A MITEL
PRODUCT
GUIDE

MiContact Center Enterprise

SAP Technical Integration User Guide

Release 9.8
Document Version 1.0

August 2025

Mitel
Powering connections

Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by **Mitel Networks™ Corporation (MITEL®).**

The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

Trademarks

The trademarks, service marks, logos and graphics (collectively “Trademarks”) appearing on Mitel's Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively "Mitel") or others. Use of the Trademarks is prohibited without the express consent from Mitel. Please contact our legal department at legal@mitel.com for additional information. For a list of the worldwide Mitel Networks Corporation registered trademarks, please refer to the website: http://www.mitel.com/trademarks.

®,™ Trademark of Mitel Networks Corporation

INTRODUCTION 4

SAP INTERACTION CENTER WEB CLIENT 4

Integration Architecture 4

Application Programming Interface 8

Commands 8

Call Data 9

Installation and Deployment 10

Hardware and Software Requirements 10

License Requirements 10

Networking Requirements 10

Skill Requirements 10

Pre-Installation Requirements 11

Installation Process 11

Configuration Parameters 14

Other Configuration Considerations 14

Running the SAP IC Web Client 15

SAP BUSINESS API 16

Configuration Parameters 17

Log Files Created 17

Components Summary 17

Open SAP Connection 18

Execute SAP Function 19

Parameter Bindings 20

Close SAP connection 21

INTRODUCTION

This document describes the integration between MiCC Enterprise and:

1. SAP CRM IC Web Client
2. SAP Business API

Note: Please refer to *MiCC Enterprise System Engineering Guidelines* for supported versions of Siebel eBusiness Application.

SAP INTERACTION CENTER WEB CLIENT

This section describes the integration between SAP CRM IC Web Client and MiCC Enterprise. The reader is assumed to be familiar with both products.

For more information on SAP IC Web Client architecture and functionality, refer to the SAP document Integrated Communication Interface.

INTEGRATION ARCHITECTURE

An overview of the SAP IC Web Client and MiCC Enterprise integration is shown in Figure 1.

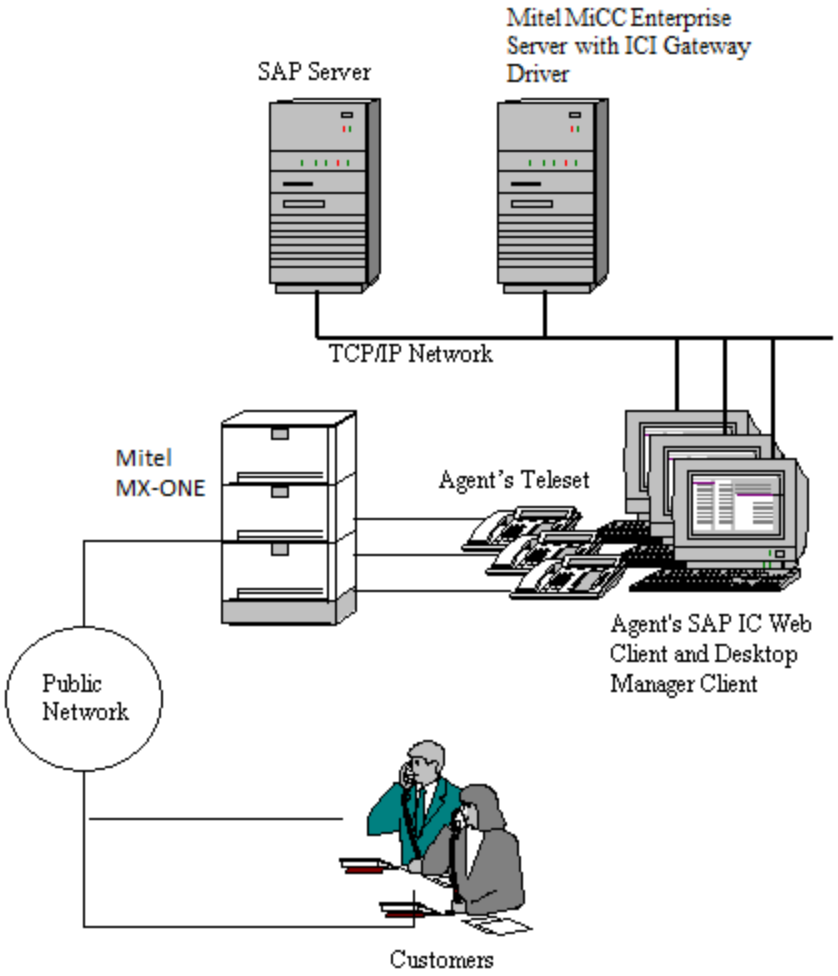


Figure 1: Integration Architecture Overview

An SAP ICI gateway driver is run on the MiCC Enterprise server. The ICI gateway driver communicates with SAP and the MiCC Enterprise server over HTTP(S) and SOAP. The driver is developed as a .NET web service running under Internet Information Server (IIS).

On each client workstation:

- A SAP IC Web client is run under an Internet browser.
- The MiContact Center Agent needs to be installed. Note that SAP integration is only supported for the desktop Agent. It is not supported for Web Agent.

On the MiCC Enterprise server:

- Internet Information Server and the MiCC Enterprise ICI gateway driver need to be installed.

Every user also has a physical telephone or uses softphone for the voice communication. The integration enables the agents to execute all the basic call controls from their SAP CRM IC Web Client interface as well as receive screen pops based on different types of telephony data, such as Automatic Number Identification (ANI), Service Group name, Interactive Voice Response (IVR) data and the like.

The screenshot in Figure 2 illustrates the SAP CRM IC Web Client handling an incoming call.

