

SAP Automation (BC-FES-AIT)



HELP.BCFESRFC

Release 4.6C



Copyright

© Copyright 2001 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft[®], WINDOWS[®], NT[®], EXCEL[®], Word[®], PowerPoint[®] and SQL Server[®] are registered trademarks of Microsoft Corporation.

IBM[®], DB2[®], OS/2[®], DB2/6000[®], Parallel Sysplex[®], MVS/ESA[®], RS/6000[®], AIX[®], S/390[®], AS/400[®], OS/390[®], and OS/400[®] are registered trademarks of IBM Corporation.

ORACLE[®] is a registered trademark of ORACLE Corporation.

INFORMIX[®]-OnLine for SAP and Informix[®] Dynamic Server[™] are registered trademarks of Informix Software Incorporated.

UNIX[®], X/Open[®], OSF/1[®], and Motif[®] are registered trademarks of the Open Group.






HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C[®], World Wide Web Consortium, Massachusetts Institute of Technology.

JAVA[®] is a registered trademark of Sun Microsystems, Inc.

JAVASCRIPT[®] is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, SAP Logo, R/2, RIVA, R/3, ABAP, SAP ArchiveLink, SAP Business Workflow, WebFlow, SAP EarlyWatch, BAPI, SAPPHIRE, Management Cockpit, mySAP.com Logo and mySAP.com are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other products mentioned are trademarks or registered trademarks of their respective companies.

Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Inhalt

SAP Automation (BC-FES-AIT)	5
Integrating External Applications with R/3	6
GUI Interfaces to SAP Systems	7
RFC and BAPI Interfaces to SAP Systems	9
Overview of SAP Automation Tools	11
SAP Automation GUI Interfaces	12
SAP Automation RFC and BAPI Interfaces	13
Relationship between SAP Automation Tools	18
Choosing SAP Automation Tools to Work with	19
Choosing an Interfacing Method	20
SAP Automation Tools: Details	22
Class Libraries.....	23

SAP Automation (BC-FES-AIT)

SAP Automation is a suite of tools, components, and class libraries that allow application developers to integrate external applications from the desktop, a server, or a Web server with R/3 systems.

More specifically, the SAP Automation suite of tools enables the programmer to write applications that:

- Connect to R/3 from outside of the SAP GUI interface environment
- Access SAP business objects and Business APIs (BAPIs), their contents or metadata, from outside of R/3 (read and write access)
- Access R/3 RFC function modules, their contents or metadata, from outside of R/3 (read and write access)
- Input data into R/3 in a batch mode through R/3 transactions from outside of R/3
- Access and manipulate R/3 application screens data from outside the SAPGUI
- Capture the interaction between an end user and SAPGUI screens
- Provide an alternative GUI or non-GUI interface to R/3 applications and transactions
- Send or receive inbound or outbound IDoc documents to or from R/3

Integrating External Applications with R/3

There are two major "channels" or interfaces for integrating external applications with R/3:

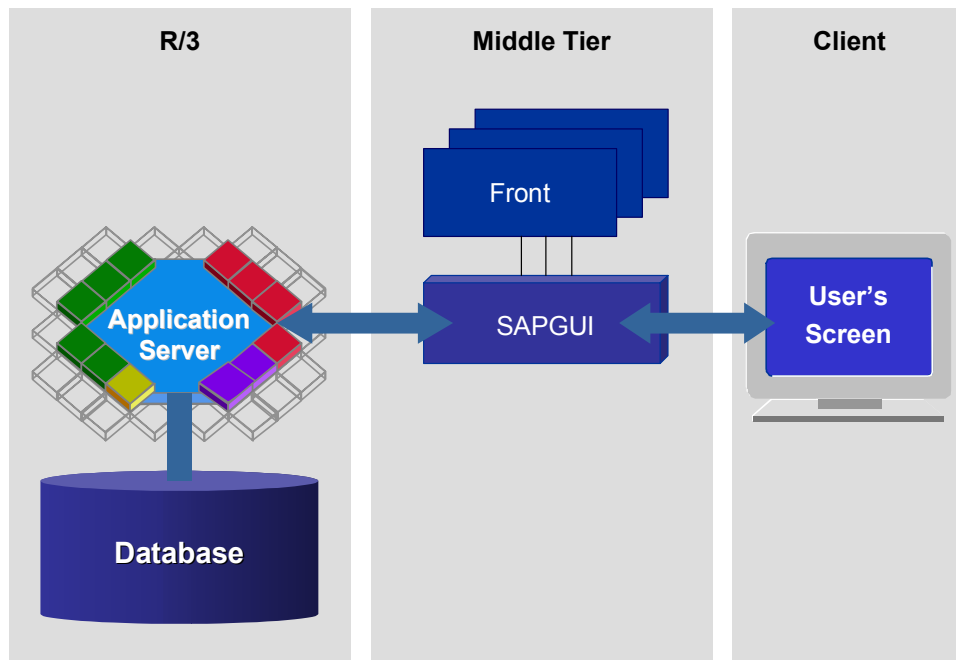
- [GUI interfaces \[Seite 7\]](#)
- [Remote Function Call \(RFC\), which also includes using SAP business objects and Business APIs \(BAPIs\) \[Seite 9\]](#)

