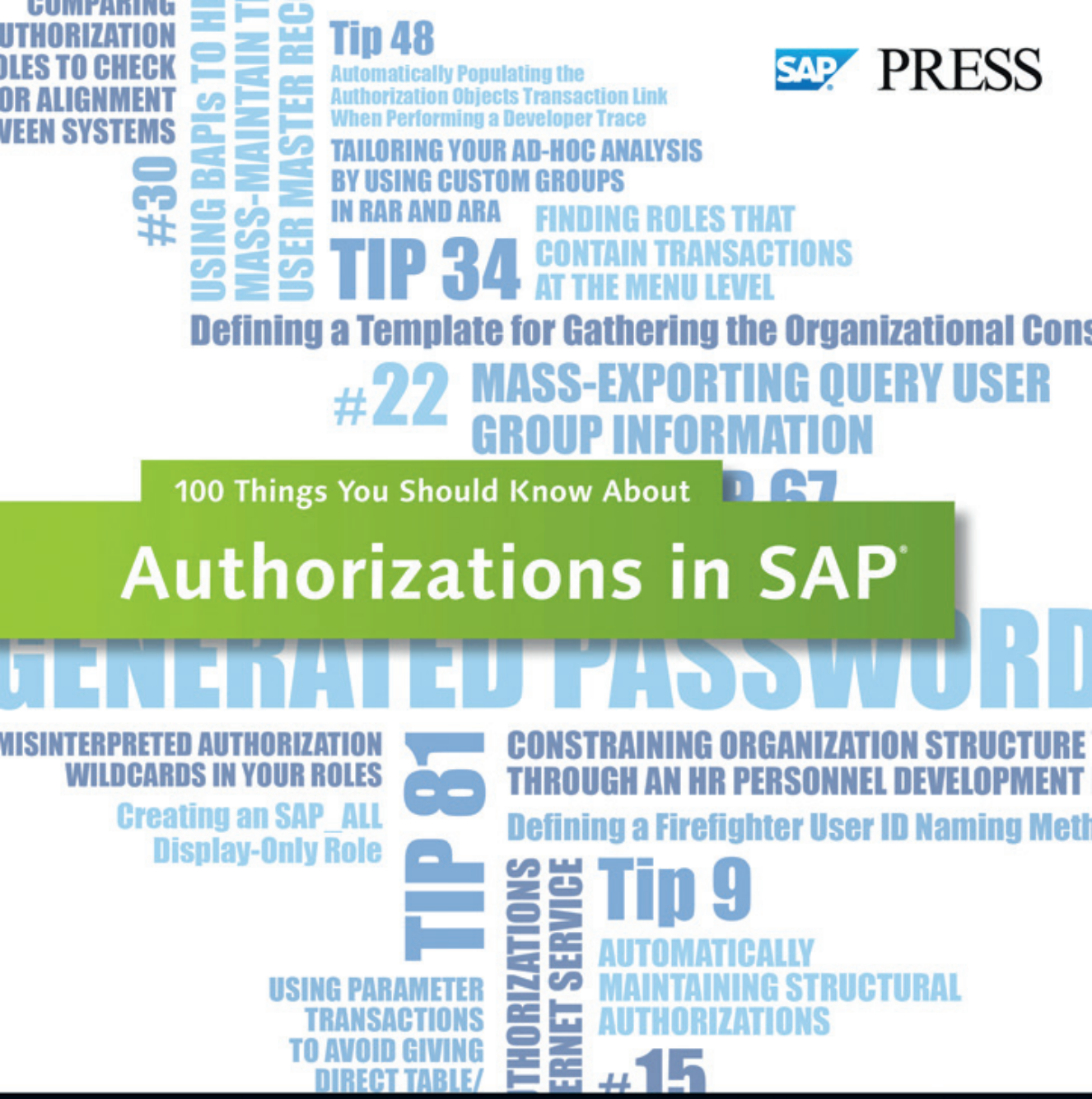


- Provides 100 little-known time-saving tips and tricks
- Features detailed instructions and guiding screenshots
- Presents practical, expert advice for system administrators

Andrea Cavalleri
Massimo Manara

Galileo Press

SAP PRESS is a joint initiative of SAP and Galileo Press. The know-how offered by SAP specialists combined with the expertise of the Galileo Press publishing house offers the reader expert books in the field. SAP PRESS features first-hand information and expert advice, and provides useful skills for professional decision-making.

SAP PRESS offers a variety of books on technical and business related topics for the SAP user. For further information, please visit our website: http://www.sap-press.com.

Mario Linkies and Horst Karin
SAP Security and Risk Management (2nd Edition)
2011, 742 pp. (hardcover)
ISBN 978-1-59229-355-1

Volker Lehnert, Katharina Stelzner, Larry Justice
Authorizations in SAP Software: Design and Configuration
2010, 684 pp. (hardcover)
ISBN 978-1-59229-342-1

Sebastian Schreckenbach
SAP Administration—Practical Guide
2011, 883 pp. (hardcover)
ISBN 978-1-59229-383-4

Steve Biskie
Surviving an SAP Audit
2010, 299 pp. (hardcover)
ISBN 978-1-59229-253-0

Andrea Cavalleri and Massimo Manara

100 Things You Should Know About
Authorizations in SAP®

Galileo Press

Bonn • Boston

Dear Reader,

As a security administrator, you have a very important and complex job. Make it easier with this book, where you'll find practical, useful tips and workarounds that will help you accomplish moderate to advanced authorization tasks. With information ranging from using templates, to managing user IDs, to setting up a security project to managing information for a smooth audit, you're sure to find helpful tidbits that will save you time and help you avoid many potential headaches.

It was a pleasure working with this book's authors, Andrea and Massimo. They are easily some of the most organized people I've ever had the privilege of working with (which isn't easy when it comes to writing about 100 separate topics!). Between being early with all of their work and answering *all* of my many questions, they ultimately created a bit more work for themselves in sheer times the book was able to pass between us, but which benefits you with a fantastic, concise book of tips and tricks. I'm confident that you'll find it just as rewarding to navigate this book as I did, and better yet, that you'll find that helpful hint that will really add a little something to your day!

We at SAP PRESS are always eager to hear your opinion. What do you think about *100 Things You Should Know About Authorizations in SAP*? As your comments and suggestions are our most useful tools to help us make our books the best they can be, we encourage you to visit our website at *www.sap-press.com* and share your feedback.

Thank you for purchasing a book from SAP PRESS!

Laura Korslund
Editor, SAP PRESS

Galileo Press
Boston, MA

publishing@galileo-press.com
http://www.sap-press.com

Notes on Usage

This e-book is **protected by copyright**. By purchasing this e-book, you have agreed to accept and adhere to the copyrights. You are entitled to use this e-book for personal purposes. You may print and copy it, too, but also only for personal use. Sharing an electronic or printed copy with others, however, is not permitted, neither as a whole nor in parts. Of course, making them available on the Internet or in a company network is illegal as well.

For detailed and legally binding usage conditions, please refer to the section Legal Notes.

This e-book copy contains a **digital watermark**, a signature that indicates which person may use this copy:

Imprint

This e-book is a publication many contributed to, specifically:

Editor Laura Korslund
Copyeditor Julie McNamee
Cover Design Graham Geary
Production E-Book Kelly O'Callaghan
Typesetting E-Book Publishers' Design and Production Services, Inc.

We hope that you liked this e-book. Please share your feedback with us and read the Service Pages to find out how to contact us.

The Library of Congress has cataloged the printed edition as follows:
Manara, Massimo.
100 things you should know about authorizations in SAP /
Massimo Manara, Andrea Cavalleri. — 1st ed.
p. cm.
Includes bibliographical references.
ISBN 978-1-59229-406-0 — ISBN 1-59229-406-5 1. Computers—Access control. 2. Computer networks—Security measures. 3. SAP ERP.
I. Cavalleri, Andrea. II. Title. III. Title: One hundred things you should know about authorizations in SAP.
QA76.9.A25M31847 2012
005.8—dc23
2012005022

ISBN 978-1-59229-406-0 (print)
ISBN 978-1-59229-803-7 (e-book)
ISBN 978-1-59229-804-4 (print and e-book)

1st edition 2012

Contents

Acknowledgments

We wish to thank Andrea's sons, Maria Stella and Nicolò; Andrea's wife, Daria; Massimo's wife, Maria Elena; and, last but not least, our parents and friends. Thank you very much for supporting and believing in us during all this time, as well as for contributing to our passion for SAP authorizations matters.

Many thanks also go to all of our Aglea colleagues, who gave us many suggestions for what might make the most helpful tips. And we thank our customers for believing in Aglea and involving us in various security, identity management, and GRC projects, which allowed us to provide the solutions described in the tips in this book.

We would also like to thank Kelly Harris, Laura Korslund, and all of the Galileo Press team for believing in this project. Many thanks for your support and assistance.

Massimo Manara and **Andrea Cavalleri**

Part 1

User Master Records

Things You'll Learn in this Section

User master records play an essential role in the authentication and authorization processes. *Authentication* is your way of identifying who is logging in to a secure domain, and *authorization* protects transactions, programs, and services in the SAP system from unauthorized access. Both are critical activities in any business. Depending on the size of your company, you might manage only a few users or several thousand. This part of the book will help user administrators and business analysts manage the daily and routine user administrator tasks in the system.

Tip 1

Displaying the Technical Names of Transactions in the SAP Easy Access Menu en Masse

You can help users save time by displaying the technical names of transactions in order to execute the transaction directly.

Each activity in SAP is performed through a transaction, which has a technical code name. You can simplify the test phase before go-live by setting up the technical transaction names. During a test phase or the transition into a live system, showing the transaction code technical name in the SAP GUI allows a user to execute the transaction directly, without wasting time finding the transaction in the menu. Although you can set this up manually for each user, this tip will show you how to mass-process this change to save time.

And Here's How ...

The first interface you see after you're logged into the SAP system is SAP Easy Access. Figure 1 shows the SAP Easy Access interface. By pressing Shift+F9, or by following the path Extras • Settings, you can click on the Display technical names checkbox.

As result of this activity, you can find the technical names of each transaction code near the description of each transaction. For example, you find Transaction VA01 near the name Create Sales Order. Therefore, as a first step after the logon in SAP, a user should set this flag.

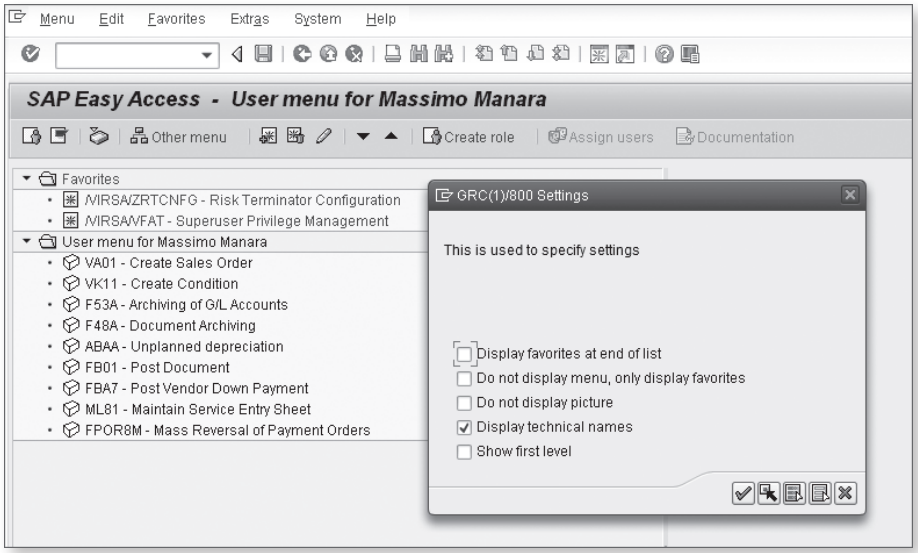


Figure 1 Display Technical Names Settings

SAP Table AGR_DATEU (Personal Settings for Roles) contains the personal settings options for the table values shown in Figure 1 for all users defined. By checking this box, you can find the technical names of each transaction code next to the description of the transaction. Each flag in Figure 1 is driven by a customizing switch in this table.

Through a standard function module, you can maintain these customizing switches without needing each user to log on and manually maintain the DISPLAY TECHNICAL NAMES FLAG. To execute the function module, follow the menu path:

SAP MENU • TOOLS • ABAP WORKBENCH • DEVELOPMENT • SE37

The DISPLAY TECHNICAL NAMES flag corresponds to the fourth switch in this function module for the user analyzed (Figure 2). If this switch contains the value X, the DISPLAY TECHNICAL NAMES flag shown in Figure 1 is active.

You can use the standard function module PRGN_SET_BROWSER_OPTIONS_USER to mass-set this customizing switch for each user. Unfortunately, this function module doesn't allow a mass-maintenance mode. For this reason, you need to write a small piece of ABAP code to enhance the standard function module for allowing a mass maintenance.

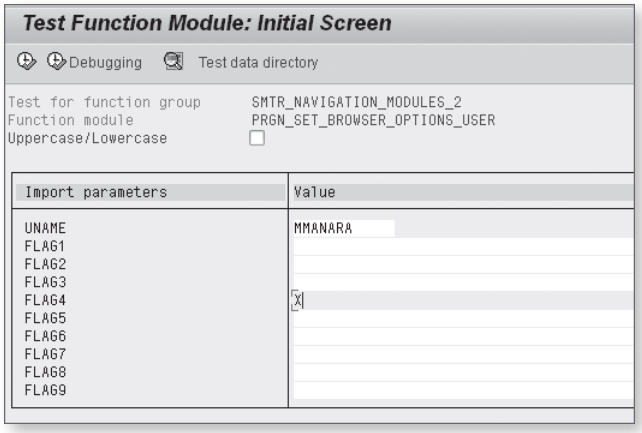


Figure 2 PRGN_SET_BROWSER_OPTIONS_USER Function Module with FLAG4 Active

The following listing shows an example of ABAP code to use the standard function module on several users:

```
REPORT  Z_SET_MENU.
TABLES : USR02.
DATA: T_USR TYPE TABLE OF USR02,
      W_USR TYPE          USR02.

SELECT-OPTIONS: TOP_USR FOR USR02-BNAME.

CHECK TOP_USR-low <> SPACE.

SELECT * INTO TABLE T_USR
  FROM USR02
  WHERE BNAME IN TOP_USR.

LOOP AT T_USR INTO W_USR.

  CALL FUNCTION 'PRGN_SET_BROWSER_OPTIONS_USER'
       EXPORTING
            UNAME = W_USR-BNAME
            FLAG4  = 'X'.

ENDLOOP.
```

Tip 2

Improving Your User Master Record Accuracy with Hidden Fields

You can enhance the completeness and accuracy in your user master records by entering custom data via hidden fields in Transaction SU01.

By default, some user master record fields in Transaction SU01 are not displayed. This may cause a problem because some of the attributes in the fields that aren't displayed can be used to keep the user master data clean and updated for better accuracy and to enter custom data to better classify your user master data.

This tip is addressed especially to those companies that do not have SAP ERP Human Capital Management (SAP ERP HCM) in place (so it's not possible to directly link the user master record with personnel number HR data). Refer to Tip 13 if your company *does* have SAP ERP HCM in place.

And Here's How ...

This tip is most useful in the following situations:

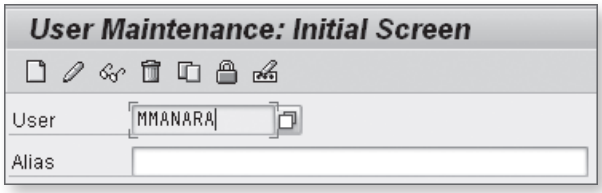
- An interface exists that automatically synchronizes and updates the user master record by using an external HR repository (not SAP) as master data. You can use these supplemental fields to map some HR attributes (company code, department, personnel area, etc.) in Transaction SU01.
- SAP ERP HCM is not in place and you need to classify your users by several taxonomies (cost center, division, region, skill, and other business-company custom attributes requirements).

You can gather all of this information by enquiring and correlating Tables USR02, USR21, ADRP, ADCP, and ADR6. Or, you can use table views V_USERNAME and USER_ADDR.

To access the user master record maintenance transaction, use the menu path:

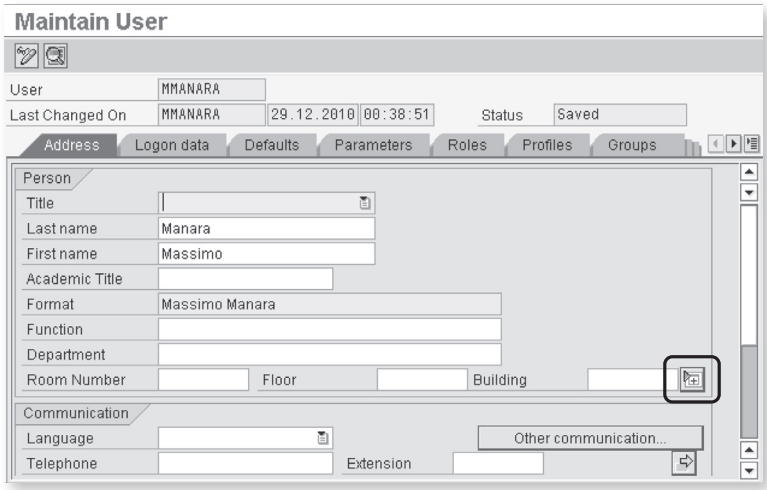
SAP Menu • Tools • Administration • User Maintenance • SU01

Click Execute, enter a user ID, and then click on the pencil icon. The screen that appears allows you to maintain a user master record (shown in Figure 1).



« *Figure 1 Transaction SU01: User Master Record Maintenance*

You can change the user attributes by clicking on the pencil icon again. Click on the magnifying glass icon in Figure 2 to expand all hidden fields in the user master data via the More Fields button.



» *Figure 2 The More Fields Button in Transaction SU01*

Now you will see several other fields that you can use to enhance your user master record data and keep it accurate (shown in Figure 3).

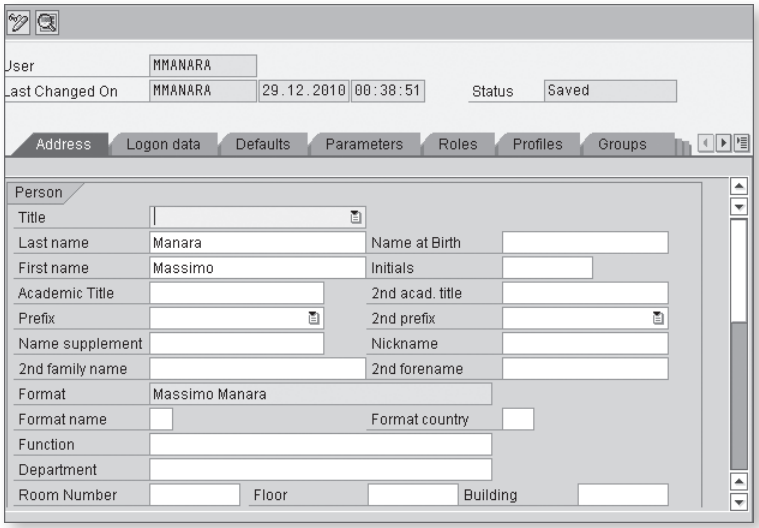


Figure 3 Hidden Fields in Transaction SU01

Tip 3

Defining an SAP User ID Naming Convention to Manage User Master Records

You can define an SAP user ID naming convention to simplify and better manage user master records during the lifetime of an SAP system.

You can use Transaction SU01 when you need to define an SAP user ID. Changing a user-naming convention when the system is live can be very difficult or impossible, so this tip will help you establish an effective naming convention that you should never have to change.

And Here's How ...

If you use any type of personal information in a name, it may be subject to national and international laws (e.g., privacy law), legacy software authentication rules, language peculiarities, and system technical limits. The best approaches to define an SAP user ID naming convention are outlined here (using the name Galileo Galilei as an example):

- You can concatenate a few characters from a person's last name with the first name (or vice versa), and enter an incremental number to avoid duplicate homonyms.

1	2	3	4	5	6	7	8	9	10	11	12
G	A	L	I	L	E	O	G	A	L	0	1

- Use the user's personnel number defined by the HR department during the hiring process.

1	2	3	4	5	6	7	8	9	10	11	12
8	7	3	4	5	7	8					

- Use a mixed mode (say, the personnel number concatenated with the person's last name).

1	2	3	4	5	6	7	8	9	10	11	12
8	7	3	4	2	G	A	L	I	L	E	I

- Use a different coding based on other attributes (e.g., country/region).

1	2	3	4	5	6	7	8	9	10	11	12
U	K	0	0	0	1						

However, note that these methods have a few disadvantages:

- Some national or international laws can treat the first name and last name as personal data, which can cause some law restrictions.
- Concatenating some characters of the last and first name from a different language/country can generate some unexpected user IDs (e.g., Walbake Lynne can become WALLY01; in England, *wally* is a colloquial way to say *stupid*).
- Using special characters (_ # \) in user ID naming conventions cannot be supported from external devices (e.g., a radio frequency device).
- Defining a user ID naming convention that's based on an attribute that can change during the entire lifecycle of a user can cause a renaming. For example, if a user moves.

General Tips and Tricks

Remember, the maximum length for a user in the SAP system is 12 characters.

The best choice for an SAP user that represents a person is based on the personnel number. Also keep in mind that having names of an equal length for all users in the company can be helpful while processing data into a spreadsheet.

For SAP users that represent a technical interface, there is no specific advice or rules to follow for a naming convention, except documenting what these users do and how they do it. This is useful when it's necessary to edit something at the user level.

Tip 4

Using BAPIs to Help Mass-Maintain the User Master Record

When your system contains several thousands of user IDs, you can save time by managing these user IDs with mass changes.

You can use Transaction SU10 to mass-maintain the user master record. Unfortunately, you can't make changes to certain areas in this transaction (e.g., you can't maintain a user's address tab). You can fill these gaps by using specific BAPIs (Business Application Programming Interface).

And Here's How ...

Browse BAPIs by function group using Transaction SE80. After you've identified the function you're interested in, execute it by accessing Transaction SE37. To perform mass changes in the user master record, maintain the Function Module field (the BAPI_USER_CHANGE BAPI is used here), and click the Display button.

Now that you've found a BAPI that could help you, you have to decide how it will be called in the system. If you use it directly in the SAP ERP system, all BAPIs are valid, but to use it with external programs, you have to verify that the BAPI can be called remotely. Verify this by looking at the Remote-Enabled Module selection in the BAPI Attributes tab displayed through Transaction SE37 as shown in Figure 1.

Fill in the fields or tables with all of the necessary values, and call the BAPI. The result will be in the output data.

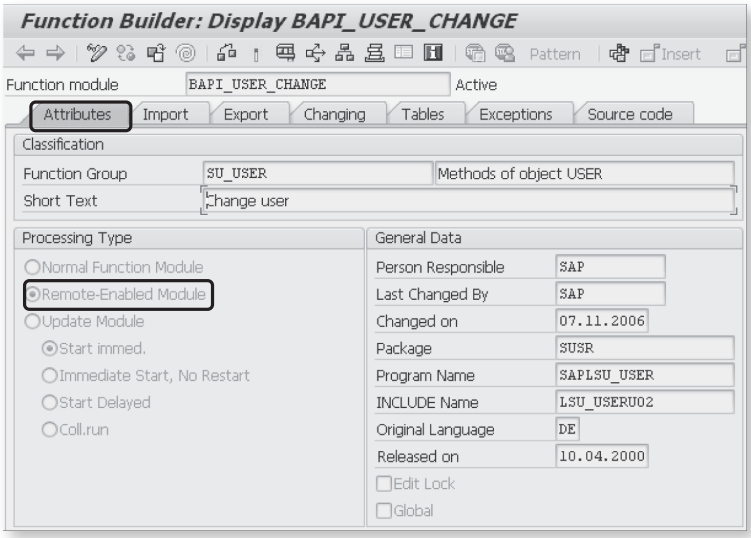


Figure 1 Transaction SE37 and BAPI Attributes

Example

Suppose you have received the function and department information for all SAP end users from HR. You need to populate the two fields highlighted in Figure 2 in the Address tab of Transaction SU01 for each user.

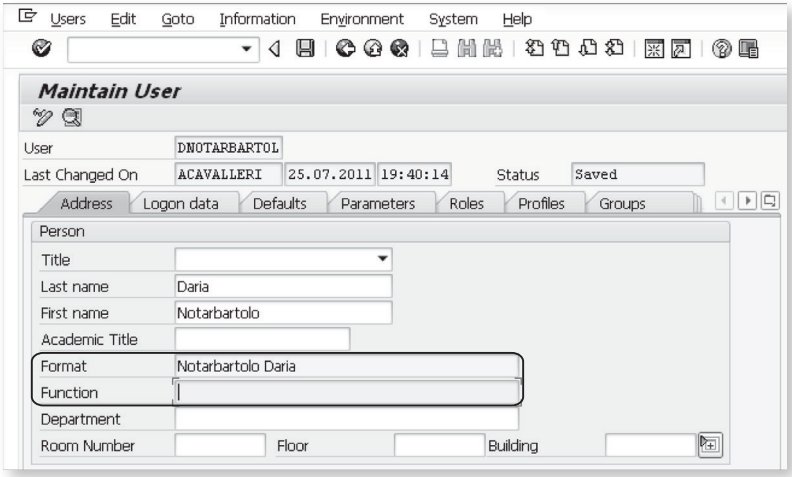


Figure 2 Fields Involved in Mass Change

Modify these two fields using the B_USER_CHANGE BAPI. After you specify the BAPI name in Transaction SE37, you can use it by clicking on the TEST button (see Figure 3).

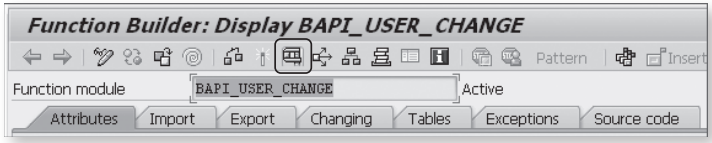


Figure 3 SE37 Transaction BAPI Test Functionality

After the BAPI has been started, specify the import parameters. Looking at Figure 4, notice the two different kinds of parameters: the field type (as USERNAME) and the table type (as LOGONDATA and LOGONDATAX). This example uses USERNAME to specify the user ID involved in the change. The FUNCTION and DEPARTMENT fields are available in ADDRESS table. Notice that the table names are very similar to Transaction SU01 tabs (ADDRESS, LOGON DATA, DEFAULT, etc.).

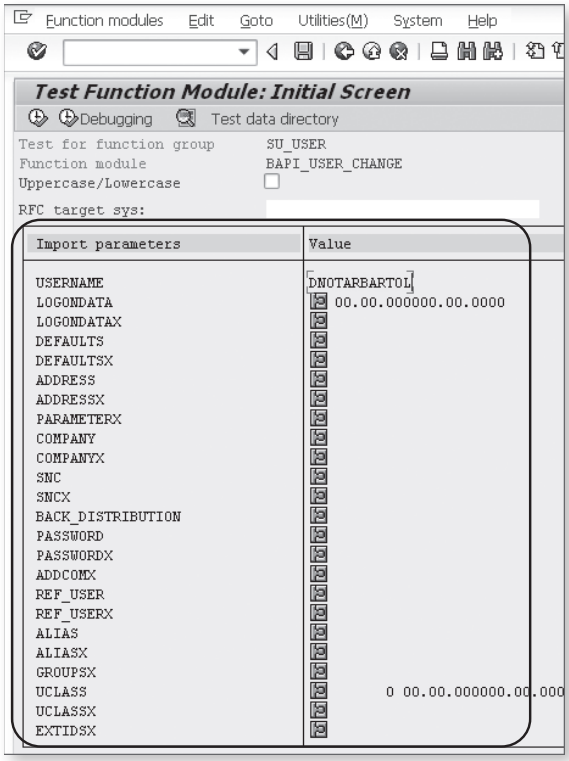


Figure 4 BAPI Import Parameters

Specify the values in the first table (ADDRESS). In the second table (ADDRESSX), specify the involved fields by double-clicking on these fields.

Updating values into the correspondent X-tables (in this example, ADDRESSX) is not as simple because the field names are not visible. As shown in Figure 5, you can click on a field and press F1 to see the technical name. In this example, DEPARTMENT (D) and FUNCTION (F) fields are set as X to update these fields. Because the first letter of a field is not enough to convey where it is, look at the result of the METADATA button (above the fields).

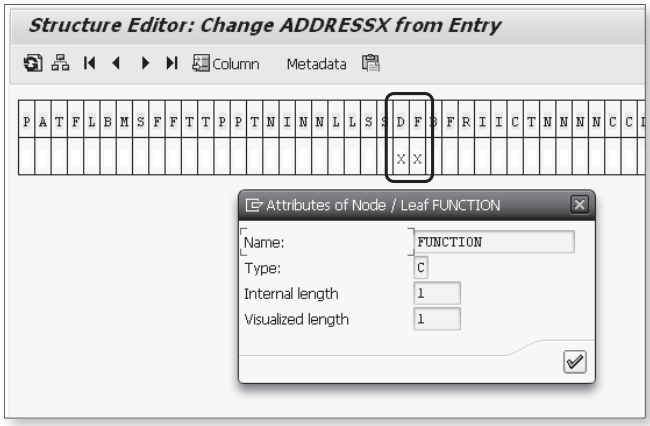


Figure 5 BAPI Table Parameters (Involved Fields)

Last, execute the BAPI to update the user master record, and then click on the EXECUTE button.

If all parameters are correct and the BAPI was successfully called, you will see a runtime value as shown in Figure 6.

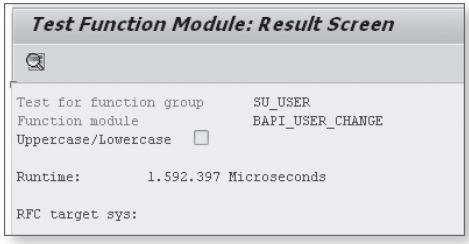


Figure 6 Valid Runtime Value

The final result will be visible in Transaction SU01, and the FUNCTION and DEPARTMENT fields have been updated.

Tip 5

Customizing the Rules for Automatically Generated Passwords During User Creation

You can customize the rules that Transactions SU10 and SU01 use to automatically generate a random password to match customer password complexity requirements.

