

Predict

Security

Version 8.5.2

October 2022

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-SECURITY-852-20221008

Table of Contents

| | |
|---|-----|
| Preface | vii |
| 1 About this Documentation | 1 |
| Document Conventions | 2 |
| Online Information and Support | 2 |
| Data Protection | 3 |
| I | 5 |
| 2 Introduction | 7 |
| Terminology | 8 |
| Predict Security using Natural Security | 9 |
| Internal Check Routine | 10 |
| Security Conflicts | 11 |
| Command DISSEC | 11 |
| 3 Setting Up Your Security Environment | 13 |
| Step 1 Define Predict Libraries in Natural Security | 14 |
| Step 2 Define Predict Users in Natural Security | 14 |
| Step 3 Define Predict Defaults | 14 |
| Step 4 Add Natural Security Definitions | 15 |
| Step 5 Protect Predict Functions - Recommended | 15 |
| Step 6 Set Protection Flag | 16 |
| 4 Natural Security Entities | 17 |
| Conceptional Data Model - Extract | 18 |
| Instance | 18 |
| User | 19 |
| Group | 21 |
| Library | 22 |
| NSC External Object Types | 23 |
| 5 Security Profiles | 37 |
| General Information | 38 |
| Defining Access Rights | 38 |
| II Handling Protected Objects in Predict | 49 |
| 6 General Information | 51 |
| General Security Concept in Predict | 52 |
| Sample Function | 53 |
| Response Times | 54 |
| Security Check in the Main Menu | 54 |
| 7 Maintenance | 57 |
| Standard Maintenance Functions | 58 |
| Type-Specific Maintenance functions | 60 |
| 8 Retrieval | 63 |
| List-Oriented Functions | 64 |
| Display-Oriented Functions | 67 |
| Attribute-Oriented Functions | 69 |
| 9 Active Retrieval | 73 |

| | |
|---|-----|
| 10 LIST XREF | 77 |
| Save Set | 78 |
| 11 LIST XREF for 3GL | 79 |
| 12 Generation | 81 |
| Checks for all Generation Functions | 82 |
| Security Definitions in Natural Security at Object Type Level | 82 |
| Type-Specific Security Checks | 82 |
| PUNCH | 85 |
| 13 File Implementation | 87 |
| Additional Checks with File Implementation Functions | 88 |
| Execute implementation plan | 88 |
| Add / Extend implementation plan | 88 |
| Extend implementation plan | 89 |
| Modify / Display implementation plan | 89 |
| Display Implementation Plan | 89 |
| 14 Preprocessor | 91 |
| 15 Incorporation | 93 |
| Checks for all Incorporation Functions | 94 |
| Additional Checks for individual Incorporation Functions | 94 |
| 16 Comparison | 97 |
| Security Checks for all Comparison Functions | 98 |
| Function-specific Security Checks | 98 |
| 17 Administration | 99 |
| Security Checks for all Administration Functions | 100 |
| 18 Defaults, Special Functions | 103 |
| 19 Coordinator | 105 |
| Security Checks when working with different FDICs | 106 |
| Security Checks at Function Level | 106 |
| No Security Protection for Coordinator FDIC | 107 |
| Security Definitions at Object Level | 107 |
| 20 Metadata Administration | 109 |
| UDEs | 110 |
| 21 Conversion | 111 |
| III | 113 |
| 22 Interfaces To Other Software AG Products | 115 |
| Adabas Native SQL | 116 |
| API | 116 |
| Natural | 116 |
| Predict Application Control | 116 |
| Super Natural | 117 |
| SYSAOS | 117 |
| SYSDB2 | 117 |
| SYSHELP | 117 |
| 23 Protecting Predict Programs With Natural Security | 119 |
| Functional Scope | 120 |

| | |
|---|-----|
| Naming Conventions for Library SYSDIC | 120 |
| Protecting Programs in Library SYSDICBE | 124 |
| Naming Conventions for Library SYSDICCO | 125 |
| Naming Conventions for Library SYSDICMA | 126 |
| 24 Protecting External Objects in Predict With Natural Security | 129 |
| Protecting Adabas Databases and Files | 130 |
| Protecting DDMs | 136 |
| Protecting Processing Rules | 137 |
| Protecting Natural Source Programs | 138 |
| 25 Protecting Predict With Other Security Systems | 139 |
| Protecting Predict using Adabas Security | 140 |
| Protecting Predict with User Exit U-SEC | 140 |

Preface

This document covers the following topics:

| | |
|---|--|
| Introduction | Explains the basic concepts of Predict Security and the prerequisites that must be met in order to work with Predict Security successfully. A glossary at the start of this section contains the most important terms used in this document. |
| Setting up your Security Environment | All the steps required in order to protect your environment with Predict Security. |
| Natural Security Entities | Describes the entities in Predict that can be protected with security definitions in Natural Security. |
| Security Profiles | Describes how access rights are defined in Natural Security. |
| Handling Protected Objects in Predict | Describes in detail how individual Predict functions behave when you try to access protected objects. |
| Interfaces | Interfaces to other Software AG Products. |
| Protecting Predict Programs with Natural Security | How to protect individual Predict system programs using Natural Security. |
| Protecting External Objects in Predict with Natural Security | How to protect Adabas Databases and Files, DDMs, processing rules and Natural source programs using Natural Security. |
| Protecting Predict with other Security Systems | How to protect your Predict environment with Adabas Security, and points to consider when protecting your environment with external security systems such as RACF. |

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 <i>folder.subfolder.service</i> , APIs, Java classes, methods, properties. |
| <i>Italic</i> | 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.software-ag.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.

I

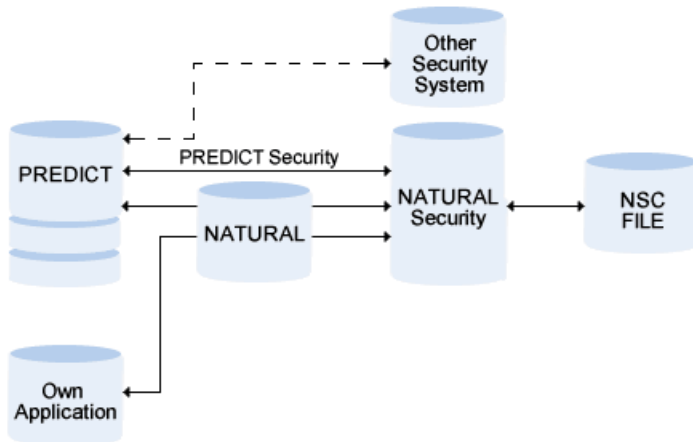
| | |
|--|----|
| ■ 2 Introduction | 7 |
| ■ 3 Setting Up Your Security Environment | 13 |
| ■ 4 Natural Security Entities | 17 |
| ■ 5 Security Profiles | 37 |

2 Introduction

| | |
|---|----|
| ■ Terminology | 8 |
| ■ Predict Security using Natural Security | 9 |
| ■ Internal Check Routine | 10 |
| ■ Security Conflicts | 11 |
| ■ Command DISSEC | 11 |

The Predict Security System controls access to a Predict environment using security definitions stored in a Natural Security file.

An individual environment can be defined for each user or group of users and protected against unauthorized access.



Terminology

Link ID

When a user logs on to a library, the Link ID is determined as follows:

- If the user is linked directly to the library, the link ID is the same as the user ID.
- If the user is linked to the library indirectly as member of a group, the link ID is the group ID.

See your *Natural Security documentation* for more information.

Natural Security File

This file contains the security definitions used by the Predict Security System for protecting objects against unauthorized access.

NSC External Object

In terms of Predict security, an instance of an external object type.

NSC External Object Type

In terms of Predict security, an external object type is a class of objects to be protected. The following external object types are available for protecting Predict data:

- PRD-Docu-Object
- PRD-Ext-Object
- PRD-Function
- PRD-3GL-Library

Predict Security

Predict Security means the following:

- the external object types in Natural Security, for example PRD-Docu-Object
- the security profiles defined to limit access to instances of external object types.

Security Object

A security definition in Natural Security.

Predict Security using Natural Security

Predict Security is realized with the Software AG product Natural Security. This product allows you to

- define the persons who can process the protected objects
- define the objects to be protected

Predict transfers the administrative functions listed above to Natural Security. A security object is a security definition valid for one of the following data types:

- **Documentation objects**
 - All object types
 - Documentation object types, either predefined or user-defined
 - Subtype of a predefined object type (for example file of type Adabas)
 - All objects of an object type or subtype

- Range of documentation objects (for example all files that start with USER1)
- Fully qualified documentation object (for example file USER1-FI-ADA2)
- **Special Objects**
 - Retrieval Model
 - Association Type
 - Object Type
 - Implementation Plan
- **External Objects**
 - External object type (for example DDMs)
- **Predict Functions**
 - Entire group of functions, for example Special Functions
 - Individual functions, for example Special Function Reposition implementation data.
- **XRef data**
 - XRef data in all 3GL libraries
 - XRef data in a range of 3GL libraries
 - XRef data in a fully qualified 3GL library

Internal Check Routine

When a Predict function is called, an internal Predict routine generates an Natural Security call. This call checks the security definitions in Natural Security for

- the current link ID
- the function the user wants to execute
- the data the user wants to access.

Activating Natural Security via Parameter in Predict

The Security Check does not depend on whether Natural Security is installed. The Predict parameter Protect current Predict file in the General Defaults > Protection screen determines whether Predict Security is called. This parameter can be defined for each FDIC file.

Security Check when Calling Predict

The system checks whether the current user is authorized to logon to library SYSDIC.

Security Conflicts

If the user does not have the necessary access rights, the system behavior depends on the type of security conflict:

- If the user does not have access to a function, the function is not executed and an error message appears.
- If the user executes a retrieval function and does not have READ access to certain objects or object types, the system behavior depends on whether the retrieval function was implicitly or explicitly limited by attributes or restrictions:
 - Retrieval function not limited by restrictions or attributes: Only the ID of the read-protected object with the remark >>>protected<<< is output.
 - Retrieval function limited by restrictions or attributes: Objects are suppressed completely.

See [Retrieval](#).

- If the user executes a function other than retrieval and has insufficient access to an object or object type, the function is not executed and an error message is given.

Example: User has MODIFY but no DELETE access to objects of type file. He cannot delete files.

For more information see the section [Handling Protected Objects in Predict](#) in this documentation.

Command DISSEC

With the command DISSEC, users can display their own individual security definitions that have been defined for them by the Security Administrator. This command can be entered in the command line from any point within Predict.

This command corresponds to the Special Function Maintain NSC Definitions > Display NSC Definitions. See Maintain NSC Definitions in the section *Special Functions* in the *Predict Administration documentation*.

3

Setting Up Your Security Environment

| | |
|---|----|
| ■ Step 1 Define Predict Libraries in Natural Security | 14 |
| ■ Step 2 Define Predict Users in Natural Security | 14 |
| ■ Step 3 Define Predict Defaults | 14 |
| ■ Step 4 Add Natural Security Definitions | 15 |
| ■ Step 5 Protect Predict Functions - Recommended | 15 |
| ■ Step 6 Set Protection Flag | 16 |

Predict definitions are not protected in Natural Security as default. This means that when Predict is delivered, each user has access to every object and can execute any Predict function. Predict Security only takes effect when access to objects, object types or functions is explicitly restricted for individual users or groups of users. This section lists the steps necessary to set up your environment so you can protect objects and functions against unauthorized access.

Step 1 Define Predict Libraries in Natural Security

Libraries SYSDIC, SYSDICBE, SYSDICCO and SYSDICMA must be defined as libraries in Natural Security. See [Library](#).

Step 2 Define Predict Users in Natural Security

Predict users must be defined as Users in Natural Security. See [User](#).

New users can be added manually with the function Add user in Natural Security.

Existing Natural Security users can be incorporated in Predict with the function Incorporate Natural Security user. See the section *Incorporation* in the *External Objects in Predict documentation*.

The users must be authorized to logon to the library SYSDIC in Natural Security. See your Natural Security documentation for more information.

Step 3 Define Predict Defaults

Enter database and file number of the NSC file in the Defaults > General Defaults > Protection screen.



Note: Do *not* set the parameter Protect current Predict file to Y at this point! You must first add the Natural Security definitions as described in Step 4 below.

Step 4 Add Natural Security Definitions

Default Definitions (Mandatory)

Add the standard definitions for Natural Security with the Special Function Maintain NSC Definitions > Add NSC Default Definitions.

See Maintain NSC Definitions in the section *Special Functions* in the *Predict Administration documentation*.

Additional Security Definitions Individually (Optional)

If you wish, you can create your own Security definitions for any object in Natural Security - either for an individual user or for a group of users. See [Defining Access Rights](#).

Adding Security Definitions with Special Function Mass Grant (Optional)

In earlier versions of Predict, information such as keywords or owners was evaluated to restrict access to objects. many customers used this method to adapt their environment to the particular security requirements of their company.

If you want to use these definitions, you can create an Extract where the corresponding keywords or owners are used as retrieval criteria, then with the special function Mass Grant in NSC you can create security definitions in Natural Security for all Predict objects contained in this extract. See Mass Grant in NSC in the section *Special Functions* in the *Predict Administration documentation*.

Step 5 Protect Predict Functions - Recommended

To prevent users changing Security parameters, we recommend you protect the following function in Natural Security for normal Predict users.

| NSC Security Object | Function |
|--------------------------|---|
| DEFAULTS-GENERAL-P | General Default function Protection |
| SPECIAL-MAINTAIN-NSC-DEF | Special function Maintain NSC Definitions |

Step 6 Set Protection Flag

After you have added Natural Security default definitions, set the parameter Protect current Predict file in the General Defaults > Protection screen to Y to activate Predict Security. This parameter can be defined for each FDIC file.

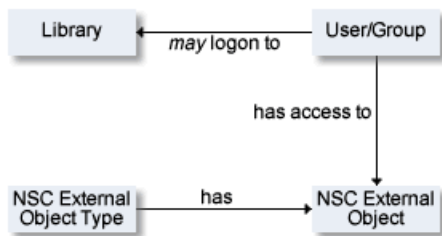
4

Natural Security Entities

| | |
|---|----|
| ■ Conceptual Data Model - Extract | 18 |
| ■ Instance | 18 |
| ■ User | 19 |
| ■ Group | 21 |
| ■ Library | 22 |
| ■ NSC External Object Types | 23 |

This chapter describes the entities in Natural Security that are used for security definitions in Predict.

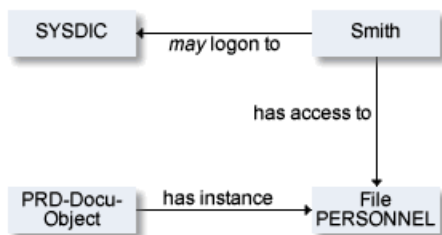
Conceptual Data Model - Extract



Instance

This example illustrates the following situation:

The user Smith is authorized to logon to library SYSDIC and has access to file object PERSONNEL, an instance of NSC external object type PRD-docu-object.



