[Application Server](javascript:call_link('abensap_nw_app_server_glosry.htm')" \o "Glossary Entry) (AS), is a part of SAP NetWeaver. It provides a standardized environment for ABAP and Java EE applications.

Application Server

Abbreviated as AS ABAP. Variant of the [Application Server](javascript:call_link('abensap_nw_app_server_glosry.htm')) in [SAP NetWeaver](javascript:call_link('abensap_netweaver_glosry.htm')) for programming applications in ABAP. It is the successor of [SAP Basis](javascript:call_link('abensap_basis_glosry.htm')). Application Server ABAP provides the [ABAP runtime environment](javascript:call_link('abenabap_runtime_envir_glosry.htm')). It can be described as a client server system made up of at least three layers of software. These three software layers are the [presentation layer](javascript:call_link('abenpresentation_layer_glosry.htm')), [application layer](javascript:call_link('abenapplication_layer_glosry.htm')), and [database layer](javascript:call_link('abendatabase_layer_glosry.htm')). AS ABAP guarantees the independence of the ABAP application programs of the hardware, the operating system, and the [database](javascript:call_link('abendatabase_glosry.htm')). [ABAP programs](javascript:call_link('abenabap_program_glosry.htm')) can be executed only in AS ABAP.

Database Interface

[Interface](javascript:call_link('abeninterface_glosry.htm')) to an [AS ABAP](javascript:call_link('abensap_nw_abap_glosry.htm')) [database](javascript:call_link('abendatabase_glosry.htm')) that is integrated into the [ABAP runtime environment](javascript:call_link('abenabap_runtime_envir_glosry.htm')). Statements in [Open SQL](javascript:call_link('abenopen_sql_glosry.htm')) and [Native SQL](javascript:call_link('abennative_sql_glosry.htm')) access the database using the database interface. Accordingly, the database interface is divided into an [Open SQL interface](javascript:call_link('abenopen_sql_interface_glosry.htm')) and a [Native SQL interface](javascript:call_link('abennative_sql_interface_glosry.htm')). The database interface is responsible for data transport between the [application server](javascript:call_link('abenapplication_layer_glosry.htm')) and [database server](javascript:call_link('abendatabase_layer_glosry.htm')), automatic [client handling](javascript:call_link('abenclient_handling_glosry.htm')), and [table buffering](javascript:call_link('abensap_buffering_glosry.htm')).

Open SQL interface

Part of the [database interface](javascript:call_link('abendatabase_interface_glosry.htm')) responsible for [Open SQL](javascript:call_link('abenopen_sql_glosry.htm')) statements. The Open SQL interface transforms all Open SQL statements that access the [standard database](javascript:call_link('abenstandard_db_glosry.htm')) of an [AS ABAP](javascript:call_link('abensap_nw_abap_glosry.htm')) to vendor-specific [SQL](javascript:call_link('abensql_glosry.htm')) and forwards the results to the [database system](javascript:call_link('abendatabase_system_glosry.htm')).

Open SQL

Open SQL is the umbrella term for a subset of [SQL](javascript:call_link('abensql_glosry.htm')) realized using [ABAP statements](javascript:call_link('abenabap_statement_glosry.htm')), including the [DML](javascript:call_link('abendml_glosry.htm')) part. The Open SQL statements use the [Open SQL interface](javascript:call_link('abenopen_sql_interface_glosry.htm')) of the [database interface](javascript:call_link('abendatabase_interface_glosry.htm')) to access an [AS ABAP](javascript:call_link('abensap_nw_abap_glosry.htm')) [database](javascript:call_link('abendatabase_glosry.htm')). Open SQL can be used to read ([SELECT](javascript:call_link('abapselect.htm'))) and modify ([INSERT](javascript:call_link('abapinsert_dbtab.htm')), [UPDATE](javascript:call_link('abapupdate.htm')), [MODIFY](javascript:call_link('abapmodify_dbtab.htm')), or [DELETE](javascript:call_link('abapdelete_dbtab.htm'))) data in [database tables](javascript:call_link('abendatabase_table_glosry.htm')) defined in [ABAP Dictionary](javascript:call_link('abenabap_dictionary_glosry.htm')). Database tables, [views](javascript:call_link('abenview_glosry.htm')), and all [CDS entities](javascript:call_link('abencds_entity_glosry.htm')) can be accessed directly.

