Book *1*

The mySAP Revolution

U:\Book\Book_01.doc

The mySAP Revolution

Fehler! Es wurden keine Einträge für das Inhaltsverzeichnis gefunden.

1   The Philosophy Behind the mySAP Framework

Many a people say that mySAP had been a marketing campaign to keep pace with the internet revolution and new economy. Whether this was the case or not: mySAP meanwhile stands for a new philosophy of collaborative and distributed computing, that allows to integrate developments written on different platforms and in different languages into a single system.

1.1 Components Of The mySAP Framework

20

1.2 Best of Breed vs. All-In-One Software

1.3 The Global Picture

Figure 1:   The Global Picture

Client Server With R/3

- We are dealing with client server programming
- The goal is to use the best-of-breed software applications and let them cooperate
- A server waits for requests by a client
- A server renders services to a client
- A client sends requests to a server and waits for response
- SAP R/3 can act as a server or as a client for non-R/3 systems
- Communication takes place via the TCP/IP protocol
- R/3 can call ActiveX components via the Microsoft DCOM+ protocol
- R/3 can call JAVA Beans through the CORBA protocol
- ActiveX DCOM+ clients can call R/3 using SAP provided interface DLL libraries
- JAVA can call R/3 via SAP provided CORBA libraries
- When a program calls another component on a distant computer it is called Remote Program Call (RPM)
- Remote program calls in R/3 are called Remote Function Calls (RFC)

1.4 ITS®, DCOM® Connector or RFC-Server?

SAP provides different strategic approaches for web development with R/3 access, namely ITS, DCOM and Java connector and the RFC Server object. However, all these utilities are based on the same RFC interface.

The RFC Libraries

Every R/3 access from external uses RFC	The common communication method for R/3 and non-R/3 applications is the RFC – Remote Function Call – protocol, which is mainly a variation of the IBM CPIC protocol. There are libraries provided by SAP for different operating system platforms – especially Windows and UNIX – to be used by non R/3 applications to call R/3 functions. All the popular access technologies like ITS, DCOM or Java Connector are shell applications and access themselves the low level RFC libraries.

DCOM Access through Windows©

For Windows there is a DCOM low level socket and convenient ActiveX components	Windows access all external components through its DCOM component architecture. In Windows the protocol socket is implemented in the librfc32.dll library and a number of more convenient higher level application interface libraries (API) are available as well. Especially the wdtlog.ocx and wdtfuncs.ocx libraries make it very easy to call RFC through Windows applications.

DCOM Connector - Proxy Application

DCOM connector generates Active-X (DCOM) proxy stubs	The DCOM connector is basically a program code generator. It is a set of interface routines that allow communication between R/3 and external applications using the ANSI DCOM (Distributed Common Object Model) specifications. This is a widely used protocol that defines how a computer can access and call applications and application libraries on a remote computer. The DCOM connector generates C++ interface code which mirrors the interface structure of chosen RFC function modules.
Active-X stubs are proxy objects on the local server based on the RFC interface during development	The DCOM connector is a proxy generator. It will download the object specifications from R/3 and generate locally a DLL or OCX module. Your web server application will access R/3 exclusively through these generated proxy DLLs. The interface specifications are determined once during development and compiled into the proxy object. This is contrary to the RFC server object which retrieves the interface specification every time a connection is established.
Proxy objects allow fast connection to server services but require generally a restart of the server if the proxy changes	Using a proxy application shim allows very fast access to the remote computer, especially if a large number of objects are to be accessed. Other than the RFC server objects the object definitions are loaded once during development time and compiled into your project. The disadvantage may be that a change in the interface structure of the remote R/3 object, e.g. the interface of a function module will ultimately require regenerating the proxy DLL object library.

ITS - Internet Transaction Server

ITS is a broker application from the very beginning of SAP internet activities	The ITS has been the early approach to give a full access to R/3 through a web browser. ITS is a typical broker application. It is a program that stocks templates for each potential R/3 transaction dynpro screen. Whenever a browser client requests an R/3 transaction it looks up the template that corresponds to the R/3 dynpro, requests the dynpro data from R/3 and merges template and data into a valid HTML or XML page. This page is then sent to the browser.
ITS emulates a SAPGUI and allows full access to all R/3 functionality which is also the security hole	The advantage is clear: you can easily access all R/3 transaction from an internet or intranet connection, as long as there is a template for all of the transaction's dynpros. The disadvantage is a potential security hole. The ITS gives full access to all R/3 transactions, thus relying fully on the SAP authorization mechanism.