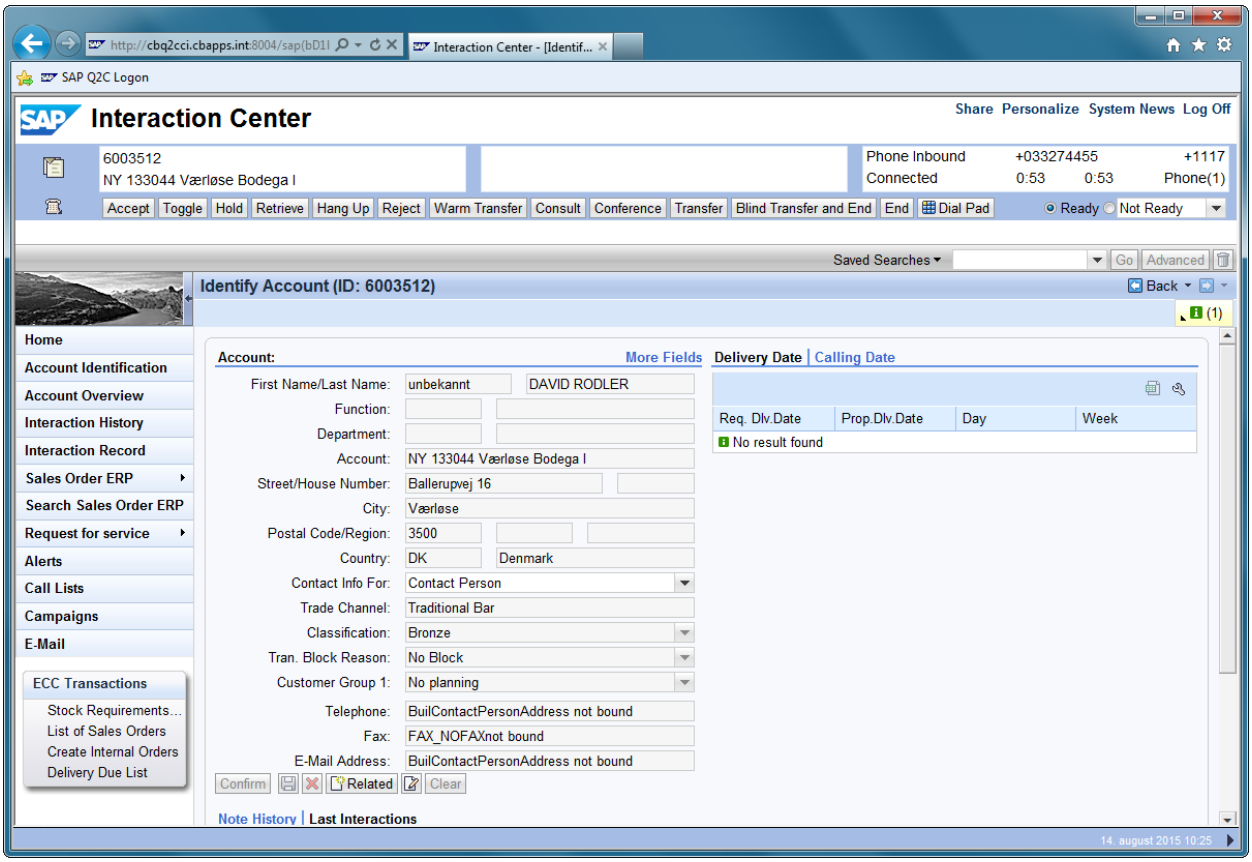


Figure 2: SAP CRM IC Web Client

The command and event flow between the SAP CRM IC Web Client and MiContact Center Agent is illustrated in Figure 3.

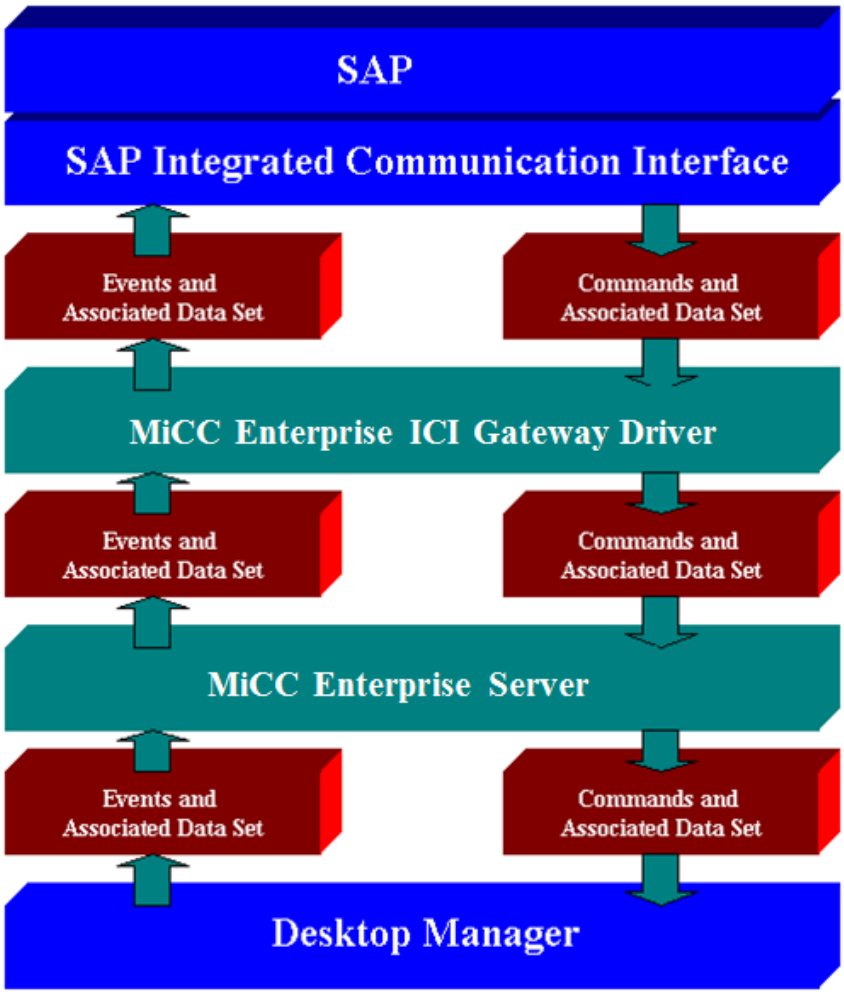


Figure 3: Commands and Event Flow

The following is an inbound call scenario of command and event flow in the integration:

1. Customer calls to the call center.
2. The call is delivered to a call center agent that is logged on to MiContact Center Agent and using the SAP CRM IC Web Client.
3. When the call arrives to the agent, MiContact Center Agent sends an Alerting event along with associated data event such as ANI and Service Group name to the MiCC Enterprise server.
4. The MiCC Enterprise server sends this information to the ICI gateway driver.
5. The ICI gateway driver sends the information to SAP.
6. The SAP ICI Web Client gets notified of the event and its associated data from the SAP server. It recognizes Alerting as a new inbound call, and uses the ANI to retrieve the customer profile.
7. The agent initiates an Accept call command by clicking the Accept button from the IC Web Client to be in speech with the customer.

