



Application Notes for CGI Computer Assisted Collection System Enterprise Dialer Remote Control with Avaya Predictive Dialing System - Issue 1.0

Abstract

These Application Notes describe the configuration steps required for CGI Computer Assisted Collection System Enterprise (CACS) Dialer Remote Control to successfully interoperate with Avaya Predictive Dialing System. The Dialer Remote Control (DRC) component of CACS 8.0 is a browser-based agent application that uses the Agent API of Avaya Predictive Dialing System 12.0 (PDS12). Information in these Application Notes has been obtained through interoperability compliance testing and additional technical discussions. Testing was conducted via the Developer*Connection* Program at the Avaya Solution and Interoperability Test Lab.

1. Introduction

These Application Notes describe the compliance-tested configuration utilizing Avaya Predictive Dialing System 12.0 (PDS12) and CGI CACS 8.0 Dialer Remote Control. The CACS 8.0 application is an online system that supports the collection of delinquent and over limit accounts. The Dialer Remote Control (DRC) is a component of CACS 8.0 which interacts with PDS12 via an ActiveX control. The DRC allows agents to handle outbound and inbound calls from PDS12. The DRC uses the following feature buttons to invoke functions on PDS12.

- Logon/Logoff
- Go available/Go unavailable
- Terminate Call
- Transfer/Consult/Conference Call
- Release Call
- Callback (Recall)
- Manual Dial
- Place Managed Call
- Play Message

The following diagram illustrates the Dialer Remote Control:

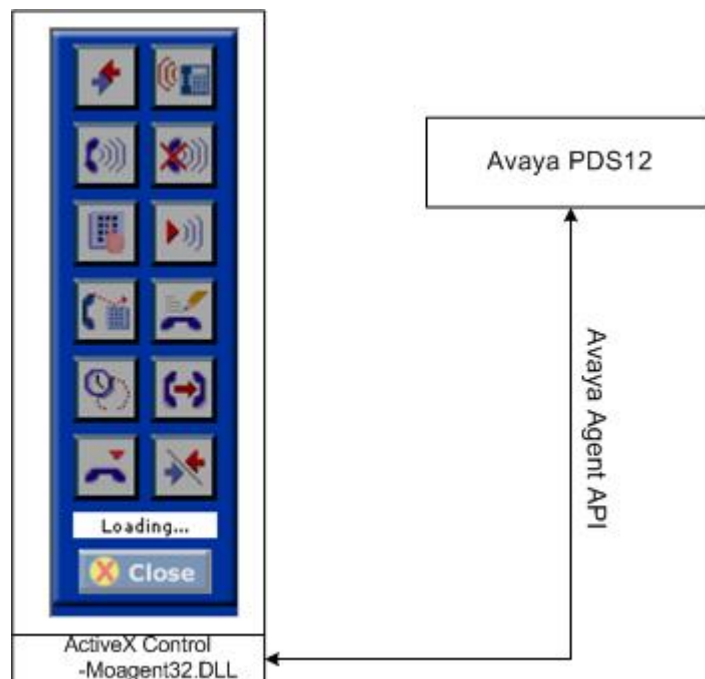


Figure 1: Dialer Remote Control

1.1. Integration Overview

Figure 2 depicts an overview of the CGI CACS 8.0 – Dialer Remote Control integration to Avaya Predictive Dialing System 12.0. The Dialer Remote Control uses Microsoft Visual Basic

ActiveX Control to interface to PDS12's Avaya Agent API DLL, Moagent32.dll. The DialerProperties.xml file on the CGI web server is used to configure the DRC. The configuration parameters on the DialerProperties.xml file are passed to the ActiveX Control on the agent machine and are used to establish a connection with PDS12.

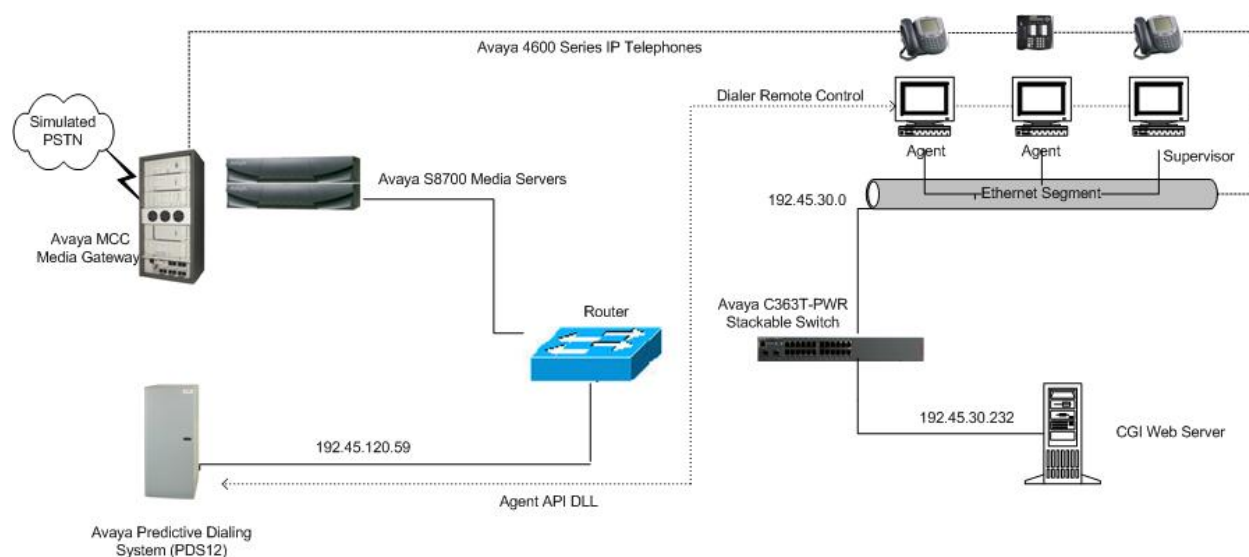


Figure 2: Avaya PDS12 and CGI CACS 8.0 - DRC Configuration

2. Equipment and Software Validated

The following equipment and software were used for the test configuration provided:

Equipment	Software
Avaya Predictive Dialing System	12.0 SP4
Avaya Agent API – Moagent32.dll	3.0.0.406
Avaya S8700 Media Servers	Avaya Communication Manager 3.0.1 (R013x.00.1.346.0)
Avaya MCC1 Media Gateway	
TN2312BP IP Server Interface	HW03 FW012
TN799DP C-LAN Interface	HW01 FW015
TN2302AP IP Media Processor	HW13 FW095
Avaya 4610 IP Telephones	2.1.3 (4610SW)
Avaya C363T-PWR Converged Stackable Switch	4.3.12
CGI Web Server	IBM Websphere 5.1
Agent PC for CGI Dialer Remote Control	<ul style="list-style-type: none"> • CACS 8.0.4 • Microsoft Windows 2003 • Microsoft Internet Explorer 6.0

3. Configure Avaya Predictive Dialing System 12.0

These Application Notes assume that Avaya Predictive Dialing System 12.0 is configured and operational for outbound, managed, inbound and blended jobs. The Intelligent Call Blending feature of PDS12 is used for blended jobs. The following features should have already been configured on PDS12.

- Intelligent Call Blending
- Completion Codes
- Agent Owned Recall
- Recall/Callback
- Agent Playable Message
- Autowrap
- Job Linking
- Native Voice and Data Transfer

The PDS12 configuration files for the features above are used to configure the DialerProperties.xml file located on the CGI web server. The sections below highlight the fields in the PDS12 files that are needed to configure the DialerProperties.xml file.

3.1. Completion Codes Configuration Files

Avaya Predictive Dialing System 12.0 uses completion codes to identify the result of a phone call with a customer. The completion code configuration file, compcode.cfg, is located in the /opt/avaya/pds/config folder on the PDS12 server. The configuration code numbers will be used in Section 4.3 to configure the DialerProperties.xml file.

```
0:NOTCALLED:NO::Record not yet called:SYS:
2:ERROR:YES::Invalid number:SYS:
3:TIMEOUT:YES::No dial tone:SYS:
4:HANG_PORT:YES::Line idle after dial:SYS:
5:NOTINZONE:NO::Outside calling hours:SYS:
11:BUSY:YES::Busy signal:SYS:
12:CONTTONE:YES::FAX or MODEM:SYS:
13:AUTOVOICE:YES::Answering machine:SYS:
14:VOICE:NO::Person on the line:SYS:
15:NOANSWER:YES::No answer:SYS:
16:RINGING:YES::Ringing phone:OP:
17:CUSTHU:YES::Cust hung up in queue:OP:
18:TRANSFER:YES::Transfer release:OP:
19:RECALL:NO::Recall release:OP:
20:CODE20:YES::Code 20:OP:
21:CODE21:NO::Code 21:OP:
22:CODE22:NO::Code 22:OP:
```

Figure 3: compcode.cfg file

3.2. Agent Owned Recall Configuration files

Agent Owned Recall allows an agent on an outbound call to set a specific recall time for a customer record. This will mark the record as an “*agent owned recall*” record. PDS12 will then make the recall when the specified time occurs and pass the call to the agent that owns the recall.

