

Predict 4.2.2 Release Notes



These Release Notes are delivered with the current Natural Documentation CD-ROM and in printed form.

References contained in the Version 4.2.2 Release Notes generally refer to the corresponding Version 4.2.2 documentation.

This document covers the following topics:

- Introduction
 - Availability
 - New Features in Predict 4.2
 - Installation
 - Multiple associations between two object types
 - Changes in the Application Programming Interface PALUD
 - Links to and from Fields
 - Adabas
 - Vista Support in Predict
 - DB2
 - NDV Support
 - Miscellaneous
 - SYSHELP
 - Predict and Other Software AG Products
 - Applying Summary Updates or Subsequent Maintenance Levels
 - Modified Fields in the Predict System File
 - Maintenance End of Predict Versions
-

Introduction

These Release Notes inform you of the enhancements provided with Version 4.2.2 of Predict. Predict Version 4.2.2 contains all INPL updates and source changes applied to Predict Version 4.2.1 as error corrections. All known SAGSIS problems were solved.

Availability

Predict Version 4.2.2 is available on the following platforms:

- BS2000/OSD
- VM/CMS
- OS/390
- VSE/ESA

New Features in Predict 4.2

Installation

- List Xref is an internal product with product code PXR, that has to be installed in a separate installation step.
- The INPL datasets now also contain the error messages. Hence only one INPL for Predict and one INPL for PXR are needed.

- No Adabas unload for the FDIC documentation data is delivered. These data are only delivered in migrate format.
- The FDT of the Predict system file did not change. No adaptations to the FDT are necessary when using the inplace installation method. However the existing Predict 4.1 data must be converted. Conversion is only possible from Version 4.1 of Predict. Data from earlier versions must be converted using the Migrate Utility.
- A Coordinator FDIC with 4.1 FDT, can be used with both versions 4.1 and 4.2. Thus it is not necessary to load a new Coordinator FDIC, when using the inplace installation method.

Multiple associations between two object types

With Predict version 4.2 it is possible to define several relationship types (associations) between two object types. This enables you to document different kinds of relationships very precisely.

For example: Two relationships have to be documented for an object type named "Employee", and those are "Has subordinates" and "Has project employees". This was not possible in previous versions of Predict.

For each association, a definition of the active and the passive relationship in the meta data administration must be defined. Each of these relationships has a short code, a name and a title. The short code and the name of the active relationship must be unique for the parent object type. The short code and the name of the passive relationship must be unique for the child type.

In Predict Version 4.2, the identification of the relevant relationship type (active or passive) has to be written where the identification of the child type (active) or the identification of the parent type (passive) was indicated in previous versions of the product. You can use either the short code or the name of the association in commands. When converting from 4.1 to 4.2, all passive relationships are assigned the short code, the name and the title of the parent object type and all active relationships are assigned the short code, the name and the title of the child object types. As a result, all direct commands function exactly as they did before the conversion. The commands only have to be changed if the short code or the name is changed. This information cannot be changed for predefined associations.

You are recommended to use the kind of the parent or child types in the relationship title. This serves as an indicator as to what objects you are dealing with when working with editors or retrieval results.

If you have several associations between two object types, you must mark one of them with the "Unload association" flag in order to make the transport into earlier versions possible. Only then will the link data of this flagged association be transported to version 4.1 during unloading.

Changes in the Application Programming Interface PALUD

The #CHILD-TYPE parameter is now interpreted as a short code of the association.

Links to and from Fields

With Predict 4.2, you can define associations with the object type field (EL) as parent and/or as child type. These associations can be used in all Predict evaluation functions.

- these associations can be used in retrieval models
- these associations can be migrated
- there is a new Link function in Field Maintenance Menu

However some restrictions apply:

- Only real fields are allowed as parent and child objects: no redefinitions, no fields named Filler or Dummy. Both the file and the field object must exist when the link list is saved.
- Whenever a field is deleted, it is removed from all link lists, since dummy fields are not allowed.
- In the new link editor, an existing field can be modified with the .E command, or a new field can be added to an existing file. New fields are added at the end of the field list.
- Fields cannot belong to extracts; it must also be possible to export (migrate) them individually. Fields are

regarded as attributes of a file, not as independent objects.