User

General Rules

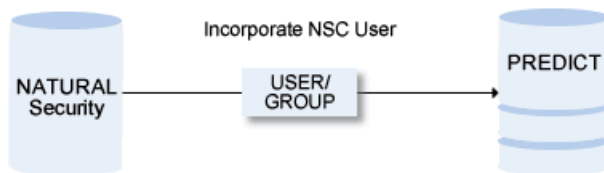
- A user is the central point of the Predict Security system. This object represents a person that works with the system.
- Users can be a member of one or more groups. See [Group](#).
- When a user logs on to a library, the link ID is determined as follows:
 - If the user is linked directly to the library, the link ID is the same as the user ID.
 - If the user is linked to the library indirectly as member of a group, the link ID is the group ID.

See your *Natural Security documentation* for more information.

Incorporating Users from Natural Security



Note: See also Concepts of Incorporation *External Objects in Predict documentation* for basic information on how to use incorporation functions



New users can be added manually with the function Add user in Natural Security.

Prerequisites and Restrictions

Only a Natural Security System Administrator can incorporate a Natural Security user.

Selecting Natural Security Users

The Incorporate Natural Security user screen is displayed by selecting function code I and object code NS in a Predict main menu or by entering the direct command INCORPORATE Security.

```
13:32:58          ***** P R E D I C T *****          2007-05-31
- Incorporate Natural Security User -
```

```
User ID.....
```

```
Incorporation options
```

```
Add user..... N (Y/N)
From date..... 0000-00-00
User type.....*
with comments..... Y (Y/N)
with edit description N (Y/N)
```

| Parameters | |
|-----------------------|---|
| User ID | ID of the Natural Security user to be processed. Asterisk notation is allowed. |
| Add user | Y Natural Security users that are not defined in Predict will be added to it. |
| From date | Limit the incorporation to user IDs which were added to the Natural Security system after the specified date. |
| User type | The type of user defined in Natural Security: A Administrator M Member P Person blank any |
| with comments | Y User ID comments in the Natural Security system will be copied to Predict. Each comment line will be split and stored as two halves. |
| with edit description | Y User ID comments in the Natural Security system will be copied to the extended description. |

Incorporating Natural Security Users in Batch Mode

Command: INCORPORATE SECURITY

Enter parameters on next line in positional or keyword form.

| Field | Keyword | Position |
|-----------------------|----------|----------|
| User | USER-ID | 1 |
| Add user | ADD-USER | 2 |
| from date | DATE | 3 |
| User type | TYPE | 4 |
| with comments | COMMENT | 5 |
| with edit description | DESC | 6 |

To incorporate Natural Security administrators whose names start with 'A', code the command:

```
INCORPORATE SECURITY
USER-ID=A*,ADD-USER=Y,TYPE=A
↵
```

or in positional form:

```
INCORPORATE SECURITY
A*,Y,,A
↵
```

The example above uses the Natural parameters IA==, ID=, and IM=D

Group

General Rules

- A group in Natural Security is a collection of users. See [User](#).
- The number of users in a group is unlimited.
- Groups have no relationships with other groups. This means a group cannot be part of another group.
- A user can belong to several groups.
- New groups can also be added in Natural Security with the function Add user.

Library

To use Predict Security, the libraries SYSDIC, SYSDICBE, SYSDICCO and SYSDICMA must be defined as libraries in Natural Security.

Defining Predict Libraries to Natural Security

Use the Natural Security function Add Library with the following parameters:

library ID = SYSDIC (for example)
startup transaction = MENU
restart transaction = blank
error transaction = blank

If the use of Natural commands from Predict is to be disallowed, the Predict libraries must be defined in Natural Security accordingly. Use the Natural Security function Modify Predict library > Additional options > Restrictions > Security options to define the following values for parameter Definition of command mode:

Allow NEXT, MORE line = N
Allow System commands = Y

The Natural commands CHECK, LIST, HELP, RETURN, SAVE and RUN are used by Predict internally and must therefore be allowed.

We recommend you set the parameters MADIO and MAXCL to 0 (zero).

Creating XRef Data for Startup, Restart or Error Transactions Defined in Natural Security

Natural Security writes XRef data for programs that are used as startup, restart or error transactions in a library or a special link: this requires that the parameter XREF is set to Y or F in the Natural Security definition of the libraries and that a user system file is defined for the library.

In XRef data, programs used as startup, restart or error transactions are indicated as used by a dummy program *NSC.

If XRef data for startup, restart or error transaction has not yet been created, you can use the conversion function Add Natural Security XRef data. For more information see Add Natural Security XRef Data in the section *Conversion* in the *Predict Administration documentation*.

NSC External Object Types

An NSC external object type is a group of things to be protected, such as objects or functions.

Adding NSC External Object Types for Predict

These NSC external object types and their standard definitions are added in Natural Security with the special function Maintain NSC Definitions > Add NSC Default Definitions.

■ **PRD-Docu-Objects**

Definitions for this NSC external object type are automatically added in Natural Security. For example: FI and FI-A for files or files of type Adabas. See list of Resources for predefined object types in [Adding Predefined Object Types](#).

User-defined object types are also added with this function. Security definitions for instances of this external object type, for example files that begin with ABC, must be added manually in Natural Security. See [Security Definitions at Object Level](#).

■ **PRD-Ext-Objects**

For this NSC external object type, the instances are automatically added in Natural Security.

■ **PRD-3GL-XRef library**

If you wish to protect 3GL libraries, you must define security objects of this type manually in Natural Security.

■ **PRD-Function**

For this NSC external object type, the instances are automatically added in Natural Security. See Maintain NSC Definitions in the section *Special Functions* in the *Predict Administration documentation*.

Access to NSC External Object Types

The default value for all NSC external object types is allowed. If you keep the default value as allowed and do not add any security definitions, each user can execute any function and access any documentation or external object.

If you set the default value for a NSC external object type to disallowed, you have to give each user or group explicit access to all instances of this NSC external object type they needs for their work.

Access Modes

- The following access can be given for instances of all NSC external object types except PRD-Function:
 - READ
 - ADD
 - MODIFY
 - DELETE

The following rules apply:

- If a user does not have READ access, he cannot be granted ADD/MODIFY/DELETE access.
- READ access is a prerequisite for access modes ADD, MODIFY and DELETE.
- The following access can be given for instances of NSC external object type PRD-Function:
 - EXECUTE

Access Mode Values

Possible values for the access modes listed above are:

Y

Access is granted

N

Access is denied

*

Inherit. The security definition of the higher-level object is taken if appropriate.

PRD-Docu-Object

Resources of NSC external object type PRD-Docu-Object can be divided up as follows:

- **Predefined Object Types**
- **User-defined Object Types**
- **Special object types**
- **Security Definitions at Object Level** for all of the above object types.

Security definitions on this level are usually intended to protect functions. Example: If the user does not have any READ access to object type FI, he cannot execute any Predict functions that process files. He cannot call the File Maintenance or Retrieval Menu.

See also [Hierarchy of Security Definitions](#).

Predefined Object Types

General Rules

- This group includes main object types (for example FI, PR) as well as subtypes of predefined object types (for example FI-A for files of type Adabas).
- If you deny a user or group access to a *main object type* such as FI or PR, it is not possible to call the maintenance or retrieval menu of this type.
- In Predict Security, object type field is regarded as an attribute of object type file.

Adding Predefined Object Types

With the Special Function Maintain NSC Definitions you can add all object types and all subtypes as instances of NSC external object type PRD-Docu-Objects automatically:

- For main object types, the name of the Resource consists of the two-character object code in Predict, for example DA, FI.
- For subtypes, the name of the Resource consists of the two-character object code, a hyphen and the one or two-character code for the subtype, for example DA-A (Database of type Adabas, FI-B (File of type Adabas SQL view). See list below.

| Object Code | Object Type | Subtype Code | Subtype |
|-------------|-----------------------|--------------|---------------------|
| DA | All database types | DA-A | Adabas database |
| | | DA-B ∷ | Adabas D handler |
| DC | Dataspace | | |
| ET | Extract | | |
| FI | All file Types | FI-A | Adabas file |
| | | FI-B ∷ | Adabas SQL view |
| IE | Interface | | |
| KY | Keyword | | |
| LS | Library Structure | | |
| MD | Method | | |
| NO | Node | | |
| NW | Network | | |
| OW | Owner | | |
| PG | All packagelist types | PG-T | Total collection |
| | | PG-Q ∷ | DBRM |
| PR | All program types | PR-A | Parameter data area |

| Object Code | Object Type | Subtype Code | Subtype |
|-------------|-------------------------|--------------|-------------------------|
| | | PR-C ∷ | Copy code |
| RL | File relation | | |
| PY | Property | | |
| RT | Report listing | | |
| SC | Storagespace | | |
| SV | Server | | |
| SY | All system types | SY-A | Application |
| | | SY-C ∷ | Conceptual system |
| TR | Trigger | | |
| US | User | | |
| VE | All verification status | VE-A | Automatic verification |
| | | VE-C ∷ | Conceptual verification |
| VM | Virtual machine | | |

Access to Predefined Object Types

Access to predefined object types in Predict is handled by NSC external object type PRD-Docu-Objects. The following rules apply:

- The default value for *main* object types such as FI or PR is defined with the special function Maintain NSC Definitions > Add NSC Default Definitions. Permitted values are Y and N. See Maintain NSC Definitions in the section *Special Functions* in the *Predict Administration documentation*.
- The default value for *subtypes*, such as files of type Adabas, is asterisk. This means that the subtype 'inherits' the security definition of the main object type.
- Possible access modes for the instances of this NSC external object type are READ, ADD, MODIFY and DELETE.
- If a user has no READ access to a main object type such as FI, he cannot execute any functions that process this object type. Nor can he call the Maintenance and Retrieval menus for this object type.
- If an access mode is *allowed* for a main object type and *disallowed* for a subtype, the definition at subtype level has priority.

Example: A user has MODIFY access to Predict objects of type file (PRD-Docu-Object FI in Natural Security), but has no MODIFY access to files of type Adabas (Resource FI-A). He cannot modify files of type Adabas.

User-defined Object Types

Adding User-defined Object Types

The name of the NSC object type corresponds to the two-character object code for the UDE in Predict.

Access to User-defined Object Types

The same rules apply as for predefined object types. See [Access to Predefined Object Types](#).

Special object types

Adding Special Object Types

Use the special function Maintain NSC Definitions > Add NSC Default Definitions to add all special object types as instances of NSC external object type PRD-Docu-Objects. See relevant parts of section *Special Functions* in the *Predict Administration documentation*.

- **-A**
Association Type
- **-O**
Object Type (for UDEs, only applies to Metadata Administration)
- **-R**
Retrieval Model
- **-I**
Implementation Plan

Access to special object types

The following rules apply:

- With object type -I you can create a security definition for a unique Plan ID or - with asterisk notation - for a range of Plan IDs, as you can with predefined object types.
- With other object types (-A , -O , -R), security definitions at object level have no effect.

Security Definitions at Object Level

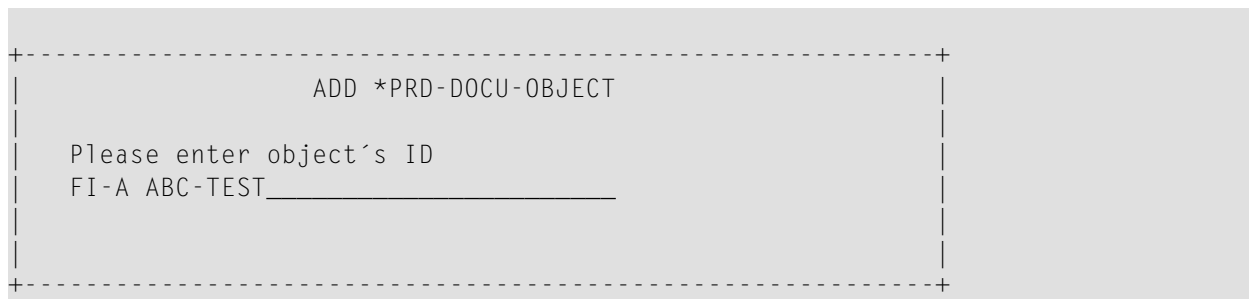
Security definitions can be added in Natural Security for documentation objects that can be processed with Predict functions.

Adding Definitions at Object Level

All instances must be added manually in Natural Security.

To add security definitions to documentation objects:

- Logon to library SYSSEC
- Select Maintenance, then choose *PRD-Docu-Object
- In the command line enter ADD
- In the upcoming window enter <type> <name> separated by a blank. Where <type> is the object type (e.g. FI-A) and <name> is the object name (e.g. ABC-TEST). See also screen below.



```
+-----+
|                                     |
|               ADD *PRD-DOCU-OBJECT |
|                                     |
| Please enter object's ID           |
| FI-A ABC-TEST_____              |
|                                     |
+-----+
```

With the special function Mass Grant you can create security definitions in Natural Security on the basis of data in Predict. Objects for which you wish to create security definitions must be placed in an Extract. See Mass Grant in NSC in the section *Special Functions* in the *Predict Administration documentation*.

Access Modes

Possible access modes for instances of this NSC external object type are READ, ADD, MODIFY and DELETE.

With Predict Security you can determine that certain users or groups only have access to certain objects. There are three strategies you can follow when protecting objects:

- Protect *Individual objects* Example: The file SALARY can only be read/modified by certain users.
- Protect a *range of objects* Use naming conventions to group objects and take advantage of asterisk notation in Natural Security. Example: User USER1 has been denied READ access to file objects in general, but READ access for files that begin with his user ID. The more specific authorization has priority.

- Protect *all objects* of a particular type.

PRD-Ext-Objects

All external object types that can be processed with Predict functions can be maintained by Predict Security. Instances of this NSC external object type are for example CO (COBOL Copy Code) or D2 (DB2 database).

Security definitions for PRD-Ext-Objects are used to protect functions.

Adding External Object Types

Use the special function Maintain NSC Definitions > Add NSC Default Definitions to add all external object types as instances of the NSC external object type PRD-Ext-Objects. The following external objects can be added:

| Code | External Object |
|------|----------------------|
| AC | ADACMP/ADAWAN |
| AD | Adabas database |
| AF | Adabas file |
| AI | ADAINV cards |
| AT | Vista table |
| AS | ADASCR |
| AV | Adabas - VSAM |
| BA | BAL/ASSEMBLER |
| BF | Adabas D table/view |
| CC | Language C |
| CO | COBOL |
| CR | SQL CREATE Statement |
| DD | DDM for Natural |
| D2 | DB2 database |
| EQ | Adabas table/view |
| FO | FORTTRAN |
| JF | Ingres table/view |
| LA | Language ADA |
| ND | Natural DBD |
| NF | NSC file |
| NO | No synonyms |
| NS | NSC user |
| OF | Oracle table/view |

| Code | External Object |
|------|---------------------|
| PA | Pascal |
| PL | PL/I |
| RC | Server table |
| RU | Verification rule |
| SN | NSP file |
| SG | DB2 storagegroup |
| SQ | Static SQL |
| SU | NSP user |
| TS | DB2 tablespace |
| T2 | DB2 table/view |
| UD | UDF for DL/1 |
| UL | User language |
| W | Sequential file |
| XF | Informix table/view |
| YF | Sybase table/view |

Access to External Object Types

- The default value for NSC external object type PRD-Ext-Objects is Y (allowed).
- Possible access modes for instances of this NSC external object type are READ, ADD, MODIFY and DELETE.
- Each user or group can be granted or denied access to certain external object types.
- A security definition at External-Type level is generally used to protect functions. Example: A user without ADD or MODIFY access to object type CO cannot execute the function Generate COBOL Copy Code.

