

SAP ABAP - ABAP and SAP terminology

- [ABAP](#)
- [ABAP Dictionary](#)
- [ABAP Programming Models](#)
- [ABAP Program](#)
- [ABAP Workbench](#)
- [BADI](#)
- [BAPI](#)
- [CAAT](#)
- [Code Inspector](#)
- [Development Class](#)
- [Function modules](#)
- [Function Groups or Function Pool](#)
- [IDoc interface](#)
- [Include program](#)
- [Interface Communication](#)
- [Native SQL](#)
- [Open SQL](#)
- [Module Pool \(Also called Online Programs\)](#)
- [RFC](#)
- [SAP Database Interface](#)
- [SAP NetWeaver](#)
- [Screens](#)
- [Transactions](#)
- [Transport](#)
- [User exit](#)
- [Web Dynpro for Abap](#)



Summary: This section provides some short information about the ABAP and SAP vocabulary.

ABAP

ABAP originally is an abbreviation of **Allgemeiner Berichtsaufbereitungs prozessor**

It is the German for "generic report preparation processor". Later this was renamed "**Advanced Business Application Programming**".

ABAP/4 is a registered trademark of SAP Aktiengesellschaft Systeme, Anwendungen, Produkte in der Datenverarbeitung, Neurtstr. 16, D-69190 Walldorf.

ABAP is a proprietary language of SAP.

ABAP Dictionary

The ABAP Dictionary permits a central description of all the data used in the system without redundancies. New or modified information is automatically provided for all the system components. This ensures data integrity, data consistency and data security. Corresponding objects (tables or views) can be created in the underlying relational database using these data definitions. The ABAP Dictionary describes the logical structure of the objects used in application development and shows how they are mapped to the underlying relational database in tables or views

ABAP Programming Models

- Object-Oriented programming model (based on classes and interfaces)
- Procedural programming model (based on function modules and subroutines)

ABAP Program

Each ABAP program consists of an introductory program part, a global declaration part and a number of processing blocks which implement different functions depending on the application (procedures, dialog modules, event blocks). Each ABAP program has a program type which must be determined in the program attributes when the program is created. There are seven program types from which you can choose: executable program, module pool, function group, class pool, interface pool, subroutine pool, and include program.

ABAP Workbench

The ABAP Workbench is the SAP's integrated graphical programming environment. The code editor can be called by using the transaction SE38. It is used to load the CAST SAP extractors in the SAP environment.

BADI

BADI's (full form is "Business Addins") are a new SAP enhancement based on ABAP Objects. These are like user exits but are implemented using ABAP Objects. BADI's can be implemented in every stage of the systems landscape and so can be used any number of times, where as USER-EXITS can be used only one time.

BAPI

BAPI means "Business Programming Application Interfaces". External access to the data and processes is only possible by means of specific methods - BAPIs. A BAPI is defined as a method of an SAP Business Object. The BAPIs in the R/3 System are currently implemented as function modules, all of which are held in the Function Builder. BAPI mechanism is used for synchronous transactions.

CAAT

CATT (Computer Aided Testing Tool) is intended to be used as a testing tool in the SAP world. However, CATT is often used in other ways including:

- Uploading master data from vendors and changing other master records (which is helpful for SAP users)
- creating test data (which is helpful to SAP consultants and Abapers)

Code Inspector

Code Inspector is a tool for evaluating the quality of ABAP programs and other repository objects. Code Inspector checks Repository objects regarding performance, security, syntax, and adherence to name conventions. The range of functions in the Code Inspector is limited to checking static object definitions and can therefore only point out certain problems. It cannot, for example, give a precise statement as to how the overall performance of a program is. To be able to do this in SAP, you need to analyze program execution yourself at runtime.

Development Class

A development class is a group of logically related development objects. In the latest versions of SAP, this terminology is replaced by "Package" in order to avoid confusing development classes with ABAP Object classes.

Function modules

Function modules are procedures that are defined in special ABAP programs only, so-called function groups, but can be called from all ABAP programs. The SAP System contains several predefined function modules that can be called from any ABAP program.

Function Groups or Function Pool

Function groups or Function pools are containers for function modules and are the only programs that are allowed to contain function modules. A function group cannot be executed. When a function module is called, the system loads the whole of its function group into the internal session of the calling program (if it has not already been loaded). Function groups, like executable programs and module pools, can contain screens (selection screens and lists) as components

IDoc interface

IDoc is the standard data structure for electronic data interchange (EDI) between application programs written for the SAP system or between an SAP application and an external program. IDocs are used for asynchronous transactions.

Include program

When the same sequence of statements in several programs is to be written repeatedly they are coded in include programs (External programs) and are included in ABAP programs. In contrast to all other program types, include programs do not represent stand-alone compilation units with a memory area of their own. Include programs cannot be executed. They merely serve as a library for ABAP source code. They are exclusively used to organize program texts into small editable units which can be inserted at any place in other ABAP programs using the INCLUDE statement.