With Transactions SU10 and SU01, you can exploit a standard functionality to generate a random initial password. By default, SAP generates passwords of 40 characters with numbers, letters, and special characters.

However, this kind of password is sometime not suitable for end users—it's too difficult to enter because it's so long. There are other issues associated as well, such as the user locking himself out of the system with too many failed attempts, or your need to create some test temporary users.

And Here's How ...

The default rules in SAP generate a very complex random password. For example, clicking on the magic wand icon shows the generated password in Transaction SU01 (see Figure 1).

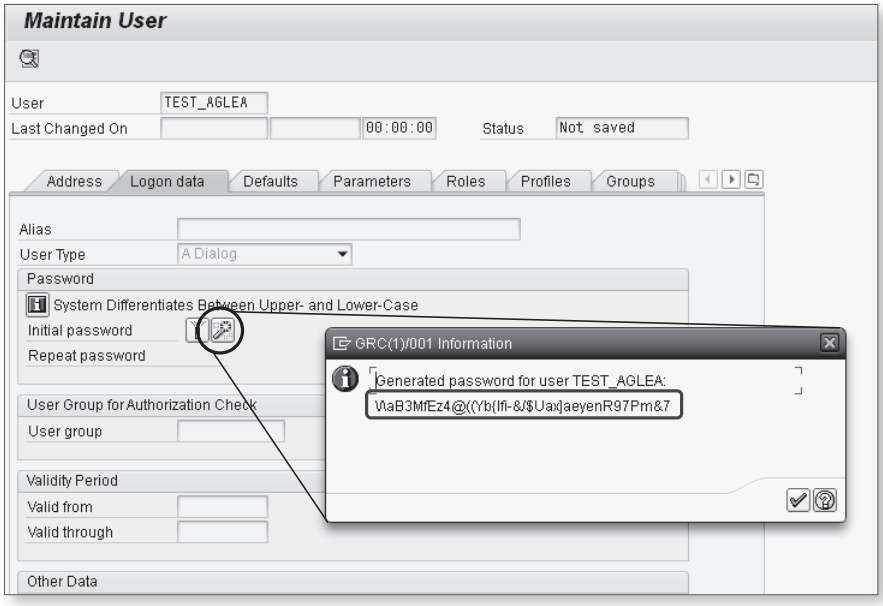


Figure 1 Generated Password in Transaction SU01

You can control automatic password generation rules with Table PRGN_CUST (Customizing Switches), as shown in Figure 2.

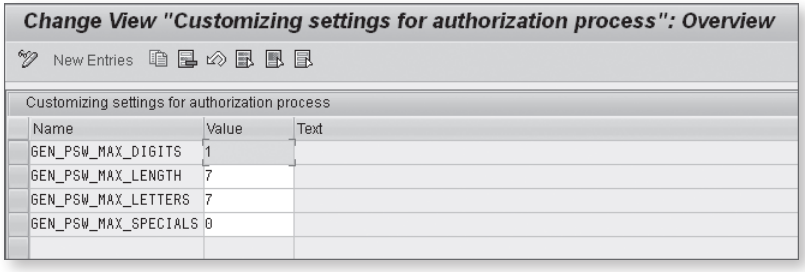


Figure 2 Customizing Switches for Password Generation Rules

You can maintain these switches with Transaction SM30 (Call View Maintenance):

- GEN_PSW_MAX_DIGITS: Maximum number of letters in the generated password
- GEN_PSW_MAX_LENGTH: Maximum length of the generated password

- GEN_PSW_MAX_LETTERS: Maximum number of digits in the generated password
- GEN_PSW_MAX_SPECIALS: Maximum number of special characters in the generated password

Figure 3 shows the results of these customizations using Transaction SU10. After the creation of three new users, the passwords generated by SAP are compliant with the customized switches. For example, test user TEST_AGLEA01 has the password k7ZeEGR. This password contains one digit, six letters, has seven characters as its max length, and does not contain special characters.

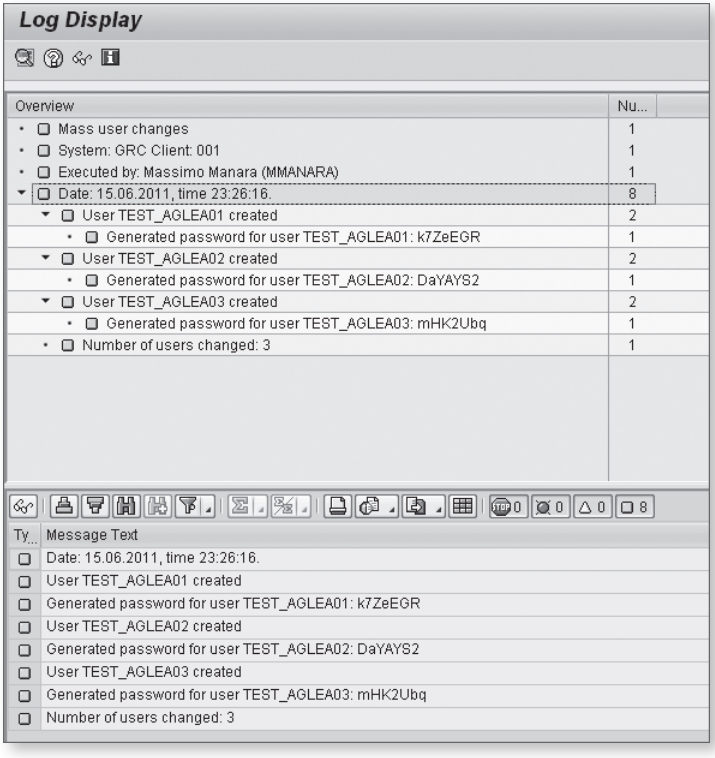


Figure 3 Generated Password in Transaction SU10 for Three New Users

Tip 6

Finding and Using User Parameters to Prepopulate Transactional Fields

You can use user parameter IDs to prepopulate some fields in transactions, which can speed up the end users' work or influence a transaction flow.

If you know a specific parameter ID and the value to put into the system, it's easy to maintain user parameter IDs with Transaction SU01 to speed up an end user's work or put constraints on a transaction behavior. However, this process isn't as simple when you don't know which field can be used through parameter IDs.

And Here's How ...

To select a field to insert as a prepopulated field in a user parameter ID, press F1 in any transaction field to access the Performance Assistance pop-up window (see Figure 1).

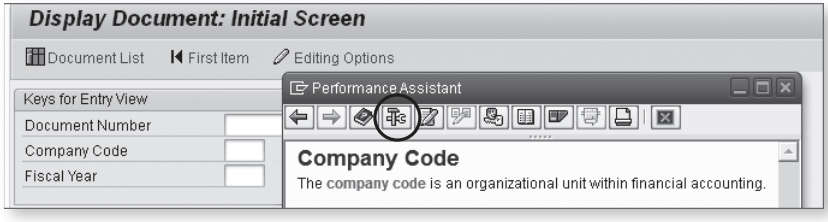
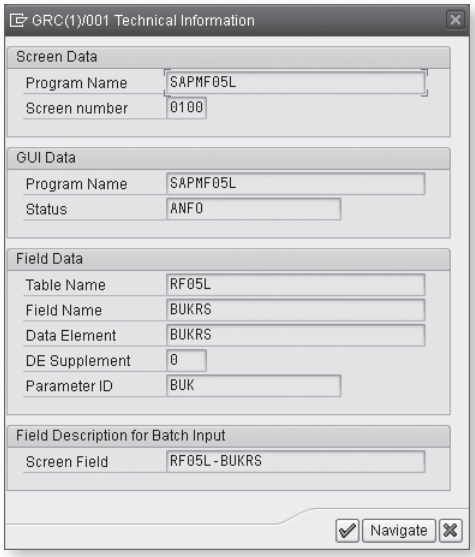


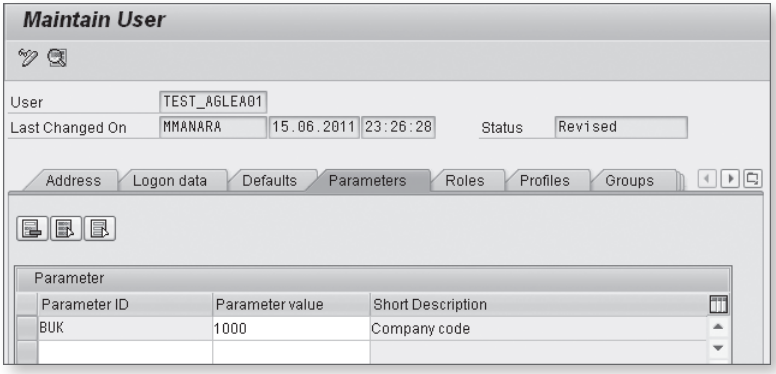
Figure 1 The Performance Assistance Pop-Up Window Includes the Technical Information Button

Click on the TECHNICAL INFORMATION button circled in Figure 1. The screen shown in Figure 2 will appear, which shows the parameter ID.



« *Figure 2 Technical Information Pop-Up with Parameter ID Code Field*

After you access this information, copy the parameter ID (BUK). Then go to the Transaction SU01 PARAMETERS tab, and paste the parameter ID value as shown in Figure 3. This means that this user ID (TEST_AGLEA01) in each transaction that contains the company code fields will be prepopulated with a value of 1000; 1000 in this example represents the company code.



» *Figure 3 Transaction SU01, Parameters Tab Showing the Tab Parameter ID Value*

Keep in mind that not all transaction fields have an assigned parameter ID. Some parameter IDs can be defined from a customizing transaction, which can change the transaction flow behavior. Let's look at some examples of this:

- Through Transaction OMET, you can configure a purchase document reference customization. This setting is a driver for a parameter ID that allows you to use it within the user master record to design your purchase document reference process.
- As a note for administrators, leaving user parameters maintenance at the end-user level can be critical in some circumstances (e.g., there are some parameters ID that drive a transaction flow), so it's best not to do this.
- Through Transaction SU3, every user can maintain his own parameter ID. With this transaction a user can change his address, last name, first name, department, and other data in Transaction SU01 with only three tabs (ADDRESS, DEFAULTS, and PARAMETERS tabs). The use of this transaction by end users is discouraged because of this reason. End users should only have access to Transaction SU50 where there are only two tabs present (the ADDRESS tab in only view mode and the DEFAULTS tab).

Tip 7

Improving Your Business Reporting through User Groups

You can use Transaction SU01 to maintain your user group concept, which will help improve your reporting, better maintain your user master data, and ensure enhanced governance.

To ensure the control and governance of master data, it's essential to keep SAP user master records up to date, as well as set up a reaffirm process. Introducing and using user groups can improve productivity during mass user ID updates and enhance user review. You can use user groups to speed up mass maintenance (Transaction SU10), improve user reporting, and lock or unlock several users in the same group, which is useful during the roll-out phase.

And Here's How ...

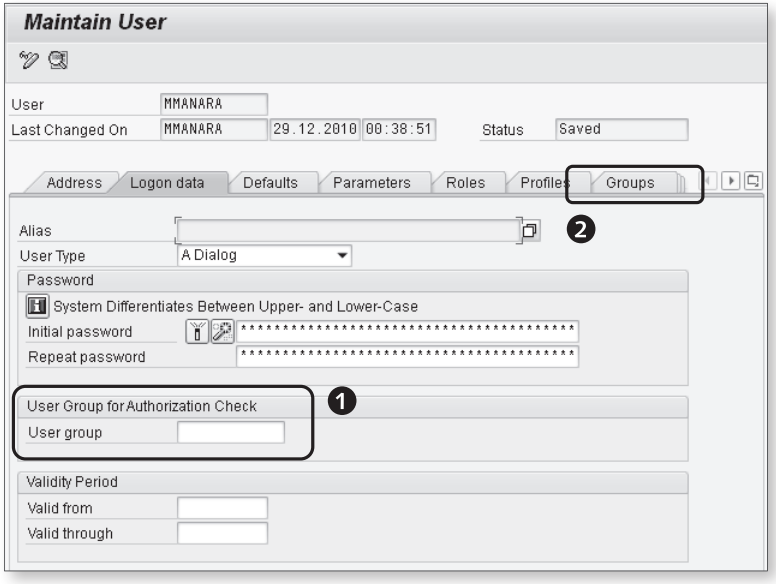
You first need to create a user group by accessing Transaction SUGR via the following menu path:

SAP Menu • Tools • Administration • User Maintenance • Sugr

The screen shown in Figure 1 is displayed. Keep in mind that you can't transport user groups that you have created from the development system to the production system. You have to manually create the user groups directly in the production system.

« *Figure 1 Transaction SUGR*

After creating a user group in the production system (in the screen shown in Figure 1, enter the user group name, and then click on the NEW button), you can assign it to all of your users. You can find the two types of user groups we've already discussed in Transaction SU01, as shown in Figure 2.



» *Figure 2 User Group Field and Groups Tab*

The first user group in the LOGON DATA tab is designed to manage authorizations in the user master record maintenance process by using authorization object `S_USER_GRP` (in this case, one user has one group assigned). The second user group is designed to enable you to assign more than one user group to one user, which you can find in the USER GROUP tab (Figure 2):

- In the LOGON DATA tab, USER GROUP field (❶)
- GROUPS tab (❷)

Each group concept can be used for reporting. Avoid using a mixed-mode grouping because maintaining both assignments can be a lot of work as a daily maintenance activity.

There are three ways to assign a user group at the user level:

- Use Transaction SU10 to select all users and then assign the user groups. Use this approach when you have to assign a group to several users.
- Use Transaction SU01 for daily maintenance. After the user group is defined, you can enter it into the USER GROUP field or GROUPS tab of Transaction SU01, as shown earlier in Figure 2.
- Use Transaction SUGR to assign the group to the users in the GROUPS SU01 tab.

After you have defined the grouping you can do the following:

- Exploit the group to mass-manage and select user IDs.
- Use Transaction SUIM (User Information System Transaction) to quickly extract user information and user group selection criteria, user data, and reports.
- Allow report readers to filter or create groups based on their needs.

If you need to revalidate your user master records, you can easily split the user master records by your own grouping concept (see Part 8, Tip 92).

Tip 8

Working with Inactive Users

When a user ID is not used after being created, you can decide whether to delete it from the system or deactivate it.

Every time an employee leaves the company, the SAP user managers must update the corresponding user ID. Many companies decide to keep the users and never delete them; others decide to delete the users. Which is the best strategy?

Many companies always keep users due to tracking requirements. In this case, they just lock the user and/or set the validity date as expired in the user master record. Very few companies clean the user's authorization—removing roles and profiles—because if the user needs to be live again, it's easier to unlock the user and reset the expiration date.

In this tip, we'll help you decide what decision is best for your company, based on our experience.

And Here's How ...

Companies often don't delete users for the following reasons:

- This keeps all users in order to maintain the history of the system.
- Removed users could be erroneously activated.

As an alternative, some companies lock most of the users in the annual closing period to avoid writing records in the database. When a task is finished (e.g., mass users lock, system administrator activities), the company performs a mass unlock, but the danger with this method is that the company might reintroduce users that should remain locked.

Again, when companies perform the annual revalidation or make a Segregation of Duties (SoD) risk analysis, they will have to manage users that are no longer in the company because they often forget to distinguish between active users and inactive users.

However, we suggest that you delete inactive users in order to maintain a clean system. Remember that the history tables (USH*) will preserve the data, and it will be easy to retrieve a user's data through Transaction SUIM history reports. If the final decision is to delete the users, it's important to use Transaction SM37 to check whether jobs are scheduled for that user.

If the final decision is to maintain the users, lock the user and always set the expiration date. Locked users will be considered in licensing; an expired user will be not considered. The most important risk is related to the authorizations (roles and profiles). Also, always remove roles and profiles assignments. Even if the user has to be activated again, it makes sense to reassign new (and validated) authorizations instead of using the old one. If you regularly download Tables AGR_USERS and UST04, you will have a snapshot of roles and profiles assigned to users.

Finally, if you want to know which users are actually locked or expired, you can browse Table USR02 as shown in Figure 1.

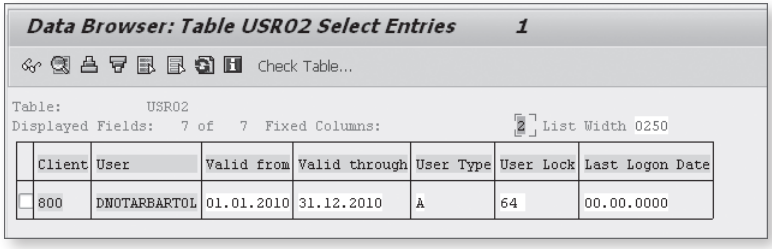


Figure 1 Table USR02 Record

Tip 9

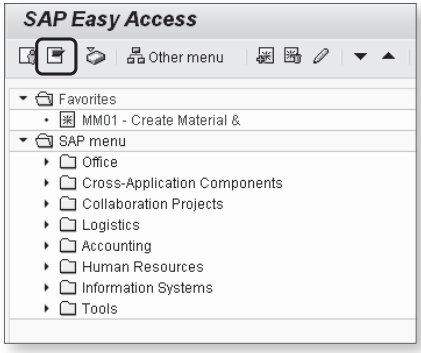
Customizing SAP and User Menus through the Session Manager

You can use SAP Easy Access to customize the user menu through the standard SAP transactions menu or by using the menus inherited from all authorization roles.

Before going live with an SAP system, it's essential to determine the menu policies for the business because changing the menu policy after all users use the system can create a lot of confusion. The users may see different menus from one day to another, which can cause several help desk calls and decreased productivity. You can allow users to switch between the user menu and the SAP menu, or set a default for all users. Each method has some advantages and disadvantages, as discussed in this tip.

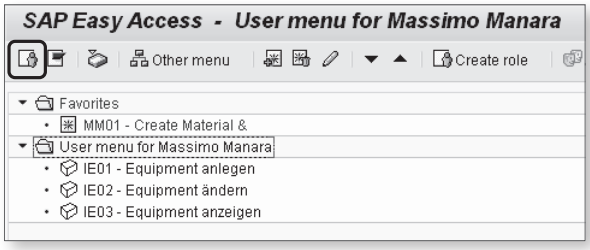
And Here's How ...

You can customize the session manager to allow the user to see and access the user menu or the SAP standard menu. Figure 1 shows the icon to display the SAP standard menu.



« *Figure 1 The Icon Used to Display the SAP Standard Menu*

Figure 2 shows the icon for the user menu. The user menu is the sum of all menu roles assigned at the user level.



« Figure 2 Icon Used to Display the SAP User Menu

However, you should note that allowing users to switch between these two menus can have some disadvantages:

- The role menu is complicated and time consuming to maintain overall if you use the composite role menu.
- Each user might have a different menu (because each user can have a different role assigned), so if a transaction is present in more than one role, it is replicated. Several overlapping transactions might arise.
- When the user menu contains several transactions, the user menu transaction list is split into several different lists, which can cause difficulty when the user is searching for a transaction.
- The user training cannot be based on SAP standard menus because each user has his own menu.

We recommend using the default SAP menu for all users and disabling the SAP user menu icon. To do this, access Table SSM_CUST, where you can disable or enable the use of the user menu icon in Figure 2. There are two customizing switches: SAP_MENU_OFF for enabling or disabling SAP standard menus (Figure 1), and ALL_USER_MENUS_OFF for enabling or disabling user menus (Figure 2).

Here, you can also find several others customizing switches to improve the performance and usability of an SAP user menu (sorting, removing duplicates, etc.).

After you've set up the menu rules, leave only the SAP menu or user menu active, as this will be active for all users defined in the system. You can create exceptions at the user level by entering a user in Table USERS_SSM and determining which menu this user should see. However, this approach is not recommended because it introduces an alternative way to manage users and the SAP menu.

Tip 10

Assigning Roles through an Organization Structure without SAP HCM Deployed

You can directly assign a role to a user by using Transactions SU01, PFCG, or SU10, or you can do so indirectly by using the organization structure.

You can use the organization structure in the SAP ERP HCM component to assign a job role at the position level (indirect assignment), instead of directly for the user, and then all users assigned to this position will inherit all authorizations. Indirect role assignment can enhance and improve the exchange of authorization data requirements and the governance awareness between the basis/security department and the business department (financial, sales, production, etc.). If those in the business department understand the organization structure's design, they'll have an easier time understanding the authorization assignment.

And Here's How ...

You can create an organization structure in Transaction PPOC to improve the awareness of the business department by enhancing the exchange information between the business department and the IT department. The business department can more easily understand an organization structure rather than a view of many user roles and authorization matters.

First, create the root. After executing the transaction, enter the root name (in Figure 1, this is COMPANY TEST). Then, during maintenance, you can use Transaction PPOMW to assign roles and users at the position level (shown in Figure 1) by right-clicking on a position and then clicking on Assign.

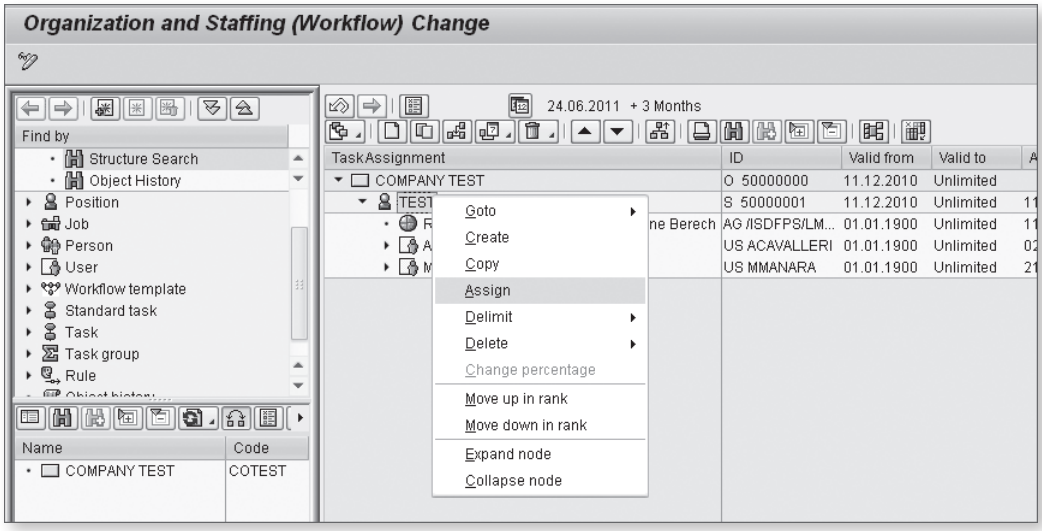


Figure 1 Assign Roles or Users with Transaction PPOMW at the Position Level

When you assign a user to a position, the user doesn't immediately receive the authorization; you first need to perform user master data reconciliation with Transaction PFUD. This transaction checks whether a user should receive a role (because it has been assigned at the position level). If the answer is yes, assign the role and the authorization profile in Transaction SU01. You can find the organizational role assignments in Transaction SU01; they appear in blue text in the Role tab. Direct roles are shown in Transaction SU01 in black.

Tip 11

Constraining Organization Structure Visibility through an HR Personnel Development Profile

You can use a personnel development profile to limit users from seeing specific aspects of the organization structure.

When a user executes Transaction PPOM, that person can see all organization structures defined in the system, which can create a security risk. With a personnel development (PD) profile, you can constrain Transaction PPOM's visibility to only certain organizational objects. For example, an organization structure can be formed by several branches. Under the company root for each country, you can have the corresponding branch company: Company UK, Company IT, Company US, and so on. If your company requires that each company administrator can administer only his own branch, a PD profile can achieve this.

And Here's How ...

To set up the structural authorization profile, perform these steps:

1. Activate customizing switch ORGPD in Transaction OOAC (as shown in Figure 1).
2. Decide how to manage SAP* user.
3. Define a structural profile via Transaction OOSP.
4. Assign a structural profile to users via Transaction OOSB.

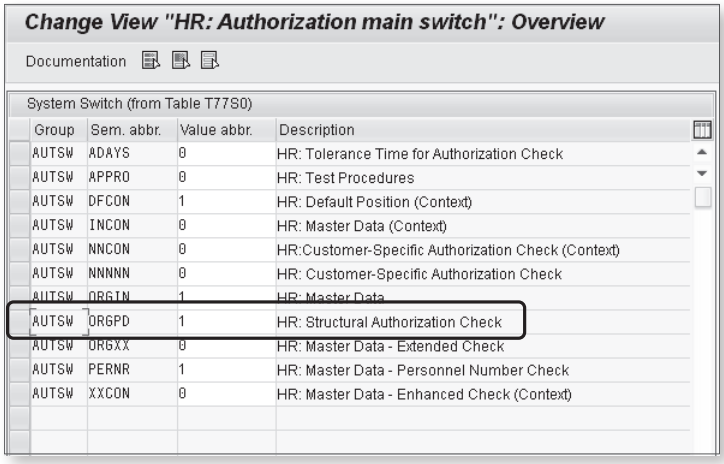


Figure 1 ORGPD Customizing Switch

By default, there's an entry in Transaction OOSB that contains the special user SAP* assigned to the ALL structural authorization profile. This entry means that if a user isn't in this table, that user will inherit the SAP* structural profile. You can leave this user, SAP*, or otherwise assign this special user as a dummy structural authorization profile to ensure that if a user isn't in this table, he cannot see any of the organization structure.

You can define the structural profile through Transaction OOSP (determine which objects the users will be able to see).

Figure 2 shows you Transaction OOSP during PD profile definition.

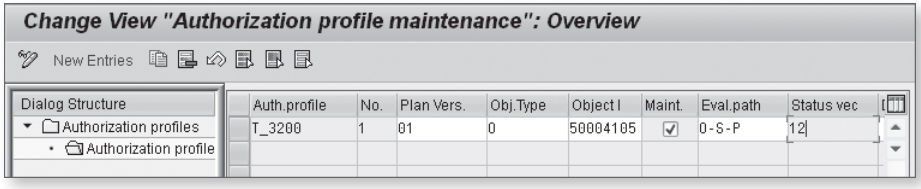


Figure 2 Transaction OOSP during PD Profile Definition

You need to define several attributes of the PD profile. Here we recap only the header information. You can read the SAP help for details of each header meaning by clicking on the field and then pressing F1.

- Auth. Profile: Profile name
- No.: Sequence number
- Plan Vers.: Plan version
- Obj. Type: HR object type representing the root of the PD profile
- Object ID: HR object ID where the PD profile starts
- Maint.: Maintenance flag
- Eval.path: Evaluation path used by the PD profile
- Status vector: Status of the relationship
- Depth: Organization depth tree used in the PD profile
- Sign: Process structure from top or bottom
- Period: Authorization according to the validity period of the structure
- Function module: Specify a function module to dynamically determine the root object of the PD profile without populating the Object ID field

At the end of the process, after the structural authorization profile is defined, you need to assign it to a user by accessing Transaction OOSB and entering the profile name and user ID.

After the users are in Transaction OOSB, the profile is immediately active and assigned. The SAP table that contains the user profile assignments is T77UA.

Tip 12

Automatically Maintaining Structural Authorizations

You can improve PD profile maintenance from a manual approach, where you assign profiles at the user ID level, to automatic maintenance. You can speed up this process through an SAP standard report.

Personnel development (PD) profiles are useful for constraining organizational data. When the company size is too large to allow a manual process to maintain users and PD profile assignment, you need to adopt an automated solution. Through standard Report RHPROFL0 you can automate the structural profile and standard authorization role assignment to bypass this issue.

And Here's How ...

By exploiting the RHPROFL0 functionality, you can assign roles and structural profiles; for example, at the position level. This ensures that the role and structural profile are correctly assigned at the position level, and it enables you to move, delete, and add users without needing to worry about authorization assignment. Users that fall below a certain position inherit all authorization.

You can use Transaction PO13 to maintain all position relationships (as shown in Figure 1). Enter the Position ID as shown in Figure 1 (50000001), select Relationships, and click on the New icon at the top left.

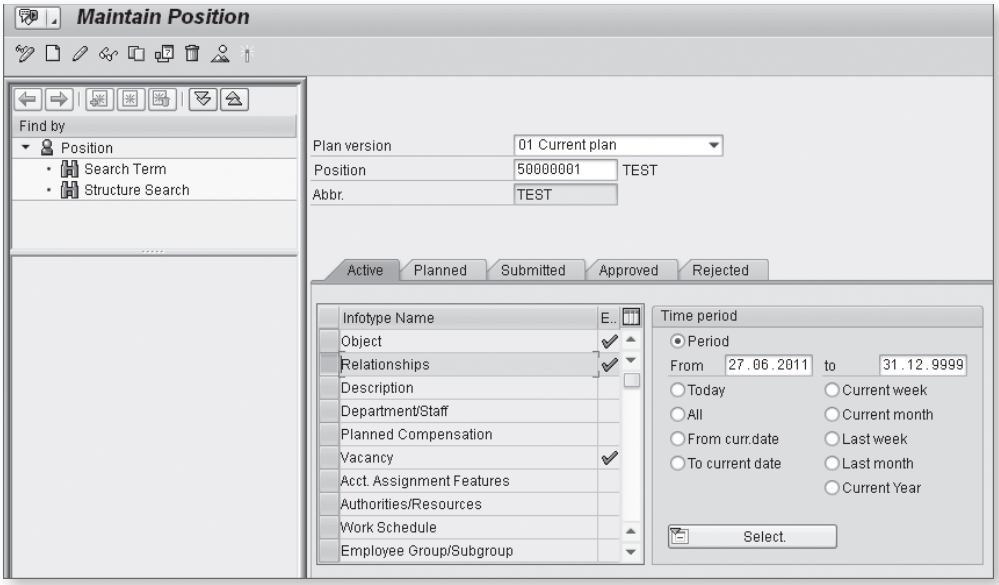


Figure 1 Transaction PO13 Maintains Relationships

Next, assign the role (see Figure 2) and structural profile assignment (see Figure 3) at the position level. To do this, enter the relationship type (in this case, B and 007), select the TYPE OF RELATED OBJECT (AG ROLE), and enter the ID OF RELATED OBJECT (TEST_AG) as shown in Figure 2.

