Predict

Predefined Object Types in Predict

Version 8.5.2

October 2022

ADABAS & NATURAL

This document applies to Predict Version 8.5.2 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 1983-2022 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at http://softwareag.com/licenses.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at http://softwareag.com/licenses/ and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at http://softwareag.com/licenses and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Document ID: PRD-OPRDPREDEF-852-20221008

Table of Contents

Preface xiii
1 About this Documentation 1
Document Conventions 2
Online Information and Support 2
Data Protection 3
2 General Information 5
Metastructure of the Predict Data Dictionary 6
Global Attributes 7
I Access Definition 13
3 Access Definition 15
Access Definition Maintenance Menu 16
Add an Access Definition Screen 17
Access Definition Retrieval 19
II Database 21
4 Maintaining Objects of Type Database 23
Database Maintenance Menu 24
Database Types 25
Defining Basic Attributes of Databases 26
5 Documenting Databases of Different Types 29
Database Type A - Adabas 30
Specifying the Size of an Adabas Database 32
Database Types C, E, P - Conceptual, General SQL Handler, Entire System Server Nodes 34
Database Type D - DB2 35
Database Type O - Oracle Handler 36
Database Types Q, M, R, H - Adabas SQL Handler, RMS Handler, rdb Handler, Other Handler 37
Database Type I - IMS 37
Database Type S - SQL Server 39
Database Type T - Target Node 40
Database Type V - VSAM Handler 41
Other SQL Database Types 42
6 Database-Specific Maintenance 43
Purge Database - Code P 44
Rename/Renumber/Retype Database - Code N 45
Special Functions for Editing the List of Files Contained in a Database 49
7 Database Retrieval 51
Database-Specific Retrieval Parameters 52
Database-Specific Retrieval Functions 52
Layout of Database Lists 54
Output Options for Database Retrieval 55
III Dataspace 57
8 Maintaining Objects of Type Dataspace 59

Dataspace Maintenance Menu 60
Adding a Dataspace 61
Defining Basic Attributes of Dataspace - DB2 Mainframe 61
Defining Basic Attributes of Dataspace - SQL/DS 67
Defining Basic Attributes of Dataspace - DB2 Open Systems 68
9 Dataspace-Specific Maintenance 73
Purge Dataspace - Code P 74
10 Dataspace Retrieval 75
Layout of Dataspace Lists 76
Output Options for Dataspace Retrieval 76
Output Options for Dataspace Retrieval - Continued 77
IV Extract 79
11 Maintaining Objects of Type Extract 81
Extract Maintenance Menu 82
Defining Basic Attributes of Extract 83
12 Extract-Specific Maintenance Functions 85
Copy Extracts - Code C 86
Operate on Extracts - Code T 86
Export an Extract - Code U 90
Edit/Link Objects - Code E 91
Extract Object Editor 91
Link Objects to Extract - Code E 96
Build/Extend an Extract - Code B 98
Purge Extract - Code P 101
13 Extract Retrieval 103
Extract-specific Retrieval Functions 104
Output Options for Extract Retrieval 104
Output Options for Extract Retrieval - Continued 105
V Field 107
14 Maintaining Objects of Type Field 109
Field Maintenance Menu 110
15 Defining Basic Attributes of Fields 113
Add/Copy/Modify Screen 114
Field Type 116
Level Number 117
Field Format 118
Character Set 119
Character Set - Adabas 120
Field Length 120
Descriptor Type 129
Descriptor Type - continued 130
Maximum Number of Values / Occurrences 131
Unique Option 131
Field Short Name 132
Suppression / Null Value Option 132

Variable Length Option - IMS 134
Null Default Option 134
Natural Field Length 134
Do Not Convert Option 135
Related Standard File 135
Check against standard 135
Natural Attributes 135
Hidden 136
Additional attributes / Associations 136
16 Defining Derived Fields 139
General Rules for Defining Derived Fields 140
Defining Derived Fields of Special Types 143
Rules Applying to Format Changes 146
Rules Applying to Suppression/Length Changes 149
Validation of Derived Field Definitions 149
17 Defining Additional Attributes of Fields 159
3GL Specification 160
Condition Name and Value 162
Field Name Synonyms 163
Old Mode Synonyms 164
Adabas Security and Edit mask 165
DBMS Extensions 166
Field Procedure 172
Derived Field Expression 173
Index Definition - DB2 174
Index Definition - Oracle 178
Default value 184
Constraint name 185
Identity definition / Change log 186
Platform Compatibilities 189
Base Extensions 190
18 Field Maintenance 195
Add a Field - Code A 196
Copy Field - Code C 196
Move Field within a File - Code H 197
Purge Field - Code P 197
Redefine Field - Code R 198
Browse Through Fields of a File - Code B 202
Edit Field Expression - Code Y 203
19 Field Retrieval 205
Field-Specific Retrieval Parameters 206
Sorting Fields and Files 207
Field-specific Retrieval Functions 209
Layout of Field Lists 210
Output Options for Fields 212

Output Options for Fields - Continued ... 213
VI File ... 215
20 Maintaining Objects of Type File ... 219
File Maintenance Menu ... 220
Common File Attributes ... 222
Defining Basic File Attributes ... 225
21 Adabas Files, File Type A ... 227
Add/Copy/Modify a File Screen ... 229
Modifying Adabas Attributes ... 230
Phys. distribution attr. ... 233
Miscellaneous Attributes and Adabas Security Definition ... 234
Extent Allocation - Size Specifications For More Than One Extent ... 235
Specifying Restrictions on Input Data - Distribution Criteria ... 236
Modifying ADAM Descriptor Definition ... 238
Modifying Vista elements ... 239
Encodings ... 239
22 File Types Conceptual, Standard and Other ... 241
23 SQL File Types ... 243
Naming Conventions for SQL Objects ... 244
Common Parameters for SQL File Types ... 245
Field Lists of SQL Views ... 246
Editing the Subquery of an SQL View ... 248
24 Adabas SQL Server ... 251
Overview ... 252
Naming Conventions ... 252
Adabas Cluster Table ... 254
Adabas SQL View ... 255
25 Adabas D ... 257
Naming Conventions ... 258
Adabas D Table, File Type BT ... 259
Adabas D View, File Type BV ... 260
26 DB2 ... 261
Naming Conventions ... 262
DB2 Table, File Type D ... 263
DB2 View, File Type E ... 266
Intermediate View, File Type IV ... 267
Intermediate Table, File Type IT ... 268
DB2 Query Table, File Type MT ... 269
27 Informix ... 273
Naming Conventions ... 274
Informix Table, File Type XT ... 275
Informix View, File Type XV ... 276
28 Ingres ... 279
Naming Conventions ... 280
Ingres Table, File Type JT ... 281

Ingres View, File Type JV 282
29 Oracle 283
Naming Conventions 284
Oracle Table, File Type OT 285
Oracle View, File Type OV 289
30 Sybase 291
Naming Conventions 292
Sybase Table, File Type YT 293
Sybase View, File Type YV 294
31 General SQL File, File Type X 295
32 RDB 297
33 IMS 299
IMS Segment Layouts and Userviews - File Types J and K 300
Editing Field Lists of IMS Files 301
34 VSAM 303
Physical VSAM File - File Type V 304
VSAM Logical Files, VSAM Userviews - File Types L, W and R 306
35 ISAM 309
36 Entire System Server 311
37 File-Specific Maintenance 313
Purge File - Code P 314
Rename File - Code N 315
Edit List of Fields - Code L 315
Force Standard - Code F 317
Push Backward - Code B 317
Modify Adabas Attributes - Code J 320
Modify Vista elements - Code K 320
Edit Subquery of a File - Code Y 320
38 Rippling - Ensuring Consistent Data Definitions 321
Overview 322
Rippling from Standard Files 324
Rippling from Master Files to Views/Userviews 328
39 File Retrieval 333
File Retrieval Screen 334
File-specific Retrieval Parameters 334
File-specific Retrieval Functions 335
Layout of File Lists 337
Output Options for File Retrieval 338
Output Options for File Retrieval - Continued 340
VII 343
40 File Relation 345
File Relation Maintenance 346
File Relation Retrieval 352
41 Interface 355
Interface Maintenance Menu 356

Defining Basic Attributes of an Interface 357
Interface Retrieval 358
VIII Keyword 359
42 Maintaing Objects of Type Keyword 361
Keyword Maintenance Menu 362
Defining Basic Attributes of Keyword 363
Keyword Maintenance Functions 363
43 Keyword Retrieval 367
Keyword-specific Retrieval functions 368
Layout of Keyword Lists 368
Output Options for Keyword Retrieval 369
IX 371
44 Library Structure 373
Library Structure Maintenance Menu 374
Add/Copy/Modify Library Structure Screen 375
Library Structure-Specific Maintenance 376
Library Structure Retrieval 377
45 Method 379
Method Maintenance Menu 380
Add/Modify a Method Screen 381
Method Retrieval 382
46 Network 383
Network Maintenance Menu 385
Add a Network Screen 385
Network-Specific Maintenance 386
Network Retrieval 386
47 Node 391
Node Maintenance Menu 392
Add a Node Screen 393
Node Retrieval 394
48 Packagelist 395
Packagelist Maintenance Menu 396
Packagelist Types 398
Add a Packagelist Screen 398
Packagelist-Specific Maintenance 399
Packagelist Retrieval 400
X Program 401
49 Maintaining Objects of Type Program 403
Program Maintenance Menu 404
Program Types 405
Languages 406
Program-Specific Libraries 407
Add a Program Screen 407
50 Defining Additional Attributes of Programs 409
Programs of Type Class 411

Programs of Type Resource 412
Programs of Type SQL Procedure 413
Programs of Type Database function 420
System Programs 423
Programs of Type dynamic 423
51 Program-Specific Maintenance 425
Editing Child Lists 426
Overview of Language-Specific Program Types 426
Generating Database Request Modules -DBRMs- from Program Objects of Language Q -Static SQL- 428
Purge Program - Code P 429
Redocument Program - Code X 429
Edit procedure code of a program - Code Y 435
52 Program Retrieval 437
Program-Specific Retrieval Parameters 438
Function Programs with Children with Child Type Program 438
Layout of Program Lists 438
Output Options for Program Retrieval 439
Output Options for Program Retrieval - Continued 440
XI 441
53 Property 443
Property Maintenance Menu 444
Add a Property Screen 445
Property Retrieval 446
54 Report Listing 447
Report Listing Maintenance Menu 448
Report Listing ID 449
Modify Report Listing Screen 450
Report Listing Retrieval 451
55 Server 453
Server Maintenance Menu 454
Add a Server Screen 455
Server Retrieval 456
XII Storagespace 457
56 Maintaining Objects of Type Storagespace 459
Storagespace Maintenance Menu 460
Add a Storagespace Screen 461
Storagespace-Specific Maintenance 462
57 Storagespace Retrieval 463
Unused Storagespaces - Code N 464
Layout of Storagespace Lists 464
Output Options for Storagespace Retrieval 464
Output Options for Storagespace Retrieval - Continued 465
XIII 467
58 System 469

System Maintenance Menu 471
System Types 472
Add/Copy/Modify System Screen 472
System-Specific Maintenance 473
System Retrieval 474
59 Trigger 477
Trigger Maintenance Menu 478
Add a Trigger Screen 479
Editing the Trigger Code of a Trigger 481
Trigger Retrieval 482
XIV User/Owner 483
60 Maintaining Objects of Type User/Owner 485
User/Owner Maintenance Menu 486
Add/Copy/Modify a User Screen 487
User Maintenance 488
61 User Retrieval 491
User-Specific Retrieval Parameter 492
User-Specific Retrieval Functions 492
Layout of User Lists 492
Output Options for User Retrieval 493
Output Options for User Retrieval - Continued 494
62 Owner Maintenance 495
Linking Objects Logically using Owners 496
Owner-Specific Maintenance Functions 496
63 Owner Retrieval 499
Owner-specific Retrieval Functions 500
Layout of Owner Lists 502
Output Options for Owner Retrieval 502
XV Verification 505
64 Maintaining Objects of Type Verification 507
Verification Maintenance Menu 508
Verification Status 509
Verification Formats 509
Add a Verification Screen 510
65 Verification-Specific Maintenance 513
Purge Verification - Code P 514
Rename/Change Status of a Verification - Code N 514
Edit Rule of a Verification - Code R 514
Rule Editor 515
66 Verification-Specific Retrieval 517
Verification-Specific Retrieval Parameters 518
Verification Specific Retrieval Functions 518
Layout of Verification Lists 519
Output Options for Verification Retrieval 519
Output Options for Verification Retrieval - Continued 520

XVI Virtual Machine 523
67 Virtual Machine 525
Virtual Machine Maintenance Menu 527
Add a Virtual Machine Screen 528
Virtual Machine Retrieval 528

Preface

This documentation describes all the predefined object types, provided by Predict. Type-specific attributes of the respective object type and the type-specific maintenance and retrieval functions are explained. Each object type is described in a separate section. The object types are arranged in alphabetical order.

This documentation covers the following topics:

General Information	Provides general information on the predefined object types in Predict. It describes global attributes such as object ID, restrictions and keywords. This general information is not repeated in the descriptions of the individual objects.
Access Definition	Objects of type Access Definition document masks and permissions of DB2 columns and tables.
Database	Objects of type Database document a collection of physical and/or logical files.
Dataspace	Objects of type Dataspace document DB2 tablespaces and SQL/DS DBspaces.
Extract	With this object type you can create sets of objects. An extract is used primarily for transferring data with the Predict Coordinator.
Field	With the object type Field you can document field definitions for a wide range of database management systems.
File	With the object type File you can document file structures for a wide range of database management systems. This section also describes the process of rippling.
File Relation	With file relations you can document the relationships between fields in a file.
Interface	Together with objects of type Method, Property and Program, interfaces document the Natural program object class.
Keyword	You can assign objects of type Keyword to other objects in order to link objects logically.
Library Structure	This object type supports the Steplib concept in Natural.
Method	This object type documents the methods of an interface.
Network	Together with objects of type Virtual Machine, networks document the hardware and operating system environment of a data processing system.
Node	This object type together with object type Server documents Remote Procedure Calls.
Packagelist	This object type documents DB2 packages.
Program	With objects of type Program you can document nearly 20 types of programs. Many different programming languages are supported.
Property	This object type documents the properties of an interface.
Report Listing	With this object type Predict Coordinator transfer operations and conversion functions are logged.

Server	This object type is used together with object type Node to document Remote Procedure Calls.
Storagespace	This object type documents DB2 storagegroups.
System	With this object type you can document complex applications.
Trigger	This object type documents SQL triggers.
User/Owner	An object of type User documents an individual user. Several users can be assigned to an Owner to represent organizational units. These owners can be assigned to other objects to link objects logically.
Verification	Objects of type Verification document the processing rules for validating field values.
Virtual Machine	Together with objects of type Network, objects of type Virtual Machine document the hardware and operating system environment of a data processing system.

1 About this Documentation

- Document Conventions 2
- Online Information and Support 2
- Data Protection 3

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
`Monospace font`	Identifies service names and locations in the format *`folder.subfolder.service`*, APIs, Java classes, methods, properties.
`Italic`	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
`Monospace font`	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Product Documentation

You can find the product documentation on our documentation website at **https://documentation.softwareag.com**.

In addition, you can also access the cloud product documentation via **https://www.softwareag.cloud**. Navigate to the desired product and then, depending on your solution, go to “Developer Center”, “User Center” or “Documentation”.

Product Training

You can find helpful product training material on our Learning Portal at **https://knowledge.softwareag.com**.

Tech Community

You can collaborate with Software AG experts on our Tech Community website at **https://tech-community.softwareag.com**. From here you can, for example:

- Browse through our vast knowledge base.
- Ask questions and find answers in our discussion forums.
- Get the latest Software AG news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at **https://github.com/softwareag** and **https://hub.docker.com/publishers/softwareag** and discover additional Software AG resources.

Product Support

Support for Software AG products is provided to licensed customers via our Empower Portal at **https://empower.softwareag.com**. Many services on this portal require that you have an account. If you do not yet have one, you can request it at **https://empower.softwareag.com/register**. Once you have an account, you can, for example:

- Download products, updates and fixes.
- Search the Knowledge Center for technical information and tips.
- Subscribe to early warnings and critical alerts.
- Open and update support incidents.
- Add product feature requests.

Data Protection

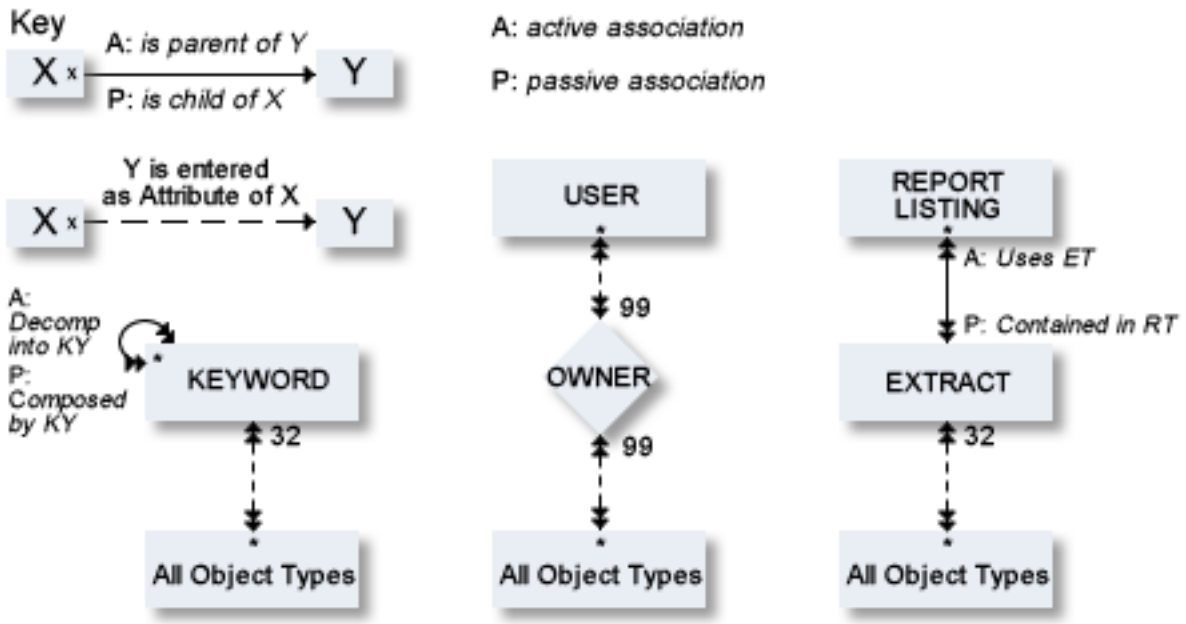
Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

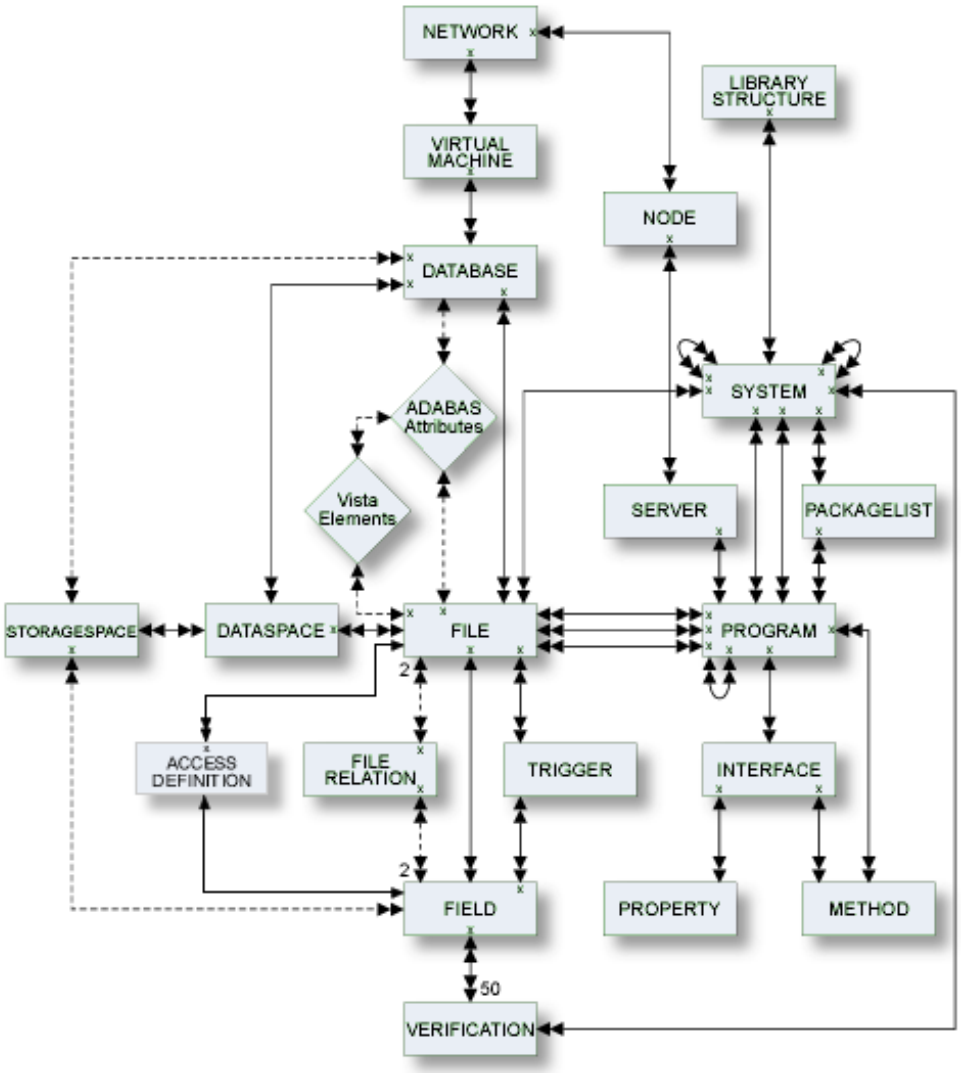
2 General Information

- Metastructure of the Predict Data Dictionary 6
- Global Attributes 7

Metastructure of the Predict Data Dictionary

The metastructure of the Predict data dictionary is illustrated below. Additional object types and association types can be defined with Metadata Administration functions. These objects are referred to as User Defined Object/Association Types or User Defined Entities (UDEs). See the section *Metadata Administration* in the *Predict Administration* documentation.





Global Attributes

The following attributes apply to all predefined and user-defined object types.

- Object ID
- Copy ID
- Default Parent
- Restrictions
- Keys
- Abstract

- Additional attributes / Associations

Object ID

Each object in Predict is identified by its ID. This ID must be unique for objects of the same type.

Note: Field objects can have the same ID if they belong to different files.

To change the ID of an object, use the function Rename. See *Renaming Objects* in the section *Maintenance* in the *Predict Reference* documentation.

Naming Conventions

IDs of all objects apart from verification are checked against the following naming conventions. (IDs of verifications are checked against Natural naming conventions.)

- The ID of all object types except user can be up to 32 characters long.
- Objects of type user can have IDs of up to 8 characters.
- The maximum length of object IDs (both for predefined and user-defined object types) can be specified with the metadata administration function Modify object type. See the section *Metadata Administration* in the *Predict Administration* documentation for more information.
- There is no minimum length for object IDs: one and two character IDs are also possible.
- An object ID must start with a letter (A - Z or a - z).
- The subsequent characters must be alphanumeric, i.e.
  - letters A - Z or a - z
  - digits 0 - 9
  - any special character except blank, asterisk, comma, question mark.
  - Up to 20 additional disallowed characters can be specified with the Metadata Administration function Modify object type. See the section *Metadata Administration* in the *Predict Administration* documentation for more information.
  - The Predict administrator can specify with the parameter General Defaults>Miscellaneous>Upper/lower case whether alpha characters in object IDs are converted to upper case. Use of lower-case letters for object IDs is not recommended.

Note: Object IDs should not contain the contents of session parameters IA and ID. Otherwise the command processor will not work properly.

Naming Conventions for Natural

If the object ID is also to be referenced by a Natural subsystem, the Natural naming conventions should also be observed.

SQL Naming Conventions

Naming conventions for SQL objects are given in the part dealing with object type **File** in this documentation.

Naming Conventions for Extracts

The following extracts are added automatically with the Coordinator:

- #SAG-TRANSFER
- #SAG-ERROR

See the Predict Coordinator documentation for more information. These IDs are reserved.

Copy ID

With most object types, this parameter is used with the Copy function for the ID of the new object to be created.

For object type field and file, this parameter is also used by other functions. See ***Field Maintenance Menu*** and ***File Maintenance Menu*** respectively.

Default Parent

With many object types, a parent object for the default passive association can be specified. For some object types, a parent object is mandatory.

Restrictions

Restrictions are available in every maintenance, retrieval or active retrieval menu. You can limit the selection of objects for processing using a combination of the following:

- **Keywords**
  Up to five keywords can be specified. See *Relating Objects Logically* in the section *Predict Functionality* in the *Introduction to Predict* documentation and section ***Keyword*** in this documentation.
- **Owner**
  You can restrict the retrieval operation to objects that are assigned to a particular owner. See *Relating Objects Logically* in the section *Predict Functionality* in the *Introduction to Predict* documentation.
- **Extract**
  You can restrict the retrieval operation to objects that are contained in a specified extract. See the section **Extract** in this documentation.

- **String**
  You can restrict the retrieval operation to objects whose abstract, extended description, rules or ID contains the specified string.
- **Date**
  Retrieval operations can also be restricted by the parameter AND from date: only objects that were added or modified after a given date are evaluated.

See the section *Retrieval* in the *Predict Reference* documentation for more information.

Keys

Up to 32 keywords can be assigned to any Predict object, including keywords.

- The keywords, separated by the current input delimiter character, can be specified in the main Add / Modify screen. The input delimiter character is defined by the Natural GLOBALS command ID parameter.
- A keyword must exist as a Predict object before it can be assigned to another object. If you specify a keyword that is not defined in Predict, a Modify Keyword window appears in which you can enter a valid keyword. Use asterisk notation to display a range of keywords for selection. Mark the keyword(s) you wish to select with any non-blank character or use cursor selection.
- An asterisk before the `Zoom` field indicates that more keywords have been specified than can be displayed on one line. In this case, enter Y here to modify existing keywords or add new keywords.

See the part on object type **Keyword** in this documentation for more information.

Abstract

Each object in Predict can have an abstract providing short comments on the object.

- An abstract can have up to 16 lines of up to 30 characters.
- Abstracts can contain upper and lower-case letters. If the general default parameter Miscellaneous > Upper/Lower case > Abstract is set to U, all alphabetic characters are converted to upper-case.
- An abstract can be added, removed or modified whenever the Add, Copy or Modify function is used to maintain an object. The number of abstract lines displayed in the Add/Copy/Modify screen depends on the object type. Enter Y in the `Zoom:` field to display the maximum 16 lines.

Abstract Editor Commands

The following line commands are available for processing abstracts:

.c	Copy one line.
.d	Delete one line.
.i	Insert three lines.
.j	Join line with next line.
.s	Split line at cursor position.

These line commands are preceded by the escape character defined in the Natural parameter module.

Additional attributes / Associations

Most predefinded object types in Predict have the following options in the `Additional attributes/Associations` line at the bottom of every Add/Copy/Modify screen. An asterisk before any option indicates that attributes of the respective type exists.

Owner	Y	Edit owner list.
Desc.	Y	Edit description. The editor called depends on the environment in which you are working and various profile parameters.
Default child association	Y	Edit child list. The Predict Link Editor is invoked.

See the section *Editors in Predict* in the *Predict Reference* documentation.

Note: All type-specific options in the `Additional attributes/Associations` line (for example Expr. for fields) are described in the respective part of this documentation.

Note: User-defined attributes are also shown if `Additional attributes/Associations` is set to Y.

I Access Definition

3 Access Definition

- Access Definition Maintenance Menu 16
- Add an Access Definition Screen 17
- Access Definition Retrieval 19

Masks and permissions for DB2 columns and tables are documented with objects of type Access Definition.

In the predefined Predict metastructure, an access definition can have passive associations of the following types:

Valid passive associations:	"Is controlled by FI" (for permissions) "Is controlled by EL" (for masks)

Access Definition Maintenance Menu

The Access Definition Maintenance menu is called with function code M and object code AN in a Predict main menu or the command `MAINTAIN AN`.

```
16:17:14                 *****  P R E D I C T  *****                  2011-10-18
Plan   0            - (AN) Access Definition Maintenance -        Profile SYSTEM

 Function                                Function

 A  Add a Access Definition              D  Display Access Definition
 C  Copy Access Definition               L  Link children
 M  Modify Access Definition             S  Select Access Definition from list
 N  Rename Access Definition
 P  Purge Access Definition


Function ..............

Access Definition ID ..                                   Attributes ....*
Copy ID ...............


Restrictions .........*   Profile Default,empty          Association ...*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation.
Access Definition ID	ID of the access definition.

Add an Access Definition Screen

The screen is displayed for the Add a Access Definition function. The Copy and Modify screens are similar.

```
17:04:52                ******  P R E D I C T  *****                  2011-11-16
                        - Add a Access Definition -
Access Definition HNO-AN
Type ...........*
Keys ..                                                                Zoom: N

Access Definition attributes
  DB2 name ..........
  Correlation name ..

Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N                    ↩
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters		
Type	The type of the access definition. Valid values:	
	M	Mask. A column mask in DB2 is used for column access control and specifies the value that should be returned for a specified column. Exactly one mask per column is allowed. In DB2 an enabled column mask does not take effect until the `ALTER TABLE` statement with the `ACTIVATE COLUMN ACCESS CONTROL` clause is used to activate column access control for the table.
	R	Permission. Multiple row permissions can be created for a table. In DB2 an enabled row permission does not take effect until the `ALTER TABLE` statement with the `ACTIVATE ROW ACCESS CONTROL` clause is used to activate row access control for the table.
DB2 Name	The name of the DB2 table or column.	
Correlation Name	The correlation name defined in the subselect clause of a view.	

Access Definition Retrieval

Layout of Access Definition Lists

```
13:26:48                ******  P R E D I C T  *****                  2011-12-01
                         - List Access Definition -

-------------------------------------------------------------------------------
Cnt  Access Definition ID

   1 HEB-AN
     Abstract
       fdsa
   2 HEB-AN-BUS
   3 HEB-AN-149
   4 HEB-AN1
     Abstract
       fdsa
   5 HEB-AN3
   6 HEB-AN4
   7 HEB-MASK
   8 HEB-PER
   9 HEB-PER_1
  10 HNO-TEST
Command ===>                                                   Scroll ==> CSR
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit        RFind Flip  -     +           Left  Right        ↩
```

Output Options for Access Definition Retrieval

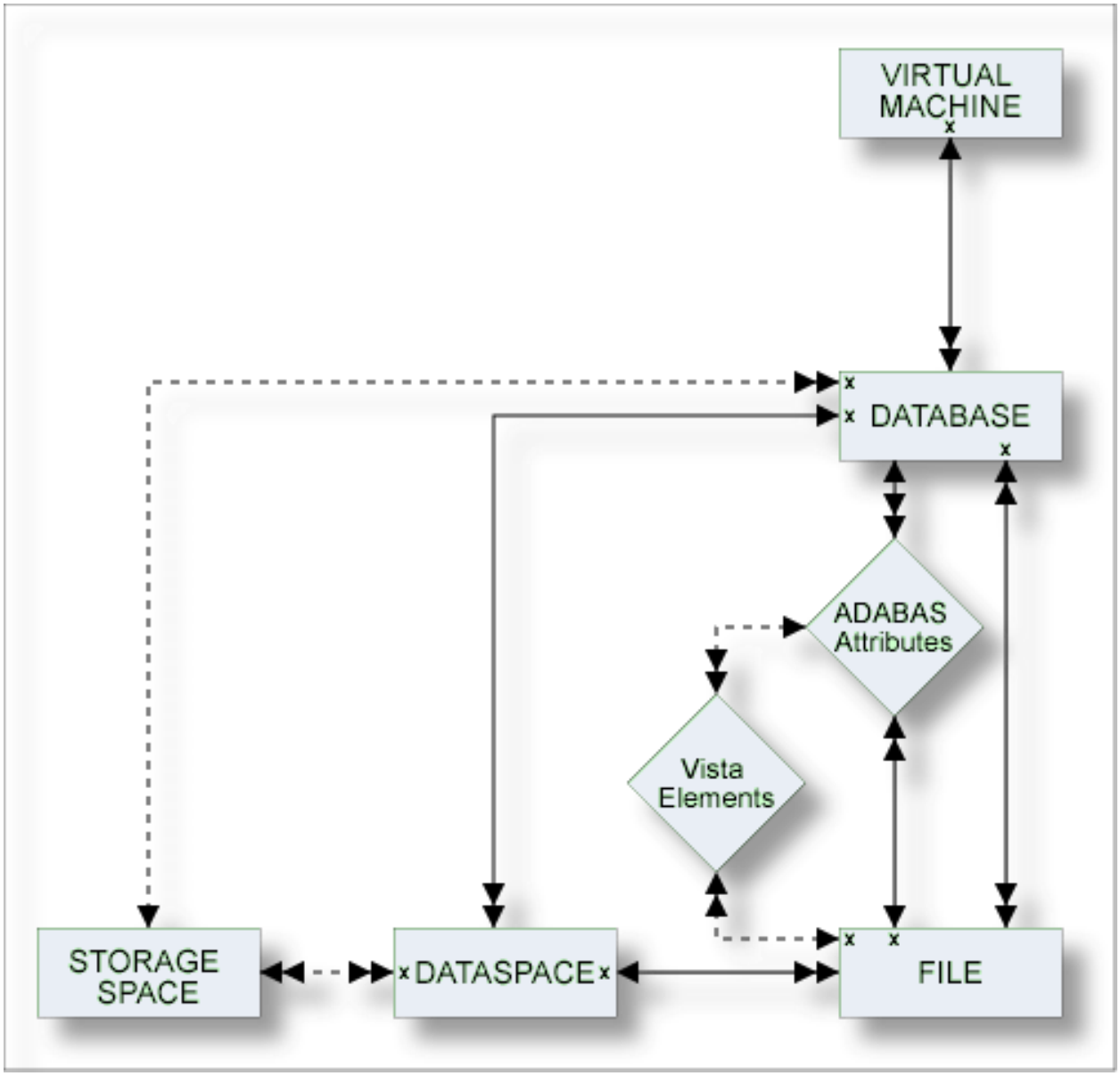
The output options valid for this object type are identical to those for object type dataspace. See *Output Options for Dataspace Retrieval*.

II Database

Databases and data storage systems of different types are documented with objects of type Database.

In the Predict metastructure a database can/must have passive and active associations of the following types:

Valid passive association:	"Belongs to VM" (mandatory for most database types)
Valid active associations:	"Contains DC" "Contains FI" (default active association)



The description of object type Database is organized under the following headings:

Maintaining Objects of Type Database

Documenting Databases of Different Types

Database Specific Maintenance

Database Retrieval

4 Maintaining Objects of Type Database

- Database Maintenance Menu 24
- Database Types 25
- Defining Basic Attributes of Databases 26

Database Maintenance Menu

The Database Maintenance menu is called with function code M and object code DA in a Predict main menu or the command `MAINTAIN DATABASE`.

```
17:26:42              *****  P R E D I C T  *****              2007-05-31
Plan   0                - (DA) Database Maintenance -            Profile SYSTEM

 Function                                Function

 A  Add a database                       D  Display database
 C  Copy database                       L  Link children
 M  Modify database                      S  Select database from a list
 N  Rename/renumber/retype database      K  Modify Vista elements
 P  Purge database                       J  Physical Attributes


Function .........

Database ID ......                                  Attributes .......*
Copy ID ..........                                  Database of type .*
Belongs to VM ....                                  Database number ...

Restrictions ....*   Profile Default ,used          Association ......*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main   ↩
```

Parameters	
Parameters not listed here are described under ***Global Attributes***.	
Function	All standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. The functions Rename/renumber/retype database and Purge are described in the section ***Database-Specific Maintenance***.
Database of type	With the Select function, a database type can be specified as selection criterion. The Add and Copy functions pass the type to the Add/Copy database screen. Note that type I (IMS) is not valid for the Add and Copy functions. Enter an asterisk to display a selection window with the valid database types for a particular function in your environment. For a list of all valid database types, see **Database Types**.
Database number	For the Add and Copy functions: the database number can be specified here. This number will be passed to the Add a database or Copy database screen. See the description of the parameter `Physical database number` in ***Defining Basic Attributes of Databases***.

Parameters	
	For the Select function: a database number can be specified as an additional selection criterion.
Association	For function Link children: objects are to be linked to the database via the selected association. Valid values: "Contains DC", "Contains FI" and user-defined.

Database Types

The table below contains a list of all valid database types.

Code	Database Type
A	Adabas
B	Adabas D handler
C	Conceptual
D	DB2
E	Gen. SQL handler
H	Other handler
I	IMS
J	Ingres handler
M	RMS handler
O	Oracle handler
P	Entire System Server
Q	Adabas SQL handler
R	rdb handler
S	SQL Server
T	Target node
V	VSAM handler
X	Informix handler
Y	Sybase handler

Defining Basic Attributes of Databases

The Add a database screen is used by the functions Add and Copy. Depending on the database type, one or several type-specific screens follow. Subsequent screens and their input fields are described in the sections below. Parameters applying to all types of databases are described below.

```
   10:29:21             *****  P R E D I C T  *****               2007-05-31
                                - Add a database -
   Database ID ..... HNO-DA                      +All-----Database types--------+
                                                   ! _ A    Adabas               ↩
!
                                                   ! _ B    Adabas D Handler     ↩
!
                                                   ! _ C    Conceptual           ↩
!
                                                   ! _ D    DB2                  ↩
!
           Database type .............* *         ! _ E    Gen. SQL Handler     ↩
!
           Belongs to VM .............* HOME      ! _ H    Other Handler        ↩
!
           Run mode ..................*           ! _ J    Ingres Handler       ↩
!
           Physical database number ..*           ! _ M    RMS Handler          ↩
!
                                                   ! _ O    Oracle Handler       ↩
!
                                                   ! _ P    Entire Sys. Server   ↩
!
                                                   ! _ Q    Adabas SQL Handler   ↩
!
                                                   ! _ R    rdb Handler          ↩
!
                                                   ! _ S    SQL Server           ↩
!
                                                   ! _ T    Target Node          ↩
!
                                                   ! _ V    VSAM Handler         ↩
!
                                                   ! _ X    Informix Handler     ↩
!
                                                   ! _ Y    Sybase Handler       ↩
!
                                                   !                             ↩
!
                                                   !                             ↩
!
                                                   !                             ↩
!
                                                   !Command ==>_______           ↩
!
                                                  +------------------------------+
```

Attributes	
Database type	See ***Database Types*** for a list of possible types.
Belongs to VM	Predict virtual machine object documenting the hardware and operating system environment of the database. See also *Defining the Distribution of Data in Predict* in the section *Vista* in the *Predict and Other Systems* documentation.
Run mode	Use of the database with respect to the distribution of data with Adabas Vista. I — Isolated. Adabas Vista is not used. The database is isolated. L — Local. The database cannot be accessed using Entire Net-Work. V — Vista. Adabas Vista is used. Only valid for Adabas databases. See *Defining the Distribution of Data in Predict* in the section *Vista* in the *Predict and Other Systems* documentation for a detailed description of the meaning of the Vista parameter.
Physical database number	Valid values depend on **database type**: Database Type — Range of Database Numbers B, E, J, O, Q, R, X, Y — 1 - 255 A, H, M, P, T, V — 1 - 65535 Others — not applicable

5 Documenting Databases of Different Types

- Database Type A - Adabas 30
- Specifying the Size of an Adabas Database 32
- Database Types C, E, P - Conceptual, General SQL Handler, Entire System Server Nodes 34
- Database Type D - DB2 35
- Database Type O - Oracle Handler 36
- Database Types Q, M, R, H - Adabas SQL Handler, RMS Handler, rdb Handler, Other Handler 37
- Database Type I - IMS 37
- Database Type S - SQL Server 39
- Database Type T - Target Node 40
- Database Type V - VSAM Handler 41
- Other SQL Database Types 42

Database Type A - Adabas

```
 10:31:03            *****  P R E D I C T  *****            2007-05-31
                             - Add a Database -
 Database ID ..... HNO-DA
 Type ............ Adabas, Isolated
 Physical DBnr ... 1
 Belongs to VM ......... HOME
 Keys ..
Zoom: N

 Adabas attributes                    Natural file numbers

   Maximal files ........               System file (FNAT) ...
   Checkpoint file ......               NAT-Security (FSEC) ..
   Adabas security ......               Predict (FDIC) .......
   Size of RABN ........* 0
   Distr. transaction ..* N

   Vista access only .... N

 Abstract     Zoom: N





   Additional attributes ..* N          Associations ..* N
```

Note: Attributes that are not in the table below are described in the section *Defining Basic Attributes of Databases*. Two additional input screens can be called from this screen.

Attributes	
Adabas attributes	
Maximal files	Number of files permitted in the database (ADADEF parameter MAXFILES). This number must either be 0 or at least 5 but not more than 32767.
Checkpoint file	The number of the Adabas file which contains checkpoint information for the database. Predict automatically creates a data dictionary object with the file ID SAG-ADA-CHECKPOINT for this file.
Adabas security	The number of the Adabas file which contains Adabas security information for the database. Predict automatically creates a data dictionary object with the file ID SAG-ADA-SECURITY for this file.
Size of RABN	Specifies the length of RABNs in the database. 0 — not specified

Attributes		
Adabas attributes		
	3	3 Byte for 24-bit RABNs
	4	4 Byte for 31-bit RABNs
Distr. transaction	N	No (Default)
	RM	Resource Manager
	TM	Transaction Manager
	Field must be filled if the database is part of the distributed transaction processing (DTP) of the Adabas Transaction Manager	
Vista access only	Y	If the attributes of the database are such that files in the database can only be accessed using Adabas Vista.
	Vista access only is set by Predict.	
	If N, it can be set to Y with the Rename/Renumber/Retype function (code N).	
Natural file numbers		
System file (FNAT)	The number of the Natural system file.	
NAT Security (FSEC)	The number of the Adabas file which contains Natural Security information.	
Predict (FDIC)	The number of the Adabas file which contains the dictionary data.	
Additional Options		
Additional Attributes	Y	A new window within the screen is displayed for specifying either one or more of the following attributes:
	blank	Base attributes
	W	Description
	O	Owner
	1	Sizes
	2	Asso sizes
	3	Data sizes
	4	Encodings
	5	OS/400
Associations	Y	A new window within the screen is displayed for specifying an association.

Specifying the Size of an Adabas Database

Physical properties of a database (device types and sizes of the data sets containing the Adabas ASSO, DATA, WORK, SORT and TEMP) can be defined in the screen shown below.

The screen is displayed by setting the parameter Size in the Additional attributes window.

```
10:31:57             *****  P R E D I C T  *****             2007-05-31
                             - Add a Database -
Database ID ..... HNO-DA                              Added 2007-05-31 at 10:29
Type ............ Adabas Isolated                          by HNO
Physical DBnr ... 1
---------------------- Database primary sizes --------------------------------
                           Number of
         *Device      Cylinder     RABN
ASSO R1
     R2
     R3
     R4
DATA R1
     R2
     R3
     R4
WORK R1
     R2
SORT R1
     R2
TEMP R1

* Additional attributes ..* S          Associations ..* S
```

Parameters	
Device	Devices are identified with a four-letter code that must have been defined with the function Adabas device types in the Special functions menu. If a device type is changed, the change should also be made in each file object that is linked to the database.
Cylinder	The number of cylinders of the specified device that are occupied by the specified extent of the specified database.
Number of RABN	The number of RABNs (relative Adabas block numbers) of the specified device that are occupied by the specified extent of the specified database.

Rules for Defining the Size of a Database

- If the device type and the size in RABNs (relative Adabas block numbers) of each extent is specified, Predict calculates and displays the equivalent size in cylinders, beginning with a greater than sign (>) unless the number of cylinders is exactly equivalent.
- If the size is specified only in cylinders, Predict calculates and displays the equivalent size in RABNs. Adabas does not use the first track of the first extent of the Associator, Data Storage and workfiles. In these extents, the number of RABNs is therefore smaller then the number of blocks contained by the specified number of cylinders. The start and end of the range of alternate RABNs can also be specified.
- Four extents for ASSO and DATA (R1 - R4) can be defined in the above screen. To define more extents (up to 99) the parameter ASSO and/or DATA in the Additional attributes window of the screen has to be selected.

Note: See the *Adabas Administration* documentation for detailed information on the topic.

Specifying the Encodings of an Adabas Database

Universal encoding support of an Adabas Database can be defined in the screen shown below.

The screen is displayed by setting the parameter `Encodings` in the Additional attributes window.

```
10:31:03            *****  P R E D I C T  *****            2007-05-31
                            - Add a Database -
Database ID ..... HNO-DA-NEW                        Added 2007-05-31 at 10:30
Type ............ Adabas, Isolated                       by HNO
Physical DBnr ... 244


Universal encoding support
  UES ..... N (Y,N)
  UACODE .* none
  UWCODE .* none
  FACODE .* none
  FWCODE .* none









  Additional attributes ..* S          Associations ..* N
```

Note: See the Adabas documentation for detailed information on this topic.

Modify Vista Elements

```
10:34:50            *****  P R E D I C T  *****              2007-05-31
                          - Add Vista element -
Database ID ..... HNO-DA-NEW                          Added 2007-05-31 at 10:30
Type ............ Adabas, Vista                           by HNO
Physical DBnr ... 250


Network ......... HOME
Vista
 Environment ID .
 DBnr ...........
 Name ...........









  Additional attributes ..* S          Associations ..* N
```

Parameters

See the section *Including the Definition in the Vista Table* in the section *Adabas Vista* in the *Predict and Other Systems* documentation for a description of all possible parameters.

OS/400 Attributes

On the **OS/400 Attributes** screen, you can specify an OS/400 database in the field **Database Name**.

Database Types C, E, P - Conceptual, General SQL Handler, Entire System Server Nodes

A screen similar to the following is displayed when adding, modifying or copying databases of the types C, E and P.

```
 10:33:21            *****  P R E D I C T  *****            2007-05-31
                             - Add a Database -
 Database ID ..... HNO-DA-C
 Type ............ Conceptual
 Keys ..
Zoom: N

 Abstract     Zoom: N
```

The database-specific parameters are described in the section *Defining Basic Attributes of Databases*.

Database Type D - DB2

The following attributes apply to databases of type D. Attributes not listed here are described in the section *Defining Basic Attributes of Databases*.

```
 13:23:06            *****  P R E D I C T  *****            2009-05-31
                             - Add a Database -
 Database ID ..... HNO-DB2
 Type ............ DB2
 Belongs to VM ......... HOME
 Keys ..
                                                                    Zoom: N
 DB2 attributes
  DB2 name .............
 Physical attributes in <Default Server>                (new)
  Buffer pool .............* BP1
  Index buffer pool .......* BP2
  Temporary database ....... N  (Y,N)
  Data sharing group member.
  Default storagespace ....*
  CCSID ...................*    (none)
 Abstract     Zoom: N




   Additional attributes ..* N          Associations ..* N
```

Attributes	
DB2 Attributes	
DB2 name	The name of the database in DB2.
Buffer pool	The buffer pool of the database. Enter an asterisk for valid values.
Index buffer pool	Buffer pool which is used for the indexes created within the database. Enter an asterisk for valid values.
Temporary database	N: No (Default). Y: Database is used for declared temporary tables.
Data sharing group member	Name of the member of the data sharing group. Leave blank or specify name with up to eight characters (letters A-Z, digits 0-9 and special characters $, # and @).
Default storagespace	DB2 tables of the database will be implemented in this storage group if no other storage group is explicitly specified.
CCSID	Defines the encoding scheme of the database. blank: not specified A: ASCII E: EBCDIC

Database Type O - Oracle Handler

A screen similar to the following is displayed when adding, modifying or copying databases of the type O.

```
12:21:22                ******  P R E D I C T  ******                  2017-08-15
                             - Add a database -
Database ID ..... XYZ-DA-ORA




        Database type .............* O Oracle Handler
        Belongs to VM .............* HOME
        Run mode ..................* L Local
        Physical database number ..*
                                   ↩
```

The database-specific parameters are described in the section *Defining Basic Attributes of Databases*.

Database Types Q, M, R, H - Adabas SQL Handler, RMS Handler, rdb Handler, Other Handler

Database type Q is used to document databases of type Adabas SQL handler. See the section *Adabas SQL Server* in the *Predict and Other Systems* documentation for more information.

Database type M is used to document RMS databases; database type R is used to document rdb databases.

Database type H is used to represent database handlers, such as USER-DB, SESAM, DL/I, WIZ-ZARD, TRS etc. Database type other handler can be used to reserve a database number (prevent it from being used by Adabas).

```
 13:03:04            *****  P R E D I C T  *****             2007-05-31
                            - Modify Database -
 Database ID ..... HNO-DA-M                           Added 2007-05-31 at 10:51
 Type ............ RMS Handler                           by  HNO
 Physical DBnr ... 1
 Belongs to VM ......... HOME
 Keys ..
Zoom: N

 Abstract     Zoom: N
```

Parameters	
Physical database number	For database type RMS Handler: the database number must be declared in the Natural NATPARM parameter file as an RMS database number if DDMs for RMS files contained in the database are to be generated. See table in the section ***Defining Basic Attributes of Databases*** for range of permitted values.

Database Type I - IMS

IMS databases cannot be added with the Add a Database function. For further information on how to create an IMS Database object in Predict, see *External Objects in Predict*.

```
 13:20:27             *****  P R E D I C T  *****            2007-05-31
                             - Modify Database -
 Database ID ..... HNO-CUSTOMER                        Added 2007-05-31 at 13:11
 Type ............ IMS                                     by  HNO
 Belongs to VM .........
 Keys ..
Zoom: N

 IMS attributes
   IMS or DL1 ..... IMS
   IMS name .......
   IMS type ....... PHYSICAL

 Abstract     Zoom: N
   This database was incorporated
   from NDB: CUSTOMER
   on 2007-05-31





   Additional attributes ..* N          Associations ..* N
```

The following attributes apply to databases of type I. For attributes that are not in the table, see the section *Defining Basic Attributes of Databases*.

Attributes	
IMS attributes	
IMS or DL1	The kind of database. Valid values: IMS DL1
IMS name	The name of the database in IMS.
IMS type	The type of the database in IMS. Valid values: LOGICAL PHYSICAL

Database Type S - SQL Server

Database type S is used to represent objects of type SQL server.

Databases of type S are defined in two screens:

```
13:29:32             *****  P R E D I C T  *****             2007-05-31
                             - Add a database -
   Database ID ..... HNO-DA-S




           Database type .............* S SQL Server
           Belongs to VM .............* HOME
           Run mode ..................*   (none)
           Physical database number ..*
```

```
   13:23:47          *****  P R E D I C T  *****             2007-05-31
                                - Add a Database -
   Database ID ..... HNO-DA-S
   Type ............ SQL Server
   Keys ..

   SQL Server attributes
     Server name
     Port number
     Host name ..
     Driver .....

   Abstract     Zoom: N
```

Attributes not listed here are described in the section *Defining Basic Attributes of Databases*.

Attributes	
Run mode	Not possible for databases of this type. Default is none.
Physical database number	Not possible for databases of this type. If a Physical database number is entered, an error message is given.
SQL Server attributes	
Server name	The name that was specified when the data source was created.
Port number	Identifies the server daemon.
Host name	Identifies the host machine on which the server runs.

Attributes	
Driver	Specifies the database driver to be used.

Database Type T - Target Node

Database type T is used to represent database nodes entered in the ID table of an SVC which cannot be documented with a corresponding database type: BROKER, NATURAL GLOBAL BUFFER POOL etc.

This type of database is used to reserve the corresponding database number and thus prevent this number being used for an Adabas database.

Databases of type T are defined in two screens:

```
13:29:32             *****  P R E D I C T  *****             2007-05-31
                             - Add a database -
   Database ID ..... HNO-DA-T




           Database type .............* T Target Node
           Belongs to VM .............* HOME
           Run mode ..................* I Isolated
           Physical database number ..* 1
```

```
   13:23:47          *****  P R E D I C T  *****            2007-05-31
                                - Add a Database -
   Database ID ..... HNO-DA-T
   Type ............ Target Node, Isolated
   Physical DBnr ... 1
   Belongs to VM ......... HOME
   Keys ..
  Zoom: N

   Abstract     Zoom: N
```

Attributes not listed here are described in the section *Defining Basic Attributes of Databases*.

Attributes		
Run mode	Must be specified for databases of this type. Valid values:	
	I	Isolated
	L	Local
Physical database number	The physical database number must be in range 1-65535.	

Database Type V - VSAM Handler

Database objects of type V are used to collect all definitions of VSAM clusters which are accessed by the same Natural VSAM handler. The database number defined in a database object of type V is used by the `GENERATE DDM` function.

Databases of type V are defined in two screens:

```
13:57:30             *****  P R E D I C T  *****             2007-05-31
                            - Add a database -
   Database ID ..... HNO-DA-V




           Database type .............* V VSAM Handler
           Belongs to VM .............* HOME
           Run mode ..................* L Local
           Physical database number ..* 2
```

```
13:59:18             *****  P R E D I C T  *****             2007-05-31
                            - Add a Database -
   Database ID ..... HNO-DA-V
   Type ............ VSAM Handler
   Physical DBnr ... 2
   Belongs to VM ......... HOME
   Keys ..
Zoom: N

   Abstract     Zoom: N
```

Attributes not listed here are described in the section *Defining Basic Attributes of Databases*.

Attributes	
Run mode	Must be local for databases of this type.
Physical database number	The physical database number must be in the range from 1 - 65535.

Other SQL Database Types

The screens used to maintain database objects of the following types are the same as for VSAM databases above. The physical database number must be less than or equal to 254.

Type	Description
J	Ingres Handler
O	Oracle Handler
X	Informix Handler
Y	Sybase Handler
B	Adabas D Handler

6 Database-Specific Maintenance

- Purge Database - Code P 44
- Rename/Renumber/Retype Database - Code N 45
- Special Functions for Editing the List of Files Contained in a Database 49

Maintenance functions applying to databases are called from the Maintain Database menu that is called with the command `MAINTAIN DATABASE` or with function code M and object type code DA in a Predict main menu.

Purge Database - Code P

Predict objects of type Database are purged with the Purge function (code P).

You have two purge options, DELETE and SCRATCH.

DELETE

The DELETE option applies to all database types apart from IMS. The following objects are deleted:

- the database object
- all links to parent and child objects.
- Rules which apply to the individual database types are given below.

SCRATCH

The SCRATCH option deletes the following objects:

- files contained in this database and the related userviews
- fields of these files
- generated code of these files
- file relations based on these files
- links to/from the scratched objects.

Two lists will be displayed before a database is purged:

- A list of objects that will not be deleted because they are used in some other object which will not be deleted. This list will only be displayed if the Purge mode option in the session profile is set to Y. See *Customizing Predict with Profiles* in the section *Predict User Interface* in the *Introduction to Predict* documentation.
- A list of objects and generated code that will be deleted.

Confirmation of the purge operation is then requested. A list of all deleted objects and links will be displayed after execution of the delete operation.

Database-specific Rules

For Database Objects of Type Adabas

- A Purge operation is not executed if the database and files in the database are implemented.
- The Delete operation purges a database object and all links to related objects. All Adabas attributes for files which are contained in this database are purged or changed to default if the file is not contained in another database.
- File objects for which DDMs or table/cluster descriptions exist will not be purged.

For Database Objects of Type SQL

- A Purge operation is not executed if the database and files contained in the database are implemented.
- File objects for which DDMs or table/cluster descriptions exist will not be purged.

For Database Objects of Type IMS

- The Purge function will not be executed if UDFs exist for the IMS files.
- Delete is not available because Predict regards an IMS database object and the files contained in it as an integral unit.

Rename/Renumber/Retype Database - Code N

```
13:48:50            *****  P R E D I C T  *****            2007-05-31
                             - Rename Database -
Database ID ..... HNO-DA                              Added 2007-05-31 at 13:48
Database type ... Adabas                                 by HNO


     Enter new values

       Database ID .......... HNO-DA
       Database type .......* A Adabas
       Belongs to VM .......* HOME
       Run mode ............* V Vista
       Physical DBnr .......* 1234
       Vista access only .... Y (Y,N)

     Enter '.' to return to menu.
```

This function can be used to change

- Database ID
- database type
- the virtual machine that belongs to the database
- the run mode
- physical database number
- the Vista access only flag.

Depending on the database to be processed, messages indicating the possible Rename/Renumber/Retype options are displayed at the bottom of the screen.

The following topics are covered below:

- General Rules
- Changing the Database ID
- Changing the Database Type
- Linking the Database to another Virtual Machine
- Changing the Run mode Parameter
- Changing the Database Number
- Changing the Parameter Vista Access Only
- Messages

General Rules

- Changes to database attributes are also applied to file objects if applicable. For example: if a database is linked to another virtual machine, existing Vista elements of files linked to the database are adapted accordingly.
- Special rules apply when renaming/renumbering/retyping databases that are connected to implemented databases. Connecting documentation and external objects is described in the section *Handling of External and Documentation Objects* in the *External Objects in Predict* documentation.

Changing the Database ID

The new database ID must not already exist in the dictionary.

Changing the Database Type

- If files contained in the database are connected to implemented files, the database type and the database number (DBnr) cannot be changed.
- For a database of type C (conceptual) the following rules apply:
  - If all files contained in the database have the same type, the database type can be changed to this type.
  - If files contained in the database have different types, the database type must not be changed.
- All other database types can be changed to type C (conceptual) without restrictions.

Linking the Database to another Virtual Machine

The following rules apply to Adabas databases:

- Old virtual machine and new virtual machine are in the *same network*: the change is applied to the database and all files linked to the database.
- Old virtual machine and new virtual machine are in *different networks*: new Vista elements are created and/or existing Vista elements are purged. Additional confirmation is requested when purging Vista elements (as shown in the screen below).

```
13:13:38            *****  P R E D I C T  *****            2007-05-31
                           - Rename Database -
Database ID ..... HEB-NO-TRANS                    Modified 2007-05-31 at 13:17
Database type ... Adabas                                 by JPE

             +-------------------------------------------+
     Enter n !              A T T E N T I O N            !
             !                                           !
       Datab ! Old and new virtual machine are not in the !
       Datab ! the same network.                         !
       Belon ! Vista elements will be purged.            !
       Run m ! old NW: HEB-NW-TEST                       !
       Physi ! new NW: HEB-NW                            !
       Vista !                                           !
             ! Do you want to continue  N (Y/N)          !
     Enter ' +-------------------------------------------+


   File with phys. distribution type partitioned found.
   File with Vista element found.
```

- If the Database is connected to an implemented database the new virtual machine must be in the same network as the old virtual machine.

Non-Adabas databases can be linked to another virtual machine without restrictions.

Changing the Run mode Parameter

The following rules apply:

- Changing from Run mode parameter I (isolated) or L (local) to V (Vista) is not possible if a replicated or master file for Entire Transaction Propagator is linked to the database.
- Changing from Run mode parameter V (Vista) to I (isolated) or L (local) is not possible if:
  - files with Vista elements are linked to the database, or
  - Vista elements for the database exist, or
  - files with phys. distribution type P (partitioned) are linked to the database.

Changing the Database Number

The following rule applies:

- If files contained in the database are connected to implemented files, the database number (DBnr) cannot be changed.

Changing the Parameter Vista Access Only

The Vista Access Only flag indicates whether Adabas Vista is required to access files in a database. When creating a database object, Predict sets this flag to Y or N (according to the attributes of the database). The following rule applies:

- Setting the flag from N to Y: Only possible for databases of type V (Vista). Vista elements for files will be created if they do not already exist.

Messages

If prerequisites for renaming/renumbering/retyping databases are not met, one of the following messages is issued.

Implemented file exists in the database

If files contained in the database are connected to implemented files, the database type and the database number (DBnr) cannot be changed.

File with physical distribution type 'partitioned' found / File with Vista element found / Database Vista element found

At least one file defined for use with Vista (phys. distribution type or Vista element is specified) is linked to the database. The Run mode parameter therefore must be V (Vista); the database type can only be changed to C (conceptual).

File with PROPAGATOR type 'master' or 'replicated' found

At least one replicated or master file for ENTIRE TRANSACTION PROPAGATOR has been found. The Run mode parameter can therefore not be changed to V (Vista); the database type can only be changed to C (conceptual).

Different File types in the database

Files of different types are linked to the database. Database therefore must be of type C (conceptual) and cannot be changed to another type.

Special Functions for Editing the List of Files Contained in a Database

The following line commands apply when editing the file list of a database.

Editor Commands	
SORT LOG	Sort the list of file IDs into ascending order of their logical file numbers.
SORT PHY	Sort the list of file IDs into ascending order of their physical file numbers.
Line Commands	
.A	Calls the Modify Adabas Attributes screen for the file.
.T	Calls the Modify Vista element screen for the file.

7 Database Retrieval

- Database-Specific Retrieval Parameters 52
- Database-Specific Retrieval Functions 52
- Layout of Database Lists 54
- Output Options for Database Retrieval 55

Retrieval functions applying to database objects are called from the Database Retrieval menu that is called with the command `RETRIEVE DATABASE` or with function code R and object type code DA in a Predict main menu.

Note: Standard retrieval functions are described in the section *Retrieval* in the *Predict Reference* documentation.

Database-Specific Retrieval Parameters

Parameters	
Database of type	Limits the scope of the function to databases of a certain type. Enter an asterisk to display possible values at your site or see complete list of database types in the section ***Database Maintenance Menu***.
Database number	Limits the scope of the function to databases with the number specified.

Database-Specific Retrieval Functions

Explode IMS Database - Code I

Shows the hierarchical structure of an IMS/DL/I database. The level number before the file ID shows the level of the IMS/DL/I segment in the hierarchy. This function is only applicable to databases of type I. Command: `EXPLODE DATABASE`.

Databases with children - with Association Type Contains FI and Output Option Adabas size=Y

The following output is produced with function Display databases with children if you specify association type "Contains FI" and output option Adabas size=Y for databases and files of type Adabas.

```
13:13:02             *****  P R E D I C T  *****         2007-05-31
                     - Display Database with Children -

Database ID ........ DA-WITH-FILE
Type ............... Adabas, Isolated      Added 2007-05-31 at 10:25 by ARH
Physical DBnr ...... 57                 Modified 2007-05-31 at 10:42 by ARH

-------------------------------------------------------------------------
Adabas attributes                     Natural file numbers
  Maximal files ........                System file (FNAT) ...
  Checkpoint file ......                NAT-Security (FSEC) ..
  Adabas security ......                Predict (FDIC) .......
  Size of RABN ......... 0
  Distr. transaction ... N No
  Vista access only .... N

              --------------- Database primary sizes ---------------
                            Number of               Alternate RABN
          Device    Cylinder     RABN             Start     End
ASSO R1    3390       15           4032             1200      3200
ASSO R2    3390       14           3780             2000      3760
DATA R1    3380       12           1611
DATA R2    3380       13           1755


Cnt  File ID                            Type   Fnr    DDM Impl Other

   1 FI-A-001                            A     123
   2 FI-A-002                            A     124

------------------------------ Summary of sizes---------------------------

Type  Device   No. of RABN    Min. RABN  Undoc. RABN
ASSO   3390           7812         1942         5870
DATA   3380           3366         1205         2161

 *** End of report ***
```

The total ASSO and DATA sizes defined for the database are calculated and displayed in column Summary of sizes/No. of RABN.

The sum of the sizes of ASSO and DATA for the individual related files is calculated and displayed in column Min. RABN.

The number of available RABNs is displayed in the column Undoc. RABN, or a message is given indicating that the sizes of the files exceed the size available in the parent database.

The system also checks whether the devices specified for the files are also specified for the parent database.