PRD-3GL-Library

A security check is carried out when you access XRef data in 3GL libraries from Predict (Preprocessor, List XRef for 3GL). This check accesses the security definition for the 8-character library name in Natural Security.

Adding 3GL Libraries

If you wish to protect 3GL libraries, you must define security objects of this type manually in Natural Security.

The following default libraries must be defined with a hyphen instead of an asterisk, for example -SYSCOB-.

| Language | Library | NSC Object to be added |
|---------------|----------|------------------------|
| Language ADA | *SYSADA* | -SYSADA- |
| BAL/Assembler | *SYSBAL* | -SYSBAL- |
| Language C | *SYSCCC* | -SYSCCC- |
| COBOL | *SYSCOB* | -SYSCOB- |
| FORTRAN | *SYSFOR* | -SYSFOR- |
| PL/I | *SYSPLI* | -SYSPLI- |
| Static SQL | *SYSSTA* | -SYSSTA- |

Accessing 3GL Libraries

Possible access modes are READ, ADD, MODIFY and DELETE.

PRD-Function

As a rule, security definitions in Predict are defined at object type or object level. The following areas of Predict do not process any objects in Predict and are therefore protected with objects of NSC external object type PRD-Function in Natural Security:

Resources of the NSC external object type PRD-Function are divided into the following groups:

- **Special Functions**
- **Coordinator**
- **Defaults**, including **Extended Description Skeletons**
- **Other Functions**

Special Functions

You can grant or deny access to all Special Functions. If all Special Functions are allowed, you can disallow individual Special Functions.

Adding Special Functions

The following objects are added automatically with the special function Maintain NSC Definitions
> Add NSC Default Definitions:

| Natural Security Object | Predict Special Function |
|------------------------------|---|
| SPECIAL* | all Special Functions |
| SPECIAL-ADABAS_DEVICE_TYPES | Adabas device types |
| SPECIAL-MAINTAIN_NSC_DEF | Maintain NSC Definitions |
| SPECIAL-MASS_GRANT | Mass Grant in NSC |
| SPECIAL-DELETE_SETS | Delete old sets |
| SPECIAL-MAINTAIN_HELP_TEXTS | Maintain Predict help texts |
| SPECIAL-RECOVER | Recover |
| SPECIAL-REPOSITION_IMPL_DATA | Reposition implementation data |
| SPECIAL-SECURITY_FOR_AOS | Security for AOS |
| SPECIAL-CONSISTENCY* | Consistency of Predict (see note below) |
| SPECIAL-CONSISTENCY-B | - check database |
| SPECIAL-CONSISTENCY-D | - check extended descriptions |
| SPECIAL-CONSISTENCY-E | - convert EDIT MASKS |
| SPECIAL-CONSISTENCY-F | - check files and fields |
| SPECIAL-CONSISTENCY-H | - compress help texts |
| SPECIAL-CONSISTENCY-K | - check keywords |
| SPECIAL-CONSISTENCY-P | - check programs |
| SPECIAL-CONSISTENCY-R | - convert rules |
| SPECIAL-CONSISTENCY-V | - check verifications |
| SPECIAL-DELETE_REPORTS | Mass delete of report listings |
| SPECIAL-MAINTAIN_ACT_REF* | Maintain XRef data |
| SPECIAL-MAINTAIN_ACT_REF-A | - delete preprocessor ABEND data |
| SPECIAL-MAINTAIN_ACT_REF-G | - delete 3-GL data |
| SPECIAL-MAINTAIN_ACT_REF-N | - delete Natural data |
| SPECIAL-MAINTAIN_STD_FIELDS | Maintain standard fields |
| SPECIAL-REFRESH | Refresh Coordinator FDIC |

Access to Special Functions

With the object SPECIAL* you can define access rights for all Special Functions in Predict.

- If SPECIAL* is disallowed, all Special Functions are disallowed.
- If SPECIAL* is allowed, you can disallow individual Special functions, for example SPECIAL-DELETE_SETS.

Access Mode Values

Access to special functions is defined using access mode EXECUTE. The following values are possible:

Possible values for the access modes listed above are:

- **Y**
Function can be executed.
- **N**
Function cannot be executed.
- *****
Inherit. The security definition of the higher-level object is taken if appropriate.

For example:

If the individual Consistency subfunctions are set to inherit, the access rights to these subordinate objects are taken from the definition for object SPECIAL-CONSISTENCY*.

Coordinator

Security Objects for the Coordinator

All the necessary Natural Security objects are added automatically with the special function Maintain NSC Definitions > Add NSC Default Definitions. See table below.

| Natural Security Object | Predict Coordinator Function |
|-------------------------|------------------------------|
| CO-IMPORT | Import, Test, Load |
| CO-EXPORT | Export, Unload |

Access Mode Values

Access to Coordinator functions is defined using access mode EXECUTE. Possible values are Y, N, *. See [Access Mode Values](#).

Defaults

With object DEFAULTS* you can define access rights for all Default functions. See list below.

If DEFAULTS* is disallowed, all Default functions are disallowed. If DEFAULTS* is allowed, you can disallow individual Default functions, for example DEFAULTS-NATIVE-SQL.

Adding Defaults

All required Natural Security objects are added automatically with the special function Maintain NSC Definitions > Add NSC Default Definitions. See table below.

| Natural Security Object | Predict Default |
|-------------------------|--|
| DEFAULTS* | all Defaults |
| DEFAULTS-GENERAL* | all General Defaults |
| DEFAULTS-GENERAL-M | General Defaults - Maintenance options |
| DEFAULTS-GENERAL-R | General Defaults - Redocumentation using source code |
| DEFAULTS-GENERAL-G | General Defaults - Redocumentation using xref data |
| DEFAULTS-GENERAL-P | General Defaults - Protection |
| DEFAULTS-GENERAL-S | General Defaults - Synonyms |
| DEFAULTS-GENERAL-D | General Defaults - Suppress display of products |
| DEFAULTS-GENERAL-C | General Defaults - Miscellaneous |
| DEFAULTS-SKELETONS* | Extended Description Skeletons |
| DEFAULTS-PROFILE | Default Profile |
| DEFAULTS-LX_PROFILE | List XREF Default Profile |
| DEFAULTS-GENERATION* | Generation Defaults |
| DEFAULTS-COORDINATOR | Coordinator Defaults |
| DEFAULTS-NATIVE_SQL | Adabas Native SQL Defaults |
| DEFAULTS-USER_EXITS | Activate User Exits |

Under certain circumstances it can be useful if you define your own security objects:

Extended Description Skeletons

All Extended Description Skeletons are protected with the object DEFAULTS-SKELETONS* as standard. If you want to grant different access rights at object type or subtype level, you need to define a corresponding object manually in Natural Security.

Examples: DEFAULTS-SKELETONS-FI-A for skeletons of files of type Adabas. DEFAULTS-SKELETONS-FI* for skeletons of all files.

Generation Defaults

All Predict generation defaults are protected with the object DEFAULTS-GENERATION* as standard. To protect generation defaults for a particular external object type, you need to add a corresponding object manually in Natural Security.

Example: DEFAULTS-GENERATION-AF for Adabas files.

Access Mode Values

Access to defaults is defined using access mode EXECUTE. Possible values are:

Y, N, *.

See *Access Mode Values*.

Other Functions

| Natural Security Object | Predict Function |
|-------------------------|--|
| LIST_XREF_3GL | LIST XREF for 3GL Note: If this Security object is protected, the user cannot access the LIST XREF for 3GL menu. |
| METADATA-DEFAULTS | Metadata Administration function Defaults |

5

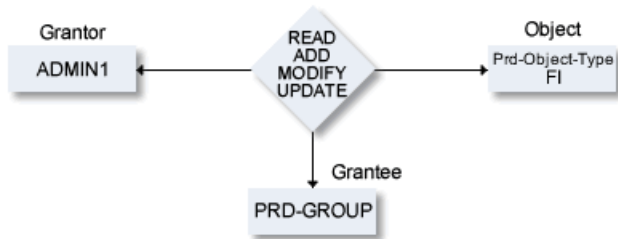
Security Profiles

| | |
|--------------------------------|----|
| ■ General Information | 38 |
| ■ Defining Access Rights | 38 |

General Information

The main task of a security administrator consists of granting users access to objects.

This access is defined in Natural Security by means of a complex relationship.



In the above example, the security administrator with the user ID ADMIN1 (Grantor) grants the group PRD-GROUP (Grantee) access to NSC external object type FI.

Access can be granted as follows:

- **Natural Security**

With Natural Security functions you can administer access rights for Predict objects, object types and functions. See also *Natural Security documentation*.

- **Special Function Mass Grant**

Use the special function Mass Grant to generate security definitions in Natural Security from data in Predict. Only applicable to objects.

Defining Access Rights

What can be Protected?

With security definitions in Natural Security you can protect the following:

- Documentation objects
- Special objects (for example Retrieval Models and Association Types)
- External object types
- Predict functions
- XRef data

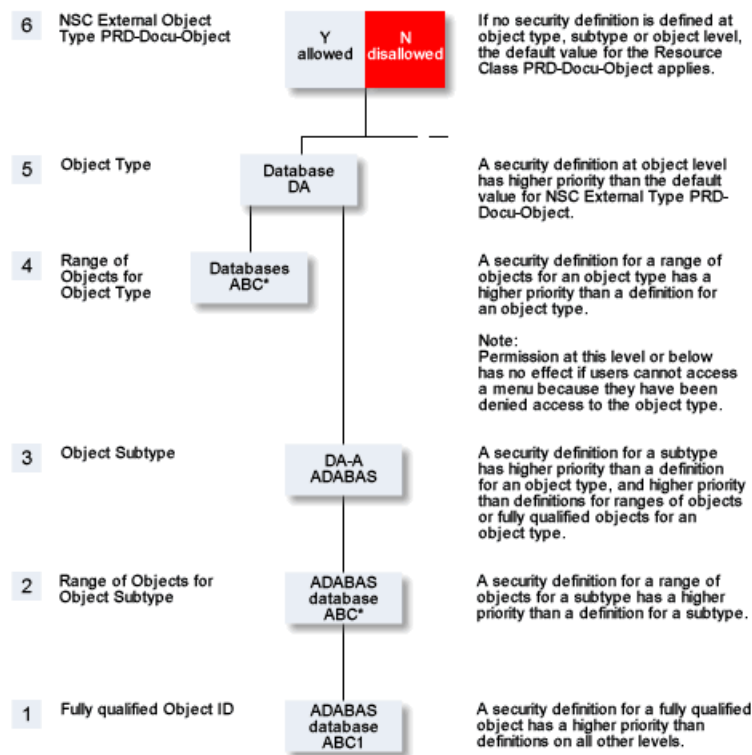
See complete list in [Predict Security using Natural Security](#).

Two Levels of Definitions in Natural Security

There are two levels of security definitions Natural Security:

- Each object in Natural Security has a *default* definition. This value is taken when no specific definition for a user or group exists.
- Each object in Natural Security can have a link to a user or group. This link contains a specific security definition for an individual user or a group.

Hierarchy of Security Definitions



Object and Object Type Level

If you disallow an object type, this object type does not appear in a Predict main menu. This means you cannot process any object of this type. Example:



Granting permission for subtypes or for individual objects has no effect, because the user is not able to call the respective menu, for example Maintain Database.

If you wish to prevent access to most databases, for example, but permit access to individual databases, create the following object-level definitions:

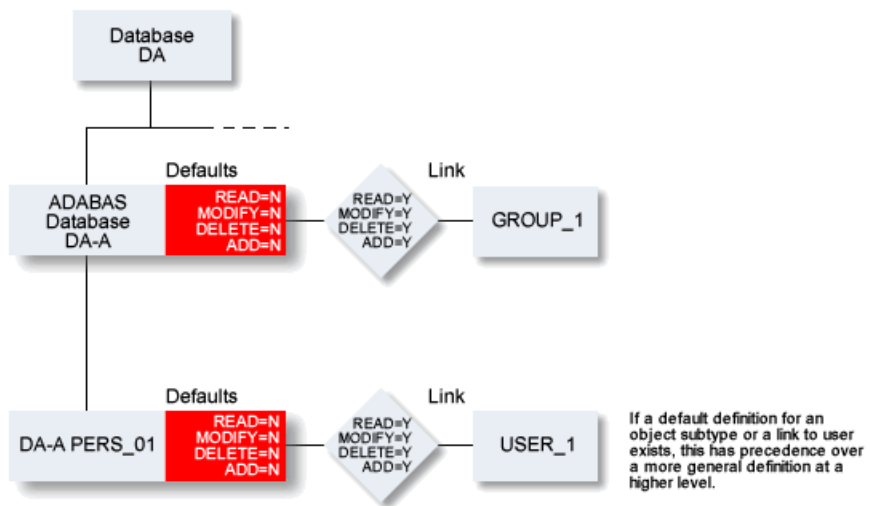
| | |
|----------------|---|
| Databases • | READ=N MODIFY=N DELETE=N ADD=N |
|----------------|---|

and

| | |
|-------------------|---|
| Databases ABC* | READ=Y MODIFY=Y DELETE=Y ADD=Y |
|-------------------|---|

In the example above, the user can access only databases which are prefixed with ABC.

Default Definitions and Links to Users

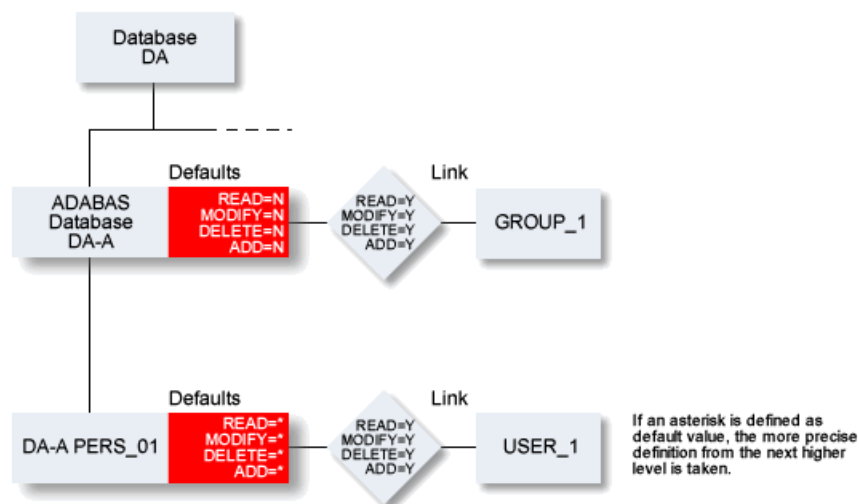


In this example, members of GROUP_1 have access to Adabas databases, but do not have access to database object PERS_01. The only user with access to PERS_01 is USER_1.