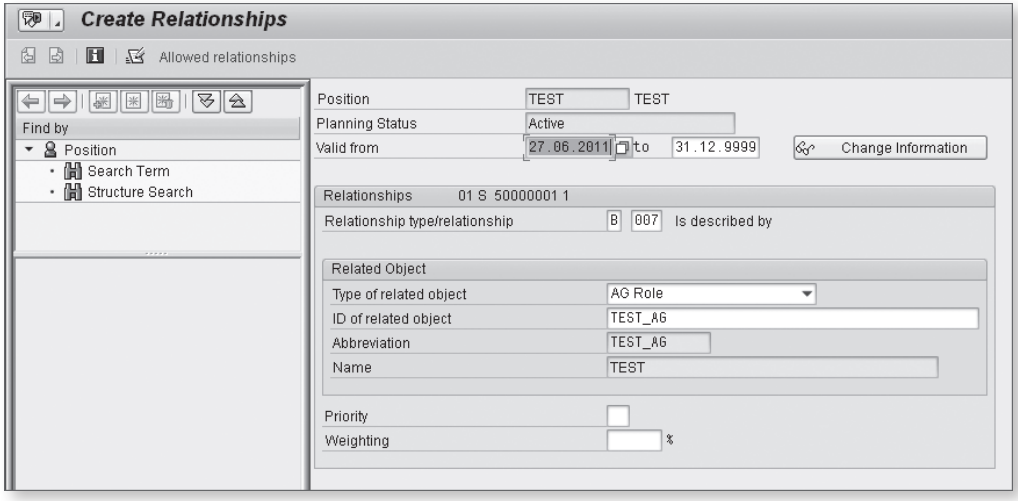


Figure 2 Role Position Level Assignment Relationship in Transaction PO13

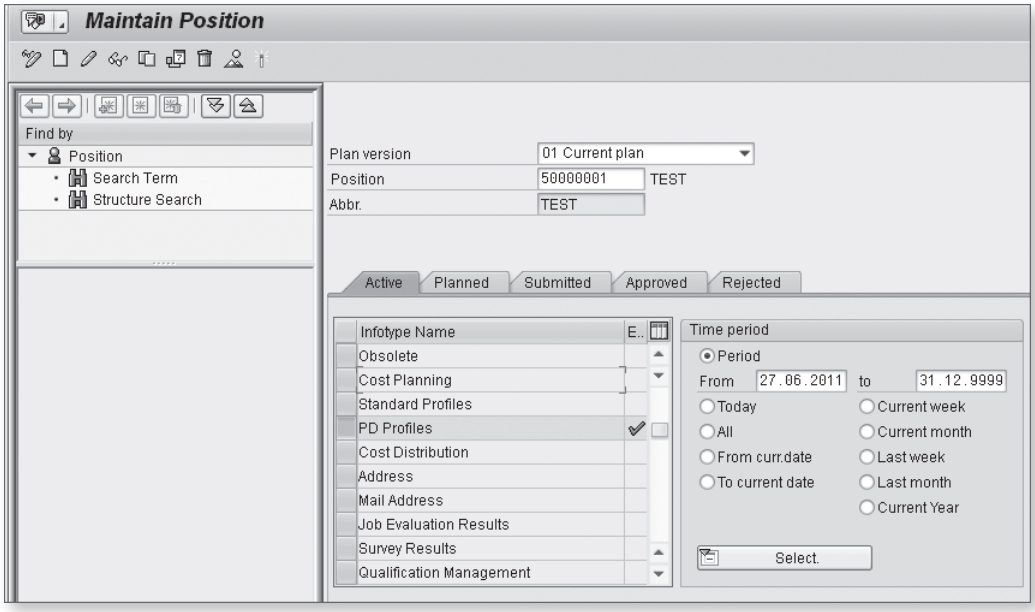


Figure 3 PD Profile Relationship Assignment in Transaction PO13

You then assign the PD profile at the position level through Infotype 1017. Browsing Table HRP1001, you can see all of the relationships of an organization's objects (e.g., roles and users assigned). Through Tables HRP1017 and HRT1017 or view HRV1017A, you can see all of the structural profiles assigned at the position level. This can help you to verify and check whether all assignments are properly done.

Tip 13

Linking User Master Records to HR Data

Because the personnel number master record is often more accurate and maintained than the user ID master record, you can use Infotype 0105 to link these two records together.

In a personnel master record, Infotype 0105 links a personnel number to a user ID. If this infotype is populated you can see all HR data for each user ID (last name, first name, cost center, company code, etc.) and vice versa. This infotype can be present only if SAP ERP HCM is deployed and used. Because SAP ERP HCM is always more accurate and updated than user master data, you can link your user master data to SAP ERP HCM through Infotype 0105 to keep your user master records updated. There are two key benefits of this process:

- Keeps your user master record up to date for a large company
- Improves your user reporting and periodic user access review by linking technical information to HR and business-understandable information

And Here's How ...

When you (or the HR department) define a personnel number, you can also define the Infotype 0105 Subtype 0001 (see Figure 1). This infotype contains the link between the personnel number and the SAP user ID. By linking HR and user master records, you can exploit HR data to produce more business-oriented and understandable user ID reporting. You can also simplify the periodic user access review. To do this, select COMMUNICATION in the INFOTYPE field and click on the NEW button.

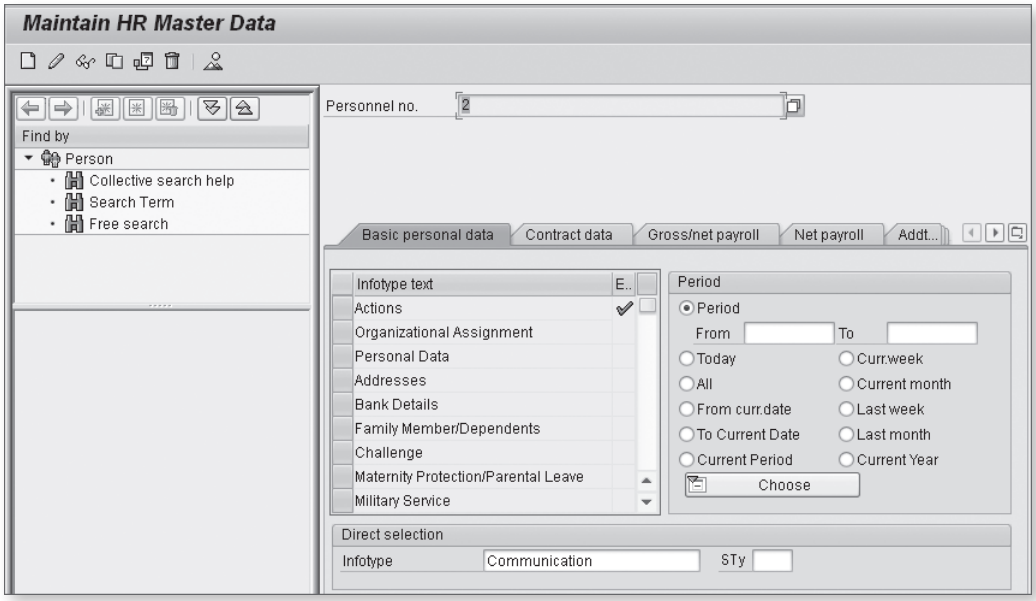


Figure 1 Defining Infotype 0105 on the Personnel Number

The resulting screen in Figure 2 shows the ID/NUMBER field, where you enter the SAP user ID to link to the personnel number. Pay careful attention here—the SAP system doesn't check whether the user exists in the system or if the name exceeds the user ID character limit of 12 characters.

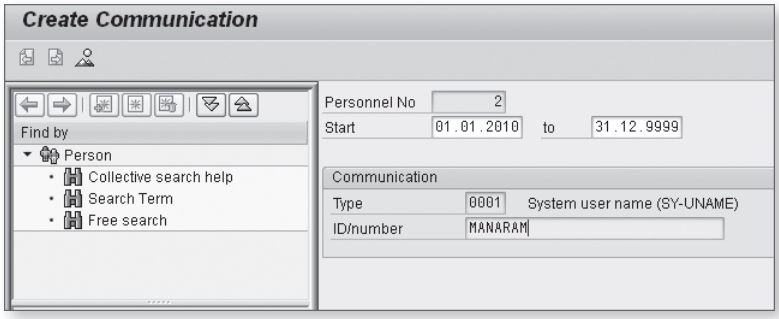


Figure 2 Communication Infotype 0105 Subtype 0001, SAP User ID

The result of the link between the user ID and infotype can also be used to produce the user's reporting, as shown in Figure 3. You can see for each company and plant the number of involved user IDs. For example, in company 8271, there are three

plants—7006, 7007, and 7005—with 77; 647; and 5,376 users defined. You can create this kind of report by exporting SAP user master data with HR information into a spreadsheet.

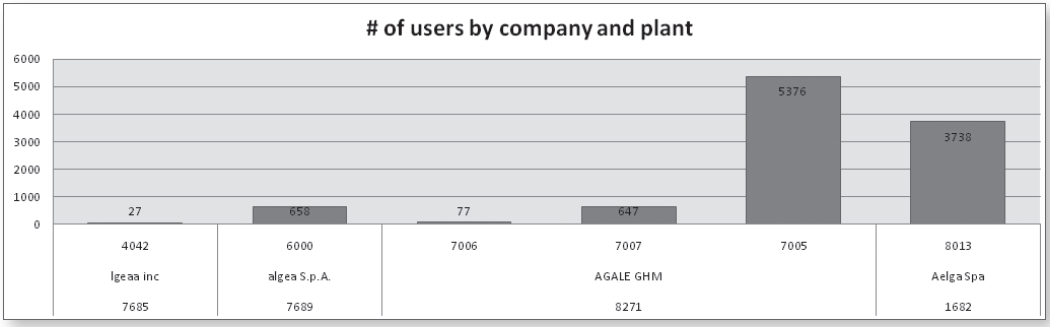


Figure 3 Example of User ID Reporting with HR Data (Users by Company Code and Plant)

For example, through the graph shown in Figure 3 you can evaluate how many users are defined for each company or plant. This is important from a SAP license point of view.

Tip 14

Performing Mass Changes for Users and Roles in Java

You can mass-maintain users and roles in a Java stack by uploading and downloading a text file.

If you're a user administrator, you know that especially for mass activities the Java web interface is not user friendly; for example, it's difficult to paste a list of previously copied data. You can work around this by uploading and downloading a text file and modifying it to perform mass activities in a Java context.

And Here's How ...

You can manage users and permission en masse through SAP User Management Engine (UME) in JAVA. You can log in to UME via the link *http://<yourserver>:<yourport>/useradmin*.

Figure 1 illustrates all of the steps you need to export the authorization data and user definition:

1. In the Search Criteria field, select User and then enter the user ID (❶).
2. Click on the Go button and then select it after the UME finds it (❷).
3. Click on the Export button (❸).

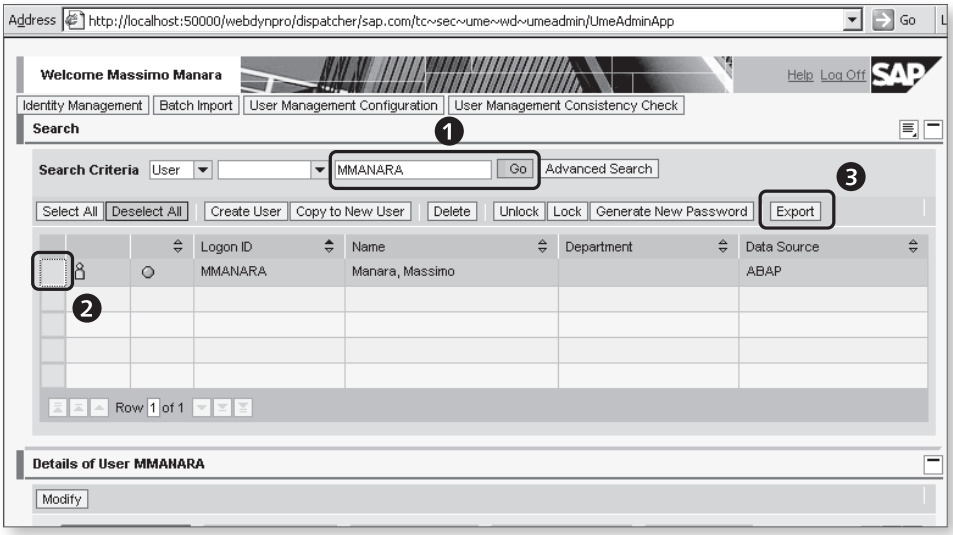


Figure 1 UME Steps to Export Authorization Data and User ID Definition

Figure 2 shows the result of these three steps. You can highlight the text in the Export box to save it in a text file. This text contains the user definition and the Java role assignment that you can upload into another system. In this manner, you can also manipulate this text file to perform some mass activities; for instance, you can add or remove a user who is assigned to a role.

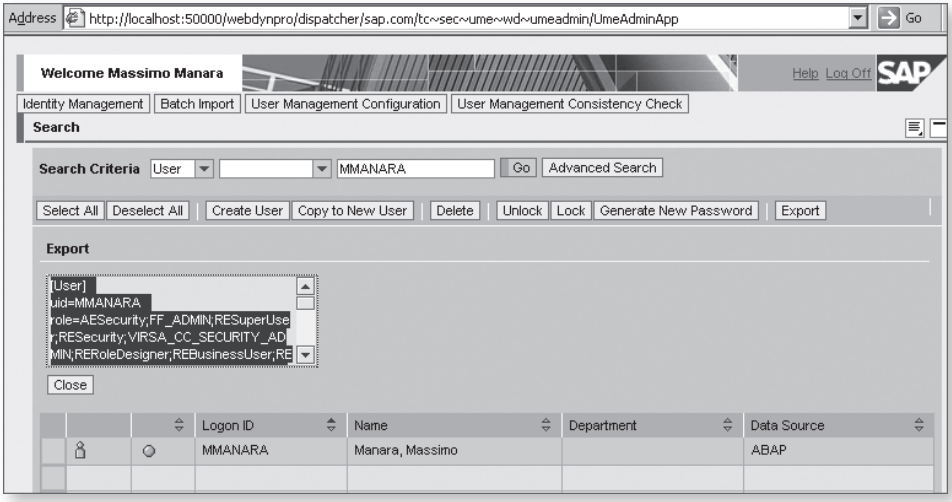


Figure 2 Export Field in the UME Interface

In the same way, you can import the text file that has been previously created:

1. Click on BATCH IMPORT (❶).
2. Click on BROWSE (❷) to retrieve the text file from the folder in which it is saved.
3. Click on UPLOAD (❸, as shown in Figure 3).

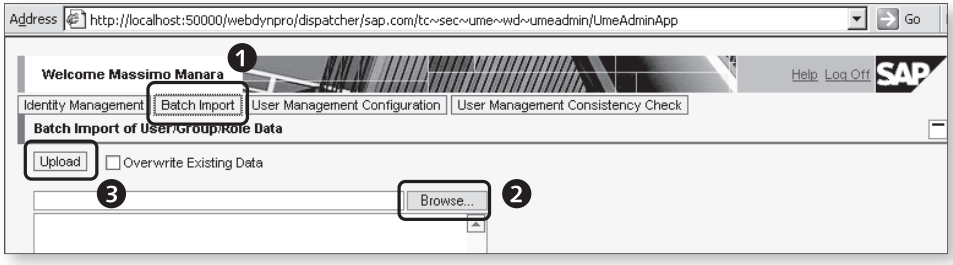


Figure 3 Import a Text File in the UME

Tip 15

Displaying Authorization Errors in Transaction Log SU53 for Different Users

If you're an administrator, you can use Transaction SU53 to display the last authorization error check for your user ID or for another SAP user ID.

You can use Transaction SU53 to display the last authorization error check. When a user receives an authorization error, this should send the Transaction SU53 output to the security administrator. In some cases, the security administrator will go directly to the user's PC to view the Transaction SU53 log if the SAP user cannot e-mail the Transaction SU53 log to the administrator. To save time, you can see the Transaction SU53 log of every user from your SAP login. We'll explain how in this tip.

And Here's How ...

Execute Transaction SU53 through this SAP Easy Access path:

System • Utilities • Display Authorization Check

After you've executed this transaction, you can see your last authorization error check. If there is no authorization check, the system shows you the message "The last authorization check was successful" (see Figure 1).

You can also see the client, user ID, date, and system; these data elements enable the administrator to know the user, system, and client for authorization troubleshooting without having to ask for specific information from the user.

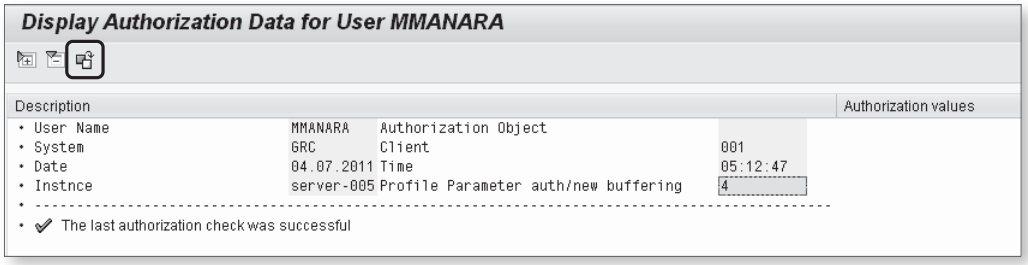


Figure 1 Transaction SU53 User Button Switch

The last data shown is the value of the instance profile: AUTH/NEW_BUFFERING. When this value is equal to 4, it means that the authorization user buffer is updated immediately after an authorization change; the user can avoid logging off and then back on. By clicking on the USER icon circled in Figure 2, you can switch the Transaction SU53 user log to see a different user.

If you enter another user ID, you can display the Transaction SU53 log for it (see Figure 2, ❶ and ❷).

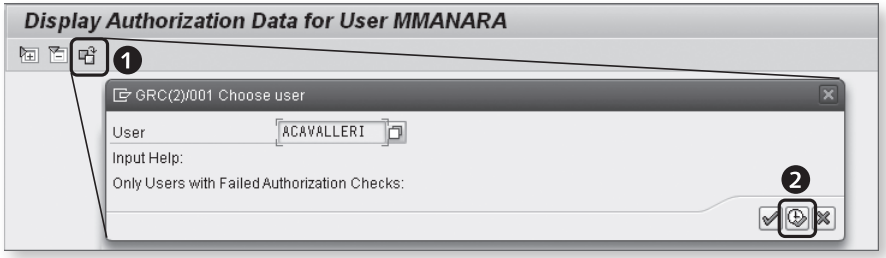


Figure 2 Display Transaction SU53 Log for a Specific User ID

You can also display the last authorization error check by browsing in SAP Table USR07.

Keep in mind that this functionality should be allowed only for the security administrator. You can allow this capability (display the Transaction SU53 log for a specific user ID) through authorization object `S_USER_GRP` with activity 03 (display). If a user has this authorization, that person has the ability to switch the Transaction SU53 user log and then be able to display all users' error logs, which could cause security holes and a possible privilege escalation.

Tip 16

Customizing Users' Selection en Masse

You can mass-maintain the user master record through Transaction SU10.

In your daily maintenance activities, you probably change several users' attributes with Transaction SU10. In the standard system however, this transaction code accepts only a small list of user IDs. If you find it necessary to tailor your users' selections based on user attributes—department, function, user groups, or an external list—you may be using a longer list of users than is accepted. Let's find out how to bypass this problem.

And Here's How ...

To access the mass user master record maintenance transaction, use the following menu path:

SAP Menu • Tools • Administration • User Maintenance • SU10

In the User column of the selection list, you can type in all users that you need to change. However, you can normally enter only around 30 users at a time.

To bypass this restriction, click on the Authorization data button, perform a user selection, and then paste a user list of previously copied users (Figure 1). To execute the selection, click on the Execute button or press F8.

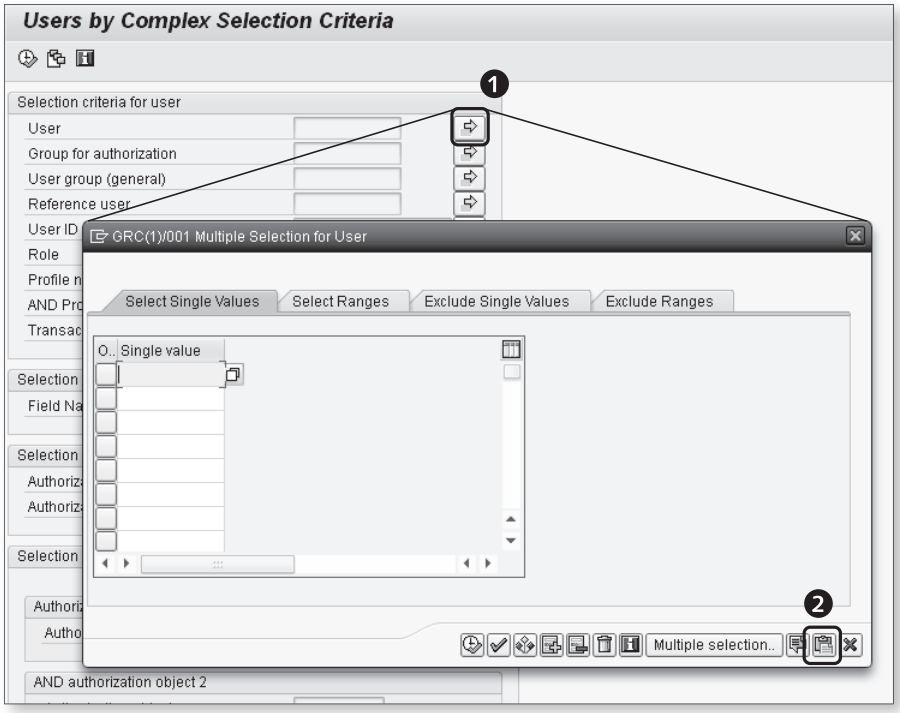


Figure 1 User Selection Criteria with Pasted List

The system selects all listed users (if defined in the system), and you can select all (❶ in Figure 2) and transfer the user list back to Transaction SU10 to process them by clicking on the TRANSFER button (❷).

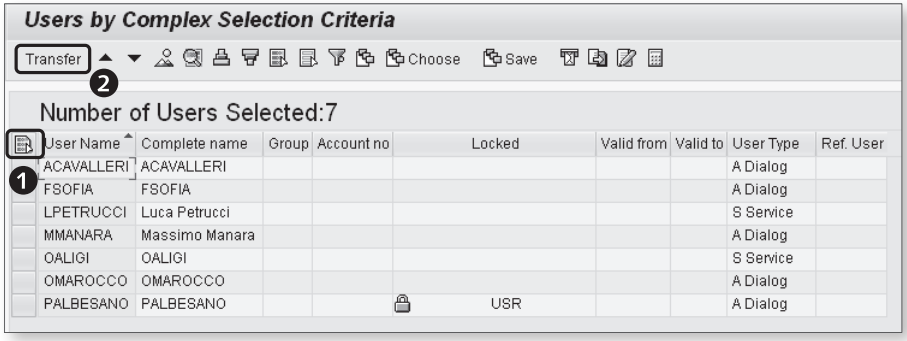


Figure 2 Select Users and Transfer to Transaction SU10

Tip 17

Mass-Changing Secure Network Communications Data for SSO User Mapping

You can mass-define your Secure Network Communications (SNC) name through Transaction SNC1.

There are different ways to set up Single Sign-On (SSO), depending on your requirements, such as whether you have SAP Enterprise Portal or not, whether you have only SAP GUI or Web GUI, whether you use an external product, and so on.

To help deploy and configure SSO, SAP released the Secure Network Communications (SNC) software layer, which provides an interface to an external security product. After you enable SNC, you have to populate the SNC tab in Transaction SU01 for all users. Unfortunately, Transaction SU10 (User Mass Update) doesn't allow you to update the SNC tab in Transaction SU01. Therefore, you must update and define SNC data through Transaction SNC1.

And Here's How ...

After you enable the SNC functionality in your system, a new tab named SNC appears in Transaction SU01 (see Figure 1).

In this tab, you have to define and enter the SNC name. The syntax of this name depends on your SNC solutions.

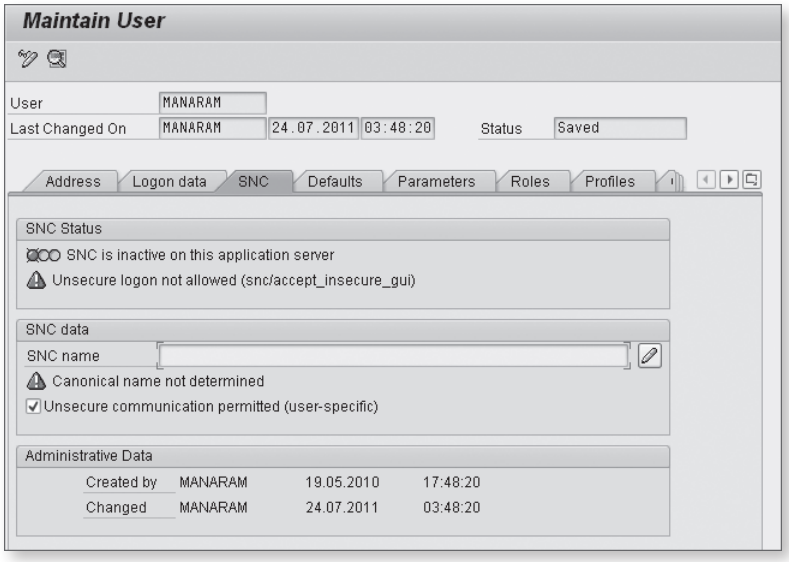


Figure 1 SNC Tab in Transaction SU01

You cannot mass-update the SNC field through Transaction SU10 because this transaction doesn't support SNC tab updating, so instead you need to use Transaction SNC1 to accomplish the task of mass-maintaining and defining this SNC field.

Using Transaction SNC1 (see Figure 2), you can set the SNC name for a set of users or for a group. Enter the user ID in the USERS field, and then populate the PREVIOUS CHARACTER STRING and FOLLOWING CHARACTER STRING with your company data specifics.

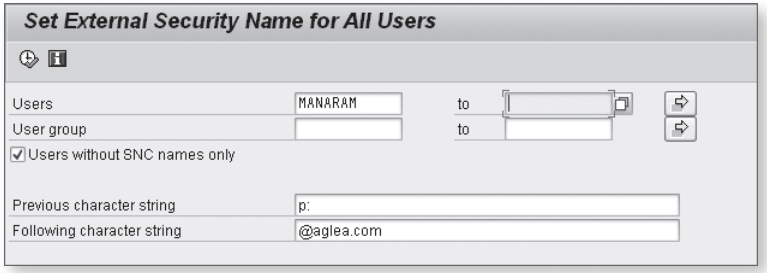


Figure 2 Transaction SNC

You can decide to populate the SNC name for only users without SNC names by checking the USERS WITHOUT SNC NAMES ONLY checkbox. You can also define how this field should be populated by entering the users and filling the PREVIOUS CHARACTER STRING and FOLLOWING CHARACTER STRING fields depending on your domain name; using the data in Figure 2, the result will be *p:<Username>@aglea.com* and then *p:MANARAM@aglea.com*.

After you've defined your SNC name, click on EXECUTE. You'll see a confirmation step before the system mass-updates the SNC name. When you see this, click on the SAVE icon.

The result in Transaction SU01 is shown in Figure 3 where the SNC NAME field is populated as defined in Transaction SNC1.

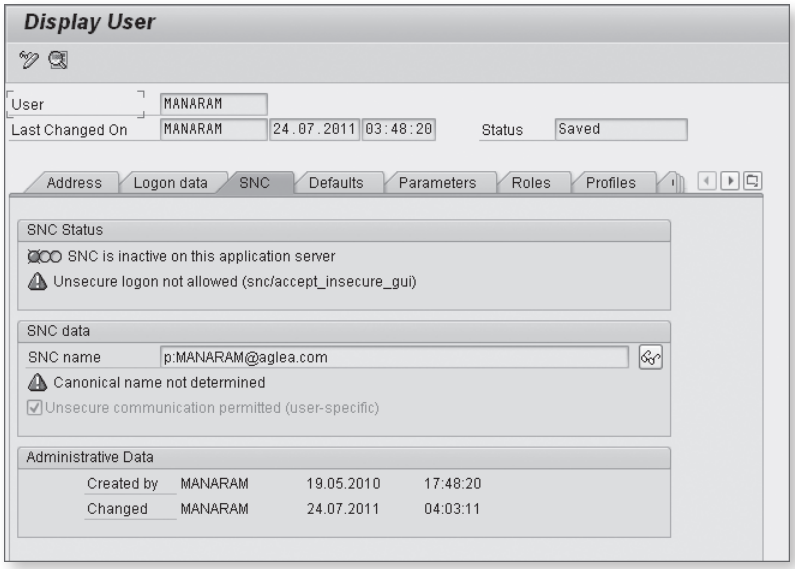


Figure 3 SNC Tab in Transaction SU01 Is Populated

Part 2

Development Security

Things You'll Learn in this Section

The important task of managing security options must be done during the development phase of a new capability or functionality of your security features. This part of the book will help you intercept some developments that are not in compliance

with your policies or remediate some nonconforming developments that are already in place. You'll also see how to solve common pain points and business problems via some useful ABAP Data Dictionary knowledge.

Tip 18

Validating Your ABAP Code before Moving into the Production System

You can perform an ABAP scan to avoid some common security ABAP holes that may be embedded in standard security checks.

You can use a standard tool to ensure that your ABAP code is safe against backdoors or other programming threats. A common gap in most security administrators' backgrounds is specific skills in the ABAP programming language. If you are a security administrator and not well-versed in ABAP security, you can use this tool to prevent someone from moving unsafe source code into a production landscape.

And Here's How ...

You can execute or display ABAP source code through Transaction SE38. You can enter a custom program in the Transaction SE38 program field by following this SAP Easy Access menu path:

PROGRAM • CHECK • CODE INSPECTOR

You can execute the source scan inspector on this ABAP program. The Code Inspector tool automatically performs several types of checks—performance, syntax, user interface, and so on—but our focus is on the security check. Figure 1 shows the security inspector result with the exploded SECURITY CHECKS tree after the Code Inspector has been run.