APPLICATION PROGRAMMING INTERFACE

COMMANDS

The commands that can be initiated from the SAP CRM IC Web Client are listed in Table 1. Refer to SAP document SAP Integrated Communication Interface for a complete description of the CTI protocol.

Table 1: Command Name Description

COMMAND NAME	DESCRIPTION
IciContainer_subscribe	Subscribes to a phone line to retrieve call information events.
IciContainer_unsubscribe	Unsubscribes from a phone line.
IciItem_accept	Answers an inbound call.
IciItem_create	Creates a blank item to be used for a new outbound call.
IciItem_deleteAttachedDat a	Deletes call attached data.
IciItem_forward	Forwards an active call to a new destination.
IciItem_getAttachedData	Retrieves call attached data.
IciItem_reroute	Rejects an incoming service group call.
IciItem_setAttachedData	Sets call attached data.
IciItem_wrapUpEnded	Cancels clerical time for a service group call.
IciItem_wrapUpRequired	Extends clerical time for a service group call.
IciPhoneCall_alternate	Toggles between an active and held call.
IciPhoneCall_conference	Initiates a conference call between three parties.
IciPhoneCall_consult	Initiates an outbound call placing the current call on hold.
IciPhoneCall_dial	Places an outbound call.
IciPhoneCall_drop	Hangs up the active call.
IciPhoneCall_getAttributes	Retrieves information on the specified call.
IciPhoneCall_hold	Places the active call on hold.
IciPhoneCall_reconnect	Hangs up the active call and retrieves the last held call.
IciPhoneCall_transfer	Transfers the last held call to the active call party.
IciPhoneCall_unhold	Retrieves the last held call.
IciPhoneLine_dropCalls	Hangs up all calls.

COMMAND NAME	DESCRIPTION
IciPhoneLine_getCalls	Retrieves a list of all calls.
IciSystem_exchangeProductInformation	Used to exchange version information between SAP and the telephony gateway. This command is only invoked during a SAP BCB connection test to determine compatibility.
IciUser_getAttributes	Retrieves information on the current user.
IciUser_setCurrentWorkmode	Sets the current workmode for the user.
IciUser_subscribe	Subscribes to the user to receive user update events.
IciUser_unsubscribe	Unsubscribes from the user.

CALL DATA

Call data fields are listed in Table 2. This data is sent to the SAP IC Web Client when call information is requested. This data is sent in the form of an XML document.

Table 2: Call Data

EVENT DATA NAME	DESCRIPTION
SERVICEGROUP	The name of the service group the call has distributed to.
IVR1	Interactive Voice Response data. The first data retrieved from the IVR system.
IVR2	Interactive Voice Response data. The second data retrieved from the IVR system.
IVR3	Interactive Voice Response data. The third data retrieved from the IVR system.
CONSULTCALLEDNUM	Called number on consultation call from one agent to another. Can be used, for example, data screen transferring based on called number.
CONSULTSERVICEGROUP	The name of the service group on consultation call from one agent to another.
CONSULTIVR1	The first data retrieved from the IVR system on consultation call from one agent to another. Can be used, for example, data screen transferring.
CONSULTIVR2	The second data retrieved from the IVR system on consultation call from one agent to another. Can be used, for example, data screen transferring.
CONSULTIVR3	The third data retrieved from the IVR system on consultation call from one agent to another. Can be used, for example, data screen transferring.
PRIVATEDATA	Data that is tag along the call event from, for example, an external IVR system.

INSTALLATION AND DEPLOYMENT

Deployment of SAP CRM and MiCC Enterprise involves the following products:

1. Installation and configuration of the MX-ONE Telephony Switch.
2. Installation of SAP CRM server.
3. Installation and configuration of MiCC Enterprise Server.
4. Installation of MiContact Center Agent.
5. Administration of SAP CRM CTI configurations, settings and files.

HARDWARE AND SOFTWARE REQUIREMENTS

Please refer to SAP R/3 documentation for details on SAP R/3 client and server hardware and software requirements.

Please refer to document Installation Preparations for server and client hardware and software requirements.

LICENSE REQUIREMENTS

The following licenses are required for the integration to work. They can be ordered in CPQ and should be loaded in ELM before the integration can be tested.

Item	Description	Quantity
FAL1047726	MiCC Enterprise SM SAP BAPI Interface, system license	1
FAL1047799	MiCC Enterprise SAP ICI Integration, system license	1
FAL1047779	MiCC Enterprise DDE/COM Desktop Integration, 1 user	1 per agent

NETWORKING REQUIREMENTS

The networking protocol must be TCP/IP for MiCC Enterprise platform.

SKILL REQUIREMENTS

The person who is installing and configuring SAP CRM CTI needs to know the basics of SAP CRM and CTI as well as how to configure SAP CRM CTI configurations parameters and modifying the events in order to do the appropriate screen pops.

See the Sap Wiki pages for an overview of SAP Interaction Center (https://wiki.scn.sap.com/wiki/display/CRM/Interaction+Center)

Product knowledge of MiCC Enterprise is required in order to do installation and configuration of it.

PRE-INSTALLATION REQUIREMENTS

1. Installation and configuration of MX-ONE Telephony Switch.
2. Installation and configuration of Open Application Server.
3. Installation and configuration of MiCC Enterprise Server including Web Services.
4. Installation and configuration of SAP CRM Server.

INSTALLATION PROCESS

The installation process assumes that SAP CRM server and MiCC Enterprise server are installed and configured. The steps describe only the necessary steps to enable SAP integration. The MiCC Enterprise ICI gateway driver is included in the installation of the MiCC Enterprise server. Be sure that the Web Services feature is installed when installing the MiCC Enterprise server.

MiCC Enterprise configuration

1. Install MiContact Center Agent software on each client PC.
2. Use MiCC Enterprise Configuration Manager to define the Broadcast Parameters for the Agent Service. They are located in the Contact Center System Properties on the Miscellaneous tab.