Recommendation

Because a default definition for an object applies to all users and groups, it can make sense to specify an asterisk as default value.

Default Definitions and Links to Users - with Inheritance



In this example, members of GROUP_1 have access to Adabas databases, and also access to database object PERS_01.

Sample Security Definitions

Example 1

In this example, all users may read files of type Adabas, but only members of group DBA-GROUP may modify them.

| Object Type | Object | Link to User | READ | MODIFY | DELETE | ADD |
|-------------|--------|--------------|------|--------|--------|-----|
| FI-A | - | default | Y | N | N | N |
| FI-A | - | DBA-GROUP | Y | Y | Y | Y |

Example 2

In this example, USER1 can only maintain files that start with his user ID. All other users can maintain files starting with their user ID, but not those with user ID USER1.

Object Type Definition

| Object Type | Object | Link to User | READ | MODIFY | DELETE | ADD |
|-------------|--------|--------------|------|--------|--------|-----|
| FI | - | default | Y | Y | Y | Y |

Object Definition

| Object Type | Object | Link to User | READ | MODIFY | DELETE | ADD |
|-------------|---------|--------------|------|--------|--------|-----|
| FI | *(=all) | default | N | N | N | N |
| FI | USER1* | default | * | * | * | * |
| FI | USER1* | USER1 | Y | Y | Y | Y |
| FI | USER2* | default | * | * | * | * |
| FI | USER2* | USER2 | Y | Y | Y | Y |

Example 3

In this example, User1 can maintain only the following files:

- Files of any type that start with his user ID
- Files of type A that start with ABC

He cannot modify

- Files of type V, even those which start with his user ID.

Object Type Definition

| Object Type | Object | Link to User | READ | MODIFY | DELETE | ADD |
|-------------|--------|--------------|------|--------|--------|-----|
| FI | - | default | Y | N | N | N |
| FI | - | USER1 | Y | Y | Y | Y |
| FI-V | - | default | * | * | * | * |
| FI-V | - | USER1 | Y | N | N | N |

Object Definitions

| Object type | Object | Link to User | READ | MODIFY | DELETE | ADD |
|-------------|----------|--------------|------|--------|--------|-----|
| FI | * (=all) | default | * | * | * | * |
| FI | * | USER1 | N | N | N | N |
| FI-A | ABC* | default | * | * | * | * |
| FI-A | ABC* | USER1 | Y | Y | Y | Y |
| FI | USER1* | default | * | * | * | * |
| FI | USER1* | USER1 | Y | Y | Y | Y |

Areas of Predict that are not protected

Profile

A user can read and use the profile of another user. It is not possible to modify the profile of another user, and it is therefore not necessary to protect profiles separately with Predict Security.

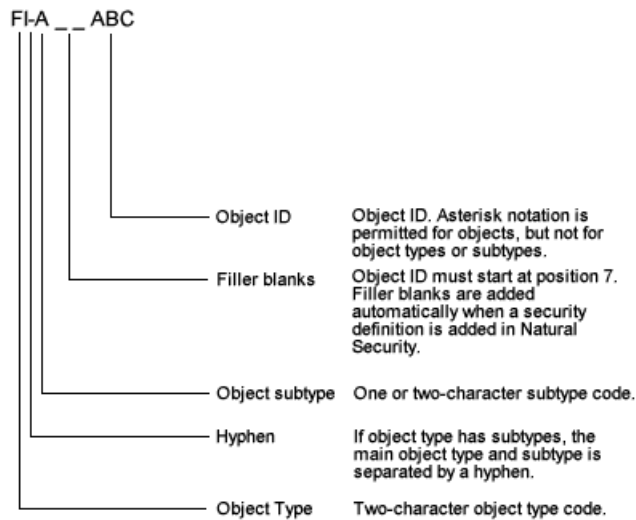
What is new, Help system

No Security checks are performed for these items in the Function Main Menu.

Checks Performed when Adding Security Definitions

The following checks are performed when security definitions are added to ensure that the user is allowed to read the data he wishes to maintain.

- User must have READ access if ADD, MODIFY or DELETE access is granted.
- READ access cannot be denied if the user already has ADD, MODIFY or DELETE access.
- Special syntax must be used when adding a definition for a Predict object. See diagram below.



Example: if you enter object FI(blank)ABC*, the name is changed to the correct syntax automatically: FI(blank)(blank)ABC*.

Tips and Tricks

Deny Globally, Grant Locally

In Example 1 below, individual subtypes or functions are allowed and at the same time all others disallowed.

Example 1

In this example the user is granted EXECUTE access to Special Function Recover but is not allowed to access any other Special Functions.

| Function | EXECUTE | Note |
|-----------------|---------|---|
| SPECIAL-* | N | All Special Functions are disallowed. |
| SPECIAL-RECOVER | Y | Special Function Recover is allowed. This specific definition has priority over the general definition. |

Granting Access to a Range of Files

In Example 2, the user may only read files with a certain prefix.

Example 2

| Object Type | READ | Note |
|-------------|------|--|
| FI | Y | This definition is not required if the default value = Y.. |

| Object | READ | Note |
|---------|------|---|
| FI * | N | READ access to all files is disallowed. |
| FI ABC* | Y | READ access is allowed for files starting with ABC. This definition has a higher priority than the general definition at the same level and does not contradict the definition at the higher level. |

Restrict Function to a specific Group

Example 3

In Example 3, only members of group SEC-DBA are allowed to execute the Special Function Protection.

| Function | EXECUTE | | Note |
|--------------------|---------|-------------------|--|
| | Default | for Group SEC-DBA | |
| SPECIAL-PROTECTION | N | Y | Security definitions for individual groups always have a higher priority than default definitions. |

Definitions at Subtype Level have Priority over Definitions for the Main Object Type

Example 4

| Object Type | READ | Note |
|-------------|------|---|
| FI | Y | |
| FI-A | N | User may not read any files of type Adabas. |

| Object | READ | Note |
|----------|------|---|
| FI TEST* | Y | User may read files that start with TEST, but not those of type Adabas. |

II Handling Protected Objects in Predict

This section covers the following topics:

[General Information](#)

[Maintenance](#)

[Retrieval](#)

[Active Retrieval](#)

[LIST XREF](#)

[LIST XREF for 3GL](#)

[Generation](#)

[File Implementation](#)

[Preprocessor](#)

[Incorporation](#)

[Comparison](#)

[Administration](#)

[Defaults, Special Functions](#)

[Coordinator](#)

[Metadata Administration](#)

[Conversion](#)

6

General Information

| | |
|---|----|
| ■ General Security Concept in Predict | 52 |
| ■ Sample Function | 53 |
| ■ Response Times | 54 |
| ■ Security Check in the Main Menu | 54 |

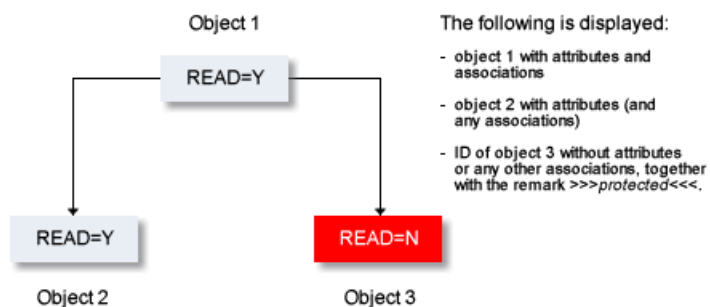
General Security Concept in Predict

Various security systems are based on the concept of see protection. This means that generally speaking a user is only able to see the objects and their IDs to which he has been granted access. This concept is not compatible with an open system such as Predict: the whole point of Predict retrieval functions is that objects are displayed together with their related objects. If Predict were to suppress protected objects completely, the user might draw the wrong conclusions - for example empty link lists would be displayed although links to protected objects are present.

Predict uses the following strategy:

- Attributes or associations of an object are only displayed if the user has at least READ access to that object.
- If a user does not have READ access to an object, the most he will be able to see of that object is its ID.
- If an object can be displayed, the IDs of all other objects to which the main object is linked can also be displayed. Under no circumstances, however, can attributes or associations be displayed for an object to which the current user does not have READ access.

Example

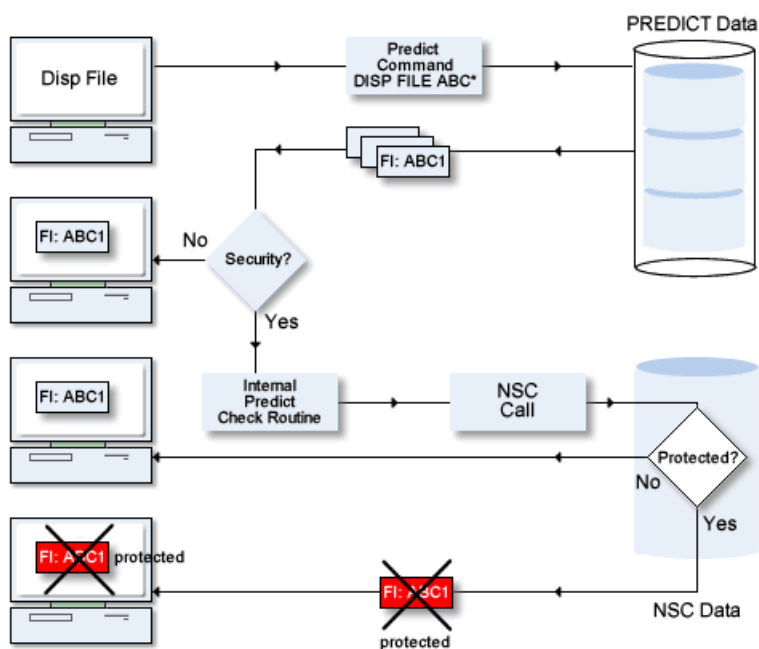


Sample Function

The diagram below illustrates what happens when function Display file is executed for all files starting with ABC*.

The command DISP FILE ABC* retrieves all files starting with ABC.

- If Predict Security is not active, no security check is performed and all the objects are displayed.
- If Predict Security is active, a security check is performed for each object in the retrieval report:
 - If the user has access to the object, the object is displayed.
 - If the user does not have access, the object is suppressed.



Response Times

Definitions in Natural Security have a large influence on response times. We therefore recommend the following:

- The security administrator should inform each user of his access rights. If the users know the scope of their access rights, they can formulate their queries more effectively and will spend less time 'groping around in the dark'. Access rights can also be displayed using the special function Display NSC definitions. See Maintain NSC Definitions in the section *Special Functions* in the *Predict Administration documentation*.
- Disallow READ access sparingly. If a user is going to link objects, it makes sense that he can read them. This is particularly important with keywords. Not having READ access also slows down response times.

Security Check in the Main Menu

The security check is called

- in the command interpreter: after entering the command
- in the main menu: when entering the object type or when selecting a type from a selection window.

This routing performs the following checks:

- **Generate:**
ADD or MODIFY access to external object types
- **Incorporate**
READ access to external object types
- **Compare:**
READ access to external object types
- **Retrieval, Active Retrieval:**
READ access to Predict object type
- **Maintenance:**
ADD, MODIFY or DELETE access to Predict object type
- **File Implementation:**
ADD, MODIFY , READ or DELETE access to Implementation Plans (object type -I)

- **Administration:**
READ access to Predict object type
- **Defaults, Special Functions:**
EXECUTE access to the corresponding function

No checks are performed for the following:

- What is new, Help system

7 Maintenance

| | |
|---|----|
| ■ Standard Maintenance Functions | 58 |
| ■ Type-Specific Maintenance functions | 60 |

Standard Maintenance Functions

Most security checks for maintenance functions are performed in the main menu. If the user does not have sufficient access rights, an error message is given.

Add

The user needs ADD access for the specified object type. For objects with subtypes, this subtype must also be entered so that a complete security check can be performed.

Copy

The user needs

- READ access for the object to be copied
- ADD access for the new object

For objects with subtypes, this subtype must also be entered so that a complete security check can be performed.

Modify

The user needs MODIFY access for the specified object.

Subtype

If the subtype can be modified, the system checks whether the user has sufficient access to the object with the new subtype. See function Rename.

Edit Owner

The user can only enter Y in the field Owner if he has at least READ access to the object type Owner.

Edit Child

The user can only enter Y in fields such as Program, for example, if he has at least READ access to the corresponding object type.

Parent object

If the parameter in parent can be specified, MODIFY access to the parent object is checked. If no permission has been granted, the field is read-only. A parent object cannot be modified.

Rename

The user needs MODIFY access to the specified object. An additional check is performed as to whether the renaming would lead to *increased* or *reduced* access.

- **Increased access**
through Rename function If a Rename would lead to a user having increased access, the function cannot be executed and a message appears. The user must enter a new ID which would give him either the same or reduced access rights.

Example:

User USR-1 has no DELETE access to programs that start with 'ABC', but he does have DELETE access to programs that start with 'X'. Renaming the program ABC-PR to XYZ-PR would increase access rights of this user and is therefore not allowed. The following message appears:

```
You are not authorized to execute this rename.
↵
```

■ **Reduced access through Rename function**

If a Rename would lead to a user having reduced access rights, a window appears in which the user must confirm the loss of access rights. See example below.

Example:

User USR-1 has MODIFY access to programs that start with 'ABC' but no MODIFY access to programs that start with 'X'. Renaming the program ABC-PR to XYZ-PR would restrict the user's access rights. The user must confirm this loss of access explicitly.

```
13:27:54          ***** P R E D I C T *****          2007-05-31
                                - Rename Program -
Program ID ..... ABC-PR          Modified 2007-05-31 at ↵
08:55
Type ..... Documented          by USR-1

Enter new Program ID ... XYZ-PR

Enter '.' to return to menu. ! If you execute this function you !
                             ! will no longer be authorized to !
                             ! modify this object.             !
                             !                                   !
                             ! Execute      (Yes/No)             !
                             +-----+
                             +-----+
```

Purge

The user needs DELETE access to the specified object.

Scratch

If objects of type database, system and program are deleted, subordinate objects are deleted, too. A security check is carried out for each object that is to be deleted additionally. If an object cannot be deleted, a message is given and this subordinate object is not deleted.

Master files

If a master file is deleted, all dependent userviews are deleted, too. Otherwise this would result in inconsistent data. If a userview cannot be deleted, the master file cannot be deleted either.

Edit Description

The user needs MODIFY access to the specified object.

Display

The user needs READ access for the specified object.

Link Children

User must have MODIFY access for the object whose link list is to be processed. No check is performed for the entries in the link list. If an object is READ-protected, the attributes are suppressed. The IDs of the linked objects are always displayed.

User must have at least READ access for the child object type. If this access has been granted, the entire link list can be processed.

With the Select function, the IDs of all objects are displayed. If an object is READ-protected, the attributes are not displayed and the object is marked with >>>protected<<<.

Edit Owner

The user needs READ access for the object type owner. For more information see function Link children, above.