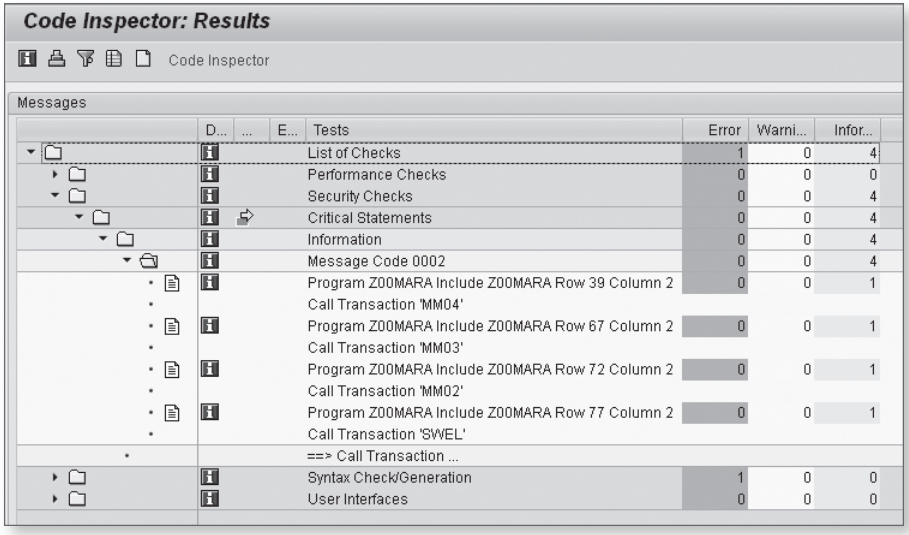


Figure 1 Code Inspector Result: Security Checks Tree

By analyzing the results of this tool, you can intercept some common and predefined by SAP security ABAP holes, or customize what the tool checks in order to enhance your ABAP code security inspections. To customize the checks, you can create your own variant. To do this, click on the New icon at the top left of Figure 1 and flag which type of check the tools perform. Relevant security checks are, for example, call to C program or transaction call. If you are unsure about a check, you can search the statement detail on SAP help.

Tip 19

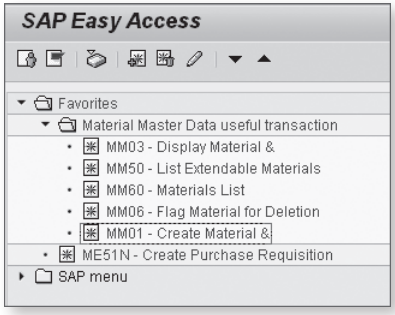
Archiving and Restoring a User's Favorites

You can easily restore a user's favorites with function modules if the user ID is deleted by mistake.

None of SAP's standard transactions will allow you to back up the entire list of a user's favorites. (In this case, "favorites" do not refer to a user's favorite weather or shopping site, but instead to favorites that are necessary to enhance the person's speed and productivity during work). You should be aware of two standard function modules that can be used to back up and restore the user's favorites in case a user ID is deleted by mistake.

And Here's How ...

An example of a user's list of favorites is shown in Figure 1 in the SAP Easy Access menu; each user can define his favorite folders and transactions.



« *Figure 1 SAP Easy Access User Favorites*

Through the SAP Easy Access menu, each user can download a text file and then upload it to back up his favorites via: Favorites • Download to PC and then Favorites • Upload to PC.

However, procedures aren't as clear cut in the following situations:

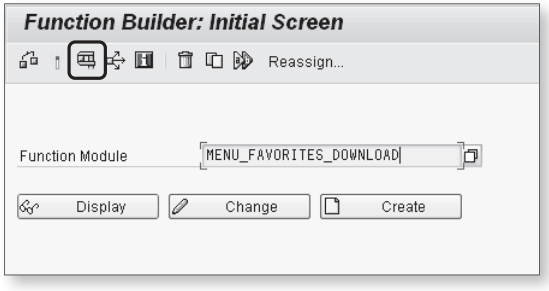
- A common company policy is that if a user doesn't use the SAP system for a certain period of time, this user should be deleted.
- An administrator mistakenly deletes some users.

In both of these cases, all favorites are lost when you delete a user. You can use two standard function modules to back up all users' favorites so that if you delete a set of users, you will able to restore the user's favorites without causing them undue frustration.

The name and the behavior of these two function module are similar:

- `MENU_FAVORITES_DOWNLOAD`
- `MENU_FAVORITES_UPLOAD`

You can execute these function modules via Transaction SE37 as shown in Figure 2 by clicking on the circled icon named Test/Execute.



« *Figure 2 Execute Function Module*

After you've executed the function module, enter the user ID for whom you want to download or upload favorites, and then click on Execute, as shown in Figure 3.

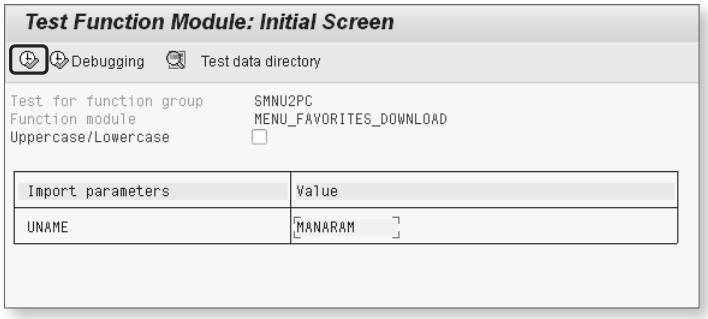


Figure 3 Enter User ID

After you've executed the function, the SAP system will ask you where you want to save the text file on your file system. In the same way as the upload function module, you have to enter the user ID for whom you want to upload the favorites, and browse in your file system to get the uploaded file.

With the support of a developer, you can enhance these function modules to add a mass functionality.

Tip 20

Displaying the Security Data Dictionary Definition with the Object Navigator

You can use the SAP Object Navigator to gather essential knowledge during security and authorization analysis and troubleshooting.

Several types of authorization-related objects are defined in the ABAP Data Dictionary in the SAP system. Due to the huge amount of data, it isn't easy to find all of the information you need at a glance. However, there are several Basis component transactions that can help you during the authorization analysis phase and troubleshooting, if you know where to find them. To achieve these ends, you can use the Object Navigator to help you investigate and answer new authorization business needs and segregation requests.

And Here's How ...

You can use Transaction SE80 (Object Navigator) to find out where an authorization object is used. This could be useful when you have to analyze and investigate a specific new segregation request. Figure 1 shows how to explore the object:

1. Click on Repository Information System (❶), and then explode the Objects tree.
2. Click on Authorization objects (❷), enter the authorization object (❸, "M_MATE_STA" in our example) in the Standard Selections area, and then press F8 to execute.

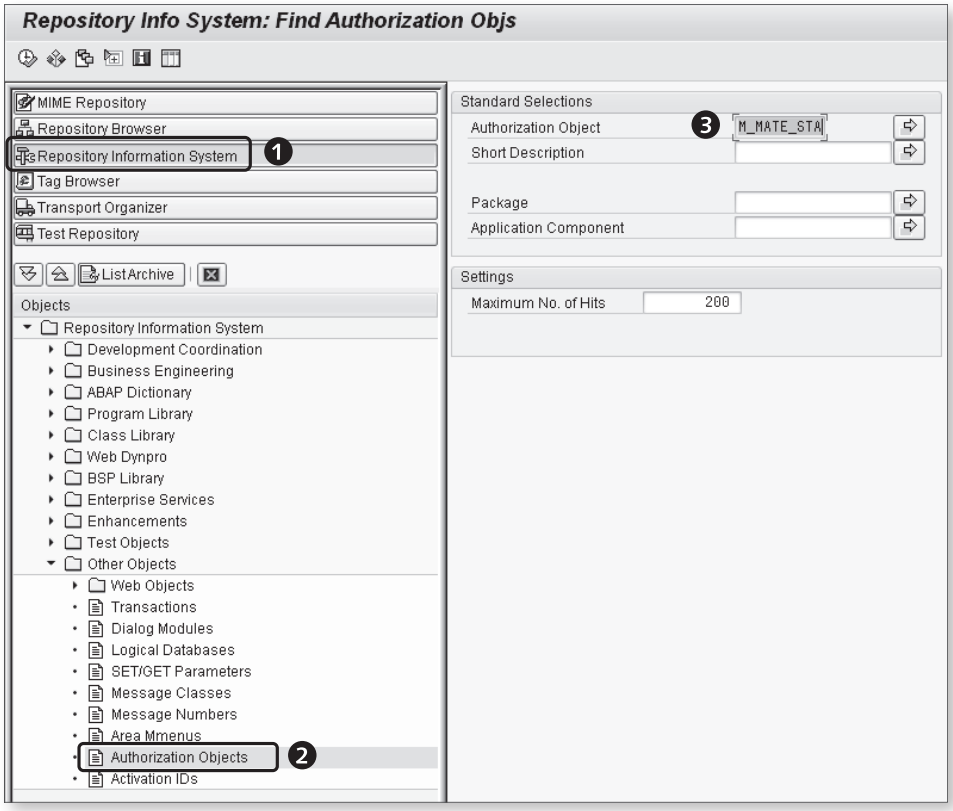


Figure 1 Authorization Objects Search through Repository Information System

3. Click on the circled icon (WHERE USED LIST) shown in the resulting screen in Figure 2 to explore where the repository object is used in some types of Data Dictionary objects (PROGRAMS, CLASSES, etc.).

The result of selection is shown in Figure 3. You can see where these authorization objects are used in the SAP system.

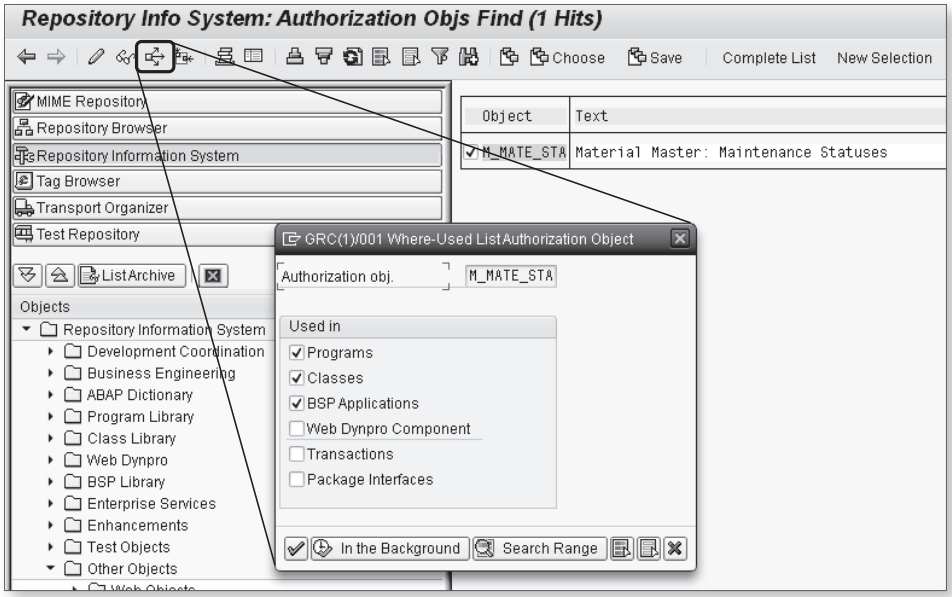


Figure 2 Where Used Authorization Object

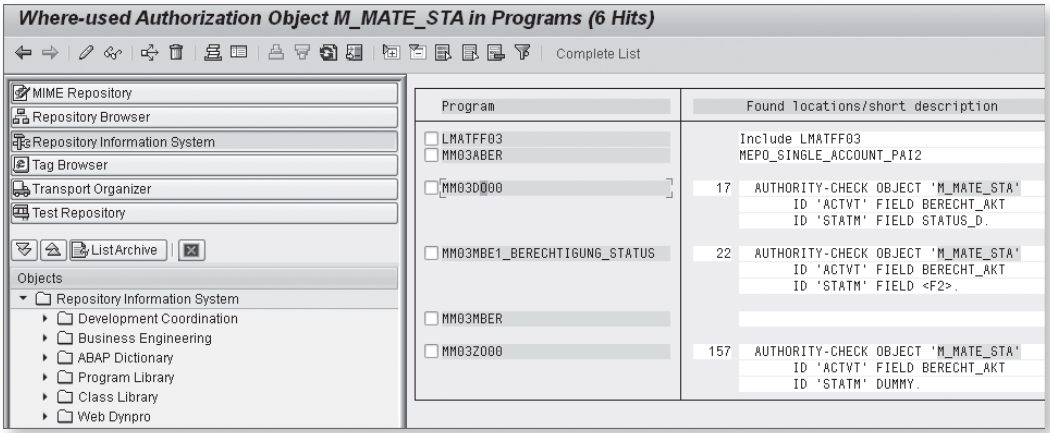


Figure 3 Result of Where Used Authorization Object M_MATE_STA

This kind of analysis could be helpful if you're investigating a custom development. Suppose you find an unknown or undocumented custom authorization object in your system, and you want know in which SAP program it is used (if it's not used, you can delete this authorization object to clean up your system). Through this analysis, you can quickly find out where and how the object is used.

Tip 21

Finding Vulnerability Strings in Your ABAP Code

You can find potential critical weaknesses in your ABAP source code that could cause code backdoor or security holes by using a standard source scanner.

When you know there may be critical weaknesses in your ABAP statements (e.g., system calls or call transactions), you need to scan the source code to intercept these statements. Weaknesses in your statements can result in a malicious developer being able to enter a backdoor into an ABAP program to bypass the standard security check. You can perform some quick research to find these statements by using standard SAP programs.

And Here's How ...

For the purpose of this tip, let's suppose you have to find the ABAP statement `AUTHORITY-CHECK` in one or more ABAP programs. (For this tip, you must already know the statement for which you're searching.) There are different transactions that you can use to perform a string search in your ABAP source code:

- `RSABAPSC` or Transaction S_ALR_87101287 (see Figure 1)
- `RS_ABAP_SOURCE_SCAN`

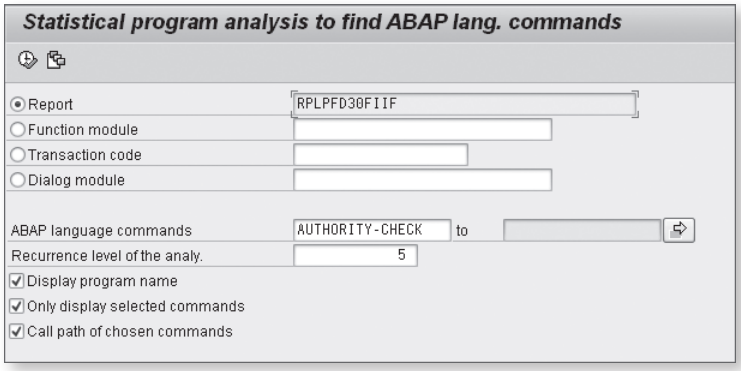


Figure 1 Find the AUTHORITY-CHECK Command in Standard Report RPLPFD30FIIF

Enter the program name in the REPORT field and the ABAP statement you want to search for in the ABAP LANGUAGE COMMANDS field. Then execute the program by pressing F8, and you can see where this statement appears. The result of execution is shown in Figure 2; you have found the statement on the authorization object S_GUI.

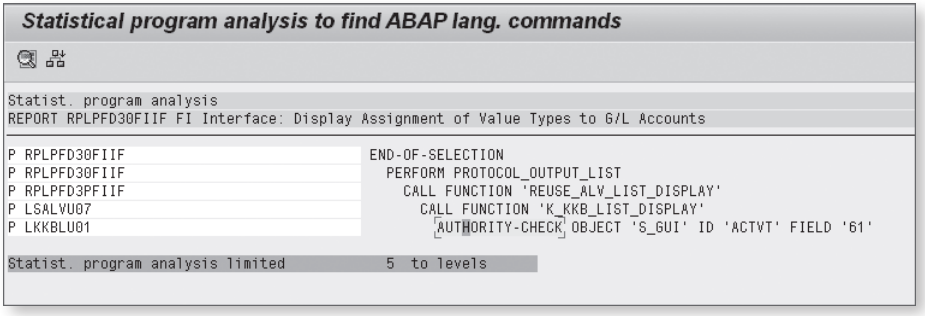


Figure 2 Transaction S_ALR_87101287 Authority Check Search Result

Unfortunately, Transaction S_ALR_87101287 doesn't allow a mass report search; you can search only in one report or function module, one transaction code at a time. However, you can set the recurrence level of analysis with this transaction. Several SAP programs recall other routines; by setting this level, you can search the string in all recursive programs.

Through Transaction SE38 you can execute Report RS_ABAP_SOURCE_SCAN, which allows you to search a string in more than one report. Figure 3 shows the selections criteria. In this case, we're looking for the statement CALL TRANSACTION (in the STRING SEARCHED FOR field) in Program RM06ELLB (in the PROGRAM NAME field). Figure 4 shows the result of this selection.

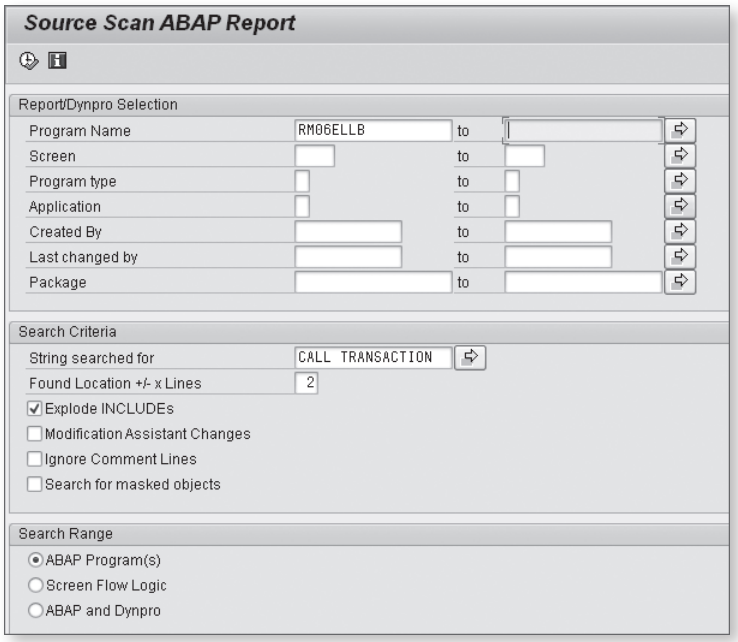
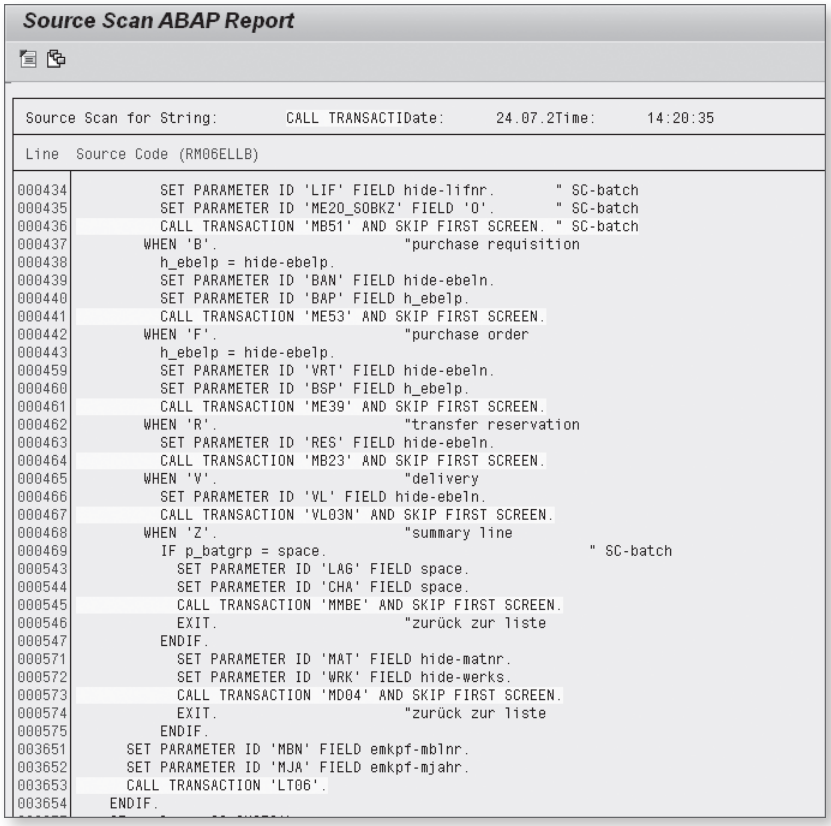


Figure 3 RS_ABAP_SOURCE_SCAN on Program Name RM06ELLB with String Search CALL TRANSACTION

After you execute this report with these selection criteria, you can see where this statement is used in the report (Figure 4). Then, in this case, you can evaluate whether the called transactions are allowed based on your internal policy and on the analysis phase requirements.



Source Scan ABAP Report

Source Scan for String: CALL TRANSACTIDate: 24.07.2Time: 14:20:35

Line	Source Code (RM06ELLB)
000434	SET PARAMETER ID 'LIF' FIELD hide-lifnr. " SC-batch
000435	SET PARAMETER ID 'ME2O_SOBKZ' FIELD 'O'. " SC-batch
000436	CALL TRANSACTION 'MB51' AND SKIP FIRST SCREEN. " SC-batch
000437	WHEN 'B'. "purchase requisition
000438	h_ebelp = hide-ebelp.
000439	SET PARAMETER ID 'BAN' FIELD hide-ebeln.
000440	SET PARAMETER ID 'BAP' FIELD h_ebelp.
000441	CALL TRANSACTION 'ME53' AND SKIP FIRST SCREEN.
000442	WHEN 'F'. "purchase order
000443	h_ebelp = hide-ebelp.
000459	SET PARAMETER ID 'VRT' FIELD hide-ebeln.
000460	SET PARAMETER ID 'BSP' FIELD h_ebelp.
000461	CALL TRANSACTION 'ME39' AND SKIP FIRST SCREEN.
000462	WHEN 'R'. "transfer reservation
000463	SET PARAMETER ID 'RES' FIELD hide-ebeln.
000464	CALL TRANSACTION 'MB23' AND SKIP FIRST SCREEN.
000465	WHEN 'V'. "delivery
000466	SET PARAMETER ID 'VL' FIELD hide-ebeln.
000467	CALL TRANSACTION 'VL03N' AND SKIP FIRST SCREEN.
000468	WHEN 'Z'. "summary line
000469	IF p_batgrp = space. " SC-batch
000543	SET PARAMETER ID 'LAG' FIELD space.
000544	SET PARAMETER ID 'CHA' FIELD space.
000545	CALL TRANSACTION 'MMBE' AND SKIP FIRST SCREEN.
000546	EXIT. "zurück zur liste
000547	ENDIF.
000571	SET PARAMETER ID 'MAT' FIELD hide-matnr.
000572	SET PARAMETER ID 'WRK' FIELD hide-werks.
000573	CALL TRANSACTION 'MD04' AND SKIP FIRST SCREEN.
000574	EXIT. "zurück zur liste
000575	ENDIF.
003651	SET PARAMETER ID 'MBN' FIELD emkpf-mblnr.
003652	SET PARAMETER ID 'MJA' FIELD emkpf-mjahr.
003653	CALL TRANSACTION 'LT06'.
003654	ENDIF.

Figure 4 Where Used CALL TRANSACTION Statement in Report RM06ELLB

Tip 22

Creating a Transaction Variant to Restrict User Activities

You can limit a user's activities though transaction variants when it isn't possible to use authorizations.

Sometimes you need to hide some buttons or set a static value in a specific field of a transaction screen but find that this isn't supported by authorizations. In such cases, you can create a new transaction by redesigning the screen without writing ABAP code. This solution in easy to implement and won't generate effort in future upgrades.

And Here's How ...

There are two main steps to create a transaction variant:

- Create a new screen layout using Transaction SHD0 (Transaction and Screen Variants).
- Create a new custom transaction code using Transaction SE93, which will use the original ABAP code but with the new layout.

Suppose you want to lock the value in the PAYT TERMS field in the header data of Transaction ME22 (Change Purchase Order). This activity is not supported using the standard authorization concept.

Start Transaction SHD0, and specify the original TRANSACTION CODE (ME22) and a name for the new layer (Z_0001) in the TRANSACTION VARIANT field. Click on the CREATE icon to use the transaction.

Specify a purchase order number and then load the header data by clicking on the hat button.

Every time you go to the following screen of the transaction, a pop-up will appear asking you to customize the screen entries. Because you just have to specify the purchase number in the first screen, it isn't necessary to customize it. Deactivate the COPY SETTINGS flag and confirm the pop-up with the green check icon to go to the next screen.

As shown in Figure 1, the PAYT TERM field is active in the purchase order header.

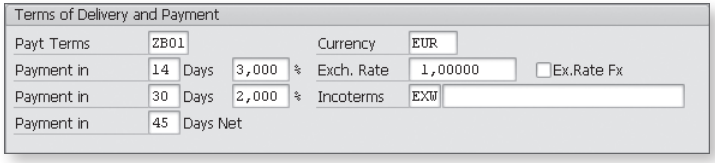


Figure 1 The Payt Terms Field Is Active

In the pop-up screen, maintain the fields, set the PAYT TERMS field as OUTPUT ONLY, and set the static value as ZB01 (see Figure 2).

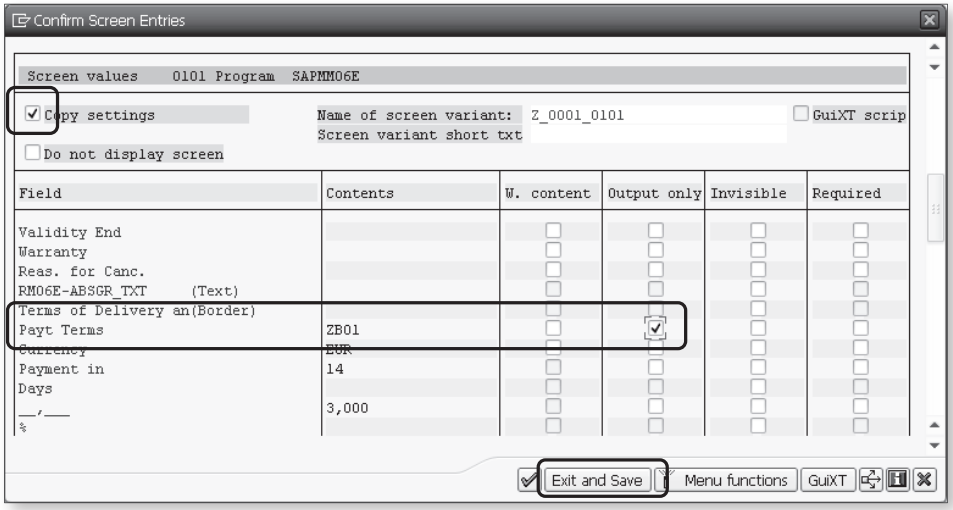


Figure 2 Customize Screen Entries

The final step is to specify a description for the transaction variant; enter a short text as description and verify that all fields are active except the PAYT TERMS field.

Now access Transaction SE93 to create a customized transaction code. In the Transaction Code field, enter Z_ME22 for this example, click on the Create button, enter a short text description, and choose the Start Object type.

After creating the new transaction, you have to choose the Transaction with variant (variant transaction) radio button.

Now Transaction SE93 can be finished by specifying the Transaction Variant name. To find the name of the transaction variant, the Cross-client flag must be activated. Link the new Transaction Z_ME22 with the transaction variant Z_0001 via standard Transaction ME22.

To check that the Payt Terms field is not modifiable, start Transaction Z_ME22 in the command field, and open a purchase order to see the final result (as shown in Figure 3).

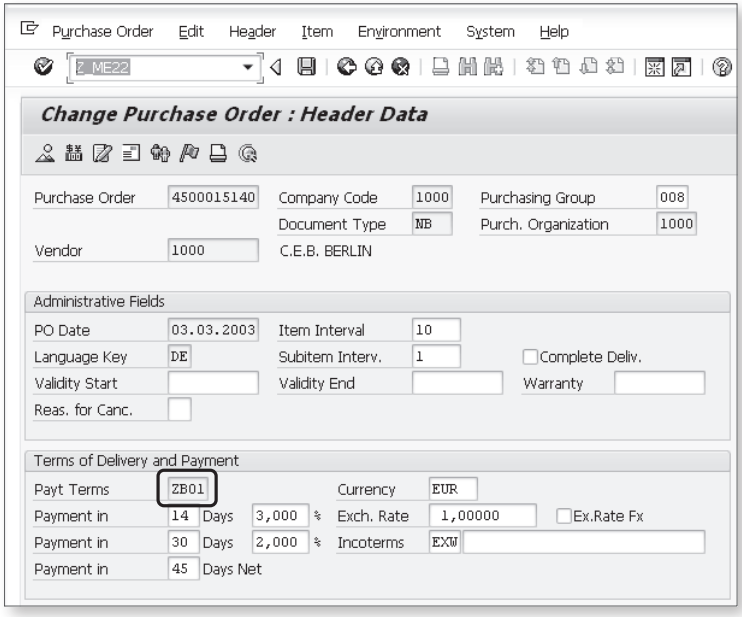


Figure 3 The Payt Terms Field Is No Longer Modifiable

Tip 23

Finding Authorization Object Documentation

You can easily find authorization object documentation by knowing where to navigate within Transaction PFCG.

When you decide to insert a standard authorization object in your own code or when you want know what an authorization is used for, it's important to look at the SAP standard documentation to discover any prerequisites, pain points, or tips. For example, a standard authorization object could require a previous customizing step to work properly, which you may be unaware of if you've never used the object before. You can easily find all of the documentation for authorization objects with Transaction PFCG.

And Here's How ...

To see all active authorization objects classified by authorization class, use Transaction AUTH_DISPLAY_OBJECTS or access Transaction PFCG by following this path:

Environment • Authorization Objects • Display

Roughly 2,500 authorization objects are defined in SAP ERP. To facilitate your search, SAP has grouped these authorization objects in authorization classes; for example, FI for Financial, SD for Sales and Distribution, and so on.

Figure 1 shows the exploded BC_Z class with the AUDIT_AUTH authorization object documentation box. Here you can read the SAP help to discover if, as in the AUDIT_AUTH object, it's necessary to perform a customizing step to use this authorization object.

