Capgemini

AB1011 - BAPI

BAPI - Business Application and programming Interface

Lesson Objectives

On Completing this course, participants will be able to

- Understand about Business Framework
- Understand of Business Objects & BAPI
- Understand the advantages & characteristics of BAPI
- Work with standard BAPI
- Work with custom BAPI
- Work with Interface to R/3 System
- Work with BAPI Uploads and BAPI Reporting

Contents

- Business Framework
- BAPI Introduction
- BAPI Function Modules
- BOR - Business Object Repository
- BAPI Development in BOR
- Custom BAPI Creation
- Data uploading by suing BAPI Functions
- Reporting through BAPI

Business Framework

Business Framework

The SAP R/3 Business Framework provides a structure of R/3 functionality based on application components (business components) and object models.

The Business Framework enables customers and partners to link their own components to the R/3 system.

The basic components of SAP R/3 Business Framework are:

- Business Components
- Business Objects
- Business Application Programming Interface (BAPI)
- Integration Service, Application Link Enabling (ALE)
- Communication Services

Basic Components of Business Framework

Business Components: SAP Business Component provide autonomous business functions and consist of business objects.

Business Objects: Business Objects encapsulate business data and functionality and define the functional scope and boundaries of a Business Component.

Business Application Programming Interface: BAPI along with business objects define and document the interface standard at the business level.

Basic Components of Business Framework

Application Link Enabling: The ALE Integration service enables the integration of business processes that are carried out in different R/3 and non-sap systems.

Communication Services: These are communication technologies like Remote function call (RFC) ,which use business framework to access BAPI.

Business Objects

Business Objects

Business Objects encapsulate business data and functionality and define the functional scope and boundaries of a Business Component.

SAP Business Objects and their BAPI provides an object oriented view of R/3 Business functionality.

SAP has introduced object oriented technologies to make the R/3 process and data is available in the form of business objects.

Business objects are used to break the SAP system down into smaller ,disjunctive units, improving its structure and reducing its complexity.

Business Objects

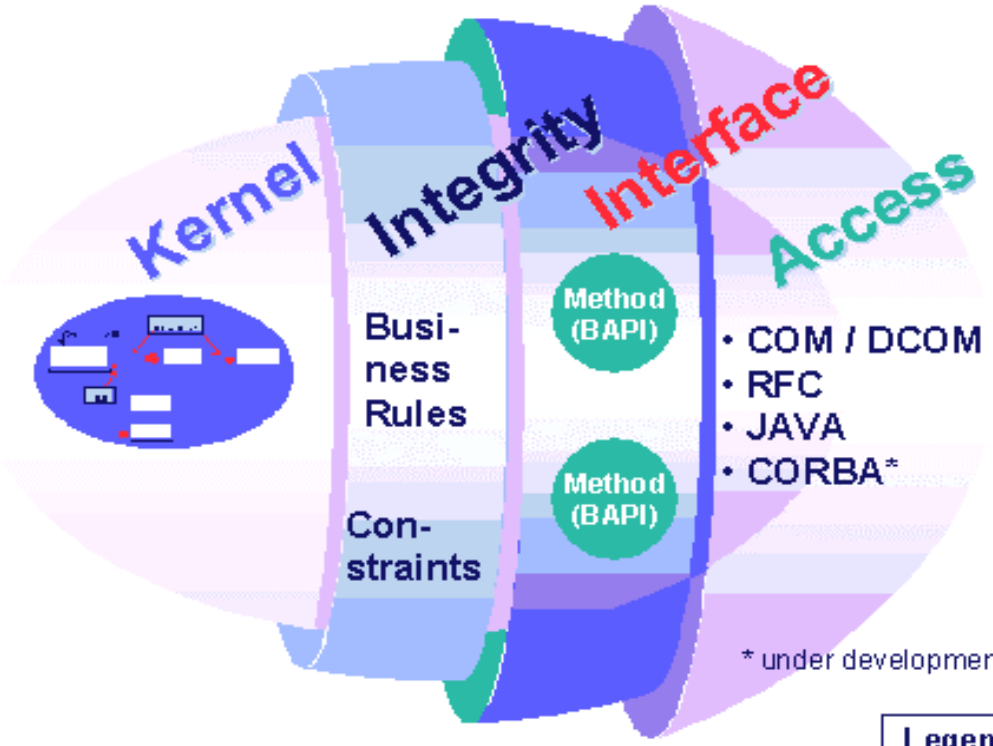
SAP Business Objects encapsulate R/3 data and business processes, thus hiding the details of the structure and implementation of the underlying data.