Type-Specific Maintenance functions

This section is arranged alphabetically by object type.

Extract

The standard extract maintenance functions are subject to the normal maintenance checks. See [Standard Maintenance Functions](#). The security checks for type-specific functions are described below.

Build / Extend and Extract

With this function, objects returned with a retrieval function can be put in an extract.

The following access is required.

- MODIFY access to the extract
- READ access to the object type specified.

When the function is started, the normal retrieval functions are carried out. This means that it is also possible that under certain circumstances protected objects are placed in the extract.

The same objects are placed in the extract irrespective of whether the user executed the List function.

Edit / Link Objects

This function is similar to the function Link children. However, since objects of different types can be linked to an extract, the security check of the function Link children is not applied. The following checks are performed instead:

- No READ access is required to object type of objects to be linked. If object type is protected, only the ID of the object is displayed; attributes are suppressed.
- The function Select is only available for object types for which the user has at least READ access.
- MODIFY access is required for the extract. The checks that are performed depend on the editor the user is working with:

Natural Editor

- When the object type is entered, the system checks that the user has READ access for this object type.
- With the List function, all objects of a type are displayed (as with Select function of the normal editors).

Software AG Editor

- No check is performed for linked object types.
- The user can enter any object in the editor. If he does not have READ access for an object, the ID of the object is displayed but attributes are suppressed.
- With the Select function, all objects of a type for which the user has permission are displayed for selection.

Export Extracts

The following access is required:

- READ access to the extract
- EXECUTE access to the Coordinator function Export (NSC object EXPORT).

Operate on Extracts

The following access is required:

- MODIFY access to the extract

The objects are added to or removed from the extract without additional checks.

File

Force Standard

With this function, only MODIFY access to the standard file is checked. No security checks on files affected by the rippling operation are carried out.



Note: It would not make sense to check access of all objects. The user would need MODIFY access to all files affected by the rippling operation, even though, for example, he can access some of these files only via rippling and does not have access to them using normal maintenance functions. MODIFY access to standard files should be granted sparingly!

Push Backward

The user needs MODIFY access to both files: to the first because it is modified, and to the second because new fields are inserted in the standard file with this function.

Other Edit Functions (Modify Adabas Attributes, Vista elements...)

The user needs MODIFY access to the specified file.

Owner

Edit Owner

If a new owner ID is specified, the user must have ADD access for this new object. No security check is performed when an owner is deleted from the owner list of an object.

Program

Redocument Program

The user needs the following access to the program:

- ADD access if the program is added
- MODIFY access if an existing program is overwritten.

If a system is specified in which the programs are to be entered, the user must also have MODIFY access to this system.

Verification

With verifications it is the status and not the subtype that can be protected separately. When the command SAVE is entered in the Rule Editor, a security check is performed on the verification with the new status.

8 Retrieval

| | |
|--------------------------------------|----|
| ■ List-Oriented Functions | 64 |
| ■ Display-Oriented Functions | 67 |
| ■ Attribute-Oriented Functions | 69 |

If the user has no READ access at object type level, the function is not executed. A window containing the valid object types for the current user appears or an error message is given.

The system behavior when protected objects are encountered depends on the type of retrieval function. These can be divided up into three groups:

- **List oriented functions** These functions provide information on the existence of objects. They do not provide information on the objects themselves. If no attributes or restrictions are specified, objects will also be retrieved for which the user has no READ access. This makes the output as complete as possible without the user learning anything about the attributes of an object.
- **Display oriented functions** These functions provide information on attributes or associations of objects. If a user does not have READ access to an object, the object will be suppressed completely.
- **Attribute oriented functions** These functions always provide information on attributes or associations of objects. If a user does not have READ access to an object, the object will be suppressed completely to prevent the user from gaining information about a protected object on the basis of attributes or associations.

List-Oriented Functions

These functions provide information on the existence of objects. They do not provide information on the objects themselves. If no attributes or restrictions are specified, objects will also be retrieved for which the user has no READ access. This makes the output as complete as possible without the user learning anything about the attributes of an object. Objects for which the user does not have READ access are marked with >>>protected<<<.

The following retrieval functions belong to this group:

- List/Select objects
- List/Select objects with no parent
- List/Select objects with no child
- List/Select dummy/placeholder objects. See also [Additional Information](#) .
- List/Select extract related to no object
- List/Select keyword related to no object
- List/Select owner with no user
- List/Select users related to no object
- List/Select unused storagespace

Result of List-Oriented Functions

The result of these functions depends on whether attributes or restrictions were entered as selection criteria:

Without attributes and restrictions

All objects are displayed, even those to which the user does not have READ access. Only the IDs are displayed together with the remark >>>protected<<<. See example below.

This has the following advantages:

- the lists are complete
- the user learns nothing about a protected object apart from the fact that it exists.

| 13:18:48 | ***** P R E D I C T ***** | 2007-05-31 |
|----------|---------------------------|-------------------------|
| Plan 0 | - Select File - | |
| Cmd | File ID | Type Fnr DDM Impl Other |
| — | CHD-CUSTOMER-CUST | I |
| — | CHD-D_FORMATE | D |
| — | CHD-DEPENDING-ON | S 53 |
| — | CHD-DESCRIPTOR | >>> protected <<< |
| — | * CHD-ESQ_FILE | B * |
| — | CHD-EXPANDED | >>> protected <<< |
| — | CHD-FB-TEST-U1 | U 75 |
| — | CHD-FB-TEST-U2 | U 75 |
| — | CHD-FB-TEST-U3 | U 75 |

In this example you can also see that field Cmd is not available for protected objects.

With attributes and restrictions

Only objects for which the user has at least READ access are displayed.

The user does not find out that objects were not shown for security reasons. The remark >>>protected<<< does not appear.

This has the following advantage:

- the user could otherwise discover that objects have a certain value for a particular attribute, which would undermine the READ protection defined by the Security administrator.

If the user does not have READ access to a range of objects - for example all files of type A that start with USER1 - the message "No objects found" is given when he specifies File ID=USER1* and File type=A.

Additional Security Checks

Each object is checked for READ access. The following additional checks are performed for the individual functions:

Select objects

An additional check is performed when an object is placed in the workplan with a command:

- the user can only select objects for which he has at least READ access
- the field Cmd is locked for objects for which the user has no READ access
- if a command is entered, the system checks that the user is allowed to execute the corresponding function.
- if an asterisk is entered, only valid commands are displayed for selection.

List/Select Dummy/Placeholder objects

- Only parent objects are checked against security.
- For the functions Link and Unlink: the user needs MODIFY access to the parent object.
- For the function Add: the user needs ADD access to the child object
- For the function Select: the user needs READ access to the child object.
- For the function Purge: no security check is performed.

List/Select owner with no user

- Only the READ access for the owner is checked. If no READ access has been defined for an owner, the output of the object in which the owner is defined is replaced by the remark >>>protected<<<.

Select owners with no user

- READ access for the owner is checked. If no READ access has been defined for an owner, the owner is marked >>>protected<<<.

Display-Oriented Functions

With display-oriented functions, the user wants to find out about attributes or associations of objects. If a user does not have READ access to an object, the object is suppressed completely.



Note: Response times can be slow if a large number of protected objects are processed by this type of function. After a few seconds, the user is asked whether he wants to cancel the function or continue.

This group includes the following functions:

Single-level output

- Display objects
- Display objects with no parents
- Display objects with no child
- Display extracts related to no object
- Display keywords related to no objects
- Display users related to no object
- Display unused storagespaces
- Difference of files



Note: With the functions above, extracts, keywords and users are regarded as objects in their own right. With other functions, they are regarded as attributes of another object and are evaluated accordingly. See [Attribute-oriented Functions](#)

Multi-level output

- Display/List objects with children
- Display/List objects with parents
- Display dummy/placeholder objects
- Execute retrieval model
- List files related to a file

Result of Display-oriented Functions

As with list-oriented functions, the result of display-oriented functions depends on whether attributes or restrictions were specified as selection criteria.

Without attributes and restrictions

Single-level output

- Only objects to which the user has at least READ access are displayed. If the user does not have READ access, the object is suppressed. This means:
 - The remark >>>protected<<< does not appear. However, a message is given indicating how many objects were not displayed due to security: >>>nn Object(s) suppressed because of security protection<<<.
 - If a retrieval operation returns only objects to which the user has no access, the following message is given:

You are not authorized to read this object

Multi-level output

- On the *first* retrieval level, the same checks are performed as for single-level output. See above.
- If protected objects are found on *other* retrieval levels (for example the child objects with function programs with children) the ID of the object together with the remark >>>protected<<< is output, but no information on attributes or associations is given.

The example below shows that on the first retrieval level 46 files (together with an unknown number of dependent objects) have been suppressed due to security. In addition, the user does not have sufficient access for some related objects and these are marked >>>protected<<<.

```
13:25:54          ***** P R E D I C T *****          2007-05-31
                        - List File with Children -                      Page:   2

File ID ..... CHD-FB-TEST-U1
Type ..... Adabas userview

Cnt  child File ID                                Type  Fnr   DDM Impl Other

    1 ARH-BT1                                >>> protected <<<
    2 ARH-OT1                                >>> protected <<<

>>> 46 File(s) suppressed because of security protection <<<
***** End of Report *****
```

With attributes and restrictions

Objects to which the user has no READ access are suppressed *completely*. This means:

- The remark >>>protected<<< does not appear
- No message is given to indicate how many objects were suppressed due to security.
- The user is not informed that objects were not displayed due to security.

If the user has no READ access to a range of objects - for example all files of type A that start with USER1 - and specifies File ID=USER1* and File type=A, he receives the message

No files found

Additional Information

Difference of files

- User must have at least READ access for both files.
- A message is given if the function cannot be executed for security reasons.

Attribute-Oriented Functions

This group contains the following functions:

- List Vista numbers
- Explode IMS databases
- List/Select verifications to regenerate
- List fields and related views
- List fields related to a Z file
- List/Select objects with no owner
- Users related to objects
- Cross reference extract
- Cross reference owner
- Cross reference keyword
- Extract related to objects
- all field retrieval functions



Note: Objects of type field are handled as attributes of file objects. If the user does not have READ access to the file, he is not able to determine which fields are contained in the file.

Result of Attribute-Oriented Functions

These functions evaluate

- attributes of an object (for example Vista numbers), or
- links to objects that are regarded as attributes in Predict Security. These are:
 - Extract
 - Field
 - Keyword
 - Owner

To prevent the user gaining information on attributes or links, an object to which he has no READ access is suppressed totally. This means:

- IDs of protected objects do not appear in the retrieval report.
- The comment >>>protected<<< does not appear.
- No message is given at the end of the report as to how many objects were suppressed due to security
- If the user has no READ access to a range of objects - for example all files starting with USER1 - the following message is given:

```
No objects found.
```

Additional Security Checks

In addition to the check for READ access to the object, the following checks are performed for the individual functions:

List Vista numbers

- If the user has no READ access to the network, this object is suppressed completely and not evaluated further.
- If the user does have READ access to the network, each individual object within the network whose Vista number falls within the specified range is checked (database, file, Vista element). If no READ access has been granted for these objects they are suppressed completely.

Cross reference field

See function [Implode fields](#).

Explode IMS database

- The first check is for READ access to the database object.
- If the user has READ access, each subordinate object is checked for READ access.
- Objects for which no READ permission exists are marked >>>protected<<<, no checks on subordinate objects are performed.

List/Select objects with no owner

Objects for which no READ access has been defined are not included in the output. The owner is regarded as an attribute of the object.

Users related to objects

This function evaluates the owner as attribute of the object.

All Field Retrieval Functions (except Implode fields, Cross reference fields)

- Object type field is regarded by Predict Security as attribute of object type file. Security checks are only performed on the files containing the fields.
- Fields of a file for which the user has no READ are not included in the output.

Implode fields

- A field is not evaluated if the user has no READ access for the file.
- If the user does have READ access, each object is checked for READ access
- Protected objects are marked >>>protected<<< and not evaluated further.

List/Display fields related to a Z-file

- If the user has only READ access - but no MODIFY or DELETE access - to the standard file, a security check is performed for each subsequent field/file.
- If the user has MODIFY or DELETE access to the standard file, *all* subsequent files are output. This is because this function is designed to predict the effects of a change to the standard file. Rippling depends only on the MODIFY access to the standard files. See [Force Standard](#).

Cross Reference Keywords/Owners, Extracts related to Objects

If a user has MODIFY or DELETE access to an extract, keyword or owner, all related objects are displayed - irrespective of the READ access to these objects. This is done for the following reason: When extracts, keywords or owners are deleted, all objects linked to these objects must be updated for reasons of consistency. The three functions above are designed to provide a *complete* list of objects affected by the deletion of an extract, keyword or owner.

9 Active Retrieval

Predict Security only performs checks on documentation objects. External objects are not checked.

Active Retrieval functions can be divided into three groups:

- **List/Display-oriented functions**

This group includes functions with D in the Active Retrieval menu for the respective object type (Files, Fields etc.). The same checks are performed as for Retrieval. The result depends on the output mode and whether attributes or restrictions are entered. See [Retrieval](#).

- **Attribute-oriented functions**

Functions such as Entries referenced by members always evaluate Predict in their relationship with external objects.

| Active Retrieval for | Function | | | | |
|---|-------------------------------|-------------------------------------|----------------|-------------------|-----------------|
| | Entries referenced by members | not referenced/ inconsistently used | not documented | using dynamic SQL | not implemented |
| | R | U | O | V | N |
| File | Y | Y | | Y | |
| Field | Y | Y | Y | | |
| Note: Checks are only performed for the file containing the field. | | | | | |
| Program | Y | Y | | | Y |
| System | | | | | Y |
| Verification | Y | Y | | | |

If a user does not have READ access to an object, the object is suppressed completely. This is to prevent him gaining information about an object through the external objects to which the documentation object is connected. Total suppression means the following:

- IDs of protected objects do not appear in the retrieval report.
- The note >>>protected<<< does not appear.
- No message appears at the end of the report saying how many objects were suppressed for security reasons.
- If the user does not have READ access to a range of objects, the following message is given:

No objects found

Example: User has no access to files starting with USER1. If he enters USER1* for File ID, he gets the message:

No files found

■ Functions that search using external objects

| Active Retrieval for | Function | | | | | |
|----------------------|-----------------------|-------------------------------------|---------------------------|----------------|--------------|-------------------|
| | Members/ Libraries | Entries referenced by members | Entries not referenced | not documented | with members | using dynamic SQL |
| | I | R | U | O | T | V |
| File | | | | Y | | |
| Member | Y | Y | Y | Y | Y | Y |
| System | Y | | | Y | Y | |

The following rules apply:

- No security check is performed on the external objects.
- All external objects are listed, but if the user does not have access to the Predict object, the object ID is not displayed. The object is marked as >>>protected<<<. See example below.
- With functions that do not display any object IDs, for example Files not documented, no security check is performed.