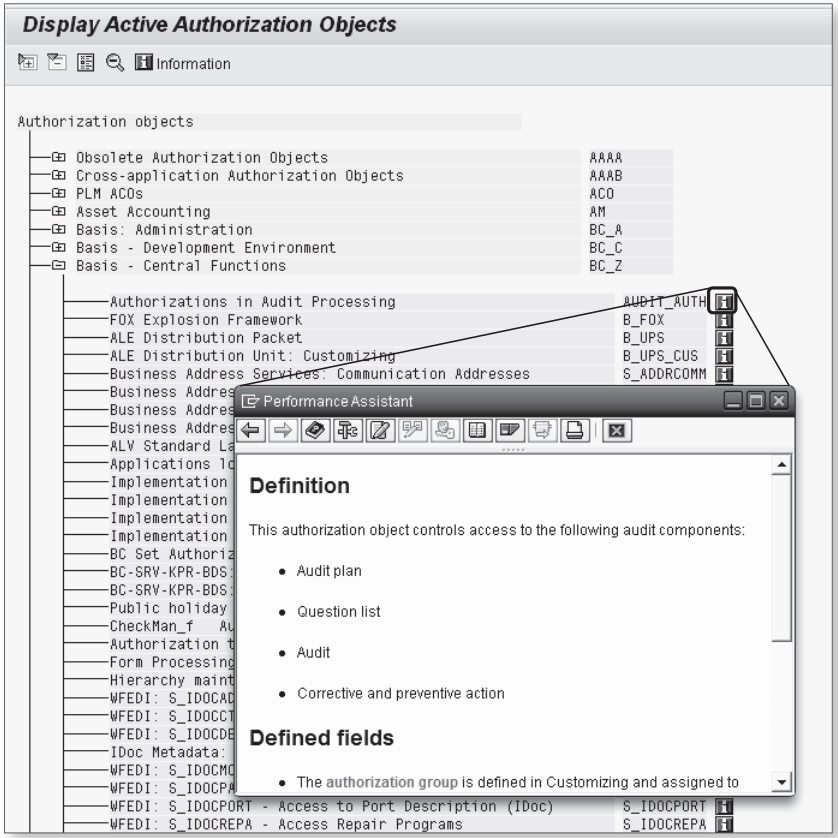


Figure 1 Display Active Authorization Objects: Documentation Check Box

An alternative method of viewing the authorization object documentation is by using the Transaction PFCG authorization tree, which you can access when you are in the AUTHORIZATION tab of Transaction PFCG. By double-clicking on an authorization object, the system shows the pop-up authorization objects documentation, as shown in Figure 2.

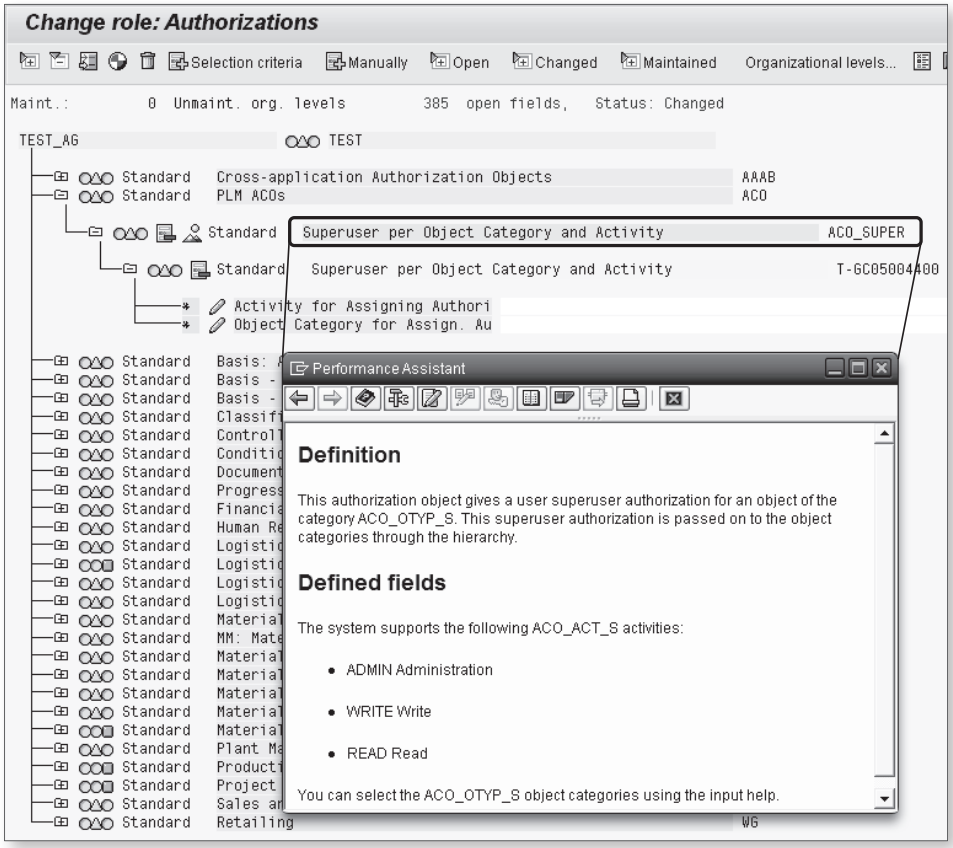


Figure 2 Authorization Object Documentation in Transaction PFCG Authorization Tree

Tip 24

Searching for Values and Definitions in ABAP Data Dictionary Tables

By browsing specific ABAP Data Dictionary tables or using a table scan, you can discover in which tables an SAP field or a specific value is used.

During the authorization design and analysis phase, it's sometimes essential to understand whether a certain field has an authorization object to protect it. If you know the table name, or at least one table field name, you can search in the ABAP Data Dictionary to gather some useful information—for example, all tables or views where this field is used—to understand and give a clear light to your authorization analysis. You may also need to search for a specific value across more than one table; for example, if you have deleted a user by mistake and you want to know if the user has stored some information.

And Here's How ...

Browse for a Data Dictionary Definition in a Table

Browsing Data Dictionary tables is useful when you're looking only for the Data Dictionary definition of the table.

Via Transaction SE16, you can browse Table DD03L and/or Table DD02L. Note that all SAP tables that start with *DD* refer to dictionary tables. The first, Table DD03L, contains the following link: SAP Table – Field table. If you know the table name, you can see all fields in this table. If you know a field name, you can discover how many tables are used and where. The second table, Table DD02L, contains the list of all SAP tables.

Search for Values Across Several Tables

Use Transaction TABLE_SCANNER (see Figure 1) to perform a more specific search on Data Dictionary tables. In this screen, we are searching for a field named BNAME for the tables that start with USR0 that also contain the value MMANARA.

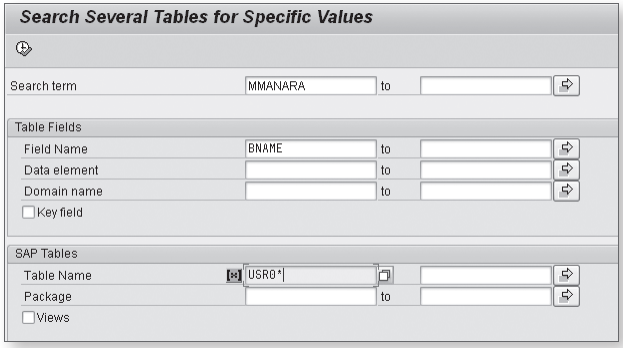


Figure 1 Search Data Dictionary Tables for Specific Requirements

Execute this transaction by pressing F8. You can see the results of your search in Figure 2. For each table in the range specified, the value of the field BNAME is defined in the search criteria. With this transaction, you can see the real content of the tables involved.

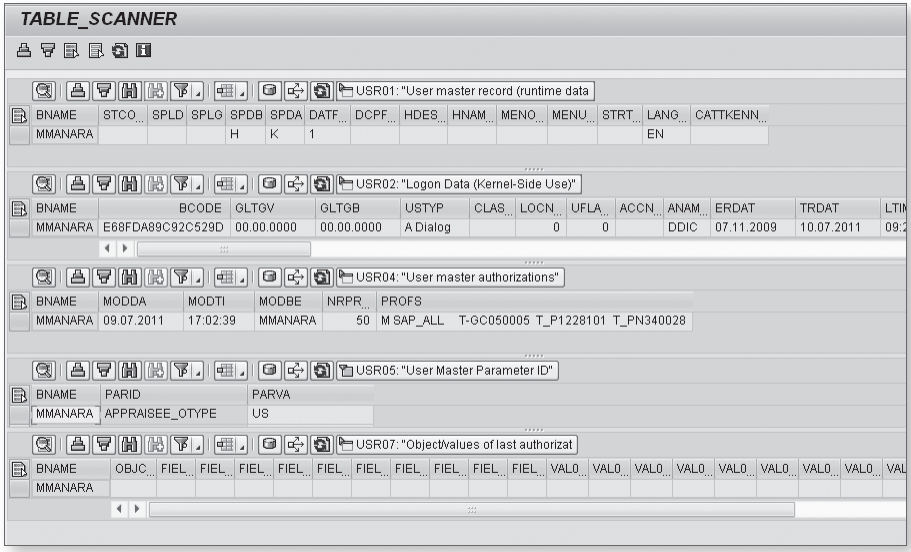


Figure 2 Transaction TABLE_SCANNER Result

Tip 25

Mass-Exporting Query User Group Information

You can export SAP query user groups to make sure that people are correctly assigned through a standard function module.

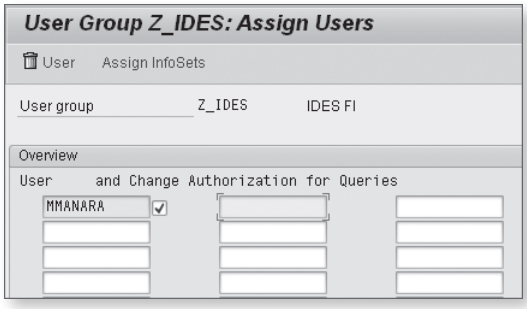
SAP user group transactions don't provide a functionality to export all user group links. However, when you have several users and groups, it's important to document these relationships, which can be difficult if you're trying to do this manually. Because SAP doesn't provide a standard report to view all users and query groups at a glance, you can export all data and then document the accuracy of your user and query group assignment through a spreadsheet. All of this can be done by using a standard function module.

And Here's How ...

There are two types of query areas depending on how you've implemented your query concept: cross client and client specific. Transaction SQ03 allows you to assign a user to a query authorization group in cross-client or client-specific areas. To set the query area, follow this path:

Environment • Query Areas

If you enter a user group (e.g., Z_IDES) when you are in Transaction SQ and then click on the Assign users and InfoSets button, you can assign users to this group (in this case, in a cross-client area, as shown in Figure 1). The user assignments in a client-specific area work in the same way.



« *Figure 1 Transcription SQ03: Assign User to Query User Group*

You can mass-extract the users and query user group links in the following ways, depending on the type of query area used:

- **Cross-client through Table AQGDBBN**
  Here, you can enter a query user group and find all users assigned, or you can enter a user and find in how many query groups it is inserted.
- **Client-specific through function module RSAQ_IMPORT_USERGROUP_CATALOG**
  By executing it through Transaction SE37, you can see all groups and users defined in the query client-specific area.

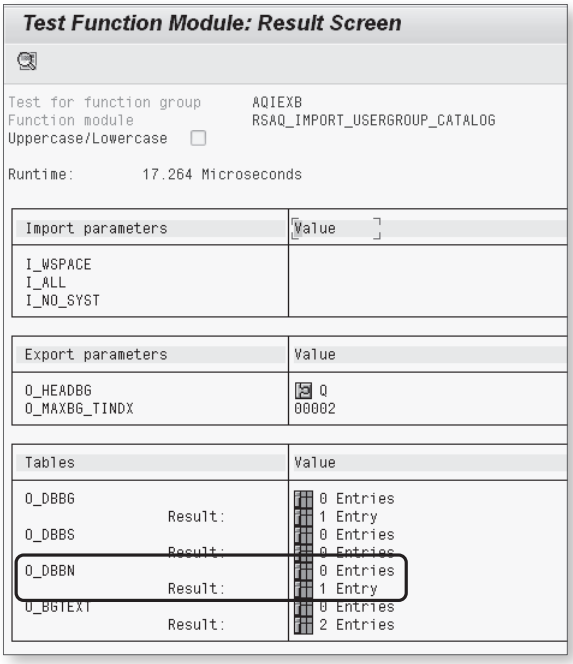
Now you need to execute the function module, enter the function module name, and then click on the TEST/EXECUTE button (or press F8).

After you've executed the function module, select the tables that are of interest for your purpose. In this example, this is Table O_DBBN, where you can find the users and query groups link as shown in Figure 2.

Make sure to pay attention to the table; the result could show only a part of the entire list. To show all data, click on OBJECTS • DISPLAY ENTIRE LIST.

As general security advice, do not allow end users to execute the query due to the following two main disadvantages:

- You have to maintain query users groups with Transaction SQ03.
- It can become difficult to control the table access authorization. If you create a specific transaction assigned at the query, you can consequently assign the correct table authorizations groups. By allowing a query through Transaction SQ*, you cannot easily establish at first which tables are involved in the query.



« *Figure 2 Display the Users and Query Groups Link in a Client-Specific Area*

To control the table view authorizations, you can create a custom transaction that is linked to a query. There are several ways to achieve this. One way is to access Transaction SQ00 and follow the menu path Query • More Functions • Display Report Name. Then you can copy the report name and access Transaction SE93 to create your custom transaction linked to this report name.

This method may appear time consuming (you must create a custom transaction for each query), but it's the best way to ensure governance. Keep in mind that there are also specific SAP functionalities that manage specific query analysis authorization, such as business intelligence tools.

Tip 26

Managing an Authorization Check in the Transaction Header

You can define an authorization check to start during the beginning of a transaction through Transaction SE93.

In some cases, you might need to enter an authorization check in a transaction code, but you don't want develop or change any of the ABAP code due to the cost and time requirements. This is possible by entering an authorization check in the header of a transaction through Transaction SE93.

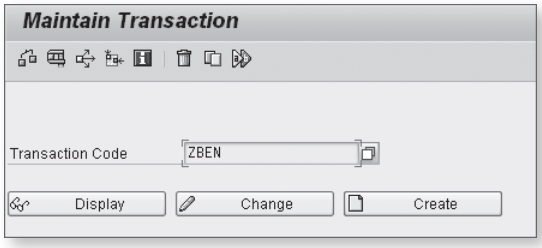
And Here's How ...

Here are the standard SAP authorization check flow steps:

1. Check if the user is authorized at the `S_TCODE` object (in this example, with value ZBEN).
2. Check if the transaction has an authorization object in the header (in this example, `M_MATE_STA` is directly linked; this is the main purpose of this tip).
3. Check all other authorization checks entered into the ABAP source code of Transaction ZBEN.

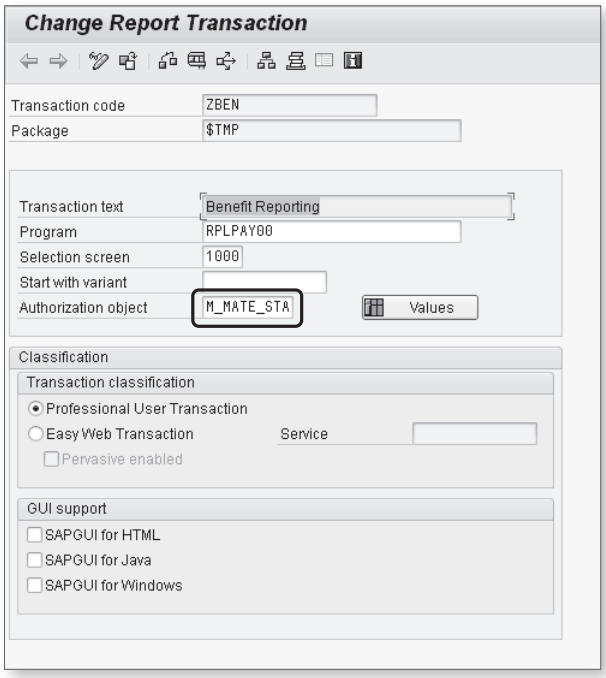
Let's walk through an example, following these steps:

1. Execute Transaction SE93 by entering the transaction code (in our example, ZBEN) for which you want to add an authorization check during the transaction start (step 2 of the SAP authorization check flow).
2. Click on the CHANGE button shown in Figure 1.



« *Figure 1 Execute Transaction SE93 for Transaction ZBEN*

3. After you're in change mode (see Figure 2), enter the name of the authorization object that will be checked during the start of Transaction ZBEN in the AUTHORIZATION OBJECT field (in our example, the object is named `M_MATE_STA`).



« *Figure 2 Transaction SE93 in Change Mode on Transaction ZBEN*

By clicking on the VALUES button in Figure 2, you can also set the object's values.

In this way, you have linked an authorization object at the transaction level without developing any ABAP code. Note that this approach only allows you to enter one authorization object per transaction.

Tip 27

Restricting a User's Access to Called Transactions

You can segregate and manage called transactions through Transaction SE97 to restrict specific users' access.

Via Transaction SE97, you can control whether the SAP system will block or allow a user's called transaction. You might want to do this if you discover that a user is able to reach a transaction via indirect methods, even if he is not authorized to do so. Every time a user starts a transaction code from the SAP GUI menu, the kernel will check the transaction code against the authorization object `S_TCODE`.

And Here's How ...

Imagine you have started Transaction ME22N (Change Purchase Order) from the SAP GUI menu, and you are working on purchase order 3004000250. As shown in Figure 1, you can see material number DPC-CPU-2600.

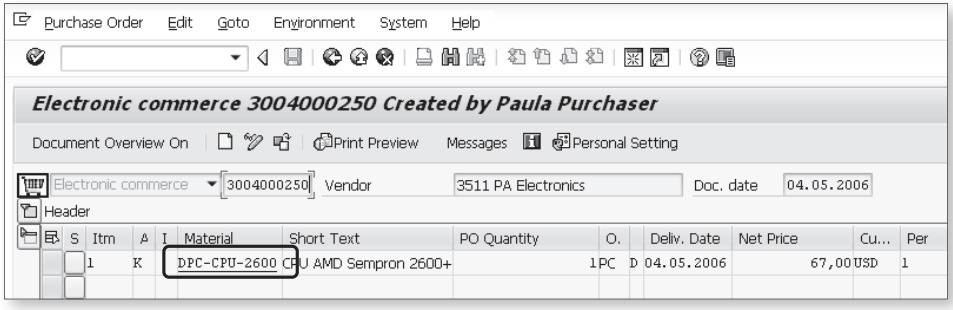


Figure 1 Transaction ME22N (Change Purchase Order)

If you double-click on the field of the material number, you will jump to Transaction MM03 (Display Material), as shown in Figure 2.

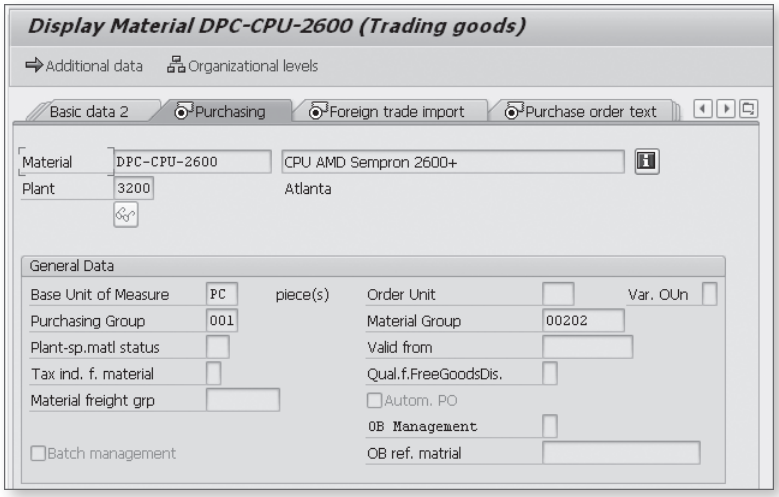


Figure 2 Transaction MM03 (Material Display)

If you look at the ABAP source code behind this action, you'll discover that Transaction MM03 is called indirectly from Transaction ME22N through the ABAP statement `CALL TRANSACTION`.[1] Therefore, you'd want to lock or manage Transaction MM03.

Going forward in the ABAP code, you'll then discover that Table TCDCOUPLES is responsible for the explicit authorization on the called transaction. You can see (and modify) the content of Table TCDCOUPLES using Transaction SE97.

First, specify the calling transaction (in the example, this is ME22N). Figure 3 shows all the possible transactions related to ME22N.

You must also understand the possible combinations of column CHECK IND:

- If the value is set to YES, an authorization check is performed when the ABAP statement `CALL TRANSACTION` is run.
- If the value is set to NO, no authorization check is performed.

1 In the help manual of the ABAP language, there is a specific note for the `CALL TRANSACTION` statement that will help you understand the logic.

- If the value is set to an empty space, one of the preceding check indicators is yet to be set. In the current release, no authorization check is performed. However, this may change in future releases.

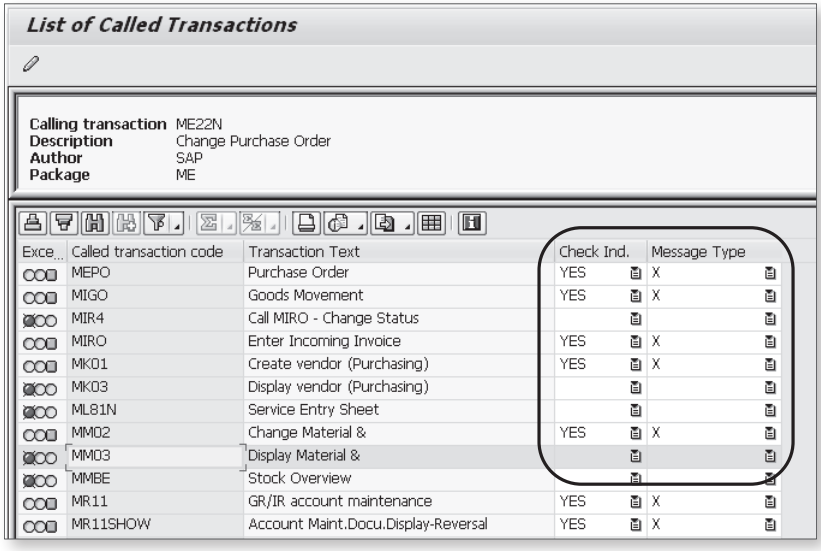


Figure 3 Table TCDCOUPLES Content

Note that in this example MM03 is set to blank. If the values are set to YES or NO, the description is clear. If the value is set to blank, the behavior of the system is related to the further setting of the system parameter `auth/check/calltransaction`.

Using the ABAP Report RSPARAM, as shown in Figure 4, you can see the values.

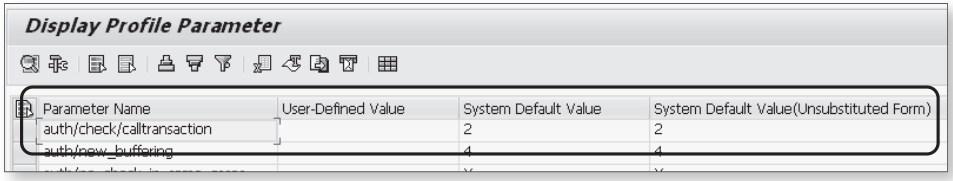
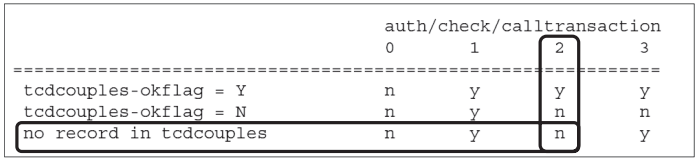


Figure 4 auth/check/calltransaction Setting

The meaning of value 2 is provided in OSS Note 515130 and SAP Note 358122. Figure 5 shows part of OSS Note 515130. In our SAP system (ECC6), if the value in Table TCDCOUPLES is not defined (bank), the authority-check for the called transaction will not be performed because the value of `auth/check/calltransaction` is set to value 2, and there is an "n" in the third row (no record in TCDCOUPLES).



	auth/check/calltransaction			
	0	1	2	3
tcdcouples-okflag = Y	n	y	y	y
tcdcouples-okflag = N	n	y	n	n
no record in tcdcouples	n	y	n	y

« *Figure 5 auth/check/calltransaction Values*

Using Transaction SE97, you can adjust the values of TCDCOUPLES records to meet your security goals.

Tip 28

Managing Customizing Tables in a Production System

Generally, you can't customize tables directly within the production system; however, in some instances you'll need to do this.

SAP best practice advises that each change to the SAP system be performed in the development system, tested in a quality system, and at the end, moved into a productive system. This is to ensure that the production client doesn't need to be (and can't be) modified. However, in some cases, you may have to maintain tables directly; for example, the exchange rates maintenance or the financial period-end closing maintenance in the production system. Let's discuss how to accomplish that in this tip.

And Here's How ...

You can use Transaction SOBJ to change the current status of customizing tables. Changing current status is useful when you need to maintain a customizing table directly in the production system. Some common examples are entering exchange rates or opening and closing posting periods.

You can check if your system is closed via Transaction SCC4 (see Figure 1). Select your client, and then click on the magnifying glass icon.

Note that if the client role is set as PRODUCTION and no changes are allowed for client-specific/cross-client objects, in this client (generally the production client is set thus), you cannot directly change customizing tables (the status of these settings is shown in Figure 2).

Figure 1 Transaction SCC4

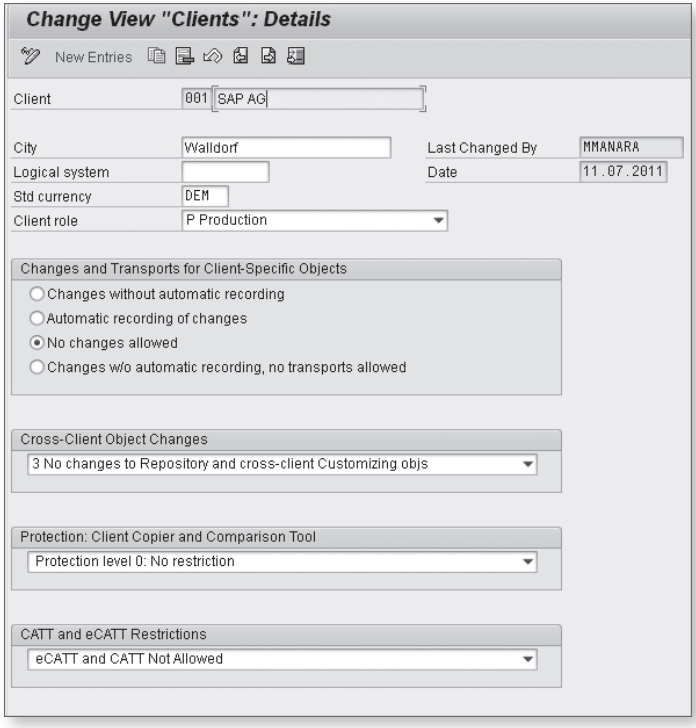


Figure 2 Transaction SCC4 Client Status Detail

Execute Transaction SOBJ to set a customizing table that is customizable directly in the production client. Click on the Display button to see the actual configuration, then click on Maintain to manage the table maintenance.

In Transaction SOBJ in the development system, you can select the table and submit a change request to allow you to leave the table as customizable despite the client status. Figure 3 shows the table of currency.

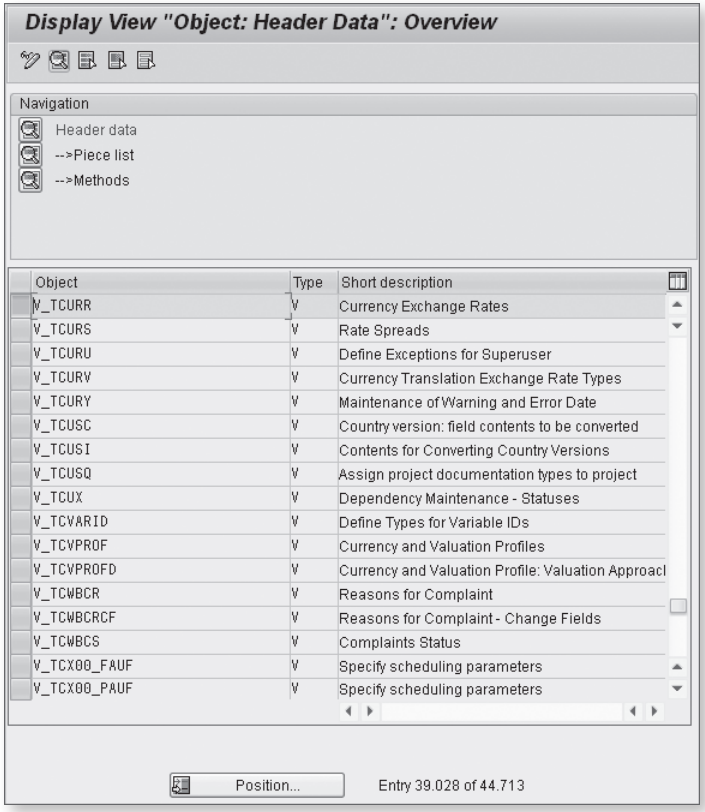


Figure 3 Transaction SOBJ on the Currency Exchange Rates Table

Select the table, and click on the magnifying glass icon to set the CURRENT SETTINGS flag. Note that when you change SAP objects, you need to record this change on the OSS SAP site.

If this flag is set, you can maintain this table directly, even if the client doesn't allow changes at client-specific/cross-client objects. After you perform this activity in the development system, at the end of this customization the system will ask you to define a change request to move into the quality and production systems.

Tip 29

Analyzing Your Security System to Keep it Updated

You can make sure that your system security level is up to date using the RSECNOTE tool.

SAP releases several updates and security patches every year. Understanding how your system is positioned with the most current and relevant security notes and patches is not a simple task. However, you can use a standard SAP tool to analyze your system and discover if you need to update it.

And Here's How ...

The tool you need to check your security system is Report RSECNOTE, which you access via Transaction ST13. Once executed, your system will give you a list of all SAP OSS security notes that you can apply in the system (see Figure 1).

The most recent and important updates are flagged with a red traffic light, and notes are flagged with a yellow traffic light. By clicking on the status traffic light, you can apply the note. Usually this activity is performed by the Basis administrator team.

From a business point of view, the main pain point about using this SAP feature is that after the patching of security notes, all programs and transactions work as before. Performing a test for each patch is extremely difficult, especially when there are several thousand patches to apply.