- Fields are not objects that can be separately protected using Predict Security. The fields are protected as attributes of the file, which means that the file must be protected.

The Crossreference and Implode Field functions are now processed using new the predefined retrieval models IM and XR. These functions have been removed from the Retrieval Field Menu.

There is a new interface program named PALUD42 with an extended parameter data area, which makes it possible to pass a field ID qualified with the appropriate file ID.

Adabas

Adabas Version 5 is no longer supported in the functions Incorporate, Compare and Generate Adabas File/Database.

Support for Adabas Star has been removed. Instead support of Adabas Vista has been incorporated.

On the level of Predict database objects that means, the field Adastar Parameter has been renamed to Run Mode. The values 'Y' (Translator) and 'N' (No Translator) are no longer valid. The logical Adastar type for files has been renamed to logical distribution type. Replication as distribution type is no longer valid. In the Adabas attributes of a file, the physical Adastar type has been renamed to physical distribution type.

The ADARUN parameter DTP (Distributed Transaction Processing) can be documented for database objects. Possible settings are N (No (default)), RM (Resource Manager) and TM (Transaction Manager) . Predict ensures that only one database with the value TM can be defined in a virtual machine.

The documentation of reflective databases has been removed.

The new features of Adabas V7 have been taken into consideration.

UES Support

Wide Fields und Collation Descriptors

Adabas has a new field format named wide character (W) to support multiple-byte character sets (MBCS). In Predict, such fields are documented with field format A or AV and character set M (Multi byte). The option NV (do not convert) can be documented for fields of type Adabas, User View und Conceptual. In Adabas, a collation descriptor can be used to define a user-defined sorting sequence. In Predict, such descriptors are documented with the field type OD. They are treated like hyperdescriptors, with the restriction that they have only one source field.

For 3GL generation there are new values I7 and R7 for Adabas Version. With I7, collation descriptors are not generated in the format and the record buffer. For R7, they are generated into these buffers.

New Parameters for Specifying the Data Architecture and the User Encoding

At the database level parameters UES, UACODE, UWCODE, FACODE and FWCODE can be specified. At the level of the physical files FACODE, FWCODE and FUWCODE can be specified. In ADACMP generation it is possible to specify values for the UARC, UACODE and UWCODE parameters.

Additional options

The option XI (exclude the occurrence (index) number from the definition of uniqueness) is represented in Predict as value X for unique descriptor. It is only allowed for descriptor fields in periodic groups and sub-, super- and collation descriptors having source fields from periodic groups.

The setting of the parameter Allocation (Force/Noforce) can be specified in the Generate ADACMP function with the option Generate Loader.

The new option Index Compression can be specified in the Modify ADABAS Attributes function.

Online invert

With Predict 4.2, the function Generate Adabas file can invert fields online when it is running with an Adabas V7 database. A separate ADAINV job is no longer needed. Since the Adabas function Online Invert is executed asynchronously, only one Online Invert can be active for a file. There is a wait interval in the Generation Defaults for Adabas file that determines how long the function waits before a second Online invert is submitted. In addition, this function can be used to modify the field formats in existing FDTs from N to P and from A to LA.

On Windows and UNIX platforms, it is now possible with the Generate Adabas File function to adapt existing FDTs to the documentation in Predict.

Vista Support in Predict

The following Adabas Vista features are supported by Predict:

- Definition of partitioned files
- Definition of translation rules for single file numbers
- Definition of translation rules for all file numbers

Both types of translation rules can be combined with environment IDs

In Predict, the partitions of a file are defined in a new screen of the Modify Adabas Attributes function. Here the distribution key and the limits that determine in which partition a record is stored can be specified. For one physical Adabas file, up to 10 logical partitions can be defined. In the function Modify Vista Elements for Files you determine which physical partitions form a logical partitioned file. Here the logical DBIDs and file numbers that will be used for access are also defined.