13:45:28

***** P R E D I C T *****
- List Library -

2007-05-31

| Cnt | Library | Fnr | DBnr | Documented system |
|-----|---------|-----|------|--------------------------------|
| 1 | SYSTEM | 41 | 10 | >>> System is protected <<< |
| 2 | SYSSTK | 45 | | STK-1 |
| 3 | SYSDIC | 7 | 177 | >>> Library not documented <<< |

10

LIST XREF

| | |
|------------------|----|
| ■ Save Set | 78 |
|------------------|----|

The function LIST XREF is checked against the security definition in Natural Security for a library.

Before a program is displayed with the LIST XREF function X, the READ access defined in Natural Security is checked.

Save Set

LIST XREF sets for Natural libraries can only be saved if the user is authorized in Natural Security to perform a LOGON to the library.

11

LIST XREF for 3GL

The user must have at least READ access to the 3GL library.

If a Library Structure is specified, the user must have at least READ access to this object.

12 Generation

| | |
|---|----|
| ■ Checks for all Generation Functions | 82 |
| ■ Security Definitions in Natural Security at Object Type Level | 82 |
| ■ Type-Specific Security Checks | 82 |
| ■ PUNCH | 85 |

Checks for all Generation Functions

For all generation functions, the user needs at least the following access rights:

- READ access to the documentation object from which the external object is generated.
- ADD or MODIFY access to the external object type.

Additional security checks for the individual functions depend on the type of external object to be generated. These checks are described below.

Security Definitions in Natural Security at Object Type Level

A Security definitions for a Predict external object type is used to protect the function itself.

Examples:

- A user with neither ADD nor MODIFY access to object CO of the NSC external object type PRD-Docu-Object is not allowed to execute the function Generate COBOL Copy Code.
- A user with only MODIFY access to object CO is only allowed to execute the function Generate COBOL Copy Code if existing members are overwritten.

Type-Specific Security Checks

- [ADACMP / ADAWAN / ADAFDU Definitions, ADAINV Definitions, ADASCR Definitions](#)
- [ADACMP / ADAWAN / ADAFDU Definitions](#)
- [ADAINV Definitions, ADASCR Definitions](#)
- [Adabas Tables/Views](#)
- [Adabas File](#)
- [DDM](#)
- [Verification Rule](#)
- [UDF for IMS](#)

- [Transparency Table](#)

ADACMP / ADAWAN / ADAFDU Definitions, ADAINV Definitions, ADASCR Definitions

READ access to the database containing the file is checked in addition to the READ access to the file:

- If parameter Database ID is not specified, a selection window appears containing linked databases to which the user has at least READ access.
- If the parameters Phys. File number or Phys. Database number are used to identify the file uniquely, a selection window may appear under certain circumstances. This window contains files and databases to which the user has at least READ access.
- If the user does not have access to any linked databases, he cannot execute the function and a corresponding message is given.

ADACMP / ADAWAN / ADAFDU Definitions

If parameter Input file ID is specified for this function, this file is checked for READ access, too.

ADAINV Definitions, ADASCR Definitions

The database specified with parameter Database ID is checked for READ access.

Adabas Tables/Views

If the default parameter Specification DB-ID forces the user to enter a database ID, either the specified database is checked or a selection window containing databases to which the user has at least READ access is displayed.

Adabas File

- If Database ID is specified, this database is checked for READ access.
- If Database ID is not specified or is specified incorrectly, an additional selection screen is displayed containing only databases to which the user has at least READ access.
- If the parameters Phys. File number or Phys. Database number are used to identify the file uniquely, a selection window may appear under certain circumstances. This window contains files and databases to which the user has at least READ access.

DDM

If the default parameter Specification DB-ID forces the user to enter a database ID, either the specified database is checked or a selection window containing databases to which the user has at least READ access is displayed.

- Database ID
 - If Database ID is specified, the READ access to this database is checked.
 - If an asterisk or invalid database ID is entered, an additional selection screen appears containing only databases to which the user has at least READ access.
 - If no valid database ID is entered and the default parameter Specification DB-ID forces the user to enter one, a selection window appears containing databases to which the user has at least READ access.
- Overwrite option
 - If Overwrite option is set to Y, the user needs MODIFY access to the external object type DD.
 - If Overwrite option is set to N, the user needs only ADD access to the external object type DD.
- Verification rules
 - If Generate or Replace list verif. rules is set to Y, only the READ access for the verification object is checked; access to the file linked to the verification is not checked.
 - If Generate verif. rules is set to Y, the user needs ADD access to external object type RU.
 - If Replace verif. rules is set to Y or S, the user needs MODIFY access to external object type RU.
 - If Generate/Replace or list verif. rules is set to Y, the user needs READ access to Predict object type VE.
 - If a processing rule cannot be generated for security reasons, the function Generate DDM terminates abnormally and an error message is given. The DDM can then be regenerated without the processing rule.
- IMS databases
 - If the Parameter Generate UDFs is set to Y, the user needs ADD access to the external object type UDF.
 - If the Parameter Replace modified UDFs is set to Y, the user needs MODIFY access for object type UDF.
 - If the DDM can only be generated with UDFs and the UDFs cannot be generated for security reasons, the DDM cannot be generated.
 - Access to files of type J is not checked, because only the ID of the I file can be entered.

Verification Rule

- With the function Replace verification rule, only the READ access to the verification object is checked; the access rights to the file object are not checked.
- The User must have ADD or MODIFY access to external object type verification rule.

UDF for IMS

A database ID must be entered for this function, and the user must have READ access to this database.

Transparency Table

The file specified under Related Adabas file ID is checked for READ access.

PUNCH

READ access to the external object and the file object in Predict is checked.

13

File Implementation

| | |
|--|----|
| ■ Additional Checks with File Implementation Functions | 88 |
| ■ Execute implementation plan | 88 |
| ■ Add / Extend implementation plan | 88 |
| ■ Extend implementation plan | 89 |
| ■ Modify / Display implementation plan | 89 |
| ■ Display Implementation Plan | 89 |

The following access is required in order to perform file implementation functions:

- READ access to the special object type -I of the NSC external object type PRD-Docu-Object.
- If Implementation Plans are protected at object level, the appropriate access for a range of plans or fully qualified Plan ID (instances of the NSC external object type PRD-Docu-Object).

Additional Checks with File Implementation Functions

In addition the checks listed above, the following checks are performed depending on the file file Implementation function.

Execute implementation plan

- This function changes the status of a generation task, so the user needs at least MODIFY access to the Implementation Plan to be executed.
- Security checks are carried out for each generation task individually. The user needs the same access for the task as he would need for the corresponding generation function. See [Generation](#).
- If a task cannot be executed due to security, the task is given the status sec. abort. The execution of the Implementation Plan continues.
- If an entire task cannot be executed due to security, the message "Implementation plan executed with errors" is given after generation.
- If the Implementation Plan is executed again, all tasks with the status sec. abort are automatically set to RE (reexecute). All other conditions must be set to RE manually. It is assumed that another user has different access rights or that the original user has acquired the necessary permission in the meantime.

Add / Extend implementation plan

- Only files to which the user has READ access can be entered in a plan (otherwise the user might be able to find out about the file via Restrictions, attributes or language).
- A user can add generation tasks for external members or external object types for which he does not have READ or MODIFY access, because no security-critical attributes are displayed with this function.
- If the Parameter with userviews is set to Y, generation tasks are first placed in the plan for the combination of File ID and external object types (depending on access rights). If the files specified under File ID contain a master file with userview(s), the userviews to which the user has READ access are also placed in the plan together with the corresponding external objects.

Extend implementation plan

- If a range of files is specified with asterisk notation, the files to which the user does not have READ access are not placed in the plan. No message is given to indicate how many files were suppressed due to security.
- If a range of files is specified with asterisk notation and the user is not authorized to access any files in is range, the message "Implementation Plan not extended" is given.

Modify / Display implementation plan

- If the user does not have READ access to the file, the file ID is displayed with the remark >>>protected<<<. The ID must be displayed, otherwise the plan's contents would not be displayed completely. By displaying the ID it is sometimes possible to make a guess at the file type, but this information cannot be found out for certain.
- If the user does not have READ access to the file, generation functions are not displayed. This would make it possible to draw conclusions about certain attributes such as file type.
- The function codes that can be entered depend on the user's access rights. The following access checks are performed.

| Function Code | Check |
|---------------|---|
| DO, MO, OO | ADD or MODIFY access to the external object type or external object |
| DI, SM | Same check as with Administration function Display: READ access to external member. |
| IN, UN, RE | No additional checks. |

Display Implementation Plan

- If the user has READ access to the file but no READ access to the generated member, the file ID and - in the following line - member and library name are displayed, the generation messages are suppressed with the remark >>>protected<<<.

14

Preprocessor

The following checks are performed:

When Creating XRef Data

- MODIFY access to the 3GL library

When Generating Copy / Include Code

- ADD or MODIFY access to the external object type
- READ access to the file

COPY Statement

- READ access to the file
- READ access to the external object type

All other Preprocessor Calls

- READ access to the file
- READ access to the database object
- ADD or MODIFY access to the external object type

If a user does not have sufficient access to execute a function, the following message appears:

```
You are not authorized ...
```

The function then continues as if the object had not been found.

15

Incorporation

| | |
|--|----|
| ■ Checks for all Incorporation Functions | 94 |
| ■ Additional Checks for individual Incorporation Functions | 94 |

Checks for all Incorporation Functions

- whether external object can be read
- whether documentation object can be written.

It is possible to add several objects one after another in a single incorporation function. When the function is started, it is not possible to know how many objects will be added in total and what the IDs of these objects will be. For reasons of consistency, it is not possible to interrupt an incorporation function once it has started. For this reason, only the object ID entered by the user is checked against security. No checks are performed on IDs of objects not specified by the user.

Additional Checks for individual Incorporation Functions

Incorporate Adabas File

Database number must be entered for the function Incorporate Adabas File. This database is checked for READ access because attributes of this database are displayed on the following screen.

This function also updates the values for the Vista elements of the file. However, no additional security checks are performed here.

Incorporate DB2 Tables/Views/Tablespaces

If the function Display Masters is executed from within the functions Incorporate DB2 Tables/Views and Incorporate DB2 Tablespaces, the names of all subordinate DB2 objects are displayed without any additional checks.

Incorporate DDM

When a DDM is incorporated, file relations are also added and the file list of the database object is modified. No additional checks are performed for these modifications. Checks are only performed for objects which can be entered in the function screen.

Only Natural Security is checked in the selection list.

Incorporate NDB

The user needs ADD or MODIFY access to the database in Predict and READ access to the NDB.

Individual files are not checked for security when files of type I and J are added or files of type I are modified.

It does not make sense protecting individual files in an IMS database. We recommend using naming conventions to give all files in an IMS database the same protection.

16

Comparison

| | |
|--|----|
| ■ Security Checks for all Comparison Functions | 98 |
| ■ Function-specific Security Checks | 98 |

Security Checks for all Comparison Functions

The user needs the following access:

- READ access to the external object type

The access required to the documentation objects depends on the parameter Update:

- MODIFY access if Update is set to Y,
- READ access if Update is set to N

Security checks are performed for all documentation objects that can be entered in the Comparison screen. If a field in the screen is write-protected - for example Current VM - no check is performed for this object.

Comparison functions correspond to display-oriented retrieval functions. Attributes of both documentation and external objects are displayed, which is why READ access is required in both cases. The system behavior depends on whether attributes or restrictions are specified when the function is called.

Function-specific Security Checks

Compare DDM

The lists of programs that use a field that has differences between documentation and implementation call active retrieval functions. As only files to which the user has READ access are compared, no further checks are performed.

17 Administration

| | |
|--|-----|
| ■ Security Checks for all Administration Functions | 100 |
|--|-----|

Security Checks for all Administration Functions

The user needs the following access:

- READ access to the documentation object database, dataspace, file or storagespace is checked when the function is called.
- ADD/DELETE/MODIFY/READ access to the external object type is checked depending on the function. See table below.

| Function | Access Mode | |
|----------------------|----------------------|--|
| | Documentation Object | External Object |
| Disconnect | READ | READ |
| Display | READ | READ |
| Purge | READ | DELETE |
| Purge Vista elements | READ | DELETE |
| Refresh file | READ | DELETE |
| Rename | READ | MODIFY for old name, ADD for new name |
| Select | READ | READ |

If the user does not have the permission required for a function, an error message is given.

Select

This function implicitly evaluates attributes or associations of an object. To prevent the user finding out any information about READ-protected objects, the display is suppressed completely if the user does not have READ access to either documentation or external object. The user is not informed that objects were not displayed due to security.

Display

This function implicitly evaluates attributes or associations of an object. To prevent the user finding out any information about READ-protected objects, the display is suppressed if the user does not have READ access to either documentation or external object. A message is given before the 'End of Report' indicating that objects were suppressed due to security, but the user does not discover how many objects were not displayed.

Purge

If other external objects are deleted with this function (for example processing rules with a DDM or Table with a DB2 database), only the main object is checked against security. Security definitions for these dependent objects are checked in the external environment, and an additional check by Predict would be superfluous.

18

Defaults, Special Functions

When these functions are called, the EXECUTE permission for the corresponding instance of NSC external object type Prd-Function is checked. See list of [Special Functions](#) and [Defaults](#). If the user does not have the necessary permission, an error message is given.

19

Coordinator

| | |
|---|-----|
| ■ Security Checks when working with different FDICs | 106 |
| ■ Security Checks at Function Level | 106 |
| ■ No Security Protection for Coordinator FDIC | 107 |
| ■ Security Definitions at Object Level | 107 |

The following rules apply when transferring data:

- The data to be transferred may contain objects to which the current user does not have READ access. It is also possible that due to selection criteria the data to be transferred only contains objects to which the user has READ access.
- Error processing is only performed for objects that are contained in the data to be transferred and for which the user does not have READ access.

For function Export/Unload, the user needs READ access; for functions Import/Load and Test ADD or MODIFY access is necessary.

Security Checks when working with different FDICs

When data is transferred to or from another FDIC, the data are checked against the corresponding security definitions in Natural Security. The database and file number of the Natural Security file is specified with parameters General Defaults > Protection > DBnr/Fnr of NSC file.

The following rules apply:

- For function Export/Unload, security checks are performed against the NSC file of the *source* FDIC.
- For functions Import/Load and Test, security checks are performed against the NSC file of the *target* FDIC.
- Source and target FDICs do not necessarily have to have the same NSC file.

Security Checks at Function Level

To disallow the Coordinator completely, you must disallow the library SYSDICBE in Natural Security. If some users are permitted to execute the extract maintenance function Export Extract but not the Coordinator itself, disallow the program MAIN in library SYSDICBE, and not the entire library.

Authorizations at function level are required to define different access rights for functions Import/Load and Export/Unload. See [Coordinator](#).

No Security Protection for Coordinator FDIC

The Coordinator FDIC is not protected by Predict security.

The function Clear deletes the Coordinator FDIC and releases this file for another import/load operation. The user can only apply this function to an FDIC he created himself by starting an import/load operation. A locked Coordinator FDIC created by another user must be released with the Special Function. See Refresh Coordinator FDIC in the section “Special Functions” in the “Predict Administration documentation”. This special function can be protected with the security object SPECIAL-REFRESH.