Your system is fully exposed to the public. While this may be acceptable for intranets and extranets, it is unwise to choose this solution for internet based applications.

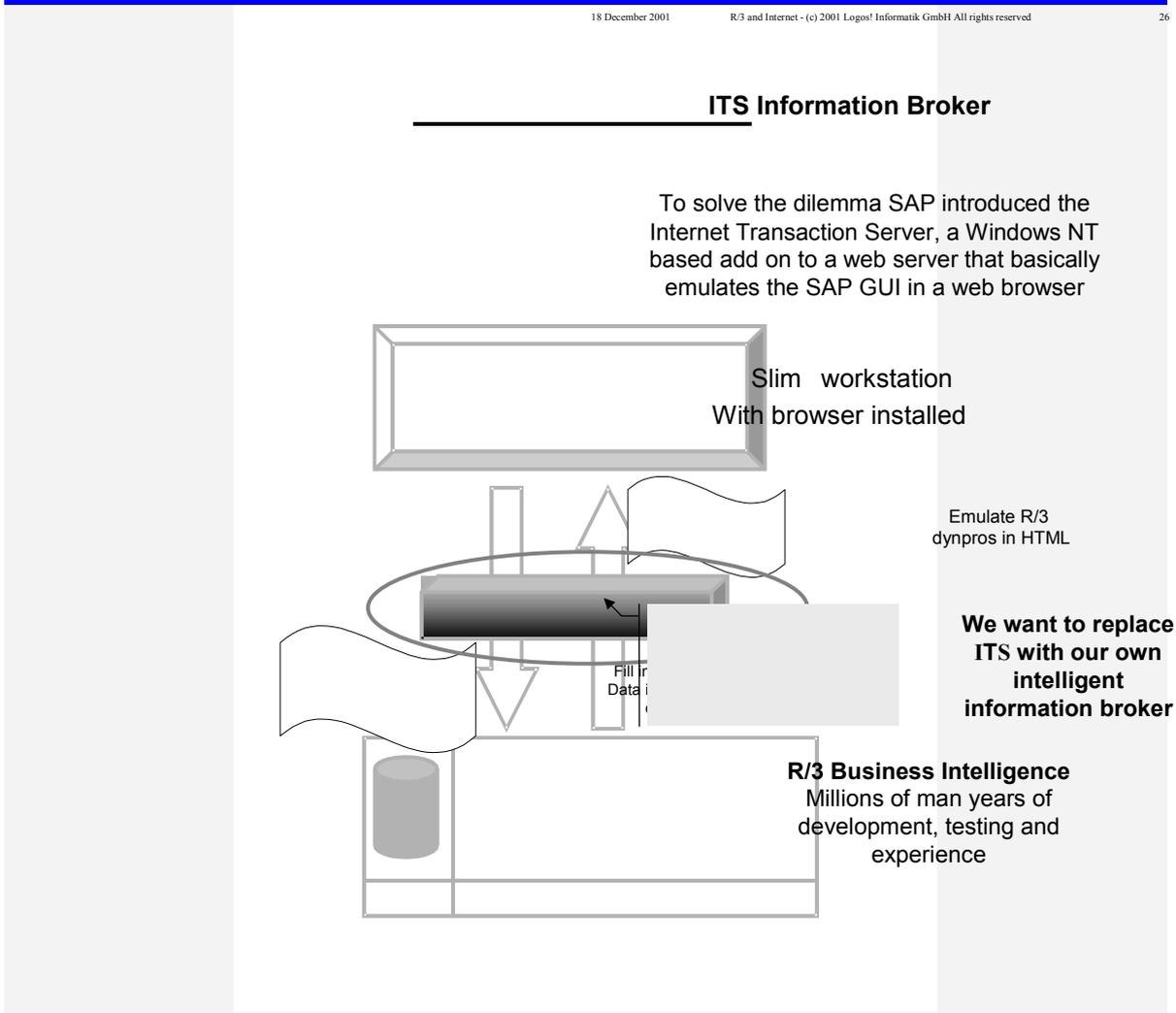**Mostly useful if you want to give full access to R/3 without installing SAPGUI**

It is interesting only, if you want to give free access to your R/3 system, while you are unable to furnish them with a proper SAPGUI. Keep in mind also, that you would have to define one template for each dynpro. SAP provides templates for all standard dynpros as of release 4.6A.

