Apache Axis 1.4
- Nuance OSR 3.0.10
- Nuance (aka Quantum) 9.0
- Loquendo 7.5.35
- IR 3.0 or 4.0
- Voice Portal 4.0, Voice Portal 4.1 or Voice Portal 5.0
- WebLM Version 4.5
- Websphere 6.1 application server
- Websphere Voice Server version 5.1.3
- Oracle Weblogic 10 (preferred), 9.1, 9.2 is required if using OSDMs
- Sun Solaris 10 on SPARC
- CM version R013x01.2.639.0 or greater, and must have the green feature 7434ND enabled and licensed is required to use the Consultation transfer feature.