Layout of Database Lists

The following list format applies when retrieving information on databases with output mode List.

```
13:37:03             *****  P R E D I C T  *****                2007-05-31
                             - List Database -
--------------------------------------------------------------------------------
Cnt  Database ID                            Type                P-DBnr Run Mode

  15 ARH-X                                  Informix Handler        24 Local
  16 ARH-Y1                                 Sybase Handler          22 Local
  17 * ARTICLE                              IMS
  18 AZ-PREDICT                             Adabas                  66 Isolated
  19 AZ-SAGPRD                              DB2

*** End of report ***
```

Meaning of Columns	
Database ID	ID of the database object. If the output option Mark implementation is set to Y, implemented objects are marked with an asterisk. For databases, "implemented" means that it is one of the following: ■ of type A and connected to a physical Adabas database, ■ of type D and connected to a physical DB2 database ■ of type P and its database number is defined in the NTDB macro as an Entire System Server database, ■ or of type I.
Type	The database type. See *Database Maintenance Menu*.
P-DBnr	The physical number of the database.
Vista Parm	Accessibility of Adabas databases using Adabas Vista. Possible values are listed in the section *Defining Basic Attributes of Databases*.

Output Options for Database Retrieval

Notes:

1. Unless output mode is S, the option Cover page is always valid.
2. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Adabas sizes	x		x				x		x				x			
Association attributes			x	x	x	x			x	x	x	x				
Attributes	x		x				x		x				x			
Connecting character				x							x					
Description	x		x	x			x		x	x			x			
Display modifier	x		x				x		x				x			
Dummy/Placeholder										x		x		x		x
Extract	x		x	x			x		x	x			x	x		
Keywords	x		x	x			x		x	x			x			
Mark implementation	x	x	x	x	x	x	x	x	x	x	x	x	x		x	
No. abstract lines	x	x	x	x	x	x	x	x	x	x	x	x	x		x	
Owner	x		x	x			x		x	x			x			
With users	x		x	x			x		x	x			x			
Show implementation	x		x				x		x				x			
Use Con-form	x		x	x			x		x	x			x			
User exit	x		x				x		x				x			

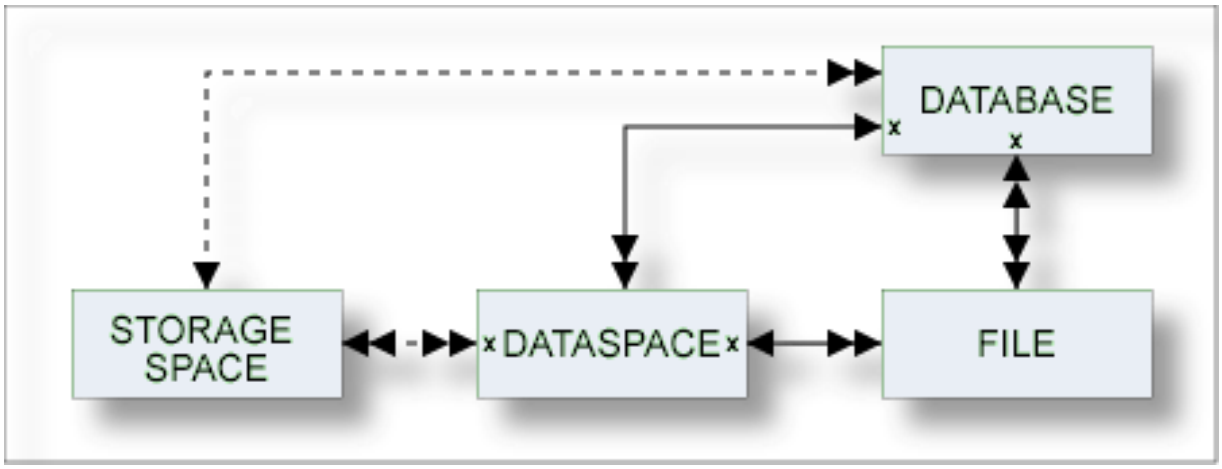
Retrieval Type	U		E				C				I	
Output Mode	D	L	T		X		L		D		T	
Current/Related	c	c	c	r	c	r	c	r	c	r	c	r
Adabas sizes	x											
Association attributes			x	x								
Attributes	x			x		x						

Retrieval Type	U		E				C				I	
Output Mode	D	L	T		X		L		D		T	
Current/Related	c	c	c	r	c	r	c	r	c	r	c	r
Connecting character				x		x				x		
Description	x					x				x		
Display modifier	x											
Dummy/Placeholder				x		x	x		x			
Extract	x			x		x			x	x		
Keywords	x			x		x				x		
Mark implementation	x	x	x	x	x	x		x		x	x	x
No. abstract lines	x	x		x		x		x		x		x
Owner	x			x		x				x		
With users	x									x		
Show implementation	x											
Use Con-form	x					x				x		
User exit	x											

III Dataspace

DB2 table spaces or SQL/DS DBspaces are documented with objects of type Dataspace.

Note: DB2 storagegroups are documented with objects of type **Storagespace**.



In the Predict metastructure a dataspace can have passive and active associations of the following types:

Valid passive association:	"Located in DA" (default passive association)
Valid active association:	"Contains FI" (default active association)

The description of object type Dataspace is organized under the following headings:

Maintaining Objects of Type Dataspace

Dataspace-Specific Maintenance

Dataspace Retrieval

8 Maintaining Objects of Type Dataspace

- Dataspace Maintenance Menu 60
- Adding a Dataspace 61
- Defining Basic Attributes of Dataspace - DB2 Mainframe 61
- Defining Basic Attributes of Dataspace - SQL/DS 67
- Defining Basic Attributes of Dataspace - DB2 Open Systems 68

Dataspace Maintenance Menu

The Dataspace Maintenance menu is called with function code M and object code DC in a Predict main menu or the command `MAINTAIN DATASPACE`.

```
   13:34:10             *****  P R E D I C T  *****             2007-05-31
   Plan   0               - (DC) Dataspace Maintenance -             Profile XYZ

    Function                                Function

    A  Add a Dataspace                      D  Display Dataspace
    C  Copy Dataspace                       L  Link children
    M  Modify Dataspace                     S  Select Dataspace from a list
    N  Rename Dataspace                     J  Physical Attributes
    P  Purge Dataspace


   Function .........

   Dataspace ID .....                                   Attributes ....*
   Copy ID ..........
   Located in DA ....

   Restrictions ....*   Profile XYZ ,used               Association ...*

   Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Next
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Executes one of the maintenance functions. Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. The function Purge is described in the section *Dataspace-Specific Maintenance*.

Adding a Dataspace

The following screen is displayed for the Add a Dataspace function.

```
13:00:49            *****  P R E D I C T  *****            2007-05-31
                           - Add a Dataspace -
Dataspace ID .... XYZ-DC




        Dataspace type ....* D DB2
        Located in DA .....*
```

Parameters		
Dataspace type	D	DB2 mainframe
	S	SQL/DS
	2	DB2 open systems
	A second input screen is displayed depending on the type. The screens are described below.	
Located in DA	The ID of the database which contains the dataspace. Applicable to DB2 dataspaces.	

Defining Basic Attributes of Dataspace - DB2 Mainframe

The following screen applies to DB2 mainframe dataspaces (type D).

```
15:56:34                ******  P R E D I C T  *****                  2020-05-25
                            - Add a Dataspace -
Dataspace ID .... XYZ-DC
Type ........... DB2 mainframe
Located in DA ... DB2-V12A
Keys ..                                                                Zoom: N

Dataspace attributes
  Tablespace name ..
Physical attributes in <Default Server>                 (new)
  Type ............*   Simple                 Member cluster ....   (Y,N)
  Nr of partitions .                          Large ............ N (Y,N)
  Buffer pool .....*                          Partition size ...*
  Locksize ........*   (none)                 Pages per segment .
  Close option .....   (Y,N)                  Logged ............   (Y,N)
  Lockmax ..........                          Pagenum ..........*   (none)
  Lockpart .........   (Y,N)                  Insert algorithm .*   (none)
  Maxrows ..........
  CCSID ...........*   (none)
Abstract     Zoom: N


  Additional attributes ..* N          Associations ..* N                    ↩
```

Parameters		
Tablespace name	Name of the table space in DB2.	
Type	Table space type.	
	S	Segmented.
	P	Partitioned.
	R	Range partitioned.
	G	Partition by growth.
	''	Simple. This is the default setting.
Nr of partitions	Number of partitions used by the table space (corresponding to the `NUMPARTS` parameter, max. 4096) for table space type P or R. If 0 is specified, the table space is not partitioned. Maximum number of partitions used by the table space (corresponding to the `MAXPARTITIONS` parameter, max. 4096) for table space type G. Partitions can be defined Partitions can be defined explicitly or with default values (see **Using/free clause** below) if table space type is either P or R. Partition definitions are used when generating table spaces from Predict dataspace objects.	
Large	Identifies a table space as large.	
	Y	Yes.
	N	No.

Parameters		
Buffer pool	Name of the buffer pool to be associated with the table space. Enter asterisk for list of valid values.	
Partition size	Only valid for partitioned tablespaces. Enter asterisk (*) for valid values.	
Locksize	Locking level for the table space. Valid values:	
	A	Any level locking.
	P	Page level locking.
	R	Row level locking.
	S	Table space level locking.
	T	Table level locking (only valid for segmented DS).
Pages per segment	How many pages are to be assigned to each segment (parameter SEGSIZE) for table space type S, G or R. Pages per segment must be zero for table space type ' ' (Simple) or P.	
Logged	blank	Not specified. This is the default setting.
	Y	Yes.
	N	No.
Pagenum	Specifies the type of page numbering that is used when you alter a partition-by-range table space.	
	' '	None. This is the default setting.
	R	Relative.
	A	Absolute.
Insert algorithm	Specifies the algorithm that is used when rows are inserted into tables in this table space.	
	' '	None. This is the default setting.
	1	Level 1.
	2	Level 2.
Close option	Y	The data sets which support the table space are closed when nobody is using the table space.
Lockmax	The maximum number of pages or row locks an application can hold simultaneously in the table space. Valid values:	
	blank	
	SYSTEM	
	0 - 2,147,483,647.	
	If parameter **Locksize** is set to S or T, Lockmax must be set to 0.	
Lockpart	Partition locking. Valid values:	
	blank	not specified.
	Y	Yes.
	N	No.
Maxrows	The maximum number of rows.	

Parameters		
CCSID	Encoding scheme. Valid values:	
	blank	not specified.
	A	ASCII.
	E	EBCDIC.
	U	Unicode.
Member cluster	Specifies that data inserted by the INSERT statement is not clustered by the implicit clustering index (the first index) or the explicit clustering index. Instead, DB2 chooses where to locate the data in the table space based on available space. Valid values:	
	blank	not specified.
	Y	Yes.
	N	No.
Additional attributes	Y	The partitions of the table space are to be defined. The following two options are available: ■ A default definition can be specified (the Using/free clause). The default values are used for partitions that are not defined explicitly. ■ Individual partitions can be defined. The screens to define individual partitions follow the screen for the definition of the default values.

Definition of Using/Free Clause

The values specified in the Definition of using/free clause section are used as default values.

Partitions can be defined explicitly in subsequent screens. See below.

```
16:32:28                 *****  P R E D I C T  *****                2015-05-05
                             - Modify Dataspace -
Dataspace ID .... XYZ-PCTUPDP
Server Name ..... <Default Server>

Definition of using/free clause
     VSAM catalog name .......
  or Storagespace ...........*

     Primary attributes
       Free pages ............
       Percentage free ....... 11
              for update ..... -1
       Compress option .......   (Y,N)
       GBPCACHE .............*
       TRACKMOD ..............   (Y,N)

     Additional for storagespace
       Primary allocation ....
       Secondary allocation ..
       Erase option ..........   (Y,N)

* Additional attributes ..* y          Associations ..* N
```

Parameters	
VSAM catalog name	Name of the VSAM catalog containing an entry for the data sets of the table space. Must not be specified if the parameter Storagespace is specified.
Storagespace	Name of the storagespace for the table space documented with the Predict Dataspace object. Must not be specified if the parameter VSAM catalog is specified.
Primary attributes	
Free pages	How often pages are to be left free when loading or reorganizing table spaces or partitions. Max. value is 255. Default is 0, leaving no free pages.
Percentage free	Percentage of each page to be left free. Equivalent in DB2 is the PCTFREE option.
for update	Percentage of each page that is reserved to be used only by future update operations. Equivalent in DB2 is the PCTFREE FOR UPDATE option. When you specify -1, DB2 uses real-time statistics to automatically calculate how much free space to reserve for updates. This is the default.
Compress option	Specifies whether data compression applies to the rows of the tablespace.
GBPCACHE	Only relevant in a data sharing environment. Specifies what pages of the table space or partition are written to the group buffer pool. Leave this field blank or enter: C — Changed. Only pages that have been changed are written to the group buffer pool. A — All pages are written. N — No pages are written to the group buffer pool.
TRACKMOD	Specifies whether DB2 tracks changed pages in the space map pages.

Parameters		
	Y	Changed pages are tracked in the space map pages.
	N	Changed pages are not tracked.
Additional for storagespace		
Primary allocation	Primary space allocation for DB2 defined data sets.	
Secondary allocation	Secondary space allocation for DB2 defined data sets.	
Erase option	Determines if DB2 defined data sets are to be erased when the table space is dropped:	
	N	Do not erase data sets (default).
	Y	Erase data sets.

Defining Partitions

Each individual partition can be defined in the Definition of partitioned dataspace section. The maximum number of partitions is 4096.

Two partitions can be defined in one screen. To modify a specific partition, skip previous definitions by pressing ENTER.

```
14:58:46                 *****  P R E D I C T  *****                2021-01-03
                             - Modify Dataspace -
Dataspace ID ... XYZ-DC                            Modified 2021-01-03 at 14:58
Server Name .... <Default Server>                        by XYZ
--------------------- Definition of partitioned dataspace ---------------------
Partition
1    VSAM catalog .          or Storagespace *
     Primary attributes                    Additional for storagespace
       Free pages ..........                 Primary allocation ...
       Percentage free .....                 Secondary allocation .
              for update ...                 Erase option .........   (Y,N)
       Compress option .....   (Y,N)
       GBPCACHE ...........*                 DSSIZE (G) ..........
       TRACKMOD ...........   (Y,N)
2    VSAM catalog .          or Storagespace *
     Primary attributes                    Additional for storagespace
       Free pages ..........                 Primary allocation ...
       Percentage free .....                 Secondary allocation .
              for update ...                 Erase option .........   (Y,N)
       Compress option .....   (Y,N)
       GBPCACHE ...........*                 DSSIZE (G) ..........
       TRACKMOD ...........   (Y,N)
  Additional attributes ..* S          Associations ..* N    Scroll to .. 1
```

Parameters	
DSSIZE	Specifies the maximum size (in Gbyte) for each partition of a partitioned index.
For all other values/options see previous table above for a description of the parameters.	

Defining Basic Attributes of Dataspace - SQL/DS

The following screen applies to SQL/DS dataspaces (type S).

```
13:37:47            *****  P R E D I C T  *****            2007-05-31
                           - Add a Dataspace -
 Dataspace ID .... HNO-SQ
 Type ............ SQL-DS
 Keys ..                                                          Zoom: N

 Dataspace attributes
   Tablespace name .........
 Physical attributes in <Default Server>                (new)
   Private dataspace .......   (Y/N)
   Size for header .........
   Size for dataspace ......
   Percentage for indices ..
   Percentage free .........
   Lock size .............*
   Storage pool number .....

 Abstract     Zoom: N
```

Parameters	
Tablespace name	Identifier of the table space and name of the DBspace in SQL/DS.
Private dataspace	Y: SQL/DS DBspace is private. N: Dataspace is public.
Size for header	Number of 4096-byte logical pages reserved for header.
Size for dataspace	Size reserved for the dataspace.
Percentage for indices	Percentage of the reserved space that can be used for indexes.
Percentage free	Percentage of reserved space to be kept free.
Locksize	Locking level for the dataspace. Valid values: A: Any level locking P: page S: dbspace

Parameters		
	R	row
	T	table
Storage pool number	Storage pool number. This parameter tells SQL/DS to acquire the dbspace from a specified storage pool.	

Defining Basic Attributes of Dataspace - DB2 Open Systems

The following screen applies to DB2 open systems dataspaces (type 2).

```
13:37:47            *****  P R E D I C T  *****            2007-05-31
                           - Add a Dataspace -
 Dataspace ID .... HNO_TEST
 Type ............ DB2 open systems

 Keys ..                                                          Zoom: N

 Dataspace attributes
  Tablespace name ..
 Physical attributes in <Default Server>                 (new)
  Storagespace ....*
  Data type .......*                          Transfer rate .... 0.90
  Page size .......*                          Dropped table .... N (Y,N)
  Space type ......*
  Extent size ......
  Prefetch size ....
  Buffer pool ......
  Overhead .........
 Abstract     Zoom: N



  Additional attributes ..* N          Associations ..* N
```

Parameters		
Tablespace name	Name of the tablespace in DB2.	
Storagespace	Name of the storagespace for the tablespace documented with the Predict dataspace object.	
Data type	Valid data types:	
	R	Regular
	L	Large
	U	User temporary

Parameters	
	S: System temporary blank: not specified
Page size	Defines the size of pages used for the tablespace. Valid values: 4K, 8K, 16K, 32K, 4096, 8192, 16384, 32768 or not specified.
Space type	Specifies how the tablespace is to be managed: S: System managed D: Database managed
Extent size	Specifies the number of PAGESIZE pages that will be written to a container before skipping to the next container. Valid values are 0 or integer or integer with unit K, M or G.
Prefetch size	Specifies the number of PAGESIZE pages that will be written to a container before skipping to the next container. Valid values are 0 or integer or integer with unit K, M or G.
Buffer pool	Name of the buffer pool to be associated with the tablespace.
Overhead	Specifies the I/O controller overhead and disk seek and latency time (in milliseconds).
Transfer rate	Specifies the time to read one page into memory (in milliseconds).
Dropped table	Specifies if dropped tables in the tablespace may be recovered. Valid values: Y: Yes N: No blank: Not specified

Definition of System Managed Containers

The following screen applies when defining system managed containers.

```
15:25:25              *****  P R E D I C T  *****             2007-05-31
                             - Modify Dataspace -
Dataspace ID ... HEB-DC-2                          Modified 2004-07-28 at 09:37
Server Name .... <Default Server>                        by HEB

------------------------ Definition of system containers ----------------------

Container
1   Name ......... aaa                                                  Zoom N

    Database partition number
      From ....... 0
      To ......... 0

2   Name .........                                                      Zoom N

    Database partition number
      From .......
      To .........



* Additional attributes ..* S          Associations ..* N     Scroll to .. 1   ↩
```

Parameters	
Container	Specifies the container for a tablespace.
Partition	Specifies the partitions on which the containers are created in a partitioned database.

Definition of Database Managed Containers

The following screen applies when defining database managed containers.

```
15:41:18              *****  P R E D I C T  *****             2007-05-31
                             - Modify Dataspace -
Dataspace ID ... HEB-DC-2                          Modified 2004-07-28 at 09:37
Server Name .... <Default Server>                        by HEB
----------------------- Definition of database containers ---------------------
Container
1   Name ......... aaa                                                   Zoom N
    Type ........* D Device
    Pagesize .....

    Database partition number
     From ....... 0
     To ......... 0

2   Name .........                                                       Zoom N
    Type ........*
    Pagesize .....

     Database partition number
       From .......
       To .........

* Additional attributes ..* S          Associations ..* N     Scroll to .. 1   ↩
```

Parameters	
Container	Specifies the container for a tablespace.
Type	Specifies the type of the container. Valid values: F — File D — Device
Pagesize	Specifies the container size.
Database partition number	Specifies the partitions on which the containers are created in a partitioned database.

9 Dataspace-Specific Maintenance

- Purge Dataspace - Code P 74

When maintaining dataspaces, only standard maintenance functions are needed. However, specific rules apply when purging objects of type dataspace. These rules are described below.

Purge Dataspace - Code P

If you confirm the purge operation with DELETE, the following objects are deleted:

- the dataspace object
- all links to child objects and from parent objects
- the connection from the dataspace to the DB2 database is undone. All DB2 tables contained in this dataspace are removed from the file list of the corresponding DB2 database object.

10 Dataspace Retrieval

- Layout of Dataspace Lists 76
- Output Options for Dataspace Retrieval 76
- Output Options for Dataspace Retrieval - Continued 77

Information on dataspace objects is retrieved with standard retrieval functions. These are described in the section *Retrieval* in the *Predict Reference* documentation.

Layout of Dataspace Lists

The following list format applies when retrieving information on dataspaces with the output mode List.

```
   13:46:51            *****  P R E D I C T  *****            2007-05-31
                               - List Dataspace -                   Page:   4


-------------------------------------------------------------------------------
   Cnt  Dataspace ID                       Tablespace name        Part Segsize

     41 PD-DC2                             PD_DC2                  0
     42 PD-DC3                             PD_DC3                  0
     43 PD-DC4                             PD_TABLE                0
     44 PD-D1                              PDPD                    0
     45 PD-TABSPACE                        TABSPACE                0
     46 * PRDSUPDB-BRUNO                   BRUNO                   0
     47 * PRDSUPDB-FSTTEST                 FSTTEST                 8
```

Meaning of Columns	
Dataspace ID	ID of the Predict dataspace object.
Tablespace ID	Name of the DB2 tablespace.
Part	Number of partitions.
Segsize	Size of segments.

Output Options for Dataspace Retrieval

Notes:

1. Unless output mode is S, the option Cover page is always valid.
2. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Association attributes			Y	Y	Y	Y			Y	Y	Y	Y				
Attributes	Y		Y				Y		Y				Y			
Connecting Character				Y						Y						
Description	Y		Y	Y			Y		Y	Y			Y			
Display modifier	Y		Y				Y		Y				Y			
Dummy/Placeholder										Y		Y		Y		Y
Extract	Y		Y	Y			Y		Y	Y			Y	Y		
Keywords	Y		Y	Y			Y		Y	Y			Y			
Mark implementation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
No. abstract lines	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
Owner	Y		Y	Y			Y		Y	Y			Y			
With users	Y		Y	Y			Y		Y	Y			Y			
Show implementation	Y		Y				Y		Y				Y			
Use Con-form	Y		Y	Y			Y		Y	Y			Y			
User exit	Y		Y				Y		Y				Y			

Output Options for Dataspace Retrieval - Continued

Retrieval Type	U		E				C					
Output Mode	D	L	T		X		L		D			
Current/Related	c	c	c	r	c	r	c	r	c	r		
Association attributes			Y	Y								
Attributes	Y			Y		Y						
Connecting character				Y		Y						
Description	Y					Y				Y		
Display modifier	Y											
Dummy/Placeholder				Y		Y	Y		Y			
Extracts				Y		Y			Y	Y		
Keywords	Y			Y		Y				Y		
Mark implementation	Y	Y	Y	Y	Y	Y	Y	Y		Y		

Retrieval Type	U		E				C					
Output Mode	D	L	T		X		L		D			
Current/Related	c	c	c	r	c	r	c	r	c	r		
No. abstract lines	Y	Y		Y		Y		Y		Y		
Owner	Y			Y		Y				Y		
With users	Y									Y		
Show implementation	Y											
Use Con-form	Y					Y				Y		
User exit	Y											

IV Extract

An object of type Extract in Predict fulfills two functions:

- to group objects logically
- to determine the objects to be transferred with the Predict Coordinator.

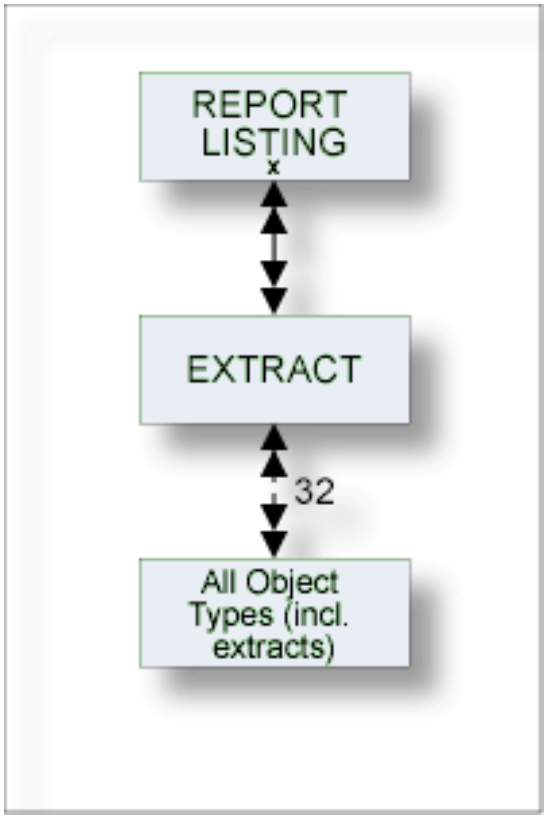
An object can be contained in a maximum of 32 extracts. The number of objects in an extract is virtually unlimited. An extract can contain other extracts - including itself.

Extracts #SAG-TRANSFER and #SAG-ERROR are created automatically by the Coordinator. See the Predict Coordinator documentation.

In the Predict metastructure, an extract can have passive and active associations of the following types:

Valid passive association:	"Contained in RT" (association is created automatically)
Valid active association:	no predefined association

When you transfer objects with the Predict Coordinator, a report listing is created automatically and the extract containing the objects to be transferred is linked as a child object to this report listing. See the Predict Coordinator documentation.



The description of object type Extract is organized under the following headings:

Maintaining Objects of Type Extract

Extract-Specific Maintenance Functions

Extract Retrieval

11 Maintaining Objects of Type Extract

- Extract Maintenance Menu 82
- Defining Basic Attributes of Extract 83

Extract Maintenance Menu

The Extract Maintenance menu is called with function code M and object code ET in a Predict main menu or with the command `MAINTAIN EXTRACT`.

```
   10:09:17             *****  P R E D I C T  *****            2007-05-31
   Plan   0                - (ET) Extract Maintenance -             Profile HNO

    Function                                Function

    A  Add an Extract                       D  Display Extract
    C  Copy Extract                         L  Link children
    M  Modify Extract                       S  Select Extract from list
    N  Rename Extract                       E  Edit/link objects
    P  Purge Extract                        B  Build/extend an Extract
    T  Operate on Extracts
    U  Export an Extract

   Function ..............
   Extract ID ............                                  Attributes ....*
   Copy ID ...............


   Restrictions .........*   Profile HNO,empty              Association ...*

   Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters	
Function	Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. Extract-specific maintenance functions are described in the section ***Extract-Specific Maintenance Functions***.
Extract ID	ID of the extract to be processed. See ***Naming Conventions***.

Note: For parameters not listed here, see ***Global Attributes***.

Defining Basic Attributes of Extract

The following screen is displayed for functions Add/Copy/Modify Extract:

```
13:34:03            *****  P R E D I C T  *****            2007-05-31
                            - Modify Extract -
Extract ......... HNO-ET                          Modified 2007-05-31 at 09:16
                                                         by HNO
Keys ..                                                              Zoom: N



 Abstract    Zoom: N









 Additional attributes ..* N          Associations ..* N          Objects ... N
```

Parameters	
Extract	ID of the extract.
Additional Attributes	Valid values: blank — Base attributes. W — Description. O — Owner.
Objects	Y — If you are using the Software AG Editor, the Extract Object Editor is called. See *Extract Object Editor*. If you are using the Natural Editor, the function Link Objects to Extract is called. See *Link Objects to Extract - Code E*.

Note: For parameters not listed here, see *Global Attributes*.

12 Extract-Specific Maintenance Functions

- Copy Extracts - Code C 86
- Operate on Extracts - Code T 86
- Export an Extract - Code U 90
- Edit/Link Objects - Code E 91
- Extract Object Editor 91
- Link Objects to Extract - Code E 96
- Build/Extend an Extract - Code B 98
- Purge Extract - Code P 101

Copy Extracts - Code C

If you copy an extract which contains objects, an asterisk is displayed next to field `Edit: Objects`. Enter Y in this field. The system behavior depends on the editor you are using.

- If you are using the SAG Editor, the Extract Object Editor is called. See ***Extract Object Editor***. To copy the extract with objects, this list must be cataloged, otherwise the extract will be copied without objects.
- If you are using the Natural Editor, the Link Objects to Extract function is called. See ***Link Objects to Extract - Code E***. If you confirm the object list that is displayed with ENTER, the extract is copied with objects.

Operate on Extracts - Code T

With this function, the result of a set operation is added to the objects in the current extract (if parameter Drop existing objects is set to N) or the extract will correspond exactly to the result of the set operation (parameter Drop existing objects=Y). See ***Overview of Operations available for Function Operate on Extracts***.

An object may only be contained in a maximum of 32 extracts. If an operation would lead to one or more objects being contained in more than 32 extracts, the object(s) already contained in 32 extracts are displayed and the user has the following possibilities:

- the objects are not entered in the object list of the extract, or
- the original object list is restored.

```
13:46:47             *****  P R E D I C T  *****             2007-05-31
                          - Operate on Extracts -
Extract ......... HNO-ET                              Added 2007-05-31 at 13:04
                                                   Modified 2007-05-31 at 13:41




Operation ..............*
Drop existing objects ... N (Y/N)

Search criteria
  Extract ID ............


Restrictions ...........*   Profile HNO ,empty
```

Parameters		
Extract	ID of the extract to be processed with this operation.	
Operation	Enter one of the following values:	
	U	Union. You can select any number of extracts. All objects in the selected extracts are added to the current extract.
	D	Difference. Mark one extract with X, the other with Y. Objects that are contained in extract X but not contained in extract Y are added to the current extract.
	I	Intersection. You can enter any number of extracts (but at least two). Objects that are contained in all of the selected extracts are added to the current extract. See *Overview of Operations available for Function Operate on Extracts*.
Drop existing objects	Y	Existing objects are removed from the object list of the extract.
	N	New objects are added to existing objects in the extract.
	This parameter must be specified.	
Search criteria		
Extract ID	With this selection criterion you can limit the scope of objects to be displayed for selection.	
	blank	All extracts are displayed for selection.
	ABC*	All extracts starting with ABC are displayed for selection.
	A unique extract ID makes sense only for the operation Union, because for Difference you must specify two and for Intersection you need at least two extracts.	
Restrictions	Additional criteria can be selected to restrict the scope of extracts to be processed.	

Selecting Extracts

Enter the parameters above to display a list of extracts which meet the selection criterion Extract ID and any restrictions you may have entered. See example below.

```
13:35:03              *****  P R E D I C T  *****             2007-05-31
Plan  11                      - Select Extract -


Cmd  Extract ID                              No. of Ref.

 __  HNO-ET-123                                  1
 __  HNO-ET-124                                  2
 __  HNO-ET-2                                    4
 __  HNO-ET1                                     1
 __  HNO-ET2                                     2
 __  HNO-ET3                                     0
```

- **Union**
  For the operation Union you can select any number of extracts by marking them with /, X or S in the Cmd column.
- **Intersection**
  For the operation Intersection you can select any number of extracts - but at least two - by marking them with /, X or S in the Cmd column.
- **Difference**
  For the operation Difference you must mark one extract with X and one with Y.

If you enter another command in the Cmd column, this command is added to the workplan. Enter an asterisk in this column to display the valid commands.

An Object can be contained in up to 32 extracts.

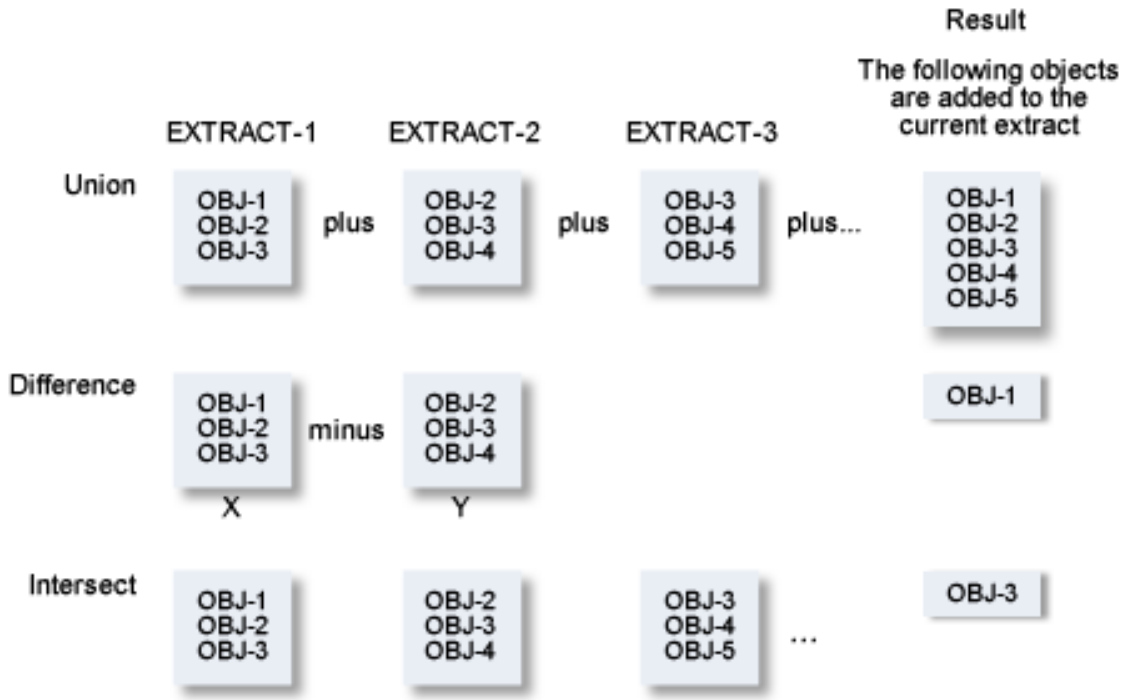
Objects which you want to add to the object list of the current object, but which are contained in 32 extracts already, are listed as shown in the screen below.

```
13:27:32              *****  P R E D I C T  *****             2007-05-31
                            - Operate on Extracts -

 Following objects are already in 32 Extracts:

 Program ........... GER-PR
 Program ........... ARH-PR
```

If you confirm this list with ENTER, a window appears in which you can enter a backout option.

Enter backout option	Y	Terminate the operation. No new objects are added to the current extract.
	N	Continue the operation. Objects linked to fewer than 32 extracts are linked to the current extract. Objects already linked to 32 extracts are not added to the current extract.

Overview of Operations available for Function Operate on Extracts



Export an Extract - Code U

This function transfers data from a Predict environment to an ALF file. Workfile 1 is the transfer medium. The following default parameter settings apply:

```
With code   N
With profile   N
With internal ID  Y
Include Extracts  N
Target environment S
```

The parameters are described in detail under *Export* in the section *Coordinator Functions* in the *Predict Coordinator* documentation.

Edit/Link Objects - Code E

With this function you skip the Modify Extract screen to edit the object list directly. The system behavior depends on which Editor is activated.

- If you are using the Software AG Editor, the Object List Editor is called. See ***Extract Object Editor***.
- If you are using the Natural Editor, the function Link Objects to Extract is called. See ***Link Objects to Extract - Code E***.

Extract Object Editor

This editor is available when you are using the Software AG Editor. This editor is called

- with function code E from the Extract Maintenance menu
- with the functions Add/Copy/Modify Extract: by entering Y in the field `Objects`
- with the command `EDIT EXTRACT OBJECTS` *`Extract-id`*.

```
10:20:26                    - Extract : HNO-ET -                      2007-05-31
        Extract object                  Type Subtype
 *****  ************************ top of list ***********************
 00001  HNO-BT                            FI    B
 00002  HNO-D2                            FI    D
 00003  HNO-A                             FI    A
 00004  HNO-H                             DA    E
 00005  HNO-D                             FI    D
 00006  HNO-E                             FI    E
 *****  ********************** bottom of list **********************
```

All functions of the Software AG Editor are available. See the section *Editors in Predict* in the *Predict Reference* documentation.

Meaning of Columns	
Extract object	ID of the object contained in the extract.
Type	Object type of the object. If you enter objects manually, you must enter ID and type.
Subtype	Subtype of the object (if applicable). If you enter ID and object type manually, the subtype is entered automatically. If an object type does not have any subtypes, this column is blank. Dummies are marked with a question mark.

Selecting Objects

With the SEL command you can

- add objects of a specific type to the extract, see below
- add objects of any type to the extract, see *Adding Objects of any type*.

With the line command H you can add objects to a specific position in the list.

Adding objects of a specific type

Enter the SEL command. The following screen appears in which you must enter an object type.

```
10:40:29             *****  P R E D I C T  *****             2007-05-31
Plan  11                 - Object Selection Menu -                 Profile HNO

Extract ID ......... HNO-ET
                                                   Modified 2007-05-31 at 10:07
                                                          by HNO
Select object type .............*

Select object type .............*
```

The following screen appears, for example, if you specify object type DA.

```
11:02:31             *****  P R E D I C T  *****             2007-05-31
Plan  11                - Database Selection Menu -                Profile HNO

Extract ID ......... HNO-ET
                                                     Modified 2007-05-31 at 10:07
                                                           by HNO
Select object type ............... DA  ( Database )

Retrieval type .....* D
Output mode ........* S Select

Search criteria
  Database ID .......                                Database of type*
  Belongs to VM .....                                Database number .





Restrictions .......*   Profile HNO,used
                                                     Association ....*
```

Alternatively you can enter one of the following commands in the Extract Object Editor:

- `SEL DA`, to restrict the selection to objects of type DA, or
- `SEL DA ABC*`, to restrict the selection to objects of type DA which start with ABC. If only one object starts with ABC, the Database Selection Menu is skipped.

From this screen you can execute any retrieval function for which the output mode Select is valid. For Databases, for example, the following functions can be executed:

- Databases
- Dummy/Placeholder databases
- Databases with no parent
- Databases with no child

You can limit the scope of the function using selection criteria and output options. All objects which meet the selection criteria and output options are listed.

```
08:13:54             *****  P R E D I C T  *****             2007-05-31
Plan  11                    - Select Database -

Cmd  Database ID                        Type              P-DBnr  Run Mode

 __  HNO-DA1                            Adabas               122  Isolated
 __  HNO-H                              Gen. SQL Handler     111  Local
 __  HNO-LEASY                          Other Handler        254  Local
 __  JPE-10                             Conceptual
```

From this list you can either

- select objects with /, S or X in the Cmd column to add them to the extract, or
- add functions to the workplan by entering a command other than /, S or X in the Cmd column. Enter an asterisk in the Cmd column to display the commands valid for the particular object.

Adding Objects of any type

To add objects of any type to the extract, enter one of the following commands in the Extract Object Editor:

- `SEL ALL`, or
- `SEL`, and leave the field `Select object type` in the Object Selection Menu empty.

The following screen appears:

```
10:38:50             *****  P R E D I C T  *****             2007-05-31
 Plan  11                 - Object Selection Menu -               Profile HNO

 Extract ID ......... HNO-ET
                                                   Modified 2007-05-31 at 10:07
                                                         by HNO
 Select object type ..............     ( All objects )

 Retrieval type .....* D
 Output-mode ........* S Select

 Search criteria
   Object ID .........
```

For object type All, only two retrieval functions are possible:

- Objects (Code D)
- Dummy Objects (Code C)

Objects of all types (except Field and Owner) are displayed for selection.

The only selection criterion is Object ID. With asterisk notation you can specify a range of object IDs.

You can only specify restrictions valid for all object types:

- Keywords
- Owner
- in Extract
- containing the string
- from date

A selection screen is displayed for each object type containing all objects that meet the selection criterion Object ID and any restrictions specified.

From this list you can

- select objects to be included in the extract by marking them with /, X or S in the Cmd column, or
- put functions in the workplan. Enter a command other than /, X or S in the Cmd column. Enter an asterisk in this column to display the commands valid for the respective object type.

All objects selected are added to the extract.

Extract-Specific Editor Commands

SORT N[AME]	Objects are sorted by columns Extract object and Type.
SORT [[T]YPE]	Objects are sorted by columns Type and Extract object.

Saving the Object List

When you have put all objects you require in the extract, enter `CAT` or `SAVE` to save the object list.

The following objects may not be added to the extract:

- duplicate objects
- non-existent objects (for example due to a typing error when adding objects manually)

If any duplicate or non-existent objects are contained in the list, the following screen appears:

```
13:25:49                    - EXTRACT : HNO-ET -                    2007-05-31
       EXTRACT OBJECT                   TYPE SUBTYPE
00002  HNO-FI2                            FI



       Correct the error,
       hit     ENTER to return to the editor
       or enter  D   to remove object

DIC2519 FILE DOES NOT EXIST.
```

This screen offers you the following possibilities:

- Correct the error by overwriting an incorrect object ID or changing the type.
- Remove the object from the list by entering D in the prefix area on the left of the screen.
- Return to the editor by pressing ENTER.

When the list is cataloged, the extract will be added to every object in the list.

Link Objects to Extract - Code E

With this function you can add objects of a specific type to the current object or remove objects from this extract. This function is available if you are using the Natural Editor and is called using one of the methods below:

- with function code E in the Extract Maintenance menu
- with the function Add/Copy/Modify Extract: by entering Y in the `Objects` field.
- with the command `EDIT EXTRACT OBJECTS` *`Extract-id`*.

The following screen appears:

```
15:13:20             *****  P R E D I C T  *****             2007-05-31
Plan  11                - Link Objects to Extract -

Extract ID ....... HNO-ET                         Modified 2007-05-31 at 13:20
                                                         by HNO

Link to object type ..*
```

Enter an object type to add objects of this type to the extract.

Note: With this function you cannot enter objects of all object types in a single operation.

If you specify object type DA, for example, the following screen appears:

```
15:48:34             *****  P R E D I C T  *****              2007-05-31
Plan  11                - Link Objects to Extract -

Extract ID ....... HNO-ET                          Modified 2007-05-31 at 13:20
                                                          by HNO

Link to object type ..* DA ( Database )


Search criteria
  Database ID ......... *
  Type ...............*
  Database number .....
  Belongs to VM .......




Restrictions .........*   Profile HNO ,used              List option ....* A


Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Limiting the Scope of Objects for Selection

The following possibilities are available to limit the scope of selection.

- **Search Criteria and Restrictions**
  The available search criteria depend on the object type. For object type database, for example, you can restrict the selection with the criteria Database ID, Type, Database number and "Belongs to VM". You can also specify Restrictions to further limit the selection.
- **List Option**
  This parameter determines which objects are displayed for selection.

List option	A	All objects that meet the selection criteria and restrictions are displayed for selection.
	L	Only objects that meet the selection criteria and restrictions and that are linked to the current extract are displayed for selection.
	U	Only objects that meet the selection criteria and restrictions and that are not yet linked to the current extract are displayed for selection.

The following screen appears for object type DA:

```
09:52:53             *****  P R E D I C T  *****             2007-05-31
                         - Link Objects to Extract -

 Extract ID ....... HNO-ET

  CMD L Database                          Type              P-DBnr  Run Mode

   _    HNO-DA1                           Adabas               122  Isolated
   _  L HNO-H                             Gen. SQL Handler     111  Local
   _    HNO-LEASY                         Other Handler        254  Local
```

Meaning of Columns		
CMD	Enter one of the following commands:	
	L	Link the object to the current extract.
	U	Unlink the object from the current extract.
L	An L in this column indicates that the object is already contained in the current extract.	

The other columns are type-dependent.

Build/Extend an Extract - Code B

With this function you can create or extend the object list of an extract. The following screen is displayed:

```
10:56:01              *****  P R E D I C T  *****              2007-05-31
Plan   0                 - Build/extend an Extract -                Profile HNO

Extract ID ...... HNO-ET                              Added 2007-05-31 at 09:54
                                                   Modified 2007-05-31 at 10:02

Build Extract for object type ..*
```

Parameters		
Extract ID	ID of the extract whose object list is to be added or extended.	
Build extract for object type	Enter one of the following values here:	
	code	The two-character code of a predefined or user-defined object type. Objects of this type are added to the extract. See ***Build/Extend an Extract for a specific object type***.
	blank	All object types. See ***Build/Extend an Extract for all object types***.

Build/Extend an Extract for a specified Object Type

This example shows the screen for the object type Database.

```
14:37:38              *****  P R E D I C T  *****              2007-05-31
Plan  11                 - Build/extend an Extract -                Profile HNO

Extract ID ......... HNO-ET
                                                   Modified 2007-05-31 at 13:40
                                                         by HNO
Build Extract for object type ..* DA  ( Database )

Retrieval type .....*
Output mode ........* S Select

Search criteria
  Database ID ....... *                                Database of type*
  Belongs to VM .....                                  Database number



Drop existing objects N (Y,N)
List objects ........ Y (Y,N)

Restrictions .......*   Profile HNO,used               Model ..........*
Output options .....*   Profile HNO                    Association ....*
```

Parameters	
Retrieval type	All retrieval functions are available.
Output mode	The valid values depend on the retrieval type. Enter an asterisk to display the possible values.
Object ID	Asterisk notation is possible to specify a range of object IDs.
Search criteria	Search criteria can be used to restrict the function further. These additional selection criteria are type-dependent.
Drop existing objects	Y: Objects that are already contained in the extract are deleted. N: New objects are added to the objects already contained in the extract.
List objects	Objects are displayed or suppressed. The default value is taken from the profile parameter Maintenance options > List action.
Restrictions	You can use Restrictions to further limit the scope of objects for selection. See *Restrictions* in the section *Retrieval* in the *Predict Reference* documentation.
Output options	With output options you can determine the amount of information displayed. See *Output Options* in the section *Retrieval* in the *Predict Reference* documentation. The valid output options depend on Object type, Retrieval type and Output mode.

Build / Extend an Extract for all Object Types

```
13:13:58             *****  P R E D I C T  *****              2007-05-31
Plan   0                - Build/extend an Extract -                Profile HNO

Extract ID ...... HNO-ET                              Added 2007-05-31 at 09:54
                                                   Modified 2007-05-31 at 10:02

Build Extract for object type ..*     ( All )

Retrieval type .....*
Output-mode ........*

Search criteria
  Object ID .........



Drop existing objects N (Y,N)
List objects ........ Y (Y,N)

Restrictions .......*   Profile HNO,used
Output options .....*   Profile HNO
```

Parameters not listed here are described above. See *Build/Extend an Extract for a Specific Object Type*.

Parameters	
Retrieval type	With object type All, the following retrieval types are available: D — Objects C — Dummy/Placeholder Objects.
Output mode	L — All objects that meet the selection criterion Object ID and the restrictions are put in the extract. S — All objects that meet the selection criterion Object ID and the restrictions are displayed for selection. Enter /, X or S in the Cmd column to add objects to the extract. If you enter a command other than /, X or S in this column, the command is added to the workplan.
Object ID	Asterisk notation is possible. Object ID and Restrictions are the only additional selection criteria for object type All.
Output options	Only output options valid for all object types are displayed. As you can only use output modes List and Select for this function, only the following output options are applicable: ■ No. Abstract lines ■ Mark implementation ■ Cover page

Purge Extract - Code P

This function deletes extracts and all links to other objects.

13 Extract Retrieval

- Extract-specific Retrieval Functions 104
- Output Options for Extract Retrieval 104
- Output Options for Extract Retrieval - Continued 105

Extract-specific Retrieval Functions

Standard retrieval functions are described in the section *Retrieval* in the *Predict Reference* documentation.

Extracts Related to no Object - Code Y

Lists extracts which contain no objects.

Command: `UNUSED EXTRACT`

Valid output modes: Select, List, Display.

Extracts related to Objects - Code X

Lists all objects contained in the current extract or - with asterisk notation - contained in a range of extracts.

Command: `XREF EXTRACT`

Valid output mode: Cross reference.

Layout of Extract Lists

Meaning of Columns	
No. of Ref.	Number of objects contained in the extract.

Output Options for Extract Retrieval

Notes:

1. Unless output mode is S, the option Cover page is always valid.
2. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Association attributes			Y	Y	Y	Y			Y	Y	Y	Y				
Attributes																
Connecting character				Y						Y						
Description	Y		Y	Y			Y		Y	Y			Y			
Display modifier	Y		Y				Y		Y				Y			
Dummy/Placeholder										Y		Y		Y		Y
Extract	Y		Y	Y			Y		Y	Y			Y	Y		
Keywords	Y		Y	Y			Y		Y	Y			Y			
Mark implementation				Y						Y						
No. abstract lines	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
Owner	Y		Y	Y			Y		Y	Y			Y			
With users	Y		Y	Y			Y		Y	Y			Y			
Use Con-form	Y		Y	Y			Y		Y	Y			Y			
User exit	Y		Y				Y		Y				Y			

Output Options for Extract Retrieval - Continued

Retrieval Type	U		E				C				Y		X	
Output Mode	D	L	T		X		L		D		D	L	X	
Current/Related	c	c	c	r	c	r	c	r	c	r	c	c	c	r
Association attributes			Y	Y										
Attributes				Y		Y								Y
Connecting character				Y		Y								Y
Description	Y					Y				Y	Y			Y
Display modifier	Y										Y			
Dummy/Placeholder				Y		Y	Y		Y					
Extract	Y			Y		Y			Y	Y	Y			Y
Keywords	Y			Y		Y				Y	Y			Y
Mark implementation				Y		Y		Y		Y				Y
No. abstract lines	Y	Y		Y		Y		Y		Y	Y	Y		Y
Owner	Y			Y		Y				Y	Y			Y

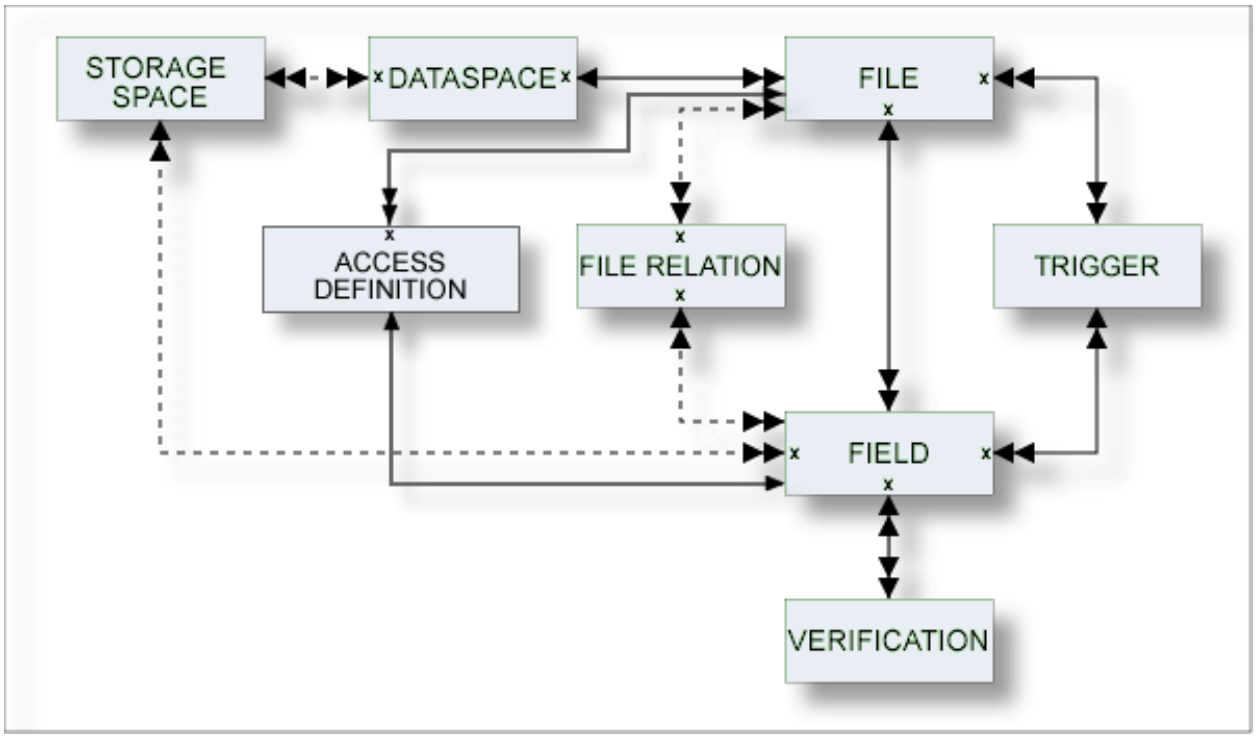
Retrieval Type	U		E				C				Y		X	
Output Mode	D	L	T		X		L		D		D	L	X	
Current/Related	c	c	c	r	c	r	c	r	c	r	c	c	c	r
With users	Y									Y	Y			Y
Use Con-form	Y					Y				Y	Y			Y
User exit	Y										Y			

V Field

With Predict, data definitions can be documented for a wide variety of data storage systems and for use with different programming languages. Field definitions are documented with objects of type Field.

In the predefined Predict metastructure, a field can have passive and active associations of the following types:

Valid passive association:	"Belongs to FI" (default passive association, mandatory)
Valid active associations:	"Is verified by VE" (default active association, up to 50 verifications may be linked) "Triggered by TR"



The description of object type Field is organized under the following headings:

Maintaining Objects of Type Field
Defining Basic Attributes of Fields
Defining Derived Fields
Defining Additional Attributes of Fields
Field Maintenance
Field Retrieval

14 Maintaining Objects of Type Field

- Field Maintenance Menu 110

Field Maintenance Menu

The Field Maintenance menu is called with function code M and object code EL in a Predict main menu or with the command `MAINTAIN ELEMENT`.

The functions Add a Field and Modify Field can also be called with the editor line command .E when maintaining the field list of a file object.

```
17:24:26              *****  P R E D I C T  *****              2007-05-31
Plan   0                  - (EL) Field Maintenance -              Profile SYSTEM

 Function                                Function

 A  Add a field                          R  Redefine field
 C  Copy field                           B  Browse through fields of a file
 M  Modify field                         H  Move field within a file
 N  Rename field                         L  Link children
 P  Purge field                          S  Select field from a list
 D  Display field                        J  Physical Attributes

Function .........
Field ID .........                                  Attributes .........*
Belongs to FI ....                                  Files of type ......*
Copy field ID ....
Copy file ID .....

Restrictions ....*   Profile Default ,used          Association ........*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main   ↩
```

Parameters	
Function	Executes one of the maintenance functions. The following functions are described in this section: ■ **Add a Field** ■ **Copy Field** ■ **Move Field within a File** ■ **Purge Field** ■ **Redefine Field** ■ **Browse through Fields of a File** ■ **Edit Field Expression**

Parameters	
	All other functions are described in the section *Maintenance* in the *Predict Reference* documentation.
Field ID	See ***Naming Conventions***. For the Select function: specifies a field ID which is to be used as a selection criterion. The field ID can be used alone or in combination with the file ID. If this field is left blank, all fields in the specified file(s) are listed. Asterisk notation is possible.
Belongs to FI	For the Add/Copy/Modify function: file containing the field. For the Select Field function: File ID is used as a selection criterion, either alone or in combination with the field ID. Asterisk notation is possible. If this field is left blank, all files are included in the search.
Copy Field ID	Specifies the ID of a field that is added or the position of a field that is copied or moved. See ***Copy Field***. For functions Add a Field and Move Field within a File: the position of the newly added or moved field. See ***Add a Field*** and ***Move Field within a File*** respectively.
Files of type	For the Select Field function: the scope of the function is restricted to fields in files of the specified type.
Copy File ID	Used for function Copy Field to identify the file to which a field is copied. See ***Copy Field***.

15 Defining Basic Attributes of Fields

- Add/Copy/Modify Screen 114
- Field Type 116
- Level Number 117
- Field Format 118
- Character Set 119
- Character Set - Adabas 120
- Field Length 120
- Descriptor Type 129
- Descriptor Type - continued 130
- Maximum Number of Values / Occurrences 131
- Unique Option 131
- Field Short Name 132
- Suppression / Null Value Option 132
- Variable Length Option - IMS 134
- Null Default Option 134
- Natural Field Length 134
- Do Not Convert Option 135
- Related Standard File 135
- Check against standard 135
- Natural Attributes 135
- Hidden 136
- Additional attributes / Associations 136

The functions Add Field and Modify Field can also be called from within the function Edit elements of a File with the editor line command .E. See the section *Editors in Predict* in the *Predict Reference* documentation.

Most attributes are applicable to fields of all file types.

Basic attributes applying to different field types are described below. Type-specific attributes are described in the section ***Defining Derived Fields***.

Add/Copy/Modify Screen

The following screen is displayed for the functions Add/Copy/Modify Field.

```
13:29:09             *****  P R E D I C T  *****                    2009-07-31
                              - Add a Field -
Field ID ........ HNO-EL-A
File ID ......... HNO-FI-A
Keys ..                                                                Zoom: N

Ty L Field ID                          F Cs Length   Occ   D U DB S NAT-l Cnv
*- - ------------------------------- *- * -------- ----- * * -- * ----- ---
   1 HNO-EL-A                                                  AA N




Natural attributes
  Header1 ..................
  Header2 ..................
  Header3 ..................
  Index on PE group level ..           Dynamic length .....
  Edit mask ........
Abstract     Zoom: N


  Additional attributes ..* N          Associations ..* N                    ↩
```

The screen for maintaining fields of SQL file types contains some different attributes and is shown below.

Possible Types of Fields

The table below lists all possible types of the object type field.

A	Adabas (with parameter Adabas SQL usage set to Y)
AT	Adabas Cluster Table
B	Adabas SQL view
BT, BV	Adabas D table/view
D, E, IT, IV, MT	DB2 table/view, Intermediate table/view, DB2 query table
OT, OV	Oracle table/view
JT, JV	Ingres table/view
X	General SQL file
XT, XV	Informix table/view
YT, YV	Sybase table/view

Add/Copy/Modify Screen for SQL Fields

The following screen is displayed for the functions Add/Copy/Modify Field and applies to SQL file types.