The completion code for the Agent Owned Recall feature is in the configuration file, compcode.cfg, located in the /opt/avaya/pds/config folder on the PDS12 server. The configuration code number for the AORECALL completion code will be used in Section 4.1 to configure the DialerProperties.xml file.

```
84:CODE84:NO::Code 84:OP:
85:CODE85:NO::Code 85:OP:
86:CODE86:NO::Code 86:OP:
87:CODE87:NO::Code 87:OP:
88:CODE88:NO::Code 88:OP:
89:MANAGEDA:NO::Managed non-connect A:OP:
90:MANAGEDB:NO::Managed non-connect B:OP:
91:VIRTVOICE:NO::Virtual msg to VOICE:OP:
92:VIRTAUTOV:NO::Virtual msg to AUTOV:OP:
98:AORECALL:YES::Agent Owned Recall:OP:
```

Figure 4: compcode.cfg file

3.3. Recall/Callback Configuration files

Recall/Callback feature allows an agent on an outbound call to set a specific recall time for a customer record to be recalled. PDS12 will then make the recall when the specified time occurs and passes the call to any available agent. The completion code for the Recall feature is in the configuration file, compcode.cfg, located in the /opt/avaya/pds/config folder on the PDS12 server. The configuration code number for the RECALL completion code will be used in Section 4.1 to configure the DialerProperties.xml file.

```
13:AUTOVOICE:YES::Answering machine:SYS:
14:VOICE:NO::Person on the line:SYS:
15:NOANSWER:YES::No answer:SYS:
16:RINGING:YES::Ringing phone:OP:
17:CUSTHU:YES::Cust hung up in queue:OP:
18:TRANSFER:NO::Transfer release:OP:
19:RECALL:NO::Recall release:OP:
20:CODE20:YES::Code 20:OP:
21:CODE21:NO::Code 21:OP:
```

Figure 5: compcode.cfg file

3.4. Agent Playable Message Configuration files

The Agent Playable Message feature allows an agent to play a message to a customer at the end of the call. The message to play and the completion code number used to terminate the call are defined in the ag_cmd1.ky file located in the /opt/avaya/pds/config folder on the PDS12 server. The message file, voicemsg.cfg, is located in the /opt/avaya/pds/config folder on the PDS12 server. The completion code for the Agent Playable Message feature is in the configuration file, compcode.cfg, located in the /opt/avaya/pds/config folder on the PDS server. The message number of “19” and the completion code number of “20” will be used in Section 4.5 to illustrate an example of how to configure the DialerProperties.xml file.

```

KEY: ^A^M:::F1
KEY: ^AA^M:LOGOUT:::F2 Logout of job
KEY: ^AB^M:DIAL:::F3 Dial previewed record
KEY: ^AC^M:CANCEL:cancel_call:35:F4 Managed cancel call
KEY: ^AD^M:RELEASE:call_complete:89:F5 Managed non-connection
KEY: ^AE^M:RELEASE:pf_msg_1:20:F6 Play pre-recorded msg
KEY: ^AF^M:HOME:::F7 Goto first editable field
KEY: ^AG^M:RELEASE:call_complete:21:F8

```

Figure 6: ag_cmd1.ky

```

1:fwait1
2:fwait2
3:fwait3
4:fwait4
5:infwait1
6:infwait2
7:infwait3
8:infwait4
9:mwait1
10:mwait2
11:mwait3
12:mwait4
13:inmwait1
14:inmwait2
15:inmwait3
16:inmwait4
#17:fvirt1
#18:mvirt1
19:mwait2 (pf_msg_1)
#20:pf_msg_2
#21:pf_msg_3
#22:pf_msg_4
#23:pf_msg_5
#24:pf_msg_6

```

Figure 7: voicemsg.cfg file

```

0:NOTCALLED:NO::Record not yet called:SYS:
2:ERROR:YES::Invalid number:SYS:
3:TIMEOUT:YES::No dial tone:SYS:
4:HANG_PORT:YES::Line idle after dial:SYS:
5:NOTINZONE:NO::Outside calling hours:SYS:
11:BUSY:YES::Busy signal:SYS:
12:CONTONE:YES::FAX or MODEM:SYS:
13:AUTOVOICE:YES::Answering machine:SYS:
14:VOICE:NO::Person on the line:SYS:
15:NOANSWER:YES::No answer:SYS:
16:RINGING:YES::Ringing phone:OP:
17:CUSTHU:YES::Cust hung up in queue:OP:
18:TRANSFER:YES::Transfer release:OP:
19:RECALL:NO::Recall release:OP:
20:CODE20:YES::Code 20:OP:
21:CODE21:NO::Code 21:OP:
22:CODE22:NO::Code 22:OP:P:

```

Figure 8: comcode.cfg file

3.5. Auto Release/Wrap Configuration files

The Autowrap feature automatically puts an agent into update mode when a customer hangs up. When the hang-up is detected, an AutoRelease message is sent to the agent. This is configured in the configuration files, *.job, located in the /opt/avaya/pds/job folder on the PDS12 server. The **AUTORELEASE** field is set to “YES”.

```
ANI::
AUTOEND:NO:
AUTORELEASE:YES:
DATAPROCESS:generic:
DATASCRIP:alljobs:
DISABLE_SHADOWJOB:NO:
DISPLAY:YES:
DNC_GROUP:DNC:
DONE_TME:5=50,10=20,15=20,20=10:
EDTFILE:outbound:
EXPIRED_RECALL:NO:
HITRATE:100:
```

Figure 9: outbnd.job file

3.6. Calling List Fields Configuration files

The calling list contains the customer records. The calling list fields are configured in the configuration files, *.fdict, located in the /opt/avaya/pds/lists folder on the PDS12 server. The calling list fields will be used in Section 4.7 to configure the DialerProperties.xml file.

```
RECLN:976:
ACCTNUM:16:C:ACCOUNT NUMBER:
BALANCE:10$:BALANCE:
TOTALDUE:10$:TOTAL DUE:
NAME1:25:C:NAME LINE1:
NAME2:25:C:NAME LINE2:
CITY:25:C:City:
STATE:2:C:State:
ZIPCODE:5:N:ZIPCODE:
PHONE1:12:C:HOME PHONE:
PHONE2:12:C:BUSINESS PHONE:
COMMENT1:60:C:COMMENT LINE 1:
AGENT:8:C:AGENT ID:
DTE:10:D:SYSTEM DATE:
TME:8:T:SYSTEM TIME:
CODE:2:C:COMPLETION CODE:
JOBNAME:20:C:Job Name:
COUNTER:3:N:RECORD ATTEMPT COUNTER:
ENTRYDATE:10:D:1ST DATE ON SYSTEM:
STATUSFLAG:1:C:RECORD STATUS:
```

Figure 10: list1.fdict file

4. Configure CGI Web Server

The CGI web server contains the configuration file, `DialerProperties.xml`, which is specific for the Dialer Remote Control. It is used to configure the dialer interface, to configure various settings for the dialer dialogs, and to provide mappings for the dialer call record. The following parameters need to be configured on the `DialerProperties.xml` file for the DRC to function with PDS12.

- Configuration Parameters
- Logon Dialog Parameters
- Terminate Call Dialog Parameters
- Recall Dialog Parameters
- Play Message Dialog Parameters
- Transfer Call Dialog Parameters
- Field Map Parameters

4.1. Configuration Parameters

The Configuration parameters are used to establish a connection with PDS12 and to pre-define some of the termination codes. The following parameters need to be defined:

- **DialerServerIP** – set to the IP address of PDS12.
- **ServerPortNumber** – set to “22700”.
- **HeadsetVolume** – set to “5”.

The following parameters are optional. If a value is provided, the call will be automatically terminated using the administered completion codes.

- **RecallRelease** – set to the completion code number for **RECALL** from Section 3.3.
- **AORRelease** – set to the completion code number for **AORECALL** from Section 3.2.
- **TransferRelease** – set to the completion code number for **TRANSFER** from Section 3.1

The remaining parameters can retain the default values.

```
<<ConfigParms>
= <ConfigParm>
  <DialerServerIP>192.45.120.59</DialerServerIP>
</ConfigParm>
= <ConfigParm>
  <ServerPortNumber>22700</ServerPortNumber>
</ConfigParm>
= <ConfigParm>
  <HeadsetVolume>5</HeadsetVolume>
</ConfigParm>
= <ConfigParm>
  <KeyField>ACCOUNT_NUMBER</KeyField>
</ConfigParm>
= <ConfigParm>
  <ConnectWithLogon>Y</ConnectWithLogon>
```



```

    </ConfigParm>
  - <ConfigParm>
    <RecallRelease>19</RecallRelease>
  - <ConfigParm>
    <AORRelease>98</AORRelease>
  - <ConfigParm>
    <TransferRelease>18</TransferRelease>
  - <ConfigParm>
    <AvayaLogging>N</AvayaLogging>
  - <ConfigParm>
    <DebugLogging>Y</DebugLogging>
  </ConfigParms>

```

Figure 11: Configuration Parameters in DialerProperties.xml

4.2. Logon Dialog Parameters

The Logon Dialog parameters are used to define the logon dialog that the agent will use. This provides the userid, password, extension and agent type fields that will be available to the agent when logging into PDS12. The following parameters need to be defined.

- **Dialog Prefix** – set to “AVAYA_MULTI”.