Interface Communication

Related to communication between:

- SAP modules: RFC (Remote Function Call),

- IDOCS: data structure with other applications,
- BAPI (Business Programming Application Interfaces),

Native SQL

ABAP Native SQL allows the inclusion of database-specific SQL statements in an ABAP program. It is generally recommended to use Open SQL instead.

Open SQL

Open SQL statements are a subset of Standard SQL (see Native SQL) that is fully integrated in ABAP. They allow data access irrespective of the database system that your installation is using. Open SQL consists of the Data Manipulation Language (DML) part of Standard SQL; in other words, it allows you to read (SELECT) and change (INSERT, UPDATE, DELETE) data. ABAP applications contain as little Native SQL as possible (only used in a few components, for example, to create or change table definitions in the ABAP Dictionary).

Module Pool (Also called Online Programs)

Module pools do not produce lists. These programs define more complex patterns of user interaction using a collection of screens. The term "screen" refers to the actual, physical image that the users see. Each screen has its own Flow Logic, which is divided into a "PBO" (Process Before Output) and "PAI" (Process After Input) section. Online programs are not invoked directly by their name, but are associated with a transaction code. Users can then invoke them through customizable, role-dependent, transaction menus. Apart from reports and online programs, it is also possible to develop sharable code units such as class libraries, function libraries and subroutine pools.

RFC

RFC means "Remote Function Call". ABAP programs can be started either by the user or by the system (for example, in background processing), or through an external interface called Remote Function Calls (RFC). Each function module underlying a BAPI supports the Remote Function Call (RFC) protocol.

SAP Database Interface

The database interface provides the following services:

- Establishing and terminating connections between the work process and the database.
- Access to database tables
- Access to Repository objects (ABAP programs, screens and so on)
- Access to catalog information (ABAP Dictionary)
- Controlling transactions (commit and rollback handling)
- Table buffer administration on the ABAP application server.

SAP NetWeaver

SAP NetWeaver is an application builder platform from SAP for integrating business processes across various systems, databases and sources. It is the technological foundation for all SAP products since the SAP Business Suite. NetWeaver includes a group of relatively independent business applications running on a single technical platform, SAP Web Application Server (Web AS). The list of products is:

- SAP Web Application Server
- SAP Exchange Infrastructure (XI)
- SAP Enterprise Portal
- SAP Master Data Management (MDM)
- SAP Mobile Infrastructure (MI)
- SAP Business Intelligence (BI) aka
- SAP Business Information Warehouse
- SAP Knowledge Warehouse (KW)
- SAP Composite Application Framework

Screens

Selection Screen:

Selection screens are one of the three types of screen in the SAP R/3 System, along with dialog screens and lists (using ABAP statements). They are used to display data, and also allow user interaction. They can be output to the screen, but also to a printer. Unlike screens, which contain defined elements like input/output fields and pushbuttons, each of which is identified by a name, and where data is exchanged with the ABAP program by means of identically-named fields, lists provide a freely-definable area that you fill using the WRITE, ULINE; and SKIP statements.

Processing Screen:

It is a screen in the sense of a 'dynpro' (DYNamic PROgram) and it consists of a screen and its underlying flow logic.

Transactions

In SAP the code of the customization is written in separate code entities usually Z transactions. SAP has a strict rule about customization: names of objects have to start with a Z. If not the transaction can't run. SAP may allow some customers to use Y transactions. Certified partners delivering SAP complements can be authorized to write transaction that can start with Y or sometimes some other letters. A transaction is made of a series of screens called "dynpro".

Transport

A transport request allows the transfer of R/3 System components from one system to another. It is generally used in maintenance cycle when developers deliver the components they changed.

User exit

User exit is a functionality provided by SAP to add custom validation or enhancements to existing SAP transaction. User exits are user-defined FORM routines defined in the standard SAP programs, which are used to calculate and/or replace values within a validation, substitution, or rule. There are 3 types of areas for user exits:

- External data transfer
- Derivation of characteristics for profitability analysis
- Costing

Example: Due to the complexities of American tax rules and regulations, which seem to vary based on everything from type of industry to business practice, a user exit has been provided in the tax interface programming logic.

Web Dynpro for Abap

Web Dynpro is the SAP NetWeaver programming model for user interfaces (UIs).

The Web Dynpro programming model is based on the Model View Controller paradigm, and has the following features that build on the classic dynpro model:

- Clear separation of business logic and display logic
- Uniform metamodel for all types of user interfaces
- Execution on a number of client platforms
- Extensive platform independence of interfaces

Web Dynpro is available both in the Java and the ABAP development environment.