```
17:11:27             *****  P R E D I C T  *****                    2009-07-31
                              - Add a Field -
Field ID ........ XYZ_EL_S
File ID ......... XYZ-DB2
Keys ..                                                                Zoom: N

Ty L Field ID                          F Cs Length   Occ   D U N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * * *- -----
   1 XYZ_EL_S                                                U




Natural attributes
  Header1 ....
  Header2 ....
  Header3 ....
  Edit mask ..
Hidden .......   (Y/N)
Abstract     Zoom: N


  Additional attributes ..* N          Associations ..* N                    ↩
```

Field Type

The field type is indicated in the column Ty of the **Add/Copy/Modify screen**. The following types can be specified:

CM	Counter Field for multiple value field of type MU/MC
CP	Counter Field for periodic group of type PE/PC
DV	Derived field (SQL File types) see note below
GR	Group
HM	Hyperdescriptor as a multiple value field
HP	Hyperdescriptor as a field of a periodic group
HQ	Hyperdescriptor as a multiple value field of a periodic group
HY	Hyperdescriptor
MC	Multiple value field with automatic counter
MD	Indicator for union view. Can be set in the subquery editor only. MD indicates from which fields of the selected master file(s) the union, except or intersect view is created. See *Editing the Subquery of an SQL View*.
MU	Multiple value field
OD	Collation descriptor
PC	Periodic group with automatic counter
PE	Periodic group
PH	Phonetic descriptor
QN	SEQNO field
SB	Subfield/descriptor
SP	Superfield/descriptor
**,/*	Comment line, see *Edit List of Fields - Code L*
blank	None of the above. Normal field

Note: Derived field is also used in Predict as a generic term for hyperdescriptors, phonetic descriptors and sub/superfields and descriptors.

If HM, HP, HQ, HY, OD, PH, SB or SP is specified, an additional input screen is displayed. See *Defining Derived Fields*.

See also section *ADACMP (COMPRESS-DECOMPRESS)* in the *Adabas Utilities* documentation.

Redefining Fields

See *Redefine Field - Code R*.

Defining Periodic Groups in Periodic Groups

- Within a redefinition, nested periodic groups (PE within a PE) can be defined in files of all types.
- Outside of a redefinition, nested periodic groups can only be defined in files of the following types:

S	Sequential file
C	Conceptual file
M	ISAM file
Z	Standard file
O	Other file

Level Number

The level number of the field is indicated in the column L of the **Add/Copy/Modify screen**. The level number is used to define a group structure. Level numbers 1 to 9 can be used (except for Adabas files, see below).

- The level number must be increment by 1 immediately following a field of type RE, PE, PC or GR.
- For redefinitions, the level number must be at least one greater than the level number of the field being redefined. See *Redefine Field - Code R*.

Adabas Files

The following rules apply to level numbers for Adabas files:

- The PE/PC-groups, sub/superfields/descriptors, hyperdescriptors and phonetic descriptors must be at level 1.
- Level numbers of fields outside a redefinition must be in the range 1 - 7. See the section *ADACMP (COMPRESS-DECOMPRESS)* of the *Adabas Utilities* documentation for a complete description of Adabas levels.

Field Format

The format of the field is indicated in the column F of the **Add/Copy/Modify screen**. One of the following values can be specified (depending on the file type):

A	Alphanumeric	IV	Interval
AL	Long varchar	L	Logical
AV	Varchar	LO	Large object
B	Binary/char for bit data	LX	Bfile
BL	Long varchar bit data	MO	Money
BT	Bit	MS	Smallmoney
BV	Varchar for bit data	N/U	Numeric unpacked
D	Date	NS/US	Numeric unpacked with sign
DS	Smalldatetime	OK	Object key
DT	Datetime	P	Packed numeric
F	Floating point	PS	Packed numeric with sign
FD	Decimal floating point	S	Serial
G	Graphic	T	Time
GL	Long vargraphic	TK	Table key
GV	Vargraphic	TS	Timestamp
I	Integer	blank	Undefined

See tables in the section ***Field Length*** for valid combinations of format and length.

The following rules apply:

- Any format/length combination is allowed for the file types C (conceptual) or Z (standard).
- For groups, this attribute must be blank.
- For sub/superfields/descriptors in Adabas files, the appropriate format is provided by Predict based on the formats of the fields used. See ***Rules Applying to Format Changes***.
- The following formats are valid for fields within a redefinition: A, B, D, F, I, L, N/U, NS/US, P, PS, T.

Character Set

The parameter Character set determines the format in which data is stored. It is indicated in column Cs of the **Add/Copy/Modify screen**. The possible values depend on the file type and format.

File Type	Format		Character Set						
			ASCII	EBCDIC	Bitdata	Single Byte	Double Byte	Mixed Data	XML
Adabas	A, AV, LO					Y		Y	
Adabas D	A, AL, AV		Y	Y	Y				
DB2	A, AL, AV				Y	Y		Y	
	LO				Y	Y	Y	Y	Y
Oracle	A, AL				Y				
	AV							Y	
	LO				Y			Y	
Informix	A, AV							Y	
	AL				Y				
Ingres	A,AV				Y				
	AL	*			Y				
Sybase	A, AV	*			Y	Y	Y		
	AL				Y				

Note: A value must be specified for field types and formats marked with an asterisk (*).

Character Set	
ASCII	Data is stored in ASCII format.
EBCDIC	Data is stored in EBCDIC format.
Bitdata	Data is stored in binary form, no conversion is performed.
Single Byte	Data is stored in single-byte format. Double-byte characters are not possible.
Double Byte	Data is stored in double-byte format. String comparisons function differently to single-byte data.
Mixed Data	Data is stored in single and double-bytes. Data is subject to DB2 rules for multiple-byte character sets.
XML	Data is stored in XML format.

Character Set - Adabas

The following character set is used to define wide character fields.

Adabas		Predict	
Format	Option	Format	Character Set
A		A	blank or single
W		A	Mixed
A	LA	AV	blank or single
W	LA	AV	Mixed
A	LB, L4	LO	blank or single
W	LB, L4	LO	Mixed

Note: Option L4 applies to Adabas on Open Systems Version 5 or above only.

Field Length

The field length is indicated in column Length of the **Add/Copy/Modify screen**. This length is independent of its internal representation. When generating external objects, the field length is adjusted according to the internal representation of data used by the data storage system. For example: a field which is documented with length P9 is implemented with length P5 by the Adabas LOADER utility and the Adabas nucleus.

The following additional rules apply:

Table of Field Formats and Lengths

The table on the following pages contains the valid format/length combinations for fields of the following file types:

Column	File Type
A / U	Adabas file / userview
A(SQL) / AT / B	Adabas file with SQL usage, Adabas cluster table, Adabas SQL view
BT / BV	Adabas D table / view
D / E / IT / IV / MT	DB2 table / view / Intermediate table / view / DB2 query table
F	rdb file
I / J / K	IMS segment / segment layout / userview
JT / JV	Ingres table / view

Column	File Type
L / R / V / W	Logical VSAM file / view / Physical VSAM file / view
M	ISAM file
O	Other
OT / OV	Oracle table / view
P / Q	Entire System Server file / userview
S	Sequential file
T	RMS file
X	General SQL file
XT / XV	Informix table / view
YT / YV	Sybase table / view
1	LEASY
2	ISAM BS2000

Note: The tables do not contain the file types C (conceptual) and Z (standard). For these file types, any format/length combinations are allowed.

Key for the following table

no length	Format is valid; length must not be specified.
no restr.	No restrictions: any length may be specified.
p.q (m/n)	p — number of places before the decimal point q — number of places after the decimal point Where: 0 <= p <= m 0 <= q <= n 1 <= p+q <= m
n.m - n2.m2	Range of places before and after the decimal point. For example, fields of format MO for Sybase tables and views can have up to 15 places before the decimal point and up to 4 places after the decimal point (1.0 - 15.04).
*1	0 means 2GB
*2	0 means 4GB

Field Format	A, U	A(SQL) AT, B	BT, BV	D, E, IT, IV, MT	F	I, J, K	JT, JV	L, R, V, W	M
A	1-253	1-253	1-4000	1-254	1-253	1-253	1-2000	1-253	no restr.
AL			0-99999 *1	1-99999			0-99999 *1		
AV	1-16381	1-32767	1-4000	1-32767			1-2000		
B	1-126	1-126		1-255	1-126	1-126	1-2000	1-126	no restr.
BL							0-99999 *1		
BT									
BV				1- 32704			1-2000		
D	no length	no length	no length	no length	no length	no length		no length	no length
DS									
DT							no length		
F	4 / 8	4 / 8	4 / 8	4 / 8		4 / 8	4 / 8	4 / 8	4 / 8
FD				9 / 17					
G				1-127					
GL				1-16383					
GV				1-16383					
I	1 / 2 / 4 / 8	1 / 2 / 4 / 8	2 / 4	2 / 4 / 8	1 / 2 / 4 / 8	1 / 2 / 4 / 8	1 / 2 / 4	1 / 2 / 4 / 8	1 / 2 / 4 / 8
ID									
IV									
IY									
L	no length		no length			no length		no length	no length
LO	up to 2 GB			up to 2 GB					
LX									
MO							no length		
MS									
N	p.q (29/29)	p.q (29/29)			p.q (29/29)	p.q (29/29)		p.q (29/29)	p.q (29/29)
NS	p.q (29/29)	p.q (29/29)	p.q (18/18)	p.q (31/31)	p.q (29/29)	p.q (29/29)		p.q (29/29)	p.q (29/29)
OK							no length		
P	p.q (29/29)	p.q (29/29)			p.q (29/29)	p.q (29/29)		p.q (29/29)	p.q (29/29)

Field Format	A, U	A(SQL) AT, B	BT, BV	D, E, IT, IV, MT	F	I, J, K	JT, JV	L, R, V, W	M
PS	p.q (29/29)	p.q (29/29)	p.q (18/18)	p.q (31/31)	p.q (29/29)	p.q (29/29)		p.q (29/29)	p.q (29/29)
S									
T	no length	no length	no length	no length	no length	no length		no length	no length
TK							no length		
TS			no length	no length					
U	p.q (29/29)	p.q (29/29)			p.q (29/29)	p.q (29/29)		p.q (29/29)	p.q (29/29)
US	p.q (29/29)	p.q (29/29)	p.q (18/18)	p.q (31/31)	p.q (29/29)	p.q (29/29)		p.q (29/29)	p.q (29/29)

Field Format	O	OT, OV	P, Q	S	T	X	XT, XV	YT, YV
A	no restr.	1-2000	no restr.	no restr.	1-253	1-253	1-32762	1-255
AL		0-99999 *1					0-99999 *1	0-99999 *1
AV		1-2000					1-32762	1-255
B								
BL	no restr.		1-126	no restr.	1-126			1
BT								
BV								no length
D	no length		no length	no length	no length		no length	
DS								no length
DT		no length					no length	no length
F	4 / 8	4 / 8	4 / 8	4 / 8		4 / 8	4 / 8	4 / 8
FD								
G								
GL								
GV								
I	1 / 2 / 4 / 8	2 / 4	1 / 2 / 4 / 8	1 / 2 / 4 / 8	1 / 2 / 4 / 8	1 / 2 / 4 / 8	2 / 4	1 / 2 / 4
ID								
IV							7 / 17	
IY								
L	no length		no length	no length				
LO		0-99999 *2						
LX		0-99999 *2						
MO							p.q (32767/99)	1.0 - 15.04

Field Format	O	OT, OV	P, Q	S	T	X	XT, XV	YT, YV
MS								1.0 - 6.04
N	p.q (29/29)		p.q (29/29)	p.q (29/29)	p.q (29/29)	p.q (32/32)		p.q (38/38)
NS	p.q (29/29)	p.q (29/29)	p.q (29/29)	p.q (29/29)	p.q (29/29)		p.q (32/32)	
OK								
P	p.q (29/29)		p.q (29/29)	p.q (29/29)	p.q (29/29)	p.q (32/32)		
PS	p.q (29/29)	p.q (29/29)	p.q (29/29)	p.q (29/29)	p.q (29/29)		p.q (32/32)	
S							no length	
T	no length		no length	no length	no length			
TK								
TS		no length						no length
U	p.q (29/29)		p.q (29/29)	p.q (29/29)	p.q (29/29)	p.q (32/32)		
US	p.q (29/29)	p.q (29/29)	p.q (29/29)	p.q (29/29)	p.q (29/29)		p.q (32/32)	

Field Format	1	2
A	1-253	1-253
AL		
AV		
B		
BL	1-126	1-126
BT		
BV		
D	no length	no length
DS		
DT		
F	4 / 8	4 / 8
FD		
G		
GL		
GV		
I	1 / 2 / 4 / 8	1 / 2 / 4 / 8
ID		
IV		
IY		
L	no length	no length

Field Format	1	2
LO		
LX		
MO		
MS		
N	p.q (29/29)	p.q (29/29)
NS	p.q (29/29)	p.q (29/29)
OK		
P	p.q (29/29)	p.q (29/29)
PS	p.q (29/29)	p.q (29/29)
S		
T	no length	no length
TK		
TS		
U	p.q (29/29)	p.q (29/29)
US	p.q (29/29)	p.q (29/29)

Note: For format LO the length can be specified in units of KB, MB or GB. Enter the desired value followed by the corresponding character (K for KB, M for MB and G for GB). For limitations refer to the above tables.

SQL: DBMS Format and Corresponding Predict Format

The table below indicates the DBMS format and the corresponding Predict format for fields in files of the following types:

BT, BV	Adabas D table/view
JT, JV	Ingres table/view
OT, OV	Oracle table/view
XT, XV	Informix table/view
YT, YV	Sybase table/view

Key for the following table

n	length	
p,q	p	total number of places
	q	number of places after the decimal point

File Type	DBMS Format	Predict Format	Character Set
BT, BV	BOOLEAN	L	
	CHAR(n)	A(n)	
	CHAR(n) ASCII	A(n)	ASCII
	CHAR(n) BYTE	A(n)	Bitdata
	CHAR(n) EBCDIC	A(n)	EBCDIC
	DATE	D	
	FIXED(p,q)	NU, US, or PS	
	FLOAT(15)	F4	
	FLOAT(18)	F8	
	INTEGER	I4	
	LONG	AL	
	LONG ASCII	AL	ASCII
	LONG BYTE	AL	Bitdata
	LONG EBCDIC	AL	EBCDIC
	SMALLINT	I2	
	TIME	T	
	TIMESTAMP	TS	
	VARCHAR(n)	AV(n)	
	VARCHAR(n) ASCII	AV(n)	ASCII
	VARCHAR(n) BYTE	AV(n)	Bitdata
	VARCHAR(n) EBCDIC	AV(n)	EBCDIC
JT, JV	BYTE(n)	B	
	BYTE VARYING	BV	
	C(n)	A(n)	
	CHAR(n)	A(n)	Bitdata
	DATE	DT	
	DECIMAL (p,q)	PS	
	DECIMAL (p,q)	NS	
	DOUBLE PRECISION	F8	

File Type	DBMS Format	Predict Format	Character Set
	LONG BYTE	BL	
	LONG VARCHAR	AL	Bitdata
	INTEGER	I4	
	INTEGER1	I1	
	MONEY	MO	
	OBJECT_KEY	OK	
	REAL	F4	
	SMALLINT	I2	
	TABLE_KEY	TK	
	TEXT(n)	AV(n)	
	VARCHAR(n)	AV(n)	Bitdata
OT, OV	BFILE	LX	
	BLOB	LO	Bitdata
	CHAR(n)	A(n)	
	CLOB	LO	
	DATE	DT	
	DECIMAL(p,q)	NS	
	DECIMAL(p,q)	PS	
	DOUBLE PRECISION	F8	
	INTEGER	I4	
	INTERVAL DAY	ID	
	INTERVAL YEAR	IY	
	LONG	AL	
	LONG RAW	AL	Bitdata
	NCLOB	LO	Mixed data
	NVARCHAR2(n)	AV(n)	Mixed data
	RAW(n)	A(n)	Bitdata
	REAL	F4	
	ROWID	A and type QN	
	SMALLINT	I2	
	TIMESTAMP	TS	
	VARCHAR2(n)	AV(n)	
XT, XV	BYTE	AL	Bitdata
	CHAR(n)	A(n)	
	DATE	D	

File Type	DBMS Format	Predict Format	Character Set
	DATETIME YEAR TO FRACTION(5)	DT	
	DECIMAL(p,q)	NS	
	DECIMAL(p,q)	PS	
	FLOAT	F8	
	INTEGER	I4	
	INTERVAL DAY TO FRACTION(5)	IV	
	MONEY	MO	
	NCHAR(n)	A(n)	Mixed data
	NVARCHAR(n)	AV(n)	Mixed data
	REAL	F4	
	SERIAL	S	
	SMALLINT	I2	
	TEXT	AL	
	VARCHAR(n)	AV(n)	
YT, YV	BINARY(N)	A(n)	Bitdata
	BIT	BT	
	CHAR(N)	A(n)	Single byte
	DATETIME	DT	
	FLOAT	F8	
	IMAGE	AL	Bitdata
	INT	I4	
	MONEY	MO	
	NCHAR(N)	A(n)	Double byte
	NUMERIC, DECIMAL (p,q)	NS	
	NUMERIC, DECIMAL (p,q)	PS	
	NVARCHAR(N)	AV(n)	Double byte
	REAL	F4	
	SMALLDATETIME	DS	
	SMALLINT	I2	
	SMALLMONEY	MS	
	TEXT	AL	
	TIMESTAMP	TS	
	TINYINT	I1 or B1	
	VARBINARY(N)	AV(n)	Bitdata
	VARCHAR(N)	AV(n)	Single byte

Descriptor Type

The descriptor type is indicated in column D of the **Add/Copy/Modify screen**. The possible values are given in this and the following table.

Additional Information on Descriptor Fields

Descriptor fields can be of various types and control which fields can be used in search criteria (for example in Natural `FIND`, `READ` or `HISTOGRAM` statements). Natural programs can only read Adabas files using fields that are defined as a descriptor (usually type D). It should be part of the design to decide which fields are going to be descriptors, superdescriptors etc.

Adabas however, has a feature where a file can be read using any field - regardless of whether it is a descriptor or not. This is called a non-descriptor search. For large data volumes, this can be very slow. This is because there are no indexes built for non-descriptor fields and a physical scan of the data is required. Despite, if there are only a few records, and it is understood that a physical file scan is taking place, then it can be useful on some occasions.

By putting N into the descriptor type in a DDM, Natural programs are allowed to use this field as a non-descriptor search field. That way you can control in the DDM which fields are allowed to be searched on.

The various descriptor types are documented in more detail in *Columns of Field Attributes* in the section *Using the DDM Editor* of the *Natural Editors* documentation.

Code	Description	File Type										
		A, U	AT, B, A(SQL)	M	O	F	S	T	IV, D, E, MT	C	I, J, K	P, Q
D	Descriptor/Index	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y
	Disallow											
A	Alternate index									Y	Y	
N	Not inverted	Y	Y	Y	Y	Y	Y	Y	Y	Y		
	Search field										Y	
P	Primary Index		Y						Y	Y		
Q	Sequence									Y	Y	
E	Foreign key		Y						Y	Y		
F	Foreign index		Y						Y	Y		
	Force											
K	Common Key											
blank	No descriptor	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
	None											

Descriptor Type - continued

Code	Description	File Type									
		Z	1	2	L, R, V, W	X	BT, BV	OT, OV	JT, JV	YT, YV	XT, XV
D	Descriptor/Index							Y			
	Disallow	Y									
A	Alternate index		Y	Y	Y						
N	Not inverted					Y	Y	Y	Y	Y	Y
	Search field										
P	Primary Index		Y	Y	Y	Y	Y	Y	Y	Y	Y
Q	Sequence										
E	Foreign key					Y	Y	Y	Y	Y	Y
F	Foreign index							Y			
	Force	Y									
K	Common Key									Y	
blank	No descriptor		Y	Y	Y	Y	Y	Y	Y	Y	Y
	None	Y									

The following rules apply:

- In an Adabas file, the descriptor must be D if type HM, HP, HQ, HY (all hyperdescriptors), OD (collation descriptor) or PH (phonetic descriptor) is specified.
- For a subdescriptor in an Adabas file, descriptor D and type SB (subfield) must be specified.
- For a superdescriptor in an Adabas file, descriptor D and type SP (superfield) must be specified.
- In a DB2 table, DB2 query table or Oracle table, if a key, partitioning key, cluster column or index (descriptor D, E, F or P) includes more than one field, the type SP (superfield) must be specified
- In a VSAM file or userview (type L, R, V or W), the descriptor must be either P or A if type SP (superfield) is specified.
- If A is specified for a field of a VSAM file (type L or V), an additional screen is displayed for entering the required definitions (see below).
- Descriptor type must be blank for fields within a redefinition.

Maximum Number of Values / Occurrences

Maximum number of values for a multiple value field or occurrences of a periodic group is indicated in the Occ column of the **Add/Copy/Modify screen**. This parameter must be specified for multiple value fields and for periodic groups in a redefinition.

Field	Occurrences in range
Within a redefinition	1- 99999
Outside a redefinition	1 - 65535

When generating Copy Code, the value specified is used as the default for generating the specifications of MU/MC or PE/PC fields in a format buffer and/or record buffer.

When generating ADACMP/ADAFDU definitions, the Occ parameter is evaluated. If Occ is specified, the number of occurrences of each input data record is constant.

If Occ is not specified, the number of occurrences is taken from a counter field preceding a MU/MC or PE/PC field.

See also the section *ADACMP (COMPRESS-DECOMPRESS)* in the *Adabas Utilities* documentation.

Note: For fields of type QN, the Occ column is used to identify either the table level or an individual occurrence of a multiple value field or periodic group.

Unique Option

The unique option is indicated in column U of the **Add/Copy/Modify screen**. For groups, this attribute must be blank; for other fields, one of the following values can be specified:

U	Unique.
X	Used for unique descriptors in PE to exclude the occurrence (index) number from the definition of uniqueness.
blank	Not unique.

Unique option must be blank for fields within a redefinition.

Field Short Name

For file types listed below, the field short name is indicated in the column DB of the **Add/Copy/Modify screen**. This two-character short name must be defined for the following file types:

A	Adabas file	L	Logical VSAM file
AT	Adabas cluster table	R	Logical VSAM view
I	IMS segment	U	Adabas userview
J	IMS segment layout	V	VSAM file (physical)
K	IMS userview	W	Physical VSAM view

A field short name must conform to the rules for coding Adabas field names. See the section *ADACMP (COMPRESS-DECOMPRESS)* in the *Adabas Utilities* documentation and *Field Names* in section *Adabas Design* of the *Adabas Concepts and Facilities* documentation.

Field short names for userviews of Adabas, IMS and VSAM files need not be unique.

For fields within a redefinition, parameter Field short name must be blank.

Field short names for SQL tables and views are maintained internally by Predict and cannot be modified by users.

Rotated fields of files of type A (with SQL usage), type AT and B have the same short name and are identified uniquely by an occurrence number (column Occ).

Suppression / Null Value Option

- For fields of Adabas files, the suppression option is indicated in column S of the **Add/Copy/Modify screen**.
- For fields of SQL files, the null value option is indicated in column N of the **Add/Copy/Modify screen**.

For groups and for fields within a redefinition, this attribute must be blank. For other fields, one of the following values can be specified:

F	Fixed length
N	Null value suppression
R	Not null
U	Null counted
blank	Normal suppression

Parameter	SQL File Types *	Other File Types
Null value suppression		N
Fixed Length		F
Null allowed	U	U
Not null	R	R
Normal suppression		blank

* See ***Add/Copy/Modify Screen for SQL Fields*** for a list of SQL file types.

See also section *ADACMP (COMPRESS-DECOMPRESS)* in the *Adabas Utilities* documentation.

Profile Parameter Automatic Null Value

With the profile parameter Automatic null value you can determine an automatic Suppression/Null Value option when fields are added in Predict. See *Customizing Predict with Profiles* in the section *Predict User Interface* in the *Introduction to Predict* documentation.

The value depends on the file type. See table below.

Parameter	All SQL File Types except X	File Type X	Other File Types
Unique option = Unique or Descriptor type = Primary or Field format = serial	R	R	N
Others	U	blank	N

Note: SQL file types include files of type A with parameter Adabas SQL usage set to Y. See list in the section ***Add/Copy/Modify Screen for SQL Fields***.

For DB2 fields with Unique option = unique, values R and U are possible.

Variable Length Option - IMS

The variable length option for IMS fields is indicated in column S of the **Add/Copy/Modify screen**. The following values are valid:

Y	Variable length
blank	Fixed length

Null Default Option

The NULL default option for fields of SQL tables/views is indicated in the column Df of the **Add/Copy/Modify screen**. Possible values:

N	No default
Y	With default
blank	none

For Ingres fields with format OK or TK, the following additional values are possible:

S	SYSTEM_MAINTAINED
T	not SYSTEM_MAINTAINED
U	with default SYSTEM_MAINTAINED
V	with default not SYSTEM_MAINTAINED
W	not default not SYSTEM_MAINTAINED

This parameter must be blank for fields within a redefinition.

Natural Field Length

The Natural field length is indicated in column NAT-l of the **Add/Copy/Modify screen**. The following rules apply:

- The parameter has to be specified if the field can be:
  - alphanumeric and greater than 253
  - graphic and greater than 126
  - numeric p.q (m/n) where p+q > 29 or q > 7.

See table of valid formats and lengths in the section ***Field Length***.

- The value specified here is the length that Natural can use for the field as defined in the DDM.

Do Not Convert Option

The "do not convert" option is allowed for A and AV format fields of the following file types:

- Adabas file/userview
- Conceptual file

Related Standard File

This parameter is described in the section ***Rippling***.

Check against standard

This parameter is described in the section ***Rippling***.

Natural Attributes

Headers

The Natural headers 1 - 3 are included in DDMs generated from the file containing the field.

Alphabetic characters in Natural headers are converted to upper-case if the Predict parameter Upper/lower case has been set to Y. See the section *Defaults* in the *Predict Administration* documentation.

Index on PE Group Level

If you enter Y in this field and execute the Natural Area Editor command .V for a DDM containing the field object, the maximum occurrences of periodic groups is generated on group level.

If this parameter is left blank (default), the maximum occurrences is generated for each element in the group.

Edit mask

The Natural edit mask. See the description of the DISPLAY statement in the *Natural Statements* documentation for further details.

Alphabetic characters in the Natural edit mask are converted to upper-case if the Predict parameter Upper/lower case has been set to Y. See the section *Defaults* in the *Predict Administration* documentation.

Dynamic length

If you enter Y, fields with format LO or AV in files of type A/U are generated with dynamic length in the DDM.

Hidden

Specifies that the column is not visible in the result for SQL statements unless you explicitly refer to the column by name.

If you enter Y, columns are not visible.

Additional attributes / Associations

The following additional options in the Additional attributes / Associations line are available for fields. Standard options are described in the section ***Additional attributes / Associations***.

Verification

Enter Y in Associations field to call the Predict Link Editor to edit the verification list of the field. Up to 50 verifications can be linked to a field via "Is verified by VE".

This editor can also be invoked by:

- Selecting L (link children with association VE) in the Field Maintenance Menu.
- Entering command `LINK ELEMENT VERIFICATION` *`field-id file-id`*.

See the section *Editors in Predict* in the *Predict Reference* documentation.

Additional attributes

Enter Y in this field to define additional attributes. The attributes that can be defined depend on the field type. See ***Defining Additional Attributes of Fields***.

16 Defining Derived Fields

- General Rules for Defining Derived Fields 140
- Defining Derived Fields of Special Types 143
- Rules Applying to Format Changes 146
- Rules Applying to Suppression/Length Changes 149
- Validation of Derived Field Definitions 149

Note: Derived field is a generic term in Predict for fields and descriptors defined on the basis of one or more source fields. This term should not be confused with field type DV applicable to SQL views (see *Field Type*).

Defining derived fields and keeping the definitions consistent is a complex task. Predict offers a variety of functions to help with it.

General Rules for Defining Derived Fields

If a derived field is modified, a table containing the source fields appears when ENTER is pressed in the Modify Field screen. The size and format of this table varies with the type of the derived field.

The table in the screen below only applies to files of type D or MT with format SP (superfield). Editor functions and a selection mechanism help when defining derived fields. Other SQL file types lack the Random option as shown in the A/D/R column and have no Expression option either. Refer to **Key or Index Fields in SQL Files - Superfields** for details on these options.

```
13:24:29             *****  P R E D I C T  *****             2009-07-31
                                - Modify Field -
    Field ID ........ ARH_SP                          Modified 2007-05-31 at 13:14
    File ID ......... ARH-D1


    Ty L Field name                        F  Length   Occ   D U DB N NAT-l
    -- - -------------------------------- -- -------- ----- - - -- - -----
    SP 1 ARH_SP                                                D U AB

                Index Name .... ARH-ARH_SP
                Source field name *               A/D/R  Expression
              1   ARH1                              A    _
              2   ARH3                              D    _
              3   ARH2                              A    _
              4   ARH4                              A    _
              5   ARH5                              A    _
              6   ARH6                              A    _
              7   ARH7                              A    _
              8   ARH8                              A    _
              9   ARH9                              A    _
              10  ARH10                             A    _
     Additional attributes ..* N          Associations ..* N        Scroll to: __
```

General Attributes of Definitions of Derived Fields

The following attributes are contained in most definitions of derived fields. Attributes specific to certain types of derived fields are described in the respective sections.

Attributes	
Source field name	Name of the fields used by derived fields.
F, Length	Format and length of the source field. These columns are read-only. Section *Rules Applying to Format Changes* describes how the format of the derived field is determined by Predict.
Start	The relative byte position where the part of the source field to be used by the derived field starts (not applicable to phonetic descriptors). See also *Specifying the Start and End Position* below.
End	The relative byte position where the part of the source field to be used by the derived field ends (not applicable to phonetic descriptors and VSAM Primary Superdescriptors or Alternate Indices). See also *Specifying the Start and End Position* below.
DB	Field short name of the source field. This column is read-only.

Specifying the Start and End Position

The start and end values given in the definition of a derived field are always byte positions within the source fields (beginning with 1 and counting from left to right for alphanumeric fields and binary fields and from right to left for numeric fields).

The full length is used if no start and end values are specified. In Adabas it is possible to address byte positions outside of the length of field. If this feature is used and a start byte outside of the source field specified, an end byte must be specified.

Note: Special rules apply when specifying the length of subfields/descriptors. See *Specifying the Length of Subfields*.

The following rules apply:

- Superfields/descriptor definitions can be based on up to twenty source fields.
- Only formats A, B and N are possible for superfields/descriptors.
- Format N can be useful for Natural, but is not recommended because an alphanumeric or binary value cannot be converted to a numeric field.

Editor Functions

Deleting Source fields

Source fields can be deleted from the definition by overwriting their name with blanks. The remaining lines will be reorganized automatically.

Moving Source fields

Source fields can be moved with the .m command. Enter .m at the beginning of the line to be moved, position the cursor in the line where the moved line is to appear, and press ENTER. The table of source fields is automatically reorganized.

Scrolling

If a definition of a derived field contains more source fields than can be displayed in one screen, the source field to be displayed on top of the list can be specified in the field Scroll to. See *General Rules for Defining Derived Fields*.

Selection Mechanism

New source fields can be added to the definition of a field by selecting them from a list of all fields contained in the file. This list is displayed in a Source field window if a name with asterisk notation (*) is entered as selection criterion in the list of the current definitions (as shown in the screen below).

```
13:09:01             *****  P R E D I C T  *****              2007-05-31
                              - Modify Field -
Field ID ........ ARH_SP                +Top---------Source field------------+
File ID ......... ARH-D1                ! _ ARH1                              !
                                        ! _ ARH2                              !
                                        ! _ ARH3                              !
Ty L Field name                       F ! _ ARH4                              !
-- - ------------------------------- - ! _ ARH5                              !
SP 1 ARH_SP                             ! _ ARH6                              !
                                        ! _ ARH7                              !
           Index Name .... ARH-ARH_SP   ! _ ARH8                              !
           Source field name *          ! _ ARH9                              !
         1   ARH1                       ! _ ARH10                             !
         2   ARH3                       ! _ ARH11                             !
         3   ARH2                       ! _ ARH12                             !
         4   ARH4                       ! _ ARH13                             !
         5   ARH5                       ! _ ARH14                             !
         6   *RH6                       ! _ ARH15                             !
         7   ARH7                       ! _ ARH16                             !
         8   ARH8                       ! _ ARH17                             !
         9   ARH9                       ! _ ARH18                             !
         10  ARH10                      !Command ==> +______                  !
 Additional attributes ..* N            +More--------------------------------+
```

A source field is selected by marking it in the left column or by positioning the cursor in the respective line and pressing ENTER. One field can be selected at a time.

Defining Derived Fields of Special Types

The following topics are covered below:

- Superfields/Descriptors for Files of Type A, C and Z
- Subfields/Descriptors for Files of Type A, C and Z

Superfields/Descriptors for Files of Type A, C and Z

The screen for the definition of superfields/descriptors for files of type Adabas, Conceptual and Standard looks as follows.

```
19:13:06            *****  P R E D I C T  *****              2007-05-31
                              - Modify Field -
Field ID ........ SUPER-1                            Added 2007-05-31 at 13:39
File ID ......... PD-A2                            Modified 2007-05-31 at 14:53


Ty L Field name                        F  Cs Length   Occ   D U DB S NAT-l
-- - ------------------------------- *-  * -------- ----- * * -- * -----
SP 1 SUPER-1                           A     168.0              AJ N

           Source field name *                F  Length   Start  End  DB
         1   STD-EL1                          A  30.0      1     10   AA
         2   STD-EL2                          N  23.0      2     2    AB
         3   STD-EL2                          N  23.0      2     3    AB
         4   LOGICAL                          B  10.0                 AC
         5   LOGICAL                          B  10.0                 AC
         6   MAIG                             A  50.0                 BA
         7   TIME                             T                       BB
         8
         Start/End: Relative byte position in source field. Default
         .          is first and last byte of source field

  Additional attributes ..* N          Associations ..* N        Scroll to: __
```

See also *General Rules for Defining Derived Fields*.

Subfields/Descriptors for Files of Type A, C and Z

Subfields/Descriptors for files of type Adabas, Conceptual and Standard are defined in the screen below.

```
09:13:28            *****  P R E D I C T  *****              2007-05-31
                             - Add a Field -
Field ID ........ PHON-4                              Added 2007-05-31 at 09:11
File ID ......... HEB-FI                                 by HNO


Ty L Field name                        F  Cs Length   Occ   D U DB N NAT-l
-- - ------------------------------- --  - -------- ----- - - -- - -----
SB 1 PHON-4                                                   SS N

           Source field name *                F  Length   Start  End   DB
         1   HEB-TEST                         A  1.0                   AF




         Start/End: relative byte position in source field. Default
         is first and last byte of source field.

  Additional attributes ..* N          Associations ..* N        Scroll to: __
```

With subfields/descriptors, only one source field can be entered in the window.

Specifying the Length of Subfields

If the source field of a subfield/descriptor has format P and the start byte is greater than 1, the length of the subfield/descriptor is normal length+1, because the sign of the source field is always included in the subfield/descriptor field (see *Adabas Utilities* documentation).

Example:

Given that

- the source field has format P and length 5,
- the subfield/descriptor definition is source field from 2 to 3,
- the length of the subfield is 3 bytes (2 bytes + 1 byte for sign),
- the 3 bytes packed are 5 digits,

then the subfield/descriptor has format P and length 5.

Note: See also *General Rules for Defining Derived Fields*.

Rules Applying to Format Changes

The format of derived fields is determined by Predict or can be defined manually.

Note: To understand the following, some knowledge of the hierarchical data structures of Predict and the process of rippling is required. See ***Rippling*** in the section *File* for more information.

The following topics are covered below:

- Determining the Format of Sub/Superfields/Descriptors
- Changing the Format of Superfield/Descriptors Manually
- Impact of Changes to Standard Fields - Rippling
- How the Rippling of Changes to Source Fields is Indicated

Determining the Format of Sub/Superfields/Descriptors

The format of sub/superfields/descriptors in files of type Adabas, Conceptual and Standard (codes A, C, Z) is generated automatically by Predict. A format of a derived field that has been determined by Predict can, however, be overwritten manually. The following sections describe the rules applying.

Subfield/Descriptor

Subfield/descriptors always have the same format as the source fields they are derived from. If the format of a source field is changed, the format of the subfield/descriptor is changed accordingly.

Superfield/Descriptor without Format

If a superfield/descriptor is defined without a format, Predict assigns the format as follows:

- **Format=A**
  if at least one source field of the SP field is defined with format A, or if one of the source fields specified in the definition does not yet exist in the file.
- **Format=B**
  if no source field is defined with format A.

Superfield/Descriptor with Format

If the format of source fields has been changed, Predict checks if the new and the old format of the source field are compatible. If they are compatible, the change does not have any impact on the format of the superfield/descriptor.

The formats NS, US, N and U and the formats P and PS are compatible. So, if the format is changed from N to US, for example, the format of the superfield/descriptor will not change.

If the new and the old format of the source field are *not* compatible, a window appears in which a format change proposed by Predict can be confirmed or a new format can explicitly be assigned to the superfield/descriptor (see screen below).

```
13:51:16            *****  P R E D I C T  *****              2007-05-31
                                - Modify field -
Field ID ........ FELD5                              Added 2007-05-31 at 13:15
File ID ......... HNO-STA1                         Modified 2007-05-31 at 13:15

Ty L Field-name                        F  Cs Length   Occ   D U DB N NAT-l
-- - -------------------------------- --  - -------- ----- - - -- - -----
SP   FELD5                             B     25.0


              +-----------------------------------------------+
              !  Superdescriptor definition changed           !
              !  to the correct format.                       !
              !                                               !
              !   old format .. B    new format .. A          !
              !                                               !
              !  Hit 'ENTER' to continue or change format.    !
              +-----------------------------------------------+
```

Changing the Format of Superfield/Descriptors Manually

The format of a superfield/descriptor can be changed manually (with the Modify Field function). If a source field of the superfield/descriptor is then changed again, Predict checks if the change affects the format of the superfield/descriptor.

Impact of Changes to Standard Fields - Rippling

Changes to sub/superfield/descriptors and fields used in sub/superfield/descriptors (source fields) are rippled as described in the sections below.

Changes to Sub/Superfield/Descriptors

It is not recommended to define sub/superfield/descriptors in standard files and to use these in real files. It is however possible to do it. The following rule then applies:

Changes to the format and length and changes to the definition of derived fields in standard files are not rippled from standard files to real files and userviews. This is because the definition of derived fields is not coupled, and rippling format and length alone could lead to inconsistent data definitions in real files and userviews.

Changes to Source fields

Changes to the format of a standard field are rippled as normal to all fields in a file connected to this standard field.

If a field in an Adabas file is used in the definition of a sub/superfield/descriptor, the format of the sub/superfield/descriptor is also changed if one of the following conditions is met:

- the resulting format is A, or
- the resulting format is B and the old format was A.

Note: In the case of superdescriptors, if the format in the Adabas file is set (manually) to N and the correct format were B, no change is made (unless the field length is greater than 29).

How the Rippling of Changes to Source Fields is Indicated

If changes to standard fields are rippled to derived fields in real files and userviews, two screens are displayed indicating this process of rippling.

In the first screen the *changes of source fields* are indicated.

```
13:51:35            *****  P R E D I C T  *****              2007-05-31
                             - Modify file -                        Page:   1

                        list of field updates
                        ---------------------

FELD2                            *** upd ***
FELD2                            HNO-ADA1                        *** upd ***
FELD5                            HNO-ADA1                        *** upd ***
FELD5                            *** upd ***
```

In the second screen the *changes of the format and/or length* of derived fields are indicated.

```
SUB/SUPER/PHON/COLLATION- fields, -descriptors length are changed
-----------------------------------------------------------------

Ty Field name                        File name
-- -------------------------------- --------------------------------
SP FELD5                             HNO-STA1                        updated
```

Rules Applying to Suppression/Length Changes

The suppression and length of derived fields is determined by Predict.

The suppression and length of sub/superfields/descriptors in files of type Adabas, Conceptual and Standard (codes A, C, Z) is generated automatically by Predict. A derived field gets:

- null suppression if at least one source field has null suppression.
- the total length of the source field(s).

Note: The full source field length is used if no start and end values are specified.

If the suppression or length of a source field is changed, suppression and length of the sub/superfields/descriptors are changed accordingly.

Validation of Derived Field Definitions

If the format of derived fields is changed manually, Predict performs validation checks. These checks are described in this section.

Predict performs the following validations for derived fields:

- A superfield/descriptor can have only one source field which is a multiple-value field.
- Source fields with format D, T, or L must not have a start or end character.

The following rule applies for all file types except Conceptual and Standard: All source fields must exist in the file. This check is performed when a `CHECK` or `CAT` command is entered in the field list editor or when the Add/Copy/Modify field function is executed from the Field Maintenance menu.

These validations can be executed differently:

- If a source field is changed with the list editor (function Link children in the Modify File menu with Related type set to EL), the validation can be executed explicitly with the `CHECK` command. The `CAT` command will also perform the validation.
- If a source field is changed with the function Modify Field, the validations are performed directly.

The following topics are covered below:

- Phonetic Descriptors for Files of Type A, C and Z
- Hyperdescriptors for Files of Type A, C and Z
- Collation Descriptors for Files of Type A, C and Z
- Key or Index Fields in SQL Files - Superfields
- VSAM Primary Superindex or Alternate Superindex

Phonetic Descriptors for Files of Type A, C and Z

The screen for defining phonetic descriptors for files of type Adabas, Conceptual and Standard is identical to that for subfields/descriptors. See ***Subfields/Descriptors for Files of Type A, C and Z***.

With phonetic descriptors, only one source field can be entered in the window.

The Start and End attributes do not apply to phonetic descriptors: Adabas always uses the first 20 bytes of this field to build a phonetic descriptor.

Further information on sub/superfields/descriptors and phonetic descriptors can be found in the section *ADACMP (COMPRESS-DECOMPRESS)* in the *Adabas Utilities* documentation.

Note: See also ***General Rules for Defining Derived Fields***.

Hyperdescriptors for Files of Type A, C and Z

The screen for defining hyperdescriptors looks as follows:

```
13:00:05            *****  P R E D I C T  *****              2007-05-31
                             - Add a Field -
Field ID ........ FIELD3                              Added 2007-05-31 at 12:59
File ID ......... DEMO                                   by HNO


Ty L Field name                        F  Length   Occ   D U DB N NAT-l
-- - --------------------------------- -- -------- ----- - - -- - -----
HY 1 FIELD3                            A  20.0           D   XZ N

    User exit nr ...
    Source field name *                  Source field name *
  1                                     2
  3                                     4
  5                                     6
  7                                     8
  9                                     10
  11                                    12
  13                                    14
  15                                    16
  17                                    18
  19                                    20
  Additional attributes ..* N           Associations ..* N
```

Attributes	
User exit nr	A number between 1 and 31 identifying the user exit that defines the hyperdescriptor. See the *Adabas User Exits* documentation.

Collation Descriptors for Files of Type A, C and Z

The screen for defining collation descriptors looks as follows:

```
18:13:37            *****  P R E D I C T  *****                    2006-04-20
                             - Add a Field -
Field ID ........ HNO_COLLATION                       Added 2006-04-20 at 18:13
File ID ......... HNO-ADA                                by HNO


Ty L Field name                        F  Length   Occ   D U DB N NAT-l
-- - ------------------------------- -- -------- ----- - - -- - -----
OD 1 HNO_COLLATION                                       D   AB N

Source field ........*
User exit nr .........
HE option ............   (Y,N)
Locale string ........
Collation strength ..*    (none)
Case first ..........*    (none)
Alternate ...........*    (none)
Case level ..........*    (none)
French ..............*    (none)
Normalization .......*    (none)


X Additional attributes ..* N          Associations ..* N                    ↩
```

Attributes	
User exit nr	A number between 1 and 8 identifying the user exit that defines the collation descriptor. See the *Administration* section of the *Adabas on Open Systems* documentation for further information.
HE option	If you specify this option, you must specify the corresponding parent field value in the value buffer for search operations, rather than the internal collation key. See the *Administration* section of the *Adabas on Open Systems* documentation for further information.
Locale string	One of the locales supported by ICU. See the *Administration* section of the *Adabas on Open Systems* documentation for further information.
Collation strength	You can specify one of the following values: P (Primary), S (Secondary), T (Tertiary), Q (Quarternary), I (Identical) or blank (none). The value specified represents the comparison levels. See the *Administration* section of the *Adabas on Open Systems* documentation for further information.
Case first	Specifies whether uppercase letters will be sorted before lowercase letters or vice versa. Valid values: U (Upper), L (Lower) or blank (none). See the *Administration* section of the *Adabas on Open Systems* documentation for further information.

Attributes	
Alternate	Specifies the sorting sequence for punctuation characters such as space or hyphen. Valid values: S (Shifted), N (Non ignorable) or blank (none). See the *Administration* section of the *Adabas on Open Systems* documentation for further information.
Case level	If specified, an additional case level is formed between secondary and tertiary. Valid values: C (Caselevel), N (No caselevel) or blank (none). See the *Administration* section of the *Adabas on Open Systems* documentation for further information.
French	Specifies whether or not diacritics will be sorted as in French. Valid values: F (French), N (No french) or blank (none). See the *Administration* section of the *Adabas on Open Systems* documentation for further information.
Normalization	Specifies whether or not Unicode canonical equivalence is to be taken into account. Valid values: O (Normalization), N (No Normalization) or blank (none). See the *Administration* section of the *Adabas on Open Systems* documentation for further information.

Key or Index Fields in SQL Files - Superfields

The screen below is used for defining Keys or Indexes in fields of the following file types:

A	Adabas file (with parameter Adabas SQL usage set to Y)
BT	Adabas D table
D	DB2 table
MT	DB2 query table
JT	Ingres table
OT	Oracle table
X	General SQL
XT	Informix table
YT	Sybase table

The following rules apply:

- If the field type is blank (normal field), the key or index is based on one field.
- If the field type is SP (superfield), the key or index includes more than one field.
- If the field type is SP (superfield), the index includes one or more fields if the file is D or MT.

```
18:21:07                *****  P R E D I C T  *****                  2011-11-24
                              - Modify Field -
Field ID ........ POLICY                          Modified 2011-11-21 at 17:13
File ID ......... XYZ-POLICY                             by XYZ


Ty L Field name                        F  Length   Occ   D U DB N NAT-l
-- - ------------------------------- -- -------- ----- - - -- - -----
SP 1 POLICY                                            D U AJ
           Index Name .... XYZ-POLICY
           No Overlaps ...   (Y/N)
           Source field name *              A/D/R/I  Expression
         1   COVERAGE                          A     _
         2   BUS_END                           A     _
         3   BUS_START                         A     _
         4                                             _
         5                                             _
         6                                             _
         7                                             _
         8                                             _
         9                                             _
         10                                            _
* Additional attributes ..* S          Associations ..* N     Scroll to ... __ ↩
```

Attributes		
Index name	The name of the key or index. Must be entered in qualified form: creator/schema name followed by key or index name, separated by a hyphen. The creator/schema and key or index name are subject to SQL naming conventions. Creator name and field name are concatenated and proposed as index name.	
No Overlaps	Only applies when documenting DB2 temporal tables that record the application period (BUSINESS_TIME). Specifies whether (Y) or not (N) the `BUSINESS_TIME WITHOUT OVERLAPS` clause applies. If BUSINESS_TIME WITHOUT OVERLAPS is specified, the BUSINESS_TIME period will not overlap in time periods for the same column-name values.	
Source field name	The name of a column (source field) from which the key or index is derived. If the key or index is based on one field (field type blank), the name of that field is displayed and cannot be changed. If the key or index includes more than one field (Field type SP), up to 64 column names can be entered. Each must name a column of the table. **Note:** For fields in files of type X (General SQL), you can enter up to 16 column names. Enter a value in the Scroll field to define source fields greater than 10.	
A/D/R/I	A	Puts key or index entries in ascending order by source fields (column). Default.
	D	Puts key or index entries in descending order by source fields (column).
	R	Puts key or index entries in random order by source fields (column). Only applies to files of type D or MT.
	I	Specifies additional columns to append to the set of index key columns of a unique index. Only applies to files of type D or MT.

Attributes	
Expression	Only applies when an index and if files of type D or MT. Mark the column Expression to display an editor for the expression. Up to 64 expressions can be maintained. An asterisk indicates that an expression exists.

Expression

If you mark the Expression field, the following editor is shown.

An index is composed of source field name(s) and/or expression(s). If the index contains at least one expression, the source field name(s) are saved as an expression also. If an index consists only of expressions that are reduced to source field names, then these source field names will be shown in the column labeled Source field name when the field is edited again.

```
16:16:04             *****  P R E D I C T  *****                      2009-07-31
                              - Modify Field -
Field ID ........ EL_1_SP                            Added 2009-02-05 at 15:31
File ID ......... HEB-D                            Modified 2009-07-11 at 07:59


Ty L Field name                        F  Length   Occ   D U DB N NAT-l
-- - ------------------------------- -- -------- ----- - - -- - -----
SP 1 EL_1_SP                                            D   AI

**** ************************* Expression  1 ********************************
0001 SUBSTR (EL_INDEX,1,2)
0002
0003
0004
0005
0006
0007
0008
0009
0010
Command ===>
**** ***************************************************************************
```

Enter `Help` in the command line or press PF1 to display the editor help:

```
16:16:04             *****  P R E D I C T  *****                    2009-07-31
                              - Modify Field -
Field ID ........ EL_1_SP                            Added 2009-02-05 at 15:31
File ID ......... HEB-D                            Modified 2009-07-11 at 07:59


Ty L Field name                        F  Length   Occ   D U DB N NAT-l
-- - ------------------------------- -- -------- ----- - - -- - -----
SP 1 EL_1_SP                                            D   AI

+--------------------------------- Commands ----------------------------------+
! Top or T or --    Position to top                 PF1      Help             !
! Bottom or B or ++ Position to bottom              PF3      Exit             !
! +P or +           Position forward one page       PF5      Sel              !
! -P or -           Position backward one page      PF6      --               !
! +nn               Position forward 'nn' lines     PF7      -                !
! -nn               Position backward 'nn' lines    PF8      +                !
! nn                Position to line 'nn'           PF9      ++               !
!                                                                             !
! Save or Cat       Save Changes            Exit or E or .   Exit from editor !
! Check or C        Check Change            Sel              Select field     !
!                                                                             !
+-----------------------------------------------------------------------------+
```

VSAM Primary Superindex or Alternate Superindex

VSAM superdescriptors (Field type SP) in a file of type V (physical VSAM) and L (logical VSAM) are defined in the following screen.

```
13:45:57            *****  P R E D I C T  *****              2007-05-31
                             - Modify Field -
Field ID ........ KEY                                Added 2002-07-05 at 13:01
File ID ......... PD-V1                            Modified 2007-05-31 at 13:42


Ty L Field name                        F  Cs Length   Occ   D U DB N NAT-l
-- - ------------------------------- --  - -------- ----- - - -- - -----
SP 1 KEY                                     30.0           A   AE N

           Source field name *                F  Length   Start  End   DB
         1 ___________________________________ _ _______ _____ _____ __
                                                                 __
                                                                 __
                                                                 
```

Additional Attributes for VSAM Alternate Fields

```
13:44:37            *****  P R E D I C T  *****              2007-05-31
                            - Modify Field -
Field ID ........ KEY                                Added 2002-07-05 at 13:01
File ID ......... PD-V1                            Modified 2007-05-31 at 13:42
Keys ..                                                                Zoom: N

Ty L Field name                        F  Length   Occ   D U DB N NAT-l
*- - -------------------------------- *- -------- ----- * * -- * -----
   1 KEY                               A  30.0           A   AE N


                                         +- VSAM descriptor attributes -+
Natural attributes                       !                              !
  Header1 ....                           !  Upgrade flag ... Y (Y,N)    !
  Header2 ....                           !  Sort flag ...... N (Y,N)    !
  Header3 ....                           !  Null flag ...... N (Y,N)    !
  Edit mask ..                           !  DD name ........ DDNAME     !
Comments     Zoom: N                     !                              !
                                         +------------------------------+



  Additional attributes ..* N          Associations ..* N
```

Additional Descriptor Attributes		
Upgrade flag	Y	Alternate index is updated by Natural.
	N	Alternate index is updated by VSAM.
Sort flag	Y	If the upgrade flag is also Y, the alternate index is read in ascending order. Otherwise, the alternate index is read in the order in which the values were entered during field update.
Null flag	Y	Records with a null value in this index field are suppressed.
DD name	The DD name associated with this alternate index file. In CICS, the FCT name of the VSAM file.	

17 Defining Additional Attributes of Fields

- 3GL Specification 160
- Condition Name and Value 162
- Field Name Synonyms 163
- Old Mode Synonyms 164
- Adabas Security and Edit mask 165
- DBMS Extensions 166
- Field Procedure 172
- Derived Field Expression 173
- Index Definition - DB2 174
- Index Definition - Oracle 178
- Default value 184
- Constraint name 185
- Identity definition / Change log 186
- Platform Compatibilities 189
- Base Extensions 190

If Additional attributes is set to Y, a window is displayed containing additional attributes for selection.

Only those types of additional attributes appear in the window that apply to the type of field. For example: the option Adabas security & Edit mask is not contained in the list when a DB2 index field is processed.

More than one choice can be made at a time. The respective input maps are then displayed one after the other.

3GL Specification

```
13:06:25             *****  P R E D I C T  *****              2007-05-31
                                - Modify Field -
Field ID ........ HNO-EL1                              Added 2007-05-31 at 12:55
File ID ......... HNO-FI1                                  by HNO


Ty L Field ID                          F  Cs Length   Occ   D U DB N NAT-l
*- - ------------------------------- *-  - -------- ----- * * -- * -----
   1 HNO-EL1                           A     2.0                AA N

Specifications for 3GL
  Gr.structur ........   (n)
  Justify ............   (R)
  Synchronized .......   (S)
  Initialize with ...*
    Init value .......
  Indexed by .........
  Depending on .......
```

Attributes	
Gr.structure	The field attribute Gr.structure is used to change the record layout generated from a PE/PC field. If Gr.structure is set to N, all fields within a PE group are treated as multiple value fields. Setting Gr.structure to N prevents the format buffer for Adabas from becoming very large. Gr.structure = N can only be specified for real fields in the deepest PE group (highest level number). For example: if there are 3 PE groups in the file on level 1, 4 and 6, only the PE groups on level 6 can be marked with Gr.structure = N. If Gr.structure is set to blank, PE/PC groups are to be generated as groups which occur n times as a whole.

Attributes		
Justify	R	When COBOL copy code is generated, the statement JUSTIFIED RIGHT is added for this field. Any data written to this field is then right-justified.
	L	Data will be left-justified. Default.
Synchronized	Applicable to fields of format I, F or B and length 1, 2, 4 or 8.	
	S	When Assembler, COBOL or PL/I copy/include code or a record layout is generated, this field can be aligned on a half-word, word, or double-word boundary (speeding up arithmetic operations). This affects format buffer generation and the offsets of the fields in the record buffer. Slack-bytes are inserted into the record buffer by the assembler or compiler but they are built into any format buffer by Predict using space characters X.
Initialize with	Determines the initial value for generation. To be used instead of the standard value (zeros for a numeric field, blanks for an alphanumeric field).	
	S	blank
	L	low value
	H	high value
	Z	zero
	Q	quote
	F	Fill with string specified in the parameter Init. value (mandatory). For example: if X is specified and the field length is 4, XXXX will be used for initialization.
	blank	Field will be initialized with the string specified in the field Init. value. If no Init. value is specified, no initialization is performed.
Init. value	If Initialize with is either F or blank a value used for initialization of a field must/can be specified. Length and format of the Init value must be valid for the field. For binary fields hexadecimal constants such as FB0A are valid. Hexadecimal values can be specified in two ways: ■ if Format=B, hexadecimal values can be specified directly. Example: F0 ■ if Format=A, hexadecimal values must be preceded by uppercase X or H and be enclosed in single quotes. Example: X'F0' or H'F0' See also *Condition Name and Value*.	
Indexed by	String that is used when generating the COBOL INDEXED BY clause (only valid for fields of type MU/MC or PE/PC).	
Depending on	String used when generating the COBOL DEPENDING ON clause (only valid for fields of type MU/MC or PE/PC).	

Condition Name and Value

```
13:04:26              *****  P R E D I C T  *****              2007-05-31
                                - Modify Field -
Field ID ........ VE-FIELD                        Modified 2007-05-31 at 12:21
File ID ......... HEB-A                                  by HEB


Ty L Field ID                          F  Cs Length   Occ   D U DB N NAT-l
*- - ------------------------------- *-  - -------- ----- * * -- * -----
   1 VE-FIELD                          A     3.0                AA N

Condition name                 FC * Condition value









  Additional attributes ..* S          Associations ..* N     Scroll to .. 1
```

Attributes		
Condition name	A value to be used when generating either equate data in Assembler copy code or a level 88 entry in COBOL copy code. Up to 29,970 condition names can be entered. Each name needs at least one corresponding condition value. Using condition names can make logical conditions and assignments easier to handle.	
FC	Figurative constant. Valid values:	
	S	blank
	L	low value
	H	high value
	Z	zero
	Q	quote
	F	Fill with string specified in the parameter Condition. value. For example: if X is specified an the field length is 4, XXXX is used as condition value.
	blank	The string specified in the field Condition value is used.
Condition value	The length and format of this value must be valid for this field. This value must have a corresponding condition name.	

Attributes	
	Up to 29,970 condition values can be entered. If several values correspond to the same name, put the name before the first value and leave the name field blank before later values. THRU in the name field indicates a range of values ending with the value on that line and beginning with the value on the previous line.

Field Name Synonyms

```
13:48:12            *****  P R E D I C T  *****              2007-05-31
                             - Modify Field -
Field ID ........ PD-A2                              Added 2007-05-31 at 12:10
File ID ......... PD-A-TEST3                       Modified 2007-05-31 at 13:08


Ty L Field name                        F  Cs Length   Occ   D U DB N NAT-l
*- - ------------------------------- *-  - -------- ----- * * -- * -----
   2 PD-A2                             A     2.0                AF

Field name synonyms
  Natural .............
  COBOL ...............
  PL/I ................
  BAL/Assembler .......
  FORTRAN .............
  Pascal ..............
  Language ADA ........
  Language C ..........
  User defined ........


  Additional attributes ..* S          Associations ..* N
```

Attributes	
Field name synonyms	Synonyms to be assigned to the field when definitions in the following programing languages are generated: Natural, COBOL, PL/I, BAL (Assembler), FORTRAN, PASCAL, ADA or C.

Old Mode Synonyms

This option is only provided for compatibility with old versions of Predict.

Whether this option is displayed depends on the parameter Old mode synonyms of the screen Defaults ->General defaults -> Synonyms:

N Default setting. Compatibility with old versions is not required.

Y Up to 90 synonyms can be defined as Natural synonyms. These create additional entries in the DDM which have the same attributes as the original object but different names.

```
13:10:46             *****  P R E D I C T  *****             2007-05-31
                              - Modify Field -
Field ID ........ PDS                             Modified 2007-05-31 at 13:01
File ID ......... PD-D1                                  by PD

     Field synonym                          Field synonym

  1                                       2
  3                                       4
  5                                       6
  7                                       8
  9                                      10
 11                                      12
 13                                      14
 15                                      16
 17                                      18
 19                                      20
 21                                      22
 23                                      24
 25                                      26
 27                                      28
 29                                      30

  Additional attributes ..* S          Associations ..* N        Scroll to:
```

Adabas Security and Edit mask

```
   17:22:19                ******  P R E D I C T  *****                  2011-09-05
                             - Modify Field -
Field ID ........ HNO-EL-A                            Added 2011-09-05 at 17:16
File ID ......... SMR-AV82                               by HNO


Ty L Field ID                          F Cs Length   Occ   D U DB S NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * -----
   1 HNO-EL-A                          T                       AJ N

Adabas attributes
  Edit mask .............*
  Time zone ..............     (Y/N)
  Daylight saving ........     (Y/N)
  Security access level ..     (0-15)
  Security update level ..     (0-15)





  Additional attributes ..* N          Associations ..* N                      ↩
```

Attributes		
Edit mask	The Adabas edit mask to be used for the field. Determines how numeric fields are to be edited.	
	Valid values for Edit mask rules used in the COBOL programming language:	
	E1...E15. Edit mask is supported for compatibility reasons and for documentation purposes only. See the section *Format Buffer Syntax* in the *Adabas Command Reference* documentation for more information.	
	Valid values for Date-Time Edit masks:	
	D	Date.
	T	Time.
	DT	Datetime.
	TS	Timestamp.
	NT	Natural time.
	ND	Natural date.

Attributes		
	UT	Linux time.
	XTS	Linux timestamp.
	blank	None.
	In Predict, fields with format D by default use Natural date (NATDATE) and fields with format T by default use Natural time (NATTIME) Edit masks. The different Date-Time Edit masks require different minimum numeric field lengths. Detailed information is given in section *Date-Time Edit Mask Reference* of the *Adabas Mainframe* documentation.	
Time zone	The time zone in which a timestamp is applied. Only available for fields using one of the following Adabas edit masks described above: DT, NT, TS, UT or XTS.	
Daylight saving	The daylight savings indicator can only be specified for date-time fields defined with the TZ option. Indicates whether daylight saving applies (Y) or not (N).	
Security access level	The Adabas access security level of the field.	
Security update level	The Adabas update security level of the field.	

DBMS Extensions

Applicable only to DB2 fields of type D or MT, Oracle fields of type OT and to Adabas fields of type A.

The following screen shows the options for DB2 fields of type D and MT.

```
16:16:33                *****  P R E D I C T  *****                  2011-11-17
                              - Modify Field -
Field ID ........ SUPER_HASH                          Added 2011-10-17 at 15:29
File ID ......... XYZ-D_HASH_PART                        by XYZ


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
SP 1 SUPER_HASH                                           D U AC

DBMS extensions in <Default Server>
  Use as security label ..   (Y,N)
  Partitioning ..........* Q DPSI and table partitioning key
  Use as hash key ........ Y (Y,N)
  Cluster index .......... N (Y,N)
  Exclude NULL keys ...... Y (Y,N)
                                                                              ↩
* Additional attributes ..* S          Associations ..* N                    ↩
```

Additional attributes are available for files of type D and MT that are used as accelerator tables.

```
18:17:10                *****  P R E D I C T  *****                  2020-07-02
                              - Modify file -
File ID ......... XYZ-DB_V12                       Modified 2020-04-06 at 13:35
                                                         by XYZ

DBMS extensions in <Default Server>
  Keylabel name .... on default                                        Zoom: N
  Accelerator name . ACC IN D

* Additional attributes ..* S        * Associations ..* S                    ↩
```

The following screen shows the options for Adabas fields of type A.

```
15:03:54                *****  P R E D I C T  *****                  2011-11-10
                              - Modify Field -
Field ID ........ AA-FIELD                          Modified 2011-08-05 at 09:28
File ID ......... SMR-AV82                                 by XYZ


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
MU 1 AA-FIELD                          A        8.0  300       AA


DBMS extensions
  System field .........* JN Jobname









* Additional attributes ..* S          Associations ..* N                     ↩
```

The following screen shows the options for Oracle fields of type OT.

```
17:55:26                *****  P R E D I C T  *****                  2017-07-31
                              - Modify Field -
Field ID ........ XYZ_OT_INDEX                      Modified 2017-07-31 at 17:50
File ID ......... XYZ-OTI                                  by XYZ


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
   1 XYZ_OT_INDEX                      A       15.0            AL U

DBMS extensions in XYZ-ORA1
  Partitioning key ....*
  Partitioned index ...*
  Sort ................   (Y/N)
  Cluster column .......   (Y/N)
  Encryption ..........*
  Integrity algorithm .*
  Salt ................   (Y/N)
  Include column .......   (Y/N)



* Additional attributes ..* S          Associations ..* N                     ↩
```

Attributes		
Use as security label	Specifies that the table column will contain security label values. A table can have only one security label column. To define a table with a security label column, the primary authorization ID of the statement must have a valid security label, and the RACF SECLABEL class must be active. In addition, the following conditions are also required: ■ The data type of the column must be CHAR(8). ■ The subtype of the column must be SBCS. ■ The column must be defined with the NOT NULL and WITH DEFAULT clauses. ■ No field procedures, check constraints, or referential constraints are defined on the column. ■ An edit procedure is not defined on the table. Refer to your *DB2* documentation for further details.	
Partitioning	Indicates for each implementation of a table in an SQL server, which field controls the partitioning. Valid values:	
	blank	Non-partitioning. The current field is not a partitioning key and has no index.
	T	Table partitioning key. The table is data partitioned and the current field is the partitioning key. Additionally, it may have a non-partitioned index.
	I	Index controlled key. The table partitioning is controlled by a partitioning index. The current field must have an index defined; this is indicated by descriptor type D (index), P (primary key) or F (foreign index).
	N	Non-partitioned secondary index. The field has a non-partitioned index, but it is not the partitioning key.
	P	Data partitioned secondary index (DPSI). The field has a partitioned index, but it is not the partitioning key. Within the file a different field must be defined as partitioning key.
	Q	Table partitioning key and DPSI. The table is data partitioned and the current field is the partitioning key. In addition, a partitioned index is defined for the field.
Cluster index	Only available for descriptor type D (index), P (primary key) or F (foreign index).	

Attributes		
	Y	The records (rows) in the DB2 table are stored in the sequence of this index. Valid for max. one index per table. A table contained in a partitioned tablespace must have one index marked as a clustered index, if the partitioning option Index controlled key is to be used.
Use as hash key	Only available for DB2 fields of type D. Indicates whether the current field is used as hash key. Must be unique.	
	Y	Yes. Must be set if the table is hash organized.
	N	No.
Exclude NULL keys	Only available for DB2 fields of type D and MT. Indicates whether NULL keys are excluded.	
	blank	Not specified.
	Y	Yes.
	N	No.
System field	Only available for Adabas fields of type A. Type of the system-generated field in Adabas. Valid values:	
	JN	Job name. For alphanumeric fields only. Length must be 8 or greater.
	OU	Open command user. For alphanumeric fields only. Length must be 8 or greater.
	SC	Security user. For alphanumeric fields only. Length must be 8 or greater.
	SI	Session ID. For alphanumeric fields only. Length must be 28 or greater.
	SU	Session user. For alphanumeric fields only. Length must be 8 or greater.
	T	Time. For numeric fields only.
Index partitioning as default	**Note:** This option can only be set for DB2 fields of type D and MT using the **Default Table Attributes** function in the Default DB2 Attributes screen. Defines the preset values for the Partitioning and Cluster index attributes (see above in this table) when a new field is created using the **Add a Field** function. Valid values:	
	Y	If a new field is created using the Add a Field function and no cluster field exists on the file, the preset value for Partitioning is I (Index controlled key) and the preset value for Cluster index is Y.
	N	This is the default. The preset value for Partitioning is N (Non-partitioned secondary index) and the preset value for Cluster index is N.

Attributes		
Keylabel name	This option can only be set for DB2 fields in files of type D and MT. Used for encryption.	
Accelerator name	This option can only be set for DB2 fields in files of type D and MT that are used as accelerator tables. For a definition of accelerator tables, refer to the IBM DB2 documentation.	
Partitioning key	Only available for Oracle fields of type OT. Indicates how the partitioning key is organized:	
	R	Range partitioned.
	H	Hash partitioned.
	blank	None.
Partitioned index	Only available for Oracle fields of type OT. Indicates how the partitioned index is organized:	
	A	Global range.
	B	Global hash.
	C	Local range.
	D	Local hash.
	blank	None.
Sort	Only available for Oracle fields of type OT. Specifies whether the rows of the cluster on this column are sorted after applying the hash function	
Cluster column	Only available for Oracle fields of type OT. Specifies whether or not this field (or combination of fields in case of an SP field) is used as cluster column	
Encryption	Only available for Oracle fields of type OT. Specifies the encryption method to be used:	
	1	AES256
	2	AES192
	3	AES128
	4	3DES168
	blank	None.
Integrity algorithm	Only available for Oracle fields of type OT. Specifies the integrity algorithm to be used:	
	S	SHA1
	N	NOMAC
	blank	None.
Salt	Only available for Oracle fields of type OT.	

Attributes	
	Specify SALT to instruct the database to append a random string to the clear text of the column before encrypting it.
Include column	Only available for Oracle fields of type OT. If this column is part of the primary key, this column determines where to divide an index-organized table row into index and overflow portions.

Field Procedure

```
16:41:49             *****  P R E D I C T  *****              2007-05-31
                              - Add a Field -
Field ID ........ HNO_TEST                            Added 2007-05-31 at 16:41
File ID ......... HNO-DB2                                by HNO


Ty L Field ID                          F Cs Length   Occ   D U DB N NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * -----
   1 HNO_TEST                          AL     100.0            AF U

Physical attributes in <Default Server>                 (new)
            +-------------------------------------------------+
            ! Procedure name .......                          !
            ! Procedure parameter                             !
            !                                                 !
            !                                                 !
            !                                                 !
            !                                                 !
            !                                                 !
            !                                /                !
            +-------------------------------------------------+

  Additional attributes ..* S          Associations ..* N                    ↩
```

Attributes	
Procedure name	Name of a field procedure (DB2 parameter FIELDPROC). See the *Natural for DB2* documentation for more details.
Procedure parameter	Parameters passed to the field procedure. See the *Natural for DB2* documentation for more details.

Derived Field Expression

```
>                      > + EL: A5                                L: 1    S: 1
    All  ....+....1... Expression for derived field..+....5....+....6....+....7..