The agents can log in to PDS12 as outbound, inbound, managed or blended types of agents. The TypeCode and TypeName parameters should be defined based on the agent type supported by the DRC.

To allow agents to log in as outbound agents, set the following parameters.

- **TypeCode** – set to “O”. This will allow agents to log in as outbound agents to handle outbound calls.
- **TypeName** – set to “Outbound”.

To allow agents to log in as inbound agents, set the following parameters.

- **TypeCode** – set to “I”. This will allow agents to log in as inbound agents to handle inbound calls.
- **TypeName** – set to “Inbound”.

To allow agents to login as blended agents, set the following parameters.

- **TypeCode** – set to “B”. This will allow agents to log in as blended agents to handle inbound and outbound calls
 - **Note** – CGI currently only supports Intelligent Call Blending on PDS12.
- **TypeName** – set to “Blended”.

To allow agents to log in as managed agents, set the following parameters.

- **TypeCode** – set to “M”. This will allow agents to log in as managed agents to handle previewed outbound calls.
- **TypeCode** – set to “Managed”.

To allow the agents to select from a list of dialers to log on to, the dialer host ip addresses and hostname can be defined in the following parameters.

- **ServerIPAddress** – set to the IP address of PDS12.
- **ServerHostName** – set to the PDS12 host name that should be displayed to the agent.

```
<LogonDialogParms>
  <DialogPrefix>AVAYA_MULTI </DialogPrefix>
  - <!--
    Set height and width as appropriate for the logon dialog (usually
    through trial and error).
  -->
  <Height>300</Height>
  <Width>400</Width>
  - <!--
    AgentType lists the codes to use for the Agent Type radio
    buttons.
  -->
  = <AgentType>
    <TypeCode>O</TypeCode>
    <TypeName>Outbound</TypeName>
    <Default>Y</Default>
  </AgentType>
  = <AgentType>
    <TypeCode>B</TypeCode>
    <TypeName>Blended</TypeName>
  </AgentType>
  = <AgentType>
    <TypeCode>I</TypeCode>
    <TypeName>Inbound</TypeName>
  </AgentType>
  = <AgentType>
    <TypeCode>M</TypeCode>
    <TypeName>Managed</TypeName>
  </AgentType>
  = <DialerHost>
    <ServerHostIP>192.45.120.59</ServerHostIP>
    <ServerHostName>Avaya PDS 12.0 Dialer</ServerHostName>
  </DialerHost>
  + <DialerHost>
  </LogonDialogParms>
```

Figure 12: Logon Dialog Parameters in DialerProperties.xml

4.3. Terminate Call Dialog Parameters

The Terminate Call dialog parameters define the termination codes/completion codes that the agents can use. The **TerminationCode** parameters are set to the PDS12 configured completion code numbers from the compcode.cfg file in Section 3.1.

```
<C<TerminateCallDialogParms>
- <!--
Set height and width as appropriate for the terminate call dialog (usually
  through trial and error).
-->
  <Height>365</Height>
  <Width>400</Width>
- <!--
Indicates if AutoTerminate logic should be used
-->
  <UseAutoTerminate>N</UseAutoTerminate>
  <SubmitDelay>1000</SubmitDelay>
- <!--
TerminationCodes are vendor specific values. Enter as many as desired. The
  order they appear here is the order they display in the dialog.
-->
= <TerminationCodes>
  <TerminationCode>2</TerminationCode>
  <TerminationCode>3</TerminationCode>
  <TerminationCode>4</TerminationCode>
  <TerminationCode>11</TerminationCode>
  <TerminationCode>12</TerminationCode>
  <TerminationCode>13</TerminationCode>
  <TerminationCode>15</TerminationCode>
  <TerminationCode>16</TerminationCode>
  <TerminationCode>17</TerminationCode>
  <TerminationCode>20</TerminationCode>
<
</TerminationCodes>
</TerminateCallDialogParms>
```

Figure 13: Terminate Call Dialog Parameters in DialerProperties.xml

4.4. Recall Dialog Parameters

The recall dialog parameters define the Recall dialog. The **ShowAgentOwned** parameter is set to “y” if the Agent Owned Recall feature on PDS12 is administered.

```
= <RecallDialogParms>
- <!--
Set height and width as appropriate for the recall dialog (usually through
  trial and error).
-->
```

```

    <Height>240</Height>
    <Width>400</Width>
  - <!--
  PhoneNumberFieldSize is the max characters that can be entered in the free
    form field.
    -->
    <PhoneNumberFieldSize>10</PhoneNumberFieldSize>
  - <!--
  StartTime and EndTime are the ranges of the times displayed in the recall
    time dropdown, in military time.
    -->
    <StartTime>0800</StartTime>
    <EndTime>2100</EndTime>
  - <!--
  Indicates if Agent Owned field should display
    -->
    <ShowAgentOwned>Y</ShowAgentOwned>
  </RecallDialogParms>

```

Figure 14: Recall Dialog Parameters in DialerProperties.xml

4.5. Play Message Dialog Parameters

The Play Message dialog displays the recorded audio messages that have been defined in DialerProperties.xml. The Play Message dialog will play the message for the customer and then terminate the call using the defined completion code. The format of the **MessageCode** parameter is “message number, completion code number”. In this example, the MessageCode parameter is set to “19, 20” which corresponds to the message number and configuration code number from Section 3.4.

```

<<PlayMessageDialogParms>
  - <!--
  Set height and width as appropriate for the play message dialog (usually
    through trial and error). Add approx 30 to height to display another row.
    -->
    <Height>295</Height>
    <Width>400</Width>
  - <!--
  MessageCodes are vendor specific codes for automated messages to play.
    Enter as many as desired. The order they appear here is the order they
    display in the dialog.
    -->
  = <MessageCodes>
    <MessageCode>19,20</MessageCode>
  </MessageCodes>
</PlayMessageDialogParms>

```

Figure 15: Play Message Dialog Parameters in DialerProperties.xml

4.6. Transfer Call Dialog Parameters

The Transfer Call dialog allows the agent to transfer, consult or conference. The **ShowTransferButton**, **ShowConferenceButton**, and **ShowReleaseButton** parameters should be set to “Y”. If the PDS12 has the Native Voice and Transfer feature configured, the **ShowSupervisorSection** parameter should also be set to “Y”.

```
<<TransferCallDialogParms>
- <!--
Set height and width as appropriate for the transfer call dialog (usually
through trial and error).
-->
<Height>300</Height>
<Width>400</Width>
- <!--
FieldSize tags are the max characters that can be entered in the field.
-->
<ExtensionFieldSize>5</ExtensionFieldSize>
<SupervisorIdFieldSize>8</SupervisorIdFieldSize>
- <!--
The Showxxx tags indicate if buttons (or section) displays on the dialog.
-->
<ShowTransferButton>Y</ShowTransferButton>
<ShowConferenceButton>Y</ShowConferenceButton>
<ShowReleaseButton>Y</ShowReleaseButton>
<ShowSupervisorSection>Y</ShowSupervisorSection>
</TransferCallDialogParms>
```

Figure 16: Transfer Call Dialog Parameters in DialerProperties.xml

4.7. Field Map Parameters

The Field Map parameters are needed in order for the DRC to map a field name as expected by the CACS application to the field name as defined on the call record on PDS12. For example, **CACSField** parameter “NAME” maps to the **DialerField** parameter “NAME1”. The **DialerField** parameters are the field names in the call record as shown in Section 3.6. The mapping needs to be configured based on the fields required by the CACS application.

```
<<OutboundCallRecord>
- <!--
The Outbound call record is defined by a set of FieldMaps. For each
CACSField, specify the corresponding field on the dialer (as accessed by
the DialerImpl class or ActiveX control).
-->
- <FieldMap>
<CACSField>NAME</CACSField>
<DialerField>NAME1</DialerField>
</FieldMap>
<FieldMap>
```

```

- <CACSTField>ZIP</CACSTField>
  <DialerField>ZIPCODE</DialerField>
</FieldMap>
- <FieldMap>
  <CACSTField>BALANCE</CACSTField>
  <DialerField>BALANCE</DialerField>
</FieldMap>
- <FieldMap>
  <CACSTField>NUMBER1</CACSTField>
  <DialerField>PHONE1</DialerField>
</FieldMap>
  <FieldMap>
    <CACSTField>NUMBER2</CACSTField>
    <DialerField>PHONE2</DialerField>
  </FieldMap>
- <FieldMap>
  <CACSTField>CITY</CACSTField>
  <DialerField>CITY</DialerField>
</FieldMap>
- <FieldMap>
  <CACSTField>ACCOUNT</CACSTField>
  <DialerField>ACCTNUM</DialerField>
</FieldMap>
- <FieldMap>
  <CACSTField>STATE</CACSTField>
  <DialerField>STATE</DialerField>
</FieldMap>
</OutboundCallRecord>
- <InboundCallRecord>
- <!--
The Inbound call record is defined by a set of FieldMaps. For each
CACSTField, specify the corresponding field on the dialer (as accessed by
the DialerImpl class or ActiveX control).
-->
- <FieldMap>
  <CACSTField>NUMBER1</CACSTField>
  <DialerField>ANI</DialerField>
</FieldMap>
</InboundCallRecord>

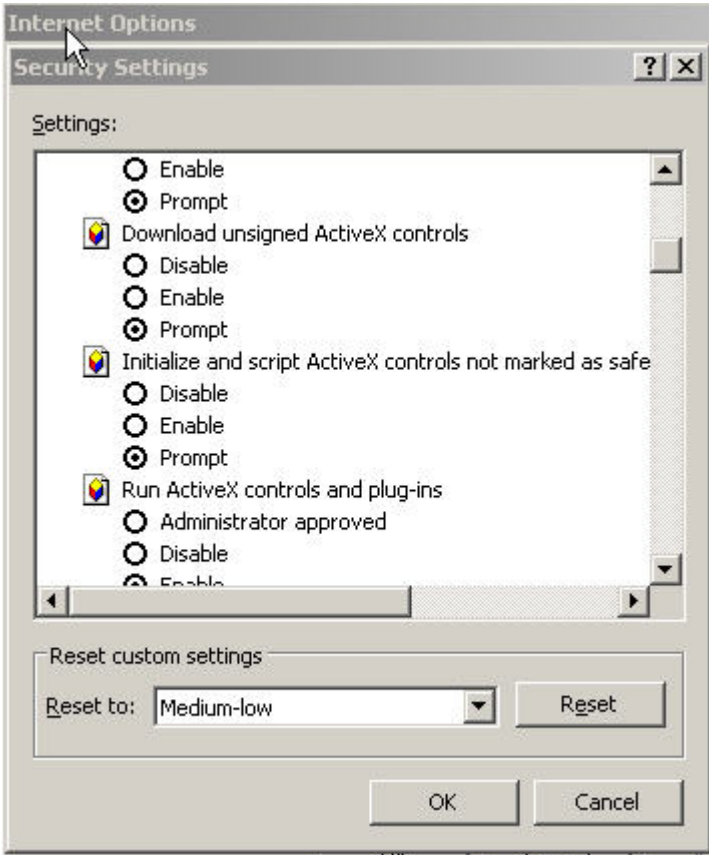
```