The set of methods (BAPI) that is associated with a business object represents the object behavior.

Object types : Each individual business object belongs to a specific object class, depending on nature and general characteristics of the object. These object classes are called object types.

Object Instance: Each individual SAP Business object is a representation, or instance of its object type.

Business Objects



Legend:
COM/DCOM = Component Object Model / Distributed Component Object Model
RFC = Remote Function Call
CORBA = Common Object Request Broker Architecture

SAP Business Object Types

Object Type : The object type describes the features common to all instances of that object type.

Key Fields : The key fields determine the structure of an identifying key, which allows an application to access a a specific instance of the object type.

Method : A method is an operation that can be performed on a business object and that provides access to the object data.

Attributes : An attribute contains data about a business object, thus describing a particular object property.

Events: An event indicates the occurrence of a status change of a business object.

BOR (Business Object Repository)

Business Object Repository.

Transaction code for BOR is SWO1.(O not Zero)

It is the central access point for the SAP business object types and their BAPI.

BOR allows an object-oriented view of all data and processes in an SAP system.

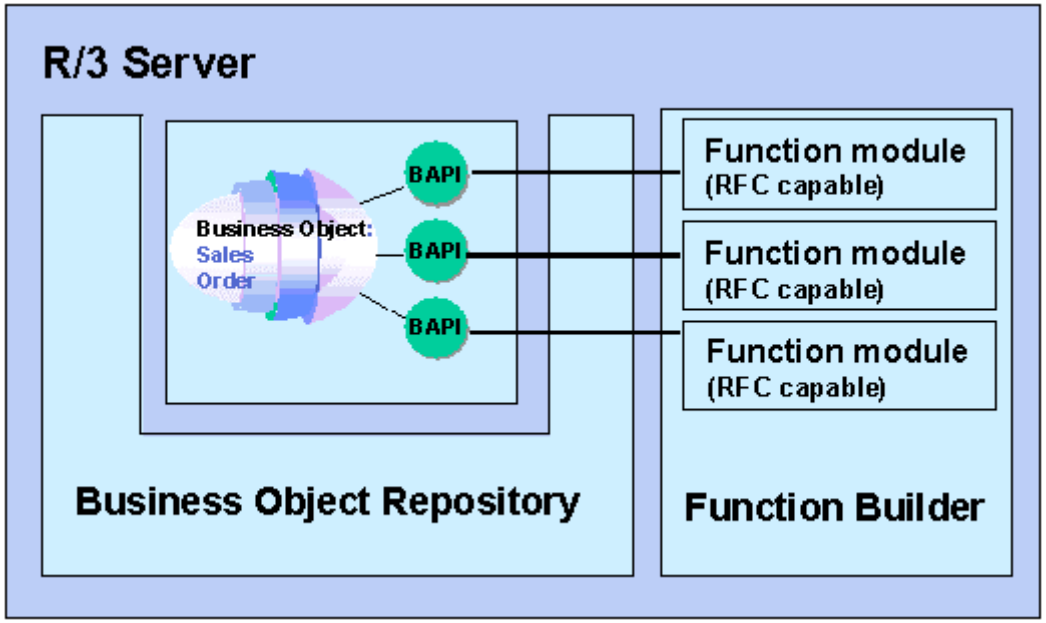
BOR (Business Object Repository)

This enables to find the required functions easily:

- Storage of all relevant information
- It ensures BAPI interface stability
- It manages BAPI in release updates
- It creates instances of SAP business objects

The SAP Business objects held in the business object repository (BOR) encapsulate their data and processes

Business Object Type with BAPIs and Associated Function Modules



Business Application Programming Interface (BAPI)

BAPI

Business Application Programming Interfaces (BAPI) are standard SAP interfaces.

The transaction code for BAPI is BAPI .A BAPI is implemented as an RFC enabled function module.