ITS is too inflexible and no longer state of the art

75

I do not see much use for the ITS anymore. I consider it to be an obsolete technology, which does not provide benefits over the DCOM and active server approach. It is only interesting for people who clearly decide, that they want to open and expose their system to the public, e.g. in an EXTRANET, where you cannot install the SAPGUI on the individual workstations. It may also be useful for those of you who have great experience in R/3 but little know-how in web application design.

Figure 2: Concept of the ITS, the standard web application middleware that has been designed to build HTML applications to access SAP R/3 transactions via RFC access (uses DCOM)

18 December 2001   R/3 and Internet - (c) 2001 Logos! Informatik GmbH All rights reserved   26

ITS Information Broker

To solve the dilemma SAP introduced the Internet Transaction Server, a Windows NT based add on to a web server that basically emulates the SAP GUI in a web browser

Slim   workstation
With browser installed

Emulate R/3 dynpros in HTML

We want to replace ITS with our own intelligent information broker

Fill in
Data in

R/3 Business Intelligence
Millions of man years of development, testing and experience

80

DCOM Usage and Communication Protocols

A DCOM component is the handling agent between R/3 and the non-R/3 client application

The DCOM layer is the handling agent between R/3 Remote Function Calls and the applications running on a COM compatible platform (which in effect is a Windows platform). All non R/3 applications connect through a DCOM proxy layer to R/3. The SAP Logon ActiveX component and SAP Function ActiveX

85

component which we use later in this publication are such DCOM components. They are very flexible; however you can create your own DCOM components (DLLs) with the DCOM connector tool that comes with the SAPGUI RFC software development kit.

There are RFC sockets for other OS as well, e.g. UNIX and AS/400

If you develop clients for a non DCOM compatible environment, you have to replace the DCOM components with other object standards. For UNIX there is the internationally accepted CORBA object standard which does basically the same thing as DCOM, COM or COM+. SAP R/3 supports CORBA for UNIX platforms. We do not program for UNIX so we leave this to the UNIX Gurus to exploit further. However, in theory you may simply replace the word DCOM with CORBA in this book.

95

Table 1:   The role of the DCOM tier in an R/3 communication

Communication Tiers

⇨
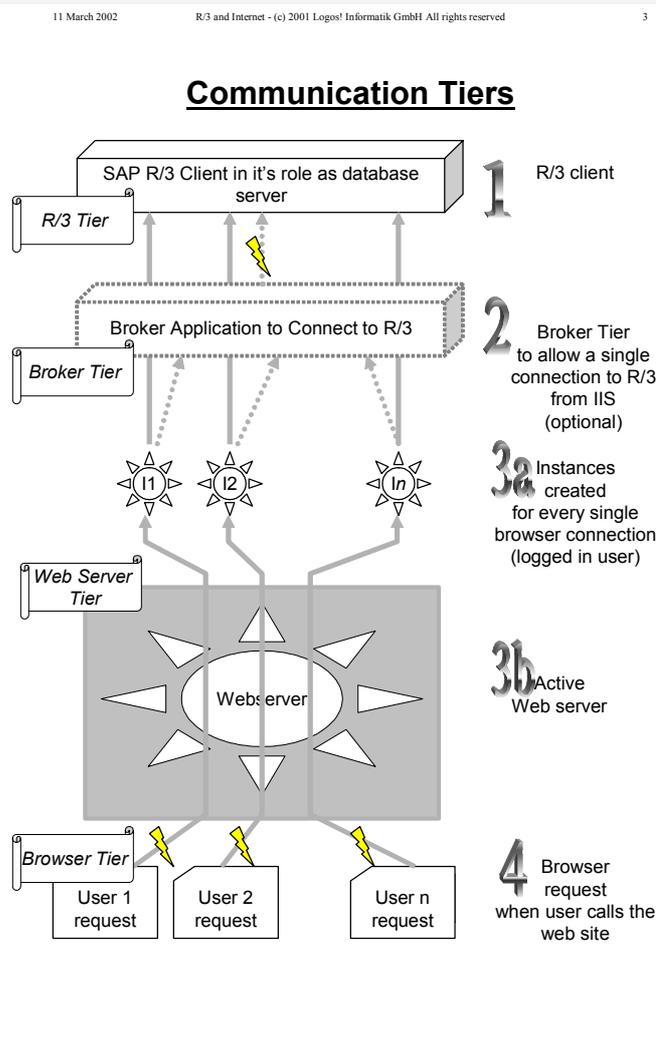
1.5 Details on ITS Internet Transaction Server