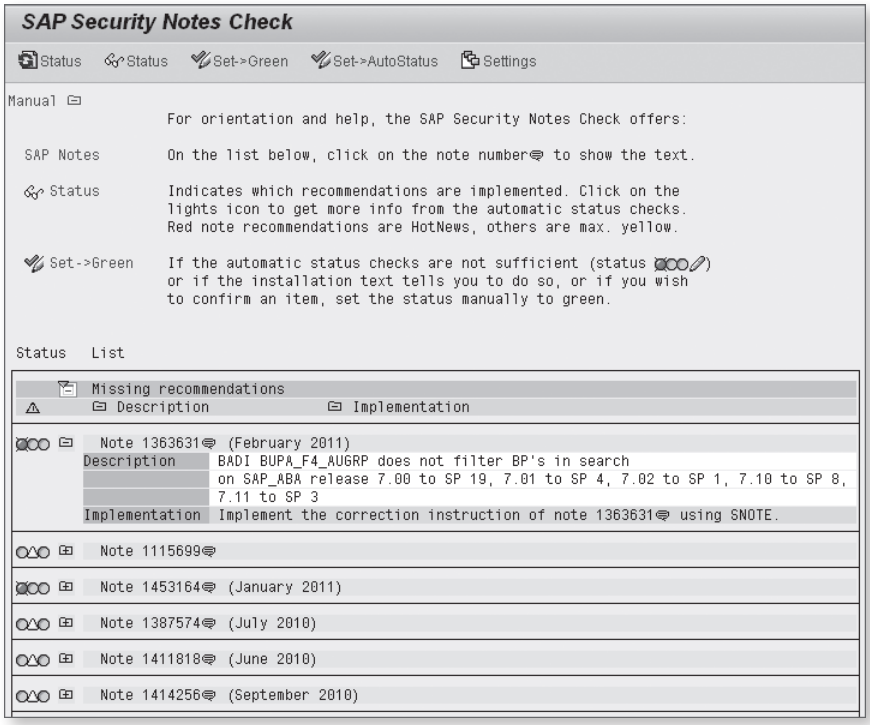


Figure 1 SAP Security Notes Check Result

Security patching is not a threat to business continuity, as it doesn't cause a lock. If there's an activity that you can't perform after the patching, that means the activity had security holes. As an example, we refer to the possibility through Transaction SE16N with the `&sap_edit` command (see SAP Note 1420281) to directly change the data into a table. After a security patch, this action can't be performed anymore due to limiting the misuse of this feature.

Tip 30

Using Parameter Transactions to Avoid Giving Direct Tables/ Programs Access to End Users

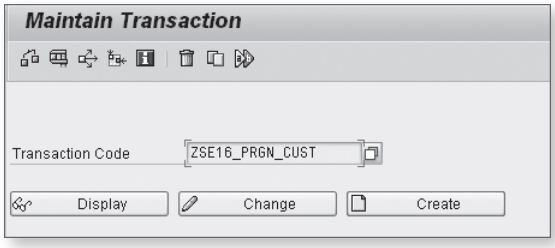
You can avoid releasing Transactions SE38 or SE16 at the end-user level by using parameter transactions instead.

End users sometimes find it necessary to maintain a table or execute a program. The standard transactions for these actions are considered critical from a security standpoint because they allow the user to potentially execute or maintain several tables or programs. To avoid using Transactions SE38 or SE16, you can define a parameter transaction that exploits the standard transaction but allows the end user to edit or execute a predefined table or program.

And Here's How ...

Through Transaction SE93, you can define transaction codes in SAP. This transaction also allows you to define a type of transaction, called a parameter transaction, based on a standard transaction. In this example, you can define a parameter transaction based on Transaction SE16, thus avoiding releasing this last critical transaction (SE16).

You first access Transaction SE93 to create a parameter transaction code. Enter a new transaction code and click on the CREATE button, as shown in Figure 1.



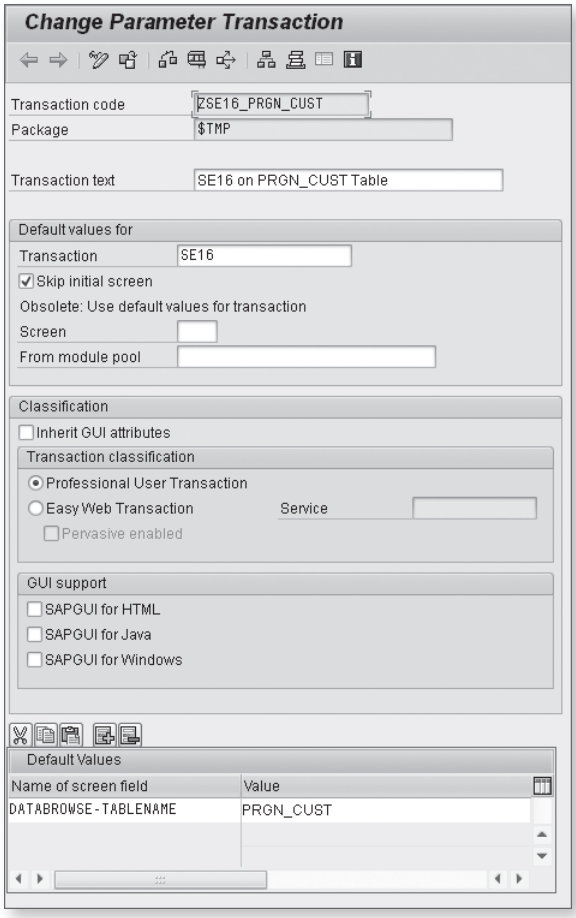
« *Figure 1 Define a New Transaction*

After you click on the CREATE button, the system asks what type of object you would like to use to start the new transaction (choose the START OBJECT). For this tip, select TRANSACTION WITH PARAMETERS (PARAMETER TRANSACTION).

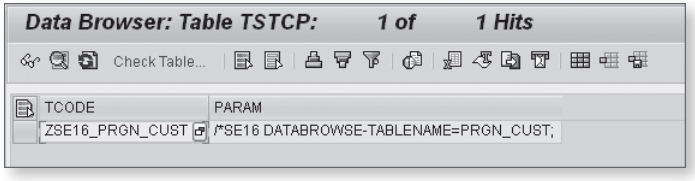
Parameter transactions allow you to pre-assign values to the fields on the initial screen. If you supply all of the necessary entries for the initial screen in this way, you can suppress the screen when the transaction is executed by checking the SKIP INITIAL SCREEN option shown in Figure 2. When Transaction ZSE16_PRGN_CUST is defined and started, the system uses the standard critical transaction (e.g., SM30 or SA38) but skips the initial screen in Figure 2. At the top of the screen, enter the table to be started with Transaction ZSE16_PRGN_CUST (e.g., in the NAME OF SCREEN FIELD column in the DEFAULT VALUES section, enter "DATABROWSE-TABLENAME" and "PRGN_CUST" in the VALUE field).

You can use this method to release Transactions SE16, SE38, and SM30, or another similar transaction at the end-user level.

Behind these parameter transactions, the standard SAP Table TSTCP shows the linked table or program (Figure 3).



« *Figure 2 Create Parameter Transaction Based on Transaction SE16*



« *Figure 3 Table TSTCP, Parameter for Transaction*

Imagine having a system with several parameter transactions defined but undocumented. By browsing Table TSTCP you can find the program or tables related to these parameter transactions and simplify the documentation process.

Tip 31

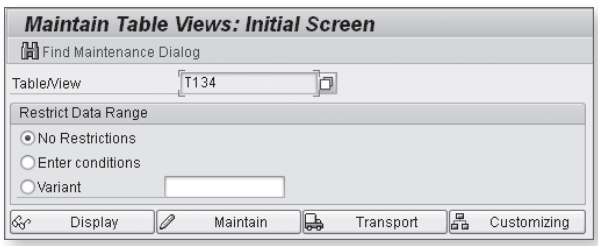
Discovering Maintenance Customizing Transactions with a Table Name

You can use Transaction SM30 to discover whether a table can be maintained with a customizing switch or a transaction.

To segregate financial documentation or the material master data type, for instance, you have to set up some authorization object constraints. Through these two authorization objects, you cannot directly specify the document type to protect; instead, you have to classify your document type by using a group concept. To classify the document type, you have to customize some tables. But if you only know the tables, how you can find out the transaction you need to maintain it?

And Here's How ...

Execute Transaction SM30, enter the table name for which you want know if a transaction exists to maintain the data in this table (in this example, Table T134, Material Master Type), and then click on the CUSTOMIZING button as shown in Figure 1.



« *Figure 1 Transaction SM30 on Table T134*

You will now see a pop-up screen where you must specify what customizing projects are used. If you don't have a customizing project, click on the CONTINUE W/O SPECIFYING PROJECT button (see Figure 2). Otherwise, maintain the PROJECT field.

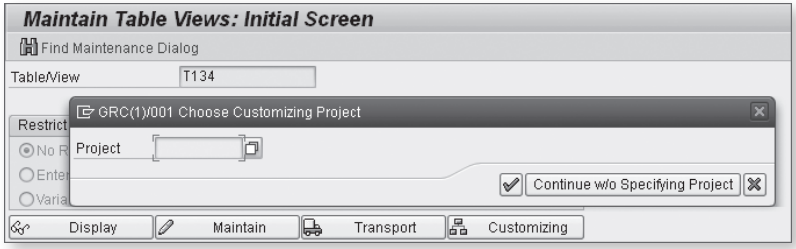


Figure 2 Transaction SM30 Chose Customizing Project Pop-Up

Now you can see the customizing tree (Transaction SPRO), where you can maintain this table. By clicking on these results, you can access the customizing tree. After this table (Table T134) is defined for the MATERIAL MASTER data type, select the material master row DEFINE ATTRIBUTES OF MATERIAL TYPES (see Figure 3).

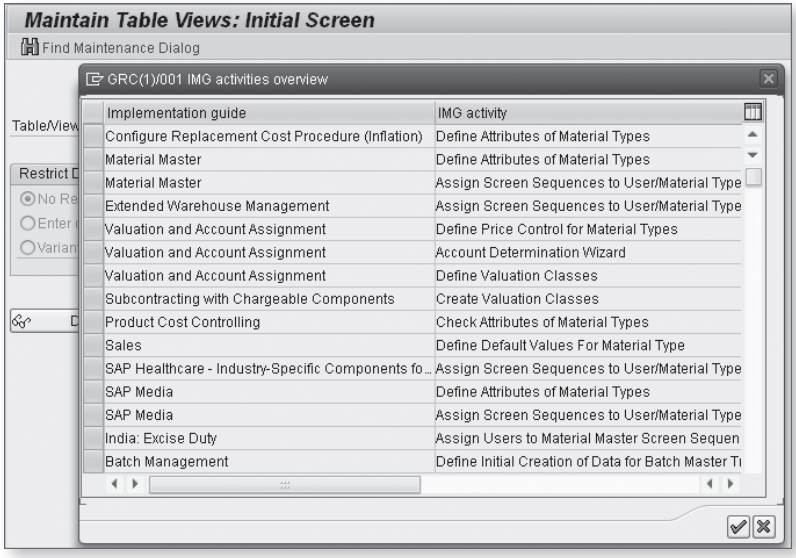


Figure 3 IMG Activities to Maintain Table T134

Figure 4 shows the customizing tree with the transaction name circled after you have activated the IMG technical names by following this menu path:

> Additional Information • Display Key • IMG Activity

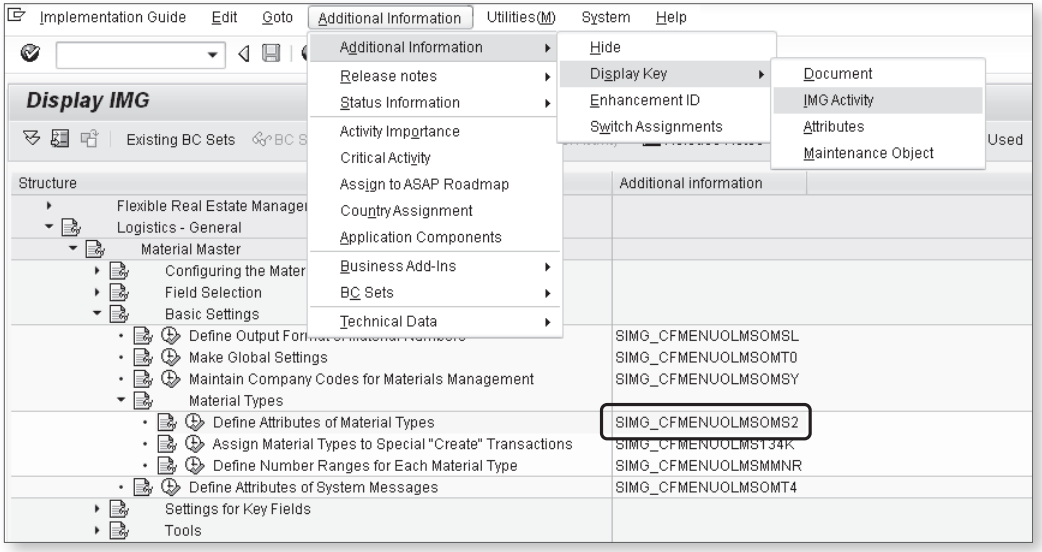


Figure 4 IMG Activity with Transaction OMS2

Now that you know the SAP table, you can find the transaction code to maintain the table.

Part 3

Profile Generator

Things You'll Learn in this Section

In this part of the book you'll find a set of tips that are mainly related to the authorization tool used by SAP to manage the authorization—the profile generator, Transaction PFCG.

We'll show you how to save a lot of time during the daily maintenance of your authorization concept by using this tool. You'll also see some common cases where SAP best practice compliance authorization maintenance methods should be used (but often aren't), and you'll see how to restore procedures for optimal maintenance and governance.

Tip 32

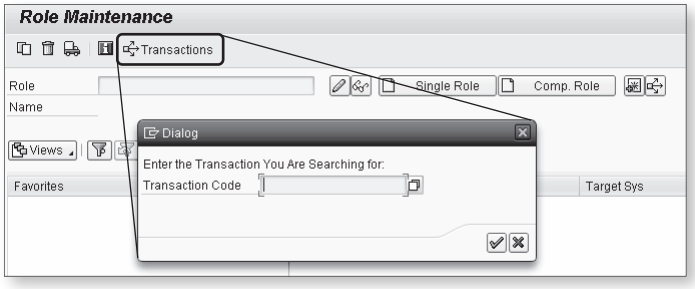
Finding Roles That Contain Transactions at the Menu Level

You can quickly and easily find which roles contain a transaction, and how many roles contain that same transaction.

Knowing which transaction codes are associated with which role is an essential authorizations task; otherwise, you're sure to have trouble during authorization maintenance. Through Transaction PFCG, you can perform a classical and commonly performed search—discovering in which roles a transaction code is inserted at the menu level. Although there are several ways to find the answer, there are a few unknown and quick ways to accomplish this—we'll show you the easiest, which is also the least known.

And Here's How ...

First, go to Transaction PFCG and click on the TRANSACTIONS button (see Figure 1) to find where a transaction code is inserted. This will display a pop-up dialog where you need to enter the transaction code you're searching for.



« *Figure 1*
Transactions Button in Transaction PFCG

In this tip we'll search for Transaction MM03. Click on the checkmark button, and you can see all of the roles containing that specific transaction within the role menu (see Figure 2).

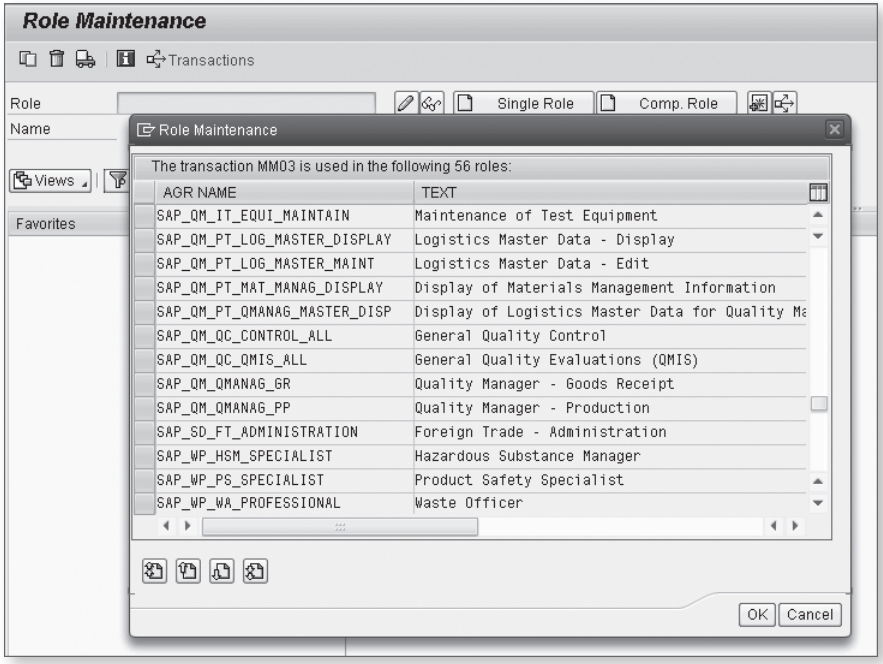


Figure 2 Transaction MM03 Role Maintenance

This kind of search must be performed at the Transaction PFCG menu level and not through authorization object `S_TCODE`. You can perform this kind of search only if you are sure that your roles are not misaligned between the transaction code that is inserted in the menu and the transaction code inserted in authorization object `S_TCODE` (see Tip 40).

Tip 33

Permanently Enable the Technical Name View in Transaction PFCG's Authorization Tree

You can improve and simplify your profile generator (Transaction PFCG) authorization usage and understanding by enabling the technical name view.

By default, the AUTHORIZATIONS tab isn't active in Transaction PFCG and therefore doesn't display the technical names of all authorization elements (authorization, authorization objects, authorization object fields, and authorization class). This can cause difficulty when reading all authorization elements and when performing a quick troubleshoot during authorization maintenance.

You can improve your Transaction PFCG usage and save time during authorization troubleshooting by permanently enabling the technical authorization objects.

And Here's How ...

In the AUTHORIZATIONS tab of Transaction PFCG, click on the pencil icon to see the authorizations tree, as shown in Figure 1.

By default, you can't see all authorization elements' technical names in the authorization tree.

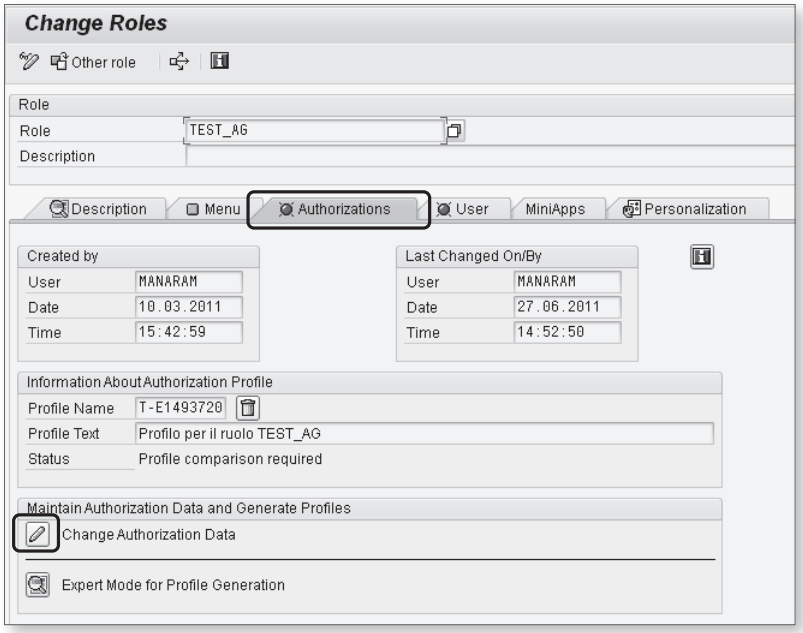


Figure 1 Transaction PFCG Authorization Tab

The only way to distinguish all authorization elements is by color:

- **Pink:** Authorization class
- **Green:** Authorization object
- **Yellow:** Authorization on an authorization object
- **White:** Values entered in an authorization object fields

However, you can't see the technical names of these authorization elements. To view the technical names, select Utilities • Technical names on (see Figure 2).

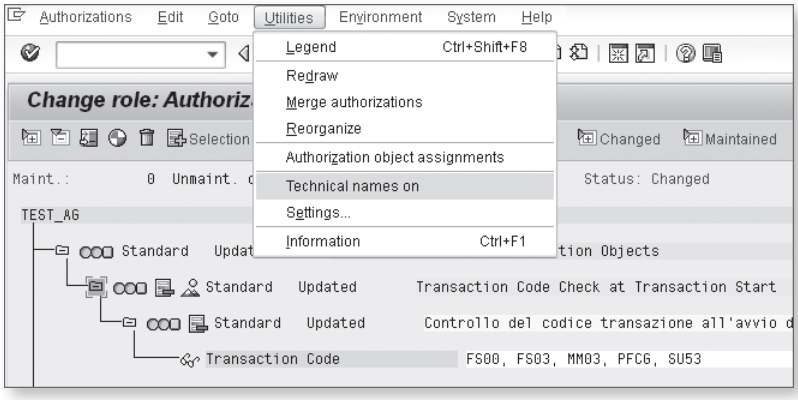


Figure 2 Enable Technical Names in the Authorization Tree

Note that this will enable the technical names only during your session. To enable the names permanently, select Utilities • Settings.

On the resulting screen, select Show Technical Names and click on the Settings button. The result of this setting is shown in Figure 3, where you can see all of the technical names for each authorization element.

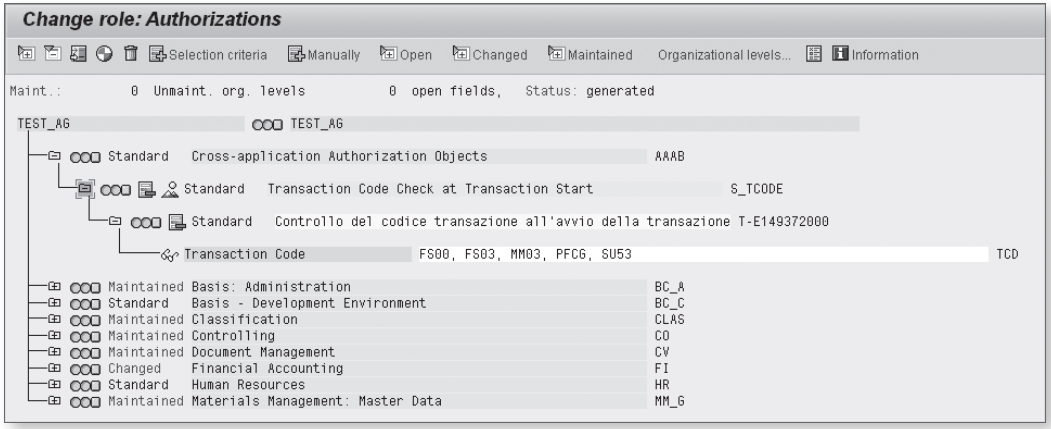


Figure 3 Technical Names Permanently On

You can also see all Transaction PFCG configuration setting for users by browsing Table TPR_PREF or by executing the `SUPRN_GET_USER_PREFS` function module and entering a user ID.

Tip 34

Creating a Sustainable Authorization Roles Naming Convention

Because the number of roles in an SAP system can be very large, you need to adopt a good naming convention to quickly discover errors and increase governance.

Many years ago we adopted the naming convention described in this tip—without this convention, maintaining a clean roles library and sharing it with other security managers would be impossible. A naming convention will not satisfy everybody, but it's important to adopt one anyway; otherwise, each security manager working in an SAP system will choose his own. With a smart naming convention, you can establish a set of rules to check weekly for errors.

And Here's How ...

When you're involved in a new project, you should make two assumptions:

1. All authorizations are created through the role concept using Transaction PFCG (also called Roles Maintenance or Profile Generator), so you aren't considering the naming convention of profiles (automatically generated or manual).
2. The security concept is implemented through the job role concept (end users will only have composite roles directly assigned to them).

To fully understand the naming, it's important to classify the kind of "roles" you'll be using. If you go to the Role Maintenance screen, you'll see that you can create (and manage) the following:

- Single roles (also called authorization roles) that are catalogued as the following:
  - Simple role containing transactions (in the menu) and authorizations
  - Parent (or imparting) role necessary to create family roles
  - Derived role related to one parent role
  - Exception roles containing only manual authorizations and no transaction codes (not in the menu nor in the `S_TCODE` authorization object)
- Composite roles (also called job roles). These contain all kinds of authorization roles. A composite role cannot contain another composite role (the nesting technique is not possible).[1]

Each role has a technical name (30 characters), a short description (80 characters), and a long description. The descriptions (short and long) data will be contained in Table AGR_TEXTS.

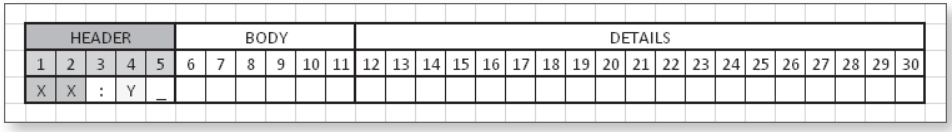
First, you need to decide if it's important that the technical name tells the purpose of the roles; for example, MM_MATERIAL_MANAGEMENT. The answer is no. It's important to understand the purpose of a role, but it isn't necessary to use the technical name for it. Because Table AGR_TEXTS contains a "language" field, we suggest using the short name to describe the role purpose. In this case, the logon language will determine which description will be displayed.

Next, know that it's important to understand the role type from the technical name. If your security concept says that the end user must have only composite roles assigned directly, it will be very easy to find errors. In fact, Table AGR_USERS contains the relationship between users and roles. The Boolean field COL_FLAG will detail whether the roles are direct or not.

To define a naming convention, you must know how many characters are available. The role's technical name is a field of 30 characters. For each character, you have to define the meaning and the possible values. Our proposed naming convention is composed of three parts: header, body, and details.

As shown in Figure 1, the naming convention uses the first five characters as the header.

1 Note that if you specify a language different from English (EN) in the SAP GUI logon screen when you use Transaction PFCG, you may not be able to see the term "Composite" after "Role" on the CREATE button due to the width of the button.



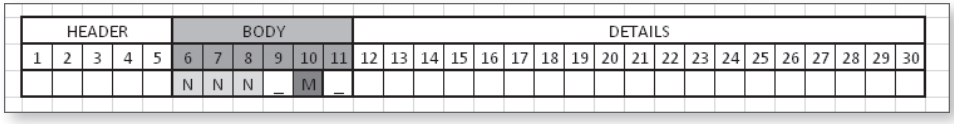
HEADER					BODY						DETAILS																		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
X	X	:	Y	_																									

Figure 1 Role's Naming Convention: Header

- The first two characters (XX) are used to specify the SAP module (e.g., FI, CO, MM, SD, HR, etc.).
- The third character (:) is a delimiter.
- The fourth character (Y) is used strategically to determine the role type:
  - **T:** For template and used for simple and parent roles.
  - **D:** For derived roles.
  - **C:** For composite roles.
  - **E:** For exception roles.
- The fifth character (_) is a delimiter.
- Starting from the sixth character, the naming convention is different for authorization roles and for composite roles.

Authorization Roles

In the authorization roles naming convention, you use characters from number 6 to 8 to specify a counter starting from 001 (see Figure 2). The counter is in the name's space of the SAP module specified by characters 1 and 2 (e.g., MM:T_001 and FI:T_001). We don't expect to go over 999 template authorization roles for each SAP module.



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

Figure 2 Role's Naming Convention: Counter and Access Mode

The tenth character helps you understand the authorization types. In fact, a database can be accessed in three different modes: create, modify, and display. This naming convention extends this concept using the following:

- **M: Management:** This means that the authorizations in the role will grant full authorizations (no limits).

- **E: Edit:** In some cases, it's necessary to authorize a transaction only for modify and not for create (e.g., Transaction MM02).
- **V: View:** There are many situations in which a user can only see business data.

Adopting this logic, a role that has the tenth character "M" will not have any limitation on the ACTVT field. When this character is "E," you should not expect ACTVT = 01 (or an equivalent value) but only ACTVT = 02, 03, and so on. When character "V" is specified, you should not expect ACTVT = 01 or 02 but just ACTVT = 03 or the equivalent.

Suppose that you have to manage the concept of material master data management corresponding to Transactions MM01 (Create), MM02 (Modify), and MM03 (Display).

You'll have three different roles:

- MM:T_001_M will grant all transactions for material master data (Transactions MM01, MM02, and MM03).
- MM:T_001_E will grant all transactions (Transactions MM02 and MM03) except the one for the creation (Transactions MM01).
- MM:T_001_V will grant Transaction MM03 only for display of a material.

In this example, even if the number of roles is three, from the conceptual point of view, they are related only to one business action: material master data management.

The trick is not to reduce the number of roles in the system but to abstract them. As shown in Figure 3, character number 11 is used as a delimiter, and characters from character 12 to 30 give details on the role's content. This area is very important for derived roles. In fact, the goal is to define a "pattern language" that will determine the role's content.

HEADER					BODY						DETAILS																	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
										_	B	0	0	_	W	0	1	_G	0	0									

Figure 3 Details for Authorization Roles

Imagine that the roles are derived on the following organizational levels:

- Company code (BUKRS)
- Plant (WERKS)
- Purchasing Group (EKGRP)

Each level will have a pattern composed by a letter (BUKRS = B, WERKS = W, EKGRP = G) and a counter. Each pattern uniquely determines one or more values for the corresponding field.

Let's see some examples:

- B00 means BUKRS = *
- B01 means BUKRS = 1000
- B02 means BUKRS = 2000
- W00 means WERKS = *

Note that the "*" character is also classified to intercept errors.

Composite Roles

As illustrated in Figure 4, the body area has two sections: CC and NNN. The final character of the body is an underscore ("_") character used as the delimiter (when it makes sense).

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

Figure 4 Composite Role's Naming Convention Body

The naming convention described in this tip has been used in companies with many countries. For each country, there is a set of composite roles.

The meaning of the CC characters is:

- **XX:** Composite roles template (see Tip 37).
- **IT:** Italy country code.
- **FR:** France country code.

The three characters after the country code make up a counter that uniquely determines a job role. For example, MM:C_XX001 is the "Buyer" job role template, FI:C_XX001 is the "G/L clerk."

As described in Tip 37, if you have to create localized job roles for Italy and company code 1000, you use the detail area to determine those roles (see Figure 5). The details will just have a counter that determines a localized composite role.