BAPI are defined as methods for the business object types.

They play an important role in the technical integration and in the exchange of business data between SAP components & between SAP & non SAP components.

BAPI ‘s allow integration at the business level, not the technical level.

BAPI

External Applications can access SAP Business objects through standardized, platform independent interfaces called BAPI.

External access to the data and processes is only possible by means of specific methods – BAPI (Business Application Programming Interface).

Definition: A BAPI is a well defined Interface to process the data of a Business Application System, Implemented as a method of an object in Business Object Repository (BOR).

BAPI

The BAPIs in the R/3 system are currently implemented as Function Modules, which are created and managed in the Function Builder (SE37).

The function modules within a BAPI should:

- Supports the Remote Function Call (RFC) protocol.

- Has been assigned as a method to an SAP Business Object in the BOR.

- Is proceeded without returning any screen dialogs to calling application.

Advantages and Benefits of BAPI

Advantages of BAPI

- Business Standard

- Easy to use and Understood

- Object Oriented Access

- Usable from different programming languages

- Compatible with different Communication Technologies

- Follows Industry Standards

- Stability and downwards compatibility

Benefits of BAPI

Can be used in diverse languages / Development Environments (ABAP, Visual Basic, Java, C++, etc.)

Can be called from diverse platforms (COM, CORBA, Unix)

Reduced development cost

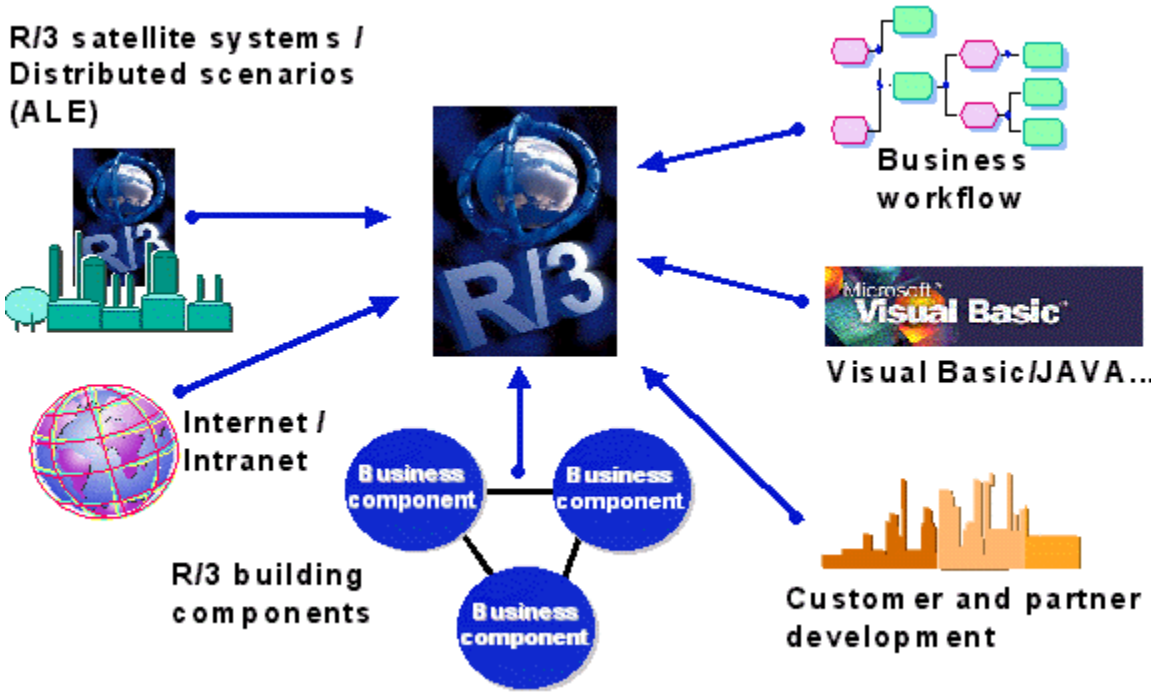
Reduced maintenance cost

“Best-of-both-worlds” approach

Rich functionality of the R/3 system

User-specific front-ends

Where BAPIs can be used



Characteristics of BAPI

The following are the characteristics of BAPI :

Naming Conventions : <business object>.<BAPI name>.