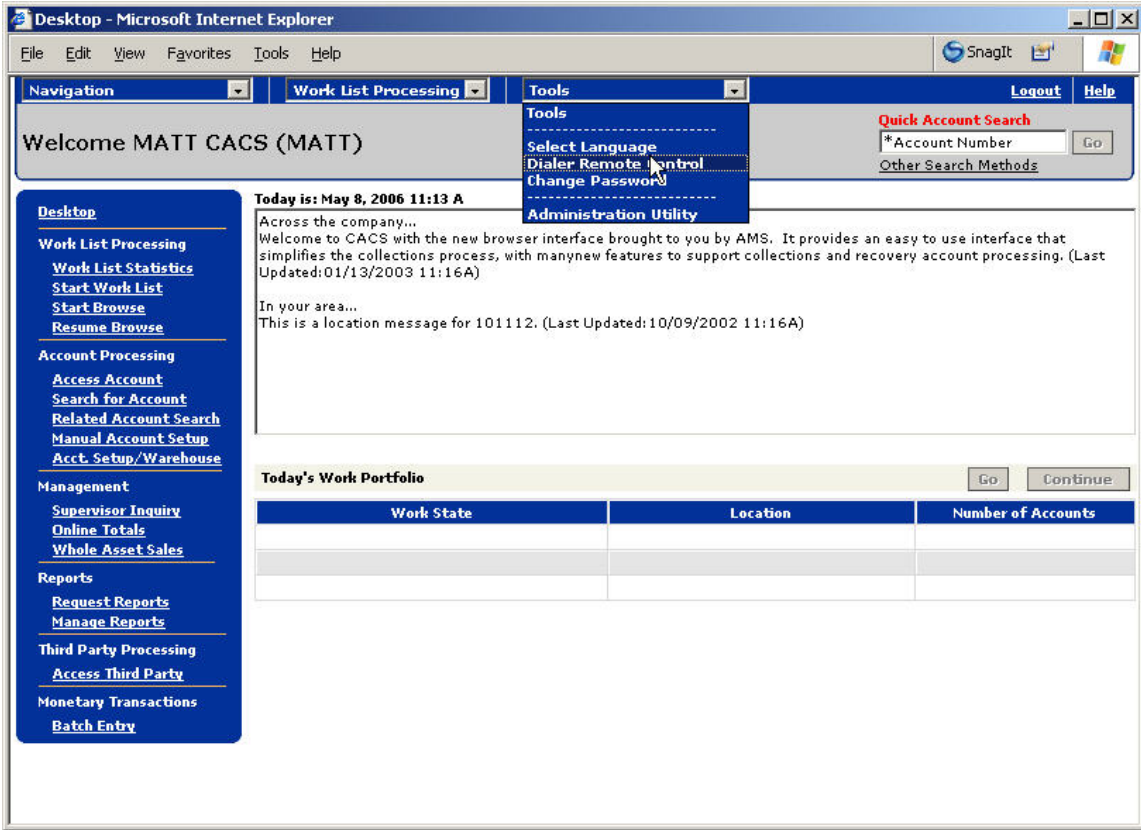
Figure 17: Field Map Parameters in DialerProperties.xml

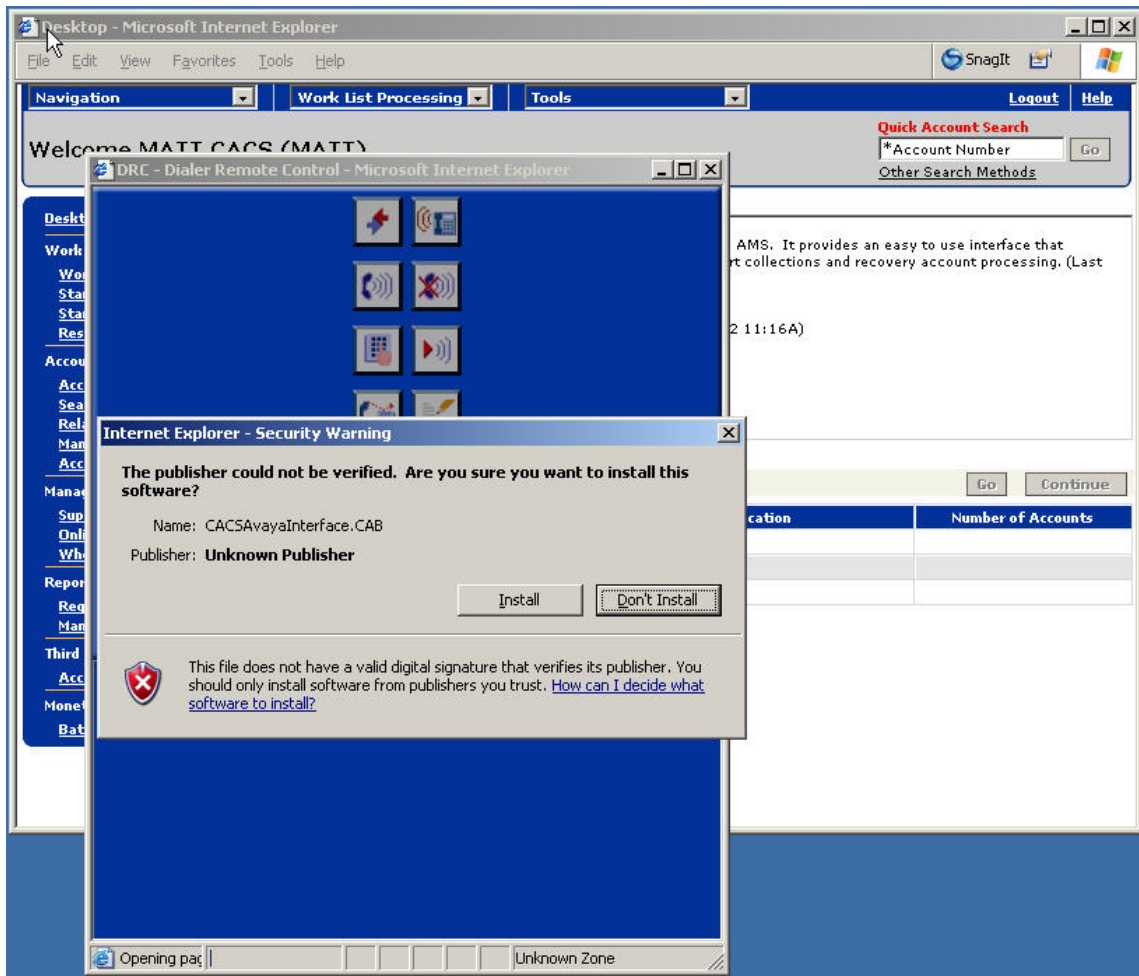
5. Configure CGI Dialer Remote Control server