The SAP Automation suite includes [tools and components that use the GUI interfaces \[Seite 12\]](#), as well as [tools and components that are based on the RFC interfaces \[Seite 13\]](#).

GUI Interfaces to SAP Systems

The R/3 application server sends and receives data to and from its SAPGUI when displaying R/3 application screens to an end user. The SAPGUI process interacts with one or more Front processes, one for each user session.

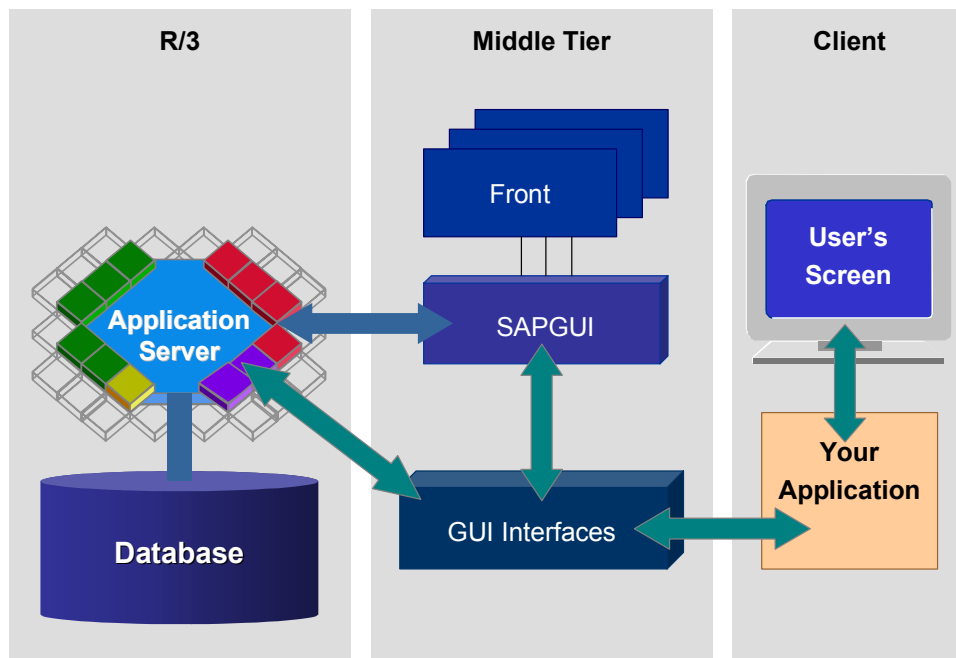
The following diagram shows the standard SAPGUI and its Front processes.



Using the GUI interfaces to R/3, a client program can access the data stream that is communicated between the R/3 application server and its SAPGUI as a method of getting data from or into R/3.

As the following diagram illustrates, the GUI interfaces allow a client program to access R/3 screen data by communicating with the R/3 application server and the SAPGUI.

GUI Interfaces to SAP Systems



By using the GUI interfaces an external program can replace the standard SAPGUI with another user interface, which can be either graphical or non-graphical (for example, it can be voice driven).

Programming using the GUI interfaces also allows an external program to monitor or record an end user interaction with SAPGUI screens.

Using the GUI interfaces method requires no knowledge of ABAP programming. It also eliminates the need to learn the business and transaction logic behind an existing R/3 application for which the external program is providing an alternative user interface.

However, since R/3 transaction screens may change between R/3 versions, using this method of integration with R/3 is the most vulnerable to changes.

RFC and BAPI Interfaces to SAP Systems

RFC is the protocol used by SAP for remote communication, that is, for communications between remote independent systems.