Database Consistency : Each BAPI that creates an instance of an object or updates the data of an object is responsible for database consistency.

No Dialog Orientation : BAPI do not return dialog screens from the R/3 server system to the calling program.

Authorization : Any interaction with the R/3 system requires the user to have a certain set of authorizations.

Standard BAPI

Some BAPI and methods provide basic functions and can be used for most SAP business objects. Such BAPI are known as ‘Standardized BAPI’.

Features : With BAPI (methods) ,we can differentiate between instance methods and class methods.

- Instance Method : Refer precisely to one instance
- Class Method : Instance independent.

BAPI for reading data :

- Getlist (Class Method) : List of object key values.
- Getdetail (Instance Method) : Uses a key to retrieve details about an instance of a business object and returns this data to the calling program.
- Getstatus (Instance Method) : The BAPI Getstatus is used to query the status of an SAP Business object.
- Existence Check (Instance Method) : This checks whether an entry exists for an SAP Business object.

BAPI for creating and changing data

The following BAPI can create, change or delete instances of a business object :

Create or createfromdata (Instance Method) : The BAPI create or createfrom data creates an instance of an SAP Business Object.

Change (Instance Method) : The BAPI changes an existing instance of a SAP Business Object.

Delete and Undelete (Instance Method) : The BAPI deletes an instance of an SAP Business Object from the database.

BAPI for Mass Processing: When a BAPI is called, several business object instances are processed at the same time.

With BAPI for Mass Processing 'Multiple' is added to the method name as

- CreateMultiple
- ChangeMultiple
- DeleteMultiple

Service BAPI for help functions

Service BAPI provide the information or services for the BAPI from individual business components.

HelpValues.Getlist : This method determines the allowed input values (F4 help).

BapiService.FieldHelpGetDocu : This method reads the documentation (F1 help).

BapiService.InterfaceGetDocu : This method reads the entire interface documentation of a bapi.

Service BAPI

BAPI for interpreting error messages :

BapiService.MessageGetDetail : This method displays the short and long texts of BAPI Error messages.

BapiService.ApplicationLogGetDetail : This method reads the details of entries in the application log.

BAPI for controlling COMMIT and ROLLBACK :

BapiService.TransactionCommit : This method executes a COMMIT WORK command.

BapiService.TransactionRollback : This method executes a ROLLBACK work command.

BAPI for conversion between internal and external data formats:

BapiService.DataConversionInt2Ext : This BAPI converts from internal format into the required external format.

BapiService.DataConvrsionExt2Int : This BAPI converts from external format into the required internal format.

Standardized Parameters

There are some parameters that can be created for various BAPIs because they contain some equivalent data in all BAPIs.Such parameters are known as ‘Standardized Parameters’.

The various Standardized Parameters are :

- Change Parameters
- Extension Parameters
- Return Parameters
- Selection Parameters
- Test Run Parameters

Prerequisites to work with Existing BAPI

The information required to call the method (BAPI) :

The name of the BAPI.

Details of BAPI interface :

- **Import Parameters** :

  Data to be transferred from the calling program to the BAPI.

- **Export Parameters** :

  Data to be transferred from the BAPI back to the calling program.

- **Import/Export (Table) Parameters** :

  Importing and Exporting data.

Customized BAPI

Custom BAPI

Custom BAPI are developed when there is no Standard BAPI to meet the requirements.

Process Flow

The BAPI process flow can be divided into four phases:

1. Analysis
2. Design
3. Implementation
4. Test and Release

Analysis in Detail

The analysis phase involves determining which business process is to be implemented using the BAPI.

The business objects types used in the scenario and their BAPI need to be identified.

The BAPI required for the scenario are simply identified, and their functions and the data they require are defined.

The analysis can be divided into three phases:

- Describing the scenario
- Defining the scenario in the BAPI Explorer
- Reviewing the scenario

Analysis Phase in detail

Describing the scenario

A scenario is a computerized implementation of a business process and Business process consists of a series of individual business functions.

Defining the scenario

Procedure to define a project form in the BAPI Explorer :

1.Start the BAPI Explorer using TC BAPI.

2.Select tab page Project and create a project to implement new BAPI.