  *
  * Field expression of a derived field
  *
  USER-TABLE1-SALARY * 12     /* SALARY FOR 12 MONTHS
  + CORRELATOR2-BONUS
```

Applicable only to fields of type DV in files of the following types:

B	Adabas SQL views
E	DB2 views
IV	Intermediate view
JV	Ingres view
OV	Oracle view
X	General SQL
XV	Informix view
YV	Sybase view

The expression used to derive the field is to be edited using one of the following depending on the your settings in the Profile > Handling screen:

- the Natural-based Subquery Editor, or
- the Software AG Editor

The editor can also be called with

- function Edit Field expression (Code Y) in the Field Maintenance Menu, or
- command `EDIT ELEMENT EXPRESSION` *`file-id field-id`*

See the section *Editors in Predict* in the *Predict Reference* documentation for more information.

The subquery of the file that contains the current field can specify a correlation name for any file whose fields it uses. The name of each field referenced in the expression must be qualified (preceded) by the correlation name of the file from which the field is taken, if a correlation namehas been specified for that file, or the ID of the file from which the field is taken, if no correlation name has been specified for it. The expression can include both comment lines (with /*, * or ** in the first two columns) and line comments (preceded by /*).

Example: A field which contains the annual salary:

```
*
* Field expression of a derived field
*
USER-TABLE1-SALARY * 12     /* SALARY FOR 12 MONTHS
+ CORRELATOR2-BONUS
↩
```

Index Definition - DB2

Index fields (descriptor type D, F or P) in a file of type D (DB2 table), are defined in the screens below.

- Index Default Definition / Defaults Values of Using- and Free-Block
- Partition definition

Index Default Definition / Defaults Values of Using- and Free-Block

```
0:55:34                ***** P R E D I C T  *****                  2020-07-02
                              - Modify Field -
Field ID ........ EL_IDX                              Added 2020-01-28 at 09:26
File ID ......... XYZ-DB_V12                             by XYZ

Index default definition in <Default Server>
  Index name ......... XYZ-EL_IDX                   Number of partitions: 88
  Cluster index ...... N (Y,N)                      Piece size ...* 0
  Close option ....... Y (Y,N)                      Reverse scans .   (Y,N)
  Bufferpool ........*                              Padded ........   (Y,N)
  Copy ............... N (Y,N)                      Compression ...   (Y,N)
  UQ where not null .*    determined by gener.      DSSIZE (G) ....
Default values of using- and free-block
  VSAM catalog name ..
  Storagespace ......*
  Primary alloc ......
  Secondary alloc.....
  Erase opt ..........   (Y,N)
  Free pages .........
  Percentage free ....
  GBPCACHE ..........*    Default

* Additional attributes ..* S          Associations ..* N                    ↩
```

Attributes	
Index default definition	
Index name	The name of the DB2 index. See *Key or Index Fields in SQL Files - Superfields*. A read-only field.
Cluster index	Y: The records (rows) in the DB2 table are stored in the sequence of this index. Valid for max. one index per table. A table contained in a partitioned tablespace must have one index marked as a clustered index, if the partitioning option Index controlled key is to be used..
Bufferpool	The buffer pool associated with the index.
Close option	Y: The data sets supporting this index are closed when nobody uses the index.
Copy	Indicates whether the copy utility is allowed for the index. Y: Full image or concurrent copies allowed. N: Full image or concurrent copies not allowed.
Piece size	The maximum piece size for a non-partitioned index. Valid values: 0, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768, 65536, 131072, 262144, 524288, 1048576, 2097152, 4194304.
Reverse scans	Specifies whether an index supports forward scans only or reverse scans of the index also. Y: Reverse scans allowed. N: Reverse scans not allowed. This is the default.
Padded	Specifies how varying-length string columns are to be stored in the index. Y: Padded. N: Not padded. blank: Not specified.
UQ where not null	Null value option. Specifies whether a null value is allowed for a unique index. Valid values: U: Null allowed. Any two null values are taken to be unequal. If the key is a single column, that column can contain any number of null values, though its other values must be unique. R: Not null. Any two null values are taken to be equal. For example, if the key is a single column, that column can contain no more than one null value. blank: Determined by gener. Null value option is set via default server settings in Predict.
Compression	Specifies whether compression for index data will be used. Y: Compression is used. N: No compression is used. This is the default.
DSSIZE	Specifies the maximum size (in Gbyte) for each partition of a partitioned index.

Attributes		
Index default definition		
Default Values of Using- and Free-Block		
VSAM catalog name	Name of the VSAM catalog for the index. The parameter VSAM catalog name must be specified if data sets are already defined for the index.	
Storagespace	Storagespace where DB2 defines the data sets for the index (optional). If no storagespace is specified, DB2 uses the default storagespace.	
Primary alloc	Minimum primary space allocation (in Kbyte) for DB2-defined index data sets. A value specified in this field is stored only if the attribute storage group has been specified.	
Secondary alloc	Minimum secondary space allocation (in Kbyte) for DB2-defined index data sets. A value specified in this field is stored only if the attribute storage group has been specified.	
Erase opt	Y	The DB2-defined data sets are to be erased (filled with nulls) when the index is dropped.
	A value specified in this field is stored only if the attribute Storage group has been specified.	
Free pages	A number from 0 to 255 which indicates that one page is to be left free each time this number of pages is used when the load operation creates index entries or when the index is reorganized. Zero indicates that no pages are to be left free.	
Percentage free	A number from 0 to 99: the percentage of each page to be left as free space when index entries are created by a load operation or when the index is reorganized.	
GBPCACHE	Only relevant in a data sharing environment. Specifies what pages of the table space or partition are written to the group buffer pool. Leave this field blank or enter:	
	C	Changed. Only pages that have been changed are written to the group buffer pool.
	A	All pages are written.
	N	No pages are written to the group buffer pool.

Partition definition

For a partitioned index (a cluster index for a table in a partitioned table space), the following screen is displayed for every two partitions. Each partition can then be defined in accordance with the Default values of using- and free-block (see description above).

```
21:13:44                ******  P R E D I C T  *****                  2020-07-02
                              - Modify Field -
Field ID ........ EL_IDX                          Modified 2020-07-02 at 21:12
File ID ......... XYZ-DB_V12                             by XYZ
SQL Server ...... <Default Server>
---------------------------- Partition definition -----------------------------
Partition 1
  Value .....                                                            Zoom: N
  VSAM catalog name .....                 DSSIZE (G) .......
  Storagespace .........*
  Primary allocation ....                 GBPCACHE ........*
  Secondary allocation ..                 Free pages .......
  Erase option ..........   (Y/N)        Percentage free ..
                                          Hash size (KB)....
Partition 2
  Value .....                                                            Zoom: N
  VSAM catalog name .....                 DSSIZE (G) .......
  Storagespace .........*
  Primary allocation ....                 GBPCACHE ........*
  Secondary allocation ..                 Free pages .......
  Erase option ..........   (Y/N)        Percentage free ..
                                          Hash size (KB) ...
* Additional attributes ..* N          Associations ..* N  * Scroll to .. 1    ↩
```

Note: The data displayed in this screen depend on the values selected in the Partitioning field of the **DBMS extensions** screen.

Attributes	
Value	The highest value of the index key in this partition. At least one constant must be used and as many constants as there are columns in the key can be specified. The concatenation of all the constants is the highest value of the key in this partition of the index. **Note:** No checking is performed here.
DSSIZE	The value of DSSIZE given in the PARTITION clause for that partition.

All other attributes are described above.

Index Definition - Oracle

Index fields (descriptor type D, F or P) in a file of type OT (Oracle table), are defined in the screens below.

- Index Attributes
- Index Segment Attributes
- Index Storage Attributes
- Partitioning Key and Partitioned Index

Index Attributes

```
14:30:37                ******  P R E D I C T  *****                  2017-08-11
                              - Modify Field -
Field ID ........ IDENTITY                            Added 2017-08-11 at 14:30
File ID ......... XYZ-OTI                                by XYZ

Index attributes in XYZ-ORA1
  REVERSE .............    (Y,N)
  VISIBLE ............. Y  (Y,N)
  PARTIAL ............. Y  (Y,N)
  SORT ................ Y  (Y,N)
  COMPRESSION ........* Y  Yes
  Compression length .. 23









* Additional attributes ..* S          Associations ..* N                    ↩
```

Parameters		
Field ID	ID of the Predict object documenting the Oracle table field.	
File ID	ID of the Predict object documenting the Oracle table. See *Naming Conventions* for Oracle objects.	
Index attributes		
REVERSE	Specifies whether the index is a reverse key index.	
	Y	Yes.
	N	No.

Parameters		
VISIBLE	Specifies whether the index is visible.	
	Y	Yes.
	N	No.
PARTIAL	Specifies whether the index is a partial index in a partitioned table.	
	Y	Yes.
	N	No.
SORT	Specifies whether the index is sorted in ascending order.	
	Y	Yes.
	N	No.
COMPRESSION	Specify whether or not to use compression.	
	Y	Yes.
	N	No.
	A	Advanced.
	blank	Not specified. This is the default.
Compression length	Specify the compression length.	

Index Segment Attributes

```
14:15:26                ******  P R E D I C T  *****                  2017-08-11
                              - Modify Field -
Field ID ........ IDENTITY                         Modified 2017-08-11 at 14:15
File ID ......... XYZ-OTI                                 by XYZ

Index segment attributes in XYZ-ORA1
  PCTFREE ........ 12
  PCTUSED ........ 2
  INITRANS ....... 33
  Tablespace ..... TAB
  LOGGING .......* F File system like









* Additional attributes ..* S          Associations ..* N                    ↩
```

Index segment attributes		
PCTFREE	If an integer from 1 - 99 is specified here, the clause `PCTFREE` *n* is generated in the `CREATE TABLE` statement. `PCTFREE` reserves a set amount of room in every block allocated to a table for future updates to that table's data.	
PCTUSED	If an integer from 1 - 99 is specified here, the clause `PCTUSED` *n* is generated in the `CREATE TABLE` statement. `PCTUSED` specifies the minimum level of space usage that Oracle will maintain for each block of the table.	
INITRANS	If a value from 1 - 255 is entered here, the clause `INITRANS` *n* is generated in the `CREATE TABLE` statement. `INITRANS` is the initial number of transaction entries that are allocated within each block.	
Tablespace	If a tablespace name is entered here, the clause `TABLESPACE` name is generated in the `CREATE TABLE` statement. This name represents the tablespace in which the table will be created.	
LOGGING	Specify whether or not to use the `LOGGING` clause in a `CREATE TABLE` or `ALTER TABLE` statement.	
	Y	Yes.
	N	No.
	F	File system like.
	blank	Not specified. This is the default.

Index Storage Attributes

```
14:17:52                *****  P R E D I C T  *****                  2017-08-11
                              - Modify Field -
Field ID ........ IDENTITY                        Modified 2017-08-11 at 14:15
File ID ......... XYZ-OTI                               by XYZ

Index storage attributes in XYZ-ORA1
  INITIAL .............            Unit .*
  NEXT ................            Unit .*
  MAXSIZE .............            Unit .*                (-1 for unlimited)
  OPTIMAL .............            Unit .*                (-1 for null)
  MINEXTENTS ..........
  MAXEXTENTS ..........                                   (-1 for unlimited)
  PCTINCREASE .........
  FREELISTS ........... 1
  FREELIST GROUPS ..... 3
  BUFFERPOOL .........* D Default
  FLASH_CACHE ........* D Default




* Additional attributes ..* S          Associations ..* N                    ↩
```

Index storage attributes		
If specified, the values below are used in the `STORAGE` clause generated with the `CREATE TABLE` statement. All of the values below must be specified as integers.		
INITIAL	The size of the first extent allocated when the object is created - the original amount of space allocated to the object. A value for Unit has to be applied in addition:	
	K	Kilobyte.
	M	Megabyte.
	G	Gigabyte.
	T	Terabyte.
	P	Petabyte.
	E	Exabyte.
NEXT	The size of every subsequent extent to be allocated. A value for Unit has to be applied in addition. Possible values for Unit are described under INITIAL.	
MAXSIZE	The `MAXSIZE` clause lets you specify the maximum size of the storage element.	
OPTIMAL	Specifies an optimal size in bytes for a rollback segment.	
MINEXTENTS	The total number of extents to be allocated when the segment is created.	
MAXEXTENTS	The total number of extents, including the first, which can ever be allocated.	
PCTINCREASE	The percent by which each NEXT extent will grow over the last extent allocated.	
FREELISTS	The number of process free lists used to administer the free data blocks.	

Index storage attributes		
FREELISTS GROUPS	Magnitude of the set of free lists.	
BUFFERPOOL	Determines the configuration of the buffer cache.	
	D	Default
	K	Keep
	R	Recycle
	blank	not specified
FLASH_CACHE	Defines the configuration of a second tier of buffer cache on flash disks.	
	D	Default
	K	Keep
	N	None
	blank	not specified

Partitioning Key and Partitioned Index

For a partitioning key and partitioned index, the following can be defined depending on whether they are hash- or range-partitioned.

Note: The data displayed in this screen depend on the values selected in the Partitioning key field of the ***DBMS Extensions*** screen.

- Range-Partitioning Key Definitions
- Hash-Partitioning Key Definitions

Range-Partitioning Key Definitions

```
16:03:24                *****  P R E D I C T  *****                  2017-08-17
                              - Modify Field -
Field ID ........ XYZ_OT_INDEX                     Modified 2017-08-17 at 16:02
File ID ......... XYZ-OTI                                 by XYZ
Database ........ XYZ-ORA1
---------------------- Range partitioning key definitions ---------------------
Partition 1
  Name ......
  Literal ...                                                            Zoom: N
  More attributes
    Segment ............... N (Y/N)
    Storage ............... N (Y/N)


Partition 2
  Name ......
  Literal ...                                                            Zoom: N
  More attributes
    Segment ............... N (Y/N)
    Storage ............... N (Y/N)


* Additional attributes ..* S          Associations ..* N  * Scroll to .. 1    ↩
```

Attributes	
Name	The partition name.
Literal	The literal name.
Segment	Specifies the Segment attributes of the partition.
Storage	Specifies the Storage attributes of the partition.

Hash-Partitioning Key Definitions

```
16:10:45                *****  P R E D I C T  *****                  2017-08-17
                              - Modify Field -
Field ID ........ XYZ_OT_INDEX                     Modified 2017-08-17 at 16:02
File ID ......... XYZ-OTI                                 by XYZ
Database ........ XYZ-ORA1
---------------------- Hash partitioning key definitions ----------------------
Partition 1
  Name ................
  Tablespace ..........
  COMPRESSION ........*    (not specified)
  Compression length ..



Partition 2
  Name ................
  Tablespace ..........
  COMPRESSION ........*    (not specified)
  Compression length ..



* Additional attributes ..* S          Associations ..* N  * Scroll to .. 1    ↩
```

Parameter	Description	
Name	The partition name.	
Tablespace	The tablespace name.	
COMPRESSION	Specify whether or not to use compression.	
	Y	Yes.
	N	No.
	A	Advanced.
	blank	Not specified. This is the default.
Compression length	Specify the compression length.	

Default value

This additional attribute is only applicable for fields in

- Sybase tables with Null value option set to R and Null default option set to Y.
- Adabas D tables, DB2 tables, Informix and Oracle tables with Null value option set to R or U and Null default option set to Y.

```
13:09:33             *****  P R E D I C T  *****             2007-05-31
                              - Modify Field -
Field ID ........ TESTFIELD                        Modified 2007-05-31 at 13:09
File ID ......... HNO-YT                                 by HNO


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
   1 TESTFIELD                         A  B    10.0            AA R Y


Default name .............
Default expression ......<



                                                        >
```

Attribute	
Default name	The default specified here is used in the CREATE TABLE statement. Sybase naming conventions apply. See *Naming Conventions for SQL Objects*. **Note:** For Sybase, a default is an object in its own right. For other SQL systems, a default value is specified in the CREATE TABLE statement (not null with default default_expression). For Informix no default name is allowed.
Default expression	An SQL expression can be specified between the angled brackets. This expression determines the default value, for example a constant or function. If specified, this value is always used by the function Generate CREATE statement.

Constraint name

Depending on the field definition, up to four constraint names can be specified.

```
13:35:45             *****  P R E D I C T  *****             2007-05-31
                              - Modify Field -
Field ID ........ TESTFIELD                       Modified 2007-05-31 at 09:34
File ID ......... HNO-YT                                 by HNO


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - -------------------------------- *- * -------- ----- * * -- * *- -----
   1 TESTFIELD                         A  B    10.0        P U AA R Y


     Attributes        Constraint name

      Check constraint
      Primary key
      Unique
      Not null
```

Attributes	
Check constraint	Constraint name in the respective SQL system for the fact that a linked verification of status S exists.
Primary key	Constraint name for the fact that the field is a primary key.
Unique	Constraint name for the fact that a unique constraint exists (indicated with U in column Unique option of the field object in Predict).
Null/Not null	Constraint name for the fact that the Null or Not null default option is set to Y.

Identity definition / Change log

These additional attributes are only applicable for fields in DB2 or Oracle tables.

Identity definition

The following additional attributes are only applicable for fields of type

- QN (Row ID) or
- a numeric field.

```
 15:56:16             *****  P R E D I C T  *****                    2009-05-31
                              - Modify Field -
Field ID ........ XYZ_AC                              Added 2009-05-31 at 15:56
File ID ......... XYZ-DB2                                by XYZ


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
QN 1 XYZ_AC                            A       40.0          U AB R

Identity .............   (Y,N)
Identity definition in <Default Server>
  Generated .........* A  Always

  Start value ........
  Increment value ....
  Cache ..............
  Cycle ..............   (Y,N)
  No minvalue ........   (Y,N)      Min value ..
  No maxvalue ........   (Y,N)      Max value ..
  Order ..............   (Y,N)

* Additional attributes ..* S          Associations ..* N                    ↩
```

Attributes		
Identity	Specifies whether the column is an identity column for the table. Valid values:	
	Y	Yes
	N	No
Generated	Indicates whether DB2 or Oracle generates values for the column. Valid values:	
	A	Always
	D	By default
	blank	None
Start with limit	The first generated value will be the highest/lowest existing value +/- increment value	
Start value	Specifies the first value for the identity column.	
Increment value	Specifies the interval between consecutive values of the identity column.	
Cache	Specifies whether to keep preallocated values in memory.	
Cycle	Specifies that values continue to be generated for this column after the maximum or minimum value has been reached.	
Min value	Specifies the numeric constant that is the minimum value that is generated for this identity column.	
No minvalue	Specifies for DB2 tables whether or not a minimum end point of the range of values for the identity column has been set.	
Max value	Specifies the numeric constant that is the maximum value that is generated for this identity column.	

Attributes		
No maxvalue	Specifies for DB2 tables whether or not a maximum end point of the range of values for the identity column has been set.	
Order	Specifies whether the sequence numbers must be generated in order of request. Valid values:	
	Y	Order
	N	No order
	blank	Not specified

Change log

The following additional attributes are only applicable for fields of type

- TS (Timestamp)

```
14:16:40             *****  P R E D I C T  *****                     2009-07-31
                              - Modify Field -
Field ID ........ EL_TS                              Added 2009-02-13 at 08:56
File ID ......... HEB-D                            Modified 2009-06-04 at 14:14


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
   1 EL_TS                             TS                      AJ U

Change log ............. N (Y,N)
Change log definition in DADB29
  Generated ...........* A  Always









* Additional attributes ..* S          Associations ..* N                    ↩
```

Attributes		
Change log	Specifies whether DB2 generates a value for the column for each new row when the row is inserted and/or for any row in which a column is updated. Valid values:	
	Y	Yes
	N	No
Generated	Indicates whether DB2 generates values for the column. Valid values:	

Attributes		
	A	Always
	D	By default
	blank	None

Platform Compatibilities

The following topics are covered:

- No Blank Compression

No Blank Compression

This additional attribute is only applicable for Adabas fields with format A, AV or LO. The option No Blank Compression controls that trailing blanks are suppressed when a value is stored.

```
 16:12:31             *****  P R E D I C T  *****                    2007-05-31
                              - Add a Field -
Field ID ........ HNO_A                                Added 2007-05-31 at 16:11
File ID ......... HNO-ADA                                 by HNO


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
   1 HNO_A                             A       10.0            AA N


Platform compatibilities
  High order first .....    (Y)
  PF option ............    (Y)
  No Blank Compression .    (Y)





  Additional attributes ..* S          Associations ..* N                    ↩
```

Base Extensions

The following topics are covered:

- Fractional of Seconds and Time Zone
- Inline Length
- Generated Expression
- Period

Fractional of Seconds and Time Zone

These additional attributes are applicable for DB2 and Oracle table fields with format TS.

```
17:42:38             *****  P R E D I C T  *****                    2011-05-31
                              - Modify Field -
Field ID ........ HNO_D2T_TS                          Added 2011-05-31 at 17:41
File ID ......... HNO-D2                                 by HNO


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
   1 HNO_D2T_TS                        TS                  N   AA U N

Base extensions
  Fractional of seconds .* 12
  Time zone .............     (Y/N)









  Additional attributes ..* S          Associations ..* N                    ↩
```

Attributes	
Fractional of seconds	A timestamp value can include a precision of fractional seconds in the range from 0 to 12 (Oracle 0-9). For DB2, if no value is entered, the default value 6 is taken.
Time zone	Provides information on the time difference in hours and minutes between the local time and Coordinated Universal Time (UTC). Adds an additional time zone offset value to the local timestamp value in the format +HH:MM (ahead of UTC) or -HH:MM (behind UTC).

Inline Length

This additional attribute is applicable for DB2 LOB fields.

```
17:42:38            *****  P R E D I C T  *****                      2011-05-31
                             - Modify Field -
Field ID ........ HNO_LOB_D2                          Added 2011-05-31 at 17:41
File ID ......... HNO-D2-LOB                             by HNO


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
   1 HNO_LOB_D2                        LO S     1.0M           AA U

Base extensions
  Inline length .......... 8000









  Additional attributes ..* S          Associations ..* N                    ↩
```

Attributes	
Inline length	For BLOB, CLOB and DBCLOB columns. Specifies the maximum number of bytes that are stored in the base table space for the column. Must be between 0 and 32680 for a BLOB or CLOB column and between 0 and 16340 for a DBCLOB column..

Generated Expression

These additional attributes are applicable for DB2 table fields with formats A, AL and AV.

```
12:41:28                *****  P R E D I C T  *****                  2015-06-30
                             - Modify Field -
Field ID ........ EL_GEN_EXPR                      Modified 2015-06-11 at 13:17
File ID ......... XYZ-DB2_API                            by XYZ


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
   1 EL_GEN_EXPR                       A        4.0            AA U

Base extensions
  Generated expression ..* PN  PACKAGE_NAME









* Additional attributes ..* S          Associations ..* N                    ↩
```

Attributes		
Generated expression	For fields of DB2 tables (field types A, AL and AV) you can specify an *as-generated-expression-clause*. The value provided in Generated expression is the expression that is evaluated by DB2 when generating the value of the field. Valid values:	
	DC	Data change operation. Equivalent DB2 expression: `DATA CHANGE OPERATION`
	CC	Client accounting. Equivalent DB2 expression: `CURRENT CLIENT_ACCTNG`
	CN	Client application name. Equivalent DB2 expression: `CURRENT CLIENT_APPLNAM`
	CT	Client correlation token. Equivalent DB2 expression: `CURRENT CLIENT_CORR_TOKEN`
	CU	Client user ID. Equivalent DB2 expression: `CURRENT CLIENT_USERID`
	CW	Client workstation name. Equivalent DB2 expression: `CURRENT CLIENT_WRKSTNNAME`
	CS	Client server. Equivalent DB2 expression: `CURRENT SERVER`
	CI	SQL ID. Equivalent DB2 expression: `CURRENT SQLID`
	SU	Session user. Equivalent DB2 expression: `SESSION_USER`

Attributes		
	PN	Package name. Equivalent DB2 expression: `SYSIBM.PACKAGE_NAME`
	PS	Package schema. Equivalent DB2 expression: `SYSIBM.PACKAGE_SCHEMA`
	PV	Package version. Equivalent DB2 expression: `SYSIBM.PACKAGE_VERSION`
	blank	None.

Period

The field option Period specifies whether the value of the end column is included or excluded in the period BUSINESS_TIME.

This option can be used only for fields named BUSINESS_TIME in files of type *D* and *MT*.

```
19:32:52                *****  P R E D I C T  *****                  2020-07-02
                             - Modify Field -
Field ID ........ BUSINESS_TIME                    Modified 2020-03-11 at 11:08
File ID ......... HEB-DB_V12                             by HEB


Ty L Field ID                          F Cs Length   Occ   D U DB N Df NAT-l
*- - ------------------------------- *- * -------- ----- * * -- * *- -----
SP 1 BUSINESS_TIME                                            AF

Base extensions
  Generated expression ..*     (none)
  Period ................* I   Inclusive









* Additional attributes ..* S          Associations ..* N                    ↩
```

Attributes		
Period	Specifies whether the value of the end column is included or excluded in the period BUSINESS_TIME.	
	I	Inclusive
	E	Exclusive

18 Field Maintenance

- Add a Field - Code A 196
- Copy Field - Code C 196
- Move Field within a File - Code H 197
- Purge Field - Code P 197
- Redefine Field - Code R 198
- Browse Through Fields of a File - Code B 202
- Edit Field Expression - Code Y 203

The Field Maintenance screen is shown in the section *Field Maintenance Menu*.

Note: Predict does not perform complete consistency checks when executing field maintenance functions. It is therefore possible to spoil the integrity of field lists of files temporarily. Consistency checks are performed, however, when field lists are cataloged.

Note: Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation.

Add a Field - Code A

The function Add a Field can be used to add a field

- to the end of the specified file (copy field ID=blank)
- to the top of the specified file (copy field ID=*)
- after a specified field (copy field ID=Field_2) in the specified file.

The position of a new field is determined with the parameters Copy field ID and Copy file ID as follows.

Parameter/Function	ADD to end	ADD to top	ADD to position
Field ID	Field_1	Field_1	Field_1
in file	File_1	File_1	File_1
Copy field ID	-	*	Field_2
Copy file ID	-	-	-

Command: `ADD ELEMENT`

Copy Field - Code C

This function is useful for creating a new field entry. A field can be copied:

- to the end of the same file and renamed (copy field ID=Field_2)
- to another file (copy file ID=File_2) to a position after a specified field (copy field ID=Field_2)
- to the end of another file (copy file ID=File_2).

The position of a copied field is determined with the parameters Copy field ID and Copy file ID as follows.

Parameter/Function	COPY and rename	COPY to position	COPY to end
Field ID	Field_1	Field_1	Field_1
in file	File_1	File_1	File_1
Copy field ID	Field_2	Field_2	
Copy file ID	-	File_2	File_2

Command: `COPY ELEMENT`

Move Field within a File - Code H

This function is used to change the order of fields in a file. A field can be moved:

- to the top of the file (copy field ID=*)
- to a position after a specified field (copy field ID=Field_2).

The position of a moved field is determined with the parameters Copy field ID and Copy file ID as follows.

Parameter/Function	MOVE to top	MOVE to position
Field ID	Field_1	Field_1
in file	File_1	File_1
Copy field ID	*	Field_2
Copy file ID	-	-

If the function Move is applied to group fields (type GR, PE, PC), all fields of the group will be moved.

Command: `MOVE ELEMENT`

Purge Field - Code P

Predict objects of type field can be deleted with the Purge function (code P). The following rules apply.

- If the field to be deleted is a group, it is possible to delete all fields in the group.
- If the field is a standard field, connections to derived fields are deleted.
- If a field in a master file is deleted, all fields in userviews related to this field are deleted as well.

If you confirm the Purge function with DELETE,

- The field and its redefinition will be purged, and
- All file relations based on this field will be set to Documented.

Command: `PURGE ELEMENT`

Redefine Field - Code R

The function Redefine Field (code R) invokes the list editor for defining a redefinition (see screen below). A redefinition must be cataloged. Predict reports errors in a redefinition.

Command: `REDEFINE ELEMENT`

Note: No consistency check is performed for files of type Conceptual or Standard.

```
>                      > + Fi: HNO-FI1                            L: 1    S: 1
    Ty L Field ID                          F  Cs Length   Occ   D U DB S      All
    *- - -------------------------------- *-  - -------- ----- * * -- *
       1 HNO-EL1                           A         2.0            AA N
```

Field Types that can be Redefined

blank	Normal fields
DV	Derived fields (SQL)
GR	Group
HM	Hyperdescriptors as a multiple value fields
HP	Hyperdescriptors as a fields of a periodic group
HQ	Hyperdescriptors as a multiple value fields of a periodic group
HY	Hyperdescriptors
MC	Multiple value fields with automatic counter
MU	Multiple value fields
OD	Collation descriptor
PC	Periodic group with automatic counter
PE	Periodic group
QN	SEQNO field

SB	Subfields/descriptors
SP	Superfields/descriptors

If a field of type MU, MC, PE or PC is redefined, the whole array (including all occurrences) is redefined. When redefining fields of these types, the occurrence number must be specified.

Position and Format of a Redefinition

A redefinition is started by defining a field of type RE having the same level and Field ID as the field to be redefined.

This field definition has to directly follow the redefined field/group.

```
Ty L Field ID                          F  Cs Length   Occ
*- - ------------------------------- *-  - -------- -----
     1   FIELD-TEST                    A     20
RE   1   FIELD-TEST
     2   TEST-REDEF1                   A     5
     2   FILLER                        A     5
     2   TEST-REDEF2                   A     5
```

Format and Type of Fields within the Redefinition

The following field types can be used within a redefinition: blank, MU, GR or PE. Redefinition within the redefinition is possible. The number of occurrences must be specified for MU and PE fields. Format of occurrences are increased to N5. The occurrences can be specified in the Edit Elements of a File screen.

Properties of Fields within a Redefinition

Redefined fields have the following properties:

- The sum of the length of all fields in a redefinition must not be greater than the length of the field/group being redefined.
- The field level within a redefinition cannot be greater than 9.
- PE in PE is possible.
- The special field name FILLER is not tested for uniqueness and can be used to exclude parts of the original field from redefinition (as in previous versions of Predict).
- A field can be identified by a maximum of three indexes.
- Within redefinitions, Gr.structur must not be set to N.

Purge and Rename a Redefined Field

If a field that has been redefined is purged or renamed, all redefinitions of the field are purged or renamed as well.

Mapping of Natural Data Structures

Natural allows the definition of multiple arrays for one field. Data structures of this type cannot be defined in Predict and have to be circumscribed as shown in the two examples below.

Note: A three-dimensional array is only possible for few file types; e.g. sequential (file **type S**).

Natural structure

```
1 GROUP      (1:2,1:4,1:3)
2 ELE   (A20)
```

Predict structure

```
PE 1 GROUP1               (2)
PE 2 GROUP2               (4)
PE 3 GROUP3               (3)
   4 ELE       A  20
```

Natural structure

```
1 ELE   (A20/1:2,1:4,1:3)
```

Predict structure

```
PE 1 GROUP1               (2)
PE 2 GROUP2               (4)
MU 3 ELE       A  20      (3)
```

Mapping of COBOL Data Structures

In Predict it is not possible to define new field attributes together with a redefinition (which is possible in COBOL).

An example of a COBOL structure using this feature and the Predict definition that is used to circumscribe the structure is shown below.

COBOL structure

```
01 FIELD-A   PIC X(A20).
01 FIELD-A-RED REDEFINES FIELD-A PIX X(1) OCCURS 10.
```

Predict structure

```
   1  FIELD-A        A 20.
RE 1  FIELD-A
MU 2  FIELD-A-RED    A 1   (10)
```

In COBOL it is not possible to redefine a PE or a MU field. An additional group field has to be inserted. Predict does this automatically when COBOL Copy Code is generated from a field in which a PE or a MU field is redefined.

An example of a Predict structure and the COBOL structure that circumscribes it is shown below.

Predict structure

```
MU 1  FIELD-MU       A 20  (10)
RE 1  FIELD-MU
   2  FIELD-MU-RED   A 200
```

COBOL structure

```
05 R-FIELD-MU
    10 FIELD-MU  PIC X(A20) OCCURS 10.
05 R-FIELD-MU-REGR REDEFINES R-FIELD-MU.
    10 FIELD-MU-RED PIX X(200).
```

Example

The following example defines the structure of a sequential file.

```
Typ  Lev  Field-name      F  Len  Occ  ADA
---  ---  --------------- -- ---  ---  ---
 *
 PE   1   PE-GROUP1                 2   AA
 PE   2   PE-GROUP2                 3   AB
 PE   3   PE-GROUP3                 2   AC
      4   PE-EL1          A   6         AD
      4   PE-EL2          P   5.2       AE
 RE   1   PE-GROUP1
      2   PE-ELE-COMP     A   120
 *
 MU   1   MU-FIELD        A   250   5   AF
 RE   1   MU-FIELD
 PE   2   PE-GR1                    5
 MU   3   MU-FIELD1       A   5     4
 RE   3   MU-FIELD1
      4   MU-FIELD1-1     A   15
      4   MU-FIELD1-2     A   5
      3   FLD01           A   10
 RE   3   FLD01
      4   FLD01-1         A   5
      4   FLD01-2         A   5
 RE   3   FLD01
 MU   4   FLD01-3         A   1     10
      3   FILLER          A   2
      3   FLD02           A   5
      2   FLD03           A   20
 GR   1   GROUP                         AG
      2   GR-ELE1         A   30        AH
      2   GR-ELE2         A   20        AI
 RE   1   GROUP
 PE   2   GR-PE                     50
      3   GR-PE-EL        A   1
```

Browse Through Fields of a File - Code B

The Browse through Fields of a File function invokes the Modify Field function for each field in the field list of a file. If a field is specified in the parameter Field ID, the functions starts with this field.

The function is useful when applying general changes to all fields in a file.

Command: `BROWSE ELEMENT`

Edit Field Expression - Code Y

Depending on the editor preferences specified in the Profile > Handling screen, either the Software AG Editor or the Natural-based Subquery Editor is called. See ***Derived Field Expression***.

Command: `EDIT ELEMENT EXPRESSION`

19 Field Retrieval

- Field-Specific Retrieval Parameters 206
- Sorting Fields and Files 207
- Field-specific Retrieval Functions 209
- Layout of Field Lists 210
- Output Options for Fields 212
- Output Options for Fields - Continued 213

Field retrieval functions are called from the Field Retrieval menu, which is called with the command `RETRIEVE ELEMENT` or with Code R and object type code EL in a Predict Main Menu.

Standard retrieval types are described in the section *Retrieval* in the *Predict Reference* documentation.

Field-Specific Retrieval Parameters

See also *Selection Criteria* and *Output Options* in the section *Retrieval* in the *Predict Reference* documentation.

Parameters for Selection	
Field ID/Synonym	When retrieving information on fields, the identifiers of fields and language-specific synonyms can be used as selection criteria.
Synonym of language	Determines how Field ID/Synonym is used to select fields: none: Field ID/Synonym applies to field IDs. #: All: Field ID/Synonym applies to field IDs and to field name synonyms for all languages. *language*: If any language is specified, Field ID/Synonym applies to field IDs and to field name synonyms of this language.
Belongs to FI	ID of the file to which a field object belongs.
Files of type	Only fields contained in files of the specified type will be included in the selection. The value specified is stored in the global variables applying only to fields. See also *Specifying Parameter Values* in the section *Predict User Interface* in the *Introduction to Predict* documentation.

Field-Specific Output Options		
3GL specification	Y	The following 3GL-specific attributes of fields are displayed: Gr.structur, Justify, Synchronized, Init. value, Indexed by, Depending on, Condition name and Condition value.
Composed fields	Y	The source fields of hyper/super/subfields are displayed when fields of these types are displayed.
Display length		The format in which the length of fields is displayed. N: Natural Format P: Physical Format
DV-Field expression	Y	Derived field expressions are displayed.
Natural options	Y	Up to three headers displayed in Natural maps and the definition of the Natural edit mask are displayed.

Field-Specific Output Options		
Sorted by field	Used to determine how field and file lists are sorted:	
	N	Sort fields alphabetically by file ID. All fields are displayed in the order they are defined in the file.
	Y	Sort fields alphabetically by field ID.
	Note that the sort order also depends on the selection criteria. See *Sorting Fields and Files* below for more information.	
Synonyms	Synonyms of field names for specific languages are displayed. A language can be selected from a selection window.	

Sorting Fields and Files

Field and file lists produced by retrieval operations can be sorted by field ID or by file ID.

Sorting by Field ID

If fields and files are sorted by field ID, fields that are used in different files are sorted alphabetically by field.

```
13:25:45            *****  P R E D I C T  *****             2007-05-31
                                - List Field -                      Page:    3

Cnt   Ty L Fieldname                        F    Length D File ID

   37 GR 1 A-BINARY-GROUP                                  TSH-C-FILE
   38    1 A-CITY                           A     20.0    * MISCELLANEOUS
   39    1 A-CITY                           A     20.0  D TNG-ADABAS-FILE1
   40    1 A-CITY                           A     20.0    TSH-C-FILE
   41    1 A-DATE                           D             TNG-ADABAS-FILE1
    .       .                               .      .           .
    .       .                               .      .           .
```

Field lists will be sorted by field if parameters are specified in one of the following combinations:

Sorted by Field	Field ID specified	File ID specified
Y		
Y	Y	
Y	Y	Y
N	Y	

Note: If only a field id is specified as selection criteria, field and file lists are sorted by field, even if sorted by field is set to N.

Sorting by File ID

If fields are sorted by file, the fields appear in the order they are defined in the file.

```
13:29:12            *****  P R E D I C T  *****             2007-05-31
                                - List Field -                     Page:    1

Cnt   Ty L Fieldname                        F    Length D File ID

    1    1 AA-FIELD                         A     12.0  D * A-ADDR-FILE
    2    1 AB-FIELD                         A      1.0  D * A-ADDR-FILE
    3 MU 1 AC-FIELD                         A     20.0  D * A-ADDR-FILE
    4    1 AD-FIELD                         A     60.0  D * A-ADDR-FILE
    5    1 AE-FIELD                         A     60.0    * A-ADDR-FILE
```

Field lists will be sorted by file if parameters are specified in one of the following combinations:

Sorted by Field	Field ID specified	File ID specified
N		
N		Y
N	Y	Y
Y		Y

Note: If only a file ID is specified as a selection criterion, field and file lists are sorted by file even if sorted by field is set to Y.

Field-specific Retrieval Functions

- Fields and Related Views - Code R
- Non-Standard Fields - Code N
- Fields Related to a Z-File - Code Z

The following field-specific retrieval functions no longer exist. Alternatives are shown below:

- **Implode Fields**
  Use standard retrieval function Execute retrieval models with model IM (Predict standard implode) and output mode T instead
- **Fields with Verification**
  Use standard retrieval function Fields with children with association VE ("Is verified by VE") instead
- **Cross Reference Fields**
  Use standard retrieval function Execute retrieval models with model XR (Predict standard cross reference) and output mode X instead
- **Fields with no Verification**
  Use standard retrieval function Fields with no child with association VE ("Is verified by VE") instead

Standard retrieval functions are described in the section *Retrieval* in the *Predict Reference* documentation.

Fields and Related Views - Code R

Reports on fields and the related fields in related files. Related file means a master file and its userview. The relationship between fields is established as described below depending on whether the view is derived from a single master file or from several master files.

- **Single-Master Views**
  Views and userviews derived from a *single* master file, for example an Adabas file and its userview, are related by field short name (see ***Field Short Name***).
- **Multiple-Master Views**
  For views which can be derived from *several* master files (SQL tables and views), the coupling is established by parameters from Table/View ID and from Field ID in the field List of the file documenting the view.

Command: `RELATED ELEMENT`

See *Rippling* in the section *File* for more information on related fields and files.

Non-Standard Fields - Code N

Lists fields which are not derived from standard files, and also fields which were derived from standard files but subsequently changed to non-Standard fields.

Command: `NONSTANDARD ELEMENT`

Fields Related to a Z-File - Code Z

Reports on fields which are derived from standard files.

Command: `STANDARD ELEMENT`

Layout of Field Lists

Three different list formats are used for displaying information on fields:

- when fields are listed without entering a specific file (format 1 below)
- when fields used in a specific file are listed (format 2 - non SQL file)
- when fields used in a specific file are listed (format 3 - SQL file)

Format 1

The first list format applies when fields of several files are listed.

```
08:58:18             *****  P R E D I C T  *****             2007-05-31
                                - List Field -                       Page:    1

Cnt   Ty L Field ID                          F    Length D File

    1    1 FH-001                            A     50.0    PD-ADA-LONG
    2    1 FH-002                            A     50.0    PD-ADA-LONG
    3    1 FH-003                            A     50.0    PD-ADA-LONG
```

Meaning of Columns	
Ty	Type of field. See *Field Type* for a complete list of Field types and codes. RE indicates a redefinition.
L	The field level. Level number of the field. See *Level Number*.
Field ID	ID of the field object.
F	The field format. See *Field Format*.
Length	The field length. See *Field Length*.

Meaning of Columns	
D	Descriptor type. See *Descriptor Type*.
File	ID of the file containing the field.

Format 2

The second list format is used when fields used in a specific non SQL file are listed.

```
13:44:34              *****  P R E D I C T  *****              2007-05-31
                                - List Field -

File ID ............ ARH-A1
Type ............... Adabas file
Fnr ................ 12
-------------------------------------------------------------------------------
Cnt   Ty L Field ID                          F    Length D U DB S   Occ

    1    1 FIELD1                            A      5.0      AA N
    2    1 FIELD2                            A      3.0      AB N
    3    1 FIELD3                            A      4.0      AC N
    4    1 FIELD4                            A      6.0      AD N
    5 RE 1 FIELD4
    6    2 FIELD4-RE1                        A      3.0
    7    2 FIELD4-RE2                        A      3.0
```

Meaning of Columns	
U	Unique option. U is displayed if the field is a unique descriptor.
DB	Field short name. See *Field Short Name*.
S	Suppression / Null Value option. See *Suppression / Null Value option*.
Occ	Number of occurrences for multiple fields. See *Maximum Number of Values / Occurrences*.

Format 3

The third list format is used when fields used in a specific SQL file are listed.

```
13:44:34              *****  P R E D I C T  *****              2007-05-31
                                - List Field -

File ID ............ HEB-DB2
Type ............... DB2 table
Fnr ................ 13
-------------------------------------------------------------------------------
Cnt   Ty L Field ID                          F Cs  Length D U DB N Df  Occ

    1    1 HEB-EL                            A       5.0      AU U
    2    1 HEB-EL1                           A       3.0    U AA R Y
    3    1 HEB-EL2                           A M     4.0      AC U
    4    1 HEB-EL3                           A       6.0      AD U
    5 RE 1 HEB-EL4
    6    2 HEB-EL-RE                         A       3.0
    7 SP 2 HEB-SUP6                          A            D   AC
```

Meaning of Columns	
Cs	Character set.
N	Null Value option.
Df	Null default option.

Output Options for Fields

Notes:

1. Unless output mode is S, the option Cover page is always valid.
2. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.
3. With option Mark Implementation the corresponding file ("Belongs to FI") is marked.
4. If "Descriptors only" is set to Y, only fields with descriptor option not set to ' ', 'N' or 'E' are shown.

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Association attributes			Y	Y	Y	Y			Y	Y	Y	Y				
Attributes	Y		Y				Y		Y				Y			
Composed fields	Y		Y				Y		Y				Y			

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Connecting Character				Y						Y						
Description	Y		Y	Y			Y		Y	Y			Y			
Descriptors only	Y	Y	Y		Y		Y	Y	Y		Y		Y		Y	
Display length		Y														
Display modifier	Y		Y				Y		Y				Y			
Dummy/Placeholder										Y		Y		Y		Y
DV-field expression	Y		Y				Y		Y				Y			
Extract				Y						Y				Y		
Keywords	Y		Y	Y			Y		Y	Y			Y			
Mark implementation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
No. abstract lines	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
Natural options	Y		Y				Y		Y				Y			
Owner	Y		Y	Y			Y		Y	Y			Y			
With users	Y		Y	Y			Y		Y	Y			Y			
Show implementation																
Sorted by field	Y	Y	Y		Y		Y	Y	Y		Y		Y		Y	
Synonyms	Y	Y	Y		Y		Y	Y	Y		Y		Y		Y	
Use Con-form	Y		Y	Y			Y		Y	Y			Y			
User exit	Y		Y				Y		Y				Y			
3GL specification	Y		Y				Y		Y				Y			

Output Options for Fields - Continued

Retrieval Type	U		E				N		R	Z	
Output Mode	D	L	T		X		D	L	L	D	L
Current/Related	c	c	c	r	c	r	c	c	c	c	c
Association attributes			Y	Y							
Attributes	Y			Y		Y	Y			Y	
Composed Fields	Y						Y		Y	Y	
Connecting character				Y		Y					

Retrieval Type	U		E				N		R	Z	
Output Mode	D	L	T		X		D	L	L	D	L
Current/Related	c	c	c	r	c	r	c	c	c	c	c
Description	Y					Y	Y			Y	
Descriptors only	Y	Y	Y		Y		Y	Y	Y	Y	Y
Display length		Y						Y	Y		Y
Display modifier	Y						Y			Y	
Dummy/Placeholder											
DV-Field expression	Y						Y			Y	
Extract				Y		Y					
Keywords	Y			Y		Y	Y			Y	
Mark implementation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
No. abstract lines	Y	Y		Y		Y	Y	Y	Y	Y	Y
Natural options	Y						Y			Y	
Owner	Y			Y		Y	Y			Y	
With users	Y						Y			Y	
Show implementation											
Sorted by Field	Y	Y	Y		Y		Y	Y	Y	Y	Y
Synonyms	Y	Y					Y	Y	Y	Y	Y
Use Con-form	Y					Y	Y			Y	
User exit	Y						Y			Y	
3GL specification	Y						Y			Y	

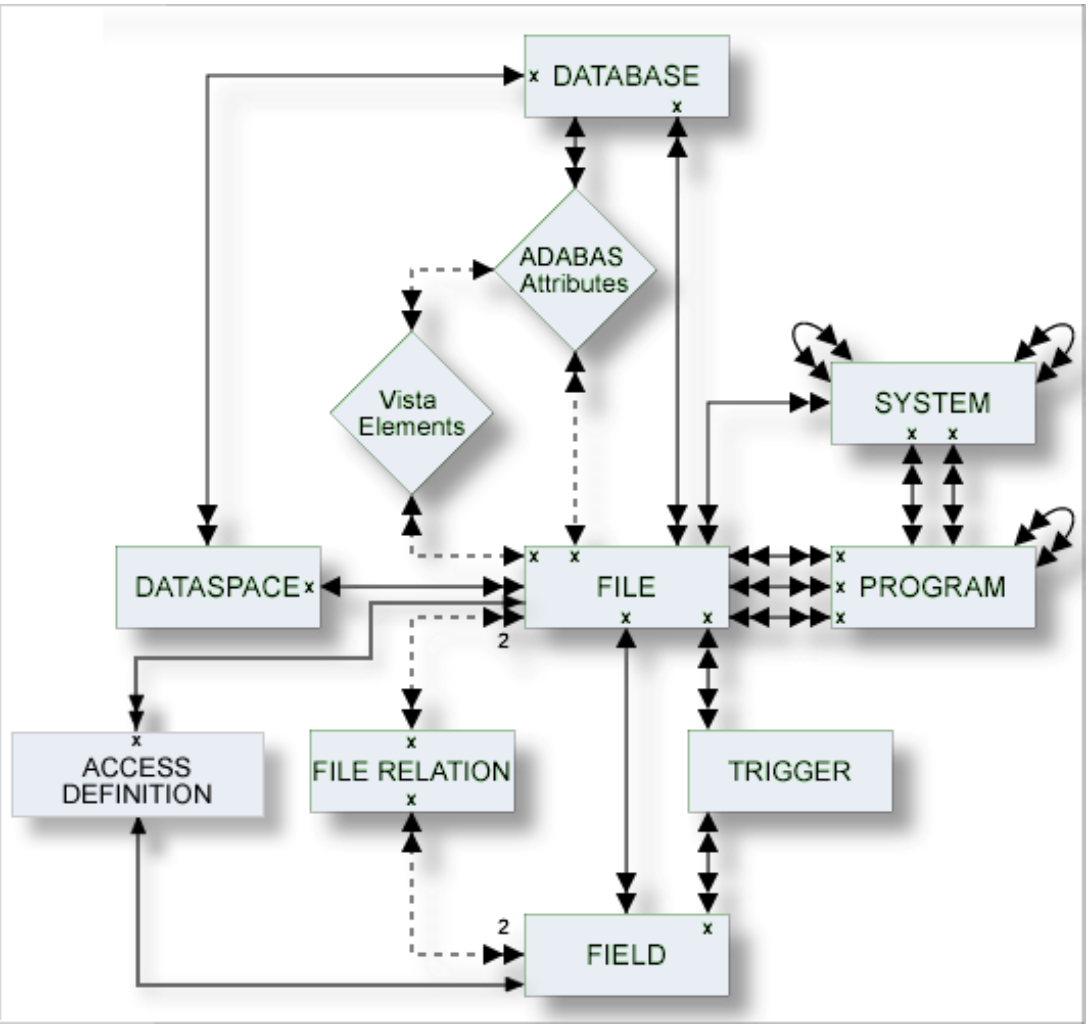
VI File

With Predict objects of type File, file structures can be defined for a wide variety of data storage systems and for use with different programming languages.

In the predefined Predict metastructure, a file can have passive and active associations of the following types:

Valid passive associations:	"Contained in DA" "Contained in DC" "Ref. by PR" "Input to PR" "Result of PR" "Is comp. of SY"
Valid active associations:	"Has Fields" "Has TR"

In addition, Adabas attributes and Vista elements can be defined for Adabas files with function "Attributes of Link" of mode "Contains FI".



Where to Find Detailed Information on Defining Distributed Data Structures

Basic information on attributes of files and how to execute file-specific functions is given in the sections below. If you wish to define data using simple files in a database not accessible via a network, you will find all the required information in the sections below. Additional information needed when defining complex data distribution structures using Adabas Vista or Entire Transaction can be found in the respective sections of the *Predict and Other Systems* documentation.

The description of object type File is organized under the following headings:

Maintaining Objects of Type File

Adabas Files, File Type A

File Types Conceptual, Standard and Other

SQL File Types

Adabas SQL Server

Adabas D

DB2
Informix
Ingres
Oracle
Sybase
General SQL File, File Type X
RDB
IMS
VSAM
ISAM
Entire System Server
File-Specific Maintenance
Rippling - Ensuring Consistent Data Definitions
File Retrieval

20 Maintaining Objects of Type File

- File Maintenance Menu 220
- Common File Attributes 222
- Defining Basic File Attributes 225

File Maintenance Menu

The File Maintenance menu is displayed with function code M and object code FI in a Predict Main Menu or with the command `MAINTAIN FILE`.

```
17:17:15             *****  P R E D I C T  *****              2007-05-31
Plan   0                 - (FI) File Maintenance -               Profile SYSTEM

 Function                                Function

 A  Add a file                           L  Link children
 C  Copy file                            S  Select file from a list
 M  Modify file                          B  Push backward
 N  Rename/Renumber file                 F  Force standard
 P  Purge file                           J  Physical Attributes
 D  Display file                         K  Modify Vista elements

Function .........
File ID ..........                                  Attributes ........*
Copy ID ..........                                  File of type ......*
Contained in DA ..                                  File nr ...........
External name ....

Restrictions ....*   Profile Default ,used          Association .......*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main   ↩
```

Parameters	
Function	All standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. The function Edit list of Fields does not appear on the menu but can be called with function code E. The functions Add a File and Modify File can be called with the editor line command .E when editing a file list (no matter which type of object the file list belongs to). See the section *Editors in Predict* in the *Predict Reference* documentation for more information. The following file maintenance functions are described later in the section ***Rippling***: ■ Purge file ■ Rename/Renumber file ■ Edit list of fields ■ Force standard ■ Push backward

Parameters	
	■ Modify Adabas attributes ■ Modify Vista elements ■ Edit subquery of a file
File ID	For the Select function: specifies a file ID to be used as a selection criterion, either alone or in combination with parameters Files of type and Contained in DA. If this field is left blank, all files which satisfy other selection restrictions specified are listed. See naming conventions for individual file types in the section ***Common File Attributes*** .
File of type	For the Select function: a file type can be specified as an additional selection criterion. For the Add and Copy functions: if file type is specified here, it will be passed to the Add/Copy File screen. Enter an asterisk to display a selection window with the file types valid for a particular function in your environment. See complete list of valid file types in the section ***File Type***.
Copy ID	Identifies the target file ID for the functions Copy and Push backward. For function Push backward: the ID of a standard file (type Z).
Contained in DA	For the Select function: a database ID can be specified as an additional selection criterion. Asterisk notation is possible. For the Add and Copy functions: the database ID can be specified here. This ID will be passed to the Add/Copy file screen. See list of valid database and file types in the section ***Contained in DA***.
External name	For the Select function: name of the file in another environment. Up to 50 characters can be specified here. Up to 250 characters can be specified with the Modify file function. If External name exceeds 50 characters, enter Y in the Zoom field.
File nr	For the Select function: A file number can be specified as an additional selection criterion. For the Add and Copy functions: The file number can be specified here. This number is passed to the Add a file or Copy file screen.
Restrictions	Additional criteria can be specified to restrict the scope of files to be processed. See *Restrictions* in the section *Predict User Interface* in the *Introduction to Predict* documentation.
Association	For function Link children: Objects of this type are to be linked to the file. Valid values: Field (default) or via user-defined association to any other object type.

Common File Attributes

The following attributes are applicable to all or most file types.

- File ID
- File Type
- Contained in DA
- File number
- Natural Construct Parameters

File ID

For naming conventions valid for all object types see *Naming Conventions*.

Special naming conventions apply to SQL file types. See overview in the section *Naming Conventions for SQL Objects*.

File Type

A file object has one of the following types. The file type must be compatible with the database in which it is contained. See table in the section *Contained in DA*.

File Type	Description
A	Adabas File
AT	Adabas Cluster Table
B	Adabas SQL view
C	Conceptual File
D	DB2 table
E	DB2 view
F	rdb file
I	IMS segment
J	IMS segment layout
K	IMS userview
L	Logical VSAM file
M	ISAM file
O	Other file
P	Entire System Server file
Q	Entire System Server userview
R	Logical VSAM view
S	Sequential file

File Type	Description
T	RMS file
U	Adabas userview
V	VSAM file (physical)
W	VSAM userview
X	General SQL file
Z	Standard file
1	LEASY
2	ISAM BS 2000
OT	Oracle table
OV	Oracle view
BT	Adabas D table
BV	Adabas D view
JT	Ingres table
JV	Ingres view
YT	Sybase table
YV	Sybase view
XT	Informix table
XV	Informix view
IT	Intermediate table
IV	Intermediate view
MT	DB2 query table

Contained in DA

The ID of the database containing the file. The database type must be compatible with the file type.

To generate a DDM for a file, the file must be linked to a compatible database (not of type C).

File Type		Compatible Database Type	
A	Adabas File	A	Adabas Database
A(SQL)	Adabas File with SQL usage set to Y	Q	Adabas SQL Handler
AT	Adabas Cluster Table		
B	Adabas SQL view		
BT	Adabas D table	B	Adabas D Handler
BV	Adabas D view		
D, MT	DB2 table, DB2 query table	D	DB2 Database
F	rdb File	R	RDB Handler

File Type		Compatible Database Type	
I	IMS segment	I	IMS Database
JT	Ingres table	J	Ingres Handler
JV	Ingres view		
L	Logical VSAM File	V	VSAM Handler
OT	Oracle table	O	Oracle Handler
OV	Oracle view		
P	Entire System Server File	P	Entire System Server
T	RMS File	M	RMS Handler
V	Physical VSAM File	V	VSAM Handler
X	General SQL File	E	General SQL Handler
XT	Informix table	X	Informix Handler
XV	Informix view		
YT	Sybase table	Y	Sybase Handler
YV	Sybase view		
1	LEASY	H	Other Handler
2	ISAM BS2000		
All File Types		C	Conceptual

File number

The number of the file. The possible value depends on the file type:

File Type	File Number
AT, J, K, Q, R, U	File number is taken from the specified master file
B, D, E, I, X, BT, BV, IT, IV, JT, JV, OT, OV, XT, XV, YT, YV, MT	not applicable
A, V, L, T, P, 1, 2	1 - 32767
Other file types	1 - 99999

Note: The file number can only be changed with the function Rename/Renumber File.

Natural Construct Parameters

The following parameters are only relevant if you are using Natural Construct. They appear in every Add, Copy or Modify file screen.

Parameter	Description	
Literal name	String to be used by Natural Construct in messages issued to confirm (un)successful access of a file via a DDM generated from the Predict file object.	
Average count	The average number of records contained in the file.	
Stability	Indicates how permanent the data contained in the file is.	
	F	Fixed. The file contains information which will always be valid, for example days of the week.
	S	Stable. The file contains information which does not change very often, for example file EMPLOYEES.
	V	Volatile. The file contains information which is constantly being updated, for example an invoice file.
	blank	Not specified (default value).

Defining Basic File Attributes

The following screen is displayed for the Add a File and Copy File functions for all file types:

```
13:05:04             *****  P R E D I C T  *****             2007-05-31
                                - Add a file -
File ID ......... HNO-NEW3




                File type ................* C  Conceptual file
                Master file ..............*
                File number ..............*
                Logical distribution type .*
                Contained in DA ..........*
```

General Parameters	
File type	The file type. Enter an asterisk for list of possible values or see list in the section *File Type*.
Master file	For the file types listed below, enter the ID of the related file. The type of related file is given below:

File Type	Type of Master File	
AT	A	Adabas File
J and K	I	IMS segment
L and W	V	Physical VSAM File
Q	P	Entire System Server File
R	L	Logical VSAM File
U	A	Adabas File
The master file can be selected using asterisk notation.		

File number	See the table of possible values in the section *File number*.
Logical distribution type	How the logical file is to be stored:

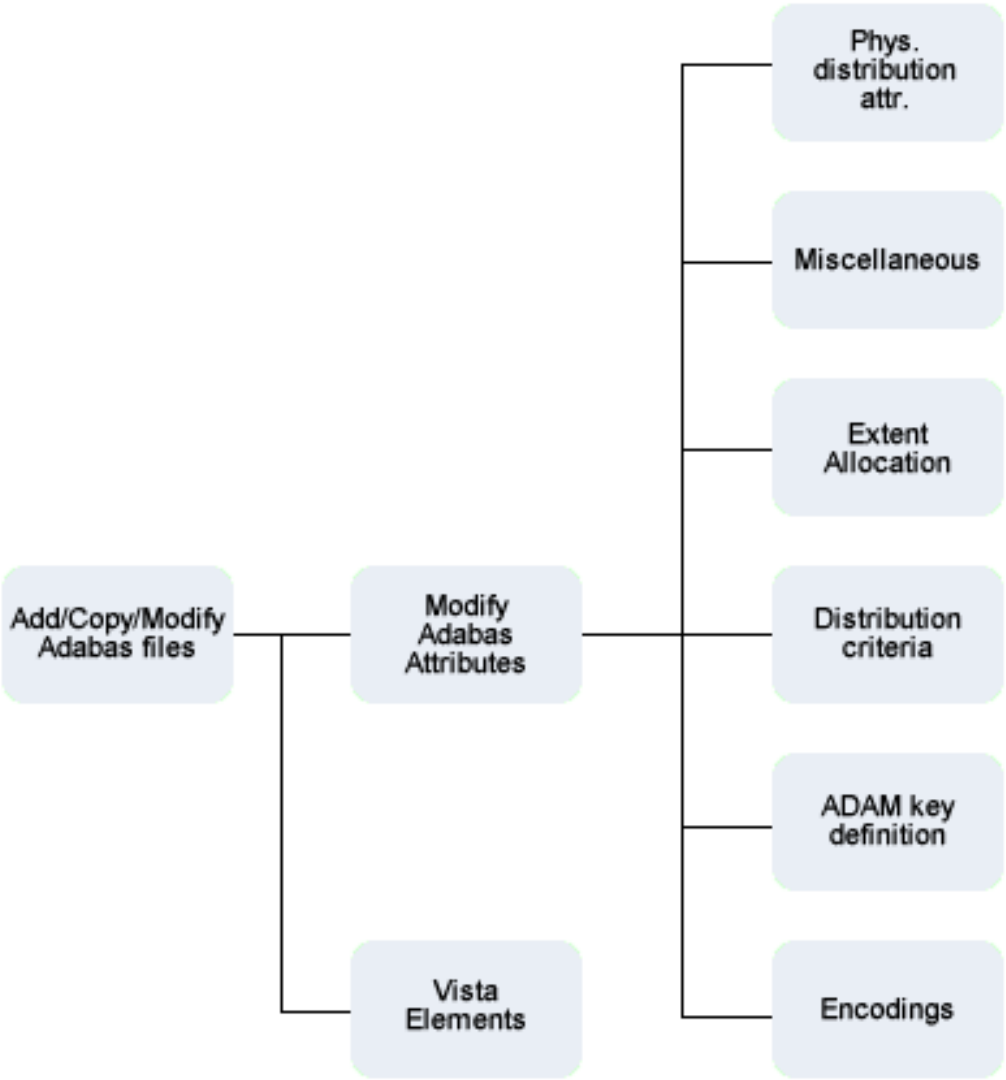
E	Expanded
P	Partitioned
N	Propagator file. Not applicable when defining data distribution for Adabas Vista.
blank	Simple file (default).
Note: This parameter is only applicable to files of type Adabas. For files of other types, this parameter must be blank.	

21 Adabas Files, File Type A

- Add/Copy/Modify a File Screen 229
- Modifying Adabas Attributes 230
- Phys. distribution attr. 233
- Miscellaneous Attributes and Adabas Security Definition 234
- Extent Allocation - Size Specifications For More Than One Extent 235
- Specifying Restrictions on Input Data - Distribution Criteria 236
- Modifying ADAM Descriptor Definition 238
- Modifying Vista elements 239
- Encodings 239

Adabas files are defined in several subsequent input screens. Screens on lower levels are called by specifying Y in one of the fields Attributes or Additional attributes in the bottom line of the higher-level screens.

The following diagram gives an overview of the input screens and the sections where these screens are described.



Add/Copy/Modify a File Screen

```
09:28:21            *****  P R E D I C T  *****              2007-05-31
                                - Add a file -
File ID ......... HNO-A-FILE
Type ............ Adabas,Simple file
File number ..... 123
Contained in DA .
Keys ..                                                          Zoom: N

Literal name ............
Average count ...........
Stability ..............*   Not specified
Sequence field .........*
Vista Access DBnr ......*
Vista Access Fnr ........
Adabas SQL usage ........ N (Y/N)
Abstract     Zoom: N









  Additional attributes ..* N  * Database link attr. .. N    Associations ..* N
```

Note: Parameters common to all object types are described under *Global Attributes*. For parameters common to all file types, see *Common File Attributes*.

Parameters	
Sequence field	The descriptor to be used by Natural for logical sequential reading. Determines the sequence in which records are delivered by the READ LOGICAL statement. The GENERATE DDM function will use this field as the default READ LOGICAL field in the Natural data definition module.
Vista Access DBnr, Vista Access Fnr	The L-DBnr and L-Fnr are used as database and file number for function Generate DDM if the parameter Use Vista access-nr is set to Y or T in the Generate DDM menu. Valid values are 0 to 65535 for DBnr and Fnr. No check for uniqueness is performed. **Note:** This parameter should not be confused with the Vista parameter Vista number, which is used to identify a file uniquely within a network. See *Including the Definition in the Vista Table* in the section *Adabas Vista* in the *Predict and Other Systems* documentation

Parameters		
Adabas SQL usage	Y	File is accessible via Adabas SQL Server.
	Note: When you add a file, this parameter can be specified in the Add a file screen. To change the value of this attribute for a file that already exists, use the Rename/renumber file function (see *Rename File*).	
Additional Options		
Additional attributes	Y	Two types of additional attributes can be specified: ■ Adabas attributes ■ Vista elements.
	The screens for entering Adabas attributes are described in the sections below.	

Modifying Adabas Attributes

There are different ways of calling the initial Modify Adabas attributes screen:

- specify Y in the field Additional attributes and mark Adabas attributes in the selection window
- select function Modify Adabas Attributes (code J) in the File Maintenance menu
- enter command `.A` in the file editor of a database object
- enter command `MODIFY ADA-ATTR`.

Note: If you do not call this screen, the default values set in the Default Adabas Attributes screen are taken. See the *Predict Administration* documentation. The Default Adabas Attributes screen is displayed with code A in the General Defaults menu.

```
16:35:58             *****  P R E D I C T  *****                    2007-05-31
                        - Modify Adabas attributes -
File ID ......... HNO-ADA-LOB                          Added 2006-04-27 at 15:27
Type ............ Adabas, Partitioned                    by HNO
Contained in DA .