RFC is used for communications between two independent SAP systems, or for communications between an SAP system and a non-SAP system, such as an external application. It can also be used for communications between modules on the same system.

Using the RFC interfaces you can extend the functionality of R/3 applications from an external program.

Compared to using the GUI interfaces, using RFC interfaces requires more knowledge of the business logic of the R/3 applications with which you are integrating the external application.

Types of RFC Interfaces

There are many levels at which external applications can integrate with R/3 through the RFC channel (interface). SAP Automation tools support several of these methods:

- Making **Remote Function Calls (RFCs) directly** to invoke SAP function modules

External programs using RFCs can either make or receive remote function calls to or from an SAP system. Therefore, using RFCs an external program can act as either the client or the server to an SAP system. Conversely, the SAP system can take the role of either the client or the server to the external application.

You can use a combination of SAP Automation tools to incorporate remote function calls into your programming environment. SAP Automation offers several products that make programming RFCs easier. It offers products for programming of RFCs from languages such as C, C++, Java, or COM-compliant languages, such as Visual Basic and applications such as Excel.

SAP Automation also includes tools that help you browse the metadata of RFC function modules, meaning that they allow you to view the syntax of RFC functions, their parameters and fields.

Using RFCs requires knowledge of ABAP programming, because calling a remote function in R/3 is calling an ABAP function. You may also need to extend the existing set of available RFC with your own ABAP code.

- Using **Business APIs (BAPIs) to access and work with SAP business objects**

R/3 systems contain a set of business objects, which are an object-oriented representation of real-world business objects. Examples of business objects are a purchase requisition, a G/L account, or an employee.

An R/3 system also contains Business Application Programming Interfaces (also called Business APIs, or BAPIs), which provide a programming interface to business objects. BAPIs are a set of methods for working with and manipulating SAP business objects.

Internally, BAPIs are implemented as function modules stored in the R/3 Function Builder. Each BAPI has an equivalent internally-named function module. However, you should use the BAPI and not the function module behind it, because the BAPI provides a stable, object-oriented view of R/3 applications and data. The internal function modules or even the fact that the BAPIs are implemented as function modules are not guaranteed to remain the same in future releases of R/3.

The advantage of using BAPIs over using RFCs is that using BAPIs requires no ABAP programming knowledge. You use BAPIs from standard programming languages such as C++ and Java.

RFC and BAPI Interfaces to SAP Systems

Using BAPIs also offers the most stability to your application, because BAPIs stay consistent over different R/3 releases.

However, business objects and their BAPIs are created by SAP, and you can only use BAPIs that were implemented by SAP. If a certain BAPI for performing a certain application task was not implemented yet, you must seek alternative programming methods.

- Using **transactions for batch input**

SAP Automation includes a component for using R/3 transactions to feed data into R/3 in batch input mode.

- Exchanging asynchronous messages with R/3 using **IDoc interface technology**

IDoc interface technology is an SAP standard for sending documents back and forth between different R/3 systems or between an R/3 system and an external application.

IDoc is useful for transferring large amounts of data in asynchronous mode.

The IDoc interface technology is based on transactional RFC, which is a more reliable flavor of RFC in that it ensures that a certain transaction is called exactly once in the RFC server system. It also ensures execution even if the RFC server system is not available at the time that the RFC client makes the call.

An IDoc is a data container that follows a certain SAP standard format. It contains a header and segments of data. SAP defines templates for various IDoc documents; for example, there is a definition for a purchase requisition IDoc.

External programs can use the IDoc method of communicating data to and from an SAP system if an appropriate IDoc template exists. Sometimes it is possible to extend an existing IDoc template. If there is no IDoc template for the type of data you wish to transfer, and there is no IDoc template that you can extend, you must seek another programming method.