[Open SQL](javascript:call_link('abenopen_sql_glosry.htm')) is a set of ABAP statements, based on SQL, that asre transformed to platform-specific SQL by the [Open SQL interface](javascript:call_link('abenopen_sql_interface_glosry.htm')) of the database interface and then passed to the database.

Native SQL interface

Part of the [database interface](javascript:call_link('abendatabase_interface_glosry.htm')) responsible for [Native SQL](javascript:call_link('abennative_sql_glosry.htm')) statements. The Native SQL interface handles the statements triggered from the [ADBC](javascript:call_link('abenadbc_glosry.htm')) framework and all Native SQL statements embedded statically between [EXEC](javascript:call_link('abapexec.htm')) and [ENDEXEC](javascript:call_link('abapendexec.htm')). Vendor-specific [SQL](javascript:call_link('abensql_glosry.htm')) statements are passed, unchanged, to the [database system](javascript:call_link('abendatabase_system_glosry.htm')) of the current [database connection](javascript:call_link('abendatabase_connection_glosry.htm')). SAP-specific static Native SQL statements are handled in the Native SQL interface before being passed. To enable this, the Native SQL interface contains platform-dependent parts (known as client libraries), which also perform conversion between ABAP types and database types.

ADBC - ABAP Database Connectivity

ADBC is an API for the [Native SQL interface](javascript:call_link('abennative_sql_interface_glosry.htm')) of the AS ABAP that is based on ABAP Objects. The ADBC methods can be used to pass [Native SQL](javascript:call_link('abennative_sql_glosry.htm')) statements to the database interface. They make it possible to

* send database specific SQL commands to a database system and process the result
* to establish and administer database connections

**Notes**

* ADBC can always be used when access to a database using the Native SQL interface instead of the Open SQL interface is necessary
* Alongside ADBC, it is also possible to [embed](javascript:call_link('abennativesql.htm')) Native SQL statically between [**EXEC SQL**](javascript:call_link('abapexec.htm')) and [**ENDEXEC**](javascript:call_link('abapendexec.htm')) in ABAP programs. The recommendation, however, is to use ADBC.
* While the [static embedding](javascript:call_link('abennativesql.htm')) of Native SQL offers exclusively static access to the Native SQL interface, ADBC makes modern object-oriented and dynamic access possible.
* New developments and improvements, such as optimized performance using bulk access across internal tables, are now made only for ADBC.

The existing static embedding of Native SQL statements is still supported but should no longer be used in new programs

ADBC - CL\_SQL\_CONNECTION

The SQL statements that are represented by objects of the [CL\_SQL\_STATEMENT](javascript:call_link('abencl_sql_statement.htm')) and [CL\_SQL\_PREPARED\_STATEMENT](javascript:call_link('abencl_sql_prepared_statement.htm')) classes work by default with the [standard AS ABAP database](javascript:call_link('abenstandard_db_glosry.htm')). The following class is employed to use additional [database connections](javascript:call_link('abendatabase_connection_glosry.htm')):

* CL\_SQL\_CONNECTION

The following can be passed to the method GET\_CONNECTION of this class:

* The name of a [secondary connection](javascript:call_link('abensecondary_db_connection_glosry.htm')) from the column CON\_NAME of the database table DBCON

ADBC - CL\_SQL\_STATEMENT

* The CL\_SQL\_STATEMENT class contains instance methods that receive and execute dynamically created SQL statements

Instances of the CL\_SQL\_STATEMENT class can be created using the statement [CREATE OBJECT](javascript:call_link('abapcreate_object.htm')) or the instance operator [NEW](javascript:call_link('abenconstructor_expression_new.htm')), which allow a reference to an object of the [CL\_SQL\_CONNECTION](javascript:call_link('abencl_sql_connection.htm')) class to be passed to the constructor. If no database connection is passed, the [standard connection](javascript:call_link('abenstandard_db_connection_glosry.htm')) from the database interface to the [standard AS ABAP database](javascript:call_link('abenstandard_db_glosry.htm')) is used.