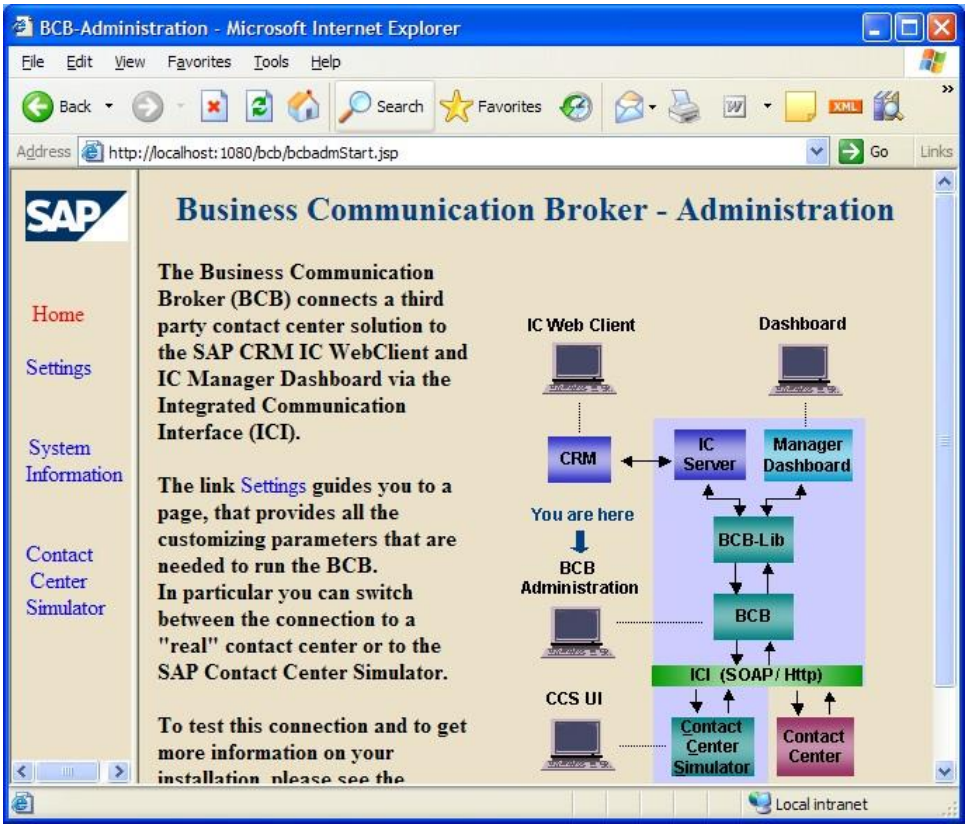
   The values to be defined are as follows:

   a. Broadcast IP Address – Specifies the IP address to be used for broadcasting (valid addresses are between 224.0.0.0 and 239.255.255.255).
   b. Broadcast Port – Specifies the port to be used for broadcasting.
   c. Broadcast Max Routers – Specifies the maximum number of routers.
   d. Broadcast IP Address for Network Card – Set to use a specific IP address when multiple network cards are installed.

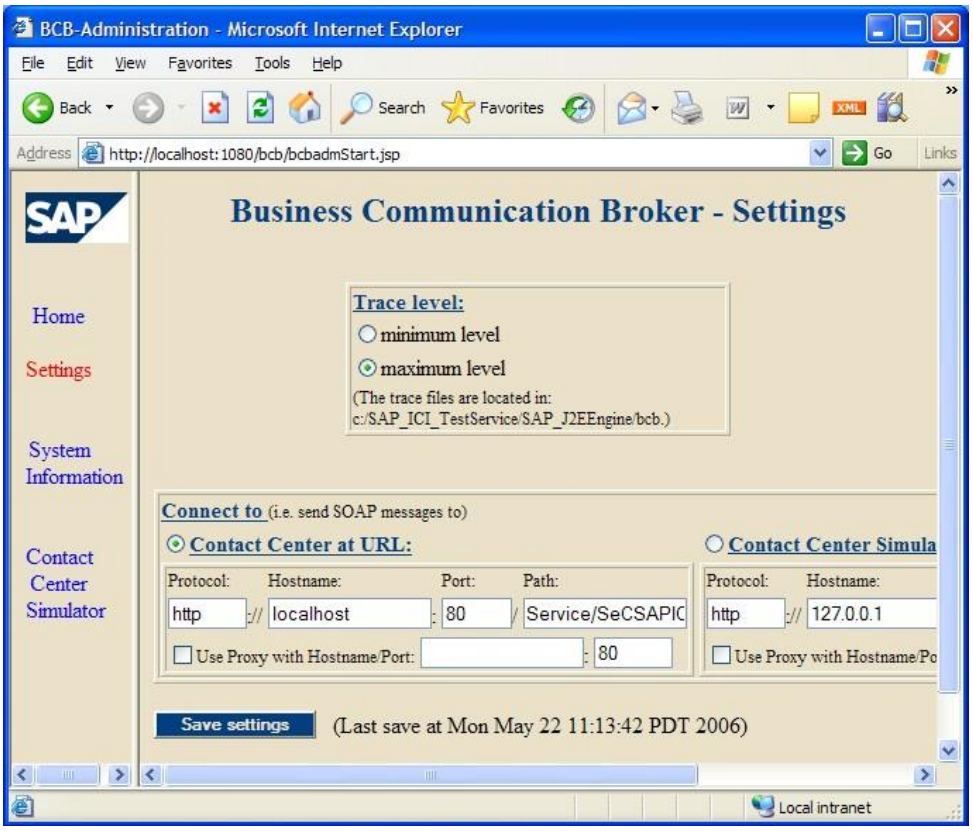
SAP CRM version 5.0 or older

The MiCC Enterprise ICI gateway driver must be configured in SAP CRM in order for the client to have CTI integration. This process only needs to be performed once and is used by all SAP CRM clients.

1. From the MiCC Enterprise Server machine, open an Internet browser window.
2. Enter the URL of the SAP BCB.
   For example: http:/<SAPSERVER>:1080/bcb



3. Select **Settings**



4. Add a Contact Center connection and set **Protocol** to **http**, **Hostname** to the name of the MiCC Enterprise server machine and **Port** to **80** (or actual port of IIS if configured to listen on another port), **Path** to SeCSAPICIDriver/SeCSAPICIService.asmx.
5. Contact your SAP administrator to see if Proxy Hostname/Port setting is necessary.
6. Set the new connection as the default.
7. Save the settings and test the connection.