Required attributes                        Physical distribution type
  Phys. file number ..* 54                   Simple file
  Min ISN ............. 1
  Max ISN ............. 667

      Device       Cylinder   Blocks      Padding factor  Max 2. alloc
      *-----       ---------- ----------  --------------  ------------
ASSO    3380   UI             1                 10
               NI             1
DATA    3380   DS             1                 10

Loading attributes                         Loading attributes
  Max recl. ...........                      One AC extent .......... N (Y,N)
  ISN reusage ......... N (Y,N)              DS reusage ............ Y (Y,N)
  User ISN ............ N (Y,N)              Mixed DS device ........ Y (Y,N)
  LOB file number ....*
* Additional attributes ..* S                                                ↩
```

Note: Up to six additional input screens can be called from this screen.

Parameters	
Required attributes	
Phys. file number	If a database is specified, the file number is taken as a physical file number automatically if this is possible. If not, a free physical number can be selected from a selection window.
LOB file number	An associated LOB file can be specified for a base file containing fields of type LO. Whenever the number of an associated LOB file is entered in the Adabas attributes of a base file, an own set of Adabas attributes is stored for SAG-ADA-LOB having the specified file number.
Physical distribution type	The distribution type of the physical file which describes how the logical file is stored. Read only field.
Min ISN	ADALOD LOAD parameter MINISN.
Max ISN	ADALOD LOAD parameter MAXISN.

Device and Size Specification for Adabas Files

```
     ....................................
     ....................................

        Device       Cylinder Blocks    Padding factor  Max 2. alloc
        *-----       -------- --------  --------------  ------------
ASSO     3380   UI                           10
                NI
DATA     3380   DS                           10
     ....................................
     ....................................
```

The device type and the size of the Upper Index (UI), Normal Index (NI) and Data Storage (DS) can be specified. If the size is specified in blocks, the equivalent size in cylinders provided by Predict is preceded by greater than (>) if the number of cylinders does not match exactly. If the size is specified in cylinders, Predict provides the equivalent size in blocks. The maximum secondary allocation in blocks can also be specified in each case.

Four characters specify the type of device used to store this part of the file. This device type must already be defined in the Predict database object containing this file. When this device type is changed in the database, the same change should be made in every file object contained in the database.

DATA padding factor	ADALOD LOAD parameter DATAPFAC.
ASSO padding factor	ADALOD LOAD parameter ASSOPFAC.
Device	The device type of the Upper Index (UI), Normal Index (NI) and Data Storage (DS). The device type for Data Storage is ADALOD LOAD parameter DSDEV.
Size (Cylinders/Blocks)	ADALOD LOAD parameters UISIZE (Upper Index), NISIZE (Normal Index) and DSSIZE (Data Storage).

Note: See also *Extent Allocation*.

Loading attributes	
Max recl.	ADALOD LOAD parameter MAXRECL.
ISN reusage	ADALOD LOAD parameter ISNREUSE.
User ISN	ADALOD LOAD parameter USERISN.
One AC extent	ADALOD LOAD parameter NOACEXTENSION.
DS reusage	ADALOD LOAD parameter DSREUSE.

Loading attributes		
Maximum secondary allocation	ADALOD LOAD parameters MAXUI (Upper Index), MAXNI (Normal Index) and MAXDS (Data Storage).	
Additional Options		
Additional Attributes	Y	Displays a window for specifying the following Adabas attributes: ■ Phys. distribution attr. ■ Miscellaneous attributes ■ ADAM key definition ■ Extent allocation ■ Distribution criteria ■ Encodings

Note: Phys. distribution attr. and Extent allocation only appear in this window if applicable.

Phys. distribution attr.

```
13:19:12             *****  P R E D I C T  *****             2007-05-31
                        - Modify Adabas attributes -
File ID ......... HNO-NEW                             Added 2007-05-31 at 13:19
Type ............ Adabas, Partitioned                   by HNO
Contained in DA . HEB-55-HEB-NW-V (PDBnr: 55)

Distribution attribute
  Phys. distribution type ..* P  Partitioned

Loading attributes
  Min ISN ............. 19
  Max ISN ............. 667
  One AC extent ....... N (Y,N)









* Additional attributes ..* N
```

Parameters	
Phys. distribution type	The types for the physical file are limited by the logical distribution type, as shown below:

Physical distribution Type		Logical distribution Type	
E	expanded	E	expanded
P	partitioned	P	partitioned
blank	simple File		any

Miscellaneous Attributes and Adabas Security Definition

```
13:33:18             *****  P R E D I C T  *****             2011-05-31
                        - Modify Adabas attributes -
File ID ......... XYZ-NEW                             Added 2011-05-31 at 13:19
Contained in DA .                                        by XYZ
PDBnr ...........       PFnr ... 123

Adabas Security definition
  Access level ...........    (0-15)
  Update level ...........    (0-15)

Loading attributes                      Record spanning
  Ciphered ............... N  (Y,N)      Spanned ............... Y  (Y,N)
  LOWNERID ............... 0  (0-8)      Max secondary ISN .... 1122
  Refresh from program ... N  (Y,N)      Secondary start RABN . 3344          ↩

  Automatic allocation ... Y  (Y,N)
  PLOG .................. Y  (Y,N)
  ISN size ..............* 0
  Erase ................. N  (Y,N)
  Index compression ...... N  (Y,N)
  No BT file ............ N  (Y,N)
  Max occ system fields .. 10 (1-20)
```

Parameters		
Access level	The Adabas access security level of the file.	
Update level	The Adabas update security level of the file.	
Ciphered	Y	The file is a ciphered file.
LOWNERID	Length of internal Owner ID of a multi-client file.	
Refresh from program	Adabas parameter PGMREFRESH. See the Adabas DBA documentation.	
Automatic allocation	Y	Adabas will automatically allocate and deallocate extents. See the *Adabas Reference* documentation.

Parameters		
PLOG	Y	Database runs with protection log. Linux only.
ISN Size	Length of ISN. Valid values: 0, 2, 3 and 4. For Adabas/Linux: 0, 2 and 4 are valid. For mainframes: 0, 3 and 4 are valid.	
Erase	Y	For Adabas/Linux. All index and data storage blocks are overwritten with zeroes when they are returned to the free space table.
Index compression	Y	Adabas reduces space requirements for the index and for data storage by removing redundant information on an individual descriptor basis.
No BT file	Y	Exclude file from BACKOUT TRANSACTION processing.
Record spanning	Spanned	When record spanning is enabled, the size of compressed records in a file may exceed the maximum data storage block size. Default is Y.
	Max secondary ISN	Defines the initial size of secondary ISNs.
	Secondary start RABN	If spanned records are used, a secondary address converter is used to map the secondary ISNs to the RABNs of the Data Storage blocks where the secondary records are stored.
Max occ system fields	This parameter specifies the maximum number of values generated for a system-generated multiple-value field.	

Extent Allocation - Size Specifications For More Than One Extent

More than one extent can be specified using the Extent allocation option in the Modify Adabas attributes selection window.

The size and first RABN (Start Rb) of the Address Converter (AC), Upper Index (UI), Normal Index (NI) and Data Storage (DS) can be specified for up to 99 extents. The total space allocated is displayed in the upper right corner of the screen.

```
  13:47:50               *****  P R E D I C T  *****               2007-05-31
                            - Modify Adabas attributes -
 File ID ......... HEB-FI-PART                      Modified 2007-05-31 at 09:24
  Contained in DA . HEB-DA-TRANS                              by FST
  PDBnr ........... 21    PFnr ... 1
                                                  +- Total -------------------+
  Extent *Device Start RABN Cylinder   Blocks         !    Cylinder     Blocks  ↩
!
   1  AC 3390                          1              ! AC              1       ↩
!
      UI 3390                                         ! UI                      ↩
!
      NI 3390                                         ! NI                      ↩
!
      DS 3390                                         ! DS                      ↩
!
                                                   +---------------------------+

   2  AC
      UI
      NI
      DS

   3  AC
      UI
      NI
      DS

  * Additional attributes ..* S                                Scroll to .... 1
```

Specifying Restrictions on Input Data - Distribution Criteria

Distribution criteria determine which data can be written to a file.

```
13:13:39             *****  P R E D I C T  *****             2007-05-31
                        - Modify Adabas attributes -
File ID ......... PD-AD1                          Modified 2007-05-31 at 10:08
Contained in DA . DEMO-DB                                 by PD
PDBnr ........... 180   PFnr ... 13

Ty Partitioning field               F  Cs Length   Occ   D U DB N NAT-l
-- *------------------------------ -- -- -------- ----- - - -- - -----



   1 Access ....*      Critical ..* (Y,N)   Shared Partition ..   (Y,N)
     Part. name .
     High value .                                                     Zoom: N

   2 Access ....*      Critical ..* (Y,N)   Shared Partition ..   (Y,N)
     Part. name .
     High value .                                                     Zoom: N

   3 Access ....*      Critical ..* (Y,N)   Shared Partition ..   (Y,N)
     Part. name .
     High value .                                                     Zoom: N


* Additional attributes ..* S                                Scroll to ....
```

Parameters	
Partitioning field	ID of the field used to distribute data into separate partitions. The field must exist in the file.
Access	Specifies the access level of a partition. F — Full. Read/write access is permitted. R — Read. Read-only access is permitted. N — No access.
Critical	Specifies whether a partition is considered critical or not. Y — Critical. N — Not critical.
Shared Partition	Specifies whether to allow partition sharing for minimal data movement or not. Y — Allowed. N — Not allowed.
Value	Value to be checked. If the value is longer than 50 characters, set Zoom to Y.
Scroll to	If more validation criteria are specified than can be displayed in one screen, the criteria to be displayed on top of the list can be specified in the field Scroll to.

Modifying ADAM Descriptor Definition

```
13:40:40             *****  P R E D I C T  *****             2007-05-31
                        - Modify Adabas attributes -
File ID ......... HNO-NEW                              Added 2007-05-31 at 13:34
Contained in DA .                                         by HNO
PDBnr ...........       PFnr ... 123



ADAM descriptor definition
  Field ID ........*
  Parm .............
  Overflow .........
```

Parameters	
ADAM descriptor definition	
Field ID	Fields to be used as ADAM descriptor. ADALOD LOAD parameter ADAMDE.
Parm	ADALOD LOAD parameter ADAMPARM.
Overflow	ADALOD LOAD parameter ADAMOFLOW.

Modifying Vista elements

```
13:51:47            *****  P R E D I C T  *****            2007-05-31
                          - Add Vista element -
File ID ......... JPE340                          Modified 2007-05-31 at 13:50
Type ............ Adabas, Partitioned                 by HEB

Network ........* HOME
Simple .......... Y (Y,N)          Partition ID assignment ..* V Vista
Vista                              Max number of partitions .. 255
 Environment ID .                  Enable Read-by-ISN ........ Y (Y,N)
 DBnr ........... 1                Part. file concurrency .... 8
 Fnr ............ 3                Store control option .....* 1 Reject
 Name ...........


    Database                         PDBnr  PFnr   Criterion
    *------------------------------  -----  -----  --------------------
  1








                                                        Scroll to:
```

Parameters

See the section *Including the Definition in the Vista Table* in the section *Adabas Vista* in the *Predict and Other Systems* documentation for a description of all parameters.

Encodings

Universal encoding support of an Adabas file can be defined in the screen shown below.

```
13:40:40             *****  P R E D I C T  *****              2007-05-31
                         - Modify Adabas attributes -
 File ID ......... HNO-NEW123                       Modified 2007-05-31 at 13:34
 Contained in DA . HNO-TEST                                 by HNO
 PDBnr ........... 12     PFnr ... 123



 Universal encoding support
   FACODE ..*      none
   FWCODE ..*      none
   FUWCODE .*      none









 * Additional attributes ..* S
```

Note: See the *Adabas Administration* documentation for further information on this topic.

22 File Types Conceptual, Standard and Other

```
10:36:36             *****  P R E D I C T  *****             2007-05-31
                                - Add a file -
File ID ......... FI
Type ............ Conceptual file
File number .....
Contained in DA .
Keys ..                                                          Zoom: N

Literal name ........
Average count .......
Stability ..........*   Not specified
Sequence field .....*
Abstract     Zoom: N
```

Note: Parameters in database and Sequence field do not apply to files of type Standard and Other.

Parameters	
File ID	The ID of the file object.
Type	The file type.
File number	Files of these types can have a file number from 0 - 99999.
Contained in DA	The ID of the database containing the file. See *Contained in DA*.
Sequence field	The function Generate DDM will use this field as the default READ LOGICAL field in the Natural data definition module. For conceptual files for documentation and later use.
Literal name, Average count, Stability	Only applicable if you are using Natural Construct. See *Natural Construct Parameters*.

23 SQL File Types

- Naming Conventions for SQL Objects 244
- Common Parameters for SQL File Types 245
- Field Lists of SQL Views 246
- Editing the Subquery of an SQL View 248

Predict offers various file types for documenting tables and views of the SQL systems listed below. The file objects which document the SQL tables and views can be used to generate SQL CREATE statements, DDMs and copy code members for 3GLs. The CREATE statements are stored as Natural members in file FDIC.

Naming Conventions for SQL Objects

Special naming conventions apply to the following objects in Predict

- SQL file types. See table below.
- Fields linked as children to these file types
- Constraint names
- Correlation names
- Tablespace for Oracle
- The file IDs must be fully qualified. A fully qualified ID consists of three parts:
  - Hyphen to separate creator/schema from table/view name
  - Table/view name. The maximum length depends on the SQL system. See table below.
- Fully qualified IDs may not exceed 32 characters.
- The permitted characters listed in the table below apply to creator/schema and table/view name.

	Filetype								
		AT,B, A(SQL)	BT, BV	D, E, IV, IT	JT, JV	OT, OV	X	XT, XV	YT, YV
Convention	Maximum length of table/view name	32	18	18	24	30	18	18	30
	Upper case			Y		Y	Y		
	Upper/lower case	Y	Y		Y			Y	Y
	'_' allowed at first pos.			Y	Y				Y
	'#' allowed at first pos.		Y	Y					
	'$' allowed at first pos.		Y	Y					
	'@' allowed at first pos.		Y	Y					
	'_' allowed from second pos.	Y	Y	Y	Y	Y	Y	Y	Y
	'#' allowed from second pos.		Y	Y	Y	Y	Y		Y
	'$' allowed from second pos.		Y	Y	Y	Y	Y		Y
	'@' allowed from sec. pos.		Y	Y	Y				Y
	Numbers allowed from second pos.	Y	Y	Y	Y	Y	Y	Y	Y

Type-specific rules are also given in the respective parts of this section.

Common Parameters for SQL File Types

The following parameters are valid for all or most SQL file types.

SQL Attributes

These parameters apply to all SQL views.

Select	A	Select all: Redundant duplicates are not eliminated.
	D	Select distinct: Redundant duplicates are eliminated.
With check option	Y	All inserts and updates to the view are checked against the view definition.

Additional attributes / Associations

Profile options are described in the section *Defaults* in the *Predict Administration* documentation. The editors are described in the section *Editors in Predict* in the *Predict Reference* documentation.

Subquery

This option is available for all SQL views.

Enter Y in the Subquery field to call an Editor to edit the subquery clause of the SQL view. The editor called depends on the preferences specified in the Profile > Handling screen:

- if your first choice editor is Natural, the Subquery Editor (a modified Natural Editor) is called.
- if your first choice editor is SAG or Word for Windows, the Software AG Editor is called.

Additional commands are available for processing subqueries and checks are performed when the subquery is cataloged.

See the section *Editors in Predict* in the *Predict Reference* documentation .

Check Expression

This option is available for the following SQL tables:

- Adabas D
- DB2
- Oracle
- Informix
- Ingres
- Sybase

It is also available for the following file type:

- General SQL file

Enter Y in the Check expression field to edit the check expression of the file. The editor called depends on the preferences specified in the Profile > Handling screen:

- if your first choice editor is Natural, the Description Editor (a modified Natural Editor) is called.
- if your first choice editor is SAG or Word for Windows, the Software AG Editor is called.

No special checks are performed when check expression is saved.

Field Lists of SQL Views

The following screen shows the layout of the field list of an SQL file.

```
>                      > + Fi: ARH-E1                            L:    1 S: 5
   Ty L Field ID           from Table/View ID          Field ID           All
   *- - ------------------ --------------------------- ------------------
      1 ARH1                ARH-D1                      ARH1
   SP 1 ARH_SP              ARH-D1                      ARH_SP
      1 ARH4                ARH-D1                      ARH4
      1 TIME_1              PD-E1                       TIME_1
```

Column	Meaning
Ty	Field type.
L	Field level.
Field ID	ID of field object documenting the SQL view. The ID of the field object in Predict documenting a field in a view can differ from the name of the field in the original table or view.
from Table/View ID	ID of the Predict file documenting the table or view from which the field was taken. If this file contains a subquery clause with a correlation name for the table or view, the correlation name must be entered instead of the file ID.
from field ID	Field in the table or view from which it was taken.

Adding new Fields to Field Lists of SQL Views

New fields can easily be inserted into the field list of an SQL view using one the following two methods:

Manually

Enter parameters Field ID, from Table/View ID and from Field ID described above. See the section *Naming Conventions for SQL Objects*.

With Command SELECTImport

Use the command SELECT to select fields from other SQL tables or views and insert them into the current field list. The following screen appears:

```
13:06:46             *****  P R E D I C T  *****             2007-05-31
Plan   2                  - Field Selection Menu -                 Profile HNO

File ID ............ HNO-XV
                                                        Added 2007-05-31 at 13:05
                                                           by HNO
Select object type ............... EL  ( Field )

Retrieval type ......* D
Output mode .........* S Select

Search criteria
  Field ID/Synonym ...                                     Synonym of language*
  Belongs to FI ......                                     Files of type .....*


Restrictions ........*   Profile HNO,used                 Association .......*
```

Note: Parameters not listed here are described in the section *Retrieval* in the *Predict Reference* documentation .

Parameters		
Retrieval type	The following field retrieval functions are available to select fields for insertion into the field list:	
	D	Fields
	N	Non-standard fields
	U	Fields with no verification.
Search criteria		
Field ID/Synonym	This parameter corresponds to from Field ID in the field list of SQL views shown above.	
Belongs to FI	This parameter corresponds to from Table/View ID in the field list of SQL views shown above.	
Files of type	It is possible to select fields of any type for insertion. However, an error message will be given when you try and catalog a field list containing fields with an incompatible type. See list of compatible field types in the section *Structure of a Subquery Clause*. If a unique field ID is specified, this parameter is ignored.	
Mark the fields to be inserted with X, S or /. Selected fields are marked *ins* in the field list. Catalog the list to add the fields to the list.		

Editing the Subquery of an SQL View

Calling the Editor

Two methods are available for calling an editor to edit the subquery of an SQL view:

1. enter Y in the Subquery field in the Additional attributes window of the Add File, Copy File or Modify File screen, or
2. enter the command `EDIT FILE SUBQUERY <File ID>`

The editor called depends on the preferences specified in the Profile > Handling screen:

- if your first choice editor is Natural, the Subquery Editor (a modified Natural Editor) is called.
- if your first choice editor is SAG or Word for Windows, the Software AG Editor is called.

```
>                        > + VI: PD-E1                              L: 1    S: 8
  All  ....+....1....+....2..Subquery clause .4....+....5....+....6....+....7..
       FROM
       PD-D1 D1 ,
       SMR-D
       WHERE
         'ABC' IN
         ( SELECT A-COL2 FROM PD-D1 A)
```

Structure of a Subquery Clause

The following rules apply:

- In the first part of the subquery clause, the related master files and their correlation names can be specified in SQL syntax.
- The file type of the related master files must be compatible with the file type:

File Type of View	Related Master File Type
B	A(SQL), AT, B
BV	BT, BV
E, IV	D, E, IV
JV	JT, JV
OV	OT, OV, IV
XV	XT, XV
YV	YT, YV

- Any correlation name that is specified must be used whenever the file is referred to. Type-dependent rules apply to the length of a correlation name and the characters permitted. See table in the section *Naming Conventions for SQL Objects*.
- The first part of the subquery is generated automatically if the fields of the file are defined in Predict before the subquery is edited.
- The second part of the subquery contains the selection criteria of the view: the WHERE clause, GROUP BY clause or HAVING clause or any combination of these. The name of each field referenced in the selection criteria must be qualified by the ID of the file from which the field is taken or - if a correlation name has been specified in the first part of the subquery - by the correlation name.
- If joined views are edited, the selection criteria and the type of join are displayed for each join. See example below.

```
>                    > + VI: LE-SEIGANZ                      L:    1 S:    8
All  ....+....1....+....2..Subquery clause .4....+....5....+....6....+....7..
      FROM
     BI-UTEREST U
      LEFT OUTER JOIN
     BI-LANGUET L
      ON U-CLANGUE = L-CLANGUE
      LEFT OUTER JOIN
     BI-ZONRSPT Z
      ON U-CRESEAU = Z-CRESEAU AND U-NZONRSP = Z-NZONRSP
```

- If union views are edited, placeholders are used instead of field selection lists, because field selection lists are maintained with the field list editor. The placeholder indicates where the selected field list will be added when generating the view. Placeholders are displayed in angle brackets. See example below.

```
>                    > + VI: LE-UTERESV1                     L:    1 S:   15
All  ....+....1....+....2..Subquery clause .4....+....5....+....6....+....7.. ↩

      <SELECTION LIST 1 >
      FROM
     BI-UTEREST U
     WHERE U-CLANGUE LIKE 'A%'
      UNION
      <SELECTION LIST 2 >
      FROM
     BI-ZONRSPT Z
     WHERE Z-CRESEAU LIKE 'B%'
      UNION
      <SELECTION LIST 3 >
      FROM
     BI-DISTRIT D
     WHERE D-NDISTRI LIKE 'D%'
```

- When generating a CREATE VIEW statement for a view, hyphens (-) are replaced by underscores (_) or points (.).
- The subquery can include comment lines (with /*, * or ** in the first two columns) and line comments (preceded by /*).

24 Adabas SQL Server

- Overview 252
- Naming Conventions 252
- Adabas Cluster Table 254
- Adabas SQL View 255

Overview

There are two methods of documenting Adabas tables:

- **Files of Type A (SQL)**
  If an Adabas table corresponds *exactly* to a base table in Adabas SQL Server, it can be documented as a file of type A (SQL). The Adabas file must not contain groups structures or multiple value fields. Rotated fields are not supported with this method. This method is retained for reasons of compatibility with earlier Predict versions.
- **Files of Type AT**
  Tables can also be documented with files of type AT (Adabas cluster table). Files of this type can be understood as userviews to an Adabas file. See ***Adabas Cluster Table***.

Adabas SQL *views* are documented with files of type B. See ***Adabas SQL View***.

Naming Conventions

The following naming conventions apply to files documenting Adabas SQL Server tables and views (files of type AT, B).

Upper / lower case

If the Predict parameter General Defaults > Miscellaneous > Upper/lower case / Object ID is set to L, the following attributes are stored in upper and lower case as entered:

- File ID (object IDs containing lower case letters are not recommended)
- Derived field expressions
- SQL verifications
- Check expressions
- Constraint names

See also section *Defaults* in the *Predict Administration* documentation.

Length

Table/View names for Adabas SQL Server objects can have up to 32 characters.

Permitted characters

See overview of permitted characters in the section *Naming Conventions*.

Qualifier

The identifier of a table or view must be given in qualified form: the schema identifier, a delimiter and the table/view name. A hyphen is used as a delimiter (not a period as in SQL). An example: SYSSAG-SYSCOLUMNS. Hyphens in names are treated as follows:

- When a table/view is generated from a Predict file object, the hyphen is transformed into a period (.).
- Because hyphens are used as delimiters, only one hyphen can occur in the SQL identifier. Column names must not contain a hyphen.
- The hyphen can be used as a minus sign or negative sign in the field expression or the subselect clause and must then be preceded by a blank.

Adabas Cluster Table

```
13:25:05            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-AT
Type ............ Adabas cluster table
File number ..... 1234  Master file: HNO-A
Contained in DA .
Keys ..                                                          Zoom: N

Literal name ............
Average count ...........
Stability ..............*   Not specified
Vista access L-DBnr ....*
Vista access L-Fnr ......
Table level ............*

Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters		
File ID	See *Naming Conventions*.	
Contained in DA	ID of the database object containing the file.	
Table level	0	Only "flat" structures are permitted (no MU or PE fields).
	1	For defining multiple fields and periodic groups.
	2	For defining multiple fields within a periodic group.
	There are two methods of documenting periodic groups and multiple value fields in AT files:	

Parameters	
	■ If the occurrences of PE/MU fields are *fixed*, you can use rotated fields in the AT file. ■ If the occurrences of PE/MU fields are *variable*, use subtables (AT files at level 1 or level 2). For more information see the section *Adabas SQL Server* in the *Predict and Other Systems* documentation.

Adabas SQL View

```
13:24:04            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-FIB
Type ............ Adabas SQL view
Contained in DA .
Keys ..                                                          Zoom: N

Literal name .............
Average count ............
Stability ...............*   Not specified
SQL attributes
  Select ................* A Select all
  With check option .....* N No

Abstract     Zoom: N





  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under ***Global Attributes***. Parameters common to all file types, for example Literal name, are described under ***Common File Attributes***. See also ***Common Parameters for SQL File Types***.

Parameters	
File ID	See *Naming Conventions* .
Contained in DA	ID of the database object containing the file.

25 Adabas D

- Naming Conventions 258
- Adabas D Table, File Type BT 259
- Adabas D View, File Type BV 260

Adabas D tables and views can be documented in Predict with file objects of type BT and BV respectively. These file objects can be used to generate DDMs or CREATE TABLE/VIEW statements.

Naming Conventions

The following naming conventions apply to files documenting Adabas D tables and views.

Upper / lower case

If the Predict parameter General Defaults > Miscellaneous > Upper/lower case / Object ID is set to L, the following attributes of Adabas D objects are stored in upper and lower case as entered:

- File ID (object IDs containing lower case letters are not recommended)
- Derived field expressions
- SQL verifications
- Check expressions
- Constraint names

See also section *Defaults* in the *Predict Administration* documentation.

Length

- Table/View names for Adabas D objects can have up to 18 characters.
- A fully qualified ID (Creator + Hyphen + Table/View name) may not exceed 27 characters.

Permitted characters

See overview of permitted characters in the section ***Naming Conventions***.

Adabas D Table, File Type BT

```
13:49:52            *****  P R E D I C T  *****            2007-05-31
                             - Modify file -
File ID ......... HNO-BT                          Modified 2007-05-31 at 13:23
Type ............ Adabas D table                        by HNO
Contained in DA .
Keys ..                                                                Zoom: N

Literal name ...........
Average count ..........
Stability .............*   Not specified
Check constraint name ..
Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	See *Naming Conventions* for Adabas D objects.
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Adabas D table, the file must be linked to a database of type Adabas D handler.

Check constraint name	If a table check expression has been defined and the name of a check constraint is entered here, the following clause is generated in the `CREATE TABLE` statement:
	
	`CONSTRAINT constraint_name`
	`CHECK (check_expression)`

Adabas D View, File Type BV

```
13:36:40            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-BV
Type ............ Adabas D view
Contained in DA .
Keys ..                                                          Zoom: N

Literal name .............
Average count ............
Stability ...............*   Not specified
SQL attributes
  Select ................* A
  With check option ...... N (Y/N)

Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	See *Naming Conventions* for Adabas D objects.
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Adabas D view, the file must be linked to a database of type Adabas D handler.

26 DB2

- Naming Conventions 262
- DB2 Table, File Type D 263
- DB2 View, File Type E 266
- Intermediate View, File Type IV 267
- Intermediate Table, File Type IT 268
- DB2 Query Table, File Type MT 269

DB2 tables and views can be documented in Predict with file objects of type D and E respectively. These file objects can be used to generate DDMs or `CREATE TABLE/VIEW` statements.

Naming Conventions

The following naming conventions apply to files documenting DB2 tables and views.

Upper / lower case

File IDs must be entered in upper case. If the Predict parameter General Defaults > Miscellaneous > Upper/lower case / Object ID is set to L, lower-case IDs are not converted to upper case and an error message is given.

Hyphens

- A hyphen is used to delimit the creator from the table/view name.
- Only one hyphen is permitted in the ID of a DB2 table/view object.
- When a table or view is generated from the Predict file object, the hyphen is converted to a period.

Length

- Table/View names for DB2 objects can have up to 18 characters.
- A fully qualified ID (Creator + Hyphen + Table/View name) must not exceed 27 characters.

Permitted characters

See overview of permitted characters in the section *Naming Conventions*.

DB2 Table, File Type D

```
15:45:09                *****  P R E D I C T  *****                  2015-05-05
                               - Modify file -
File ID ......... XYZ-ARCHIVE_TAB
Type ............ DB2 table
Contained in DA . DAEIDB2B
Keys ..                                                                Zoom: N

Literal name ............
Average count ...........                  Stability .......*   Not specified
Check constraint name ...
History/Archive table ..*
        usage as .......*    (none)
Physical attributes in <Default Server>
  Number of partitions ..                  Restrict on drop . N (Y/N)
  Edit program ..........                  Part. size (GB) ..
  Row attributes ........   (Y/N)         CCSID ...........*   (none)
  Validation program ....                  Temporary ........ N (Y/N)
  Audit ................* N Audit none     Volatile ......... N (Y/N)
  OBid ..................                  Append ........... N (Y/N)
  Data capture .......... N (Y/N)          Hash size (KB) ...
  Compress ..............   (Y/N)         Logged ...........   (Y/N)
Abstract     Zoom: N
  Additional attributes ..* N        * Associations ..* S                    ↩
```

Additional attributes are available for files of type D that are used as accelerator tables.

```
18:17:10                ******  P R E D I C T  *****                  2020-07-02
                                - Modify file -
File ID ......... XYZ-DB_V12                       Modified 2020-04-06 at 13:35
                                                         by XYZ

DBMS extensions in <Default Server>
  Keylabel name .... on default                                        Zoom: N
  Accelerator name . ACC IN D



* Additional attributes ..* S        * Associations ..* S                    ↩
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters		
File ID	ID of the Predict object documenting the DB2 table. See *Naming Conventions*.	
Contained in DA	ID of the database object containing the file.	
DB2 Attributes		
Number of partitions	The number of partitions of the table.	
Edit program	The name of an edit routine for the table.	
Row attributes	Specifies whether the edit procedure parameter list contains an address for the description of a row.	
	Y	Yes. This is the default.
	N	No.
Validation program	The name of a validation routine for the table.	
Audit	The type of access to this table that will cause auditing to be performed. Valid values:	
	A	All
	C	Changes
	N	None

Parameters	
OBid	Identifies the OBID to be used for the table. An OBID is the identifier for an object's internal descriptor in DB2. **Note:** This parameter is required if parameter DB2 ROSHARE parm of the database object containing the table is set to R. See ***Database Type D - DB2***. See your *DB2* documentation for more information.
Data capture	Y | Data changes are passed to a user exit.
Compress	Specifies whether data compression applies to the rows of the implicitly created tablespace. blank: Not specified. This is the default setting. Y: Yes. N: No.
Restrict on drop	Y | The DB2 table cannot be dropped. To drop a table with this setting, this parameter must be set explicitly to N.
Part. size	Specifies that the table is to be partitioned by growth, every *n* GBytes. Where *n* is to be replaced by the desired integer value.
Hash size	Specifies the amount of fixed hash space to preallocate for the partition that is associated. Hash size is *n* KBytes. Where *n* is to be replaced by the desired integer value.
Logged	Specifies whether changes that are made to the data in the implicitly created tablespace are recorded in the log. blank: Not specified. This is the default setting. Y: Yes. N: No.

Check constraint name	If a table check expression has been defined and the name of a check constraint is entered here, the following clause is generated in the `CREATE TABLE` statement:
	
	`CONSTRAINT constraint_name`
	`CHECK (check_expression)`

History/Archive Table	Only for system-period temporal tables. Name of the history or archive table linked to the DB2 base table. If this option is selected, the following values can be set for "usage as": blank: Not specified. This is the default setting. A: Archive table. E: History add extra row. This is required for a history table using the `ON DELETE ADD EXTRA ROW` option. H: History table.
CCSID	Encoding scheme. Valid values:

Parameters		
	blank	not specified
	A	ASCII
	E	EBCDIC
	U	Unicode
Temporary	Y	Global temporary table
	N	not temporary.
Volatile	Specifies how DB2 is to choose access to the table. Valid values:	
	Y	Specifies that index access should be used on this table whenever possible for SQL operations.
	N	Specifies that SQL access to this table should be based on the current statistics. This is the default.
Append	Specifies whether append processing is used for the table.	
	Y	Yes.
	N	No.
Check expression	See *Additional attributes / Associations*.	

DB2 View, File Type E

```
13:28:33            *****  P R E D I C T  *****              2007-05-31
                                - Modify file -
File ID ......... HNO-E                                Added 2007-05-31 at 13:44
Type ............ DB2 view                                  by HNO
Contained in DA . B-ARH-DA-C
Keys ..                                                                Zoom: N

Literal name .............
Average count ............
Stability ...............*   Not specified
SQL attributes
  Select ................* A Select all
  With check option .....* N No
Abstract     Zoom: N




  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the DB2 view.
Contained in DA	ID of the database object containing the file.

Intermediate View, File Type IV

The intermediate view can be used to specify subselects, joined tables and table functions in the from clause of DB2 views. The intermediate view defines a temporary view that does not exist in the DB2 catalog.

```
13:28:33            *****  P R E D I C T  *****              2007-05-31
                                - Modify file -
File ID ......... XYZ-IV                              Added 2007-05-31 at 13:24
Type ............ Intermediate view                       by XYZ
Contained in DA . B-ARH-DA-C
Keys ..                                                                Zoom: N

Literal name .............
Average count ............
Stabilty ................*   Not specified
SQL attributes
  Select ................* A Select all
Abstract     Zoom: N





  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the intermediate view.
Contained in DA	ID of the database object containing the file.

Intermediate Table, File Type IT

The field list of an intermediate table can be used to specify the parameters for:

- a database function (object type PR subtype U) or
- an SQL procedure (object type PR subtype R).

```
13:28:33             *****  P R E D I C T  *****              2007-05-31
                              - Modify file -
File ID ......... XYZ-IT                              Added 2007-05-31 at 13:24
Type ............ Intermediate table                     by XYZ
Contained in DA . B-ARH-DA-C
Keys ..                                                                Zoom: N

Literal name ....
Average count ...
Stability ......*   Not specified
Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes* . See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the intermediate table.
Contained in DA	ID of the database object containing the file.

DB2 Query Table, File Type MT

Materialized query tables in DB2 are represented in Predict as objects of type DB2 query table, file type MT.

```
13:27:03                *****  P R E D I C T  *****                  2015-06-29
                                - Add a file -
File ID ......... XYZ_DB2-MT
Type ............ DB2 query table
Contained in DA .
Keys ..                                                                Zoom: N

Literal name ............
Average count ...........                  Stability ........*   Not specified
Check constraint name ...
Select ................* A Select all
Physical attributes in <Default Server>                 (new)
  Number of partitions ..                  Partition size ....    (GB)
  Edit program ..........                  CCSID ............*   (none)
  Row attributes ........   (Y/N)         Volatile .......... N (Y/N)
  Validation program ....                  Maintained by ....* S System
  Audit ................* N Audit none     Include identity .. N (Y/N)
  OBid ..................                  Copy defaults ....* N Exclude
  Data capture .......... N (Y/N)          Enable query opt. . Y (Y/N)
  Restrict on drop ...... N (Y/N)          Append ............ N (Y/N)
  Compress ..............   (Y/N)         Hash size (KB).....
Abstract     Zoom: N                       Logged ............   (Y/N)
  Additional attributes ..* N          Associations ..* N                  ↩
```

Additional attributes are available for files of type MT that are used as accelerator tables.

```
18:17:10                ******  P R E D I C T  *****                  2020-07-02
                                - Modify file -
File ID ......... XYZ-DB_V12                       Modified 2020-04-06 at 13:35
                                                         by XYZ

DBMS extensions in <Default Server>
  Keylabel name .... on default                                        Zoom: N
  Accelerator name . ACC IN D



* Additional attributes ..* S        * Associations ..* S                    ↩
```

Note: Parameters not listed below are described in **DB2 Table, File Type D** or in other sections of this documentation: Parameters common to all object types, for example Keys, are described under ***Global Attributes***. Parameters common to all file types, for example Literal name, are described under ***Common File Attributes***. See also ***Common Parameters for SQL File Types***.

Parameters		
File ID	ID of the Predict object documenting the DB2 query table. See ***Naming Conventions***.	
Contained in DA	ID of the database object containing the file.	
DB2 Attributes		
Number of partitions	The number of partitions of the DB2 query table.	
Edit program	The name of an edit routine for the DB2 query table.	
Validation program	The name of a validation routine for the DB2 query table.	
Audit	The type of access to this table that will cause auditing to be performed. Valid values:	
	A	All
	C	Changes
	N	None

Parameters	
OBid	Identifies the OBID to be used for the table. An OBID is the identifier for an object's internal descriptor in DB2. **Note:** This parameter is required if parameter DB2 ROSHARE parm of the database object containing the table is set to R. See ***Database Type D - DB2***. See your *DB2* documentation for more information.
Data capture	Y | Data changes are passed to a user exit.
Restrict on drop	Y | The DB2 query table cannot be dropped. To drop a table with this setting, this parameter must be set explicitly to N.
CCSID	Encoding scheme. Valid values: blank | not specified A | ASCII E | EBCDIC
Volatile	Specifies how DB2 is to choose access to the table. Valid values: Y | Specifies that index access should be used on this table whenever possible for SQL operations. N | Specifies that SQL access to this table should be based on the current statistics. This is the default.
Maintained by	Specifies how the data in the DB2 query table is maintained. Valid values: blank | Not specified. S | System. U | User.
Include identity	Specifies that, if available, identity column attributes are inherited from the definition of the source table.
Copy defaults	Specifies that column defaults for each updatable column of the definition of the source table are inherited. Valid values: blank | Not specified. Y | Include. N | Exclude. U | Using type.
Enable query opt.	Specifies that the DB2 query table can be used for query optimization.

27 Informix

- Naming Conventions 274
- Informix Table, File Type XT 275
- Informix View, File Type XV 276

Informix tables and views can be documented in Predict with file objects of type XT and XV respectively. These file objects can be used to generate DDMs or `CREATE TABLE/VIEW` statements.

Naming Conventions

The following naming conventions apply to files documenting Informix tables and views.

Upper / lower case

If the Predict parameter General Defaults > Miscellaneous > Upper/lower case / Object ID is set to L, the following attributes of Informix objects are stored in upper and lower case as entered:

- File ID (object IDs containing lower case letters are not recommended)
- DV field expressions
- SQL verifications
- Check expressions
- Constraint names

See also section *Defaults* in the *Predict Administration* documentation.

Length

- Table/View names for Informix objects can have up to 18 characters.
- A fully qualified ID (Creator + Hyphen + Table/View name) may not exceed 27 characters.

Permitted characters

- IDs containing special characters must be enclosed in double quotes, for example:

```
"USR1"-"FIL£ABC"
```

- See overview of permitted characters in ***Naming Conventions***.

Informix Table, File Type XT

```
13:13:58            *****  P R E D I C T  *****              2007-05-31
                             - Modify file -
File ID ......... HNO-XT                          Modified 2007-05-31 at 10:49
Type ............ Informix table                       by HNO
Contained in DA .
Keys ..                                                                Zoom: N

Literal name ....
Average count ...
Stability ......*   Not specified
Informix ONLINE .   (Y/N)
Extensize .......
Nextsize ........
Lock mode ......*
DBspace/Path.
Abstract     Zoom: N




  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the Informix table. See *Naming Conventions* for Informix objects.
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Informix table, the file must be linked to a database of type Informix Handler.
Informix ONLINE	Y | An Informix ONLINE database is used.
Note: The following parameters are only applicable if Informix ONLINE is set to Y.	
Extentsize	Size of the initial extent for the table and and its key.
Nextsize	Size of subsequent extents which are added if necessary.
Lock mode	Determines whether locking is set to page level or row level.

Parameters		
	P	Page level locking.
	R	Row level locking.
DBspace/Path	Name of the DBspace where Informix ONLINE is to store the table. If this parameter is not specified, the table is stored in the DBspace of the database entered under in database.	

Informix View, File Type XV

```
13:13:37            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-XV
Type ............ Informix view
Contained in DA .
Keys ..                                                          Zoom: N

Literal name .............
Average count ............
Stability ...............*   Not specified
SQL attributes
  Select ................* A Select all
  With check option ...... N No

Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the Informix view. See *Naming Conventions* for Informix objects.
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Informix view, the file must be linked to a database of type Informix Handler.

28 Ingres

- Naming Conventions 280
- Ingres Table, File Type JT 281
- Ingres View, File Type JV 282

Ingres tables and views can be documented in Predict with file objects of type JT and JV respectively. These file objects can be used to generate DDMs or `CREATE TABLE/VIEW` statements.

Naming Conventions

The following naming conventions apply to files documenting Ingres tables and views.

Upper / lower case

If the Predict parameter General Defaults > Miscellaneous > Upper/lower case / Object ID is set to L, the following attributes of Ingres objects are stored in upper and lower case as entered:

- File ID (object IDs containing lower case letters are not recommended)
- DV field expressions
- SQL verifications
- Check expressions
- Constraint names

See also section *Defaults* in the *Predict Administration* documentation.

Length

- Table/View names for Ingres objects can have up to 24 characters.
- A fully qualified ID (Creator + Hyphen + Table/View name) may not exceed 32 characters.

Permitted characters

See overview of permitted characters in ***Naming Conventions***.

Ingres Table, File Type JT

```
13:13:01            *****  P R E D I C T  *****            2007-05-31
                                - Modify file -
File ID ......... HNO-JT                              Added 2007-05-31 at 10:28
Type ............ Ingres table                          by HNO
Contained in DA .
Keys ..                                                                Zoom: N

Literal name ......
Average count .....
Stability ........*   Not specified
Journaling ........   (Y/N)
Duplicated ........   (Y/N)
Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the Ingres table. See *Naming Conventions* for Ingres objects.
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Ingres table, the file must be linked to a database of type Ingres Handler.
Journaling	Y: The clause `WITH JOURNALING` is entered in the `CREATE` statement. N: The clause `WITH NO JOURNALING` is entered in the `CREATE` statement.
Duplicated	Y: The clause `WITH DUPLICATES` is entered in the `CREATE` statement. N: The clause `WITH NO DUPLICATES` is entered in the `CREATE` statement.

Ingres View, File Type JV

```
13:13:50            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-JV
Type ............ Ingres view
Contained in DA .
Keys ..                                                          Zoom: N

Literal name .............
Average count ............
Stability ...............*   Not specified
SQL attributes
  Select ................* A Select all
  With check option ...... N No

Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the Ingres view. See *Naming Conventions* for Ingres objects.
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Ingres view, the file must be linked to a database of type Ingres Handler.

29 Oracle

- Naming Conventions 284
- Oracle Table, File Type OT 285
- Oracle View, File Type OV 289

Oracle tables and views can be documented in Predict with file objects of type OT and OV respectively. These file objects can be used to generate DDMs or `CREATE TABLE/VIEW` statements.

Naming Conventions

The following naming conventions apply for Oracle objects (Files of type OT and OV)

Upper / lower case

IDs must be entered in upper case. If the Predict parameter General Defaults > Miscellaneous > Upper/lower case / Object ID is set to L and you try and enter a file ID containing lower case letters, an error message is given.

See also section *Defaults* in the *Predict Administration* documentation.

Length

- Table/View names for Oracle objects can have up to 30 characters.
- A fully qualified ID (Creator + Hyphen + Table/View name) must not exceed 32 characters.

Permitted characters

- IDs containing special characters must be enclosed in double quotes, for example:

```
"USR1"-"FIL£ABC"
```

- See overview of permitted characters in ***Naming Conventions***.

Oracle Table, File Type OT

```
17:21:09                 *****  P R E D I C T  *****                  2017-06-07
                                - Modify file -
File ID ......... XYZ-OT_HEAP                      Modified 2017-05-09 at 14:34
Type ........... Oracle table                            by XYZ
Contained in DA . XYZ-ORA2
Keys ..                                                                  Zoom: N

Literal name ...........
Average count ..........                  Stability ........*   Not specified
Check constraint name ..
Physical attributes in XYZ-ORA1                          (new)
  TEMPORARY ............   (Y/N)         COMMIT ...........*
  ORGANIZATION ........* H Heap
  Cluster name ......... CLUSTER_NAME
  Number of partitions . 3                INDEXING .......... Y (Y/N)
  CACHE ................   (Y/N)         RESULT_CACHE .....* F Force
  ROWDEPENDENCIES ......   (Y/N)         ROW MOVEMENT ......   (Y/N)
  Archive/History table.
          usage as ....*                  ROW_ARCHIVAL ......   (Y/N)
Abstract     Zoom: N


  Additional attributes ..* N        * Associations ..* S                    ↩
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters		
File ID	ID of the Predict object documenting the Oracle table. See *Naming Conventions* for Oracle objects.	
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Oracle table, the file must be linked to a database of type Oracle Handler.	
TEMPORARY	Y	Global temporary table.
	N	Not temporary.
ORGANIZATION	Specify how the table is organized.	
	H	Heap.
	I	Indexed.
	C	Clustered.

Parameters		
Cluster name	If a cluster name is entered here, the clause `CLUSTER name` is generated in the `CREATE TABLE` statement. The table is to be included in the specified cluster.	
Number of partitions	The number of partitions of the table.	
CACHE	Y	Yes.
	N	No.
ROWDEPENDENCIES	Y	Yes.
	N	No.
Archive/History table usage as	Name of the history or archive table linked to the Oracle table. If this option is selected, the following values can be set for "usage as":	
	A	Archive table.
	H	History table.
	blank	Not specified. This is the default.
COMMIT	D	Delete.
	P	Preserve.
	blank	Not specified. This is the default.
INDEXING	Specify whether or not this table is indexed.	
	Y	Yes. Table is indexed.
	N	No. Table is not indexed. This is the default.
RESULT_CACHE	Specify whether query results are stored in the result cache.	
	D	Default.
	F	Force.
	blank	Not specified. This is the default.
ROW MOVEMENT	Y	Yes.
	N	No.
ROW_ARCHIVAL	Y	Yes.
	N	No.
Segment attributes		
PCTFREE	If an integer from 1 - 99 is specified here, the clause `PCTFREE` *n* is generated in the `CREATE TABLE` statement. `PCTFREE` reserves a set amount of room in every block allocated to a table for future updates to that table's data.	
PCTUSED	If an integer from 1 - 99 is specified here, the clause `PCTUSED` *n* is generated in the `CREATE TABLE` statement. `PCTUSED` specifies the minimum level of space usage that Oracle will maintain for each block of the table.	
INITRANS	If a value from 1 - 255 is entered here, the clause `INITRANS` *n* is generated in the `CREATE TABLE` statement.	

Parameters		
	INITRANS is the initial number of transaction entries that are allocated within each block.	
Tablespace	If a tablespace name is entered here, the clause TABLESPACE name is generated in the CREATE TABLE statement. This name represents the tablespace in which the table will be created.	
LOGGING	Specify whether or not to use the LOGGING clause in a CREATE TABLE or ALTER TABLE statement.	
	Y	Yes.
	N	No.
	F	File system like.
	blank	Not specified. This is the default.
Segment storage attributes		
If specified, the values below are used in the STORAGE clause generated with the CREATE TABLE statement. All of the values below must be specified as integers.		
INITIAL	The size of the first extent allocated when the object is created - the original amount of space allocated to the object. A value for Unit has to be applied in addition:	
	K	Kilobyte.
	M	Megabyte.
	G	Gigabyte.
	T	Terabyte.
	P	Petabyte.
	E	Exabyte.
NEXT	The size of every subsequent extent to be allocated. A value for Unit has to be applied in addition. Possible values for Unit are described under INITIAL.	
MAXSIZE	The MAXSIZE clause lets you specify the maximum size of the storage element.	
OPTIMAL	Specifies an optimal size in bytes for a rollback segment.	
MINEXTENTS	The total number of extents to be allocated when the segment is created.	
MAXEXTENTS	The total number of extents, including the first, which can ever be allocated.	
PCTINCREASE	The percent by which each NEXT extent will grow over the last extent allocated.	
FREELISTS	The number of process free lists used to administer the free data blocks.	
FREELISTS GROUPS	Magnitude of the set of free lists.	
BUFFERPOOL	Determines the configuration of the buffer cache.	
	D	Default
	K	Keep
	R	Recycle
	blank	not specified
FLASH_CACHE	Defines the configuration of a second tier of buffer cache on flash disks.	
	D	Default

Parameters		
	K	Keep
	N	None
	blank	not specified
Heap organization attributes		
Table compression	Y	Yes.
	B	Basic.
	A	Advanced.
	QL	Query low.
	QH	Query high.
	Q1	Query low locking.
	Q2	Query high locking.
	AL	Archive low.
	AH	Archive high.
	A1	Archive low locking.
	A2	Archive high locking.
	NL	No row level locking.
	N	No.
	blank	Not specified. This is the default.
Index organizationattributes		
MAPPING TABLE	Y	Yes.
	N	No.
PCTTHRESHOLD	Maximum size of the portion of the row that is stored in the index block, as a percentage of block size. Must be in the range of 1 to 50.	
COMPRESS	Activate index compression for index-organized tables.	
	Y	Yes.
	N	No.
	blank	Not specified. This is the default.
Compress length	Specify the compression length.	
Index overflow attributes		
Refer to the descriptions given in **Segment attributes** above.		
Index overflow storage attributes		
Refer to the descriptions given in **Segment storage attributes** above.		

Refer to your Oracle documentation for more information on these Oracle-specific parameters.

Oracle View, File Type OV

```
13:35:07            *****  P R E D I C T  *****            2007-05-31
                                - Modify file -
File ID ......... HNO-OV                          Modified 2007-05-31 at 10:10
Type ............ Oracle view                          by HNO
Contained in DA .
Keys ..                                                                Zoom: N

Literal name .............
Average count ............
Stability ...............*   Not specified
SQL attributes
  Select ................* A Select all
  With check option ...... N No
  Check constraint name ..
Abstract     Zoom: N





  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the Oracle view. See *Naming Conventions* for Oracle objects.
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Oracle table, the file must be linked to a database of type Oracle Handler.
Check constraint name	Name of check option used if parameter With check option is set to Y. See *SQL Attributes*.

30 Sybase

- Naming Conventions 292
- Sybase Table, File Type YT 293
- Sybase View, File Type YV 294

Sybase tables and views can be documented in Predict with file objects of type YT and YV respectively. These file objects can be used to generate DDMs or `CREATE TABLE/VIEW` statements.

Naming Conventions

The following naming conventions apply to files documenting Sybase tables and views.

Upper / lower case

If the Predict parameter General Defaults > Miscellaneous > Upper/lower case / Object ID is set to L, the following attributes of Sybase objects are stored in upper and lower case as entered:

- File ID (object IDs containing lower case letters are not recommended)
- DV field expressions
- SQL verifications
- Check expressions
- Constraint names

See also section *Defaults* in the *Predict Administration* documentation.

Length

- Table/View names for Sybase objects can have up to 30 characters.
- A fully qualified ID (Creator + Hyphen + Table/View name) must not exceed 32 characters.

Permitted characters

- IDs containing special characters must be enclosed in double quotes, for example:

```
"USR1"-"FIL£ABC"
```

- See overview of permitted characters in ***Naming Conventions***.

Sybase Table, File Type YT

```
13:18:12            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-YT
Type ............ Sybase table
Contained in DA .
Keys ..                                                          Zoom: N

Literal name ....
Average count ...
Stability ......*   Not specified
Database name ...
Segment name ....
Abstract     Zoom: N





  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the Sybase table. See *Naming Conventions* for Sybase objects.
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Sybase table, the file must be linked to a database of type Sybase Handler.
Database name	Name of the database in Sybase containing the table.
Segment name	Name of the segment where the table is to be placed

Sybase View, File Type YV

```
13:19:57            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-YV
Type ............ Sybase view
Contained in DA .
Keys ..                                                          Zoom: N

Literal name .............
Average count ............
Stability ...............*   Not specified
SQL attributes
  Select ................* A Select all
  With check option ...... N No

Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes* . See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object documenting the Sybase table. See *Naming Conventions* for Sybase objects.
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type Sybase table, the file must be linked to a database of type Sybase Handler.

31 General SQL File, File Type X

Files of type General SQL File are used to document all SQL systems not explicitly supported by Predict.

```
13:10:04            *****  P R E D I C T  *****             2007-05-31
                                - Modify file -
File ID ......... HNO-X                              Added 2007-05-31 at 13:01
Type ............ General SQL file                      by HNO
Contained in DA .
Keys ..                                                                Zoom: N

Literal name ...........
Average count ..........
Stability .............*   Not specified
Check constraint name ..
Abstract     Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the Predict object
Contained in DA	ID of the database object containing the file. To generate a DDM from files of type General SQL file, the file must be linked to a database of type General SQL handler.
Check constraint name	The name of a check constraint can be entered here.
Check expression	Y The editor called to edit the check expression of the file depends on your setting in the Profile > Handling screen. See ***Additional attributes / Associations***.

32 RDB

```
13:27:16            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-RDB
Type ............ rdb file
File number ..... 123
Contained in DA .
Keys ..                                                          Zoom: N

Literal name ........
Average count .......
Stability ..........*   Not specified
Sequence field .....*
Abstract     Zoom: N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File ID	ID of the file object.
Contained in DA	ID of the database containing the file (see *Contained in DA*).
Sequence field	The descriptor to be used by Natural for logical sequential reading. Determines the sequence in which records are delivered by the `READ LOGICAL` statement. The GENERATE DDM function will use this field as the default `READ LOGICAL` field in the Natural data definition module.

33 IMS

- IMS Segment Layouts and Userviews - File Types J and K 300
- Editing Field Lists of IMS Files 301

IMS Segment Layouts and Userviews - File Types J and K

```
13:13:40            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-J
Type ............ IMS seg. layout
File number .....     IMS segment: CHD-ARTCHD-ART
Contained in DA .
Keys ..                                                          Zoom: N

Literal name ....
Average count ...
Stability ......*   Not specified
IMS attributes
  Segment name .. ART           Parent ....
  min. length ...               Source-1 ..
  max. length ... 32000         Source-2 ..
  Segment type ..
Abstract     Zoom: N





  Additional attributes ..* N          Associations ..* N
```

The following attributes of an IMS segment (type I) are shown for that file and for the related files of types J and K.

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
File number	The number of the file. A read only field. The number of the related IMS segment is shown. See *File Number* for more information.
IMS attributes	
Segment name	The name of the IMS segment from which the related Predict file object of type I was incorporated.
Min. length	The minimum length of the IMS segment (zero if the length is fixed).
Max. length	The maximum length of the IMS segment (if it is fixed).
Segment type	The type of the IMS segment. Possible values:

Parameters	
	Logical child (C) Logical (L) Physical (P) Virtual (logical) child (V). Segments of type logical occur only in logical IMS databases. Segments of types child, physical and virtual occur only in physical IMS databases.
Parent	The ID of the Predict file object of type I incorporated from the parent segment of the IMS segment (the segment one level above it in the hierarchical structure of the IMS database). For a root segment, this field is left blank.
Source-1	The following rules apply: ■ For a segment of type V, the ID of the Predict file object of type I that was incorporated from the related segment of type C. ■ For a segment of type L, the ID of the Predict file object of type I that was incorporated from the segment of a physical database from which this segment of a logical database is derived. ■ For a segment of type CHILD or P, this field is left blank.
Source-2	The following rules apply: ■ For a segment of type LOGICAL derived from a segment of type C, the ID of the Predict file object of type I that was incorporated from the logical parent of the segment of type C. ■ For a segment of type LOGICAL derived from a segment of type V, the ID of the Predict file object of type I that was incorporated from the logical parent of the segment of type V (the physical parent of the related segment of type C). ■ For any other segment, this field is left blank.

Editing Field Lists of IMS Files

Restrictions that apply when editing a field list of an IMS file depend on the type of the IMS file and are described in the table below.

File Type	Restrictions
I (IMS Segment)	The following attributes can be maintained: ID, keywords, owners, abstract, format, NAT hdr1-3 (Natural headers), NAT editm (Natural edit mask), 3GL specification, Condition name & value and Field name synonyms. See ***Defining Basic Attributes of Fields*** and ***Defining Additional Attributes of fields*** in the section *Field* in this documentation. No fields can be added or deleted. Format changes are rippled across related files of type J or K. Only the following changes of format are allowed:

File Type	Restrictions
	■ between P (packed) and PS (packed signed); ■ between P6 or P7 and D (date); ■ between P12 or P13 and T (time).
J (IMS Segment Layout)	The following rules apply: ■ A file of type J can contain user-defined fields and fields of the related file of type I. The two-character short names of the user-defined fields must fall within the range preceding the parameter Start in logical defined by the DDA in the Miscellaneous defaults of the Modify General Defaults function. Its value is normally HA. ■ Fields of the related file of type I that are included in a File of type J must have the same attributes in the File of type J as they have in the file of type I. ■ Their offset in the file of type J must be the same as their IMS-OFFSET in the file of type I. For a variable-length segment, only one field in one file of type J can be defined as variable length. ■ If it is a field, it must be the last field in the segment. ■ If it is a multiple value field or a periodic group, it can be anywhere in the segment. ■ However, if it is not the last field, its maximum occurrence must be specified. Predict checks that the above conditions are met when the field list of the file is cataloged. Changes to user-defined fields are rippled across related files of type J or K.
K (IMS Userview)	A file of type K can contain fields of the related file of type I and fields of all related files of type J. ID, keywords, owners, comments, format, NAT hdr1-3 (Natural headers) and NAT editm (Natural edit mask), 3GL specification, Condition name & value and Field name synonyms can be maintained.

34 VSAM

- Physical VSAM File - File Type V 304
- VSAM Logical Files, VSAM Userviews - File Types L, W and R 306

See also section *VSAM* in the *Predict and Other Systems* documentation.

Physical VSAM File - File Type V

```
13:38:48            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-VMS
Type ............ VSAM file
File number ..... 123
Contained in DA .
Keys ..                                                          Zoom: N

Literal name .........
Average count ........
Stability ...........*   Not specified
Sequence field ......*
VSAM attributes                  Location                Data set attributes
  VSAM DD name .......             Volume 1 ..             CI size
  VSAM file org .....* K KSDS      Volume 2 ..               Data ......
  Compressed file .... N (Y/N)     Volume 3 ..               Index .....
  Numeric zones .....* F           Volume 4 ..             Recsize
                                   Volume 5 ..               Min .......
                                                             Max .......
Abstract     Zoom: N                                        Free space ..    %


  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under ***Global Attributes***. Parameters common to all file types, for example Literal name, are described under ***Common File Attributes***. See also ***Common Parameters for SQL File Types***.

Parameters	
Sequence field	The descriptor to be used by Natural for logical sequential reading. Determines the sequence in which records are delivered by the `READ LOGICAL` statement. The function Generate DDM uses this attribute as the default `READ LOGICAL` field in the Natural data definition module.
VSAM attributes	
VSAM DD name	This parameter refers to a DD card in batch mode, or to a CICS FCT object. See the *Natural Operations* documentation.

Parameters	
VSAM file org	Valid values: K — KSDS (key-sequenced data set) E — ESDS (entry-sequenced data set) R — RRDS (relative-record data set) V — VRRDS (Variable-length relative record data set)
	The above mentioned file organizations do not use extended VSAM files. This means a four byte *ISN is used to hold the relative byte address (RBA) for ESDS files and the relative record number (RRN) for RRDS and VRRDS files.
	F — ESDS ext. S — RRDS ext. L — KSDS ext. W — VRRDS ext.
	The above mentioned file use extended VSAM files. This means for ESDS, a larger *ISN (up to eight bytes) may be used for VSAMs relative byte address (RBA). The relative record number (RRN) for extended RRDS and VRRDS files remain four bytes long, but will be displayed using this extended ISN field.
Compressed file	Only applicable to files with organization K (KSDS). Y — The record will be truncated if the trailing byte positions are unused.
Numeric zones	Valid entries are C and F. This field affects the representation of positive numbers in packed decimal format. The sign position holds hexadecimal C or F respectively.
Location	
Volume 1 - 5	The volume(s) on which the file is located. Up to five volumes can be specified.
Data set attributes	
CI size - Data	The data control interval size.
CI size - Index	The control interval size for the primary index.
Recsize - Min	The minimum record size.
Recsize - Max	The maximum record size.
Free space	The free space to be allocated (in percent).

VSAM Logical Files, VSAM Userviews - File Types L, W and R

```
13:48:33            *****  P R E D I C T  *****            2007-05-31
                                - Add a file -
File ID ......... HNO-L
Type ............ Logical VSAM
File number ..... 1
Contained in DA .
Keys ..                                                          Zoom: N

Literal name .........
Average count ........
Stability ...........*   Not specified

VSAM attributes
  VSAM prefix ........
  Sequence field ....*
  Organisation ....... KSDS
  Related ............ ARH-VSAM
Abstract     Zoom: N




  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
VSAM prefix	Only applicable to files of types L and R. If this field is left blank, the last 3 digits of the file number are taken as the prefix. Otherwise, a string of up to 20 characters can be specified. The records in the corresponding physical VSAM file (type V) whose primary keys begin with the specified prefix string will be considered as belonging to the logical VSAM file. The length of the primary key specified for the logical VSAM file must be equal to the length of the primary key specified for the physical VSAM file minus the length of the prefix. A dummy field (corresponding to the prefix) preceding the primary key in the logical VSAM file must be defined for the field offsets to be calculated correctly.
Org	The organization of the parent physical VSAM file (type V). Valid values:

Parameters		
	K	KSDS (key-sequenced data set)
	E	ESDS (entry-sequenced data set)
	R	RRDS (relative-record data set)
Related	The ID of the related physical VSAM file (type V). Only applicable to files of types L and R.	
Sequence field	The descriptor to be used by Natural for logical sequential reading. Determines the sequence in which records are delivered by the `READ LOGICAL` statement. The GENERATE DDM function will use this field as the default `READ LOGICAL` field in the Natural data definition module.	

35 ISAM

ISAM Files and Sequential Files - File Types M and S

```
13:46:54            *****  P R E D I C T  *****            2007-05-31
                                - Modify file -
File ID ......... HNO-M                          Modified 2007-05-31 at 13:04
Type ............ ISAM file                            by HNO
File number ..... 1
Contained in DA .
Keys ..                                                                Zoom: N

Literal name .....
Average count ....
Stability .......*   Not specified
Data set attributes
  External name ..                                                     Zoom: N
  Organisation             Size definition             Location
    Type ......*             Unit .........*             Device ....
    Recfm .....*             Primary .......             Volume 1 ..
    Recsize ....             Secondary .....             Volume 2 ..
    Blksize ....             Dir blocks ....             Volume 3 ..
                             Rounded up .... N (Y/N)     Volume 4 ..
                             Contiguous .... N (Y/N)     Volume 5 ..
Abstract     Zoom: N

  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
Data Set Attributes	
External name	Name of the physical file in operating system. Up to 250 characters can be specified (using the Zoom option).
Organization	
Type	The organization of the data set: DA – Direct access PO – Partitioned PS – Sequential blank – None of the above applies
Recfm	The record format of the file: F – Fixed FB – Fixed block FS – Fixed block standard V – Variable VB – Variable blocked VS – Variable blocked standard U – Undefined *blank* – None of the above applies
Recsize	The record size of the file.
Blksize	The block size of the file.
Size Definition	
Unit	The units in which storage space has been allocated to the file: BL – Blocks CY – Cylinders TR – Tracks
Primary	The number of units of storage space allocated to the primary extent of the file.
Secondary	The number of units of storage space allocated to the secondary extent of the file.
Dir blocks	The number of blocks reserved for the directory of the file.
Rounded up	Y – Each space allocation is rounded up to full cylinders.
Contiguous	Y – The space allocated to the secondary extent of the file is contiguous with the space allocated to the primary extent.
Location	
Device	The type of storage device on which the file is located.
Volume 1 - 5	The volume(s) on which the file is located. Up to five volumes can be specified.