3.In the first section the basic data can be created for the defined scenario.

Reviewing the scenario

Before the scenario can be converted and started with a concrete definition and implementation of the BAPI, the scenario

Designing Phase in Detail

After the scenario and the business object types used in it have been identified in the analysis of the scenario, the BAPI signature is defined conceptionally within the design phase.

The following conventions need to be adhered for BAPI methods:

- Check whether instance method or class method is used.
- The method name must be maximum of 30 characters.
- The individual components of the BAPI names are marked by initial letters. Underscores are not permitted in BAPI names.
- Every parameter must have a return parameter that is either an export parameter or an export table.

Implementing a BAPI

The Implementation phase can be divided into three successive stages:

Defining the Data Structures in the ABAP dictionary (SE11).

Implementing the function module in the function builder as Remote Enabled (SE37).

Defining the business object type and its methods in the BOR (sw01).

Testing and Releasing

Testing

- Testing the function module in the function builder.

Releasing

- Once all the tests have been completed successfully, the BAPI and all related development objects must be released in order to make it available to the customer.

Integration and Communication Services

Integration

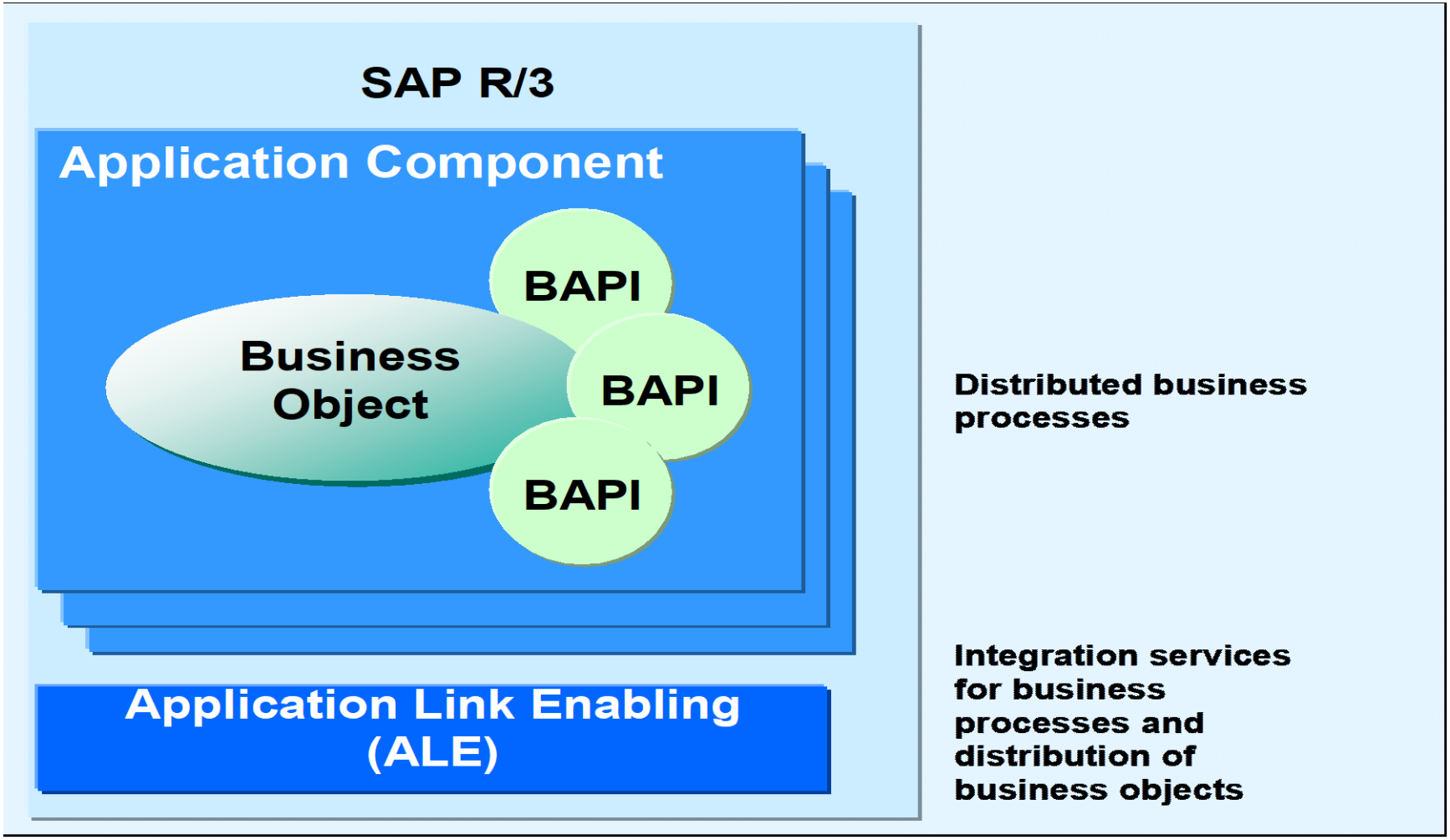
BAPI interface technology forms the basis for the following developments:

- Isolating components within the R/3 System in the context of Business Framework.
- Distributed R/3 scenarios using Application Link Enabling (ALE).
- Connecting R/3 Systems to the Internet using Internet Application Components (IACs).
- PC programs as frontends to the R/3 System, for example, Visual Basic (Microsoft) or Visual Age for Java (IBM).
- Workflow applications that extend beyond system boundaries.
- Customer‘s and partner‘s own developments.
- Connections to legacy systems.

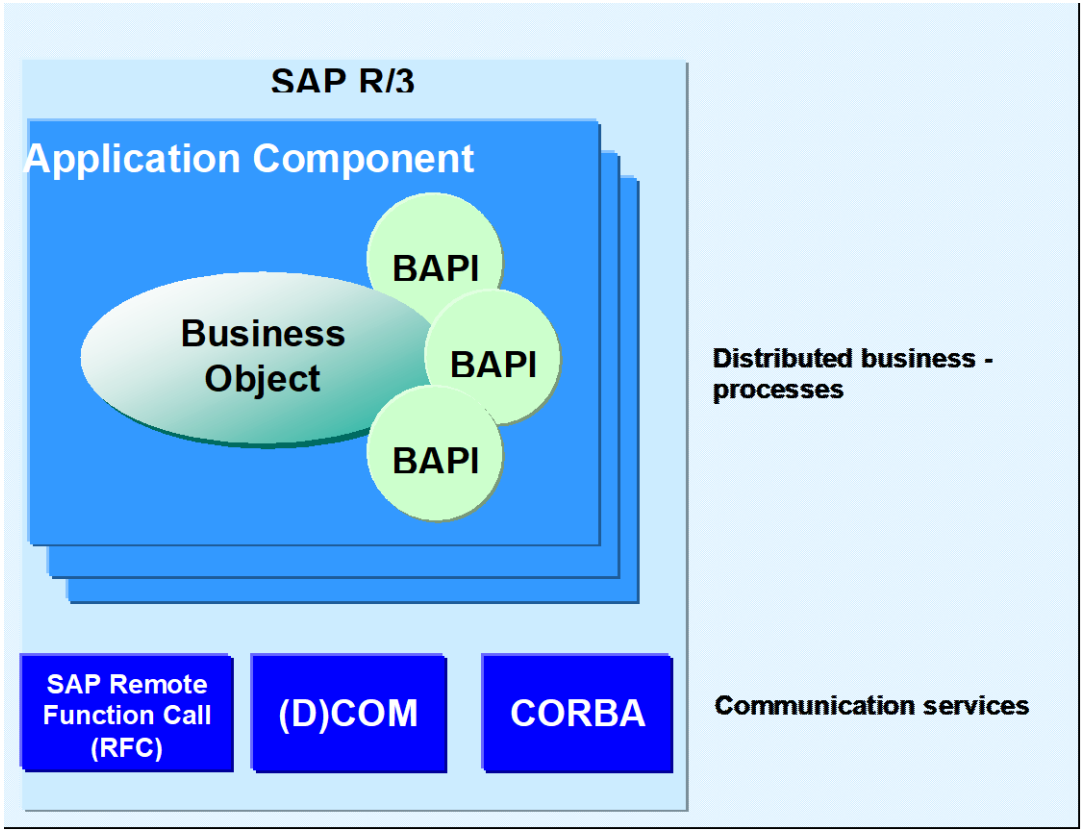
Releasing

- Once all the tests have been completed successfully, the BAPI and all related development objects must be released in order to make it available to the customer.