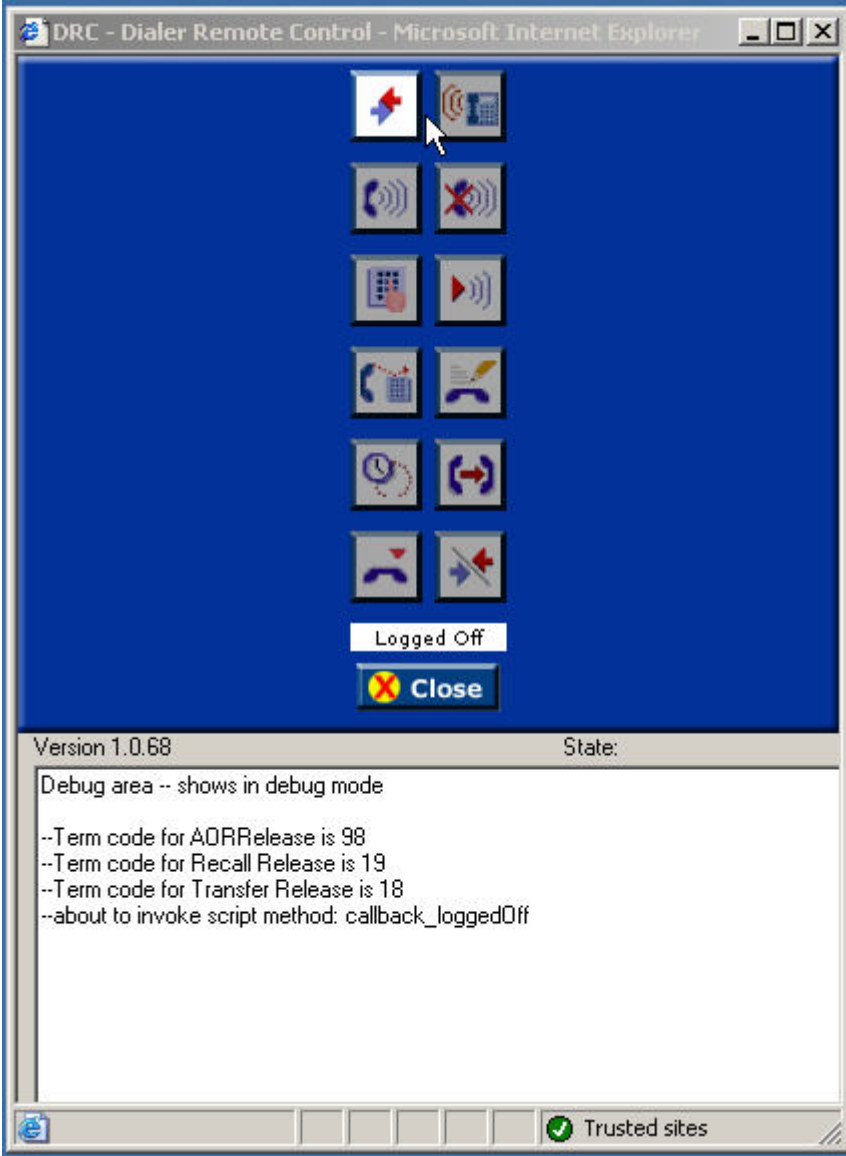
The Dialer Remote Control (DRC) can be launched when the agent connects to the CACS 8.0.4 application on the web server. The CACS 8.0.4 application contains the DRC. The DRC is the browser based agent application that is used with PDS12. When the DRC is first launched on the

agent machine, Microsoft Visual Basic Active X Control and the Avaya Agent DLL, Moagent32.dll are installed automatically on the agent machine.

Step	Description
1.	<p>Microsoft Visual Basic ActiveX Control is used to interface between the DRC and the PDS12 Agent API DLL. Launch Internet Explorer and click on Tools → Internet Options. Select the Security tab, and click on Custom Level. Select the radio button labeled “Prompt” for the Download unsigned ActiveX controls field. Click OK.</p> 

Step	Description
2.	<p>Launch Internet Explorer and enter the link to the CACS application. From the Tools drop-down menu, select Dialer Remote Control.</p> 

Step	Description
3.	<p>The following Internet Explorer – Security Warning window appears. Click Install.</p>  <p>The screenshot shows a Microsoft Internet Explorer window titled 'Desktop - Microsoft Internet Explorer'. The address bar shows 'DRC - Dialer Remote Control - Microsoft Internet Explorer'. The main content area displays a 'Welcome MATT CACS (MATT)' message. A 'Quick Account Search' section is visible with a text input field for '*Account Number' and a 'Go' button. Below this, there is a table with columns 'cation' and 'Number of Accounts'. A security warning dialog box is overlaid on the browser window. The dialog box has a title bar 'Internet Explorer - Security Warning' and a message: 'The publisher could not be verified. Are you sure you want to install this software?'. It lists the file name as 'CACS Avaya Interface.CAB' and the publisher as 'Unknown Publisher'. There are 'Install' and 'Don't Install' buttons. At the bottom of the dialog box, there is a red shield icon with a white 'X' and a message: 'This file does not have a valid digital signature that verifies its publisher. You should only install software from publishers you trust. How can I decide what software to install?'. The status bar at the bottom of the browser window shows 'Opening page' and 'Unknown Zone'.</p>

Step	Description
4.	<p>The Dialer Remote Control window appears.</p>  <p>The screenshot shows a web browser window titled "DRC - Dialer Remote Control - Microsoft Internet Explorer". The main content area has a blue background and contains a grid of 12 icons arranged in 6 rows and 2 columns. A mouse cursor is hovering over the icon in the top-right position. Below the grid of icons are two buttons: "Logged Off" and "Close". At the bottom of the window, there is a status bar with the text "Version 1.0.68" on the left, "State:" in the middle, and a "Trusted sites" indicator on the right.</p>

6. Interoperability Compliance Testing

This interoperability compliance testing covered feature functionality and serviceability testing. Feature functionality focused on verifying the proper operation of the Dialer Remote Control when integrated with Avaya Predictive Dialing System 12 using the Avaya Agent API. Serviceability testing verified that the Dialer Remote Control and CGI web server recovered from adverse conditions, such as rebooting, power failure and network disconnect.

6.1. General Test Approach

All feature functionality test cases were performed manually to verify proper operation. The general test approach was to test the features on the Dialer Remote Control that are supported with Avaya Proactive Contact 12.

- The following feature buttons on the DRC were tested.
 - Login
 - Logout
 - Return ID
 - Go Available
 - Go Unavailable
 - Terminate Call
 - Release Call
 - Transfer Call
 - Transfer Call to Supervisor
 - Manual Dial
 - Call Back
 - Agent Owned Recall
 - Place Managed Call
 - Play Message
- The following events from PDS12 to the DRC were tested.
 - AutoRelease Line
 - JobEnd
 - JobTransLink
 - JobTransRequest
 - Receive Message
 - XferCustHangup
- The following agent types were tested.
 - Outbound agent
 - Managed agent
 - Blended agent
 - Inbound agent
- Inbound calls were tested using the Intelligent Call Blending feature of Avaya Proactive Contact 12. This is the only type of inbound feature on PDS12 that is supported by CGI.