HEADER					BODY						DETAILS																
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
F	I	:	C	_	I	T	0	0	1	_	0	0	0	1															

Figure 5 Composite Role's Naming Convention Details

The following examples are helpful on a daily basis. It's easy to determine errors by just browsing (with Transaction SE16) in classic security tables:

- Are there authorization roles assigned directly to end users?
  - Table AGR_USERS with an AGR_NAME such as *:T*, *:D*, or *:E* and field COL_FLAG left blank
- Are there display roles with ACTV = 01?
  - Table AGR_1251 with AGR_NAME = *:*_V_*, FIELD = ACTVT, and LOW = 01
- Are there incorrect company codes in derived roles? Check derived roles with pattern B01 (that means BUKRS = 1000).
  - Table AGR_1252 with AGR_NAME = *:D_*, and VARBL = $BUKRS, and LOW not equal to 1000
- Are there transaction codes in exception roles?
  - Table AGR_1251 with AGR_NAME = *:E_* and OBJECT = S_TCODE
- Are all roles documented in the correct language?
  - Table AGR_TEXTS with LINE = 0000 for the short description and LINE > 0000 for the long description
- Are there unexpected "*" values for the company code (BUKRS) field?
  - Table AGR_1252 with AGR_NAME not like *:D_*B00*, field VARBL = $BUKRS, and LOW = *

Tip 35

Evaluating the Manual or Modified Authorization Status during Profile Generator Maintenance

You can evaluate the authorization status in your roles, which will be important during roles maintenance, reviews, or upgrades.

The authorization in roles can assume four statuses: standard, maintained, manual, or modified. The first two are a compliance situation; the second two can be a pitfall during your daily authorization maintenance. For instance, if you insert a manual authorization, this manual authorization isn't automatically removed even if you remove all transaction codes from the role menu. If you change a standard authorization by removing an activity, at the next change Transaction PFCG will replace the change with the standard values. In this tip, we'll show you how to identify these situations and understand why they could create critical problems for your business.

And Here's How ...

Table AGR_1251 contains the details of all roles, authorizations, and authorization objects/values. By browsing this table, you can easily identify all roles that have an authorization status of MANUAL or MODIFIED. Figure 1 shows how you can perform the query to find all roles with these authorization status data. In the MODIFIED field of Table AGR_1251, enter the value "M" (modified status) or "U" (manual status), and press F8 to execute the query.

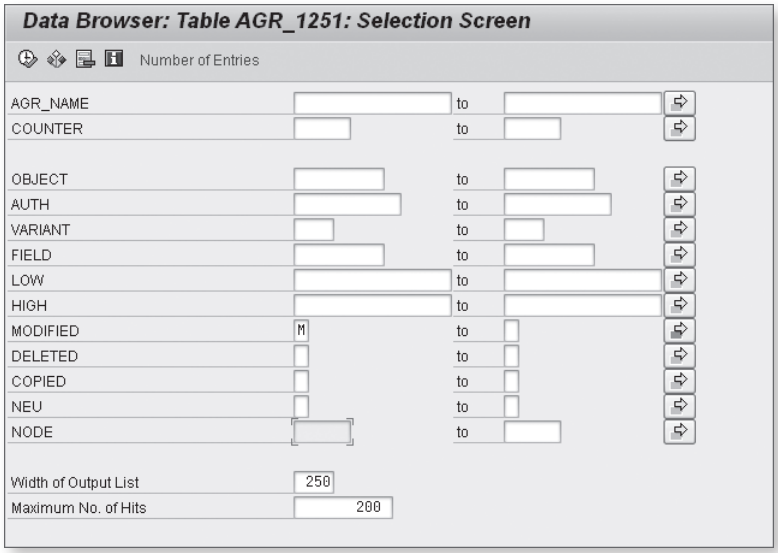


Figure 1 Table AGR_1251 Finds All Roles with M (Modified) or U (Manual) Authorization Status

These two statuses are critical if not properly managed for these main reasons:

- If an authorization has a status of MODIFIED, the values will be overridden by the standard/default details. Let's say you've changed the standard value of an authorization object from activity 01 (create) to 03 (display). The next time you use Transaction PFCG, the profile generator will restore the standard values.
- If you insert an authorization object that has a manual status, you'll have to remove this authorization object manually. For example, if you remove all transactions in the role, the manual authorization object will not be removed automatically.

Modified Authorization

After you've identified all modified authorization objects, you need to set up a manual process to restore the statuses to the standard so it isn't done automatically, thus removing all of your work. Use Transaction SU24 to fine-tune the default values—you have to enter each identified role and delete the change status authorizations. Be sure to leave only standard authorizations.

Manual Authorization

You can't enter manual authorization objects directly in the Transaction PFCG Authorizations tab. Instead, you have to maintain Transaction SU24 and then update the authorization role in order for these objects to have a standard status.

Remember that the manual status authorization can be used in certain cases. For example, to manage the material master data views (authorization object `M_MATE_STA`), you can deactivate this authorization object from all roles defined in your system and manage this authorization object alone in specific roles called authorization roles.

Tip 36

Creating an SAP_ALL Display-Only Role

Using Transaction PFCG, you can create a role that provides display-only user access starting with an existing profile.

You should have a wide range of authorization roles to display the system and configuration data in various situations; for example, to assign roles to consultants for a startup and analysis project in development and quality/training systems. You can use Transaction PFCG to create these kinds of roles and ensure display-only access.

And Here's How ...

To create an empty role, access Transaction PFCG (Change Roles), and go to the AUTHORIZATIONS tab (see Figure 1).

The system will ask if you want to use a predefined authorization model. If you want to select a template to fill the AUTHORIZATION tab, choose the SAP_ALL template profile, and then click on ADOPT REFERENCE (Figure 2). In this way, all authorizations entered in the SAP_ALL profile are used to fill up the role.

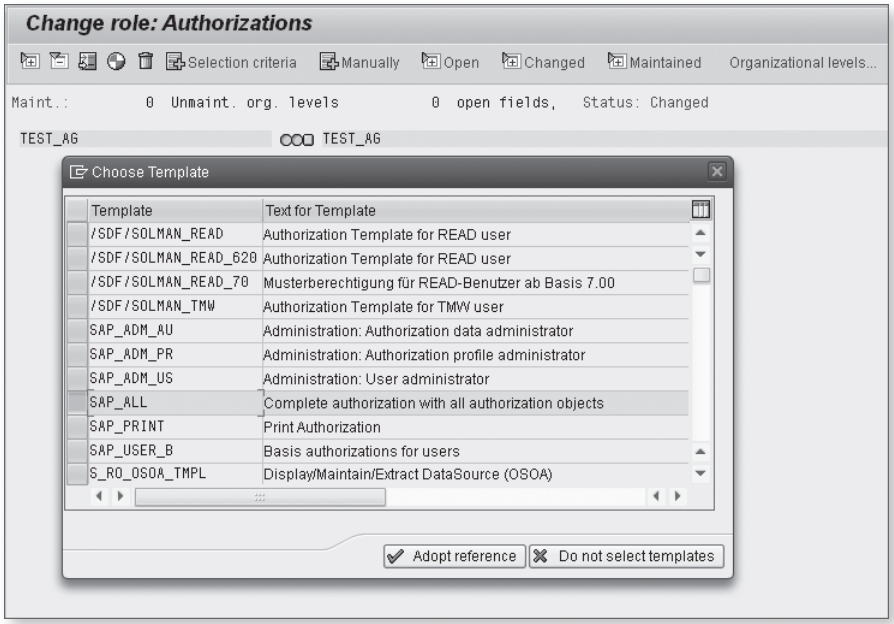


Figure 1 Choose Template Pop-Up Screen

Alternatively, when you are in the authorization tree, you can use the following menu path to enter all authorizations:

EDIT • INSERT AUTHORIZATION(S) • FULL AUTHORIZATION

By doing this, you can ensure that all authorization objects that are entered in the SAP_ALL profile are covered by the role that you are creating.

Next, ensure that this role lets the user access the system in display mode only. Use Table TACT to discover all possible activity entries for the ACTVT field. Common display activity codes are 03 (display), 04 (print), 08 (display changes documents), and 09 (display prices). Activity codes 27, 28, 29, 53, and 54 display activities mainly used in the Controlling (CO) module; 33 reads on the file system directory. Be sure that in all authorization objects with the ACTVT field, the values chosen are in display.

Look for all authorization object fields with the ACTVT field, and set them in display mode by entering the previously listed values (see Figure 2). To do this, click on the binoculars icon, and then find all authorization objects with the ACTVT field.

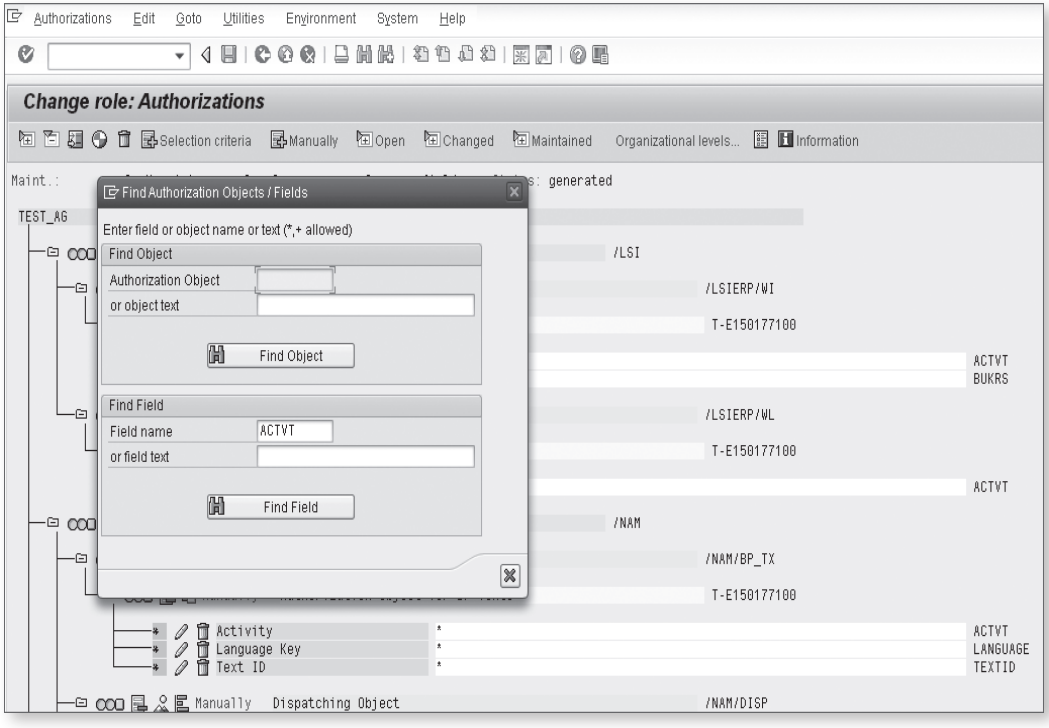


Figure 2 Display Authorizations with the ACTVT Field

If the ACTVT values entered are only in write mode for a certain authorization object, you have to deactivate the object. This is time consuming but is a one-time process. At the end of this setup activity, check whether all values are properly deactivated by accessing Transaction SE16 and browsing Table AGR_1251. Enter your role name in the AGR_NAME field, enter the value "ACTVT" in the FIELD field, and exclude all display values previously listed (03, 04, 08, etc.) in the LOW field. The query in Figure 3 shows you all of the authorization objects where ACTVT is not in the display mode values.

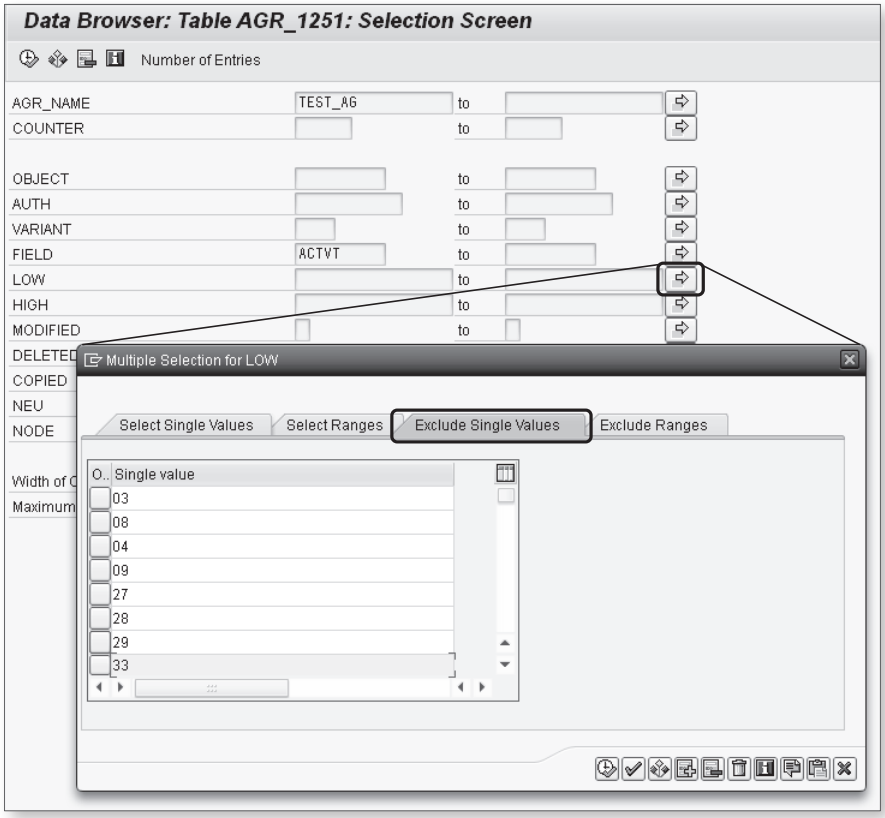


Figure 3 Browsing Table AGR_1251 with Transaction SE16

Note that not all authorization objects have an ACTVT field for managing the activity; in the HR component, the ACTVT field is AUTHC, where the value "R" means read mode. It's also a good approach to disable all authorization objects that are critical, based on your internal policy for your company. You can also limit transaction execution by entering a range in `S_TCODE` authorization objects (e.g., if this role should be allowed to run all transaction except Transaction PFCG). Keep in mind that a range approach can decrease governance because it's more difficult to document the roles and be aware of what the roles allow.

Tip 37

Maintaining an Aligned Set of Job Roles with a Naming Convention

Only simple roles can be derived, but you can create a derived concept for composite roles by using a smart naming convention.

Most people don't fully understand that the SAP security concept is based on two consequent checks. The first check is done at the transaction start (performed by the kernel through authorization object `S_TCODE`). All other checks are performed through the ABAP statement `AUTHORITY-CHECK`. For this reason, many job roles must be created to have the benefits of the `AUTHORITY-CHECK` statements.

After many months of maintenance, the initial design on job roles will be lost. By adopting a smart naming convention, however, you can periodically perform a check to ensure that the governance of your roles is maintained.

And Here's How ...

For example, from the HR point of view, it's necessary to have many users designated as "buyers." Suppose that each single buyer must be authorized only for his own purchasing group. For HR, only one job role is necessary. From the security point of view, each buyer must have a job role in which the derived roles within it contain the corresponding purchasing group.

Suppose now that a single buyer discovers that he needs a new simple role (because it contains a missing transaction code), and this simple role is given only to him (inserting the simple role in the composite role). This will create a different concept of a buyer because only one of the buyer's roles has been authorized to the missing transaction.

In SAP, it isn't possible to derive the composite roles. This powerful technique is available only for simple roles. When a simple role has one or more derived roles, it's called a parent (or imparting).

A good way to maintain an aligned set of job roles (e.g., buyers) is to create an additional composite role called *Template* in the system (Template Composite Role), which contains all necessary simple roles (or parent roles). When the time comes to implement a limited job role (to authorize the user for a single purchasing group), you can create a new composite role that contains the corresponding derived roles.

Figure 1 shows the final job roles architecture. The top layer includes the simple and derived roles. The bottom layer contains the composite roles (template on the left, and derived on the right).

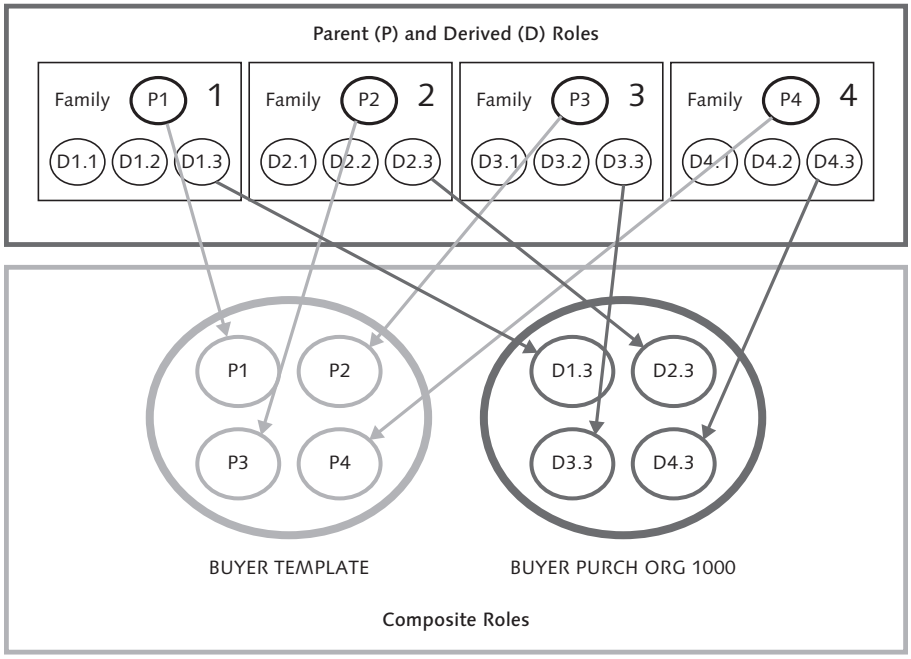


Figure 1 Job Role Template Authorization Concept

To avoid terminology misunderstanding, we call the derived job roles "localized."

We have implemented this job roles architecture in many companies (involving thousands of users) with a large return on investment (ROI). When it's necessary to perform a Segregation of Duties (SoD) risk analysis, running it only on the template job roles dramatically reduces the remediation phase. When delivering the job roles documentation (to view the transactions inside), only the template job roles are documented.

The only difficult part of this implementation is to keep the template job roles and the related localized job roles aligned. By using a smart naming convention and looking into Table AGR_AGRS (which contains the relationship between composite roles and simple/derived roles), it's easy to create a simple ABAP program that will discover all misalignments.

Tip 38

Designing and Assigning a Basic Role to All Users

You can assign all users to a customized common role that contains non critical and basic activities.

It's a common best practice to create a role to assign to every user. This role should contain some basic activities (see own print spool, job, and execute Transaction SU53 [Authorization Error Log]). Unfortunately, it's not always clear what you should or should not include in this role. Some customers may include the ability to create financial documents in this role because SAP ERP is used mainly in financial department transactions or other business transactions. However, after you roll out the SAP ERP system to other departments or external workers, this ability to create financial documents is inherit for all users, which may cause a missing Segregation of Duties (SoD).

And Here's How ...

Through Transaction PFCG, you can create a common role that contains some uncritical and useful transactions to assign to every user that is defined. Table 1 shows a small list of transaction codes that you can insert into the common role and some recommendations regarding authorization object values used from these transactions.

Transaction Code	Description
SBWP	SAP Business Workplace
SEARCH_SAP_MENU	Find in SAP Menu
SMX	Display Own Jobs

Table 1 Minimal List of Transaction to Insert in Common Roles

Transaction Code	Description
SOSB	Send Order Overview (User)
SP02	Display Spool Requests
SU50	Own Data
SU53	Evaluate Authorization Check

Table 1 Minimal List of Transaction to Insert in Common Roles (Cont.)

If all of your users use SAP Office, you can also insert SAP Office transactions such as Transaction SBWP.

All of the transactions in Table 1 need to perform some auhtorization objects—you have to set these up properly to avoid security holes. Some of these transactions bring some critical objects to Transaction PFCG's Authorization tab. Table 2 lists the objects that you should deactivate.

Object	Object description	Status
S_ADMI_FCD	System Authorizations	Deactivated
S_USER_GRP	User Master Maintenance: User Groups	Deactivated
S_DEVELOP	ABAP Workbench	Deactivated

Table 2 Critical Authorization Ojects to Deactivate

S_ADMI_FCD authorization object authorization objects allow some administrative functions; S_USER_GRP allows you to see beyond their own authorization check errors, as well as the authorization errors for other users and the possibility to perform developing activities through S_DEVELOP. This last object shouldn't be present in any production roles. You can leave the standard values for all other objects.

Based on your company policy, you can directly enter the authorization object S_GUI in this basic role to allow any user to export data from SAP.

Some further reccomendations are:

- Your basic role might also be assigned to external users. If this is the case, make sure to reevaluate the minimum set of authorizations you want to grant.
- You shouldn't insert business transactions in this role. All business roles must be inserted into the business composite roles for each business area, for the prior reason.

Tip 39

Maintaining Derived Roles to Improve Authorization Maintenance

You can define a derived role via Transaction PFCG to help customers manage their authorizations on different domains.

The ability to derive roles has been provided by SAP from release 4.5 of SAP R/3. This kind of role helps customers manage the same role activities and authorizations they normally perform, but on different data domains. As an example, when performing material master creation on plant 1000 or 2000, the SAP activity and transaction is the same, but the domain data is different and there's a different plant. A common misunderstanding is that managing derived roles can be done similarly to simple/single roles, but if you manage derived roles as simple roles, you'll lose the benefit of using derived roles.

And Here's How ...

You can create a derived role through Transaction PFCG by clicking on the Single Role button (see Figure 1).

The system will take you to the Create Roles interface, where you enter the parent role name in the Derive from Role field, as shown in Figure 2, and confirm the pop-up proposed by the system. Here you can maintain information in the Menu tab and all authorization data already defined in the parent role without defining several simple roles.

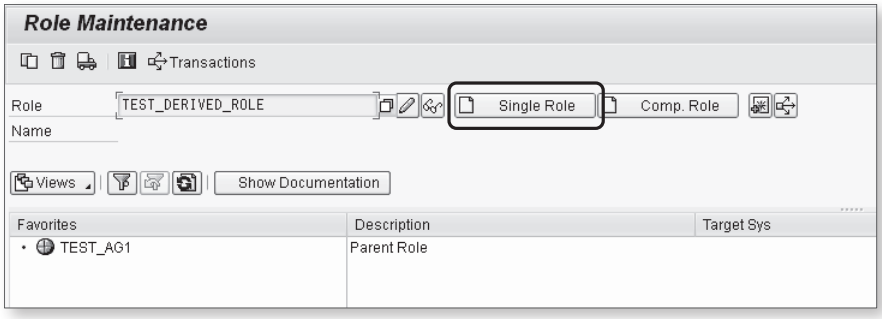


Figure 1 Create a Single Role in Transaction PFCG

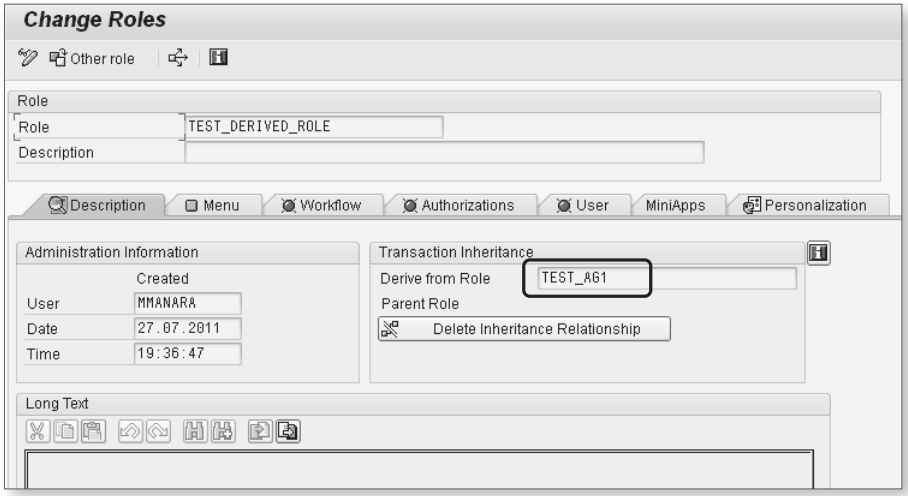


Figure 2 Enter the Name of Parent Role in the Derived from Role Field

The authorization data in derived roles are inherited from the parent role, with the exception of the value of organizational levels. When you're in the AUTHORIZATION tab, if the role contains an organizational field, an organizational pop-up will appear in order for you to populate this field (see Figure 3). All authorization values are inherited from the parent role except these organizational levels that are populated directly in the derived roles.

The goal of this architecture is to maintain only the parent role menu and authorizations and then adapt all of the data in derived roles. This will keep you from accidentally creating several copies of identical roles that are different only at the organizational level.

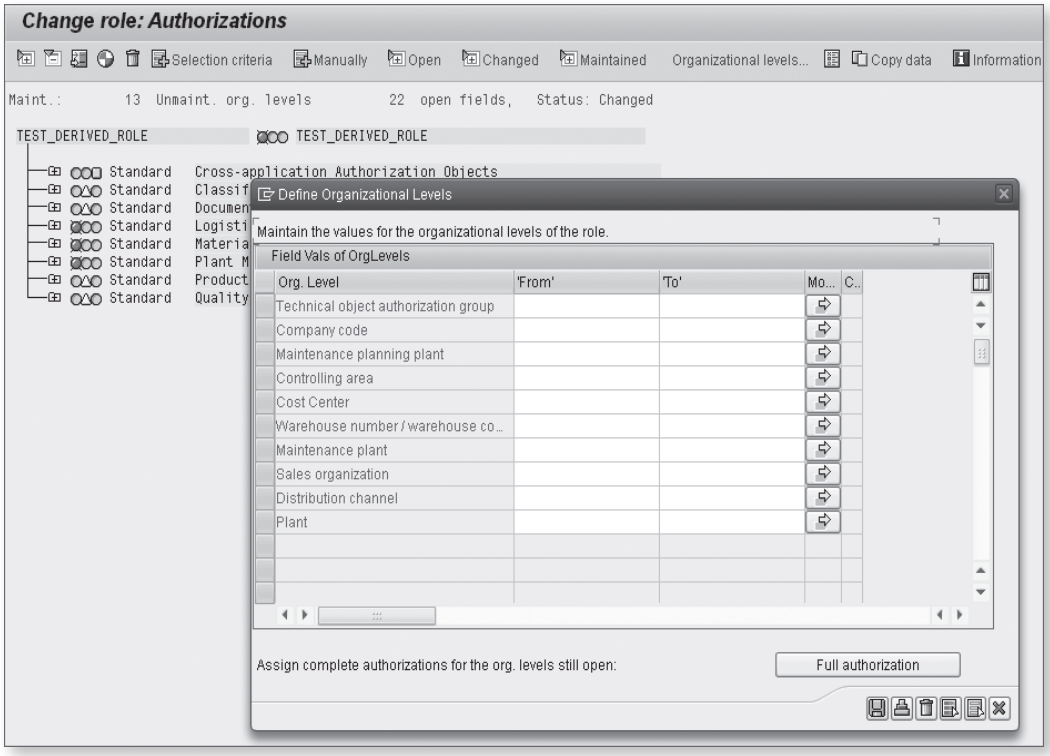


Figure 3 Define Organizational Levels Pop-Up in the Transaction PFCG Authorization Tree

Make sure that you don't maintain authorization values (non organization fields) directly in the derived roles. This will cause the override of these values when you perform the role adjustments (which you do by clicking on the circled button Generate Derived Roles in Figure 4) in the parent role.

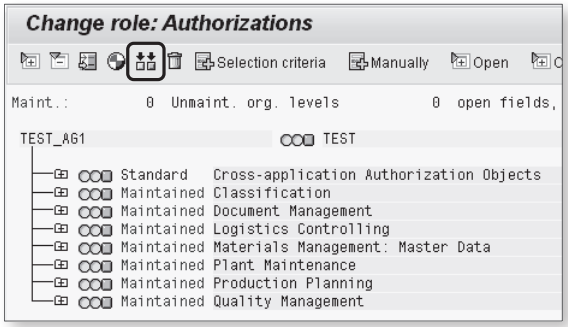


Figure 4 Generate Derived Role Icon in Transaction PFCG Authorization Tree

Tip 40

Discovering Misalignment between Transactions by Downloading Data to Spreadsheets

You can ensure that the role menu and authorization object S_TCODE are always aligned to avoid security issues.

To avoid security holes, you need to maintain and check the alignment between the role menu and authorization object S_TCODE in the Role Authorization tab. If your roles are not aligned, you can waste a lot of time during the upgrade phase. This tip shows you how to discover and manage these misalignments.

And Here's How ...

Suppose you've defined several roles. You need to check whether these roles are aligned; in other words, you want to know if all transaction codes entered in the Role Menu tab are the same as those that have been entered into the Role Authorization tab in authorization object S_TCODE. To do this, you have to perform two steps:

1. Download Table AGR_TCODES in a spreadsheet for all roles involved by using Transaction SE16 and then following the path System • List • Save • Local File.
2. Download Table AGR_1251 in a spreadsheet for all roles involved where the authorization object is S_TCODE in the same manner.

After these two tables have been downloaded, you can check for misalignments by following these steps:

1. Create an additional with two sheets; the first sheet contains the first table you downloaded, and the second sheet contains the second table downloaded. The first sheet represents the role menu, and the second represents the transaction inserted into the `S_TCODE` authorization object. Figure 1 shows the data contained in these two sheets. Remove all columns that aren't being used.