Overview of SAP Automation Tools

SAP Automation GUI Interfaces

The following table lists the tools in the SAP Automation suite that use the [GUI channel \[Seite 7\]](#). It describes their function and lists the programming language(s) with which you can use each of the tools:

Tool	Type	Function	Language(s)
GUI Library [Extern] (Document ed as part of the SAP Automation GUI interfaces [Extern])	C API	Like "screen-scraping" programs, the GUI Library takes the contents of R/3 SAPGUI screens and makes them accessible through standard data structures and function calls. Allows you to write programs that provide an alternative user interface to the standard R/3 SAPGUI. This alternative interface can be GUI or non-GUI. Also allows you to write programs that record the interaction between an end user and SAPGUI screens.	C
GUI Component [Extern] (Document ed as part of the SAP Automation GUI interfaces [Extern])	ActiveX Server (both in process and out of process)	Exposes a subset of the GUI Library to COM-compliant programs and applications. Simpler to use but also less flexible than using the GUI Library directly.	Any COM-compliant application
GUI Code Generator [Extern]	Stand-alone executable	A code generator for the GUI Component. Can act as a macro recorder allowing you to generate Visual Basic and other code as you or an end-user work with the SAP system. Also lets you capture SAP screens as Visual Basic forms.	Produces VB, HAHTtalk Basic, or Delphi's Object Pascal language code

Product Documentation

For more details, see the HTML Help documentation of the individual products.

The details of the GUI Library and the GUI component are described in the HTML Help for the [SAP Automation GUI interfaces \[Extern\]](#). The details of the [GUI Code Generator \[Extern\]](#) are discussed in a separate HTML Help topic.

SAP Automation RFC and BAPI Interfaces

The following tables list the SAP Automation tools, components, and libraries that use the [RFC interfaces \[Seite 9\]](#). The tables describe the function of these products and lists the programming language(s) with which you can use each of them.

Use the links from these tables to access the detailed documentation of the various products.

Underlying Library

Tool	Type	Function	Language(s)
RFC API [Extern]	C API	A set of C routines that allows you to write external applications that communicate with an SAP system through remote function calls (RFCs). The external application can act as either a client or a server to the SAP system.	C

SAP Automation RFC-based Class Libraries

Tool	Type	Function	Language(s)
RFC C++ Class Library [Extern]	Class library	Provides a framework for the RFC API. Makes it easier to use RFCs in C++ programs by providing pre-built functions for some common tasks, such as establishing a connection to R/3.	C++
Java RFC Class Library [Extern]	Class library	Provides an object oriented Java view over the RFC API. Makes it easier to use RFCs in Java programs by providing pre-built functions for some common tasks, such as establishing a connection to R/3, or performing data type conversion.	Java
IDoc C++ Class Library [Extern]	Class library	Provides a framework for processing inbound and outbound IDocs. Makes it easier to use IDoc technology in C++ programs by providing pre-built functions for some common tasks, such as creating and appending IDoc segments, accessing segments and fields by names, and working with fields.	C++

ActiveX Controls and Components

Tool	Type	Function	Language(s)
------	------	----------	-------------

SAP Automation RFC and BAPI Interfaces

Logon Control [Extern] (Documented with the SAP Automation RFC and BAPI Interfaces [Extern])	ActiveX Control (OCX)	Allows you to create a connection object with which you can establish a connection to R/3. Note that for programs using the SAP DCOM Connector, we recommend that you use the DCOM Connector Logon Component [Extern] instead of the Logon Control.	Any COM-compliant application
Function Control [Extern] (Documented with the SAP Automation RFC and BAPI Interfaces [Extern])	COM Server	Provides COM objects for handling RFC functions and their parameters. This allows you to create external applications that call RFC function modules in an SAP system.	Any COM-compliant application
Table Control [Extern] (Also called Table Factory Control. Documented with the SAP Automation RFC and BAPI Interfaces [Extern])	COM Server	Allows you to create a Table object representing a table parameter of an RFC function. Works with the Function control to manage tables attached to Function objects.	Any COM-compliant application
Transaction Component (Also called Transaction Control [Extern] , which is documented with the SAP Automation RFC and BAPI Interfaces [Extern])	COM Server	Exposes R/3 batch input capability to COM-compliant programs and applications. Allows an external program to send input field values to an R/3 screen (output field values are not returned). Makes using batch input easier by eliminating the need to populate the fields of the BDC table, which is a prerequisite to using the standard R/3 batch input method.	Any COM-compliant application

SAP Automation RFC and BAPI Interfaces

BAPI ActiveX Control [Extern]	ActiveX Control (OCX)	Allows you to develop external client/server applications (with R/3 as a server) that access business functions in the SAP system by calling BAPIs (Business APIs) through OLE Automation. Achieves this by allowing you to create (on the client) local instances of business objects, which act as proxy objects for the business object in the SAP system.	Any COM-compliant application
--	-----------------------	--	-------------------------------