Integrated Services



Communication Services



Use for Data Upload and Reporting

BAPIs can also be used in ABAP for DATA upload and Change existing data and Interfaces.

Reduces some complexities which arise while write BDC’s.

Also while doing interfaces its helpful as it doesn’t trigger a new LUW as a Call Transaction would.

ABAP Reports that uses BAPI Function Modules.

Other types of Access

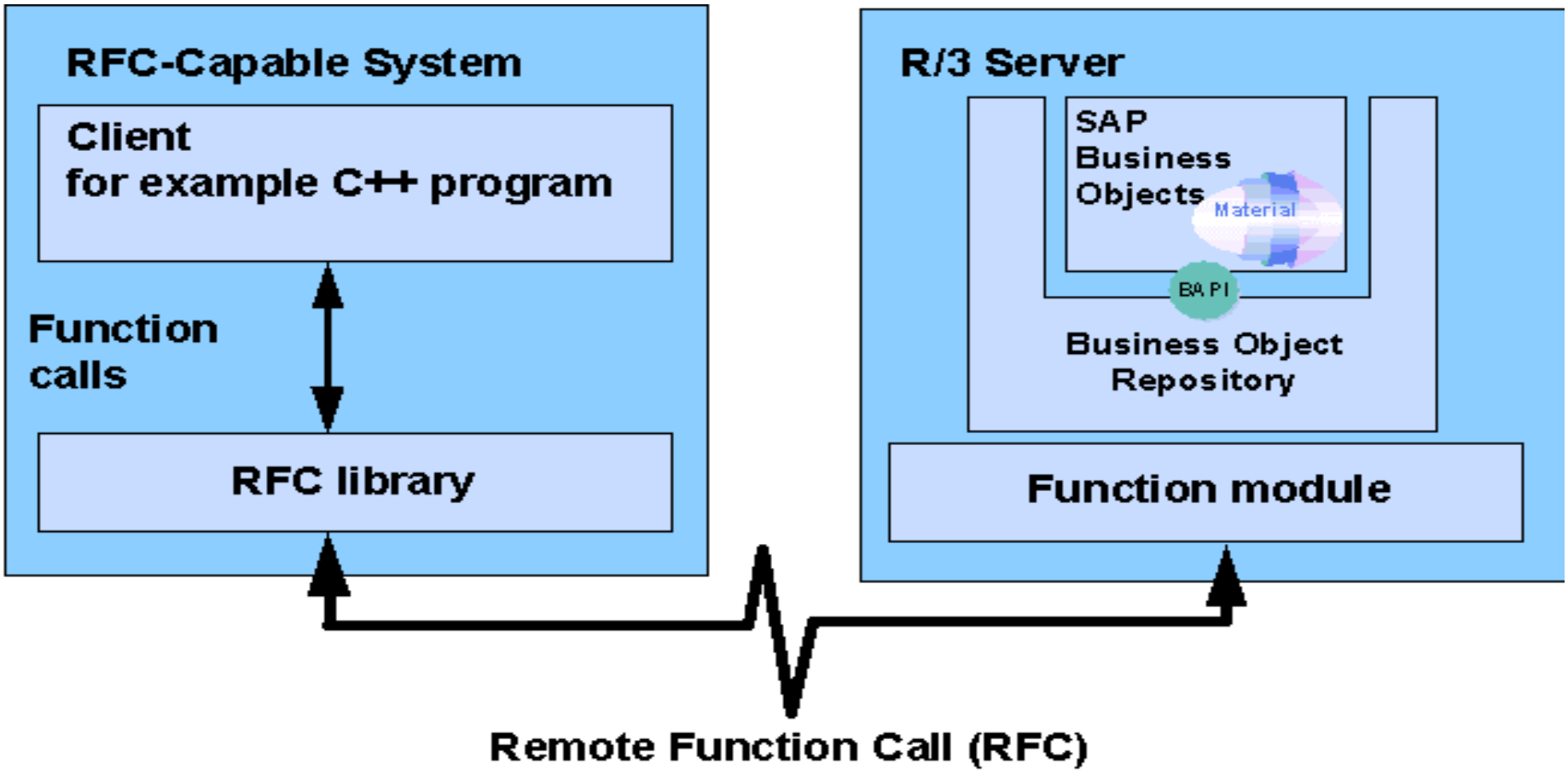
Can be accessed as a Remote Function from any programming language using commands for remote access for that particular language.