SAP CRM version 6.0 and later

Define and configure an RFC Destination pointing to the SAP ICI Driver web service of MiCC Enterprise, as illustrated in Figure 4. Refer to the SAP documentation for more details.

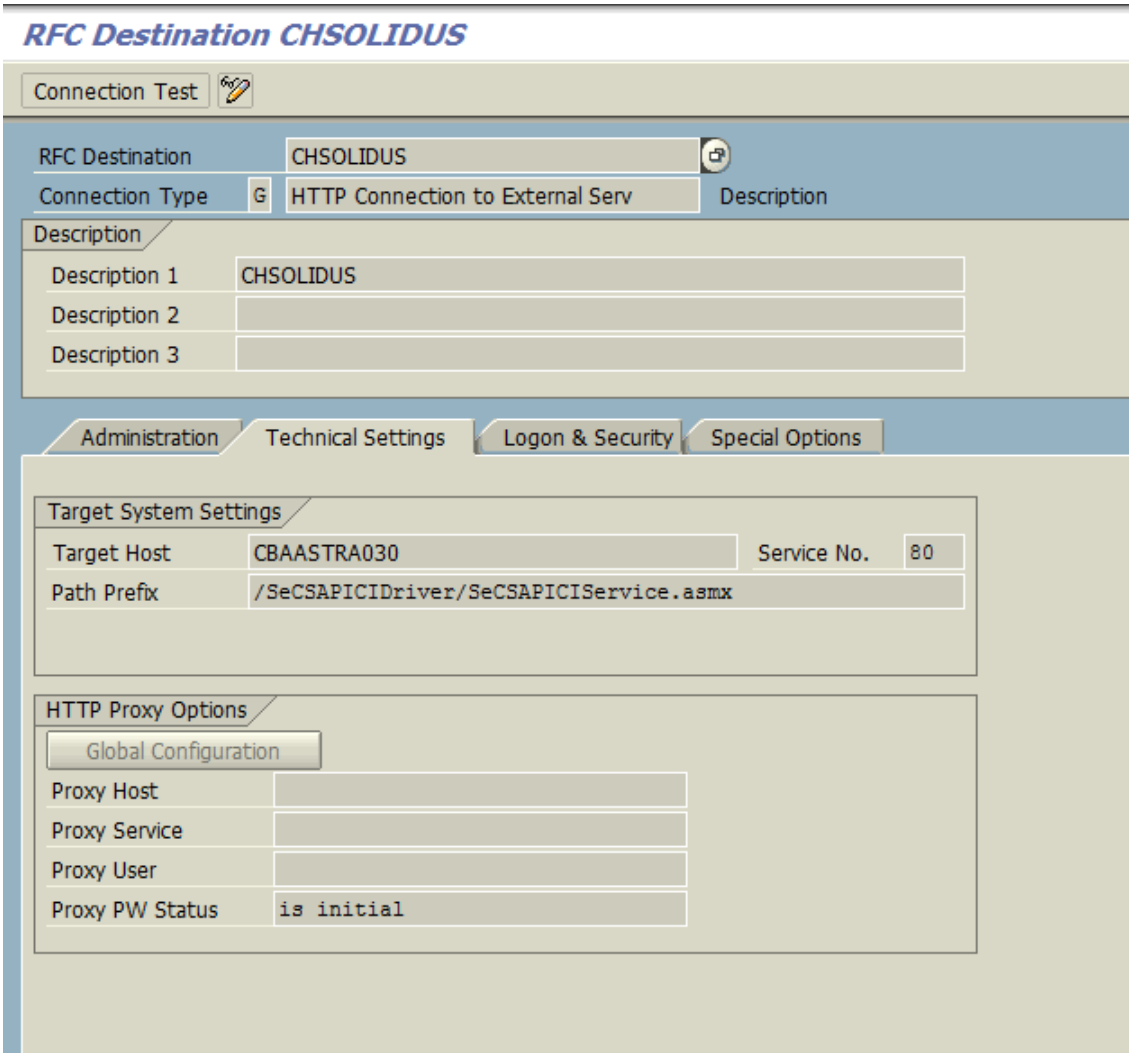


Figure 4: RFC Destination

CONFIGURATION PARAMETERS

Supported configuration parameters and applicable default values for a configuration is shown in Table 3. These configuration parameters are viewed and edited by using the Registry Editor installed as part of the operating system. All entries are defined under the following key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Mitel\SeC\Common\Parameters\Services\SeCSAPICIDriver
```

VALUE NAME	TYPE	DESCRIPTION
Enabled	REG_DWORD	The SAP ICI Gateway Driver has to be enabled. It is always installed, but installed as disabled. To enable it, set Enable to 1. Values: 0 = Disabled (default) 1 = Enabled
LogLevel	REG_DWORD	Sets the level of trace information stored in the SeCSAPICIDriver.log file. Values: 0 = No trace information 1 = Minimal trace (default) 2 = Extended trace
MaxLogFiles	REG_DWORD	Sets the maximum number of log files to create. Default = 10.
IncludeNotReady Reason	REG_DWORD	Indicates if the current not ready reason is included in the workmode setting. Values: 0 = Not included (default) 1 = Included If set to include the not ready reason, the workmode will be in the format: <workmodeid>[:notreadyreasonid> The workmode is set in IciUser_setCurrentWorkmode call and retrieved in the IciUser_getAttributes call. The workmode is passed as a simple string in IciUser_setCurrentWorkmode. The workmode returned in IciUser_getAttributes is a complex type consisting of an id member and description member.
ClientCertificates	REG_SZ	Specifies any client certificates required when sending events to SAP over https. This should specify the subject name of the certificate. Multiple certificates may be specified by separating each name with a semicolon ; Certificates must be installed under Personal\Certificates in the local machine certificate store.

Table 3: SAP ICI Configuration Parameters

OTHER CONFIGURATION CONSIDERATIONS

In MiCC Enterprise Configuration Manager there is an option to set the Number Translation for outgoing calls. This option should be turned off to use number translation rules defined in SAP R/3.

MiContact Center Agent allows users to log on from any workstation; this is called the Free Seating feature.