DCOM Connector-based Components and Tools

Tool	Type	Function	Language(s)
IDoc Connector for XML [Extern]	COM Server	Provides an XML interface for working with IDoc documents. The <i>IDoc Connector for XML</i> component accepts your XML documents and sends them as inbound IDocs to a specified R/3 system. Conversely, it can receive outbound IDocs and it translate them into XML documents for your application to use.	Any COM/DCOM-compliant application
DCOM Connector Logon Component [Extern]	COM Server	Helps programs that use the SAP DCOM Connector handle the connection parameters of the COM objects created by the SAP DCOM Connector. The DCOM Connector Logon Component provides a logon dialog with which you can get the necessary connection parameters from an end user. It also allows the end user to create and maintain destination definitions. The DCOM Connector Logon Component also allows you to easily copy connection parameters into a DCOM Connector COM object.	Any COM/DCOM-compliant application

SAP Automation RFC and BAPI Interfaces

Repository Services [Extern]	COM Server	<p>Provides read access to the metadata of business objects, RFC function modules, tables and structures in an R/3 system to COM-compliant programs and applications.</p> <p>Also allows you to save a copy of the metadata in a local database. Using a local database allows you to access the same metadata faster and without connecting to R/3.</p>	Any COM/DCOM-compliant application
BAPI Gateway [Extern]	COM Server	<p>Allows you to dynamically call BAPIs and RFCs through the DCOM Connector, that is, it allows you to determine at run time which BAPIs or RFCs you call.</p> <p>Eliminates the need to use the SAP DCOM Connector wizard for creating a DCOM Connector proxy component.</p> <p>You can use the SAP Automation Repository Services component [Extern] in conjunction with the BAPI Gateway to obtain the metadata for the BAPIs or RFCs you wish to call at run time.</p>	Any COM/DCOM-compliant application
<p>Repository Browser (Also called the SAP Browser Control [Extern], which is documented with the SAP Automation RFC and BAPI Interfaces [Extern])</p>	ActiveX Control (OCX)	<p>A control that can be hosted by any ActiveX container. It consists of a window with two panes for browsing SAP BAPI and RFC metadata information.</p> <p>Allows online calling of RFC functions from within the control.</p> <p>Also exposes several methods to enable the container application to control and automate metadata browsing.</p> <p>Allows you to export metadata information to an Excel spreadsheet.</p>	Any COM/DCOM-compliant application

SAP Automation RFC and BAPI Interfaces

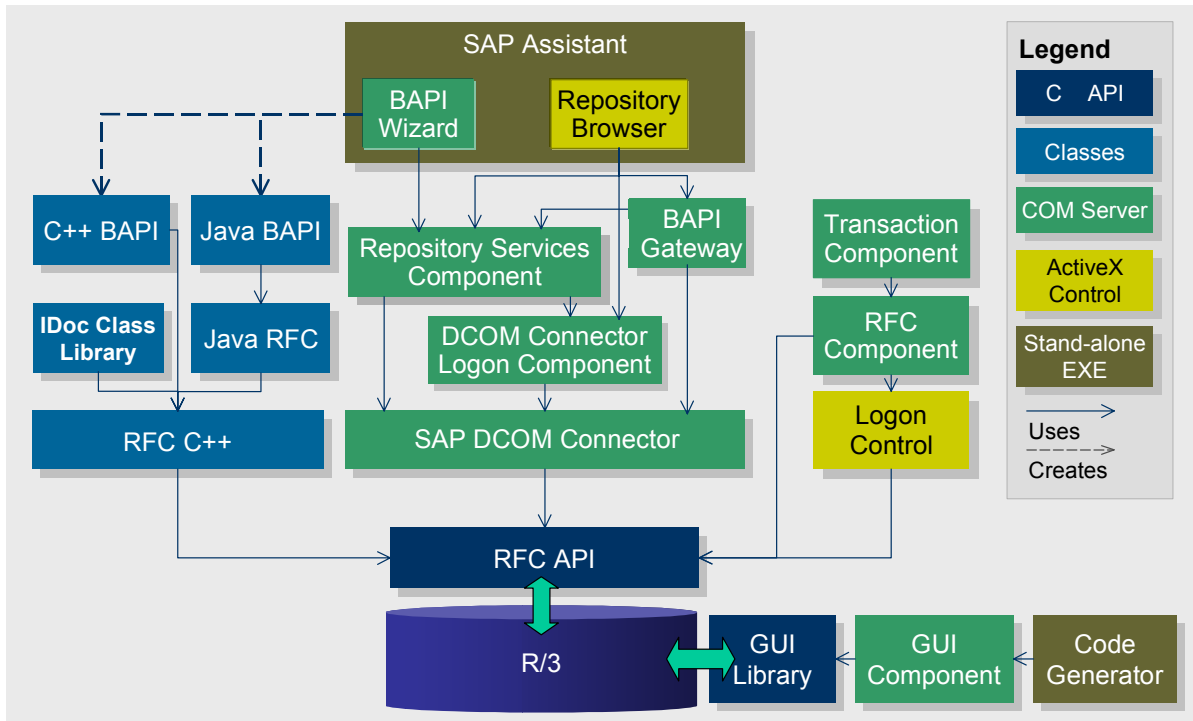
<p>SAP Assistant [Extern] (documented with the SAP Automation RFC and BAPI Interfaces [Extern])</p>	<p>Stand-alone executable</p>	<p>A tool for browsing (viewing and searching) the metadata of SAP RFC functions and SAP business objects and their methods (BAPIs). Uses the SAP Browser Control (see above). Allows you to call RFC functions directly online. Also allows you to generate either Java or C++ classes for working with SAP business objects and their BAPIs.</p>	<p>You can use the code generated with the BAPI Wizard tool for Java or C++ programming. The SAP Assistant window itself is not programmable.</p>
---	-------------------------------	--	---