It is also possible to define translation elements for individual files (physical distribution type ' ' = simple). With the new Modify Vista Elements function for Databases translation, elements can be defined that can be used to route all calls with one logical DBID to another physical DBID.

In addition to the definition of logical DBIDs and Fnr, further attributes can be specified for a partitioned file or a translation element, with which Vista controls access to the data.

Predict checks that existing databases and files also continue to be accessible, that is, that no logical DBID and file number combination is defined that already exists as physical DBID, Fnr. Using the option 'Unique DBnr/Fnr' in the General Defaults -> Miscellaneous, you can force all DBID, Fnr combinations to be unique throughout all networks.

At file level, (types A and U), you specify which DBID and Fnr will be used for access. When generating a DDM, you can then specify with the option Use 'Vista access-nr' how to proceed with this specification. The value N (No usage of access-nr) causes the DDM to be created with a physical DBID, Fnr. The value T (Usage of tested access-nr) checks whether there is a translation element that matches the access DBID und Fnr of the file. The value Y (Usage of access-nr without testing) causes the access DBID and Fnr to be put into the DDM whether or not there is a matching translation element. These checks also include the translation elements at database level.

In Predict 4.2, there is a Generate Vistatable function to pass the Predict definitions to Adabas Vista. In this function, you can specify that all FDTs of a partitioned file be checked for equality.

All databases that participate in Vista must have ADARUN=VISTA set. In Predict this is given with the value V (Vista) for Run Mode for the database. Partitioned files can be defined only in databases that have Run Mode V. Equally, translation elements can be defined only for files and databases that have this setting.

In network retrieval, with the function Vista numbers, you get information about which physical file and databases are accessing via a logical DBID, Fnr combination.

The functions Generate Vistatable and Compare Vistatable are built into the user interface of Predict 4.2.2. For further information refer to the section Generating, Incorporating, Comparing and Maintaining Data Definitions under Adabas Vista in the section Adabas Vista in the **Predict and Other Systems documentation**.

DB2

Predict requires DB2 version 6. The functions supported are described in the following sections.

DB2 databases and dataspace objects have some new attributes. The password clause for tablespaces and tables is no longer valid/necessary. The password parameters will still be accepted when executing generation functions in batch mode.

New data formats LOB and ROWID

LOBs are represented as the field format LO. The character set determines whether it is a BLOB, CLOB or a DCLOG. The lengths of these fields can be declared in the following units: bytes, kilobytes, megabytes or gigabytes.

Predict does not support the creation of auxiliary tablespace and auxiliary tables.

Predict uses the feature to let DB2 create auxiliary tablespaces and tables automatically when creating a table with LOB column. In order to do this, a statement SET CURRENT RULE='STD' is issued when a table like this is generated, provided that the Special Register Current Rule does not already have this setting.

ROWID fields are documented with field type QN. Their format is A and their length is 40. The field maintenance ensures that only one ROWID field exists per table. It also ensures that every table containing a LOB column also contains a ROWID column. It is possible to define an identity property for numeric fields.

Distinct Types

In Predict, distinct types are implemented with the help of an indicated standard file called SAG-DISTINCT-TYPE. The connection with the DB2 catalog is established by the names of the fields of this standard file. The field names of SAG-DISTINCT-TYPE consist of the schema name, a hyphen as qualification character and the type name.

Table fields that are connected to a standard field in SAG-DISTINCT-TYPE have the predefined format which is the basis for the type. Changes in the type definition are propagated via rippling to the derived fields.