36 Entire System Server

Entire System Server Files and Userviews - File Types P and Q

```
13:02:58            *****  P R E D I C T  *****              2007-05-31
                                - Modify File -
File ID ......... PD-P3                          Modified 2007-05-31 at 13:01
Type ............ Sys. Server userview                 by HNO
File number ..... 1
Contained in DA .
Keys ..                                                                Zoom: N

Literal name .........
Average count
Stability .......*   Not specified

Entire System Server attributes
  Sequence Field ....*
  Retrieve ........... Y (Y/N)
  Process ............ N (Y/N)

Abstract     Zoom: N




  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed below are described in other sections of this documentation: Parameters common to all object types, for example Keys, are described under *Global Attributes*. Parameters common to all file types, for example Literal name, are described under *Common File Attributes*. See also *Common Parameters for SQL File Types*.

Parameters	
Sequence field	The descriptor to be used by Natural for logical sequential reading. Determines the sequence in which records are delivered by the `READ LOGICAL` statement. The GENERATE DDM function will use this field as the default `READ LOGICAL` field in the Natural data definition module.
Retrieve	Y | Operation system information can be read with this file.
Process	Y | Operation system activities can be performed via this file.

Note: You cannot add files of type P with the function Add a file. Files of this type are added automatically when Entire System Server is installed.

37 File-Specific Maintenance

- Purge File - Code P 314
- Rename File - Code N 315
- Edit List of Fields - Code L 315
- Force Standard - Code F 317
- Push Backward - Code B 317
- Modify Adabas Attributes - Code J 320
- Modify Vista elements - Code K 320
- Edit Subquery of a File - Code Y 320

Maintenance functions applying to file objects are called from the File Maintenance menu. This menu is called with the command `MAINTAIN file` or with function code M and object code FI in a Predict main menu. The screen is shown in ***File Maintenance Menu*** .

Standard maintenance functions applying to files as well as to most other types of Predict Objects are described in the section *Maintenance* in the *Predict Reference* documentation.

Purge File - Code P

The following files *cannot* be purged with the Purge File function.

- all SAG-owned file objects
- Files of type I (IMS segment). Files of type I can be purged by scratching the IMS database (type I) containing the file.

Two lists are displayed before a file is purged:

- A list of objects and generated code which will not be deleted because they are used in some other object which will not be deleted.
- A list of objects and generated code that will be deleted.

The delete operation is then requested. A list of all deleted objects will be displayed after the delete operation has been executed.

DELETE

The following objects are purged if you confirm this function:

- the file and all its userviews
- all fields of the file and its userviews
- generated code of the file and userview
- all links to databases
- all links from the file to children/from parents
- all links from/to objects that are also purged with this function.

In addition,

- all file relations using this file are set to D (documented).

When an Adabas file is purged, all Adabas attributes and Vista elements of the file are also deleted.

Note: A file cannot be deleted if a DDM for the file exists or the file is implemented.

Rename File - Code N

This function is used to change one or several of the following in a single transaction:

- **File ID**
  The ID will be changed in all objects that are linked to the file via an association and in all file Relations. Predict checks that the ID of the file is still unique.
- **Logical File number**
  Predict checks if all logical file numbers in the database are still unique (except for conceptual databases).
- **File type**
  The field list is loaded into the Predict list editor and is checked. It can then be corrected and has to be cataloged. This is especially important if files of type C are changed to another type. The following rules apply:
  - If a standard file (File type Z) is changed to another file type, all connections to other files are deleted.
  - It is not possible to change the type of a master file if related userviews for this file exist. First connect the userviews to another master file, then change the master file.
- **Master File**
  The new related master file can be specified for files of type J, K, L Q, R, U and W.

  If a userview is connected to another real file, its field list is loaded into the Predict list editor and is checked. It can then be corrected and has to be cataloged.
- **Logical distribution type**
  Only applicable to Adabas files (File type A).
- **Adabas SQL usage**
  Only applicable to Adabas files (File type A). If set to Y, the file is accessible via Adabas SQL Server.

Edit List of Fields - Code L

The field list editor can be invoked in one of the following ways:

- With Y in the field Associations in the bottom line of every Add/Copy/Modify screen.
- With the function Link children (code L) and child type EL.
- With the function Edit list of Fields (code E). This function is not indicated in the File Maintenance menu.
- With the command LINK FILE ELEMENT.

Some additional line and editor commands can be used in the list editor:

Line Commands	
.E	Skips to the Add or Modify Field screen for the field on the current line.
.E(n)	Skips to the Add or Modify Field screen for the next n fields in the list.
Editor Commands	
ADA	Generate two-character field short names for fields that do not already have a short name.
FLIP C	Enables you to enter field IDs with a length of up to 32 characters.
FLIP T	Enables you to enter field IDs and Table/View IDs with a length of up to 32 characters.
FLIP	The default entry fields are displayed.
NU[LL]	Predict automatically sets suppression/null value options for Fields that are added to the dictionary. The value depends on the type of file

Parameter	File Types		
	All SQL Types * except X	X	Other File Types
Unique option = Unique or Desc. type = Primary or Field format = serial	R	R	N
Others	U	blank	N

Note: SQL file types include files of type A with parameter Adabas SQL usage set to Y.

READA	Delete any existing field short names and generate new ones for all fields. This command is only available when editing the field list of a real file or a standard file (not a userview). It is not applicable to field list of SQL files.
SORT ADA	Sort the fields alphabetically by two-character field short name. Fields not on level 1 are not sorted, so group structures are not changed.
SORT	Sort the fields alphabetically by field ID. Fields not on level 1 are not sorted, so group structures are not changed.
SET ADA [ON]	Apply future SCAN commands to two-character field short names instead of field IDs.
SET ADA OFF	Cancel the above setting.

Note: All general commands are described under *Link Editor* of section *Editors in Predict* in the *Predict Reference* documentation.

Comment Lines

When editing field lists of files you can enter comment lines containing descriptive information at any point in the list. The following rules apply:

- Comment lines start with ** or /* in the column Ty.
- Comment lines longer than 32 characters are truncated when files are transferred to Natural LightStorm.
- Comment lines are included in generated DDMs if parameter General comments of function Generate DDM is set to Y.
- Comment lines are ignored for all other generation functions.

Force Standard - Code F

This function compares the connected attributes of all fields defined in the specified standard file with the attributes of the connected fields in other files. If attributes of connected fields are different (and these fields are not marked as non-standard), they are changed to match the standard file if possible. Otherwise, they are marked as non-standard.

Command: `FORCE FILE`

Push Backward - Code B

This function connects fields in a master file or conceptual file to fields in a standard file. The file must not be a userview or a standard file.

The concepts of this function are described in the section ***Rippling*** .

Command: `PUSH FILE <master-file-id>`

```
10:30:50              *****  P R E D I C T  *****            2007-05-31
                            - Push Backward File -

File ID .......... EMPLOYEES




                    Function

                    A   Push back all fields of the file
                    S   Push back selected fields

Function ........

Standard File ..*
Field ID ........                                 with Adabas name .. N (Y/N)
with owner ID ...
with keyword ....


Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters		
File ID	ID of the file to be pushed backward. This value is entered in the File Maintenance Menu and cannot be overwritten here.	
Function	A	Push back all fields of the file. All fields in the master file are coupled with fields in the standard file.
	S	Push back selected fields. Fields in the master file are displayed for selection. Selected fields are coupled with fields in the standard file.
Standard file	Standard file containing the standard fields to be coupled with the master fields. Use asterisk notation to display a list of standard files for selection.	
Field ID	Enter a unique field ID to couple a single field, or display a list of fields for selection by leaving this field blank or using asterisk notation.	
with owner ID	The list of master fields for selection can be restricted to fields with the specified owner. Use asterisk notation to specify a range of owners.	
with keyword	The list of master fields for selection can be restricted to fields with the specified keyword. Use asterisk notation to specify a range of keywords.	
with Adabas name	Y	Field attribute Short name is copied from master field to standard field.

Functional Scope

The following rules apply to both options, A and S.

- Fields already connected to a standard field are not processed.
- If a field with the same ID is already present in the standard file but no link exists, a link is established. The field is marked as non-standard if one of the field attributes does not match.
- If a field is not found in the standard file, it is copied to the standard file and a connection is established.

Push Back all Fields of the File

All fields in the master file that meet the selection criteria are coupled to fields in the standard file.

Push Back selected Fields of the File

Fields in the master file that meet the selection criteria are displayed for selection. Selected fields are coupled to fields in the standard file. This is a two-step process.

1. A list of all fields in the master file which meet the selection criteria is displayed. Fields that are not yet coupled to a field in the standard file are marked will be added (see screen below).
2. Mark fields to be coupled to fields in the standard file with any non-blank character and press ENTER. Marked fields are coupled immediately and are marked is connected to ... in the column Remarks.

```
13:52:09            *****  P R E D I C T  *****            2007-05-31
                     - Push backward Field selection -

From File ID .... FILE13
To   File ID .... STANDARD_FILE

M T  L Field name                        F  Length   Remarks
_    1 Field1                            A     20.0    is connected to FILE12
_    1 Field2                            A     30.0    is connected to FILE12
_ HY 1 Field3                            A     12.0    will be added
```

Modify Adabas Attributes - Code J

Displays the Modify Adabas attributes screen for specifying the physical implementation of an Adabas file. See ***Modifying Adabas Attributes*** for a description.

Command: `MODIFY ADA-ATTR`

Modify Vista elements - Code K

Displays the Modify Vista elements screen (see ***Modifying Vista elements***).

Command: `MODIFY VISTA-FI`

Edit Subquery of a File - Code Y

Invokes the expression editor (see ***Editing the Subquery of an SQL View***). Only applies to SQL views.

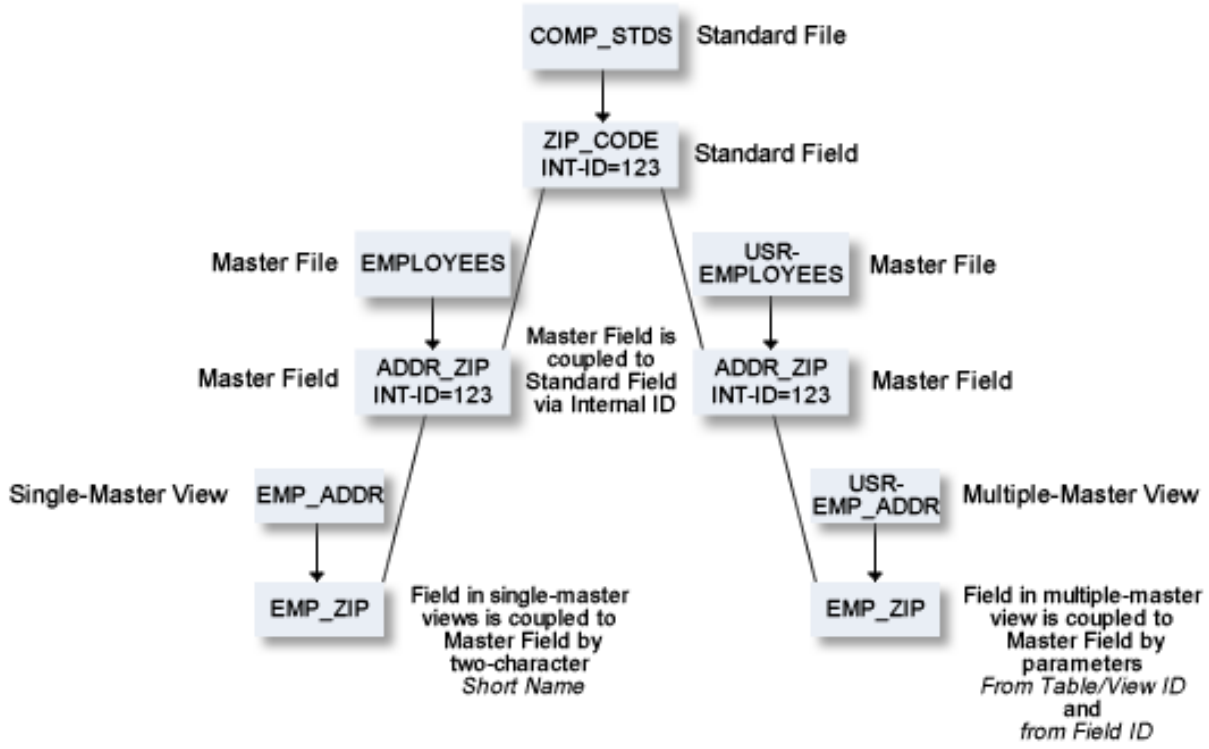
Command: `EDIT FILE SUBQUERY`

38 Rippling - Ensuring Consistent Data Definitions

- Overview 322
- Rippling from Standard Files 324
- Rippling from Master Files to Views/Userviews 328

Overview

Predict rippling options can be used to define a standard, hierarchical data structure and to ensure consistent use of this structure throughout an organization: Whenever field definitions on higher levels are changed, all data definitions on lower levels (including views/userviews) are automatically updated.



General Recommendation

Before you make changes to a standard file, execute the field retrieval function List Fields related to a Z-file.

Listing Rippling Actions

Two profile parameters are available for listing rippling actions:

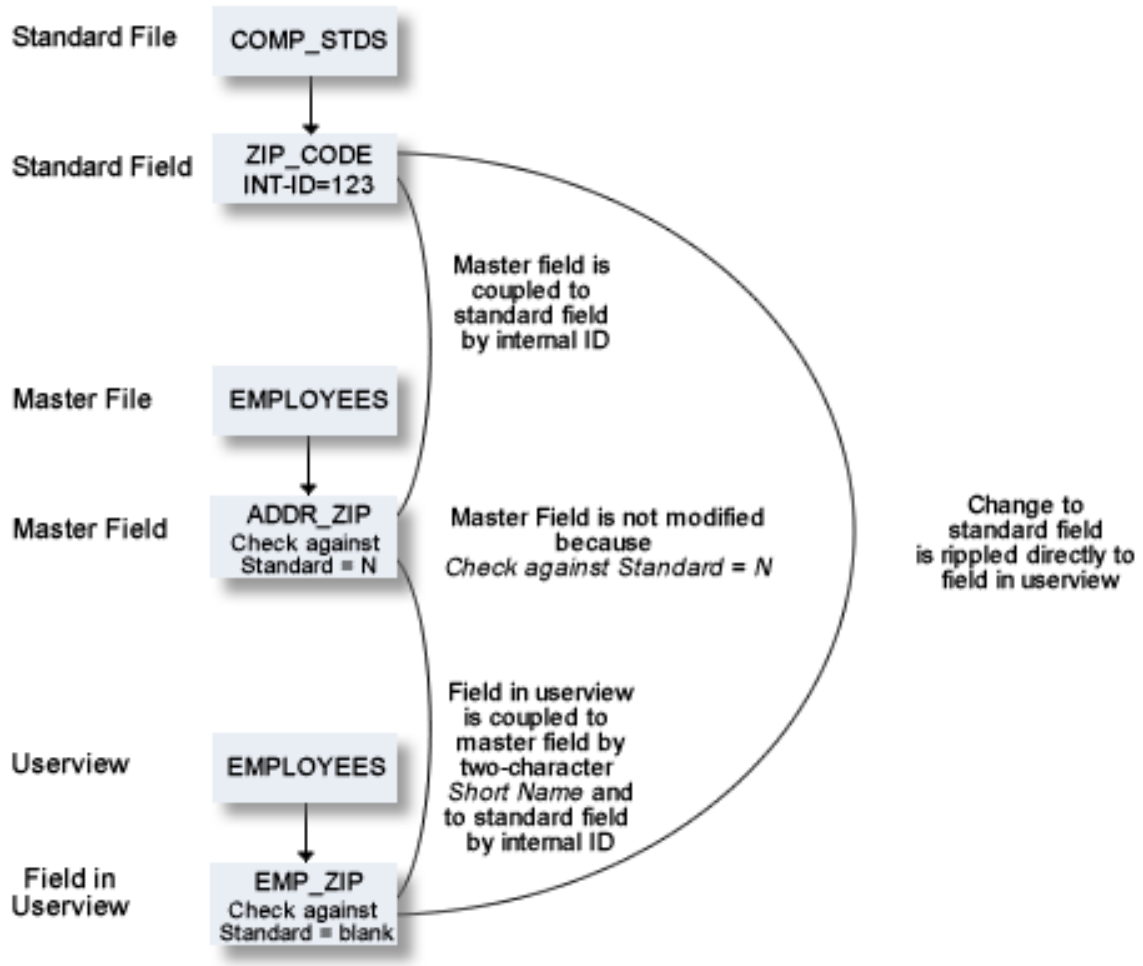
If the parameter Profile > Maintenance options > List action is set to Y, the modified object is displayed after execution.

If the parameter Profile > Maintenance options > MORE type-dependent options > List rippling is set to Y, all coupled fields affected by the modification of a higher-level object are listed.

When external objects are generated for the modified file, the external objects are marked as diff. to documentation.

Check against standard

This option determines whether attribute changes in standard fields are rippled to connected fields. See also **Check against standard** in the section *Field*.



Rippling from Standard Files

Creating a Standard File

There are two methods of creating a standard file:

- **With Coupling**
  Apply the function Push backward to a master file. See *Push Backward*. The fields in the standard file and in the master file are then coupled. Changes to the standard file automatically result in changes to the master file.

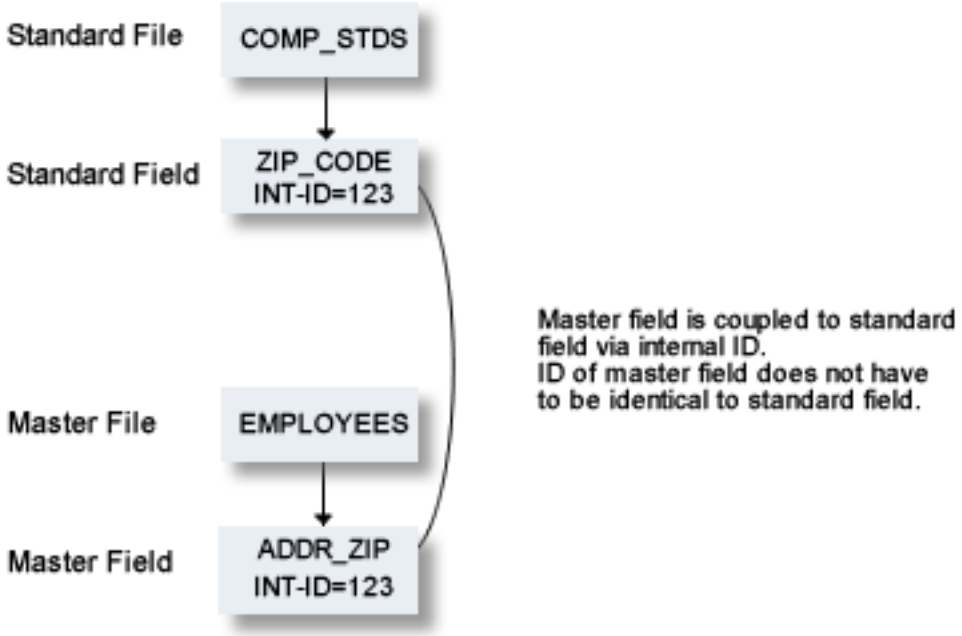
Note: A field in the master file which is already coupled with a standard field is not copied.

- **Without Coupling**
  Create a standard file (file type Z) and copy fields from a master file. Master and standard fields are not coupled and changes to the standard file are not rippled.

Coupling of Standard Fields

Standard fields and connected fields are coupled internally by means of Internal ID.

The coupling remains intact even if the connected field is subsequently renamed.



Functional Scope

The following attributes of a standard field can be rippled to coupled fields at lower levels.

- Field length
- Field format
- Field type
- Suppression option
- Uniqueness option
- Descriptor type (see below)
- Character set
- Timestamp, timezone and precision
- Inline length

If an attribute is not defined in a standard field (which means the attribute is blank if it is alphabetic, or zero if it is numeric), no rippling takes place for this attribute and the lower-level object can be modified without restriction. It is therefore possible to have some field attributes defined centrally and others modifiable without restriction at lower levels. See also *Changing Coupled fields*.

Note: If one of the attributes above is changed and this change is not compatible with the coupled field, the attribute Check against standard of the field is set to N. For example: If you change a field type to HY (hyperdescriptor, this change is not rippled to coupled fields in DB2 files and the attribute Check against standard of the coupled fields is set to N.

Rippling the Attribute Descriptor Type

The attribute Descriptor type of a standard field can have the following values:

D	Disallowed. The descriptor type of coupled fields must be blank. All non-blank descriptor types in coupled fields are set to blank.
F	Force. The descriptor type of coupled fields may not be blank. If a coupled field has a non-blank descriptor type, no rippling is performed. If a coupled field has descriptor type blank, the descriptor type is set to N and a message is given.
blank	Undefined. The descriptor type of coupled fields can be any value, including blank. No checks are performed, no rippling takes place.

Rippling Verifications

When the verification list of a standard field is edited, corresponding changes are automatically made in the verification list of every field derived from the standard field. The following rules apply:

- Every verification contained in the verification list of a standard field must also be contained in the verification list of a field coupled to that standard field. However, the sequence of verifications in the lists can differ.
- If a verification is removed from the verification list of a standard field, the verification is automatically removed from the verification lists of all coupled fields.
- If a verification is added to the verification list of a standard field (at any position), the verification is automatically added to the end of the verification list of all coupled fields.
- If the parameter Check against standard is set to N in connected fields, the checks listed above are not performed.

Changing Coupled Fields

The following rules apply when changing fields at lower levels:

- Attributes not defined in a standard field can be modified in coupled fields.
- Attributes that have been defined in standard fields cannot be modified in coupled fields.
- If an attribute of a coupled field that is defined in the standard field has to be changed, the fields must be uncoupled. See below.

Uncoupling Fields from Standard Fields

Fields can be temporarily or permanently uncoupled from the standard field with the parameter Check against standard in the Modify Field screen.

- **Temporarily**
  Set parameter Check against standard to N. The field is uncoupled temporarily from the standard field from which it was derived. The coupling can be reactivated by resetting Check against standard to blank.
- **Permanently**
  Set the parameter Check against standard to D. The field is uncoupled permanently from the standard field from which it was derived. The coupling cannot be reactivated with the parameter Check against standard. To recouple a field, you must apply the function Push backward to the file.

Defining a Standard File as Default File for SELECT Command

With parameter File for select in the screen Profile > Maintenance Options > MORE Type-dependent options you can specify a default file for the command SELECT. This command can be used in the field List editor of master files or conceptual files.

For single-master views, the default file is the related master file.

Rippling from Master Files to Views/Userviews

The following rules apply:

- Changes to master fields are rippled to fields in userviews that were derived from master files. If the master field is coupled to a standard field, changes to the standard field are rippled to the coupled master field and to the derived field in the userview.
- Changes to fields in userviews are rejected if they are not compatible with the master field.

For example: if a field in a userview is derived from a master field of type T (time), the field in the userview can only be changed to format P with length 13.

All other changes are rejected.

Coupling of Master Fields and Fields in Views/Userviews

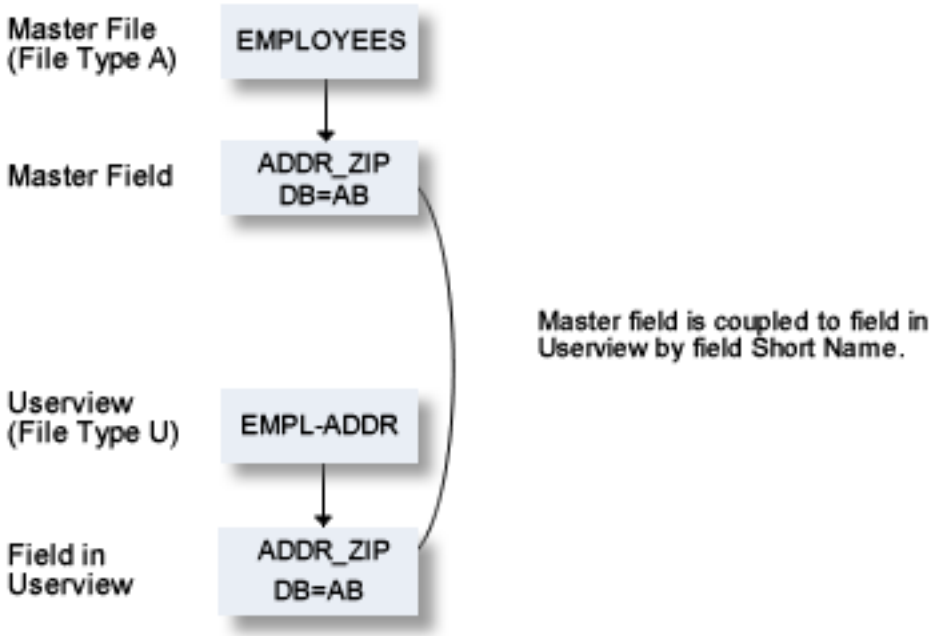
The coupling between master files and views/userviews depends on whether the view is derived from a single master file or from one or several master files.

Single-Master Views

Userviews are derived from one of the following master files:

- Adabas file
- Physical and logical VSAM files
- IMS Segments
- Entire System Server files

Master fields and fields of Userviews are coupled by field short name (column DB in field maintenance screens).



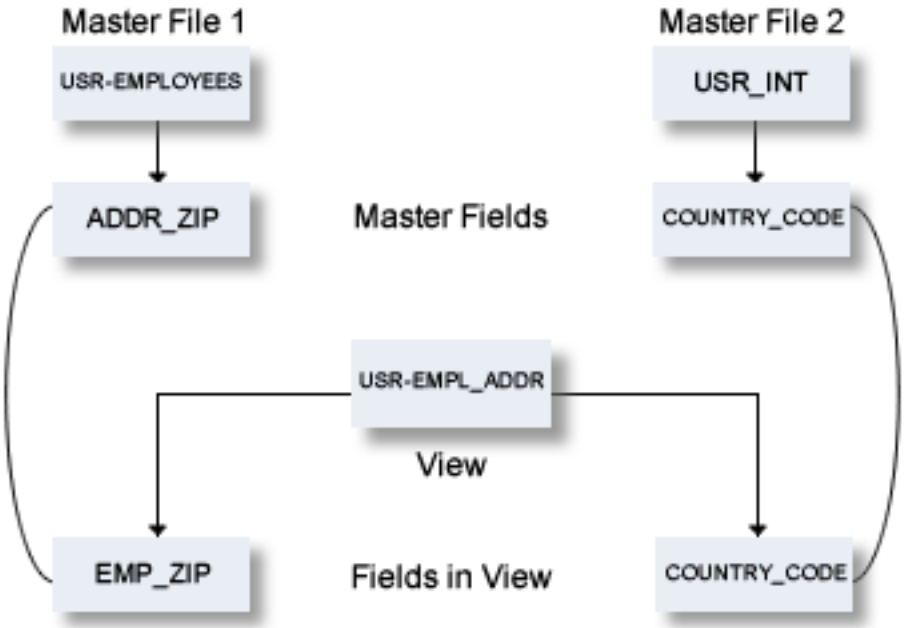
The following table indicates the valid combinations of view types and master file types:

Type of View	Type of Master File
AT	A
B	A(SQL) AT, B
BV	BT, BV
E, IV	D, E, IV
J	I
JV	JT, JV
K	I
L	V
OV	OT, OV
Q	P
R	L
U	A
W	V
XV	XT, XV
YV	YT, YV

Multiple-Master Views

For views which can be derived from several master files, the coupling is established by parameters from Table/View ID and from Field ID in the field List of the file documenting the view. This applies to the following master file types:

- Adabas Files (with SQL usage set to Y)
- Adabas Cluster Tables
- Adabas D Table
- DB2 Table
- Informix Table
- Ingres Table
- Oracle Table
- Sybase Table



The coupling above is documented as shown in the field list of file USR-EMPL_ADDR in the screen below.

```
>                    > + Fi: USR-EMP_ADDR                              L:    1 S:
   Ty L Field ID           from Table/View ID          Field ID           All
   *- - ------------------ ---------------------------- ------------------
      1 EMP_ZIP            USR-EMPLOYEES               ADDR_ZIP
      1 COUNTRY_CODE       USR-INT                     COUNTRY_CODE
```

Functional Scope

If fields in a master file are modified, views and userviews coupled to these fields are changed accordingly. The following rules apply for this rippling:

Attributes which are always Rippled

The following attributes are always rippled:

- short name (if applicable)
- Field type
- suppression / null value option
- uniqueness option
- character set
- null default option

Attributes which are Rippled if Identical

The following attributes are rippled if the attribute values in the userview and the master field were identical before the master field was modified:

- Field ID
- length, format (both must be identical)
- max. occ.
- gr. structure
- justify
- header / edit mask
- Field/View name name synonym

Abstract

The abstract of a field is rippled according to the setting of the following parameter in the screen Profile > Maintenance Options > MORE Type-dependent options:Profile SYSTEM

Ripple abstract	N	Abstract is not rippled.
	T	Abstract is rippled.
	L	Abstract is rippled only if the abstract was identical in the view/userview and the master file before the abstract was changed in the master file.

Rippling Verifications from Master Field to View/Userview

When a verification list of a master field is edited, corresponding changes are automatically made in the verification list of fields in the view/userview derived from the master file. The following rules apply:

- The verification list of a field in a userview does not have to contain all the verifications that are contained in the list of the master file field from which the userview field has been derived.
- If a verification is removed from the verification list of a master field, the verification is automatically removed from the verification list of coupled fields.
- If a verification is added to the verification list of a master field, it is automatically added to the verification list of coupled fields.

39 File Retrieval

- File Retrieval Screen 334
- File-specific Retrieval Parameters 334
- File-specific Retrieval Functions 335
- Layout of File Lists 337
- Output Options for File Retrieval 338
- Output Options for File Retrieval - Continued 340

Standard retrieval functions are described in the section *Retrieval* in the *Predict Reference* documentation.

File Retrieval Screen

The File Retrieval screen below is called with function code R and object code FI in a Predict main menu or with command `RETRIEVE FILE`.

```
13:37:40             *****  P R E D I C T  *****             2007-05-31
Plan  10                  - (FI) File Retrieval -                  Profile HNO

 Retrieval Type                            Retrieval Type

 D  Files                                  B  Files with parents
 E  Execute retrieval models               O  Files with no parent
 C  Dummy/Placeholder files                T  Files with children
 A  Difference of files                    U  Files with no child
                                           R  Files related to a file

Retrieval type ...
Output mode .....* L List

File ID ..........                                  Files of type ....*
Contained in DA ..                                  File number .......
External name ....
Restrictions ....*   Profile HNO,used               Model ............*
Output options ..*   Profile HNO                    Association ......* EL

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

File-specific Retrieval Parameters

Parameters	
Contained in DA	Restricts the scope of functions to files and userviews contained in the specified database.
Files of type	Restricts the scope of functions to files of the type specified. An asterisk displays a selection window with the valid file types. See ***File Type*** for a list.
File number	Restricts the scope of functions to files with this number.

Parameters	
External name	Name of the physical implementation (DSN, Table names). Can have up to 250 characters, but only the first 50 are evaluated by Predict retrieval functions.

File-specific Retrieval Functions

Difference of Files - Code A

This function compares files and displays the differences. The file attributes, the fields and the field attributes can be compared. The fields are compared using the field ID. If a userview is compared with its master file, however, the fields are compared by two-character Short name. The userview is always taken as first file, irrespective of which file is entered under First File ID.

A screen appears for entering the names of two files and selecting the attributes to be compared.

Command: `DIFFERENCE FILE`.

```
13:45:50             *****  P R E D I C T  *****            2007-05-31
Plan  10                   - Difference of Files -

First File ID ..... HNO-FI1
Second File ID ....


Options
  List Fields ................* D Differences only
  Compare File attributes ...... N (Y,N)


Mark Field attributes which should be compared.
  X  the order                X  the existence
  X  abstract                 X  owner IDs              X  keywords
  X  description              X  Field name synonyms    X  standard File
  X  verifications            X  Adabas attributes      X  Field definition
  X  Natural attributes       X  specification for 3GL  X  VSAM attributes



Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters	
First file ID, Second file ID	The names of the files to be compared. Asterisk notation can be used to compare one file with many files or two sets of files.
Options	
List fields	Determines how the result of the comparison operation is to be displayed: A — all fields are listed and differences are marked D — only fields with differences are listed N — a message indicates if differences were found.
Compare file attributes	Y — File definitions are to be compared.
Field attributes to be compared	
the order	Differences in the order of fields in a file. **Note:** The system checks for each field in the list whether the previous field of file 1 is identical to the previous field of file 2. Redefinitions are ignored in the check for previous field. In the example below, EL1 is regarded as previous field of EL2 for both files FIRST_FILE Ty L Field ID 1 EL_1 RE 1 EL_1 2 EL_11 2 EL_12 1 EL_2 SECOND_FILE Ty L Field ID 1 EL_1 RE 1 EL_1 2 EL_12 2 EL_11 1 EL_2 However, the difference in the order of the redefinitions is recognized and the message "Redefinitions are different" is given.
the existence	A message is issued if a field exists only in one file.
abstract	Abstract of fields.
owner IDs	Owners of fields.
keywords	Keywords of fields.
description	The description of fields.
Field name synonyms	Field-name-synonyms, language-synonym-names.
Standard file	Standard file, non-standard definition.
Verifications	Verifications linked to fields.
Adabas attributes	Security access level, security update level.
Field definition	Descriptor type, level number, field format, character set, field length, field type, max. occurrences, unique option, suppression option, user exit, Adabas EDIT mask, IMS offset, IMS variable field, DB2 field procedure, DB2 field

Parameters	
	parameter, DB2 master file, DB2 master field, DB2 index cluster, DB2 index subpage, DB2 index bufferpool.
Natural attributes	Edit mask, field headings.
specification for 3GL	Init value, justify, condition names, index name, depend name, Gr. structure.
VSAM attributes	Alternate index name, VSAM flags.

Note: This command can also be performed in batch mode. See the section *Predict Commands* in the *Predict Reference* documentation for a list of keywords and parameters. These keywords are not available online.

Files Related to a File - Code R

Certain files are considered to be logically related. For example, Adabas files and userviews; VSAM files and VSAM userviews; logical VSAM files and their userviews. This function displays the following relationships of files:

- master files with their userviews
- userviews with their master files and other userviews of these master files.

For physical VSAM files also the related logical VSAM files are listed, for IMS segments also the IMS segment layouts.

Command: `RELATED FILE`.

Layout of File Lists

```
13:46:22            *****  P R E D I C T  *****             2007-05-31
                               - List Files -                     Page:    1

 Cnt  File ID                            Type   Fnr    DDM Impl Other

    1 A                                    S
    2 * A-ADDR-File                        A     59         A
    3 * A-ANSP-File                        A     84         A
    4 A-File                               A      1
    5 A-U-File                             U      1
    6 Az-a-File                            A     54
    7 AA-TD                                D
    8 AA-TS                                S
```

Meaning of Columns	
File ID	ID of the file definition.
Type	File types and their codes are listed in the section *File Type* .
Fnr	The physical file number. Only applicable for Adabas files.
DDM	An asterisk in this column indicates either that a Natural data definition module has been generated for the file or that the file has been used by either Adabas Native SQL or the Predict Preprocessor.
Impl	How a file is implemented: A – The file has been loaded into Adabas C – ADACMP definitions have been generated for the file D – The file has been implemented in DB2 U – UDFs have been generated for the file (IMS) S – Vista translation table generated
Other	An asterisk in this column indicates that at least one copy code member for Assembler, C, COBOL, FORTRAN or PL/I or at least one ADAINV or ADASCR card member has been generated for the file.

Output Options for File Retrieval

Notes:

1. Unless output mode is S, the option Cover page is always valid.
2. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Adabas attributes	Y		Y				Y		Y				Y			
Association attributes			Y	Y	Y	Y			Y	Y	Y	Y				
Attributes	Y		Y				Y		Y				Y			
Check expression	Y		Y				Y		Y				Y			
Composed Fields										Y		Y				
Connecting character				Y						Y						
Description	Y		Y	Y			Y		Y	Y			Y			
Display length										Y		Y				
Display modifier	Y		Y				Y		Y				Y			

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Dummy/Placeholder										Y		Y		Y		Y
DV-Field expression										Y						
Extract	Y		Y	Y			Y		Y	Y			Y	Y		
Generation layout										Y		Y				
Adabas version										Y		Y				
Language										Y		Y				
Alignment/sync.										Y		Y				
Position/Offset										Y		Y				
Counter length										Y		Y				
Compiler										Y		Y				
Replace with syn.										Y		Y				
Keywords	Y		Y	Y			Y		Y	Y			Y			
Linked Verification										Y						
Mark implementation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
No. abstract lines	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
Owner	Y		Y	Y			Y		Y	Y			Y			
With users	Y		Y	Y			Y		Y	Y			Y			
Show implementation	Y		Y				Y		Y				Y			
Subquery	Y		Y				Y		Y				Y			
Synonyms										Y		Y				
Trigger	Y		Y				Y		Y				Y			
Use Con-form	Y		Y				Y		Y				Y			
User exit	Y		Y				Y		Y				Y			
Vista elements	Y		Y				Y		Y				Y			
3GL specification										Y						

Output Options for File Retrieval - Continued

Retrieval Type	U		E				C				R	
Output Mode	D	L	T		X		L		D		L	
Current/Related	c	c	c	r	c	r	c	r	c	r	c	r
Adabas attributes	Y											
Association attributes			Y	Y								
Attributes	Y			Y		Y						
Check expression	Y											
Composed Fields												
Connecting character				Y		Y				Y		
Description	Y					Y				Y		
Display length												
Display modifier	Y											
Dummy/Placeholder				Y		Y	Y		Y			
DV-Field expression												
Extract	Y			Y		Y			Y	Y		
Generation layout												
Adabas version												
Language												
Alignment/sync.												
Position/Offset												
Counter length												
Compiler												
Replace with syn.												
Keywords	Y			Y		Y				Y		
Linked Verification												
Mark implementation	Y	Y	Y	Y	Y	Y		Y		Y	Y	Y
No. abstract lines	Y	Y		Y		Y		Y		Y	Y	Y
Owner	Y			Y		Y				Y		
With users	Y									Y		
Show implementation	Y											
Subquery	Y											
Synonyms												
Trigger	Y											
Use Con-form	Y					Y				Y		

Retrieval Type	U		E				C				R	
Output Mode	D	L	T		X		L		D		L	
Current/Related	c	c	c	r	c	r	c	r	c	r	c	r
User exit	Y											
Vista elements	Y											
3GL specification												

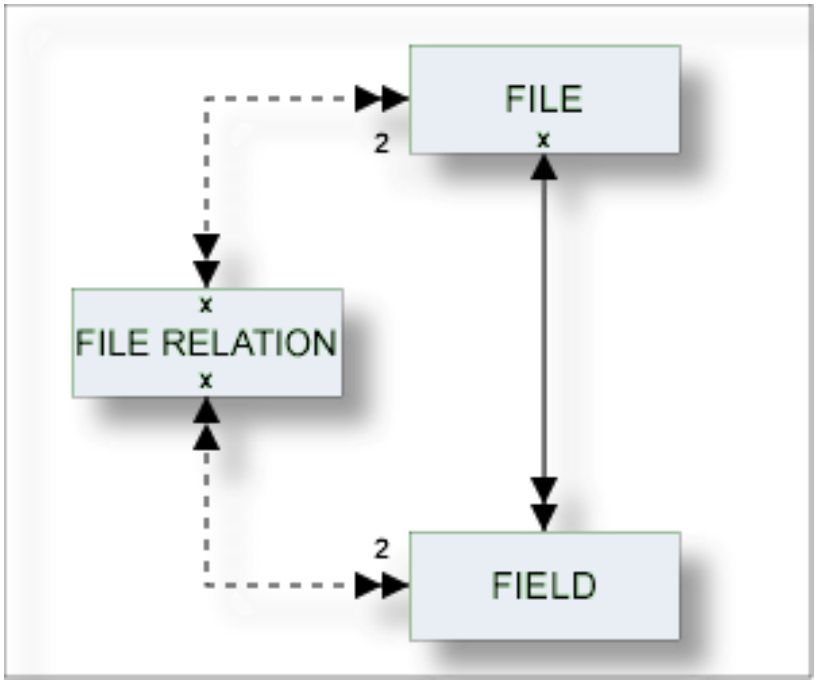
VII

- 40 File Relation 345
- 41 Interface 355

40 File Relation

- File Relation Maintenance 346
- File Relation Retrieval 352

The object type File Relation documents relationships between files. The relationship is established by means of references to fields.



File Relation Maintenance

The following topics are covered below:

- File Relation Maintenance Menu
- File Relation Types
- Add a File Relation Screen
- Validity Checks for File Relations

File Relation Maintenance Menu

The File Relation Maintenance menu is called with function code M and object code RL in a Predict main menu or the command `MAINTAIN FILE RELATION`.

```
13:05:08             *****  P R E D I C T  *****            2007-05-31
Plan   3             - (RL) File relation Maintenance -           Profile HNO

  Function                                Function

  A  Add a File relation                  D  Display File relation
  C  Copy File relation                   L  Link children
  M  Modify File relation                 S  Select File relation from a list
  N  Rename File relation
  P  Purge File relation


Function .........

File relation ID .                                   Attributes.........*
Copy ID ..........
for file ID ......

Restrictions .....*   Profile HNO,used               Association.....*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Executes one of the maintenance functions. All standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation.
for file ID	For the Select function: a file ID can be specified as an additional selection criterion. Asterisk notation is possible.

File Relation Types

The table below contains a list of all valid file relation types.

Code	File Relation Type
C	Two files of type A are physically coupled.
D	The file relation is only documented.
K	Common keys. This file relation type is only valid for file types YT and YV (Sybase tables and views). The field linked to the file relation must have a non-blank descriptor type. Predict checks whether the number, formats and character sets of the fields - or source fields in the case of superdescriptors - in file 1 and file 2 agree. For Sybase, you can generate a common key from a file relation of this type. For other database management systems, file relations of this type are used for documentation purposes only.

Code	File Relation Type
N	This file relation type documents the models used by Natural Construct. See *Defining File Relations for Objects in Predict* in the *Natural Construct User's* documentation.
R	Ref. Constraint. Files of type AT, BT, D, JT, OT, X, XT, XV, Y, and YV are connected by referential integrity.
S	Files of type A are soft coupled.

Add a File Relation Screen

The screen below is displayed for the Add a File Relation function. The Copy and Modify screens are similar.

```
13:30:23             *****  P R E D I C T  *****             2007-05-31
                          - Add a File relation -
File relation ... HNO-RL
Type ...........*
Keys ..                                                              Zoom: N

Cardinality ..*    :
File 1                                                 Minimum ...
  File ID ....* HNO-FI1                                Average ...
  Field ID ...* HNO-EL1                                Maximum ...
File 2                                                 Minimum ...
  File ID ....* HNO-FI2                                Average ...
  Field ID ...* HNO-EL1                                Maximum ...
Constraint attributes
  Update type .....*    (none)
  Delete type .....*    (none)
  Constraint name ..
  Enforce ..........   (Y/N)
Usage ........*    (none)

Abstract     Zoom: N


  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed here are described under ***Global Attributes***.

Parameters	
File Relation	The ID of the file relation object.
Type	The type of file relation. For a list of valid values, see *File Relation Types*. See also table in the section *Validity Checks for File Relations*.
Cardinality	The number of records of each file that is permitted in any occurrence of the file relation. Valid values: 1 — one (must be one) C — none or one (can be one) CM,CN — one or one or more (can be many) M, N — one or more (must be at least one)
File 1 (File ID, Field ID)	One of the related files. If the type of **file relation is R**, the field which is used to link this table must be a unique descriptor or ISN (Adabas), a primary index (for DB2) or a unique key (for other SQL systems).
File 2 (File ID, Field ID)	The other related file. If the type of **file relation is R**, the field which is used to link this table must be one of the following: ■ descriptor (**descriptor D**) ■ foreign key (**descriptor E**) ■ foreign index (**descriptor F**) ■ primary index (**descriptor P**)
Minimum	The minimum number of occurrences of a field from File 1 or File 2 in the file relation.
Average	The average number of occurrences of a field from File 1 or File 2 in the file relation.
Maximum	The maximum number of occurrences of a field from File 1 or File 2 in the file relation.
Constraint Attributes	
Update type	The type of constraint to be applied. C — Cascade. R — Restricted. L — Suffix as line number (file relation type D or N). N — Renumber suffix (file relation type D or N). S — Set NULL. D — Set Default. A — No Action.
Delete type	The type of constraint to be applied. C — Cascade. R — Restricted. L — Suffix as line number (file relation type D or N).

Parameters		
	N	Renumber suffix (file relation type D or N).
	S	Set NULL.
	D	Set default.
	A	No Action.
Constraint name	The constraint name for a file relation of type D and R. For **files of type A**, the constraint name must follow the Adabas short name conventions. For details refer to *Field Names* in *Record and Field Definitions* of the section *Adabas Design* in the *Adabas Concepts and Facilities* documentation.	
Enforce	Only applicable to file relations of type D or R. Specifies whether or not the referential constraint is enforced by DB2 during normal operations such as insert, update or delete. Valid values: Y (Enforce) or N..	
Construct Usage		
Usage	Only applicable to file relations of type N or D. Describes how the file relation is evaluated in Natural Construct:	
	A	Construct aggregate.
	I	Construct inheritance.

Validity Checks for File Relations

The validity checks performed by Predict depend on the file relation type:

Code C

Type	Applicable for	Validity Checks
Physically Coupled	Adabas	May not be any of the following: ■ redefined field ■ group ■ periodic group ■ member of a periodic group ■ hyperdescriptor ■ phonetic descriptor The two fields in the file relation must be descriptors with the same length and format.

Code D

Type	Applicable for	Validity Checks
Documented	all types	None

Code K

Type	Applicable for	Validity Checks
Common Keys	Sybase tables and views	The field linked to the file relation must have a non-blank descriptor type

Code N

Type	Applicable for	Validity Checks
Natural Construct	all types	Both the field and file containing the file relation must be defined in Predict.

Code R

Type	Applicable for	Validity Checks
Referential Constraint	Adabas File Adabas Cluster Table DB2 Table Oracle Table Adabas D Table Informix Table or View	Must be marked in the table of *file 1*: For file type DB2 table or Informix table/view: ■ as primary index (descriptor type P), ■ foreign index (descr. type F) ■ or index (descr. type D), ■ and as unique (unique option U) for file type Adabas file: ■ as unique descriptor or ISN; for file type Adabas cluster table: ■ as primary index (descriptor type P); for other file types: ■ as unique (unique option U). Must be marked in the table of *file 2*: for file type Adabas file: ■ as descriptor; For file type Adabas cluster table: ■ as foreign index (descr. type F)

Type	Applicable for	Validity Checks
		■ or foreign key (descr. type E); for other file types: ■ as primary index (descr. type P), ■ foreign index (desc. type F) ■ or foreign key (descr. type E).

Code S

Type	Applicable for	Validity Checks
Soft-coupled	Adabas	May not be any of the following: ■ redefined field ■ group ■ periodic group ■ member of a periodic group ■ hyperdescriptor ■ phonetic descriptor The first field in the file relation must be a descriptor; the second field must have the same format.

With Predict retrieval functions, file relations between physical files are treated as though they were connected with the userviews of the files.

File Relation Retrieval

File Relation Specific Retrieval Parameter

using file Restricts the scope of the function to file relations which apply to the specified file. Asterisk notation can be used to specify a range of files.

Layout of File Relation Lists

```
13:36:22             *****  P R E D I C T  *****             2007-05-31
                           - List File relation -


---------------------------------------------------------------------------
Cnt  File relation ID                 Type File 1           File 2

   1 AER-TST-SYS1-19                    D  AER-TST-SYS1    AER-TST-SYS2
   2 AER-TST-SYS2-18                    D  AER-TST-SYS2    AER-TST-SYS1
   3 AMMM                               D
   4 ARH-RL                             D  ARHTESTCHEN      ARH-BT1
   5 ARH-RL-FUER-BT-FILE                K  ARH-BT1          ARH-BT1
   6 ARH-RL-K                           K  ARH-D1           ARH-D1
   7 ARH-RL1                            D  ARH-123456789012 ARH-123456789012
   8 ARH-RL2                            R  ARH-OT1          ARH-OT1
```

Meaning of Columns	
File Relation ID	ID of the file relation object.
Type	The type of file relation. See table in the section *Validity Checks for File Relations* for list of valid types and codes.
File 1	One of the related files.
File 2	The other related file.

Output Options for File Relation Retrieval

The output options valid for this object type are identical to those for object type Dataspace. See *Output Options for Dataspace Retrieval*.

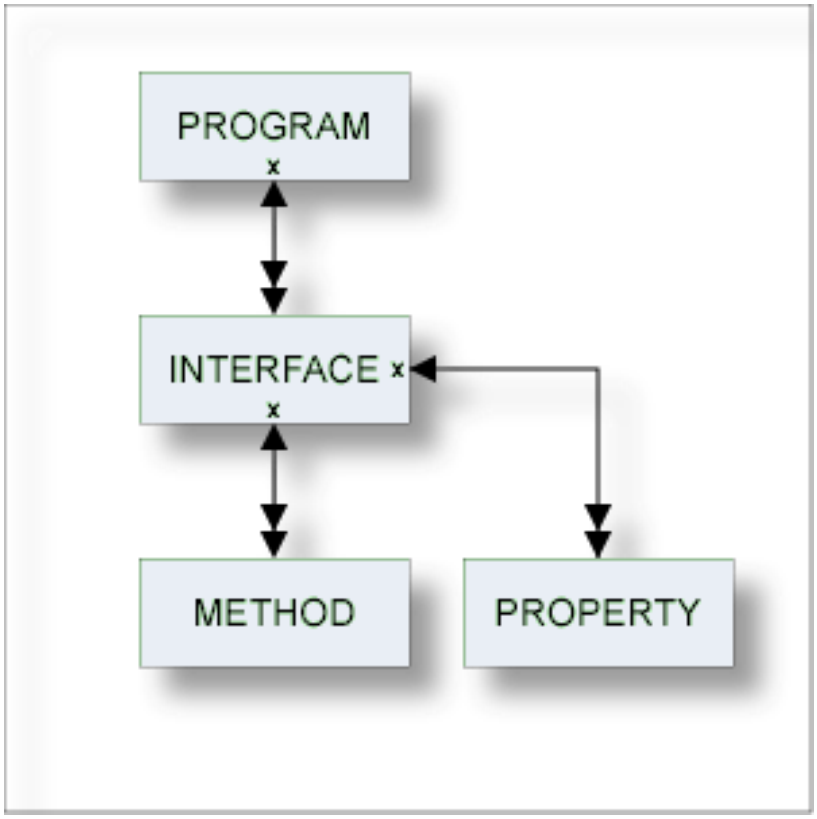
41 Interface

- Interface Maintenance Menu 356
- Defining Basic Attributes of an Interface 357
- Interface Retrieval 358

This object type, together with object types Method, Property and Program, is used to document the Natural programming object class.

In the predefined Predict metastructure, an interface can have passive and active associations of the following types:

Valid passive association:	"Defined in PR"
Valid active associations:	"Contains MD" "Contains PY"



Interface Maintenance Menu

This menu is called with function code M and object code IE in a Predict main menu, or with the command `MAINTAIN INTERFACE`.

```
13:33:11             *****  P R E D I C T  *****             2007-05-31
Plan   0               - (IE) Interface Maintenance -           Profile SYSTEM

  Function                                Function

  A  Add a Interface                      D  Display Interface
  C  Copy Interface                       L  Link children
  M  Modify Interface                     S  Select Interface from list
  N  Rename Interface
  P  Purge Interface


Function ..............

Interface ID ..........                                   Attributes....*
Copy ID ...............


Restrictions .........*   Profile Default,empty          Association...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters

The Interface Maintenance menu contains only global attributes. See ***Global Attributes***.

The functions are described in the section *Maintenance* in the *Predict Reference* documentation.

Defining Basic Attributes of an Interface

The following screen appears for the function Add an Interface. The screens for functions Copy and Modify are similar.

```
13:37:04             *****  P R E D I C T  *****             2007-05-31
                          - Add a Interface -
Interface ....... INTERFACE

Keys ..                                                              Zoom: N

Attributes
  Interface name ...
  GUID .............
  Abstract    Zoom: N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Interface	ID of the interface.
Interface name	Name of the interface.
GUID	The globally unique ID of the interface.

Interface Retrieval

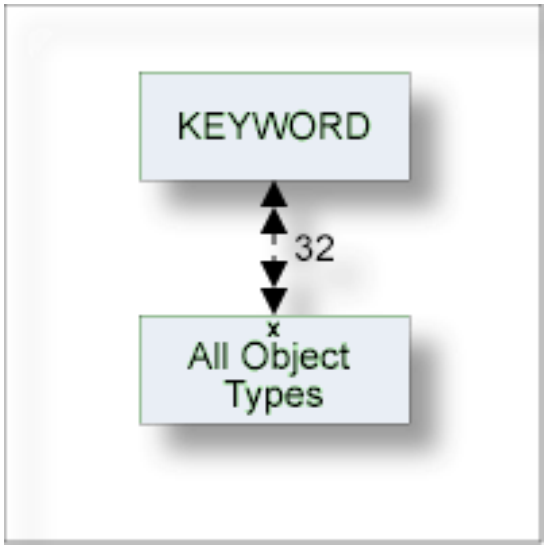
Information on interface objects is gathered using standard retrieval functions. See the section *Retrieval* in the *Predict Reference* documentation.

Output Options for Interface Retrieval

The output options valid for this object type are identical to those for object type Dataspace. See *Output Options for Dataspace Retrieval*.

VIII Keyword

Predict objects of type Keyword are used to relate objects logically, for example, all objects belonging to an application or all objects used in a particular business context.



In the predefined Predict metastructure, a keyword can be related as a child object to objects of all types including other keywords.

The description of object type Keyword is organized under the following headings:

Maintaing Objects of Type Keyword

Keyword Retrieval

42 Maintaing Objects of Type Keyword

- Keyword Maintenance Menu 362
- Defining Basic Attributes of Keyword 363
- Keyword Maintenance Functions 363

Keyword Maintenance Menu

The Keyword Maintenance menu is displayed with function code M and object code KY in a Predict main menu or the command `MAINTAIN KEYWORD`.

```
13:44:12             *****  P R E D I C T  *****              2007-05-31
Plan   0                - (KY) Keyword Maintenance -               Profile HNO

 Function                              Function

 A  Add a Keyword                      D  Display Keyword
 C  Copy Keyword                       L  Link children
 M  Modify Keyword                     S  Select Keyword from list
 N  Rename Keyword                     E  Link/Unlink objects
 P  Purge Keyword


Function .......


Keyword ID .....                                        Attributes ....*
Copy ID ........

Restrictions ..*   Profile HNO,used                     Association ...*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Executes one of the functions in the Keyword Maintenance menu. Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. Functions Purge keyword and Link/Unlink objects are described below.
Copy ID	For the Copy function: the ID of the new keyword.

Defining Basic Attributes of Keyword

The following screen is displayed for the Add/Copy/Modify Keyword function.

```
13:13:45            *****  P R E D I C T  *****            2007-05-31
                            - Add a Keyword -
Key ID .......... HNO-KY

Keys ..                                                          Zoom: N
Abstract
```

The parameters are described under ***Global Attributes***.

Keyword Maintenance Functions

Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. The following functions are described below.

- Purge Keyword - Code P
- Link/Unlink Objects - Code E

Purge Keyword - Code P

If you confirm this function with DELETE, the following are deleted:

- the keyword object
- all links to child objects
- all links from parent objects

The number of objects affected by this function is displayed.

Link/Unlink Objects - Code E

A link between a keyword and a Predict object can be established or deleted directly using the Link/Unlink objects function. Linking or unlinking a keyword and objects is a three-step process:

1. Call the Link/Unlink objects screen by entering function code E in the Keyword Maintenance menu and specify an object type. Enter an asterisk to display a list of types for selection.
2. Enter search criteria to display a list of objects to be linked or unlinked.
3. Link or unlink objects by entering L (link) or U (unlink) in the first column.

Steps 2 and 3 are described in more detail below.

Step 2: Specifying Search Criteria

The search criteria depend on the type of object to which a keyword is to be linked. The criteria in the screen below apply when linking databases.

```
13:37:03            *****  P R E D I C T  4.2.2  *****             2007-05-31
Plan   0                 - Link/Unlink objects -

Keyword ID ....... HNO-KY                            Added 2007-05-31 at 13:29
                                                        by HNO

Link to object type ..* DA ( Database )


Search criteria
  Database ID .........
  Type ...............*
  Database number .....
  Belongs to VM .......




Restrictions .........*   Profile HNO,used              List option ....* A


Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters		
Keyword ID	ID of the keyword to be linked.	
Link to object type	Type of object to which the keyword is to be linked. Keywords can be linked to any predefined or user-defined object type.	
Search criteria	These are object type dependent.	
Restrictions	Restrictions can be used to limit the number of objects for selection. See *Restrictions* in the section *Retrieval* in the *Predict Reference* documentation.	
List Option	The range of objects to be displayed in the list can be restricted as follows:	
	L	only objects linked to the keyword are listed
	U	only objects *not* linked to the keyword are listed
	A	all objects meeting the rest of the criteria are listed (default)

Step 3: Linking or Unlinking Objects

A list of objects which meet the selection criteria is displayed. These objects can be linked or unlinked to the keyword with the following commands in the CMD column:

L	link an object
U	unlink an object

```
13:27:31             *****  P R E D I C T  *****             2007-05-31
                          - Link/Unlink objects -

Keyword ID ....... HNO-KY

 CMD L Database                          Type              P-DBnr  Vista Parm.

  _  L HNO-DA-A                          Adabas           >>> now linked <<<
  _  U HNO-DA-D                          DB2              >>> now unlinked <<<
  _    HNO-DA-M                          RMS Handler          123  Local
  _    HNO-DAX                           DB2
  _    HNO-DA1                           Adabas               134  Local
  _  L HNO-H                             Gen. SQL Handler     111  Local
```

Objects already linked to the keyword are marked with L in the L column.

If the parameter Stay after modify is set to Y, the message >>> now linked <<< or >>> now unlinked <<< is issued to notify successful execution of the function (as shown above).

If the parameter Stay after modify is set to N, Predict immediately displays the next page of the selection list (if any) or skips back to the previous Link/Unlink objects screen.

43 Keyword Retrieval

- Keyword-specific Retrieval functions 368
- Layout of Keyword Lists 368
- Output Options for Keyword Retrieval 369

Keyword-specific Retrieval functions

Note: Standard retrieval functions are described in the section *Retrieval* in the *Predict Reference* documentation.

List Keywords Related to no Object - Code Y

This function lists keywords that are not assigned to any objects. Command: `UNUSED KEYWORD`.

Cross Reference Keywords - Code X

Lists all objects that have specified keywords.

Command: `XREF KEYWORD`

Valid output mode: Cross reference.

Layout of Keyword Lists

```
13:32:09            *****  P R E D I C T  *****              2007-05-31
                             - List Keyword -

--------------------------------------------------------------------------------
Cnt  Keyword ID                              No. of ref

  57 CHD-SQLDS
  58 COO                                          7
  59 DATAMODEL-BUSINESS-PARTNER                  14
  60 DEMO-VERSION                                 2
  61 DEMONSTRATION
  62 DEMONSTRATION2
```

Meaning of Columns	
No. of Ref.	Number of objects to which the keyword is assigned.

Output Options for Keyword Retrieval

The output options valid for this object type are identical to those for object type Extract. See also *Output Options for Extracts*.

IX

- 44 Library Structure 373
- 45 Method 379
- 46 Network 383
- 47 Node 391
- 48 Packagelist 395

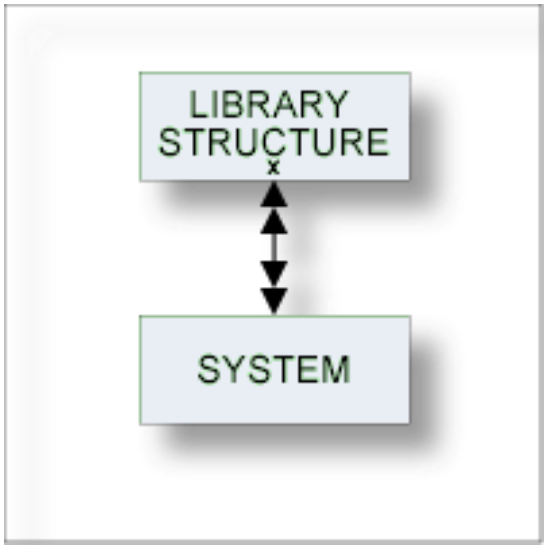
44 Library Structure

- Library Structure Maintenance Menu 374
- Add/Copy/Modify Library Structure Screen 375
- Library Structure-Specific Maintenance 376
- Library Structure Retrieval 377

Programs that are called by another program are not necessarily in the same library as the calling program: it is possible that they are loaded from a steplib at runtime. An object of type Library Structure documents a structure which describes a runtime or development environment (for example libraries for copy code). The corresponding systems are linked as child objects of type System to the library structure.

In the predefined Predict metastructure, a library structure can have active and passive associations of the following types:

Valid passive association:	No predefined association
Valid active association:	"Contains SY" (default child)



See also section *Steplib Support* in the *Predict Reference* documentation for more information.

Library Structure Maintenance Menu

This menu is called with function code M and object code LS in a Predict main menu or with command `MAINTAIN LIBRARYSTRUCTURE`.

```
13:31:50             *****  P R E D I C T  *****             2007-05-31
Plan   3           - (LS) Library structure Maintenance -          Profile HNO

  Function                                Function

  A  Add a Library structure              D  Display Library structure
  C  Copy Library structure               L  Link children
  M  Modify Library structure             S  Select Library structure from list
  N  Rename Library structure
  P  Purge Library structure


Function ..............

Library structure ID ..                                   Attributes....*
Copy ID ...............


Restrictions .........*   Profile HNO,used               Association ...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameter

The Library Structure Maintenance menu contains only global attributes. These are described in the section ***General Information*** in this documentation.

These functions are described in the section *Maintenance* in the *Predict Reference* documentation. The function Link children (with association "Contains SY") is described in this section. See ***Function Link Children - Code L***.

Add/Copy/Modify Library Structure Screen

The following screen is called for functions Add/Copy/Modify Library Structure:

```
09:38:53             *****  P R E D I C T  *****            2007-05-31
                          - Add a Library structure -
Library structure HNO-LS

Keys ..                                                          Zoom: N



  Abstract    Zoom: N









  Additional attributes ..* N          Associations ..* N
```

Parameters

The parameters are described under *Global Attributes*.

Library Structure-Specific Maintenance

Function Link Children - Code L

Note: The following description applies to children of type System linked via "Contains SY".

The link list of the library structure contains the main library and the steplibs. The following rules apply:

- The first entry in the link list is the main library, the following entries are steplibs.
- Dummy objects and systems without an implementation pointer for Library are permitted in the link list, but these objects are ignored when the library structure is evaluated for active retrieval function Program using programs and all LIST XREF functions.

Library Structure Retrieval

All retrieval functions for library structures are described in the section *Retrieval* in the *Predict Reference* documentation.

Output Options for Library Structure Retrieval

The output options available for this object type are identical to those for object type Dataspace. See ***Output Options for Dataspace Retrieval***.

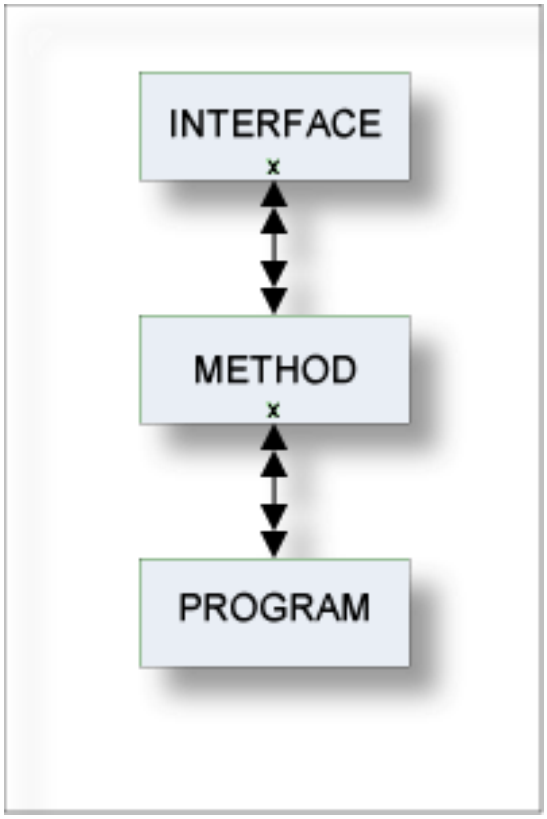
45 Method

- Method Maintenance Menu 380
- Add/Modify a Method Screen 381
- Method Retrieval 382

This object type is used to document the methods of an interface.

In the predefined Predict metastructure, a method can have passive and active associations of the following types:

Valid passive association:	"Belongs to IE" "Implemented in PR"
Valid active association:	no predefined association



Method Maintenance Menu

This menu is called with function code M and object code MD in a Predict main menu, or with the command `MAINTAIN METHOD`.