Relationship between SAP Automation Tools

Relationship between SAP Automation Tools

Some of the SAP Automation tools use other components of the SAP Automation suite.

The following diagram illustrates the relationship between the different SAP Automation components. The diagram shows the various levels of access support the tools offer.



Choosing SAP Automation Tools to Work with

Use

The tools that compose the SAP Automation suite perform a variety of functions, offer different level of access support, and are usable with various programming languages and on different platforms.

The functionality of the SAP Automation tools overlaps in some cases.

As a developer of applications that integrate an external with R/3, you can use one or more of the SAP Automation components to best fit the needs of your application.

Procedure

1. When deciding which tool or tools to use with your application, it is important to first [decide which method of interfacing to R/3 is most appropriate for your application \[Seite 20\]](#). This decision determines which of the SAP Automation tools are relevant to your application.
2. Once you have decided which interfacing method to use, choose an SAP Automation tool based on the programming language and platform it supports. See the tables listing the various tools in the topics [SAP Automation RFC and BAPI Interfaces \[Seite 13\]](#), and [SAP Automation GUI Interfaces \[Seite 12\]](#).

Choosing an Interfacing Method

Choosing an Interfacing Method

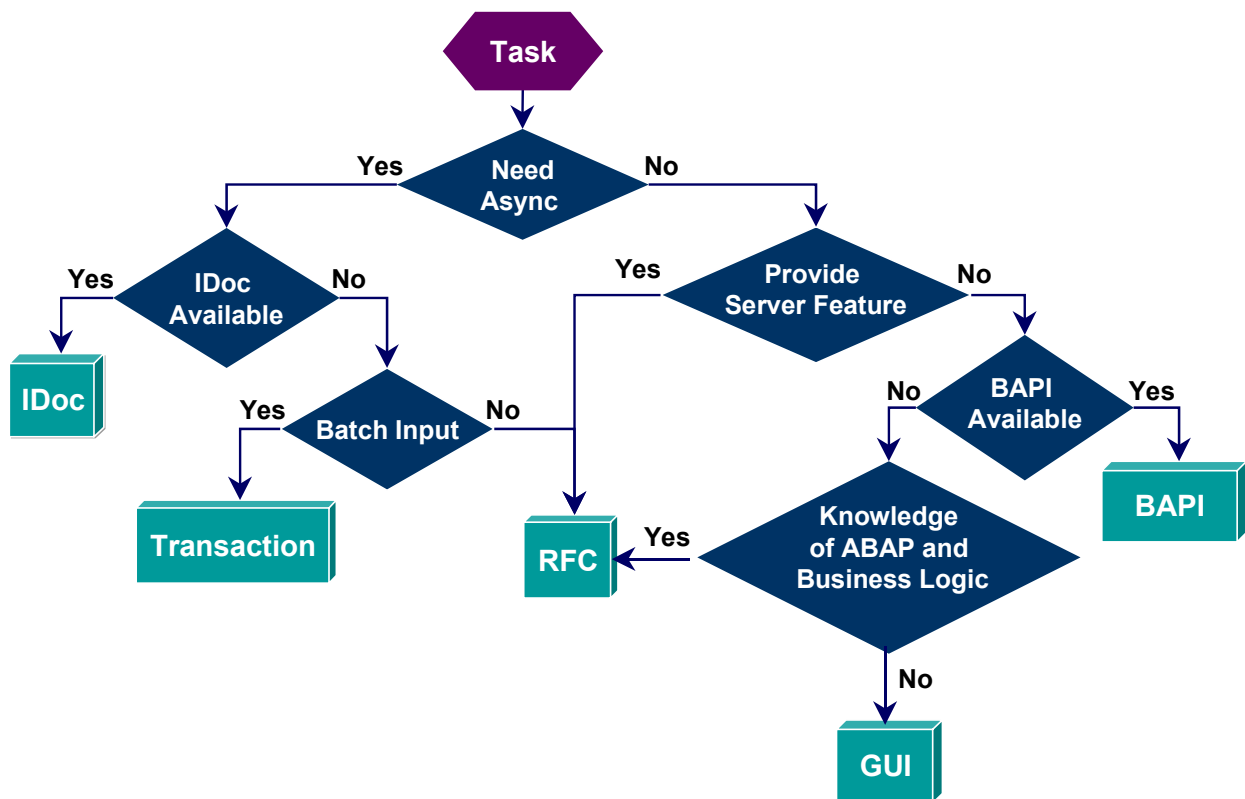
Use

Sometimes choosing the appropriate SAP Automation tool is an easy choice, in that the nature of the application clearly dictates the method for interfacing to R/3. For example, if your application needs to record end-user interaction with R/3 SAPGUI screens, then using the SAP Automation GUI Interfaces is the most appropriate choice.

However, when implementing other types of application that need to integrate with R/3, the choice may not be as clear.

Procedure

The following flow diagram describes how to choose the most appropriate SAP Automation tool for your application needs.



As summarized by the above diagram, consider the tasks you must achieve in your application when choosing the method of interfacing to R/3:

If your application needs to be sending or receiving data to or from R/3 in asynchronous mode, meaning that it should not wait for the R/3 response before continuing tasks, choose either the IDoc, the Transaction, or the RFC methods.

If you need synchronous access to R/3 data and applications, use either the BAPI, the GUI or the RFC methods.