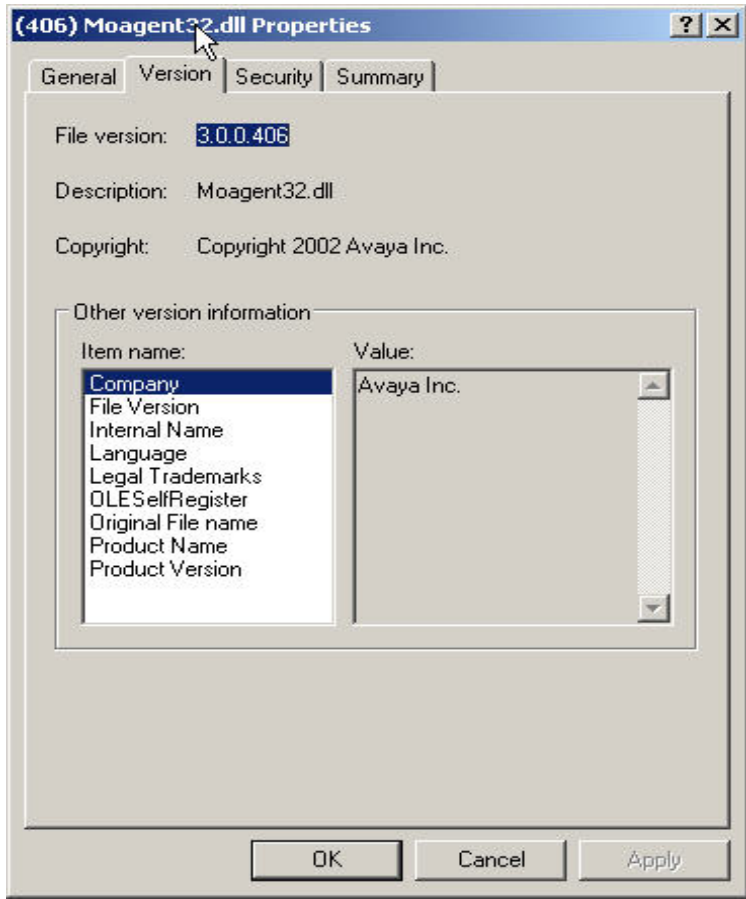
6.2. Test Results

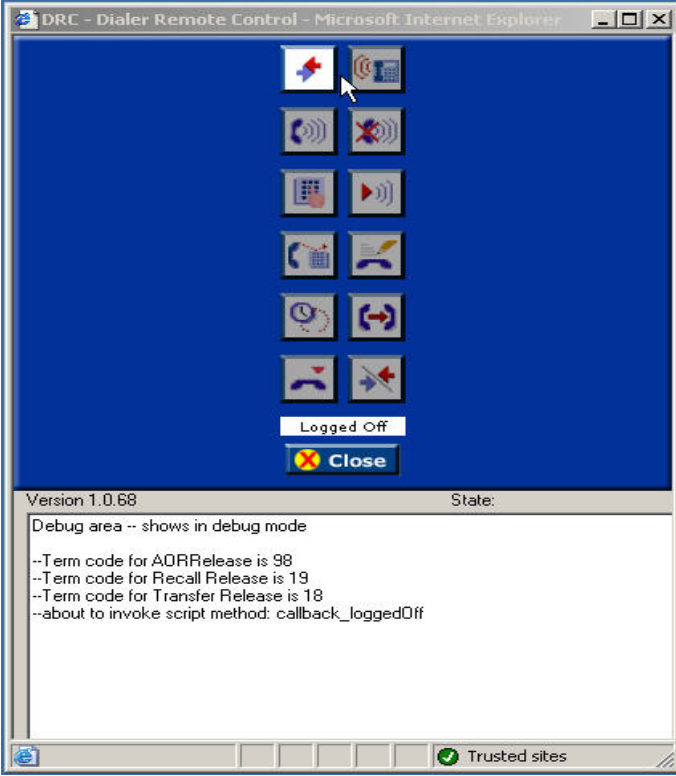
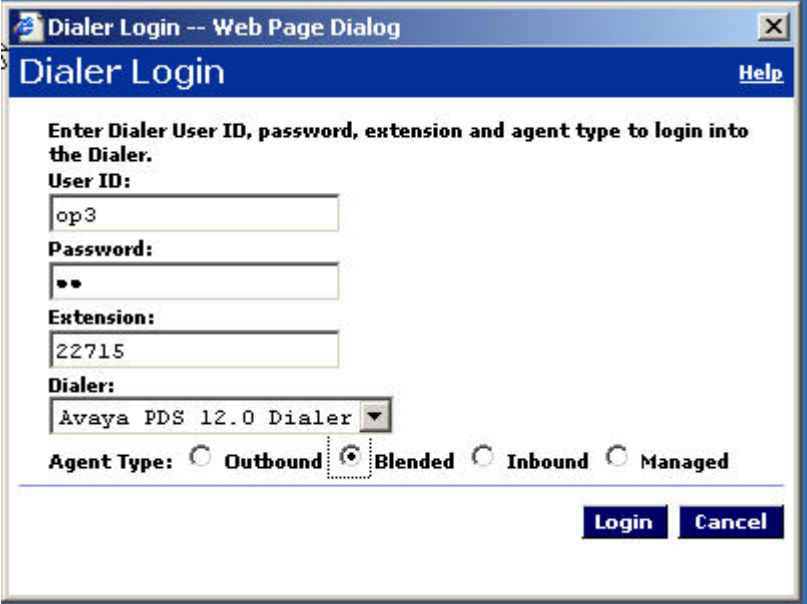
All feature and performance tests passed. The CGI CACS 8.0 –Dialer Remote Control successfully interfaced with Avaya Predictive Dialing System 12 for outbound and inbound calls. For serviceability testing, the Dialer Remote Control was able to resume handling of inbound and outbound calls after restoration of connectivity to the PDS12 server, from network disconnect/re-connect, and CGI web server resets.

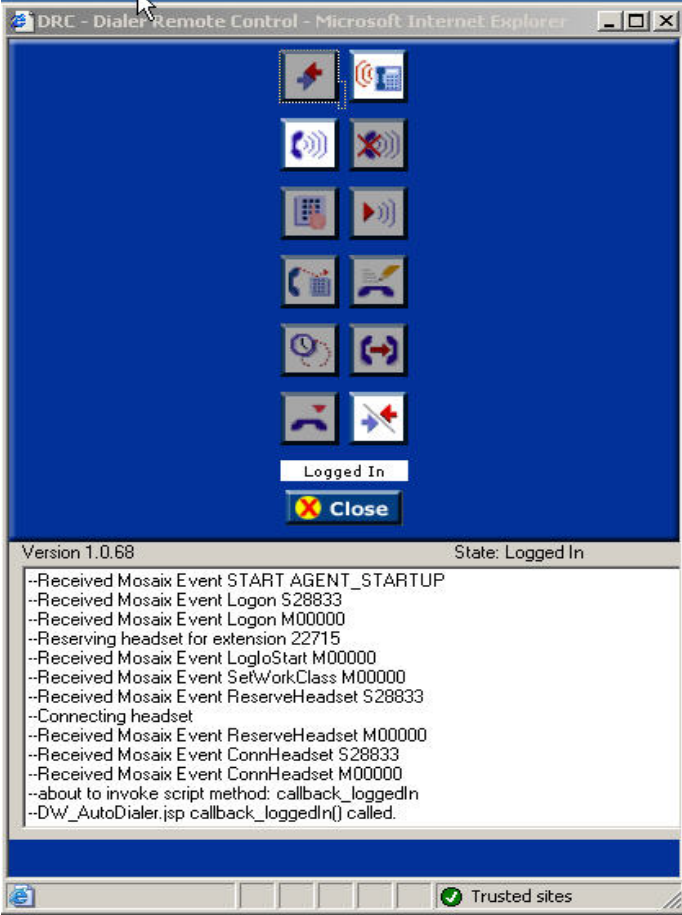
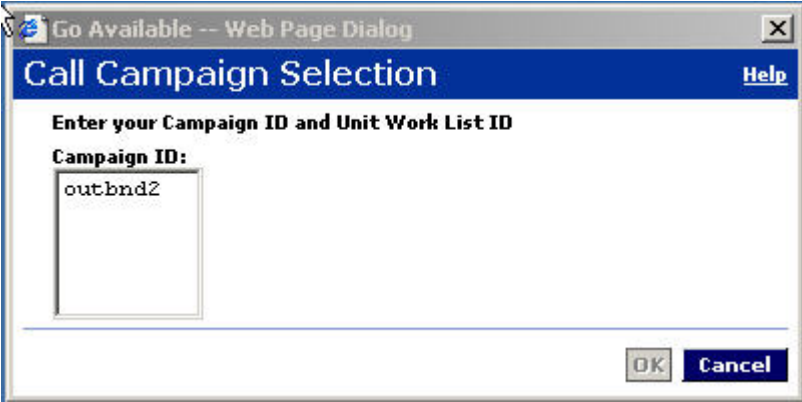
7. Verification Steps