```
13:18:41             *****  P R E D I C T  *****             2007-05-31
Plan   0                - (MD) Method Maintenance -             Profile SYSTEM

  Function                                Function

  A  Add a Method                         D  Display Method
  C  Copy Method                          L  Link children
  M  Modify Method                        S  Select Method from list
  N  Rename Method
  P  Purge Method


Function ..............

Method ID .............                                   Attributes....*
Copy ID ...............
Belongs to IE .........

Restrictions .........*   Profile Default,empty         Association ...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters

The Method Maintenance menu contains only global attributes. See *Global Attributes*.

The functions are described in the section *Maintenance* in the *Predict Reference* documentation.

Add/Modify a Method Screen

The following screen appears for the function Add a Method. The screens for functions Copy and Modify are similar.

```
13:21:30             *****  P R E D I C T  *****             2007-05-31
                               - Add a Method -
Method ..........
Belongs to IE ..*
Keys ..                                                            Zoom: N

Attributes
  Method name ......
  Abstract    Zoom: N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Method	ID of the method.
Method name	Name of the method.

Method Retrieval

Information on method objects is gathered using standard retrieval functions. See the section *Retrieval* in the *Predict Reference* documentation.

Output Options for Method Retrieval

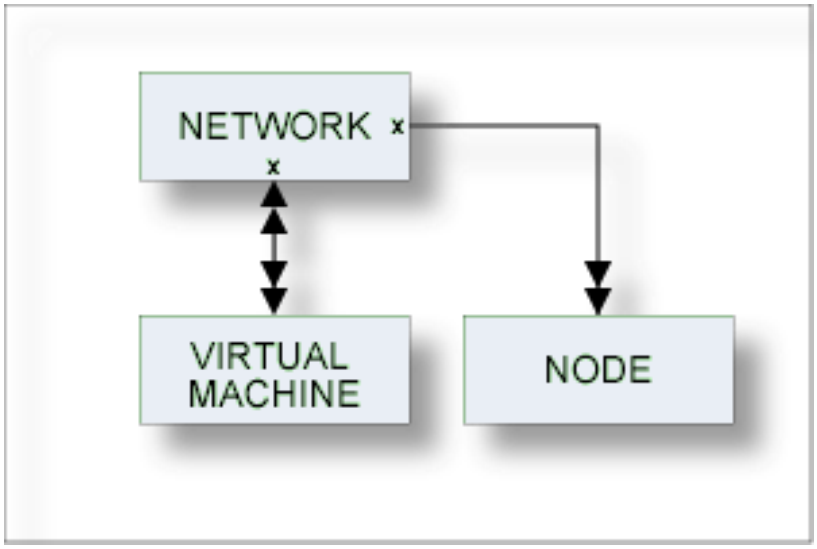
The output options valid for this object type are identical to those for object type Dataspace. See *Output Options for Dataspace Retrieval*.

46 Network

- Network Maintenance Menu 385
- Add a Network Screen 385
- Network-Specific Maintenance 386
- Network Retrieval 386

In the predefined Predict metastructure, a network can have passive and active associations of the following types:

Valid passive associations:	no predefined association
Valid active associations:	"Uses VM" "Contains NO"



The location of a database must be specified by linking each database to an object of type Virtual Machine and each virtual machine to an object of type Network. A current network can be defined in the Miscellaneous section of the General defaults of Predict or with the command `SET VM virtual-machine-ID`.

The current network will be taken as default for virtual machine objects if no network is specified.

Links between virtual machines and networks are established by entering the network in the parameter "Belongs to NW" of the virtual machine. A virtual machine cannot be linked via "Belongs to NW" to a network using the link editor.

See *Defining the Distribution of Data in Predict* in the section *Adabas Vista* in the *Predict and Other Systems* documentation for a description of how to define the distribution of data.

Network Maintenance Menu

The Network Maintenance menu is called with function code M and object code NW in a Predict main menu, or with the command `MAINTAIN NETWORK`.

```
13:13:21            *****  P R E D I C T  *****              2007-05-31
 Plan   3              - (NW) Network Maintenance -            Profile SYSTEM

  Function                                Function

  A  Add a Network                        D  Display Network
  C  Copy Network                         L  Link children
  M  Modify Network                       S  Select Network from list
  N  Rename Network
  P  Purge Network


Function ..............

Network ID ............                                   Attributes ....*
Copy ID ...............



Restrictions .........*   Profile Default,empty          Association ...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

The parameters are described under ***Global Attributes***.

All functions are described in the section *Maintenance* in the *Predict Reference* documentation. The function Purge Network is described in the section ***Purge Network - Code P***.

Add a Network Screen

The following screen is displayed for the Add a Network function. The Copy and Modify screens are similar.

```
13:06:57             *****  P R E D I C T  *****            2007-05-31
                             - Add a Network -
Network ......... HNO-NW

Keys ..
Zoom: N



Abstract    Zoom: N
```

The parameters are described under ***Global Attributes***.

Network-Specific Maintenance

Standard maintenance functions are used for maintaining networks. These are described in the section *Maintenance* in the *Predict Reference* documentation.

The special rules applying to function Purge Network are described below.

Purge Network - Code P

The following rules apply:

- A network that is linked to a virtual machine via "Uses VM" cannot be deleted.
- The network defined as current network in the General defaults cannot be deleted.

Network Retrieval

Standard retrieval functions are described in the section *Retrieval* in the *Predict Reference* documentation. The network-specific function Vista Numbers is described below.

Vista Numbers - Code N

Displays information on the use of Vista numbers in list form (see **sample output** below).

```
13:13:21            *****  P R E D I C T  *****             2007-05-31
Plan   0                - (NW) Network Retrieval -             Profile SYSTEM

 Retrieval Type                            Retrieval Type

 D  Network                                B  Network with parents
 E  Execu +------------------- Search criteria ----------------------+
 C  Dummy !                                                          !
 N  Vista ! Start value                                              !
          !   Vista DBnr ..... 0     Fnr ..... 0                     !
          !                                                          !
Retrieval ! End value                                                !
Output-mo !   Vista DBnr ..... 65535 Fnr ..... 32767                 !
          !                                                          !
Network I +----------------------------------------------------------+


Restrictions .........*   Profile Default,empty           Model..........*
Output options .......*   Profile Default                 Association....*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters		
Network ID	Restricts the report to Vista numbers used in the given network.	
Start value / End value	Restricts the report to Vista numbers in the given range.	
Vista number	A window appears for specifying Start value and End value. Valid values:	
	0 - 65535	for Vista DBnr.
	0 - 32767	for Vista Fnr.

Sample Output

```
 09:53:36             *****  P R E D I C T  *****             2007-05-31
                           - List Vista Numbers -

Network ID ......... HOME
--------------------------------------------------------------------------------
                                                          Vista
Cnt  Object ID                        Object type   DBnr      Fnr  Environment

    1 HEB-DA-MIG                       Database        2        0
      Type ... Adabas ,Isolated                     .... Thru ...
      reserved Vista numbers                          2    65535
    2 BOE-FI03                         FI Vista el.    4        2
    3 HEB-EDT                          FI Vista el.    4        3
    4 BOE-FI07                         FI Vista el.    4        4
    5 BOE-FI-E-02                      FI Vista el.    4        5
```

Meaning of Columns		
Object ID	ID of the object referencing the Vista number. The following information on the object may be displayed.	
	Type ...	For databases: the Vista parameter of the database (Isolated, Local, Vista).
	reserved Vista numbers	For databases: a range of Vista numbers is reserved depending on the DBnr. Reserved range: 0 - 65535.
Object type	Type of object referencing the Vista number. Can be one of the following: ■ Database ■ Phys. file ■ FI Vista el. ■ DA Vista el.	
DBnr / Fnr	Database and file number identifying the file uniquely in a network.	
Environment	Environment specified with the Vista element to restrict access to the data.	

Layout of Network Lists

Network lists contain the network IDs.

Output Options for Network Retrieval

The output options valid for this object type are identical to those for object type dataspace. See *Output Options for Dataspace Retrieval*.

47 Node

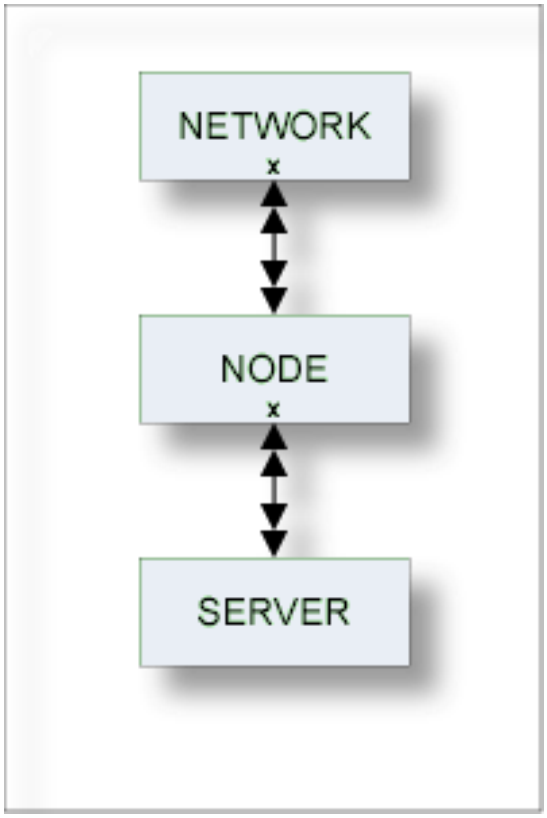
- Node Maintenance Menu 392
- Add a Node Screen 393
- Node Retrieval 394

This object type, together with object type Server, is used to document remote procedure calls.

An object of type Node documents the physical machine containing the server.

In the predefined Predict metastructure, a node can have passive and active associations of the following types:

Valid passive association:	"Contained in NW"
Valid active association:	"Contains SV"



Node Maintenance Menu

This menu is called with function code M and object code NO in a Predict main menu, or with the command `MAINTAIN NODE`.

```
13:33:11             *****  P R E D I C T  *****             2007-05-31
Plan  10                 - (NO) Node Maintenance -                 Profile HNO

 Function                                Function

 A  Add a Node                           D  Display Node
 C  Copy Node                            L  Link children
 M  Modify Node                          S  Select Node from list
 N  Rename Node
 P  Purge Node


Function ..............

Node ID ...............                                   Attributes....*
Copy ID ...............
Contained in NW .......

Restrictions .........*   Profile HNO,used              Association ...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters

The Node Maintenance menu contains only global attributes. See ***Global Attributes***.

The functions are described in the section *Maintenance* in the *Predict Reference* documentation.

Add a Node Screen

The following screen appears for the function Add a Node. The screens for functions Copy and Modify are similar.

```
13:37:04             *****  P R E D I C T  *****            2007-05-31
                                - Add a Node -
Node ............. HNO-NO
Contained in NW .*
Keys ..                                                          Zoom: N


 Node name ........
 Abstract    Zoom: N



 Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters		
Node ID	ID of the node.	
Contained in NW	ID of the parent network.	
Node name	Name of the node. Up to 8 characters.	
Associations: Contains SV	Y	Edit the "Contains SV" server list. An asterisk in front of this field indicates that a "Contains SV" server list for this node exists.

Node Retrieval

Information on node objects is gathered using standard retrieval functions. See the section *Retrieval* in the *Predict Reference* documentation.

Output Options for Node Retrieval

The output options valid for this object type are identical to those for object type dataspace. See *Output Options for Dataspace Retrieval*.

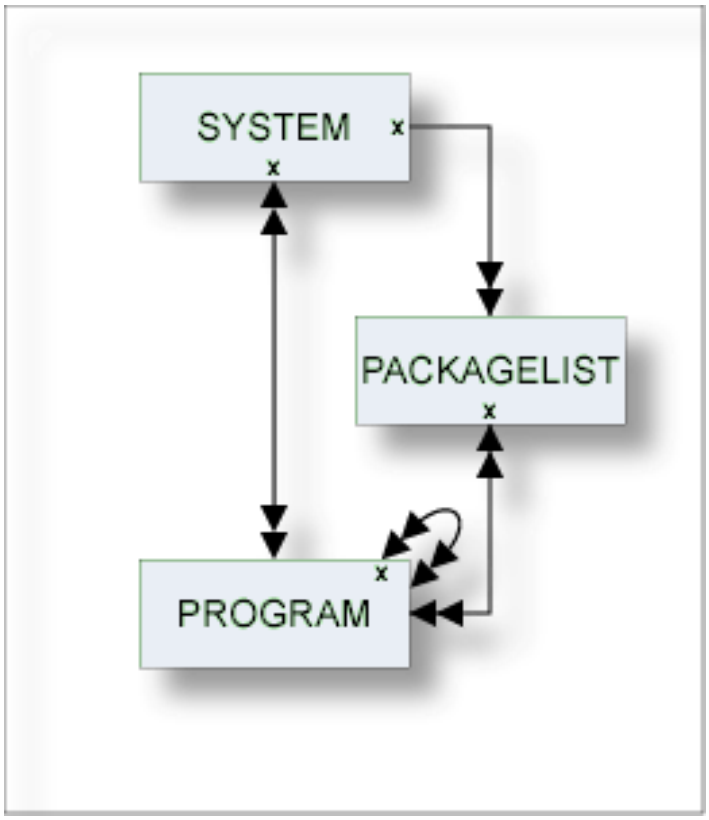
48 Packagelist

- Packagelist Maintenance Menu 396
- Packagelist Types 398
- Add a Packagelist Screen 398
- Packagelist-Specific Maintenance 399
- Packagelist Retrieval 400

The Predict object type Packagelist is used to document DB2 packages.

In the predefined Predict metastructure, a packagelist can have passive and active associations of the following types:

Valid passive association:	"Contained in SY"
Valid active association:	"Uses PR"



Note: Packagelists of type T and packagelists of type S are related using the parameters Collection name and Location name.

Packagelist Maintenance Menu

The following screen is displayed with function code M and object code PG in a Predict main menu or the command `MAINTAIN PACKAGELIST`.

```
13:47:47              *****  P R E D I C T  *****            2007-05-31
Plan   0               - (PG) Packagelist Maintenance -            Profile HNO

 Function                                Function

 A  Add a Packagelist                    D  Display Packagelist
 C  Copy Packagelist                     L  Link children
 M  Modify Packagelist                   S  Select Packagelist from a list
N  Rename Packagelist
P  Purge Packagelist


Function .........

                                                      Attributes..........*
Packagelist ID ...                                    Packagelisttype ....*
Copy ID ..........
Contained in SY ..

Restrictions .....*   Profile HNO,used                Association.........*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Menu  Canc  S-fi  E-el  M-pr  Print Impl  Last  FLIP  PROF  Next
```

Parameters	
Function	Executes one of the maintenance functions. Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. The function Purge is described in the section ***Purge Packagelist - Code P***.
Packagelist ID	Identifier of the Predict packagelist object. See ***Naming Conventions***.
Packagelist type	See ***Packagelist Types*** below.
Copy ID	For Copy function: ID of the packagelist to be created.
Contained in SY	In DB2, packagelists are used by application plans. Applications plans are documented in Predict with objects of type system, subtype P. Hence the attribute "Contained in SY" is used to document by which plan a packagelist is used.
Association	For function Link children: Objects are to be linked to the packagelist via the selected association. Valid values: "Uses PR" and user-defined.

Packagelist Types

The table below contains a list of all valid packagelist types.

Code	Packagelist Type
Q	Database request module (DBRM). Packagelists of type Q contain one DBRM which is directly bound to the plan.
S	Subcollection. Packagelists of type T and packagelists of type S are connected using the parameters Collection name and Location name. Each package in a packagelist of type S is also contained in a packagelist of type T.
T	Total collection. Packagelists of type T provide an overview of all packages used in a collection. The parameters Collection name and Location name are mandatory for packagelists of type T.

Add a Packagelist Screen

The screen is displayed for the Add a Packagelist function. The Copy and Modify screens are similar.

```
09:45:26            *****  P R E D I C T  *****            2007-05-31
                           - Add a Packagelist -
Packagelist ID .. HNO-PG
Type ...........*
Contained in SY *
Keys ..                                                        Zoom: N

Packagelist attributes
  Collection name ...
  Location name .....

Abstract     Zoom: N









Additional attributes ..* N          Associations ..* N
```

Predict ensures the consistency of related packagelists (types T and S):

- If a package is purged from a packagelist of type T, it is purged automatically from corresponding packagelists of type S.
- If a package is added to a packagelist of type S, it is added automatically to the corresponding packagelist of type T.

Note: Parameters not listed here are described with the Packagelist Maintenance menu in the section *Packagelist Maintenance Menu*.

Parameters	
Packagelist attributes	
Collection name	From version 2.3 of DB2 and above, packages are always referenced via collections. A collection is a virtual summary of packages, used to simplify references to packages. In Predict, collections are documented as attributes of packagelists. Packagelists are grouped by including several packages to the same collection. A collection is documented in Predict with the attributes collection name and location name. A collection name can be up to 18 characters long.
Location name	Together with collection names, location names identify collections uniquely. A location name can be up to 16 characters long.
Association: Uses PR	Y Edit "Uses PR" list of the packagelist. Programs of the following types can be linked to packagelists via "Uses PR": ■ Program (type P) ■ Subprogram (type N) ■ Function (type F) The Predict Link Editor is invoked. See the section *Editors in Predict* in the *Predict Reference* documentation.

Packagelist-Specific Maintenance

Purge Packagelist - Code P

The following rules apply:

- If you confirm this function with `DELETE`, the following objects are deleted:
  - the packagelist object
  - all links to child objects
  - all links from parent objects
- With packagelists of type T, all packagelists of type S connected to the packagelist via the attributes Collection/Location name are deleted as well. You must enter an additional confirmation before deleting these additional objects.

Packagelist Retrieval

Packagelist-specific Retrieval Parameter

"Contained in SY". System to which the packagelist is linked.

Layout of Packagelist Lists

```
09:50:10             *****  P R E D I C T  *****             2007-05-31
                            - List Packagelist -


--------------------------------------------------------------------------------
Cnt  Packagelist ID                    T Collection         Location

   1 AMA-PG1                           T DSDS               ERE
   2 AMMMM                             T CVXCV              XCVXC
   3 ARH-PA-1                          T COLL               LOC
   4 ARH-PA-2                          T COL                LOC
   5 BA-PG                             T JKJ                KJKKK
```

Meaning of Columns	
T	Type of packagelist: Q — DBRM T — Total collection S — Subcollection
Collection	Collection of the packagelist. Packagelists of type T and of type S that belong together have the same collection and location name.
Location	Location of the packagelist. Packagelists of type T and of type S that belong together have the same location and collection name.

Output Options for Packagelist Retrieval

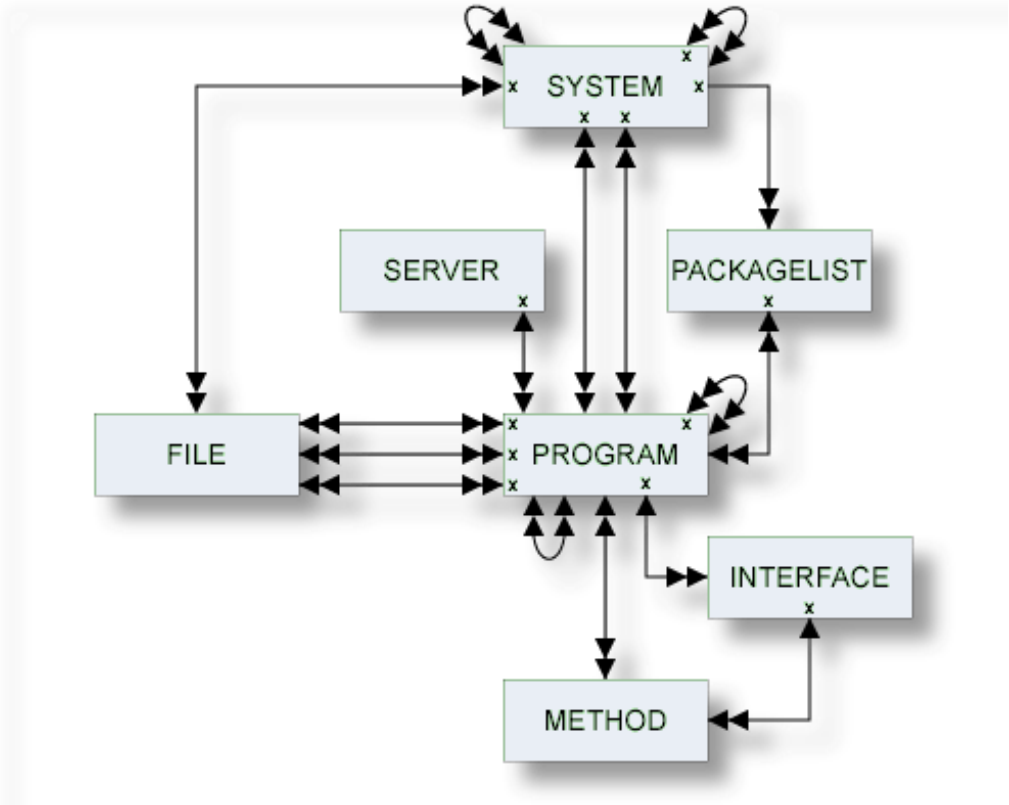
The output options valid for this object type are identical to those for object type dataspace. See *Output Options for Dataspace Retrieval*.

X Program

Predict knows more than a dozen different types of programs, ranging from parameter data area to Natural Expert model. About a dozen different programming languages are supported.

In the predefined Predict metastructure, a program can have passive and active associations of the following types:

Valid passive associations:	"Contained in PG" "Used by PR" "Belongs to SY" (default passive association) "Is comp. of SY" "Used by SV" "Implements MD"
Valid active associations:	"Uses FI concept." (first default active association) "Defines Interface" "Input File" "Invokes Method" "Has message PR" "Uses PR concept." (second default active association) "Returns File"



The description of object type Program is organized under the following headings:

Maintaining Objects of Type Program
Defining Additional Attributes of Programs
Program Specific Maintenance
Program Retrieval

49 Maintaining Objects of Type Program

- Program Maintenance Menu 404
- Program Types 405
- Languages 406
- Program-Specific Libraries 407
- Add a Program Screen 407

Program Maintenance Menu

The Program Maintenance menu is displayed with function code M and object code PR in a Predict main menu or with the command `MAINTAIN PROGRAM`.

```
17:37:24              *****  P R E D I C T  *****              2007-05-31
Plan   0                 - (PR) Program Maintenance -             Profile SYSTEM

 Function                                Function

 A  Add a program                        D  Display program
 C  Copy program                         L  Link children
 M  Modify program                       S  Select program from a list
 N  Rename program                       X  Redocument program
 P  Purge program                        J  Physical Attributes

Function .........
Program ID .......                                  Attributes .........*
Copy ID ..........                                  Program of type ....*
Belongs to SY ....                                  Language ...........*

Member ...........                                  User system Fnr .....
Library ..........                                  User system DBnr ....
Restrictions ....*   Profile Default ,used          Association ........*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main   ↩
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Executes one of the maintenance functions. Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. Program-specific functions are described in *Program Specific Maintenance*.
Program of type	For a list of valid values, see *Program Types* below. For the Select function: Program type can be specified as an additional selection criterion. For the Add and Copy function: Program type of the new object. Value is passed to the Add/Copy Program screen.
Language	For a list of valid values, see *Languages* below. For the Select function: language can be specified as an additional selection criterion.

Parameters	
	For the Add and Copy function: language of the new object. Value is passed to the Add/Copy Program screen.
Member, Library, User system Fnr / DBnr	
For the Select function: implementation pointer values can be used to restrict the scope of objects to be processed. Only those Predict program objects will be processed that document members meeting the specified Member/Library/Fnr/DBnr parameters.	
Member	Member documented by the Predict program.
Library	Library in which the member is stored. Either a Natural library or one of the libraries listed under *Program-Specific Libraries* can be specified.
User system Fnr	Number of the user system file.
User system DBnr	Number of the database in which the user system file is implemented.

Program Types

The table below contains a list of all valid program types.

Code	Program Type
A	Parameter data area
C	Copy Code
D	Documented program
E	External program
F	Function
G	Global data area
H	Help routine
I	Dynamic (see *Programs of Type dynamic*)
J	Job
K	ISPF Macro
L	Local data area
M	Map/Help map
N	Subprogram
O	Natural command processor
P	Main program
R	SQL procedure
S	Natural subroutine
T	Dialog
U	Database function

Code	Program Type
X	Text
Y	Expert Model
1	Error Message
4	Class
5	Resource
8	Adapter
blank	Undefined

Languages

The table below contains a list of all valid languages.

Code	Language
B	BAL (Assembler)
C	COBOL
E	Natural EL
F	FORTRAN
G	ADA
H	C
J	Job Control Language
N	Natural
O	Other
P	PL/I
Q	Static SQL
R	REXX
S	SQL Procedure Language
V	Java
Z	System Program, see *System Programs*
0	Language 0
1	Language 1
2	Language 2
3	Language 3
blank	Unknown

Program-Specific Libraries

The table below contains a list of all valid program-specific libraries.

Library Name	Description
SYSADA	for ADA
SYSBAL	for ASSEMBLER
SYSCCC	for C
SYSCOB	for COBOL
SYSFOR	for FORTRAN
SYSPLI	for PL/I
SYSSTA	for Static SQL
SYSSYS	for system programs
user-defined	library of a 3GL application environment; must be documented in an object of type System

Add a Program Screen

The screen is displayed for the Add a Program function. The Copy and Modify screens are similar.

```
17:39:17             *****  P R E D I C T  *****              2007-05-31
                             - Add a Program -
Program ID ...... HNO-PR
Type ...........* P  Program
Belongs to SY ..*
Keys ..                                                          Zoom: N

Program attributes
  Language .....*    All
  Mode .........*    (none)
  Load-Lib ......
Implementation  pointer
  Member ........               User system Fnr ...
  Library .......               User system DBnr ..
  NAT-Func ......
          ('*' to get NAT-Function name from Xref data)

Abstract     Zoom: N               Authors     Zoom: N




  Additional attributes ..* N          Associations ..* N                    ↩
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Program ID	The ID of the program object.
Type	Program type. Must suit the language. The language can be left blank (undefined) for any program type. Enter an asterisk for list of possible values. See *Overview of Language-Specific Program Types* for a table of valid program type/language combinations.
Belongs to SY	ID of the system to which the program is linked. If the program is connected to more than one system, >>>multiple<<< is displayed in this field in the Modify Program function and the field is protected.
Program Attributes	
Language	The language in which the program is written. Enter an asterisk for a list of possible values. See *Overview of Language-Specific Program Types* for a table of valid program type/language combinations.
Mode	Mode of operation in which the program is used. A — All (both online and batch modes) B — Batch mode O — Online blank — Undefined
Load-Lib	The load library.
Implementation Pointer	
Member	Member documented by the Predict program (not applicable to programs of type 5).

Library	The name of the library in which the member is stored (not applicable to programs of type D).
	■ For Natural programs: see the table in *Overview of Language-Specific Program Types*.
	■ For 3GL programs:
	  ■ one of the standard 3GL libraries (see *Program-Specific Libraries*),
	  ■ any library, provided that it is documented in a Predict system object of type G.

User system Fnr	The number of the user system file. For 3GL programs, the number is always 255.
User system DBnr	The number of the database in which the user system file is located. For 3GL programs, the number is always 255.
NAT-Func	Applicable only to Natural subroutines (type S). The name of the function of the subroutine (DEFINE SUBROUTINE). If an asterisk is entered, Predict derives the function name from XRef data if XRef data exists for the specified member.

50 Defining Additional Attributes of Programs

- Programs of Type Class 411
- Programs of Type Resource 412
- Programs of Type SQL Procedure 413
- Programs of Type Database function 420
- System Programs 423
- Programs of Type dynamic 423

If Additional Attributes is set to Y, a window is displayed which contains the following additional attributes for selection:

- **Description**
  A description of the program can be entered.
- **Owner**
  The owner of the program can be entered.
- **Class definition**
  See *Programs of Type Class*.
- **Resource definition**
  See *Programs of Type Resource*.
- **Database function options**
  See *Programs of Type Database function*.
- **Procedure options**
  See *Programs of Type SQL Procedure*.
- **Entry points**
  Entry points are to be modified. This is valid only for programs written in certain languages. See *Overview of Language-Specific Program Types*.
- **SQL procedure code**
  Only for programs of type R and language S. The SQL Procedure Editor is called.

The following rules apply:

- Only those types of additional attributes appear in the window that apply to the type of program. For example: the option Class definition is not contained in the list when a program of type Resource is processed.
- More than one choice can be made at a time. The respective input maps are then displayed one after the other.

Programs of Type Class

```
13:33:43             *****  P R E D I C T  *****              2007-05-31
                             - Modify Program -
Program ID ...... HNO-CLASS                        Modified 2007-05-31 at 13:31
                                                         by HEB


Class definition
  Name .............
  GUID .............
  Version ..........









* Additional attributes ..* N        * Associations ..* N
```

Parameters	
Class definition	
Name	The name of the class.
GUID	The globally unique ID of the class.
Version	The version number of the class.

Programs of Type Resource

```
17:45:19             *****  P R E D I C T  *****              2007-05-31
                             - Modify Program -
Program ID ...... HNO-PR                              Added 2007-05-31 at 17:39
                                                         by HNO



Resource definition
  File name ........




  Library ..........
  User system Fnr ..
  User system DBnr .









  Additional attributes ..* S          Associations ..* N
```

Parameters	
Resource definition	
File name	File name documented by the Predict program.
Library	The name of the library in which the file name is stored.
User System Fnr	The number of the user system file.
User System DBnr	The number of the database in which the user system file is located.

The type of Resource can be documented in the language field of a Predict program object. There is a user exit program U-PGMLAN that allows dynamic extension of possible languages in each installation.

Programs of Type SQL Procedure

```
14:51:00             *****  P R E D I C T  *****                    2009-07-31
                              - Add a Program -
Program ID ...... SQL-PROC                           Added 2009-06-10 at 14:50
                                                   Modified
Procedure option
  Procedure name ..
Physical attributes in DADB29                           (new)
  Schema name .....
  Specific name ...
  Collection .....*   (none)             Parameter style *   (none)
  WLM environment *   (none)             Parameter CCSID * E EBCDIC
  Dyn. result set . 2                    Spec. register .*   (none)
  Deterministic ... N (Y,N)              Fenced .........* Y Fenced
  Null input ...... Y (Y,N)              SQL data .......* R Reads SQL data
  DB info ......... N (Y,N)              Asutime .........
  Stay resident ... N (Y,N)              Program type ...*   (none)
  Security .......* D DB2                Commit ..........   (Y,N)
  After failure ..* D Stop (system defau Debug mode .....* I Disable
  Run options ..... run prcc ccsid E failure D                       * Zoom: N
  Packagepath ....*   (none)                                           Zoom: N


* Additional attributes ..* S          Associations ..* N
```

Parameters	
Procedure name	This name must comply with SQL naming conventions. See the section *Naming Conventions for SQL Objects* in the section *Adabas D and Other SQL Systems* in the *Predict and Other Systems* documentation.
Schema name	Used as a qualifier for an unqualified procedure name.
Specific name	Specifies a unique name for the procedure.
Collection	Identifies the package collection. N — NO COLLID Y — Use collection-ID. A collection-ID must then be specified. blank — none
WLM environment	Identifies the MVS workload manager application environment.
Dyn. result set	Specifies the maximum number of query result sets that the stored procedure can run.
Deterministic	Specifies whether the procedure returns the same results for identical arguments. Y — Yes N — No blank — undefined

Parameters		
Null input	Specifies whether the procedure is called if any of the input arguments is null at execution time.	
	Y	Yes
	N	No
DB info	Specifies whether specific information that is included in DB2 is passed to the procedure when it is invoked.	
	Y	Yes
	N	No
Stay resident	Specifies whether the load module for the procedure remains resident in memory when the procedure ends.	
Security	Specifies how the procedure interacts with an external security product.	
	D	DB2
	F	Definer
	U	User
	blank	none
After failure	Specifies the action to be taken after a failure has occured. Valid values:	
	D	Stop (system default). Stops after number of failures defined in the system defaults.
	N	Stop (number). Stops after number of failures defined by the user. A numeric value must be added.
	C	Continue.
	blank	none
Run options	Specifies the language environment run-time options to be used for the procedure.	
Packagepath	Specifies the package path to use when the procedure is run	
	N	No packagepath.
	Y	Use packagepath. A list of package collections must be specified.
	blank	none
Parameter style	Identifies the linkage convention use to pass parameters to the procedure.	
	D	DB2SQL
	G	General
	N	General with nulls
	J	Java
	blank	none
Parameter CCSID	Specifies the encoding scheme. Valid values:	
	blank	none

Parameters	
	A — ASCII E — EBCDIC U — Unicode
Spec. register	Valid values: I — Inherit. The values of special registers are inherited. D — Default. Special registers are initialized to the default values. blank — none
Fenced	Determines that the external procedure runs in an external address space. Y — Yes N — No
SQL data	Indicates whether the procedure can execute any SQL statements. M — Modifies SQL data N — No SQL R — Read SQL data S — Contains SQL blank — none
Asutime	Specifies the total amount of processor time.
Program type	Specifies whether the procedure runs as a main or a subroutine. S — Sub M — Main blank — none
Commit	Indicates whether DB2 commits the transaction immediately on return from the procedure. Y — Yes N — No
Debug Mode	Specifies whether the procedure can be run in debugging mode. The default is D (Disallow) when no Dynamic rules run behavior is in effect. Valid values: D — Disallow. A — Allow. I — Disable. blank — None.

- Native SQL Procedure

Native SQL Procedure

```
14:24:06             *****  P R E D I C T  *****                    2009-07-31
                             - Modify Program -
Program ID ...... HIB-R-S-GEN-NATIVE                 Added 2009-07-07 at 13:19
                                                   Modified 2009-07-10 at 15:17
Physical attributes in DADB29
  Native .......... Y (Y,N)
  Version ......... V29                                                Zoom: N
  Package owner ... Owner                Release at .....*   (none)
  Prepare ........* D Defer              REOPT ..........*   (none)
  Current data ....   (Y,N)              Validate .......*   (none)
  Degree ........ *   (none)             Rounding .......*   (none)
  Dynamic rules ..*   (none)             Date format ....*   (none)
  Appl. encoding .*   (none)             Decimal ........*
  Explain .........   (Y,N)              For update .....*   (none)
  Immediate write .   (Y,N)              Time format ....*   (none)
  Isolation level *   (none)
  Keep dynamic ....   (Y,N)
  Optimiz. hints .. opt 29 ----------------------------------------- * Zoom: N
  SQL path ........ path 29                                          * Zoom: N



* Additional attributes ..* S          Associations ..* N                    ↩
```

Valid values are:

Parameters		
Native	To indicate if the definition is for a native SQL procedure.	
	Y	Yes.
	N	No. This is the default.
	blank	none
Version	Specifies the procedure version identifier. The default is V1.	
Package owner	Specifies the owner of the package.	
Prepare	Specifies whether to defer preparation of dynamic SQL statements that refer to remote objects, or to prepare them immediately.	
	D	Defer.
	N	Nodefer.
	blank	Not specified. This is the default.
Current data	Specifies whether to require data currency for read-only and ambiguous cursors when the isolation level of cursor stability is in effect.	

Parameters		
	Y	Yes.
	N	No. This is the default.
	blank	none
Degree	Specifies whether to attempt to run a query using parallel processing to maximize performance.	
	1	One. This is the default.
	A	Any.
	blank	Not specified.
Dynamic rules	Specifies the values that apply, at run time, for the following dynamic SQL attributes:	
	R	Run. This is the default.
	B	Bind.
	D	Definebind.
	E	Definerun.
	I	Invokebind.
	N	Invokerun.
	blank	Not specified.
Appl. encoding	Specifies the default encoding scheme for SQL variables in static SQL statements in the routine body.	
	A	ASCII.
	E	EBCDIC.
	U	Unicode.
	blank	Not specified.
Explain	Specifies whether information will be provided about how SQL statements in the routine will execute.	
	Y	Yes.
	N	No. This is the default.
	blank	Not specified.
Immediate write	Specifies whether immediate writes are to be done for updates that are made to group buffer pool dependent page sets or partitions.	
	Y	Yes.
	N	No. This is the default.
	blank	Not specified.
Isolation level	Specifies how far to isolate the routine from the effects of other running applications.	
	C	Cursor stability.
	S	Read stability.

Parameters		
	R	Repeatable read.
	U	Uncommitted read.
	blank	Not specified.
Keep dynamic	Specifies whether DB2 keeps dynamic SQL statements after commit points.	
	Y	Yes.
	N	No. This is the default
	blank	Not specified.
Optimiz. hints	Specifies query optimization hints.	
SQL path	Specifies the SQL path.	
Release at	Specifies when to release resources that the procedure uses: either at each commit point or when the procedure terminates.	
	C	Commit. This is the default.
	D	Deallocate.
	blank	Not specified.
REOPT	Specifies if DB2 will determine the access path at run time by using the values of SQL variables or SQL parameters, parameter markers, and special registers.	
	N	None. This is the default.
	A	Always.
	O	Once.
	blank	Not specified.
Validate	Specifies whether to recheck, at run time, errors of the type `OBJECT NOT FOUND` and `NOT AUTHORIZED` that are found during bind or rebind.	
	R	Run. This is the default.
	B	Bind.
	blank	Not specified.
Rounding	Specifies the desired rounding mode for manipulation of DECFLOAT data.	
	C	Ceiling.
	D	Down.
	F	Floor.
	1	Half down.
	2	Half even.
	3	Half up.
	U	Up.
	blank	Not specified. This is the default.

Parameters		
Date format	Specifies the date format for result values that are string representations of date or time values.	
	I	ISO.
	E	EUR.
	U	USA.
	J	JIS.
	L	Local.
	blank	Not specified. This is the default.
Decimal	Specifies the maximum precision that is to be used for decimal arithmetic operations.	
	15	
	15,*n* (where *n* must be a number between 1 and 9)	
	31	
	31,*n* (where *n* must be a number between 1 and 9)	
	blank	Not specified. This is the default.
For update	Specifies whether the `FOR UPDATE` clause is required for a `DECLARE CURSOR` statement if the cursor is to be used to perform positioned updates.	
	R	Required. This is the default.
	O	Optional.
	blank	Not specified.
Time format	Specifies the time format for result values that are string representations of date or time values.	
	I	ISO.
	E	EUR.
	U	USA.
	J	JIS.
	L	Local.
	blank	Not specified. This is the default.

Programs of Type Database function

```
14:56:35             *****  P R E D I C T  *****                    2009-07-31
                              - Add a Program -
Program ID ...... HNO-FUNC                          Added 2009-06-10 at 14:56
                                                  Modified
Function option
  Function name ...
Physical attributes in DADB29                          (new)
  Function type ..* S Scalar
  Schema name .....
  Specific name ...
  Collection .....*                       After failure .*   (none)
  WLM environment .                       Style .........*   (none)
  Spec. register .*   (none)              CCSID .........* A ASCII
  Deterministic ... N (Y,N)               Fenced ........*   (none)
  Null input ......   (Y,N)               SQL data ......* N No SQL
  External action . Y (Y,N)               Scratchpad .....
  Final call ......   (Y,N)               Allow parallel .   (Y,N)
  DB info .........   (Y,N)               Cardinality ....
  Asutime .........                       Stay resident ..   (Y,N)
  Program type ...*   (none)              Security ......*   (none)
  Run options .....                                                Zoom: N
  Packagepath ....*                                                Zoom: N
* Additional attributes ..* S          Associations ..* N                ↩
```

Parameters	
Function name	This name must comply with SQL naming conventions. See the section *Naming Conventions for SQL Objects* in the section *Adabas D and Other SQL Systems* in the *Predict and Other Systems* documentation.
Function type	The type of the function. S — Scalar T — Table
Schema name	Used as qualifier for an unqualified function name.
Specific name	Specifies an unique name for the function.
Collection	Identifies the package collection. N — NO COLLID Y — Use collection-ID. A collection-ID must then be specified. blank — none
WLM environment	Identifies the MVS workload manager application environment.
Spec. register	Valid values: I — Inherit. The values of special registers are inherited.

Parameters		
	D	Default. Special registers are initialized to the default values.
	blank	none
Deterministic	Specifies whether the function returns the same results for identical arguments.	
	Y	Yes
	N	No
	blank	undefined
Null input	Specifies whether the function is called if any of the input arguments is null at execution time.	
	Y	Yes
	N	No
	blank	undefined
External action	Specifies whether the function takes an action that changes the state of an object that DB2 does not manage.	
	Y	Yes
	N	No
	blank	undefined
Final call	Specifies whether final call is made to the function.	
	Y	Yes
	N	No
	blank	undefined
DB info	Specifies whether specific information that DB2 knows is passed to the function when it is invoked.	
	Y	Yes
	N	No
	blank	undefined
Asutime	Specifies the total amount of processor time.	
Program type	Specifies whether the function runs as a main or a subroutine.	
	S	Sub
	M	Main
	blank	none
Run options	Specifies the language environment run-time options to be used for the function.	
Packagepath	Specifies the package path to use when the function is run	
	N	No packagepath.
	Y	Use packagepath. A list of package collections must be specified.

Parameters		
	blank	none
After failure	Specifies the action to be taken after a failure has occured. Valid values:	
	D	Stop (system default). Stops after number of failures defined in the system defaults.
	N	Stop (number). Stops after number of failures defined by the user. A numeric value must be added.
	C	Continue.
	blank	none
Style	Specifies the conventions used for passing parameters to and returning the value from functions. Valid values:	
	D	DB2SQL
	J	Java
	blank	not specified
CCSID	Specifies the encoding scheme. Valid values:	
	blank	none
	A	ASCII
	E	EBCDIC
	U	Unicode
Fenced	Determines that the external function runs in an external address space.	
	Y	Yes
	N	No
	blank	undefined
SQL data	Indicates whether the function can execute any SQL statements.	
	M	Modifies SQL data
	N	No SQL
	R	Read SQL data
	S	Contains SQL
	blank	none
Scratchpad	Specifies whether DB2 provides a scratchpad for the function.	
Allow parallel	Specifies whether parallelism can be used.	
	Y	Yes
	N	No
	blank	undefined
Cardinality	Specifies an estimate of the expected number of rows that the function returns.	

Parameters		
Stay resident	Specifies whether the load module for the function remains resident in memory when the function ends.	
Security	Specifies how the function interacts with an external security product.	
	D	DB2
	F	Definer
	U	User
	blank	none

System Programs

Programs that are only available as object code and hence have no language are documented with programs of type E (external object) and language Z (system program). Predict creates XRef data for these so called system programs because neither the preprocessor nor Natural can create XRef data for object code.

The implementation pointer for a system program has to be specified explicitly. One entry point (with the ID of the program object) is created by Predict, additional entry points have to be specified manually.

Programs of Type dynamic

Programs of type dynamic are used to document calls of programs of the same name from different steplibs depending on the library structure. The following rules apply:

- Because programs of type dynamic document any number of implemented members, no check is performed as to whether the members documented by the program are actually implemented.
- With the active retrieval function Programs using programs, programs of type dynamic are ignored as current objects.
- Programs of this type can only have children for association "Uses PR concept".

51 Program-Specific Maintenance

- Editing Child Lists 426
- Overview of Language-Specific Program Types 426
- Generating Database Request Modules -DBRMs- from Program Objects of Language Q -Static SQL- 428
- Purge Program - Code P 429
- Redocument Program - Code X 429
- Edit procedure code of a program - Code Y 435

Standard maintenance functions are described in the section *Maintenance in Predict* in the *Predict Reference* documentation.

Editing Child Lists

To edit the lists of entry points, programs and files linked to a program, call the object list editor using one of the following methods:

- Enter Y in the field Associations at the bottom of the Add, Copy or Modify program screens and select Uses PR concept./Uses FI concept.. An asterisk before one of these fields indicates that the program already contains a list of programs or files for the association.
- Call the function Edit entry points or Link children in the Program Maintenance menu (codes R and L).
- Enter command `EDIT PROGRAM ENTRY <program ID>` or `LINK CHILDREN`.

Overview of Language-Specific Program Types

The table below lists the program types permitted for a program written in a particular language and indicates whether the program can have a list of entry points. In third generation languages, marked * in the table below, functions and subprograms can be documented as programs of type F and N respectively, but any active references for these programs will have type P (main program). The active references of these programs will be correctly connected in the active retrieval functions to programs of types P, N and F.

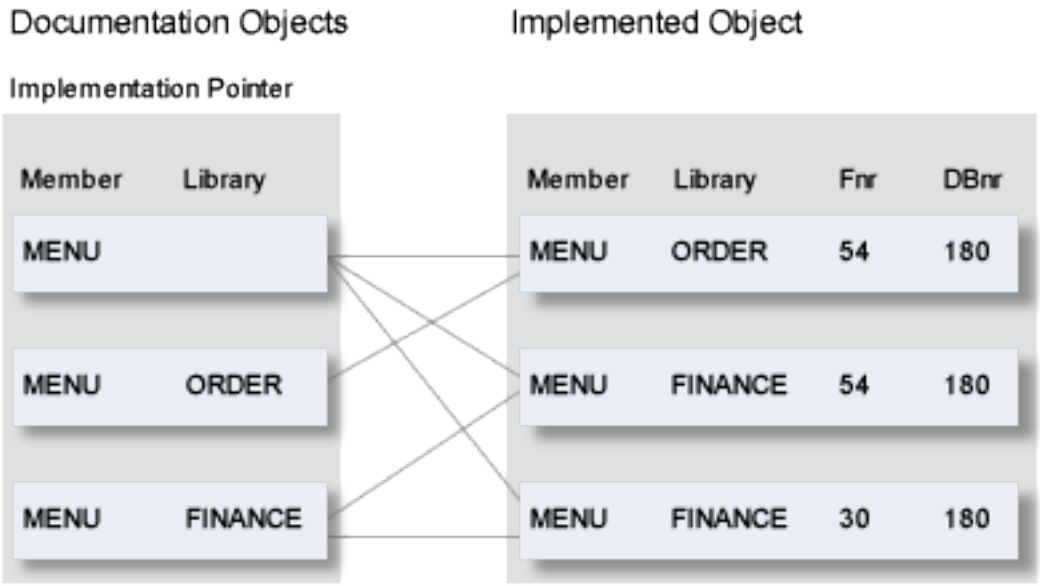
Language		Permitted Program Types	Entry Points allowed?
B	BAL (Assembler)*	C D F I N P U	yes
C	COBOL*	C D F I N P U	yes
E	Natural EL	D Y	no
F	FORTRAN*	C D F I N P	yes
G	ADA*	C D F N P	yes
H	C*	C D F I N P U	yes
J	Job Control Language	D J	no
N	Natural	A C D G H I K L M N O P S T X 1 4 5	no
O	Other	C D F H M N P	yes
R	Rexx	R	yes
P	PL/I*	C D F I N P U	yes
Q	Static SQL	D I P	yes

Language		Permitted Program Types	Entry Points allowed?
S	SQL procedure language	R U	no
V	Java	R	yes
Z	System program	D E	yes
0 - 9	user-defined	C D F N P	no

New languages (code 0 - 9) are defined with the program U-PGMLAN. See the section *U-PGMLAN - Define New Program Language* in the section *User Exits* in the *Predict Administration* documentation.

Combinations of Parameters for Natural Programs

If the same member is used in several libraries, multiple documentation of this member can be avoided by omitting parts of the implementation pointer. Predict then finds out for itself all the libraries in which this member exists. In the example below, the library name is omitted.



The valid combinations of implementation pointer parameters permitted for Natural programs are shown below.

Member	Y	Y	Y	Y
Library		Y	Y	Y
Fnr			Y	Y
DBnr				Y

Program List Specific Editor Commands

The following commands are available when editing one of the following lists:

- Uses FI conceptually
- Entry point list
- Uses PR conceptually

ACTIVE

Insert information from XRef data into the object list. Mark objects that are used with <active, and mark objects that are not used with <unused. XRef data without a corresponding documentation object is marked *NOT DOCUMENTED*. An object ID can then be entered and the .E command can be used to create a Predict object corresponding to the XRef data. The implementation pointer for the new object is derived from XRef data and automatically inserted into the input fields of the Add menu.

UPDATE

Update active reference data in the object list. Mark used objects with <active and delete unused objects from the list. Comments on the ACTIVE command (above) also apply to this command.

RESET

Switches back to normal edit mode after ACTIVE or UPDATE have been issued. Information displayed in the right column is no longer derived from XRef data but is taken from the Predict objects. All lines marked *NOT DOCUMENTED* are removed from the list. X and Y marks and scan values specified with the SCAN command are reset (as with the RESET command in any other list editor).

Generating Database Request Modules -DBRMs- from Program Objects of Language Q -Static SQL-

A Natural for DB2 database request module (DBRM) can be generated by the function CREATE DBRM of Natural DB2 from the list of entry points in a Predict object for a program of language Q (static SQL). See the section *DB2 and SQL/DS* in the *Predict and Other Systems* documentation.

Each entry point must be a Natural program that uses this DBRM. The Predict object should specify the member where the function GENERATE DBRM is to store the DBRM. The table below lists the columns of information that can be stored about entry points for a program of language Q only.

Column	Meaning
NAT-lib	The name of the library in which the Natural program is stored.
NAT-pgm	The name of the member in which the Natural program is stored.
Typ	The subtype of the Predict object for the Natural program.
Documentation	The ID of the Predict object for the Natural program.

The name of each entry point is concatenated. For detailed information on how the name is created, see the section *Static SQL* in the *Predict and Other Systems* documentation. This name is used for the entry point when displaying the DBRM's Predict definition (retrieval function) or its active references (LIST XREF command).

For any type of program except Q, the names of the entry points are stored in a single column. The editor commands ACTIVE and UPDATE can be used to insert active reference data into an entry point list.

Purge Program - Code P

The following rules apply:

- A program cannot be purged if it is linked to packagelist.
- If a program is implemented, a message tells you that XRef data will be deleted, too.

Redocument Program - Code X

Creates Predict documentation objects for implemented programs (members). The function is used when redocumenting applications.

Predict retrieves the information needed to create the documentation object for a member either by scanning source code (only for Natural programs) or by evaluating XRef data.

Calling a Redocument Function

Online

Redocument functions are executed in two steps:

1. Select the programs to be processed using the parameters in the first Redocument Program screen as selection criteria. See ***Selecting Programs to be Redocumented***.
2. Determine the scope of the redocumentation using parameters in the second Redocument Program screen. See ***Specifying the Redocument Parameters***.

Batch Mode

The function Redocument program is one of the few maintenance commands that can be entered in batch mode. The additional parameters that can be specified and a sample `REDOCUMENT` command are given in the section *Predict in Batch Mode* in the *Predict Administration* documentation.

Redocumenting Programs under Natural Security

Under Natural Security, some restrictions apply to this function to prevent unauthorized access to Natural sources. The same logic is used as in the SYSMAIN utility to check the user's access rights. The switch SYSMAIN from Natural Security is also interpreted in the Redocument program function.

See section *SYSMAIN under Natural Security* in the Natural Security documentation for more information.

Selecting Programs to be Redocumented

Enter code X in the Program Maintenance menu to display the Redocument Program screen:

```
13:25:12             *****  P R E D I C T  *****             2007-05-31
Plan   0                   - Redocument program -



Member ...........                                File number ....... 54
Library ..........                                Database number ... 180
Language ........* N          Natural             Password ..........
Source/XRef ...... S (S/X)    Source              Cipher ............
Member types ....*            All
```

Parameters	
Member name	Name of the member to be redocumented. Asterisk notation can be used.
Library	Depending on the parameter Source/XRef, either ■ library containing the members to be redocumented are stored, or ■ library of XRef data.
Language	B — BAL/Assembler C — COBOL F — FORTRAN G — Language ADA H — Language C N — Natural P — PL/I Q — Static SQL X — All but Natural If option X (All but Natural) is entered, Predict redocuments all 3GL programs that meet the specified selection criteria.
Source/XRef	S — Source code is evaluated to create the Predict object (only for Natural programs). X — XRef data is evaluated.
Member type	Additional selection criterion. Only member of the given types will be redocumented. See **Program Types** for a list of types that can be specified. A list of up to 9 member types can be specified. Member types can be specified without any delimiter (for example: ACGH) For third generation languages, only P (program) can be specified.
File number, Database number	Specify the FUSER file where the members to be processed are stored. Only applicable if Source/XRef is set to S.
Password, Cipher	Password and cipher code defined in Adabas can be specified if required. Only applicable if Source/XRef is set to S.

Specifying the Redocument Parameters

The following screen appears if language type N is entered in the Redocument Program menu.

```
13:13:28             *****  P R E D I C T  *****              2007-05-31
Plan   0                   - Redocument program -

Processing options
  Processing option ..........* L  List only
  Link to system ............. GER-SY
  Library structure ..........*

Naming options (only applicable if 'Processing option' is 'Add')
  Program ID prefix ...........
  Lib.name as sec.prefix ...... Y (Y,N)

Contents of documentation                              Implementation pointer
  Abstract ...................* S  Statistics          Library .. Y (Y/N)
  Description ................* B  Header comment      Fnr ...... Y (Y/N)
  Replace/append description .. R (R/A)                DBnr ..... Y (Y/N)
  Program list ...............* U  Update
  File list ..................* U  Update
  Default owner ...............
  First default keyword .......
  Second default keyword ......
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Using Default Values

All parameters of the Redocument function can be set to default values by the data dictionary administrator in the Redocumentation Using Source Code screen, which is called by code R in the Modify General Defaults menu. Individual input fields can be protected. Protected default values cannot be overwritten. Protected fields are skipped when positioning the cursor with TAB.

Note: Depending on the parameter Source/XRef (see table above), some parameters may not be contained in the screen. This is stated in the description of parameters below.

Processing options

Parameters		
Processing option	Determines the type of processing performed by the redocument function. Valid values:	
	A	Add: creates Predict objects for programs that are not already documented. Programs that meet the given selection criteria and are already documented will also be contained in the list and marked with the string Rejected in the column Status.
	R	Add and Replace: creates documentation for all programs that meet the given selection criteria. Existing documentation objects will then be replaced. If the program is a class, the objects of type interface, method and property are also documented.

Parameters		
	L	List: lists programs that have not yet been documented in Predict. Programs that meet the given selection criteria and are already documented will also be contained in the list and will be marked with string Rejected in the column Status.
Link to system	If Processing option is set to Add or Replace, the program objects created by the function are automatically linked as children to the specified System. If no system is specified, Predict looks for a documentation object of type system with the given Library, DBnr and Fnr. The system object that meets most of these criteria is inserted in this input field.	
Library structure	You can specify a library structure that is used to determine the used programs. If no structure is specified, evaluation is performed without a library structure.	

Naming options

Only applicable if Processing option=Add.

The ID of the program object created by the redocument function contains up to three parts, separated by hyphens.

Parameters	
Program ID prefix	A prefix that can be specified with the parameter Program ID prefix.
Lib.name as sec.prefix	The library name of the member as secondary prefix, if the parameter Lib.name as sec.prefix is set to Y.
-	The third part is the name of the member.

Contents of documentation

These parameters determine the information to be documented.

Parameters		
Abstract	Determines the information to be contained in the abstract:	
	S	Statistical data (including the date and time when a member was cataloged).
	B	Comment lines in the header. Only applicable if Source/XRef is set to S.
	A	Comment lines in the header and statistical information. Only applicable if Source/XRef is set to S.
	N	No abstract is created.
Description	Determines the information to be included in the extended description. Only applicable if Source/XRef is set to S.	
	A	Comment lines. Start with either * or /* in the first column followed by a series of characters other than * or blank.

Parameters		
	B	All comment lines in the header of the member.
	R	Comment lines and remarks. A remark starts with /* in any column and is followed by a series of any non-blank characters.
	S	The whole source program.
	N	No extended description is created.
Replace/append descr.	Determines handling of extended descriptions. Only applicable if Source/XRef is set to S.	
	A	The extended description of a Predict program object that is replaced (see Processing option above) is not overwritten. Instead, the new extended description is appended to the old extended description.
	R	The old extended description is overwritten when a Predict object is replaced. Default.
Program list	Programs that are called from within a program (for example via a CALL or FETCH statement) can be included in the program list (Uses PR concept.) of the object. The parameter Program list has the following options:	
	U	Update. The old contents of the program list (Uses PR concept.) are completely replaced by the information extracted from the XRef data.
	A	Add active links. Additional entries in the program list (Uses PR concept.) are created, documenting the use of programs not already documented. All other entries in the list will be kept. This option only makes sense if an existing documentation object is replaced.
	N	No entries in the program list (Uses PR concept.) are created.
File list	Files that are used by a program can be included in the file list Uses FI concept. of the program. See Program list above for description of the options.	
Default owner	The default Owner specified is included in the owner list of the object. The Owner must be defined in at least one object of type user. Only applicable if Processing option is set to A.	
First default keyword	Only applicable if Processing option is set to A.	
Second default keyword	Two keywords can be specified that are linked to the objects created. The keywords must be defined in Predict.	
Implementation pointer: Library, DBnr, Fnr	These parameters determine two things: ■ The amount of information to be stored in the implementation pointer of the Predict program object to be created by the Redocument function. If the library, DBnr or Fnr is to be added the implementation pointer by the redocument function, the respective parameter must be set to Y.	

Parameters		
	■ Which information of existing Predict program objects is evaluated to determine whether an implemented program is already documented. For example: if Library, DBnr and Fnr are set to Y, a Predict object is only regarded as the documentation of an implemented program if its implementation pointer contains correct values for the following: ■ member name ■ library ■ DBnr ■ Fnr. **Note:** If the parameters Implement. Library and Implement. DBNR/FNR in the Predict Defaults have been set to either Disallow (D) or Force (F), the parameters above cannot be set to Y or N respectively.	
Handle /* in columns 1+2 as comment or as remark	C	A line with /* in the first two columns is interpreted as a comment line.
	R	A line with /* in the first two columns is interpreted as a remark.
	This parameter is specified in the defaults.	

Edit procedure code of a program - Code Y

This function can only be executed for programs of type SQL procedure or Database function with language SQL procedure.

The Predict Description Editor is called. Additional checks are performed when the procedure code is cataloged. See the section *Editors in Predict* in the *Predict Reference* documentation for more information.

52 Program Retrieval

- Program-Specific Retrieval Parameters 438
- Function Programs with Children with Child Type Program 438
- Layout of Program Lists 438
- Output Options for Program Retrieval 439
- Output Options for Program Retrieval - Continued 440

Program-Specific Retrieval Parameters

The following program-specific parameters determine the scope of reports.

Program of type	Restrict report to programs of the given type. See *Program Types* for a list of valid types.
Language	Restrict report to programs of the given language. See *Languages* for a list of valid languages.
Member	Restrict report to programs documenting the given member.
Library	Restrict report to programs documenting a member in the given library. See *Program-Specific Libraries* for a list of standard libraries.
User system Fnr	Restrict report to program objects documenting implemented programs in this user system file.
User system DBnr	Restrict report to program objects documenting implemented programs in this database.
Belongs to SY	Restrict report to programs belonging to this system object.

Function Programs with Children with Child Type Program

The Retrieval function Program with Children with association "Uses PR concept." evaluates only documentation data. If you need information on the implementation of a program, use the Active Retrieval function Programs using programs.

Layout of Program Lists

```
13:54:04             *****  P R E D I C T  *****              2007-05-31
                              - List Program -

--------------------------------------------------------------------------------
Cnt  Program ID                          Type Lang Member   Library    Fnr   DBnr

  14 STK-PR-O                             P    O   CP1E     ST-PRDE
  15 STK-PR-STATIC-SQL                    P    Q   HUXEL    HUXEL      255    255
  16 STK-PR-1                             O
  17 * STK-PR-2                           N    N   N-SECCHC
                                   Implementation: N-SECCHC GMA         54    180
                                                   N-SECCHC NEWDICCO    54    180
  18 STK-PROC                             R    S   KSTK     KKKKK
  19 * STK-REDOC                          P    N   Z-HI1    STK
                                   Implementation: Z-HI1    STK         54    180
```

Meaning of Columns	
Program ID	The ID of the program object. **Note:** An asterisk in the first column indicates that the program is implemented. Implemented in this sense means that XRef data exists for the documentation object.
Type	Program type. See *Program Types* for a list of valid types and codes.
Lang.	The language in which the program is written. See *Languages* for list of valid languages and codes.
Member, Library, Fnr, DBnr	Implementation pointer of the program object, or - if the object is implemented - the physical implementation of the member(s) documented by the program. In the sample screen above, program STK-PR-2 has implementation pointer N-SECCHK (for member) and documents member N-SECCHK in libraries GMA and NEWDICCO.

Output Options for Program Retrieval

Note: Unless output mode is S, the option Cover page is always valid. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Association attributes			Y	Y	Y	Y			Y	Y	Y	Y				
Attributes	Y		Y				Y		Y				Y			
Connecting character				Y						Y						
Description	Y		Y	Y			Y		Y	Y			Y			
Display modifier	Y		Y				Y		Y				Y			
Dummy/Placeholder										Y		Y		Y		Y
Entry points	Y		Y				Y		Y				Y			
Extract	Y		Y	Y			Y		Y	Y			Y	Y		
Keywords	Y		Y	Y			Y		Y	Y			Y			
Mark implementation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
No. abstract lines	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
Owner	Y		Y	Y			Y		Y	Y			Y			
With users	Y		Y	Y			Y		Y	Y			Y			
Procedure code	Y		Y				Y		Y				Y			

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Show implementation	Y	Y	Y				Y		Y				Y			
Use Con-form	Y		Y	Y			Y		Y	Y			Y			
User exit	Y		Y				Y		Y				Y			

Output Options for Program Retrieval - Continued

Retrieval Type	U		E				C			
Output Mode	D	L	T		X		L		D	
Current/Related	c	c	c	r	c	r	c	r	c	r
Association attributes			Y	Y						
Attributes	Y			Y		Y				
Connecting character				Y		Y				
Description	Y					Y				Y
Display modifier	Y									
Dummy/Placeholder				Y		Y	Y		Y	
Entry points	Y									
Extract	Y			Y		Y			Y	Y
Keywords	Y			Y		Y				Y
Mark implementation	Y	Y	Y	Y	Y	Y		Y		Y
No. abstract lines	Y	Y		Y		Y		Y		Y
Owner	Y			Y		Y				Y
With users	Y									Y
Procedure code	Y									
Show implementation	Y									
Use Con-form	Y					Y				Y
User exit	Y									

XI

- 53 Property 443
- 54 Report Listing 447
- 55 Server 453

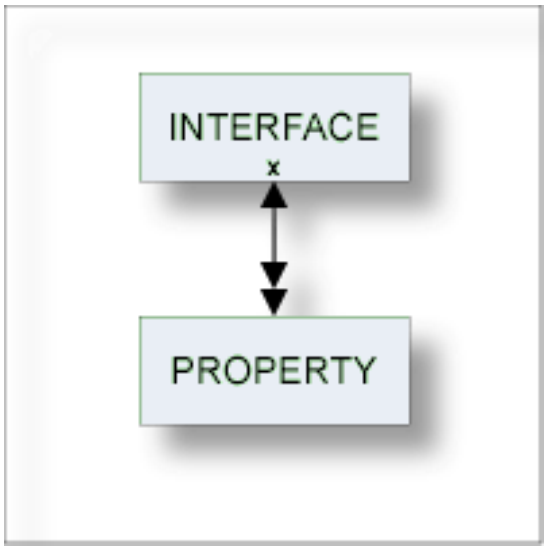
53 Property

- Property Maintenance Menu 444
- Add a Property Screen 445
- Property Retrieval 446

This object type is used to document the properties of an interface.