Notice that using RFCs you can implement either asynchronous or synchronous access.

- If you need **synchronous** access, then consider whether your application needs to act as a server to R/3 requests. If you need to provide such server capability, then write your application using RFCs.
 - If your application always acts as a client to R/3, in that it always initiates requests to R/3 and never needs to respond to requests initiated by R/3, then you can also use Business APIs (BAPIs) or the GUI interface method.
 - Using BAPIs is preferred, because it is not as dependent on changes in the GUI screens between R/3 versions. However, you are limited to the set of BAPIs that were implemented in R/3. If no BAPI with the desired functionality exists, then you must use other interface methods: either GUI or RFCs:
 - Using the GUI interfaces you do not need to know any ABAP programming, and you do not need to know the business logic involved in the R/3 application you are integrating with. Your application uses existing R/3 transaction screens to get data in and out of R/3. With or without displaying the standard SAPGUI to your end-user, your application "scrapes" the data communicated between R/3 and its SAPGUI to provide an alternative interface to your end-user.
 - Using RFCs requires some knowledge of ABAP programming, if you need to complement the set of existing RFCs with your own RFCs. Even if you do not need to write additional RFCs, you do need to understand the business logic of the R/3 application you are interacting with to program with RFCs, because with RFCs you are adding, changing, or replacing the functionality provided by an R/3 transaction or application component.
- If you need **asynchronous** access to R/3, then you can either use the IDocs or the Transaction methods. As mentioned above, you can also use RFCs, because with it you can implement either asynchronous or synchronous applications.
 - If you are sending data between two R/3 systems, or between an R/3 system and another system, you may be able to use the IDoc interface, if there are IDoc templates defined for the functionality you need.
 - If there are no IDoc templates for the functionality you need, then you can use the Transaction Component, if all you need is a batch input of data into the R/3 system.
 - If batch input of data is not sufficient, in that you need interactive access or you need to read data in addition to writing data into R/3, then the method of choice is programming RFCs.

SAP Automation Tools: Details

Class Libraries