ADDITIONAL SAP ICI CONFIGURATION SETTINGS

See the SAP ICI Wiki for a Step-by-Step guide for how to configure the ICI contact push: https://wiki.scn.sap.com/wiki/display/CRM/Step+by+step+to+show+you+how+to+set+up+the+ICI+contact+push+scenario+for+telephony

RUNNING THE SAP IC WEB CLIENT

Before launching the SAP IC Web Client, MiContact Center Agent must be running on the client PC. The account logged onto MiContact Center Agent must be the same as the account used for logon to the IC Web Client.

For the integration to be fully utilized, “MiCC Enterprise DDE/COM Desktop Integration 1 User License” will be needed for the agents. The agent must also be assigned the “Use Integration Interface” privilege.

SAP BUSINESS API

In Script Manager, the SAP Component Library can be used to develop scripts (see Figure 5) that will allow the IVR system to be integrated with the SAP system via Business Applications Programming Interface (BAPI). The SAP Component library includes three blocks:

- Open SAP Connection
- Execute SAP function
- Close SAP Connection

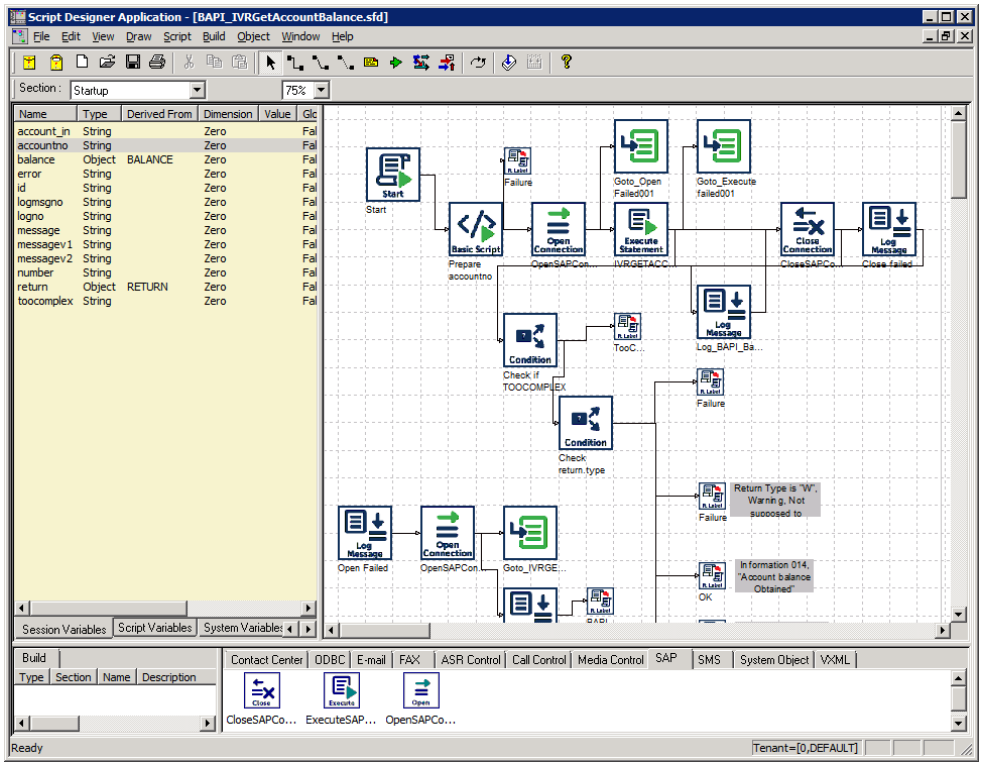


Figure 5: Screen Shot of Script Designer and SAP Component Library

CONFIGURATION PARAMETERS

Create a new registry key called HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Mitel\SM\SAPConnector. Under this key, the registry entries in Table 4 can be created:

Table 4: SAP Connector Configuration Parameters

VALUE NAME	TYPE	DESCRIPTION
RFCLogging	DWORD	Possible values: 0 , 1
LogLevel	DWORD	Possible values: 0 - 10
MaxLogFiles	DWORD	Min. value: 1 Max. value: 100
MaxLogSize	DWORD	Min. value: 1024 Max. value: 10*24 (Represented in Kilo-Bytes)

LOG FILES CREATED

Each Service Application creates a new log file for SAP Component library.
`sapconnector_<random value>.log`

Error and exception logging is always on, the enhanced and debugging option can be activated by enabling component logging in Script Manager Configuration utility.

COMPONENTS SUMMARY

In Table 5 the components in the SAP component library are described.

Table 5: SAP Components

COMPONENT	ICON	DESCRIPTION
OpenSAPConn ection		Connects to the specified SAP System.
ExecuteSAPFu nction		Executes a SAP BAPI Function and binds the resulting parameters based on the type selected. When a bind is applied to a parameter, the parameter is replaced by the content of the variable. When a bind is applied to parameter, the data retrieved from the database is copied to the variable specified.