Security Definitions at Object Level

Export/Unload

If the user does not have READ access to an object to be exported/unloaded, the object is logged in a Report Listing and taken out of the set of objects to be transferred. The export/unload operation is not terminated.

If IMS objects (databases or files) are to be exported/unloaded and the user does not have READ access to *every* IMS object, the entire IMS structure is removed from the set of objects to be exported/unloaded.

Import/Load

ADD or MODIFY access is required depending on whether a new object is added or an existing object overwritten.

When an object is renamed during import/load, the user needs MODIFY access to the old ID and ADD access to the new ID.

The security checks for the Import/Load function represent the second phase of the Coordinator (Conflict Management) have been successfully performed and all conflicts resulting from the Unique ID, have been resolved. When the first two phases of the cycle have been successfully completed, the third phase - Consistency Check - is performed. For more information see the Predict Coordinator documentation.

If the user does not have sufficient access to an object in the set of objects to be imported/unloaded, this object is logged. All other objects are checked, but the import/load operation is interrupted. The user can either acquire the necessary access in the MAIN-FDIC or remove the objects to which he does not have access from the set of objects to be imported/loaded in the Coordinator FDIC.

Importing/Loading Placeholders

Because placeholder objects cannot replace other objects, these objects can only be added. ADD access is *not* checked, however.

When a placeholder is replaced by a 'proper' object, the system checks for ADD access (and not MODIFY access).

20

Metadata Administration

| | |
|--------------|-----|
| ■ UDEs | 110 |
|--------------|-----|

To protect metadata administration totally, you must disallow the library SYSDICMA in Natural Security.

You can protect object types, association types and retrieval models with objects -O, -A and -R of NSC external object type PRD-Docu-Object in Natural Security. See [Special Object Types](#). If the user does not have the appropriate ADD, DELETE, MODIFY, or READ access, an error message is given.

In order to execute the metadata administration function Defaults, the user must have EXECUTE access to the object METADATA-DEFAULTS of NSC external object type Prd-Function.

No security checks are performed on the 2-character external names.

UDEs

When a new UDE is added in Predict, you can add a definition explicitly in Natural Security or you can use the special function Maintain NSC Definitions > Add NSC Default Definitions to add default definitions for all new UDEs.

21

Conversion

To protect the conversion utility completely, you must disallow the library SYSDICCO in Natural Security.

III

| | |
|---|-----|
| ■ 22 Interfaces To Other Software AG Products | 115 |
| ■ 23 Protecting Predict Programs With Natural Security | 119 |
| ■ 24 Protecting External Objects in Predict With Natural Security | 129 |
| ■ 25 Protecting Predict With Other Security Systems | 139 |

22

Interfaces To Other Software AG Products

| | |
|-------------------------------------|-----|
| ■ Adabas Native SQL | 116 |
| ■ API | 116 |
| ■ Natural | 116 |
| ■ Predict Application Control | 116 |
| ■ Super Natural | 117 |
| ■ SYSAOS | 117 |
| ■ SYSDB2 | 117 |
| ■ SYSHELP | 117 |

Adabas Native SQL

No security checks are performed when Predict is accessed from Adabas Native SQL.

API

The same security checks are performed as in Predict maintenance functions.

Natural

DDMs

DDMs are protected using Natural Security. No additional measures are necessary in Natural to protect DDMs.

LIST XREF

For function Report programs with XRef data, Predict Security checks whether the user has READ access to the program.

Natural Development Server (NDV)

Base and compound applications are protected.

Natural Editor

.F and .V Commands in Natural Editor

If the user has no READ access to the file object in Predict, the Predict information is not displayed.

Edit Free Rule in Natural Map Editor

If the user has no READ access to the verification object in Predict, the Predict information is not displayed.

Predict Application Control

The only security checks are performed by the Predict Coordinator.

Super Natural

A check is performed in Super Natural as to whether the user can use a particular DDM. If this is the case, the user can read the corresponding data in Predict.

SYSAOS

No security checks are performed.

SYSDB2

The same checks are performed for incorporation functions called from SYSDB2 as are performed in Predict.

A user must have READ access to the respective program object in order to read the entry points of a DBRM.

With function Bind plan, SYSDB2 checks whether the user has READ access to the corresponding System object. As in Predict generation functions, only the main object is tested.

SYSHELP

The SYSHELP utility does not need to be protected. In a production environment it is only possible to access programmed objects, and each user will have access to these objects. It would not make sense to restrict SYSHELP access in a development environment.

23

Protecting Predict Programs With Natural Security

| | |
|---|-----|
| ■ Functional Scope | 120 |
| ■ Naming Conventions for Library SYSDIC | 120 |
| ■ Protecting Programs in Library SYSDICBE | 124 |
| ■ Naming Conventions for Library SYSDICCO | 125 |
| ■ Naming Conventions for Library SYSDICMA | 126 |

Functional Scope

Protecting Entire Predict Libraries

To protect an entire Predict library, for example SYSDICCO containing all conversion programs, you must create the corresponding security definitions directly in Natural Security as in earlier versions of Predict.

Protecting Individual Predict Programs

Individual Predict functions, for example administration functions, can be protected by disallowing the respective Predict system programs in Natural Security. The following sections contain tables that can be used to determine the names of Predict programs that have to be disallowed in Natural Security to restrict access to individual functions in Predict.

However, we recommend you use the new functionality provided by Predict Security for this purpose. This is easier both for the administrator and for the user. See [PRD-Docu-Object](#) in the section **Natural Security Entities** in this documentation for more information.

Naming Conventions for Library SYSDIC

Combined Program Names in SYSDIC

Many program names are built by combining a function specific prefix and an object type specific suffix.

Example: the program name ACMFI that adds, copies or modifies file objects is built from the prefix ACM (add/copy/modify) and the suffix FI (file).

Prefixes of Program Names in SYSDIC

The prefixes used for program names in SYSDIC are as follows.

| Prefixes of Program Names in SYSDIC | | |
|-------------------------------------|------------------------|-------------------------------------|
| Name | Subsystem | Explanation and Remarks |
| ACM | Maintenance | Add / Copy / Modify objects |
| ACT | Active Retrieval Menus | |
| ADA | Active Retrieval | List master fields/files referenced |
| CAT | Maintenance | Catalog function of the editors |
| CNT | Active Retrieval | Count functions |

| Prefixes of Program Names in SYSDIC | | |
|-------------------------------------|--|---|
| COM | Comparison | Compare |
| COP | Maintenance | Copy/Move owner |
| DDA | Special functions | DDAOB calls the Special Functions menu |
| DDF | Defaults | DDFOB calls the Defaults menu. |
| DEF | Active Retrieval | List objects not documented |
| DIS | Administration Impl. | Disconnect External Objects |
| DIF | Retrieval | Difference of Files |
| DSP | Retrieval/Maintenance/Active Retrieval | Display functions |
| EDT | Maintenance | Editors |
| ERM | Retrieval | Execute retrieval models |
| EXP | Retrieval | Explode functions |
| FIL | File implementation | |
| GEN | Generation | Generation |
| IMP | Retrieval | Implode functions |
| INC | Incorporation | Incorporation |
| IPL | Active Retrieval | List implemented members/libraries |
| LNK | Maintenance | Link keywords to objects |
| LST | Retrieval | List functions |
| MNT | Maintenance | Maintenance menus |
| NEW | -- | Predict initialization |
| NOT | Active Retrieval | List objects not implemented/referenced |
| PAL | Predict application programming interf. | |
| PRE | -- | The preprocessor |
| PUL | Maintenance | Push backward |
| PUR | Maintenance | Purge modules |
| RDO | Maintenance | Re-document program |
| RED | Maintenance | Read objects/text into the editors |
| REF | Active Retrieval | List objects referenced by members |
| REN | Maintenance | Re-number, rename, change type |
| RET | Retrieval | The retrieval menus |
| RFF | Administration Impl. | Refresh file |
| SAV | Maintenance | Save description; save processing rule |
| SEL | Retrieval, Maintenance, Active Retrieval | Selection |
| SHW | Retrieval, Maintenance, Active Retrieval | Display and list functions |
| XRF | Retrieval | Cross reference functions |

| Suffixes of Program Names in SYSDIC | |
|-------------------------------------|---|
| Name | Subsystem |
| AN | Vista table |
| AVB | Adabas VSAM Bridge |
| BAL | Programming language ASSEMBLER |
| CCC | Language C |
| COB | Programming language COBOL |
| DA | database |
| DBA | Adabas database |
| DB2 | DB2 database |
| DDA | General (non-language-specific) system defaults |
| DDM | Data definition module |
| DS | Description |
| EL | Field |
| EP | Entry Point |
| ESQ | Adabas SQL Server |
| FDT | Adabas file |
| FI | File |
| FOR | Programming language FORTRAN |
| FUS | Reposition implementation data |
| KY | Keyword |
| IMS | IMS database (EXP) or user defined field (GEN) |
| INV | ADAINV |
| IP | Implemented Objects |
| LAN | Language-specific system defaults |
| ME | Member |
| NAT | Programming language Natural |
| NO | File number |
| NW | Network |
| OW | Owner |
| PA | Packagelist |
| PLI | Programming language PL/I |
| PR | Program |
| PRO | Profile |
| REC | Recovery |
| RL | File relation |
| RU | Processing rule |

| Suffixes of Program Names in SYSDIC | |
|-------------------------------------|-----------------------|
| SAP | SAP tables |
| SCR | Adabas security |
| SEC | Natural Security |
| SET | Delete old sets |
| SG | Storagegroup |
| SQL | Adabas SQL |
| ST | Storagespace |
| SY | System |
| TAB | DB2 table / view |
| TS | Tablespace |
| UD | User-Defined Entities |
| US | User |
| USU | Super Natural user |
| VE | Verification |
| VM | Virtualmachine |
| WAN | ADAWAN or ADACMP |
| XRF | Delete XRef data |

| Other Program Names in SYSDIC | |
|-------------------------------|--|
| Name | Subsystem |
| BAT* | The batch processor |
| CHERU* | Check Rule |
| CMD | The command processor |
| CON* | Subroutines for Pop-Ups |
| DELPRD | Purge (delete) Predict system data |
| EXIT | Called if a Predict session ends without the command FIN |
| H-* | Help texts (the string following H- specifies the function and object for which the help text was created) |
| HLPDSP | Display help texts |
| MAIN | The main menu program in library SYSDIC |
| MENU | Calls the main menu program |
| MOVE | Move a field |
| N-* | Common subprograms |
| NENEL | Renumber the fields of a file |
| PUNCH* | Punch out generated code |
| RESET* | Reset the global variables and set the user profile |

| Other Program Names in SYSDIC | |
|-------------------------------|---|
| S-* | Common subroutines |
| STOP | Can be called if an appropriate PF key setting is defined |
| U-* | User Exits |
| WRITE | Write (punch out) generated code |
| X* | List XRef for 3GL (without programs starting with XRF or XXX) |
| XXX* | Check consistency of Predict data |
| WP* | Workplan functions |

Examples

Protecting Predict Administration Functions

- To protect functions in the Defaults menu, disallow individual DDF* programs or DDFOB, the program that calls the Defaults main menu.
- To protect functions in the Special Functions menu, disallow individual DDA* programs or DDAOB, the program that calls the Special Functions menu.
- To protect Administration Implementation functions, disallow the respective MNTIP* programs (for example MNTIPFI for Administration Implemented File).

Protecting Functions of other types (Examples)

- To protect DDM generation, disallow the program GENDDM.
- To protect all generation functions, disallow GEN* programs.
- To protect the maintenance subsystem, disallow the MNT* programs (but not the MNTIP* programs calling Administration Implementation functions).
- To protect maintenance of fields and files, disallow the programs MNTEL (maintain field objects) and MNTFI (maintain file objects).
- To protect the incorporation functions, disallow all INC* programs.

Protecting Programs in Library SYSDICBE

To protect Coordinator programs, we recommend you define access to the Natural Security objects in the table below. These objects are added automatically with the special function Maintain NSC Definitions > Add NSC Default Definitions. See also [NSC External Object Type](#) .

| Natural Security Object | Predict Coordinator Function |
|-------------------------|------------------------------|
| CO-IMPORT | Import, Test, Load |
| CO-EXPORT | Export, Unload |

You may wish to create a security definition for the following programs in library SYSDICBE:

| Other Program Names in Library SYSDICBE | |
|---|--|
| Name | Explanation |
| BAT* | Batch Processor |
| CMD | The command processor |
| EXIT | Called if a Predict session ends without the command FIN |
| MENU | Calls the main menu program |

Recommendation

In earlier versions of Predict, Migrate functions were usually restricted to dictionary administrators. With this version, however, it is quite possible that ordinary Predict users need the Coordinator for their daily work, for example when transferring data between Predict and Natural Engineering Workbench. Please bear this in mind when maintaining your security definitions.

Naming Conventions for Library SYSDICCO

Combined Program Names in SYSDICCO

Many program names in SYSDICCO are built by combining a function-specific prefix and an object type specific suffix.

| Prefixes of Program Names in Library SYSDICCO | |
|---|---|
| Name | Explanation |
| CNV | Convert Predict data from V3.4 to V4.1 format |
| V41RP | Convert reports and modules |
| V23NSC | Add Natural Security XRef data |

| Suffixes of Program Names in Library SYSDICCO | |
|---|-------------------|
| Name | Explanation |
| DA | Database |
| DC | Dataspace |
| EL | Field |
| FI | File |
| FIADA | Adabas attributes |
| FINET | Vista elements |
| META | Meta data |
| OB | Control program |
| PR | Program |
| SC | Storagespace |
| SETS | List XRef sets |
| SY | System |
| TR | Trigger |

| Other Program Names in Library SYSDICCO | |
|---|--|
| Name | Explanation |
| BAT* | The batch processor |
| CMD | The command processor |
| EXIT | Called if a Predict session ends without the command FIN |
| MENU | Calls the main menu program |
| MAIN | The main menu program in library SYSDIC |
| NEW* | Add new control records |

Naming Conventions for Library SYSDICMA

Combined Program Names in SYSDICMA

Many program names in SYSDICMA are built by combining a function-specific prefix and an object type specific suffix.

| Prefixes of Program Names in Library SYSDICMA | |
|---|-------------------------------|
| Name | Explanation |
| ACM | Add, copy or modify functions |
| DSP | Display functions |
| MNT | Maintenance menus |
| PUR | Purge functions |
| REN | Rename functions |
| SEL | Select functions |
| XRF | Cross reference functions |

| Suffixes of Program Names in Library SYSDICMA | |
|---|------------------|
| Name | Explanation |
| RM | Retrieval model |
| UR | Association type |
| UT | Object type |

| Other Program Names in Library SYSDICMA | |
|---|--|
| Name | Explanation |
| BAT* | The batch processor |
| CMD | The command processor |
| EXIT | Called if a Predict session ends without the command FIN |
| MAIN | The main menu program in library SYSDIC |
| MENU | Calls the main menu program |

24

Protecting External Objects in Predict With Natural Security

| | |
|---|-----|
| ■ Protecting Adabas Databases and Files | 130 |
| ■ Protecting DDMs | 136 |
| ■ Protecting Processing Rules | 137 |
| ■ Protecting Natural Source Programs | 138 |

Protecting Adabas Databases and Files

Adabas Online Services (AOS) functions that process implemented Adabas databases and files are called by the following Predict functions:

- Incorporate Adabas file
- Compare Adabas file
- Generate Adabas file (with/without option Stop users using file)
- Administration Implemented file Purge Adabas file (with/without option Stop users using file)
- Refresh Adabas file (with/without option Stop users using file)

With some of the above functions not only file structures but also data itself can be deleted. To avoid accidental deletion of data and data definitions, we strongly recommend reserving the use of Predict functions executing AOS functions to a limited range of users.