From ABAP it can be accessed as a normal function module depending on whether it is called from the same SAP system or a remote system.

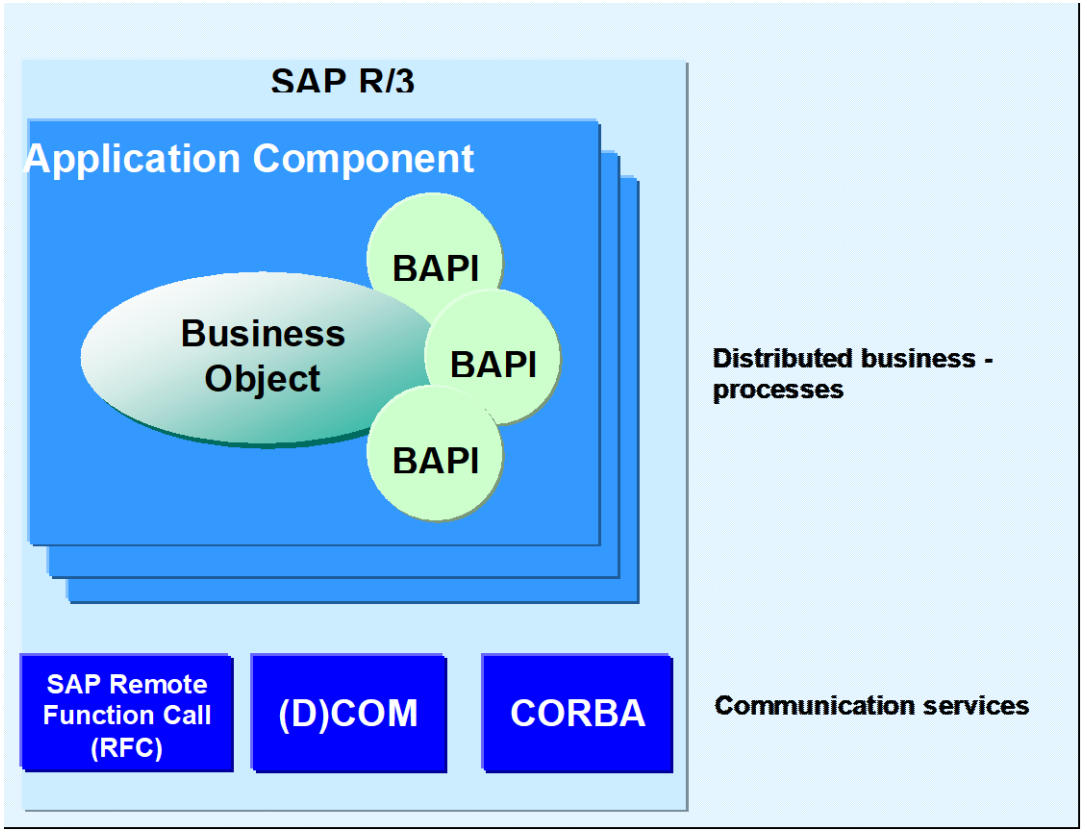
Function –Oriented Access can be made from all languages which support SAP RFC protocol.

BAPI’s can also be called from a remote SAP system or from the same SAP system. This call is very similar to a normal SAP Function Call.

RFC BAPI Access



Communication Services



Exercise

Prepare and call a customized BAPI to display the material master record.

Prepare and call a customized BAPI to create the customer master record.

Prepare and call a customized BAPI to create the vendor master record.

Upload the material master data by using standard BAPI.

Upload the sales order data by using standard BAPI.

Summary

By end of this course, participants know

- The Structure of Business Framework
- Understand Business objects and BOR
- Understand the advantages and characteristics of BAPI
- To Work with standard BAPI and custom BAPI
- BAPI Uploads and BAPI Reporting