Note:

The name of the standard field is not a valid column name. After copying from SAG-DISTINCT-TYPE via the SEL command into the field list of a table, the field name must be changed so that it conforms with the SQL naming standards. When a table is generated, a CREATE DISTINCT TYPE statement is created for every type in the fields with distinct type, if the type has not already been defined in the DB2 catalog. The data type definition of the table field is in this case the distinct type name.

When using the functions Incorporate and Compare on a table, the connections to the type definition are also compared. If the type definition in Predict is different from the one in DB2, then the field format of the table field is adapted to the catalog entry, and the field is marked as non-standard.

There are no explicit Generate, Incorporate and Compare distinct type functions. The type definitions are always regarded by the tables used. A type definition in DB2 is also deleted via Drop if the last table using it is dropped by Administration Database, Dataspace or File. Since every distinct type is based on a predefined data type, the table fields derived from these types are represented in the DDM with the predefined data type.

Trigger

There is a new object type called Trigger representing DB2 trigger objects. You can link an unlimited number of triggers to a table. Update Trigger that are only executed if certain fields of a table are changed are then linked to these fields.

The appropriate Triggers are created during the table generation process. Using Incorporate and Compare on tables creates trigger objects in Predict and establishes the links to the file, or to the fields, whichever is relevant. Triggers contain information as to when they are executed (by Insert, Update or Delete) and whether they are to be executed before or after a certain statement is executed. The code that is to be executed is defined in the trigger body, which is a text attribute in Predict.

Subselect and joins in view definitions

DB2 offers the possibility to use subqueries and joined tables in the from clause of the view definitions. This functionality is also supported with Predict 4.2. You can define in the DB2 view file whether the from clause is made up of an inner, left outer, right outer or full outer join. Then you indicate both masterfiles that are to be joined.

The syntax for this is already shown in the subquery editor. Another method to define the join type consists of the specification of the from-clause and the join condition in the subquery editor.

Predict knows a new file type Intermediate View (IV) used to define subselects in the from clause of a view definition. Files of type IV have a field list that reflects the selection clause and a subquery to show the search condition. Just like DB2 views, files of type IV can have a join type. They can be used as master files for E and IV files. When a DB2 view is generated that has a Intermediate View as a master file, a partial select statement is generated into the from clause. Files of type IV don't exist in the DB2 catalog. Nevertheless, the same naming standards are valid for files of type IV as for DB2 tables and views.

When using Incorporate and Compare, Intermediate Views are created in addition to the DB2 views in order to represent subselects in the from clause. The number of tables and views used in the definition of a DB2 views or Intermediate view has been raised to 100 in Predict 4.2. Procedures and functions

DB2 procedures and user-defined functions can be documented as program objects in Predict. The program type 'R' = SQL procedure has been enhanced to accept many specifications that are related to DB2. Procedures can be implemented in 3rd generation programming languages or SQL. The program type 'U' = Database function is a new feature of Predict 4.2. These functions can return a table as a result.

There are two new associations between object types program and file with the names 'Input FI' and 'Returns FI'. The association names indicate that the linked files represent the structure of the input parameter or the structure of the results table. Only files of type Intermediate Table (IT) can be linked (see below).

In DB2, you can use table functions in the definition of views. Predict 4.2 does not support this.

Intermediate tables

There is a new file type IT (Intermediate Table) for documenting the formal parameters of database functions. They do not exist in DB2. Their fields can, like DB2 table fields, have a link to the standard file SAG-DISTINCT-TYPE, that is interpreted as being distinct type.

NDV Support

Natural Development Server (NDV) stores the following information in the Predict system file:

- Information about the structure of applications
- Lock information

Locks are only used internally and can not be processed with Predict functions.

For more information about applications, see the Natural documentation **Introducing Natural's Single Point of Development** section **Application Concept**.