Note: The protection of Predict functions which execute AOS functions is independent from the protection defined for AOS functions in the library SYSAOS. Knowledge of Natural Security is required to carry out the tasks described in the sections below.

Activating and Deactivating Predict/AOS Security

Protection of Adabas databases and files in Predict requires that Predict/AOS Security is activated.

- Predict/AOS Security is *activated* by executing the program NSCPRDAX in the library SYSSEC and then once calling the Modify Library function for SYSDIC.
- Predict/AOS Security is *deactivated* by executing the program NSCPRDDX in library SYSSEC and then once calling the Modify Library function for SYSDIC.

Protection of Adabas Databases and Files, Concepts

Applying AOS functions to databases or file ranges in Predict can be controlled with Predict/AOS security mechanisms. Predict/AOS use can be controlled

- for *individual* users with *user-specific* AOS security profiles,
- all users *without their own user-specific profile* with *default* AOS security profiles.

How to Restrict Use of AOS Functions in Predict

Two steps are required to restrict the use of AOS functions:

■ Step 1: Specify the Dictionary Security Administrator in Natural Security

A dictionary security administrator must be specified for each Adabas database to be maintained with Predict/AOS functions. Dictionary security administrators are defined in the Predict/AOS Security Profile screen of Natural Security. See [Defining the Dictionary Security Administrator in Natural Security - Activity 1](#), for a detailed description. Dictionary security administrators can give the right to process databases (or file ranges) with AOS functions either to individual users or to all users. Rights are given using AOS security profiles (see step 2).

■ Step 2: Define AOS security profiles in Predict

AOS security profiles determine which AOS functions can be applied by users to a database or a file range. AOS security profiles are defined with the Predict special function Security for Adabas Online Services. See Security for Adabas Online Services in the section *Special Functions* in the *Predict Administration documentation*. Each profile applies to a combination of a database or file range and a Natural Security user.

Defining Default Access Rights

You may wish to specify Predict/AOS rights for all users without a user-specific profile in one profile. This can be done by defining default AOS security profiles. A default profile for a database or file range applies to all users who do not have their own profile. To define a default AOS security profile, a default user must have been defined in the Predict/AOS Security Profile screen in Natural Security.

■ Defining a Default User

A default user is defined by assigning a Natural Security user or user group to the dummy database number 999 in the Predict/AOS Security Profile screen.

■ Defining a Default AOS security profile

By assigning a profile to the default user, the profile becomes a default profile. See [Defining AOS Security Profiles in Predict - Activity 2](#).



Note: The prompt “Please specify who is to be responsible for which database” in the Predict/AOS Security Profile screen is not correct when defining the default user.

Defining the Dictionary Security Administrator in Natural Security - Activity 1

A dictionary security administrator for each Adabas database must be specified in Natural Security. The user or user group defined as dictionary security administrator for a database is responsible for defining the access rights for Predict users by maintaining the AOS security profiles for that database.

Rules for Defining the Dictionary Security Administrator

- Only one dictionary security administrator can be defined for a database.
- If more than one administrator is desired, a group can be specified. Each group member can then perform AOS security tasks, using the group ID.
- The users or groups must be linked to the library SYSDIC. If a group is specified, each individual user in the group *must not* be linked to the library SYSDIC twice (as a member of the group and as an individual user).
- If people-protection for the library SYSDIC is changed from Y to N all links and profiles will be deleted.

Prerequisites

- Predict/AOS Security must have been activated by executing the program NSCPRDAX in the library SYSSEC.
- The library SYSDIC has to be defined people-protected. The Natural Security user defining the dictionary security administrator must have the right to modify the Natural Security definition of the library SYSDIC.

The Predict/AOS Security Profile Screen

Dictionary security administrators are specified in the Predict/AOS Security Profile screen shown below. To display this screen, proceed as follows:

1. Call the function Modify Library in Natural Security for the library SYSDIC.
2. Enter Y in the field Additional options of the Modify Library screen and
3. Select the topic User Exits in the selection window that is then displayed.

02-07-31 - Predict/AOS Security Profile - 13:29:18

Please specify who is to be responsible for which database:

| Data Base | DIC-Sec. Administ. | Data Base | DIC-Sec. Administ. | Data Base | DIC-Sec. Administ. | Data Base | DIC-Sec. Administ. |
|-----------|--------------------|-----------|--------------------|-----------|--------------------|-----------|--------------------|
| 180__ | DBSECGR__ | _____ | _____ | _____ | _____ | _____ | _____ |
| 999__ | DEFAULT__ | _____ | _____ | _____ | _____ | _____ | _____ |
| _____ | _____ | _____ | _____ | _____ | _____ | _____ | _____ |
| _____ | _____ | _____ | _____ | _____ | _____ | _____ | _____ |
| _____ | _____ | _____ | _____ | _____ | _____ | _____ | _____ |
| _____ | _____ | _____ | _____ | _____ | _____ | _____ | _____ |

| Columns | Meaning |
|---------------------|--|
| Database | Number of Adabas database to be protected. 999 can be specified as a dummy database number to define a default user. See also Defining Default Access Rights . |
| DIC.-Sec. Administ. | Natural Security user or user group to be dictionary security administrator for the database <i>or</i> default user to be used when defining a default AOS security profile in Predict. See Defining Default Access Rights . |

In the above example a Natural Security user DBSECGR is responsible for the database 180, and the Natural Security user DEFAULT is defined as the default user.

Defining AOS Security Profiles in Predict - Activity 2

Prerequisites

- AOS security profiles for a database can only be defined by the dictionary security administrator for that database.
- AOS Security Profiles can only be defined for users and user groups that are defined in Natural Security and linked to the library SYSDIC. Remember: If a group is specified, each individual user in the group *must not* be linked in Natural Security to the library SYSDIC twice (as a member of the group and as an individual user).

The Security for Adabas Online Services Screen

The Security for Adabas Online Services screen is called with code S in the DDA Services / Special Functions Menu of Predict.

13:38:15

***** P R E D I C T *****
 - Security for Adabas Online Services -

2007-05-31
 DDAA0SM3

| Code | Function |
|------|-----------------|
| A | Add new Profile |
| D | Display Profile |
| M | Modify Profile |
| P | Purge Profile |
| S | Select Profile |
| ? | Help |
| . | Terminate |

Enter Code : _
 File No. : _____ To File No.: _____
 Data Base ID : _____
 Predict-user : _____

or direct command:
 Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
 Help Next Term Last E-el Flip Print Impl Conf S-fi Prof Menu

| Parameters | |
|--------------------------|---|
| Code | Calls any of the functions Add, Display, Modify, Purge, Select profile, Help or Terminate. The different functions are described in separate sections below. |
| File No. ... To File No. | A profile for a file or a range of files can be defined by entering file numbers. If these fields are left blank, a profile for a database is processed. To process a profile for a single file, enter the file number in both the fields File No. and To File No. |
| Database ID | Number of the database. |
| Predict user | The profile to be processed defines the rights for this user. If a group is specified, the profile applies to each user in the group. To define a <i>default</i> AOS security profile, the Natural Security user ID/group specified as default user must be specified. See Defining Default Access Rights . |

Functions for Processing AOS Security Profiles in Predict

Add/Display/Modify Profile - Codes A, D, M

For Databases

If a profile for a database is processed with Add/Display/Modify profile, the Predict functions Incorporate and Compare database are allowed, disallowed or the allow/disallow values are displayed.

For Files

Protection of the Predict functions Incorporate, Compare, Generate file and the functions Purge and Refresh file of the Administration Implemented File menu are allowed, disallowed or the allow/disallow values are displayed in a screen as shown below.

```

13:40:20          ***** P R E D I C T *****          2007-05-31
                  - Security for Adabas Online Services -          DDAA0SM5

Display Profile for Data Base: 180   File: 1       to File: 255
Predict-user: ACCOUNT

Please specify 'Y' to allow function or 'N' to disallow

Incorporate File.....: N
Compare File.....: Y
Generate File.....: N
- with option 'STOP USERS USING FILE': N
Maintain implementation
Purge.....: N
- with option 'STOP USERS USING FILE': N
Refresh.....: N
- with option 'STOP USERS USING FILE': N

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---
      Help Next Term Last E-e1 Flip Print Impl Conf S-fi Prof Menu

```

Purge Profile - Code P

Additional confirmation is requested before a profile is actually purged.

Select Profiles for Databases and Files - Code S

```

13:38:46          ***** P R E D I C T *****          2007-05-31
          - Security for Adabas Online Services -          DDAA0SM6

          Following profiles exist for data base 180      :
          (You may mark max. 60 profiles: M:modify D:display P:purge profile)

          File      PRD-user M   File      PRD-user M   File      PRD-user M   File      PRD-user M
          -----
          1      + ACCOUNT  _           _           _           _
          1      + DEFAULT  _           _           _           _
          1      + PREDICT  _           _           _           _
          _           _           _           _           _

```

| Columns | Meaning |
|----------|--|
| File | File number or the first file number of a range of files the profile applies to. A plus sign indicates a range of files (see screen above). |
| PRD-user | User or user group whose access rights are defined in the profile. |
| M | The functions Add, Display, Modify and Purge profile contained in Security for Adabas Online Services screen can be called from the selection list by entering the respective code (A, D, M, P) in the column M. |

Protecting DDMs

Predict functions processing DDMs/files that are protected in Natural Security are affected by security mechanisms as described in the following sections.

Generating DDMs and/or Natural Security Definitions for DDMs

Generating DDMs is affected as follows:

- DDMs for files defined in Natural Security can only be regenerated (function Generate DDM applied to existing DDMs) by users authorized in Natural Security to modify the DDM.
- Natural Security definitions can only be generated for DDMs (function Generate DDM applied with the option Generate security set to Y) by users authorized in Natural Security to add the Natural Security definition of the file.

Countersignatures may be required in both cases, depending on the Natural Security definition of the file.

Generating DDMs via an Implementation Plan

Generate DDM tasks for files defined in Natural Security are not added to an implementation plan if the user is not authorized to modify the DDM. If countersignature is necessary, a generation task will be marked as impossible and the function MO (modify generation options) must be used to enter the countersignature.

When an implementation plan is executed, the system checks that neither the Predict file/field definition nor the Natural Security definition for this file was modified. Only in this case is the Generate DDM function performed.

Purging DDMs and/or Natural Security Definitions for DDMs

DDMs protected in Natural Security and/or Natural Security definitions for a DDM can only be purged with the function Purge implementation in the Administration Implemented File menu by users authorized in Natural Security to modify the Natural Security definition of the file. Countersignatures may be required depending on the Natural Security definition of the file.

Incorporating / Comparing DDMs

DDMs protected in Natural Security can only be incorporated / compared by users authorized in Natural Security to modify the DDM. No countersignatures are necessary.

Incorporating NDBs

If the function Incorporate NDB replaces Predict database objects of type I, it may be necessary to delete Predict file objects of types I, J and K linked to these databases. If DDMs have been generated from the file objects of types I, J and K, these file objects can only be purged if the user is authorized to modify the Natural Security definition of the files.

Protecting Processing Rules

Free rules can be protected with the parameters (Rule in Map Editor / Rule in SYSDIC). Predict and the Natural map editor evaluate this parameter in combination with the attribute Modifier (Natural Security user or user group) of the respective Verification object as follows

| Parameters Rule in Map Editor / Rule in SYSDIC | Modifier specified | Effect |
|--|--------------------|---|
| N | Yes or No | Rule is not protected. |
| Y | Yes | Only users specified as modifiers in the Predict verification object may change a free rule. |
| Y | No | Rule is not protected. |
| F (force) | Yes | Predict verifications must have at least one modifier. Only users specified as modifiers may change a rule. |
| D (disallow) | Yes or No | Free processing rules may not be modified in the map editor. Disallow is not applicable to Rule in SYSDIC. |

See also the Verification attribute Modifier in the section Verification in the *Predefined Object Types in Predict documentation*.

The activation of automatic rules via GENERATE RULE is protected using the definition for PRD-Ext-Object Verification rule (RU).

All other objects of type verification are protected using the definition for PRD-Docu-Object Verification (VE).

Protecting Natural Source Programs

Some Predict functions access Natural source programs. The following sections describe how these functions are affected by security mechanisms.

Redocumenting Natural Programs

To redocument a Natural program from its source, the user must be authorized in Natural Security to work with Natural utilities in the library where the program is stored.

Countersignatures may be required depending on the Natural Security definition of the library.

Selecting Text

To copy text from a Natural program with the command SELECT in the description editor or another text editor, the user must be authorized in Natural Security to work with Natural utilities in the library where the program is stored.

Countersignatures may be required depending on the Natural Security definition of the library.

25

Protecting Predict With Other Security Systems

| | |
|--|-----|
| ■ Protecting Predict using Adabas Security | 140 |
| ■ Protecting Predict with User Exit U-SEC | 140 |

This section describes how to protect your Predict environment with other methods not described in the preceding sections.

Protecting Predict using Adabas Security

With Adabas Security, access to files can be protected with passwords. The password used to control access to the Predict system file can be specified in the Natural Security definition of the library SYSDIC.

User-specific access rights for the Predict system file can be defined by creating special links from individual users to the library SYSDIC. User-specific password and access rights are then defined in the Natural Security definition of the special link.

Defining access rights for the Predict system file with Adabas Security passwords using Natural Security requires the following:

- The desired Adabas password must be entered in the Restrictions map of the Natural Security definition for the library SYSDIC or the special link.
- The command mode must be disallowed for the library SYSDIC.

If a user-specific password is specified for the Predict system file, the password specified in the Natural parameter modulefile or as dynamic parameter is not in effect when the user is working. After leaving the library SYSDIC* the old password is restored.



Caution: If as a result of protecting the Predict system file by value, the fields K* are write-protected, XRef data cannot be written to Predict. CATALOGing and STOWing Natural programs with XREF set to Y is then impossible.

Protecting Predict with User Exit U-SEC

If you set default parameter General Defaults > Protection > Protect current Predict file to Y and do not specify a database and file number of a Natural Security file, the user exit U-SEC is called.

The parameters of this user exit are described in the PDA U-SECP.

This user exit is delivered in source form and allows you to define your own security checks. If the current user does not have sufficient access, you can set the message DIC1534 ("You are not authorized...").

This user exit also provides an interface to other security products such as RACF.

This user exit also contains a description of the parameters available. See the section User Exits in the *Predict Administration documentation*.