CloseSAPCon nection		Disconnects from the specified SAP System.

OPEN SAP CONNECTION

The OpenSAPConnection block connects to the SAP system specified in the Settings tab of the OpenSAPConnection Properties dialog box, see Figure 6. In the settings tab, the user specifies the name and the connection information for the SAP. The name entered will be used later in the ExecuteSAP and CloseSAP blocks. The user can specify if connection pooling and load balancing will be used.

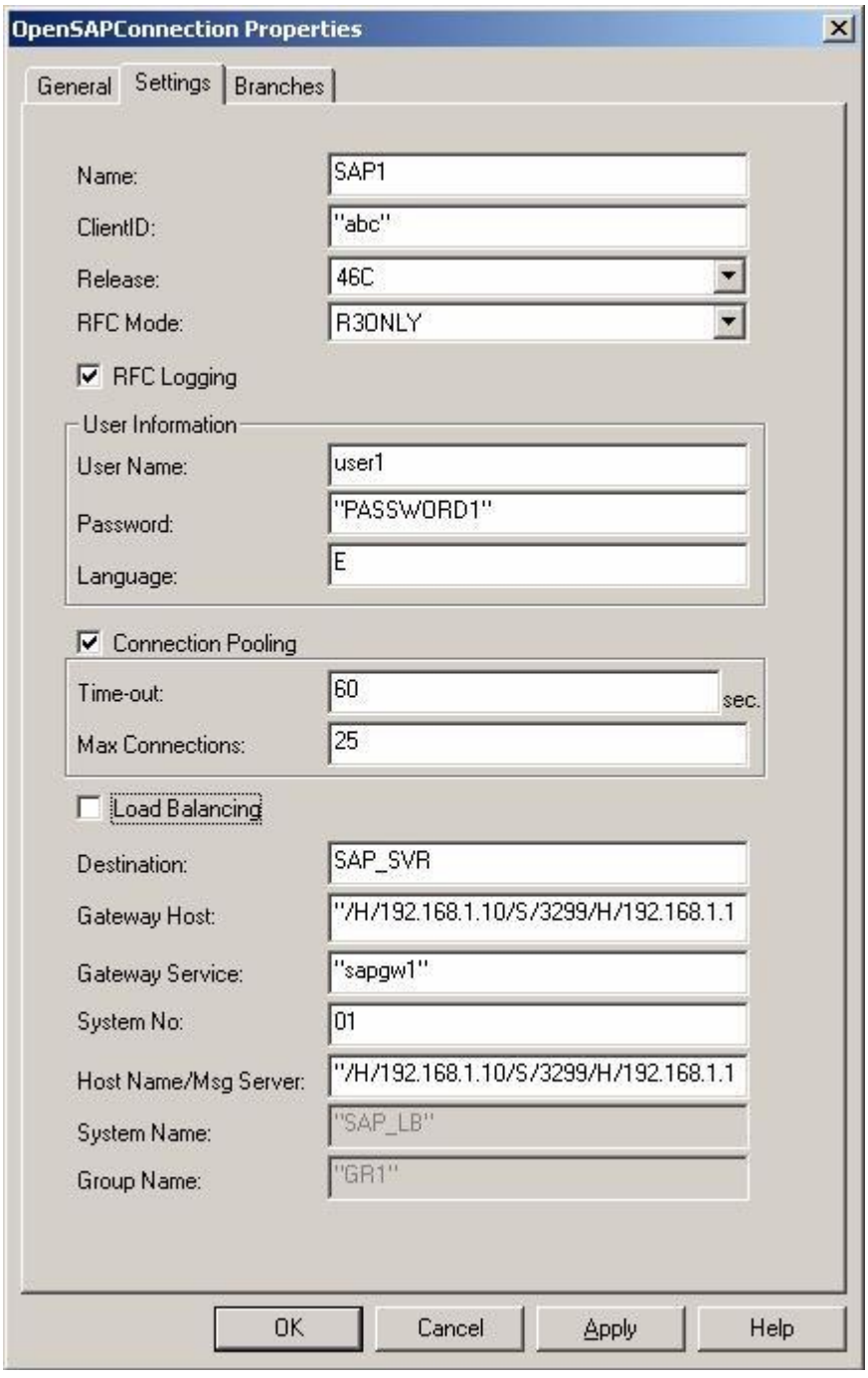


Figure 6: Settings Tab of OpenSAPConnection Properties Diaglogue Box

When connection pooling is enabled, enter the maximum number of connections that are allowed to be created in the connection pool. When this maximum is reached and a new connection is requested, the requested connection will wait the number of seconds entered in the Time-out box to see if an existing connection gets free. If no connection gets freed, the connection will time out according to the time set.

EXECUTE SAP FUNCTION

In the ExecuteSAPFunction block the user configures input and output variables, to bind the resulting parameters based on the types selected, see Figure 7. Return branches are Success, Failure or No Data Found.

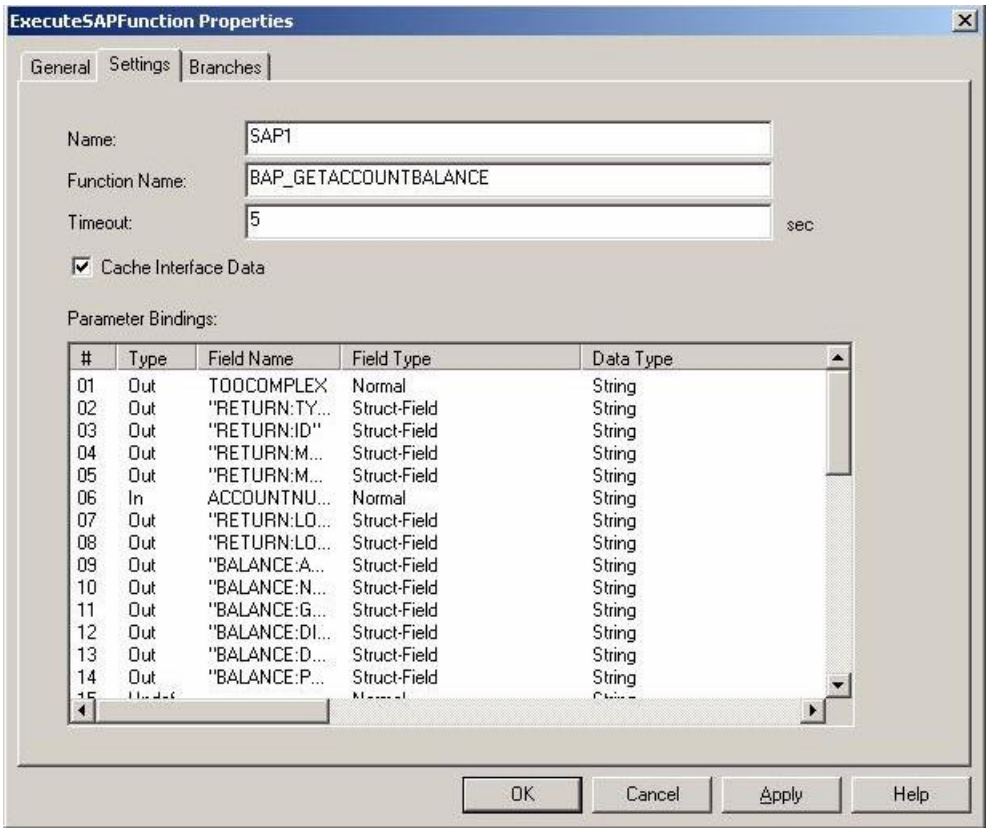


Figure 7: Settings Tab of ExecuteSAPFunction Properties Dialog Box

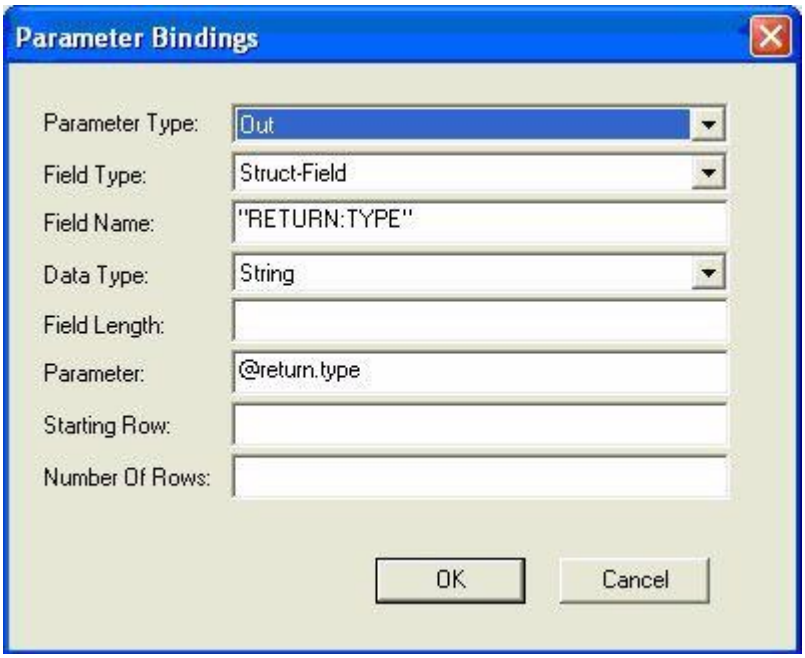
In the **Name** box, enter the name of the SAP system that are connected. The ExecuteSAPFunction waits for response from the SAP Server for the number of seconds entered in the **Timeout** text box. When this time is reached, the function is timed out and closes.

To cache Interface Data or not is optional. This means that table definitions and structure definitions are fetched as part of the execution of a SAP function, and the definitions will be cached for as long as the connection is open. If connection pooling is used, then the interface data is kept in the cache as long as the connection is kept in the pool.

Data can be transferred between an application and an SAP System. Variables are used to store data retrieved from a SAP System or to pass information to a function as parameters. The process of associating variables with parameters of a function is called Binding, and the parameter binding process is described in 4.5.1 Parameter Bindings.

PARAMETER BINDINGS