The following object types from the application manager of the Natural Development Server are handled:

- Base applications
- Compound applications
- Data Definition Modules
- Natural programming objects and resources
- Libraries
- Error message files

Base applications

Base applications are documented as objects of type System with system type B. A base application has the following specific attributes:

- Server name
- Port number
- Profile name
- Profile DBnr
- Profile Fnr

For base applications, the following specific associations are defined:

- *Has component FI* to specify which Data Definition Modules belong to an application.
- *Has component PR* to specify which Natural Programming Objects belong to an application.
- *Has library SY* contains the Natural Libraries, the programs in link *Has component PR* are located in. Links of this association can not be edited. They are maintained automatically when the links of association *Has component PR* are saved.

Compound Applications Descriptions - Compound APDs

Compound applications are documented as objects of Type System with system type O. A compound application has no specific attributes. The association from compound to base applications is titled *Has subappl. SY*.

Predict ensures that only systems of type B are linked to a compound APD.

Data Definition Modules - DDMs

Data Definition Modules are documented as objects of type File. The association from a base application to Data Definition Modules is titled *Has component FI*.

Libraries

Libraries are documented as objects of type System with system type A. These objects have a fully qualified implementation pointer. The association from a base application to libraries is titled *Has library SY* with short code.

Natural Programming Objects

Natural Programming Objects are documented as Program objects with the appropriate program type. Resources have program type 5. The association from a base application to Natural Programming Objects has the title *Has component PR*. The following conditions apply for a Natural Programming Object being linked to a base application:

- The implementation pointer must be fully qualified.
- All program objects in the link list must reside in the same Natural system file.
- For each Natural programming object, the corresponding library object must be documented as System with system type A, so that the link list for the association *Has library SY* (association code LI) can be maintained automatically.
- If the database number or the file number in the implementation pointer is changed (this corresponds to a Move

operation in another FUSER), the Natural programming object is removed from all link lists of the association *Has component PR*. If this was the last program in a library, the system representing that library is also removed from the link *Has library SY*.

Error message files

Natural error message files are documented as program objects with program type 2.

The definitions of applications can be maintained both from Predict and from Natural Studio. However in the current version of Predict the existence of locks are not considered, so it is possible to delete or modify an object in Predict although a lock exists for this object. Consideration for and setting of locks will be implemented in a later Predict version.

Each Natural program type has its own value as program type. Thus texts and Natural ISPF macros can also be documented.

Miscellaneous

New user exit USR2033N for reading LX sets. Functions supported:

- List of all sets of a user in a library
- List of all members of a set

SYSHELP

If a user copies the SYSHELP routines to a user library, the user exit USR2004N must be copied to SYSTEM or the application library.

Predict and Other Software AG Products

On Mainframe Platforms

For detailed information on product requirements refer to the section Product Requirements in the **Predict Installation documentation** for Mainframes.

On UNIX Platforms

For detailed information on product requirements refer to the section Product Requirements in the **Predict Installation documentation** for UNIX.

Applying Summary Updates or Subsequent Maintenance Levels

Detailed information on how to apply summary updates or system maintenance levels can be found in the corresponding section of the installation documentation for your platform.

On Mainframe Platforms

See Applying Summary Updates or Subsequent Maintenance Levels in the **Predict Installation documentation** for Mainframes.

On UNIX Platforms

See Applying Summary Updates or Subsequent Maintenance Levels in the **Predict Installation documentation** for UNIX.

Modified Fields in the Predict System File

See the section User Programs which access the Predict System File in the section **General Information** of the **Predict Installation for Mainframes documentation** for a description of how modified fields are marked in Predict.

Documentation Changes

With version 4.2.2 of Predict, an enhanced documentation set is provided. All known SAGSIS problems concerning the documentation were solved and included in this new documentation set.

Maintenance End of Predict Versions

Software AG's maintenance of Predict Versions is as follows:

- Maintenance of Predict version 4.1.x for UNIX platforms and Windows will end with the end of maintenance for Natural 4.1 for that specific platform.
- Maintenance of Predict version 4.1.x for Mainframe platforms will end on the 31st of December 2002.

Note:

To get further support for Predict Version 4.2.1, ensure you have at least Service Pack PD42117 installed on your system.