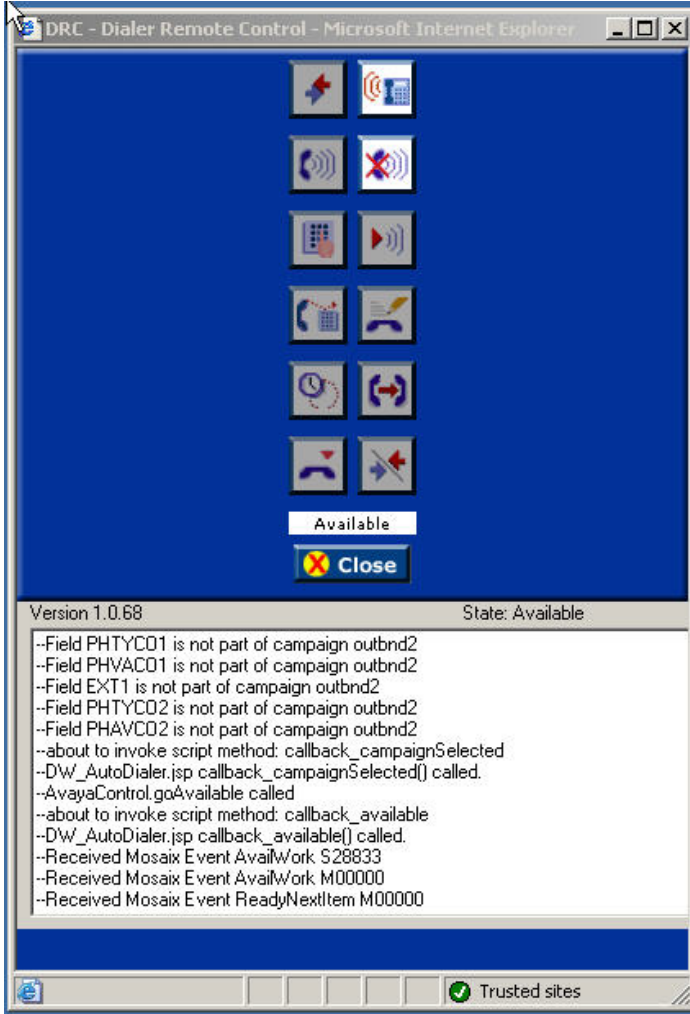
7.1. Direct Remote Control verification

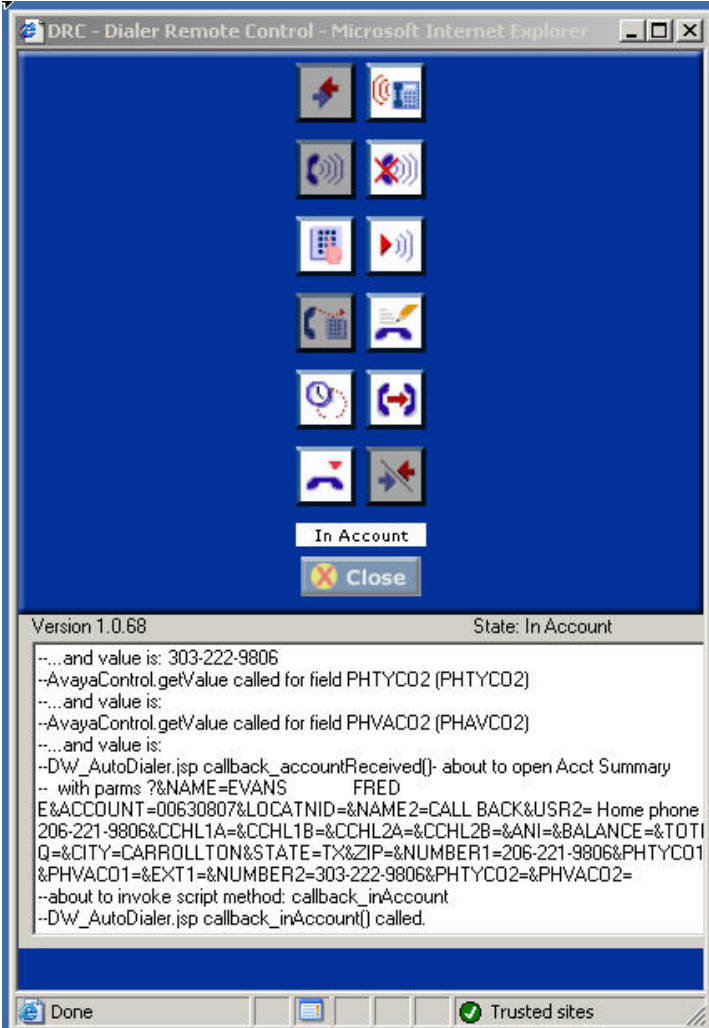
The following steps can ensure that the DRC can login to Avaya Predictive Dialing System 12.0 to receive outbound and inbound calls. PDS12 should have inbound and outbound jobs running.


Step	Description
1.	<p>On the agent machine, navigate to \WINDOWS\system32 and double click on the Moagent32.dll file. Select the Version tab. Verify the correct PDS12 Agent API DLL is installed.</p> 

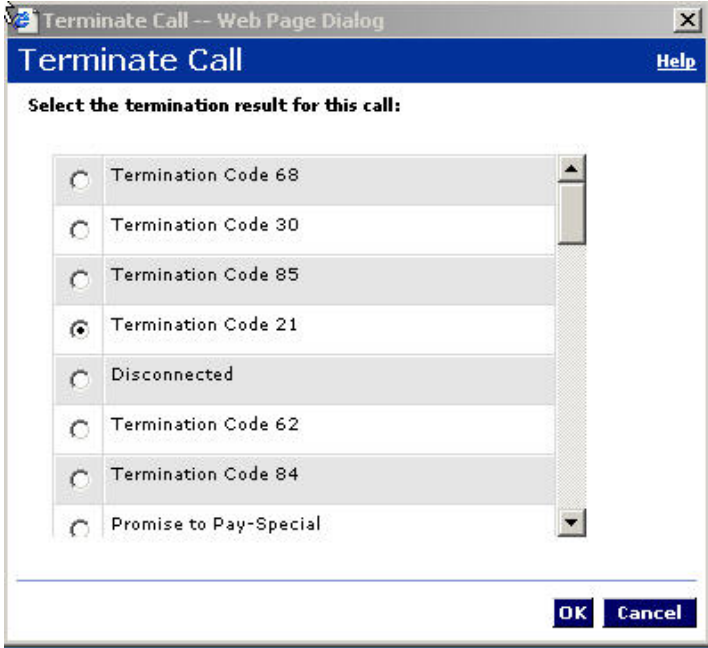
Step	Description
2.	<p>Launch the DRC from the CACS application as described in Section 5 step 2.</p> 
3.	<p>From the DRC, click the Logon button. Enter the agent login information on the Dialer Login screen.</p> 

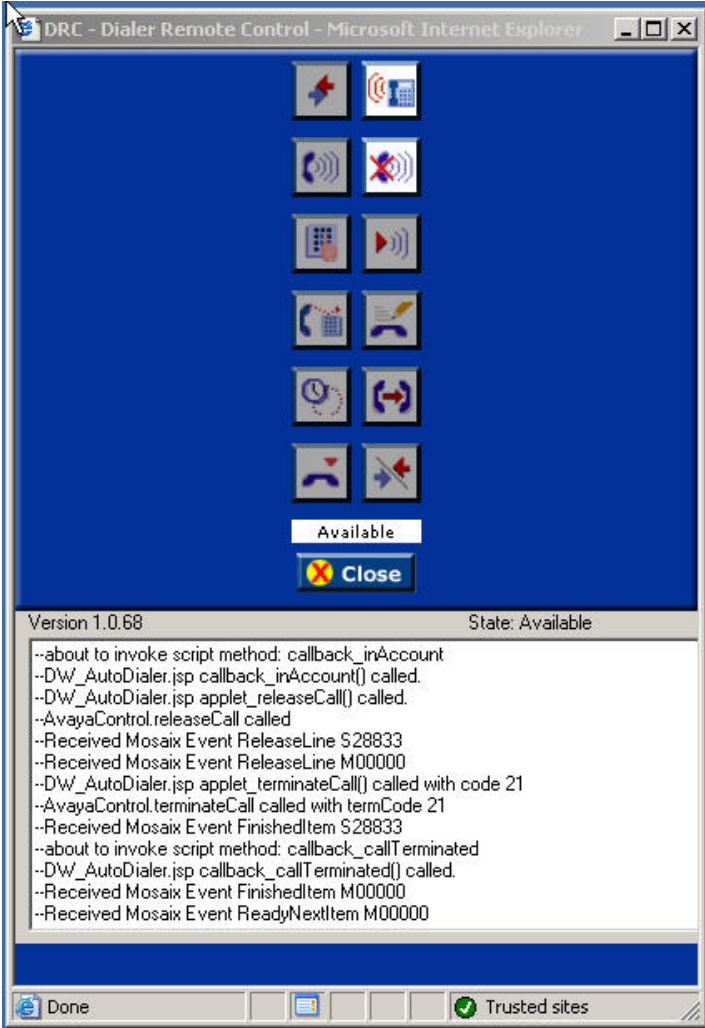
Step	Description
4.	<p>Once the agent is logged into PDS12, the DRC state should display “Logged In”.</p>  <p>Version 1.0.68 State: Logged In</p> <pre>--Received Mosaix Event START_AGENT_STARTUP --Received Mosaix Event Logon S28833 --Received Mosaix Event Logon M00000 --Reserving headset for extension 22715 --Received Mosaix Event LogloStart M00000 --Received Mosaix Event SetWorkClass M00000 --Received Mosaix Event ReserveHeadset S28833 --Connecting headset --Received Mosaix Event ReserveHeadset M00000 --Received Mosaix Event ConnHeadset S28833 --Received Mosaix Event ConnHeadset M00000 --about to invoke script method: callback_loggedIn --D'W_AutoDialer.jsp callback_loggedIn() called.</pre>
5.	<p>From the DRC, click the Go Available button. On the Call Campaign Selection screen, select the Campaign ID that the agent wants to join. Then click OK.</p> 

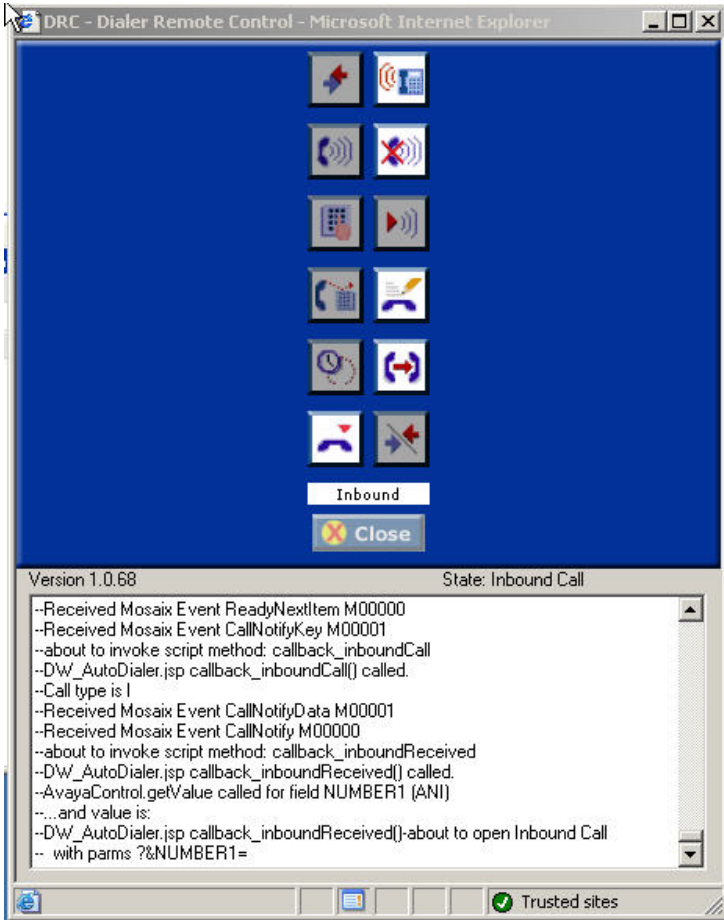
Step	Description
6.	<p>Once the agent has joined a campaign, the DRC state should display “Available”. The agent is now ready to receive inbound and outbound calls.</p>  <p>Version 1.0.68 State: Available</p> <pre>--Field PHTYCO1 is not part of campaign outbnd2 --Field PHVACO1 is not part of campaign outbnd2 --Field EXT1 is not part of campaign outbnd2 --Field PHTYCO2 is not part of campaign outbnd2 --Field PHAVCO2 is not part of campaign outbnd2 --about to invoke script method: callback_campaignSelected --D\W_AutoDialer.jsp callback_campaignSelected() called. --AvayaControl.goAvailable called --about to invoke script method: callback_available --D\W_AutoDialer.jsp callback_available() called. --Received Mosaix Event AvailWork S28833 --Received Mosaix Event AvailWork M00000 --Received Mosaix Event ReadyNextItem M00000</pre>