1. Enter **Name** and **Function Name** of the connected SAP system (see Figure 7).
2. Double-click on the first open **Data Type** to open the **Parameter Binding** dialog box.



3. Select **Parameter Type** from the drop-down list.
4. Select **Field Type** from the drop-down list.
5. Enter **Field Name**. Field names are associated with the function and dependent on the Field Type chosen:
   - Normal – Simple Field
   - Struct-Field – Structure Field (Field name is specified as [Structure Name:Member Name])
   - Table-Field – Table Field (Field name is specified as [Table Name:Column Name])
6. Select **Data Type** from the drop-down list.
7. Enter **Field Length**. Can be either long constant or variable
8. Enter **Parameter** name. Parameter names are dependent on the Parameter Type and Field Type selected:
   - Parameter Type: In, Field Type: Normal or Struct-Field - Parameter name is a Variable or a Constant.
   - Parameter Type: In, Field Type: Table-Field – Parameter name is a Variable or Array Variable (Number of Rows > 1).
   - Parameter Type: Out, Field Type: Normal or Struct-Field – Parameter name is a Variable

- Parameter Type: Out, Field Type: Table-Field – Parameter name is an Array Variable
- Parameter Type and Field Type are Undefined – Used for removing data from the selected parameter and all parameters that follow

9. Click **OK**. The Bind command will bind the variables to the parameters in the order that the parameters are specified in the function.

CLOSE SAP CONNECTION

The CloseSAPConnection block disconnects from the specified SAP system. If the connection was opened as a connection pool, the call or session will be closed back to the connection pool. If the connection was not opened as a shared pool, the RFC will be closed for the given connection name.

There is an option to remove the connection from the pool when closing the connection, see Figure 8. When **Remove from Connection Pool** is checked, the connection will be both removed from the pool and closed towards the SAP system.

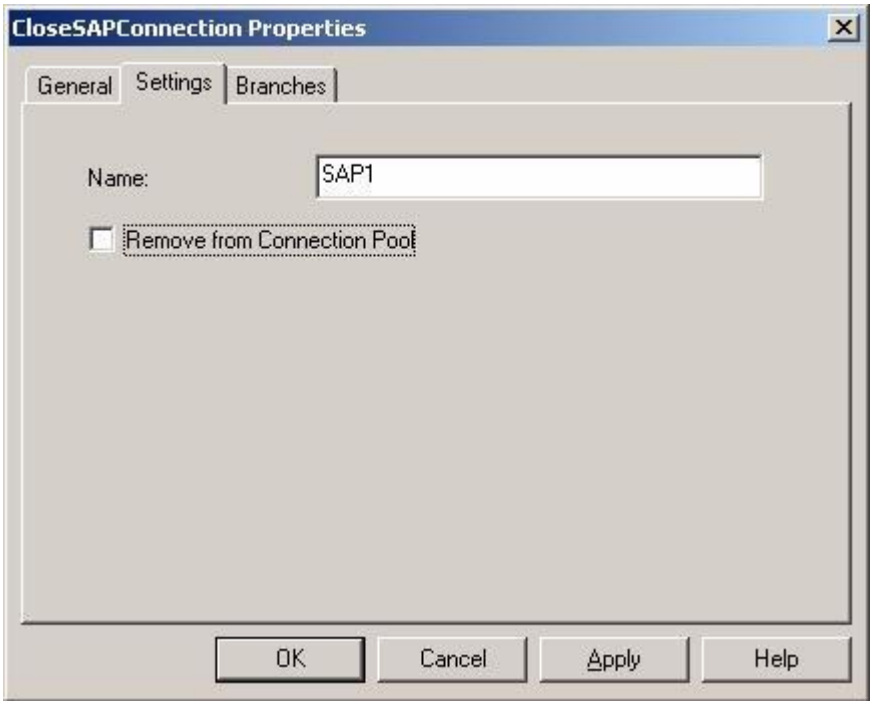


Figure 8: Settings Tab of CloseSAPConnection Properties Dialog Box

Checking Remove from Connection Pool can for example be used when an Execute SAP function times out, to allow the application to close the connection with this flag set before opening the connection again to try to execute the SAP function one more time. It can also be used to remove a faulty connection from the pool.