The Internet Transaction Server is primarily a GUI replacement that takes over the role of the SAP GUI, when R/3 is accessed via the internet.

100

ITS is a SAP provided web server application that serves the dialogue between SAP R/3 and a web browser. The ITS was designed to provide R/3 access via the internet for a selected range of users and application, mainly with e-commerce in mind.

R/3 cannot communicate directly with HTTP servers and makes use of ITS as an information broker
105

Although both R/3 and internet applications are basically client-server applications, they adhere to completely different communication standards. The communication standard between an internet browser and an internet server is HTTP, which accepts e.g. HTML, JavaScript or CGI. R/3 developed a proprietary standard, which is called the SAPGUI. The SAPGUI works technically the same way as an internet browser like IE or Netscape does, but the used language differs from HTTP supported standards. The Internet Transaction Server is an information broker to access SAP as an information server from a web site. Search engines work the same way. The user enters a request and the search engine looks up the result up in a database to be transformed into a HTML page.

110

ITS translates HTML into RFC-Calls and vice versa

ITS is a relatively simple piece of software, which does principally the following actions repeatedly:

115

Send a HTML form to the browser
Receive the filled form
Extract the data from the form
Call the appropriate transaction in R/3
Capture the transaction result screen
Transform the result into an HTML page
Send the HTML page to the browser

ITS is a web server application like a CGI application

125

Technically, ITS is a web server application that places itself as an adaptor (or "converter") between the web server object and R/3. It is therefore a dedicated web server application, like you find behind most CGI scripts. The major drawback of ITS is, that it is not a programming language, as e.g. ASP (Active Server Pages), Java, JavaScript Pages (JSP) or Perl. It merely transcribes SAP transaction into HTML code. ITS is the interpreter between the R/3 application and the browser transcribes R/3 transactions and does not provide own intelligence. In order to bring individual logic into the web communication, you would have to write a transaction in ABAP, thus putting all intelligence into R/3. Programming on the web server end is not inherently supported.

Server based RFC is the alternative

135

There are other ways to achieve the same results. A straight-forward approach would be to create a server application in any object oriented language which is able to handle RFC. If the web server is a Windows NT system, this is true for nearly every modern Windows programming language which can call DLLs. I see many advantages in writing the web application with Borland Delphi Connect or Java, which calls appropriate BAPIs in R/3. The advantage is a high degree of flexibility for the server application. The only real disadvantage of the RFC approach is, that you cannot use the SAP security concept effectively. However, a professionally set up proxy would be more than sufficient for this task.

ITS has been an early eCommerce solution

145

When the ITS was being developed, this was the time when SAP was looking for an e-commerce approach. The idea was, to have the means to prepare web sites easily with instant real-time access to the R/3 data. The original target for the application was companies with a range of remote key customers or affiliates, who should be able to enter orders directly into the system. Instead of offline recording of orders, which have to be entered into R/3 as IDocs, the data should be entered directly into R/3. The advantages are evident: real-time access, online availability check and confirmation etc. Today, there are simply more powerful and more flexible solutions to bring R/3 to the web.

2   The Global Picture

Now stand back for a while and see the objects in our client server interact and understand which role they play to make the whole system function.

2.1 What You Will In the Global Theatre

2.1 What You Will In the Global Theatre

2.2   Thinking in Messages and Objects

150

2.3   The Importance of Components

2.3   The Importance of Components