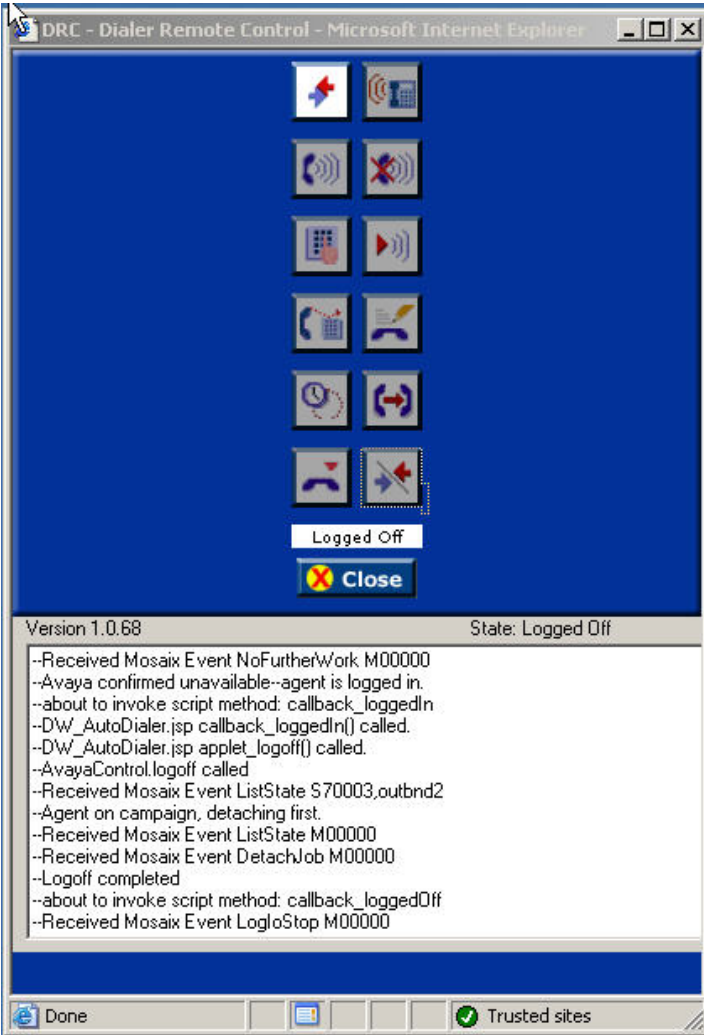
Step	Description
7.	<p>Once an outbound call is delivered to the agent, the DRC state should display “In Account”.</p> 

Step	Description
8.	<p>From the DRC, click the Release button to end the call. The DRC state should display “In Account”.</p>  <p>Version 1.0.68 State: In Account</p> <pre> --...and value is: --DW_AutoDialer.jsp callback_accountReceived()- about to open Acct Summary -- with parms ?%NAME=EVANS FRED E&ACCOUNT=00630807&LOCATNID=&NAME2=CALL BACK&USR2= Home phone 206-221-9806&CCHL1A=&CCHL1B=&CCHL2A=&CCHL2B=&ANI=&BALANCE=&TOTI Q=&CITY=CARROLLTON&STATE=TX&ZIP=&NUMBER1=206-221-9806&PTYCO1 &PHVACO1=&EXT1=&NUMBER2=303-222-9806&PTYCO2=&PHVACO2= --about to invoke script method: callback_inAccount --DW_AutoDialer.jsp callback_inAccount() called. --DW_AutoDialer.jsp applet_releaseCall() called. --AvayaControl.releaseCall called --Received Mosaix Event ReleaseLine S28833 --Received Mosaix Event ReleaseLine M00000 </pre>

Step	Description
9.	<p>From the DRC, click the Terminate button. On the Terminate Call screen, select a termination code. Click OK.</p> 

Step	Description
10.	<p>The DRC state should display “Available” and now is ready to receive outbound or inbound call.</p>  <pre> --about to invoke script method: callback_inAccount --D\W_AutoDialer.jsp callback_inAccount() called. --D\W_AutoDialer.jsp applet_releaseCall() called. --AvayaControl.releaseCall called --Received Mosaix Event ReleaseLine S28833 --Received Mosaix Event ReleaseLine M00000 --D\W_AutoDialer.jsp applet_terminateCall() called with code 21 --AvayaControl.terminateCall called with termCode 21 --Received Mosaix Event FinishedItem S28833 --about to invoke script method: callback_callTerminated --D\W_AutoDialer.jsp callback_callTerminated() called. --Received Mosaix Event FinishedItem M00000 --Received Mosaix Event ReadyNextItem M00000 </pre>

Step	Description
11.	<p>When an inbound call is delivered to the agent, the DRC state should display “Inbound”. The agent can release and terminate the calls as described in Steps 8 and 9.</p>  <p>The screenshot shows the DRC - Dialer Remote Control application running in Microsoft Internet Explorer. The main interface has a blue background with a grid of icons for call control. Below the icons are buttons labeled "Inbound" and "Close". At the bottom, a status bar shows "Version 1.0.68" and "State: Inbound Call". A log window at the bottom displays a series of system events and script calls.</p> <pre> --Received Mosaix Event ReadyNextItem M00000 --Received Mosaix Event CallNotifyKey M00001 --about to invoke script method: callback_inboundCall --DW_AutoDialer.jsp callback_inboundCall() called. --Call type is I --Received Mosaix Event CallNotifyData M00001 --Received Mosaix Event CallNotify M00000 --about to invoke script method: callback_inboundReceived --DW_AutoDialer.jsp callback_inboundReceived() called. --AvayaControl.getValue called for field NUMBER1 (ANI) --...and value is: --DW_AutoDialer.jsp callback_inboundReceived()-about to open Inbound Call -- with parms ?&NUMBER1= </pre>

Step	Description
12.	<p>From the DRC, click the Go Unavailable button to leave the campaign. Then click the Logoff button to log off the agent from PDS12. The DRC state should display “Logged off”</p>  <p>Version 1.0.68 State: Logged Off</p> <pre>--Received Mosaix Event NoFurtherWork M00000 --Avaya confirmed unavailable-agent is logged in. --about to invoke script method: callback_loggedIn --D'W_AutoDialer.jsp callback_loggedIn() called. --D'W_AutoDialer.jsp applet_logoff() called. --AvayaControl.logoff called --Received Mosaix Event ListState S70003,outbnd2 --Agent on campaign, detaching first. --Received Mosaix Event ListState M00000 --Received Mosaix Event DetachJob M00000 --Logoff completed --about to invoke script method: callback_loggedOff --Received Mosaix Event LogloStop M00000</pre>

8. Support

If technical support is required for the CGI CACS 8.0 – Dialer Remote Control solution, contact CGI Technical Support. Full details are available at <http://www.cgi.com>.

9. Conclusion

These Application Notes describe the required configuration steps for CGI CACS 8.0 – Dialer Remote Control to successfully interoperate with the Agent API of Avaya Predictive Dialing System 12.0. Functionality and performance were successfully validated. The configuration described in these Application Notes has been successfully compliance tested.

10. Additional References

The following documents may be found at <http://support.avaya.com>:

- Administrator's Guide for Avaya Communication Manager, Release 3.0.1, Issue 1, June 2005; Doc ID: 03-300509
- Avaya Predictive Dialing System Administration Manager's User Guide, August 2002; Doc ID: 585-315-501
- Avaya Proactive Contact 3.0 Administration (UNIX-based), October 2005; Doc ID: 07-300488

CGI product documentation is available on request from <http://www.cgi.com>.

- CACS Enterprise Browser AutoDialer Guide, July 2004;Release 8.0.4

©2006 Avaya Inc. All Rights Reserved.

Avaya and the Avaya Logo are trademarks of Avaya Inc. All trademarks identified by ® and ™ are registered trademarks or trademarks, respectively, of Avaya Inc. All other trademarks are the property of their respective owners. The information provided in these Application Notes is subject to change without notice. The configurations, technical data, and recommendations provided in these Application Notes are believed to be accurate and dependable, but are presented without express or implied warranty. Users are responsible for their application of any products specified in these Application Notes.

Please e-mail any questions or comments pertaining to these Application Notes along with the full title name and filename, located in the lower right corner, directly to the Avaya DeveloperConnection Program at devconnect@avaya.com.