In the predefined Predict metastructure, a property can have passive and active associations of the following types:

Valid passive association:	"Defined in IE"
Valid active association:	no predefined association



Property Maintenance Menu

This menu is called with function code M and object code PY in a Predict main menu, or with the command `MAINTAIN PROPERTY`.

```
13:23:08             *****  P R E D I C T  *****             2007-05-31
Plan   0               - (PY) Property Maintenance -            Profile SYSTEM

 Function                                Function

 A  Add a Property                       D  Display Property
 C  Copy Property                        L  Link children
 M  Modify Property                      S  Select Property from list
 N  Rename Property
 P  Purge Property


Function ..............

Property ID ...........                                   Attributes....*
Copy ID ...............
Defined in IE .........

Restrictions .........*   Profile Default,empty         Association ...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters

The Property Maintenance menu contains only global attributes. See ***Global Attributes***.

The functions are described in the section *Maintenance* in the *Predict Reference* documentation.

Add a Property Screen

The following screen appears for the function Add a Property. The screens for functions Copy and Modify are similar.

```
13:25:10             *****  P R E D I C T  *****             2007-05-31
                            - Add a Property -
Property ........ EXAMPLE
Defined in IE ..*
Keys ..
Zoom: N

Attributes
 Property name ....
 Readonly .........   (Y/N)
 Abstract    Zoom: N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters		
Property	ID of the property.	
Property name	Name of the property.	
Readonly	Y	Variables cannot be modified.

Property Retrieval

Information on property objects is gathered using standard retrieval functions. See the section *Retrieval* in the *Predict Reference* documentation.

Output Options for Property Retrieval

The output options valid for this object type are identical to those for object type dataspace. See *Output Options for Dataspace Retrieval*.

54 Report Listing

- Report Listing Maintenance Menu 448
- Report Listing ID 449
- Modify Report Listing Screen 450
- Report Listing Retrieval 451

Objects of type Report Listing log

- a transfer operation of the Predict Coordinator, or
- a conversion operation.

Report listings are added automatically with an ID assigned by the system. For this reason, the functions Add and Copy are not available for this object type.

In the predefined Predict metastructure, a report listing can have passive and active associations of the following types:

Valid passive association:	no predefined association
Valid active association:	Extract (default active association "Uses ET")

When transferring data with the Predict coordinator, the extract containing the objects to be transferred is automatically linked as a child to the report listing.

See the *Predict Coordinator* documentation for more information.

Report Listing Maintenance Menu

This menu is called with function code M and object code RT in a Predict main menu or with the command `MAINTAIN REPORTLISTING`.

```
13:24:44              *****  P R E D I C T  *****              2007-05-31
Plan   0             - (RT) Report listing Maintenance -            Profile HNO

 Function                                Function

 M  Modify Report listing                D  Display Report listing
 N  Rename Report Listing                L  Link children
 P  Purge Report listing                 S  Select Report listing from list




Function ..............

 Report listing ID .....                                   Attributes....*



Restrictions .........*   Profile HNO,used              Association ...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Enter one of the codes in the menu to execute one of the functions. These functions are described in the section *Maintenance* in the *Predict Reference* documentation. **Note:** As report listings are added automatically with an ID which is assigned by the system, the functions Add and Copy are not available for this object type.
Report Listing ID	ID of the report listing object. See *Report Listing ID* below.

Report Listing ID

The report listing ID is assigned automatically when an object is added and is composed as follows:

- **USR - User ID**
  The ID of the user who performed the coordinator function.

  In batch mode: the job name. This section is appended by underscore characters if less than 8 characters.

- **TYP - Subtype**
  One of the following:
  - EXP Export
  - IMP Import
  - CON Conversion
  - TRC Trace
  - UNL Unload
  - LOA Load
  - ALF* - ALF to Migrate conversion
  - MIG* - Migrate to ALF conversion

  Note: * Report listings of type ALF and MIG are no longer created with this version but objects of this type may exist from earlier versions.

- **200940803 - Date**
  Date on which the report listing was added. Format YYYYMMDD
- **1522453 - Time**
  The time at which the report listing was added. Format HHMMSST

Modify Report Listing Screen

```
13:35:13             *****  P R E D I C T  *****            2007-05-31
                         - Modify Report listing -
Report listing .. HNO-RT                              Added 2007-05-31 at 13:25
                                                         by HNO
Keys ..
Zoom: N

Attributes
 Subtype ............ Export

 Processing


Abstract    Zoom: N
```

Note: Parameters not listed here are described under *Global Attributes*.

Explanation	
Added by	The function with which the report listing was added. The user who created the report listing can be seen in the first eight characters of the report listing ID.
Subtype	Subtype of report listing. ■ Conversion (ALF to Migrate or Migrate to ALF conversion) ■ Export ■ Import ■ Trace ■ Load ■ Unload
Processing	
Exported / Not Exported	For the function Export: The number of objects successfully exported / objects not exported due to errors. **Note:** See the extended description of the report listing for a complete list of these objects.
Loaded / Replaced / Not Loaded	For the function Import: The number of new objects successfully loaded / existing objects overwritten / objects not loaded due to errors. **Note:** See the extended description of the report listing for a complete list of these objects.

Report Listing Retrieval

Information on report listings is retrieved using standard retrieval functions. These functions are described in the section *Retrieval* in the *Predict Reference* documentation.

Layout of Report Listing Lists

```
13:01:24             *****  P R E D I C T  *****             2007-05-31
                          - List Report listing -


-------------------------------------------------------------------------------
Cnt  Report listing ID                Subtype

1235 FH______-EXP-19950213-1133434    Export
1236 FH______-EXP-19950213-1134044    Export
1237 FH______-IMP-19950213-1135086    Import
1238 FH______-IMP-19950213-1750037    Import
1239 FH______-IMP-19950213-1758171    Import
1240 FH______-MIG-19950209-1531474    Convert
1241 GER-RT                           Import
1242 GER_____-ALF-19950206-1017009    Convert
```

Output Options for Report Listing Retrieval

The output options valid for this object type are identical to those for object type dataspace. See *Output Options for Dataspace Retrieval*.

55 Server

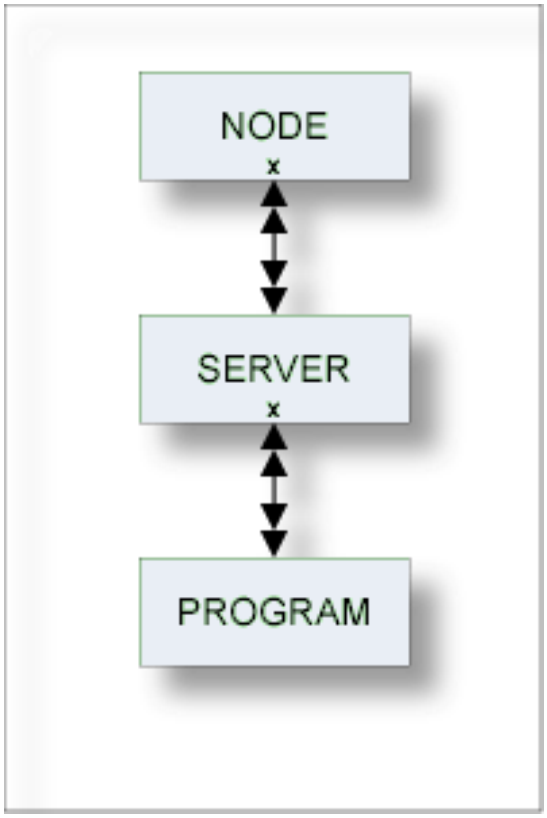
- Server Maintenance Menu 454
- Add a Server Screen 455
- Server Retrieval 456

This object type, together with object type Node, is used to document remote procedure calls.

An object of type Server documents all programs available on a logical server.

In the predefined Predict metastructure, an interface can have passive and active associations of the following types:

Valid passive association:	"Contained in NO" (default passive association)
Valid active association:	"Uses PR"



Server Maintenance Menu

This menu is called with function code M and object code SV in a Predict main menu or with the command `MAINTAIN SERVER`.

```
13:54:29             *****  P R E D I C T  *****            2007-05-31
Plan  10                - (SV) Server Maintenance -               Profile HNO

 Function                                Function

 A  Add a Server                         D  Display Server
 C  Copy Server                          L  Link children
 M  Modify Server                        S  Select Server from list
 N  Rename Server
 P  Purge Server


Function ..............

Server ID .............                                   Attributes....*
Copy ID ...............
Contained in NO .......

Restrictions .........*   Profile HNO,used              Association ...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Enter one of the codes from the menu to execute one of the maintenance functions. These functions are described in the section *Maintenance* in the *Predict Reference* documentation.

Add a Server Screen

The following screen is called for function Add a Server. The screens for functions Copy and Modify are similar.

```
13:29:37             *****  P R E D I C T  *****            2007-05-31
                               - Add a Server -
Server ........... HNO-SV
Contained in NO .*
Keys ..                                                          Zoom: N


 Server name ......
  Abstract    Zoom: N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Server ID	ID of the server object.
Contained in NO	ID of the parent node.
Server name	Name of the server must be specified. Up to 8 characters.

Server Retrieval

Only standard retrieval functions are used. See the section *Retrieval* in the *Predict Reference* documentation.

Layout of Server Lists

Server lists contain the server IDs and the server names.

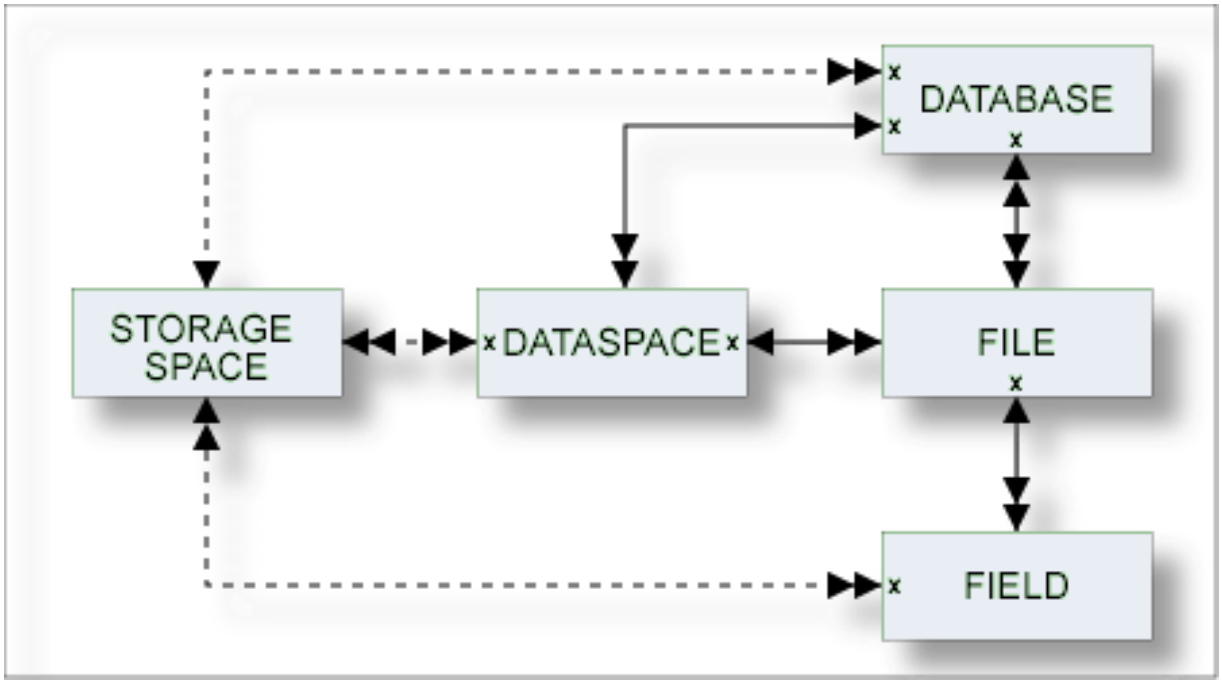
```
13:19:16              *****  P R E D I C T  *****              2007-05-31
                               - List Server -


-------------------------------------------------------------------------------
Cnt  Server ID                          Server name

   1 FST-SERVER                         FST-TEST
   2 HEB-SV                             SERVSERV
   3 HNO-SV1                            server1
```

Output Options for Server Retrieval

The output options valid for this object type are identical to those for object type Dataspace. See *Output Options for Dataspace Retrieval*.

XII Storagespace

DB2 storagegroups are documented in Predict with the object type Storagespace. See the section *DB2 and SQL/DS* in the *Predict and Other Systems* documentation.



In the predefined Predict metastructure, a storagespace has no predefined association. References to storagespaces are realized with the attribute (Default) Storagespace of objects of type Database, Dataspace and Field.

The description of object type Storagespace is organized under the following headings:

Maintaining Objects of Type Storagespace
Storagespace Retrieval

56 Maintaining Objects of Type Storagespace

- Storagespace Maintenance Menu 460
- Add a Storagespace Screen 461
- Storagespace-Specific Maintenance 462

Storagespace Maintenance Menu

The Storagespace Maintenance menu is called with function code M and object code SC in a Predict main menu or the command `MAINTAIN STORAGESPACE`.

```
10:22:43             *****  P R E D I C T  *****            2020-05-31
Plan   0             - (SC) Storagespace Maintenance -            Profile XYZ

 Function                                Function

 A  Add a Storagespace                   D  Display Storagespace
 C  Copy Storagespace                    L  Link children
 M  Modify Storagespace                  S  Select Storagespace from a list
 N  Rename Storagespace                  J  Physical Attributes
 P  Purge Storagespace


Function .........

Storagespace ID ..                                        Attributes....*
Copy ID ..........


Restrictions ....*   Profile XYZ,used                   Association ...*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. Function Purge is described under *Storagespace-Specific Maintenance*.

Add a Storagespace Screen

The screen is displayed for the Add a Storagespace function. The Copy and Modify screens are similar.

```
15:30:22                ******  P R E D I C T  ******                  2020-05-25
                           - Add a Storagespace -
Storagespace .... XYZ-SC

Keys ..                                                                  Zoom: N

Storagespace attributes
  Storagegroup name .....
Physical attributes in <Default Server>                 (new)
  VSAM catalog name .....                 Storage class name ...
  Data class name .......                 Device type ..........
  Management class name .
  Keylabel name .........                                               Zoom: N

  Volumes/Partitions
     1
    10
    19
    28
Abstract     Zoom: N


  Additional attributes ..* N          Associations ..* N                    ↩
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Storagespace	ID of the Predict storagespace object.
Storagespace attributes	
Storagegroup name	Name of the storagegroup in DB2.
VSAM catalog name	Name or alias of an ICF catalog. Aliases are used for names of ICF catalogs that are longer then eight characters.
Data class name	Identifies the name of the SMS data class to associate with the DB2 storage group. The SMS data class name must be from 1-8 characters in length. The SMS storage administrator defines the data class that can be used. A data class must not be specified more than one time.
Management class name	Identifies the name of the SMS management class to associate with the DB2 storage group. The SMS management class name must be from 1-8 characters in length. The SMS storage administrator defines the management class that can be used.

Parameters	
	A management class must not be specified more than one time.
Keylabel name	Used for encryption.
Storage class name	Identifies the name of the SMS storage class to associate with the DB2 storage group. The SMS storage class name must be from 1-8 characters in length. The SMS storage administrator defines the storage class that can be used. A storage class must not be specified more than one time.
Device type	For documentation purposes.
Volumes/Partitions	Physical volume(s)/partition(s) where the storagespace resides. Up to 36 volumes/partitions can be entered here. Specify MORE volumes/partitions=Y to specify up to 133 volumes/partitions.

Storagespace-Specific Maintenance

Purge Storagespace - Code P

The following restriction applies to this function:

- A storagespace cannot be deleted if it is still referenced by a database, a dataspace or a file.

Otherwise this function behaves as described in the section *Maintenance* in the *Predict Reference* documentation.

57 Storagespace Retrieval

- Unused Storagespaces - Code N 464
- Layout of Storagespace Lists 464
- Output Options for Storagespace Retrieval 464
- Output Options for Storagespace Retrieval - Continued 465

Unused Storagespaces - Code N

Lists unused storagespaces. A storagespace is regarded to be unused if it is not referenced in a dataspace or field object.

Layout of Storagespace Lists

```
10:56:33            *****  P R E D I C T  *****            2007-05-31
                         - List Storagespace -


-------------------------------------------------------------------------------
Cnt  Storagespace ID                    Stgr name VCAT name

   1 ARH-SC                             ARH_SC
   2 ARH-SC-2                           STOGR2
   3 BOE-ST01                           FRITZ
   4 CHD-SC                             YYYY
   5 * CHD-STORAGESPACE                 CHDSPC    PB4
   6 CHD-STOSPACE                       SPATZ
   7 * CHD-STOSPATZ                     CHDSPTZ   PB4
   8 * DEVELOP                          DEVELOP   DB2V23
```

Meaning of Columns	
Storagespace ID	ID of the storagespace. If the output option Mark implementation is set to Y, implemented objects are marked with an asterisk. 'Implemented' means here that a DB2 storagegroup has been generated from the storagespace.
Stgr name	Name of the DB2 storagegroup.
VCAT name	Name or alias of an ICF catalog.

Output Options for Storagespace Retrieval

Notes:

1. Unless output mode is S, the option Cover page is always valid.
2. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Association attributes			Y	Y	Y	Y			Y	Y	Y	Y				
Attributes	Y		Y				Y		Y				Y			
Connecting character				Y						Y						
Description	Y		Y	Y			Y		Y	Y			Y			
Display modifier	Y		Y				Y		Y				Y			
Dummy/Placeholder										Y		Y		Y		Y
Extract	Y		Y	Y			Y		Y	Y			Y	Y		
Keywords	Y		Y	Y			Y		Y	Y			Y			
Mark implementation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
No. abstract lines	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
Owner	Y		Y	Y			Y		Y	Y			Y			
With users	Y		Y	Y			Y		Y	Y			Y			
Show implementation	Y		Y				Y		Y				Y			
Use Con-form	Y		Y	Y			Y		Y	Y			Y			
User exit	Y		Y				Y		Y				Y			

Output Options for Storagespace Retrieval - Continued

Retrieval Type	U		E				C				N	
Output Mode	D	L	T		X		L		D		D	L
Current/Related	c	c	c	r	c	r	c	r	c	r	c	c
Association attributes			Y	Y								
Attributes	Y			Y		Y					Y	
Connecting character				Y		Y						
Description	Y					Y				Y	Y	
Display modifier	Y										Y	
Dummy/Placeholder				Y		Y						
Extract												
Keywords	Y			Y		Y					Y	
Mark implementation	Y	Y	Y	Y	Y	Y		Y			Y	Y

Retrieval Type	U		E				C				N	
Output Mode	D	L	T		X		L		D		D	L
Current/Related	c	c	c	r	c	r	c	r	c	r	c	c
No. abstract lines	Y	Y		Y		Y		Y		Y	Y	Y
Owner	Y			Y		Y					Y	
With users	Y										Y	
Show implementation	Y										Y	
Use Con-form	Y					Y				Y	Y	
User exit	Y										Y	

XIII

- 58 System 469
- 59 Trigger 477

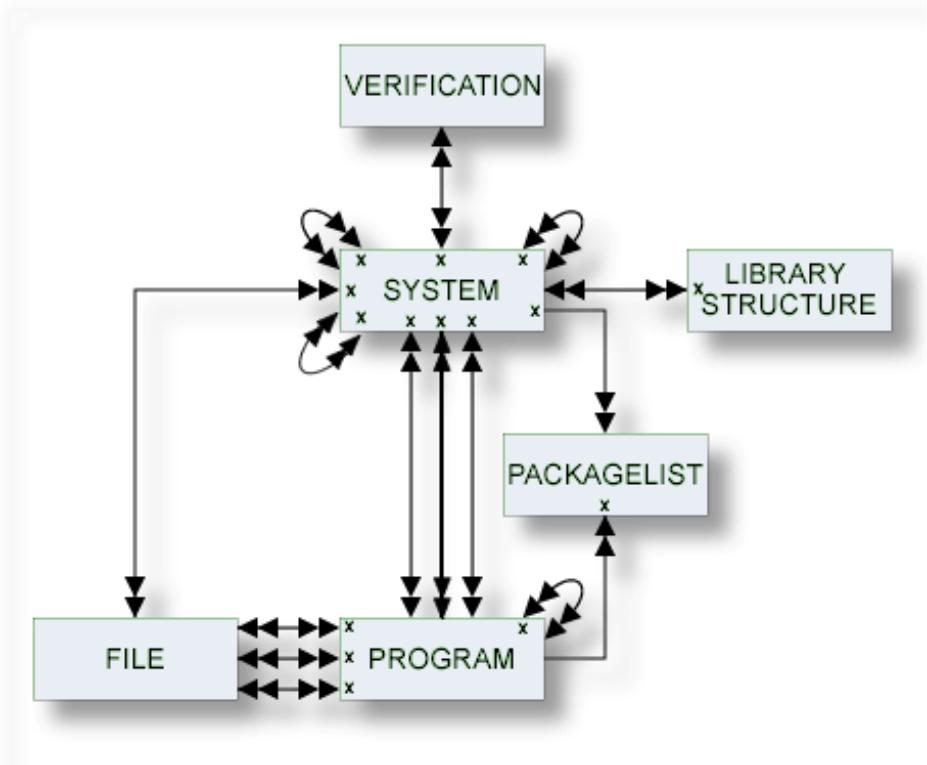
58 System

- System Maintenance Menu 471
- System Types 472
- Add/Copy/Modify System Screen 472
- System-Specific Maintenance 473
- System Retrieval 474

An application can be documented with a Predict object of type System. See *System Types* for a list of possible system types.

In the predefined Predict metastructure, a system can have passive and active associations of the following types:

Valid passive associations:	"Contained in LS" "Is subappl. of SY" "Is library of SY"
Valid active associations:	"Uses PR concept." (default active association) "Has subappl. SY" "Uses PG" "Has component PR" "Has component VE" "Has component FI" "Has library SY"



System Maintenance Menu

The System Maintenance menu is called with function code M and object code SY in a Predict main menu or the command `MAINTAIN SYSTEM`.

```
13:51:33             *****  P R E D I C T  *****            2007-05-31
Plan   0                - (SY) System Maintenance -               Profile HNO

 Function                                Function

 A  Add a system                         D  Display system
 C  Copy system                          L  Link children
 M  Modify system                        S  Select system from a list
 N  Rename system
 P  Purge system


Function .........

System ID ........                                   Attributes........*
Copy ID ..........                                   System of type....*
Library ..........                                   User system Fnr
                                                     User system DBnr ...

Restrictions ....*   Profile HNO,used                Association.......*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Standard functions are described in the section *Maintenance* in the *Predict Reference* documentation. Functions Purge system and Rename system are described under *System-Specific Maintenance*.
System of type	For the Select function: a system type can be specified as a selection criterion. For the Add and Copy functions: the system type can be specified here. This type will be passed to the add System or copy System screen. See *System Types* below for a list of valid values.
Library, User system Fnr/DBnr	For the select function: Implementation pointer values can be used to restrict the scope of objects to be processed. Only those Predict system objects will be processed that document libraries meeting the specified Library/Fnr/DBnr parameters.

System Types

The table below contains a list of all valid system types.

Code	System Type
A	Application Library
B	Base Application
C	Conceptual. Used to outline the preliminary description of an application in the design phase.
G	3GL Application
O	Compound Application
P	DB2 plan. Used to document a DB2 application.

Add/Copy/Modify System Screen

The screen is displayed for the Add a System function. The Copy and Modify screens are similar.

```
13:54:46             *****  P R E D I C T  *****            2007-05-31
                              - Add a System -
System ID ....... HNO-SY
Type ...........* C  Conceptual
Keys ..                                                          Zoom: N

Implementation pointer
  Library ............
  User system Fnr ....
  User system DBnr ...
DB2 Plan name ........
Profile
  Name ...............
  Fnr ................
  DBnr ...............
Port .................
Server name ..........                                           Zoom: N
Development platform *

Abstract     Zoom: N


  Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters		
System ID	The ID of the Predict system object. A read-only field.	
Type	System type. Enter asterisk to display valid values or see list in the section *System Types* above.	
Implementation pointer	Library	The name of the library. For type G: The library cannot be changed if XRef data exists (the library is used by a 3GL program).
	User system Fnr	The file number of the user system file (FUSER).
	User system DBnr	The database number of the user system file.
DB2 plan name	Unique DB2 plan name. Only applicable to DB2 plans (systems of type P).	
Profile	Name	The name of the profile.
	Fnr	The number of the user system file.
	DBnr	The number of the database in which the user system file is located.
Port	The port number.	
Server name	The name of the server.	
Development platform	The development platform. Enter one of the following values: ■ MAINFRAME ■ Linux ■ PC This parameter specifies for which type of server the application is developed.	

System-Specific Maintenance

Identifying Systems

Systems documented with Predict objects of type System can be identified with three parameters: library, file number and database number. The three possible combinations of these parameters are shown below.

Library	Y	Y	Y
File number		Y	Y
Database number			Y

Purge System - Code P

The following rules apply to this function:

- A system of type A (Application Library) cannot be deleted if it is linked to one or more systems via association "Has library SY".
- A system of type G (3GL application) cannot be deleted if XRef data exist.
- If you confirm the function with `DELETE`, the following objects are deleted:
  - the system object
  - all links to child objects
  - all links from parent objects
- If you confirm with `SCRATCH`, the following objects are deleted additionally:
  - Programs linked to the system via association "Belongs to SY" (programs that are linked to packagelists via "Contained in PG" are not deleted)
  - all links to/from objects that are deleted together with the system
  - XRef data for the system (including DBRMs and system programs)
  - XRef data for scratched programs (parameter Language = Ada, BAL, COBOL, FORTRAN, PL/I, Static SQL, System Program).

Rename System - Code N

Use this function to change the ID and/or type of a system object. The following restriction applies:

- You cannot change the type of a system of type 3GL application for which XRef data exists.
- You cannot change the type of a system of type A (Application Library) if it is linked to one or more systems via association "Has library SY".

System Retrieval

System-Specific Retrieval Parameter

All system-specific retrieval parameters are described in the section *System Maintenance Menu*.

Systems with Children - Code T, with Child Type Program

The retrieval function Systems with Children (with association "Uses PR concept") evaluates only documentation data. If you require information on an implemented system, use the active retrieval function Systems containing programs.

Layout of System Lists

```
13:40:59            *****  P R E D I C T  *****            2007-05-31
                                - List System -                   Page:   1

Cnt  System ID                           Type Library  Fnr DBnr

   1 ADABAS                                C
   2 ARH-LO                                C
   3 * ARH-SYS                             A  ARH
   4 ARH-SYS-P                             P
```

Meaning of Columns	
System ID	ID of the system object. If the output option Mark implementation is set to Y, implemented objects are marked with an asterisk. "Implemented" in this case means that XRef data exists for at least one program contained in a library documented by the system object.
Type	The type of system. See list of valid types and codes under *System Types*.
Library, Fnr, DBnr	Information on where a system is implemented: Library, file number and database number of the user system file.

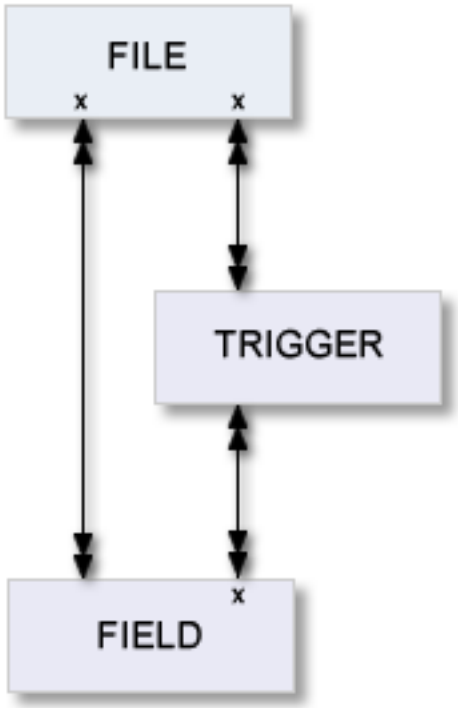
Output Options for System Retrieval

The output options valid for this object type are identical to those for object type Dataspace. See *Output Options for Dataspace Retrieval*.

59 Trigger

- Trigger Maintenance Menu 478
- Add a Trigger Screen 479
- Editing the Trigger Code of a Trigger 481
- Trigger Retrieval 482

This object type is used to define triggers for SQL tables and SQL table fields.



In the predefined Predict metastructure, a trigger can have passive and active associations of the following types:

Valid passive associations:	"Triggers FI" (default passive association) "Triggers EL "
Valid active association:	no predefined association

Trigger Maintenance Menu

This menu is called with function code M and object code TR in a Predict main menu or with the command `MAINTAIN TRIGGER`.

```
13:54:29             *****  P R E D I C T  *****             2007-05-31
Plan  10                - (TR) Trigger Maintenance -               Profile HNO

 Function                                Function

 A  Add a Trigger                        D  Display Trigger
 C  Copy Trigger                         L  Link children
 M  Modify Trigger                       S  Select Trigger from list
 N  Rename Trigger
 P  Purge Trigger


Function ..............
Trigger ID ............                                   Attributes....*
Copy ID ...............
Triggers FI ...........

Restrictions .........*   Profile HNO,used              Association ...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Enter one of the codes from the menu to execute one of the maintenance functions. These functions are described in the section *Maintenance* in the *Predict Reference* documentation.

Add a Trigger Screen

The following screen is called for function Add a Trigger. The screens for functions Copy and Modify are similar.

```
13:54:29             *****  P R E D I C T  *****            2007-05-31
                             - Add a Trigger -
Trigger .........  HNO-TR
Triggers FI ....*
Keys ..                                                          Zoom: N
Attributes
 Trigger name ....
 Trigger action ..*
 Trigger type ....*
 Abstract    Zoom: N





 Aditional attributes ..* N          Associations ..* N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Trigger ID	ID of the trigger.
Triggers FI	ID of the parent file.
Trigger name	Name of the trigger.
Trigger action	Activating a trigger with the statement: ■ Insert ■ Update ■ Delete
Trigger type	Activation time of a trigger: ■ After ■ Before ■ None

Editing the Trigger Code of a Trigger

Calling the Editor

Two methods are available for calling an editor to edit the Trigger code of Trigger:

- enter `Y` in the Additional attributes field in the bottom line of the Add/Copy/Modify Trigger screen and select Trigger code from the menu, or
- enter the command `EDIT TRIGGER TRIGGER <Trigger ID>`.

The editor called depends on the preferences specified in the Profile > Handling screen:

- if your first choice editor is "NATURAL", the Subquery Editor (a modified Natural Editor) is called,
- if your first choice editor is "SAG" or "Word for Windows", the Software AG Editor is called.

```
13:54:29                        - TR: HEB-TR -                        2007-05-31
      ------------------- <Trigger code> -------------------

***** **************************** top of data***************************
00001 save
***** *************************** bottom of data**************************









Command ===>                                                   Scroll==> CSR
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Rfind Rch   up    down        left  right leave
```

Trigger Retrieval

Only standard retrieval functions are used. See the section *Retrieval* in the *Predict Reference* documentation.

Layout of Trigger Lists

Trigger lists contain the trigger IDs and the trigger names.

```
13:19:16             *****  P R E D I C T  *****             2007-05-31
                             - List Trigger -


-------------------------------------------------------------------------------
Cnt  Trigger ID                        Trigger name

   1 HEB-TR-DEL                        HEB-TR-DEL
   2 HEB-TR-INS                        HEB-TR-INS
   3 HNO-TR1                           Trigger1
```

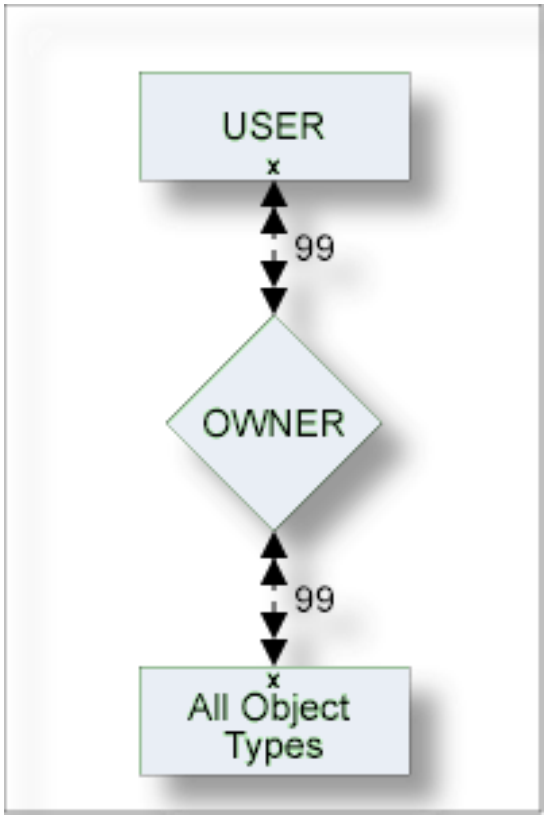
Output Options for Trigger Retrieval

The output options valid for this object type are identical to those for object type Dataspace. See ***Output Options for Dataspace Retrieval***.

XIV User/Owner

The object type User contains information on users and organizational units, such as name, ID or position within the company.

One attribute of this object type is Owner. Groups of users reflecting organizational units, such as project teams, can be formed by assigning individual users to an owner. Each user can belong to several owners. Owners can be associated to other types of Predict objects. See also *User/Owner* and *Keyword* in the section *Overview of Predict* in the *Introduction to Predict* documentation.



The description of object type User is organized under the following headings:

Maintaining Objects of Type User/Owner

User Retrieval

Owner Maintenance

Owner Retrieval

60 Maintaining Objects of Type User/Owner

- User/Owner Maintenance Menu 486
- Add/Copy/Modify a User Screen 487
- User Maintenance 488

User/Owner Maintenance Menu

The User Maintenance menu is called with function code M and object code US in a Predict main menu or the command `MAINTAIN USER`.

```
13:25:38              *****  P R E D I C T  *****             2007-05-31
Plan   4                  - (US) User Maintenance -                 Profile HNO

 Function                                Function

 A  Add a user                           L  Link children
 C  Copy user                            S  Select user from a list
 M  Modify user                          R  Rename/Merge owner
 N  Rename user                          E  Purge owner
 P  Purge user
 D  Display user

Function .........

User ID ..........                                      Attributes......*
Copy ID ..........
User name ........
Owner ............
Restrictions ....*   Profile HNO,used                   Association.....*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Executes one of the maintenance functions. Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. The functions Purge user and Edit owners of a user are described in the section *User Maintenance*, Rename/Merge and Purge owner are described in the section *Owner-Specific Maintenance Functions*.
User ID	Identifier of the Predict user object. Must start with a letter, and can be up to 8 characters long. See also section *Naming Conventions*.
User name	Name of the user. If the user name is unique, it can be specified instead of the user ID.
Owner	Owner ID. If the owner ID is unique, it can be specified instead of the user ID.
Association	For function Link children: objects of this type are to be linked to the user. Valid values: user-defined.

Add/Copy/Modify a User Screen

The screen is displayed for the Add a user function. The Copy and Modify screens are similar.

```
    13:26:31             *****  P R E D I C T  *****
2007-05-31
                                   - Add a User -
    User ID ......... USR-123
    Name ............
    Keys ..
   Zoom: N
    First Owner ID ..

    Business information
      Function ..                                  Phone ......
      Title .....                                  Extension ..
      Organiz ...                                  Mail code ..
      Usage .....     (ACC,UPD)
    User address
      Street ....                              No ....
      Zip Code ..                              City ..
      State .....
      Phone .....

    Abstract     Zoom: N



      Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
User ID	ID of the user object.
Name	The name of the user.
First owner ID	The first owner to which the user belongs can be specified. More owners can be added to the owner list: ■ by entering Y in the Owner field of the Additional attributes window. ■ with the Edit owners of a user function in the User Maintenance menu. For a complete description see *Keywords and Owners* in the *Predict Reference* documentation.

Parameters	
Business Information	Various attributes describing the user's position within the organization, telephone number and access privileges (parameter Usage with values ACCess or UPDate). The attributes are used for documentation purposes only.
User Address	Various address data for the user.

User Maintenance

Purge User

The following rules apply:

- If you confirm this function with `DELETE`, the following objects are deleted:
  - the user
  - all links to child objects
  - all links from parent objects
  - all sets created by this user
  - the workplan of the user
  - the Predict and LIST XREF profiles of the user
  - the filter definitions of the user
- A user will not be deleted with the `DELETE` option if
  - he is the only user in the user list of an owner and
  - this owner is assigned to an object where the option OWNER=FORCE has been defined in the metadata administration for this object type.
- If you confirm this function with `SCRATCH`, the following objects are deleted additionally:
  - All Owners assigned to the user are removed from the linked objects.
- A user will not be deleted with the `SCRATCH` option if
  - this would lead to all owners of an object being deleted and
  - the option OWNER=FORCE has been defined for this object type in the metadata administration.

Edit Owners of a User

```
>                    > + US: HNO                            L: 1      S: 2
 ALL          Owner ID                  Others related
    --------------------------------- - User - - Object -

    HNO                                  2          2

    RAW                                  0          7



    --------------------------------- -------- ----------
```

Meaning of Columns	
Owner ID	ID of owner.
Others related	
User	Number of other users which are related to this owner.
Objects	Number of objects except users which are related to this owner.

61 User Retrieval

- User-Specific Retrieval Parameter 492
- User-Specific Retrieval Functions 492
- Layout of User Lists 492
- Output Options for User Retrieval 493
- Output Options for User Retrieval - Continued 494

User-Specific Retrieval Parameter

User name - Limits the scope of the function to users with the name specified.

User-Specific Retrieval Functions

Users Related to Objects - Code X

Lists users and objects which are related to these users via an owner.

Command: `USED USER`

Users Related to no Object - Code Y

Lists users which are not related to any other objects in the data dictionary. The association between a user and a data dictionary object of any other type (except keyword) is always established indirectly through an owner, by associating the same owner with the user and with the other object.

Command: `UNUSED USER`

Layout of User Lists

```
13:30:17             *****  P R E D I C T  *****             2007-05-31
                              - List User -



-------------------------------------------------------------------------------
Cnt  User ID  User name                          Owner ID

   1 AAA      User1                              OW1
   2 BBB                                        >>>MULTIPLE<<<
   3 CCC      User123
   4 DDD      DDD-TEST
```

Meaning of Columns	
User ID	ID of the user object.
User Name	The name of the user.
Owner ID	Owner to which the user belongs. >>>MULTIPLE<<< indicates that the user belongs to more than one owner.

Output Options for User Retrieval

Note: Unless output mode is S, the option Cover page is always valid. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Association attributes			Y	Y	Y	Y			Y	Y	Y	Y				
Attributes	Y		Y				Y		Y				Y			
Connecting character				Y						Y						
Description	Y		Y	Y			Y		Y	Y			Y			
Display modifier	Y		Y				Y		Y				Y			
Dummy/Placeholder										Y		Y		Y		Y
Extract	Y		Y	Y			Y		Y	Y			Y	Y		
Keywords	Y		Y	Y			Y		Y	Y			Y			
Mark implementation				Y		Y				Y		Y				
No. abstract lines	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
Owner	Y		Y	Y			Y		Y	Y			Y			
With users	Y		Y	Y			Y		Y	Y			Y			
Show implementation																
Use Con-form	Y		Y	Y			Y		Y	Y			Y			
User exit	Y		Y				Y		Y				Y			
3GL specification																

Output Options for User Retrieval - Continued

Retrieval Type	U		E				C				Y	X	
Output Mode	D	L	T		X		L		D		D	L	X
Current/Related	c	c	c	r	c	r	c	r	c	r	c	c	c r
Association attributes			Y	Y									
Attributes	Y			Y		Y					Y		Y
Connecting character				Y		Y				Y			Y
Description	Y					Y				Y	Y		Y
Display modifier	Y										Y		
Dummy/Placeholder				Y		Y							Y
Extract	Y			Y		Y			Y	Y	Y		Y
Keywords	Y			Y		Y				Y	Y		Y
Mark implementation				Y		Y							Y
No. abstract lines	Y	Y		Y		Y		Y		Y	Y	Y	Y
Owner	Y			Y		Y				Y	Y		Y
With users	Y									Y	Y		Y
Show implementation													
Use Con-form	Y					Y				Y	Y		Y
User exit	Y										Y		
3GL specification													

62 Owner Maintenance

- Linking Objects Logically using Owners 496
- Owner-Specific Maintenance Functions 496

Linking Objects Logically using Owners

You can logically connect a user and another object in Predict by means of an owner, for example to document who uses an object or who is responsible for it. Enter an owner in the owner list of a user and the same owner in the owner list of the object.

The following rules apply when assigning owners:

- An owner is created by adding its ID to at least one owner list of a Predict object of type user.
- Any user can belong to several owners.
- The owner list of a dictionary object can contain up to 99 owners.

Maintaining the Owner List of an Object

These lists can be edited using the Predict Link Editor. See the section *Editors in Predict* in the *Predict Reference* documentation. The editor is invoked in one of the following ways:

- Enter `Y` in the Field Additional attributes in the bottom line of any Add, Copy or Modify screen and select Owner.
- With function Edit owners of an object in a maintenance menu.
- With command `EDIT <object-type> OWNER <object ID>`.

Disallowing or Forcing Owner Entries

The data dictionary administrator can make the adding of owners optional, prohibited or mandatory by setting the metadata administration parameter Edit owner to Allow, Disallow or Force. This parameter can be specified for each object type. If the Edit owner parameter is set to Allow, any user can specify a default to be displayed in theOwner parameter of Add/Copy/Modify screens.

Owner-Specific Maintenance Functions

Rename/Merge Owner - Code R

Owners can be renamed using the function Rename / Merge owner. After the function has been performed, the old owner will no longer exist.

```
13:36:13             *****  P R E D I C T  *****            2007-05-31
                           - Rename/Merge Owner -

Owner ID ........ HNO



Enter new owner ID .. HNOX

                       2 objects with this owner will be updated.



                 Enter '.' to return to menu.
```

If the owner name specified as the new owner already exists, the function assigns all objects of one owner to another owner. Additional confirmation is requested before this operation is carried out. ("New owner ID already exists. Move the assigned objects from one owner to another owner ID. (Y/N)").

Example

The owner Smith, who is assigned to 24 Program objects, is renamed to the existing owner Miller (because Mr. Smith accepted another assignment). Mr. Miller now has an additional 24 Programs assigned to him.

After the Rename/Merge owner function has been performed, the objects that have been updated are listed.

Purge Owner - Code E

The following rules apply when purging owners.

- The function cannot be executed if an object has only this owner in its owner list and OWNER=FORCE has been defined for this object.
- If you confirm the function with `DELETE`, the owner is deleted from the owner list of all objects.
- The number of objects affected by the DELETE option is displayed before the owner is actually purged.

63 Owner Retrieval

- Owner-specific Retrieval Functions 500
- Layout of Owner Lists 502
- Output Options for Owner Retrieval 502

```
13:37:34             *****  P R E D I C T  *****             2007-05-31
Plan   4                  - (OW) Owner Retrieval -                 Profile HNO

                    Retrieval Type

                    D  Owners
                    O  Owners with no user
                    U  Objects with no owners
                    X  Cross reference owners



Retrieval type ....
Output mode ......* S Select

Owner ID ..........


Output options  ..*   Profile HNO,used                Related type....*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Owners cannot be used as additional selection criteria (restrictions) when retrieving information on owners.

Owner-specific Retrieval Functions

Owners with no User - Code O

Lists owners which are not assigned to any user.

Direct command: `FREE OWNER`

Valid output modes: List, Select.

Objects with no Owners - Code U

Reports on objects that have no owner.

Command: `EMPTY OWNER`

Valid output modes: List, Select.

Note: It is not possible to select objects for immediate processing from lists produced with the output mode Select. Objects can however be selected for later processing from the workplan.

Cross Reference Owners - Code X

Lists all objects, that have specified owners in their owner list.

Command: `XREF OWNER`

Valid output mode: Cross reference.

```
13:39:12             *****  P R E D I C T  *****             2007-05-31
                       - Cross Reference for Owner -

Owner ID ........... BOE

-------------------------------------------------------------------------------
 Program ID ......... C-PR-P
                      Keywords
                        COO
                      Extracts
                        HEB-TEST, STK-ET-2, ARH-ET-0, BOE-ALL
                      Owner ID
                        HEB
                        ? User ID   User name
                        ?   HEB-1
                        ?   HEB-PUR   TEST
                        GER
                        ? User ID   User name
                        ?              >>> No user exists<<<
                        BOE
                        ? User ID   User name
Command ===>                                                   Scroll ==> CSR

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
                  Quit        RFind Flip  -     +           Left  Right
```

Layout of Owner Lists

```
13:40:07              *****  P R E D I C T  *****              2007-05-31
                                - List Owner -


-------------------------------------------------------------------------------
Cmd  Owner ID                           User ID  User name

   1 B                                            >>> No userexists <<<
   2 BER                                BER

   3 BOE                                BOE1
                                        BOE2
   4 BOE-OW01                           BOE3
```

Meaning of Columns	
Owner ID	ID of owner.
User ID	IDs of the persons the owner represents.
User name	Names of the persons the owner represents. In select lists >>>MULTIPLE<<< is displayed if an owner is assigned to more than one user.

Output Options for Owner Retrieval

Note: Unless output mode is S, the option Cover page is always valid. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.

Retrieval Type	D	O	U		X	
Output Mode	L	L	L		X	
Current/Related	c	c	c	r	c	r
Attributes						Y
Connecting character						Y
Description						Y
Dummy/Placeholder						Y
Extract						Y
Keywords						Y
Mark implementation				Y		Y
No. abstract lines				Y		Y

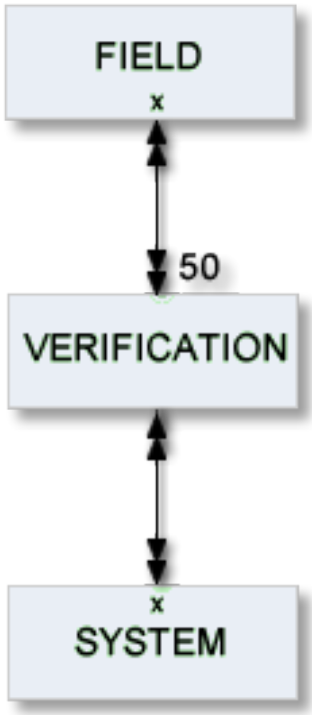
Retrieval Type	D	O	U		X	
Output Mode	L	L	L		X	
Current/Related	c	c	c	r	c	r
Owner						Y
With users						Y
Sorted by field				Y		Y
Use Con-form						Y

XV Verification

Objects of type Verification can contain code for processing rules. Verifications can have as status: documented, conceptual, free, automatic, Natural Construct or SQL.

In the predefined Predict metastructure, verifications can have passive and active associations of the following types:

Valid passive associations:	"Verifies EL" "Is comp. of SY"
Valid active association:	No predefined association



The description of object type Verification is organized under the following headings:

Maintaining Objects of Type Verification
Verification-Specific Maintenance
Verification-Specific Retrieval

Additional Information on Verifications/Processing Rules

- See the section *Verifications and Processing Rules* in the *Predict and Other Systems* documentation.
- The editor used to modify processing rules is described in the section *Editors in Predict* in the *Predict Reference* documentation.
- See also *Rippling Verifications* in the *Predict and Other Systems* documentation.

64 Maintaining Objects of Type Verification

- Verification Maintenance Menu 508
- Verification Status 509
- Verification Formats 509
- Add a Verification Screen 510

Verification Maintenance Menu

The Verification Maintenance menu is called with function code M and object code VE in a main menu or with the command `MAINTAIN VERIFICATION`.

```
09:28:30              *****  P R E D I C T  *****             2007-05-31
Plan   3              - (VE) Verification Maintenance -             Profile HNO

 Function                                Function

 A  Add a verification                   D  Display verification
 C  Copy verification                    L  Link children
 M  Modify verification                  S  Select verification from a list
 N  Rename/change status verific.
 P  Purge verification

Function .........

Verification ID ..                                   Attributes ......*
Copy ID ..........                                   Status ..........*
                                                     Format ..........*


Restrictions ....*   Profile HNO,used                Association .....*

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Parameters	
Function	Select a code from the menu to execute one of the maintenance functions. Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation. The functions Purge Verification, Rename/Change Status of a Verification and Edit Rule of a Verification (Code R) are described in the section ***Verification-Specific Maintenance***.
Verification ID	Identifier of the Predict verification object. The identifier of a verification is checked against Natural naming conventions.
Copy ID	For Copy function: ID of new verification to be created.
Status	Status of the verification. See ***Verification Status*** below. For the Select function: a status can be specified as an additional selection criterion.
Format	Format of the verification. See ***Verification Formats*** below. For the Select function: a format can be specified as an additional selection criterion.

Parameters	
Restrictions	Additional criteria can be specified to restrict the scope of verifications to be processed. See *Restrictions* in the section *Predict User Interface* in the *Introduction to Predict* documentation.

Verification Status

The table below contains a list of all valid verification status.

Code	Verification Status
A	Automatic
C	Conceptual
D	Documented (no rule)
F	Free
N	Natural Construct
S	SQL

Verification Formats

The table below contains a list of all valid verification formats.

Code	Format
A	Alphanumeric
B	Binary
D	Date/time
K	Function key
L	Logical
N	Numeric
blank	Unknown (no rule defined)

Add a Verification Screen

The screen is displayed for the Add a Verification function. The Copy and Modify screens are similar.

```
09:27:57             *****  P R E D I C T  *****             2007-05-31
                          - Modify Verification -
Verification ID . HNO-VE
Status .......... Documented
Keys ..                                                          Zoom: N

Format .........*    (none)                                Modifier    Zoom: N
Type ...........*    All
Message nr ......
Replacement 1 ...
Replacement 2 ...
Replacement 3 ...
Message text ....

Abstract     Zoom: N             Values     Zoom: N



Additional attributes ..* N          Associations ..* N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Verification ID	The identifier of the verification.
Status	The status assigned by Predict to the verification rule. See *Verification Status* for list of valid values.
Format	The format of the verification rule. See *Verification Formats* for list of valid values.
Modifier	User and or user groups defined in Natural Security who can be authorized to modify free rules of the verification. The parameter is evaluated by Predict according to the setting of the default parameters Rule in Map Editor / Rule in SYSDIC. If any of these parameters is set to force, Predict checks the following: ■ that at least one modifier is specified, ■ that each modifier of the object is a Natural Security administrator, person or group, ■ that the user is listed as a modifier of the object.

Parameters	
	See also description of Rule in Map Editor / Rule in SYSDIC in the section *Defaults* in the *Predict Administration* documentation and *Protecting Processing Rules* in the section *Protecting External Objects in Predict with Natural Security* in the *Predict Security* documentation.
Type	The type of rule. Enter a single-character code as shown in the following table. The table also shows the number of values to be specified with each type of rule:

Code	Type of Rule	No. of Values
E	Equal to	1
		0 or 1 for format logical
G	Greater than	1
L	Less than	1
N	Not equal to	n
		0 or 1 for format logical
R	Range of values	2
T	Table of values	n
U	User routine	
B	Range, but not	3 or 4
I	Not in range	2
M	Mask	n
O	Not Equal Mask	n
S	Scan	n
V	Not Equal Scan	n
blank	(none) - no rule defined	

For a list of the generated code, see *Rule Editor* in the *Predict Reference* documentation. See also ***Edit Rule of a Verification***.

Message nr	Number of Natural error message. The message will be displayed if a validation fails. Up to three replacement strings can be inserted into an error message if the respective targets (:1:, :2:, :3:) are provided.
Replacement 1 - 3	Strings to be inserted into a Natural message. See description of Message nr above.
Message text	Message to be displayed if a validation fails. A standard message will be created if neither Message text nor Message nr have been specified.

Values	The values used to perform the verification. The following rules apply:
	■ The number of values to be specified depends on the verification type. See table above.
	■ Values are delimited
	  ■ with blanks
	  ■ with the Natural INPUT delimiter character (ID) defined in the Natural environment
	  ■ by entering them in separate lines.

Parameters	
	■ Hexadecimal values can be specified in two ways: ■ if Format=B, hexadecimal values can be specified directly. Example: F0 ■ if Format=A, hexadecimal values must be preceded by uppercase X or H and be enclosed in single quotes. Example: X'F0' or H'F0' ■ Blanks can be specified in one of the following ways: ' ', BLANK or SPACE. Strings that include blanks must be enclosed in single quotes, apostrophes in strings have to be doubled (for example: 'six o''clock'). ■ Line comments can be specified by preceding them with /* (a slash and an asterisk). Line comments can be used by SYSHELP as descriptive text in input windows. Strings that include the comment delimiter /* must be enclosed in single quotes.

65 Verification-Specific Maintenance

- Purge Verification - Code P 514
- Rename/Change Status of a Verification - Code N 514
- Edit Rule of a Verification - Code R 514
- Rule Editor 515

Purge Verification - Code P

A verification of type automatic cannot be purged. To purge a verification of this type, perform the following steps:

- Remove all links from fields to the verification
- Regenerate DDMs that were generated from the files linked to these fields.

When the verification is no longer connected to any fields, the status is changed to conceptual and the rule can be purged.

Rename/Change Status of a Verification - Code N

Changing the status of a verification is limited. For further information see *Changing the Status of a Verification* in the section *Verifications and Processing Rules* in the *Predict and Other Systems* documentation.

Edit Rule of a Verification - Code R

Processing rules of verifications are edited with the Predict Verification Editor. This editor can be invoked in one of the following ways:

- Enter Y in the field Additional attributes in the bottom line of the Add a Verification, Copy Verification or Modify Verification screen and select Rule code.
- Call the function Edit rule in the Verification Maintenance menu (Code R).
- Enter the direct command `EDIT VERIFICATION RULE <Verification-ID>`

Note: Statements of the rule must not contain statement references to line numbers; use labels instead.

Rule Editor

For information on the Rule Editor, see the *Predict Reference* documentation. General editor commands are described in the section *Editors in Predict* in the *Predict Reference* documentation.

66 Verification-Specific Retrieval

- Verification-Specific Retrieval Parameters 518
- Verification Specific Retrieval Functions 518
- Layout of Verification Lists 519
- Output Options for Verification Retrieval 519
- Output Options for Verification Retrieval - Continued 520

Verification-Specific Retrieval Parameters

verif. of status	Limits the scope of the function to verifications with the status specified. Valid values:	
	A	Automatic
	C	Conceptual
	D	Documental (no rule)
	F	Free
	S	SQL
	N	Natural Construct
format	Limits the scope of the function to verifications with the format specified. Valid values:	
	A	Alphanumeric
	B	Binary
	D	Date/time
	K	Function key
	L	Logical
	N	Numeric

Verification Specific Retrieval Functions

List Verifications to Regenerate - Code K

Lists verifications whose definitions have been modified since a DDM was generated containing a field that uses one of the verifications.

Direct command: `REGENERATE VERIFICATION`.

Layout of Verification Lists

```
13:13:19             *****  P R E D I C T  4.2.2   *****            2007-05-31
                           - List Verification -


--------------------------------------------------------------------------------
Cnt  Verification ID                    S F Comp. F T

   1 HNO-PR1                            D           U
   2 HNO-S                              D           U
   3 * HNO-VE1                          A A B       E
     Verification values
       Samstag
   4 HNO-VE2                            D A         U
   5 JP-TEST                            F A B       E
     Verification values
       9
```

Meaning of Columns	
Verification ID	ID of the Predict verification object.
S	The status of the verification rule. See *Verification Maintenance Menu* for list of codes and values.
F	The format of the verification rule. See *Verification Maintenance Menu* for list of codes and values.
Comp. F	Compatible format. Not all formats are compatible with all verification types.
T	Type of the verification. See the description of Type in *Add a Verification Screen*.
Values	Verification values (Only applicable for output mode select).

Output Options for Verification Retrieval

Notes:

1. Unless output mode is S, the option Cover page is always valid.
2. Page size is only applicable when printing or if general default parameter `Use SAG Editor for output` is set to Y. Page size is not applicable in batch mode.

Retrieval type	D		B				O		T							
									dummies=Y|N				dummies=D|P			
Output Mode	D	L	D		L		D	L	D		L		D		L	
Current/Related	c	c	c	r	c	r	c	c	c	r	c	r	c	r	c	r
Association attributes			Y	Y	Y	Y			Y	Y	Y	Y				
Attributes	Y		Y				Y		Y				Y			
Connecting character				Y						Y						
Description	Y		Y	Y			Y		Y	Y			Y			
Display modifier	Y		Y				Y		Y				Y			
Dummy/Placeholder										Y		Y		Y		Y
Extract	Y		Y	Y			Y		Y	Y			Y	Y		
Keywords	Y		Y	Y			Y		Y	Y			Y			
Mark implementation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
No. abstract lines	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	
Owner	Y		Y	Y			Y		Y	Y			Y			
With users	Y		Y	Y			Y		Y	Y			Y			
Rules	Y		Y				Y		Y				Y			
Show implementation	Y		Y				Y		Y				Y			
Sorted by field*				Y		Y										
Use Con-form	Y		Y	Y			Y		Y	Y			Y			
User exit	Y		Y				Y		Y				Y			

* Only evaluated for association EL (Verifies EL)

Output Options for Verification Retrieval - Continued

Retrieval Type	U		E				C				K	
Output Mode	D	L	T		X		L		D		L	
Current/Related	c	c	c	r	c	r	c	r	c	r	c	c
Association attributes			Y	Y								
Attributes	Y			Y		Y						
Connecting character				Y		Y				Y		
Description	Y					Y				Y		
Display modifier	Y											
Dummy/Placeholder				Y		Y	Y		Y			

Retrieval Type	U		E				C				K	
Output Mode	D	L	T		X		L		D		L	
Current/Related	c	c	c	r	c	r	c	r	c	r	c	c
Extract	Y			Y		Y			Y	Y		
Keywords	Y			Y		Y				Y		
Mark implementation	Y	Y	Y	Y	Y	Y		Y			Y	
No. abstract lines	Y	Y		Y		Y		Y		Y	Y	
Owner	Y			Y		Y				Y		
With users	Y									Y		
Rules	Y											
Show implementation	Y											
Sorted by field												
Use Con-form	Y					Y				Y		
User exit	Y											

XVI Virtual Machine

67 Virtual Machine

- Virtual Machine Maintenance Menu 527
- Add a Virtual Machine Screen 528
- Virtual Machine Retrieval 528

Since data can be distributed across several databases, the exact location of data storage has to be specified: databases are linked to objects of type Virtual Machine and virtual machines are linked to objects of type Network.

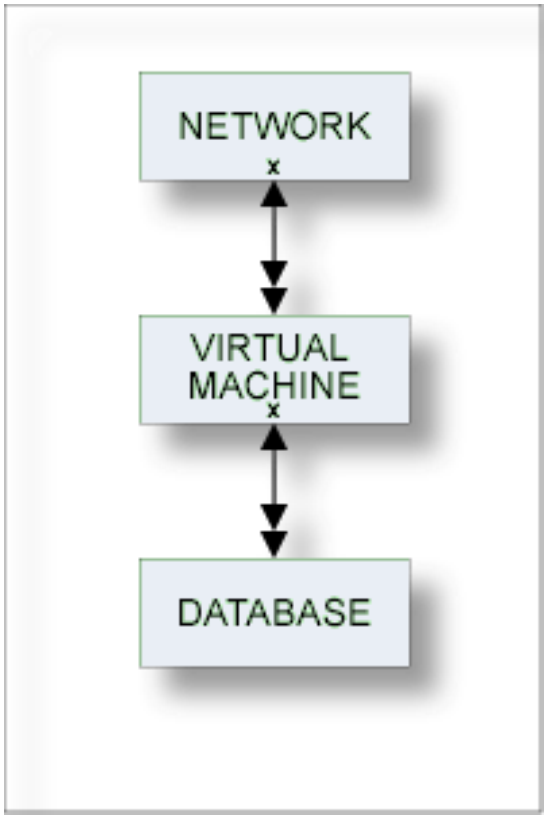
The Predict object virtual machine identifies the hardware and operating system environment of a database.

See the section *Adabas Vista* in the *Predict and Other Systems* documentation for a complete description of how to define distributed data structures with Predict.

In the predefined Predict metastructure, a virtual machine can have passive and active associations of the following types:

Valid passive association:	"Belongs to VM" (mandatory, default parent)
Valid active association:	"Contains DA"

Note: Links between networks, virtual machines and databases are established with the parameters "Belongs to NW" and "Belongs to VM", and not with active/passive associations.



Virtual Machine Maintenance Menu

The Virtual Machine Maintenance menu is called with function code M and object code VM in a Predict main menu or the command `MAINTAIN VIRTUALMACHINE`.

```
13:22:38             *****  P R E D I C T  *****             2010-05-31
Plan   0            - (VM) Virtual machine Maintenance -           Profile SYS

 Function                                Function

 A  Add a Virtual machine                D  Display Virtual machine
 C  Copy Virtual machine                 L  Link children
 M  Modify Virtual machine               S  Select Virtual machine from list
 N  Rename Virtual machine
 P  Purge Virtual machine


Function ..............

Virtual machine ID ....                                   Attributes....*
Copy ID ...............
Belongs to NW .........

Restrictions .........*   Profile SYS,used              Association ...*

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Next  Stop  Last  LnkEl Flip  Print Impl  AdmFi SelFi Prof  Main
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Function	Standard maintenance functions are described in the section *Maintenance* in the *Predict Reference* documentation.
Belongs to NW	ID of the network containing the virtual machine.

Add a Virtual Machine Screen

The screen is displayed for the Add a Virtual Machine function. The Copy and Modify screens are similar.

```
13:43:32             *****  P R E D I C T  *****            2010-05-31
                         - Add a Virtual machine -
Virtual machine . SYS-VM
Belongs to NW ..*
Keys ..                                                          Zoom: N

Attributes
 Operating system ..*
 Abstract    Zoom: N
```

Note: Parameters not listed here are described under *Global Attributes*.

Parameters	
Virtual machine	The ID of the virtual machine.
Belongs to NW	The ID of the network containing the virtual machine.
Operating system	Enter * (asterisk) for a list of valid values.

Virtual Machine Retrieval

Virtual Machine Specific Retrieval Parameters

" Belongs to NW" - only virtual machines related to the network will be included in the report.

Layout of Virtual Machine Lists

```
13:13:46             *****  P R E D I C T  *****             2010-05-31
                           - List Virtual machine -


-------------------------------------------------------------------------------
Cnt  Virtual machine ID                Operating system

   1 ARH-VM2
   2 ARH-VM4                           MVS
   3 BOE-TEST-1
   4 BOE-VM
   5 BOE-VM-01
   6 BOE-VM01                          MVS/XA
```

Meaning of Columns	
Operating system	Operating system type of the virtual machine.

Output Options for Virtual Machine Retrieval

The output options valid for this object type are identical to those for object type dataspace. See *Output Options for Dataspace Retrieval*.