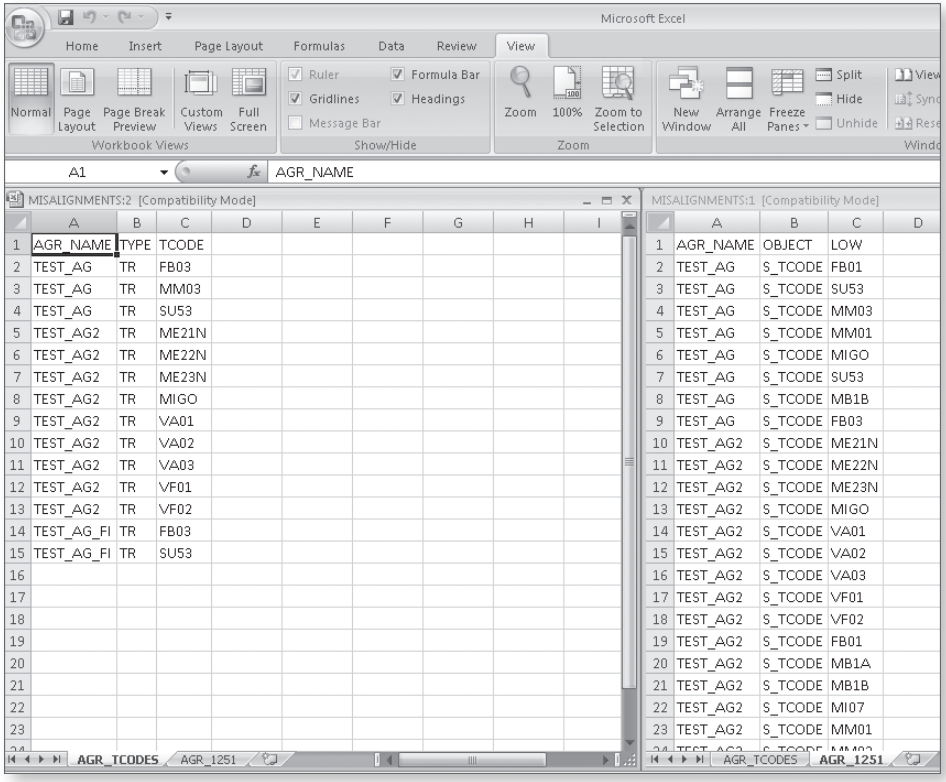


Figure 1 Tables Downloaded into Excel Spreadsheets

2. Define a new column in each sheet named "KEY" and concatenate columns A and C. This formula will use a lookup formula to discover any misalignments. Figure 2 shows the same spreadsheet with these two additional columns. Column D represents the key, and column E represents the lookup formula to find the misalignments.

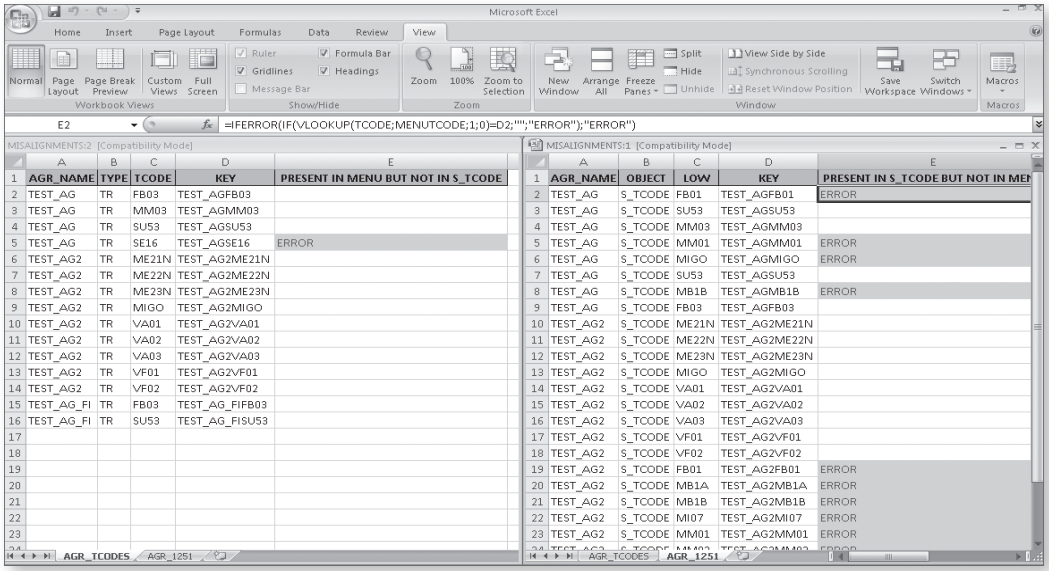


Figure 2 Misalignments between Authorization Objects S_TCODE and the Role Menu

3. Use the following formula to discover any misalignments between authorization object S_TCODE and the role menu (Figure 2, column E):

   `=IFERROR(IF(VLOOKUP(TCODE;MENUTCODE;1;0)=D2;"";"ERROR");"ERROR")`

   TCODE and MENUTCODE are Microsoft Excel names that represent column D in each sheet. TCODE is the name that represents column D in sheet AGR_1251, and MENUTCODE is the name that represents column D in sheet AGR_TCODES.

   You can use this formula to find misalignments between the role menu and S_TCODE:

   =IFERROR(IF(VLOOKUP(MENUTCODE;TCODE;1;0)=D5;"";"ERROR");"ERROR")

With this tip, you can identify whether a transaction code is present in the S_TCODE authorization object but not in the role menu, and vice versa.

In some cases, there are standard misalignments when you insert a transaction code into the role menu and the transactions inserted by SAP into object S_TCODE. For example, when you insert Transaction SUIM, this transaction represents a tree that is composed of other transaction codes. When you insert Transaction SUIM in the menu, you can find additional linked transactions in object S_TCODE.

Tip 41

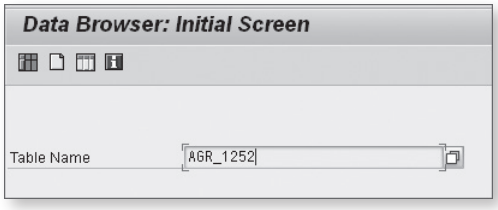
Finding Misinterpreted Authorization Wildcards in Your Roles

You can find wildcard values in your roles by browsing SAP tables and using Microsoft Excel or Access to export your results.

You can use wildcard characters in authorization values. However, if an authorization value contains other characters after an asterisk, the SAP kernel ignores these characters during the authorization check. For example, the value A*B* is actually interpreted as A*. You can find these cases quickly by browsing SAP tables and exporting your findings for analysis through Microsoft Excel or Access.

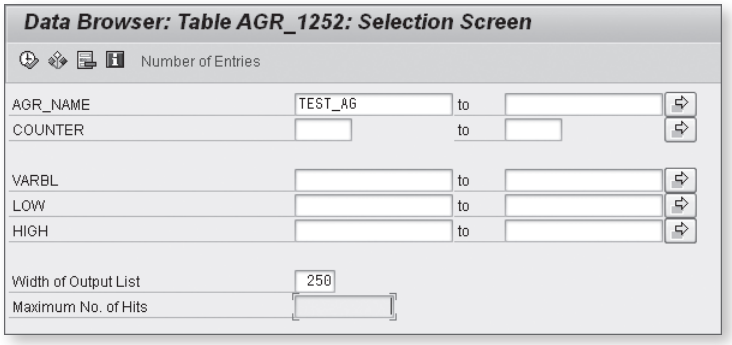
And Here's How ...

Through Transaction SE16, you can directly browse SAP tables that contain the authorization objects and values of a role (shown in Figure 1). You can also browse Tables AGR_1251 or AGR_1252. This example shows you how to find which authorization values in your system contain wildcards that have not properly been set up via Table AGR_1252, which contains the organizational authorization level values for a role. Fill in the TABLE NAME field and press Enter.



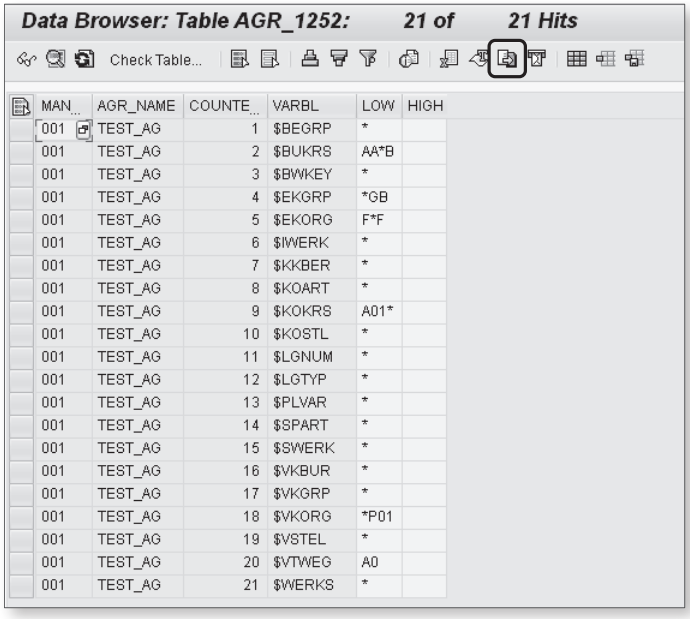
« *Figure 1 Execute Transaction SE16 on Table AGR_1252*

In the selection screen interface (see Figure 2), enter your role name in the AGR_NAME field and press F8.



« *Figure 2 Transaction SE16 Selection Screen*

As a result, you can see all authorization organization values in this role (see Figure 3).

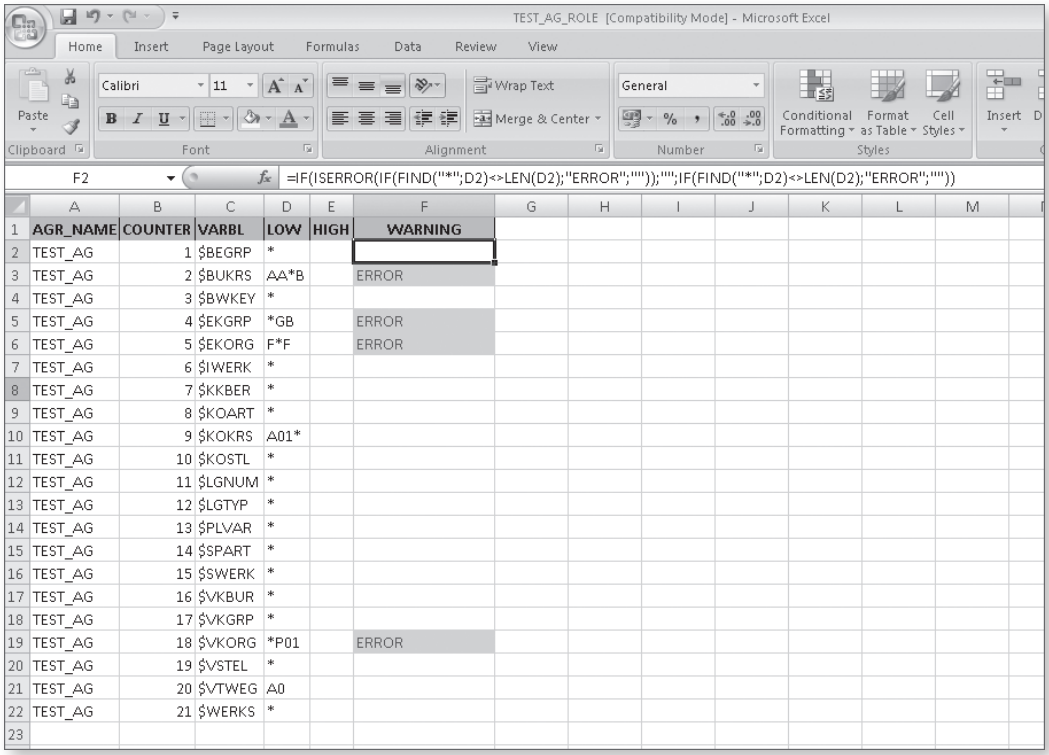


MAN...	AGR_NAME	COUNTE...	VARBL	LOW	HIGH
001	TEST_AG	1	$BEGRP	*	
001	TEST_AG	2	$BUKRS	AA*B	
001	TEST_AG	3	$BWKEY	*	
001	TEST_AG	4	$EKGRP	*GB	
001	TEST_AG	5	$EKORG	F*F	
001	TEST_AG	6	$IWERK	*	
001	TEST_AG	7	$KKBER	*	
001	TEST_AG	8	$KOART	*	
001	TEST_AG	9	$KOKRS	A01*	
001	TEST_AG	10	$KOSTL	*	
001	TEST_AG	11	$LGNUM	*	
001	TEST_AG	12	$LGTYP	*	
001	TEST_AG	13	$PLVAR	*	
001	TEST_AG	14	$SPART	*	
001	TEST_AG	15	$SWERK	*	
001	TEST_AG	16	$VKBUR	*	
001	TEST_AG	17	$VKGRP	*	
001	TEST_AG	18	$VKORG	*P01	
001	TEST_AG	19	$VSTEL	*	
001	TEST_AG	20	$VTWEG	A0	
001	TEST_AG	21	$WERKS	*	

« *Figure 3 Organization Authorization Values in Role TEST_AG and the Local File*

Export this table into an Excel spreadsheet by clicking on the Local File icon circled in Figure 3, and then choosing the Spreadsheet checkbox. Use a formula (such as "`=IF(ISERROR(IF(FIND("*";D2)<>LEN(D2);"ERROR";"")));"";IF(FIND("*`

”;D2)<>LEN(D2);”ERROR”;””))”) to find the misinterpretation of these values from the SAP kernel (see Figure 4).



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	AGR_NAME	COUNTER	VARBL	LOW	HIGH	WARNING							
2	TEST_AG	1	$BEGRP	*									
3	TEST_AG	2	$BUKRS	AA*B		ERROR							
4	TEST_AG	3	$BWKEY	*									
5	TEST_AG	4	$EKGRP	*GB		ERROR							
6	TEST_AG	5	$EKORG	F*F		ERROR							
7	TEST_AG	6	$IWERK	*									
8	TEST_AG	7	$KKBER	*									
9	TEST_AG	8	$KOART	*									
10	TEST_AG	9	$KOKRS	A01*									
11	TEST_AG	10	$KOSTL	*									
12	TEST_AG	11	$LGNUM	*									
13	TEST_AG	12	$LGTYP	*									
14	TEST_AG	13	$PLVAR	*									
15	TEST_AG	14	$SPART	*									
16	TEST_AG	15	$SWERK	*									
17	TEST_AG	16	$VKBUR	*									
18	TEST_AG	17	$VKGRP	*									
19	TEST_AG	18	$VKORG	*P01		ERROR							
20	TEST_AG	19	$VSTEL	*									
21	TEST_AG	20	$VTWEG	A0									
22	TEST_AG	21	$WERKS	*									
23													

Figure 4 Discover SAP Kernel Misinterpretation

In this manner, you can quickly find these misinterpretations on several thousands of roles and fix them.

Tip 42

Performing Mass Downloads and Uploads of Standard Authorization Values

You can save time by performing a mass download and upload of standard authorization values.

To upgrade or load several authorization objects in Transaction SU24 (Table USOBT_C), you can use a special, rarely used upload functionality to save time and avoid entering the authorizations manually. This functionality to mass-download and upload is useful mainly when you need to create a backup of your customer data, or when you have defined several custom transactions and you have to load several authorization objects and values.

And Here's How ...

Go to Transaction SU24, where you can find two buttons at the top of the screen: DOWNLOAD and UPLOAD (see Figure 1).

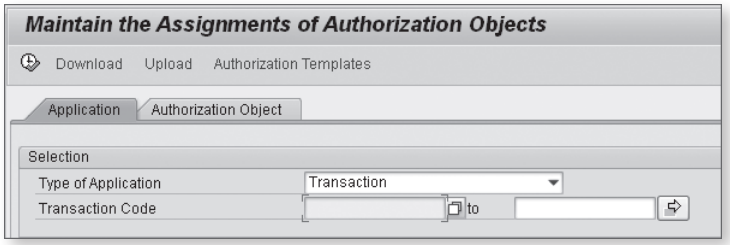


Figure 1 Transaction SU24 Download and Upload Buttons

Through this functionality, you can mass-download or upload the standard authorization values. Click on the DOWNLOAD button to access the screen shown in Figure 2, where you can see the download functionality.

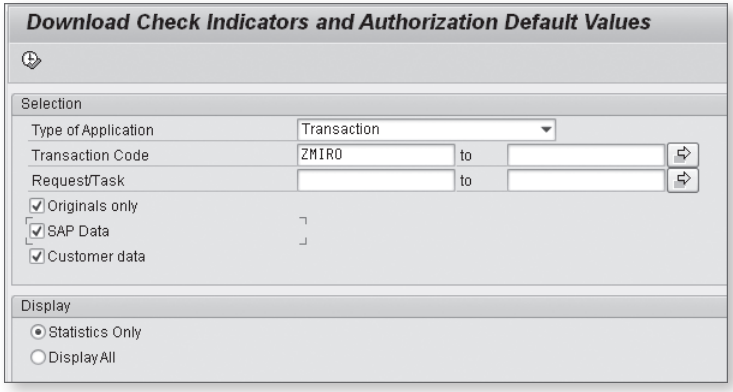


Figure 2 Download Functionality in Transaction SU24

You can download the data for originals only, SAP data, and customer data for one or more transactions by selecting the corresponding checkboxes:

- ORIGINALS ONLY means all transactions data created in that system will be downloaded on your local PC.
- SAP DATA means only SAP data Tables USOBT and USOBX will be downloaded.
- CUSTOMER DATA means only customer data Tables USOBT_C and USOBT_C will be downloaded.

After you've selected what data you want to download, execute the transaction by pressing F8. The system will ask if you want to save the data result in a text file. You can then upload or store this information as a backup to the text file.

Tip 43

Setting Up Mass Adjustments for Derived Roles

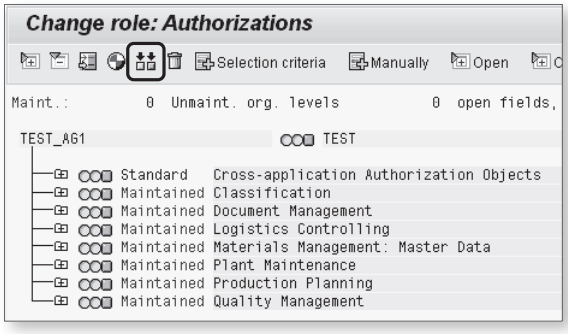
You can adjust derived roles en masse to align parent/child roles by using a standard function module.

Transaction PFCG doesn't allow you to mass-adjust derived roles to align the authorization from parent to child roles; instead, you're normally forced to manually adjust them role by role. This tip shows you a standard function module that saves time by performing mass adjustments for your roles.

And Here's How ...

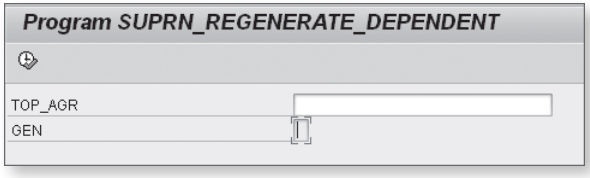
After connecting the template role with the derived role, the DERIVE ROLE icon (circled in Figure 1) appears in the AUTHORIZATIONS tab in the template role. You use this to transfer authorization data from the template to the derived role.

You're now performing maintenance at the template role, and all derived roles are aligned and updated automatically. Unfortunately, there's not a mass way to perform this action on several template roles at the same time.



« *Figure 1 Generate Derived Role Icon in Transaction PFCG Authorization Tree*

You can work around this limit by exploiting the standard program SUPRN_REGENERATE_DEPENDENT, which you execute through Transaction SE38. This program is shown in Figure 2, which allows you to transfer the authorization data from a role template into all derived roles of that template.



« *Figure 2 SUPRN_REGENERATE_DEPENDENT Form Executed through Transaction SE38*

Enter the father role name in the TOP_AGR field, and type "X" in the GEN field when you want to generate the role.

You can write a small piece of ABAP code to recall this program and pass in all template roles. In this way, you can also schedule a background job in the development system to keep your roles aligned between father and child roles.

Tip 44

Troubleshooting Authorization Problems for Users

As an administrator, you can monitor and analyze an authorization problem during a user session with Transaction ST01.

In normal cases, when a user receives an authorization error, the end user or administrator will execute Transaction SU53; this transaction shows that the last authorization check failed. However, you'll find that this error log isn't sufficient to resolve the problem when working with custom or standard transactions. In these cases, you'll find it necessary to perform an authorization trace during a user session with Transaction ST01.

And Here's How ...

To place a trace on a user, execute Transaction ST01 and flag the AUTHORIZATION CHECK indicator under the TRACE COMPONENTS box (❶). Click on the GENERAL FILTERS button (❷), and enter the user ID in the TRACE FOR USER ONLY field (❸, see Figure 1).

After setting Transaction ST01 filters, click on the TRACE ON button. From this moment until you click on TRACE OFF, all activities performed by the user MMANARA are logged depending on your earlier settings (in this case, only whether the authorization check passed and failed).

After the user completes the test session and after you terminate the trace, you can analyze the activities performed by the trace by clicking on the ANALYSIS button. The system shows you the analysis filter selections (see Figure 2) where you have to enter the test user ID and type F8.

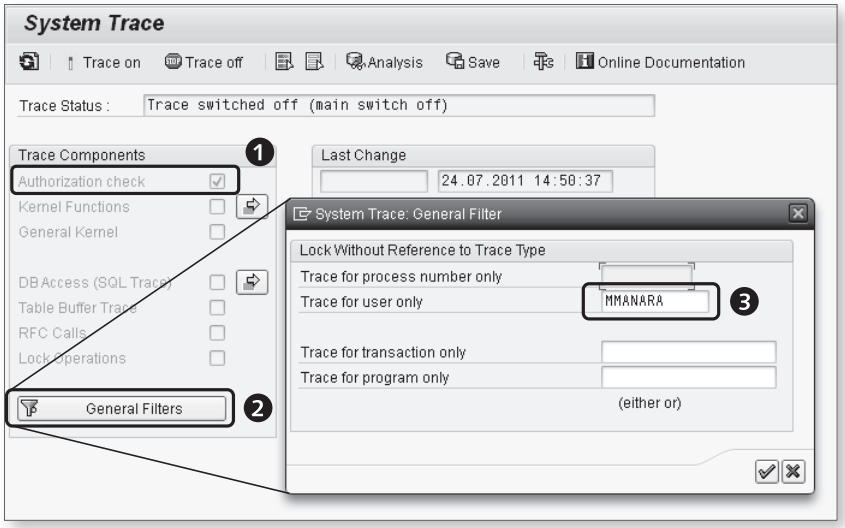


Figure 1 Setting Up Transaction ST01 Filters

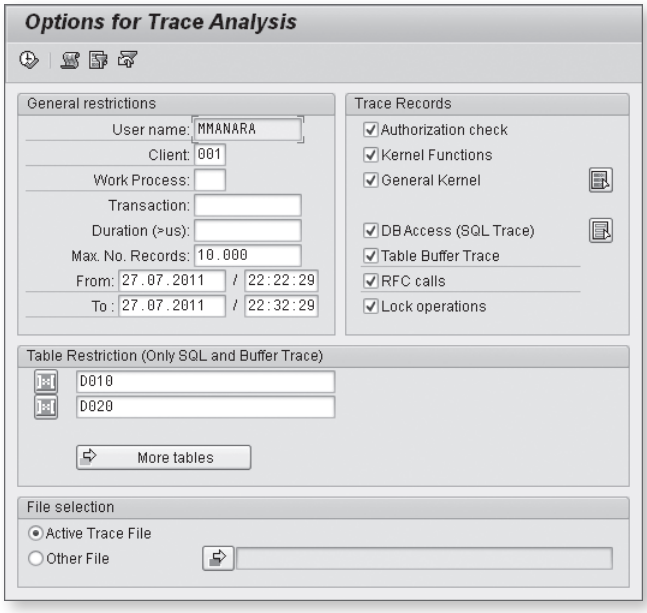
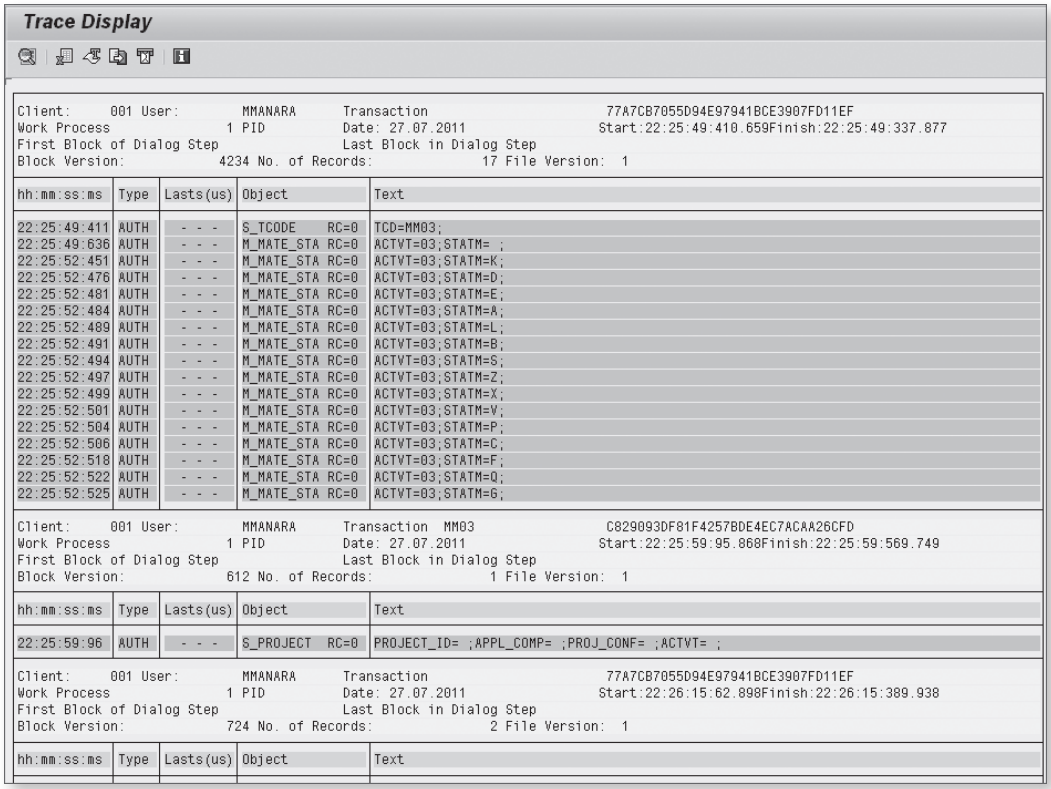


Figure 2 Selection Criteria in Trace Analysis

You'll now see the log that shows all authorization checks that have been performed in Figure 3. If the return code equals zero, the authorization check has passed. If the value is different from zero, the authorization check failed.



Trace Display

Client: 001 User: MMANARA Transaction 77A7CB7055D94E97941BCE3907FD11EF
Work Process 1 PID Date: 27.07.2011 Start:22:25:49:410.659Finish:22:25:49:337.877
First Block of Dialog Step Last Block in Dialog Step
Block Version: 4234 No. of Records: 17 File Version: 1

hh:mm:ss:ms	Type	Lasts(us)	Object	Text
22:25:49:411	AUTH	- - -	S_TCODE RC=0	TCD=MM03;
22:25:49:636	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM= ;
22:25:52:451	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=K;
22:25:52:476	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=D;
22:25:52:481	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=E;
22:25:52:484	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=A;
22:25:52:489	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=L;
22:25:52:491	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=B;
22:25:52:494	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=S;
22:25:52:497	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=Z;
22:25:52:499	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=X;
22:25:52:501	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=V;
22:25:52:504	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=P;
22:25:52:506	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=C;
22:25:52:518	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=F;
22:25:52:522	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=Q;
22:25:52:525	AUTH	- - -	M_MATE_STA RC=0	ACTVT=03;STATM=G;

Client: 001 User: MMANARA Transaction MM03 C829093DF81F4257BDE4EC7ACAA26CFD
Work Process 1 PID Date: 27.07.2011 Start:22:25:59:95.868Finish:22:25:59:569.749
First Block of Dialog Step Last Block in Dialog Step
Block Version: 612 No. of Records: 1 File Version: 1

hh:mm:ss:ms	Type	Lasts(us)	Object	Text
22:25:59:96	AUTH	- - -	S_PROJECT RC=0	PROJECT_ID= ;APPL_COMP= ;PROJ_CONF= ;ACTVT= ;

Client: 001 User: MMANARA Transaction 77A7CB7055D94E97941BCE3907FD11EF
Work Process 1 PID Date: 27.07.2011 Start:22:26:15:62.898Finish:22:26:15:389.938
First Block of Dialog Step Last Block in Dialog Step
Block Version: 724 No. of Records: 2 File Version: 1

hh:mm:ss:ms	Type	Lasts(us)	Object	Text

Figure 3 Authorization Trace Log

Let's quickly review a few important things to keep in mind when using Transaction ST01:

- Transaction ST01 is application server dependent. In other words, if your system has more than one application server (you can determine this by using Transaction SM51), you have to be sure that the Transaction ST01 trace is performed in the same application server the user is logged into. To do that, in Transaction SM51, you can perform a remote logon to another application server by typing Ctrl+Shift+F8 (see Figure 4).
- Take a look at your time zone and clock. If the application server has a time that is different from your PC, then during your analysis, after performing the trace, you might select a trace log outside the time recorded.

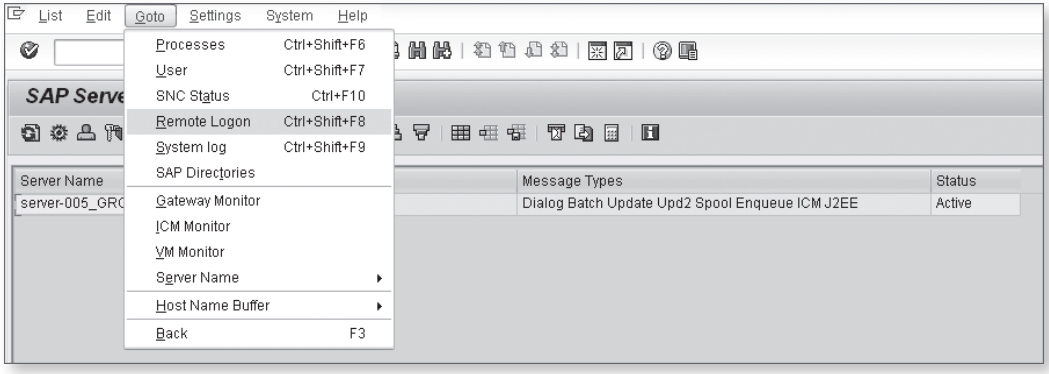


Figure 4 Perform a Remote Logon in Transaction SM51

- You can view in what application server a user is currently logged in to by using Transaction AL08 or by asking the user directly—he can see the application server name in the bottom-right area of your SAP GUI (Figure 5).

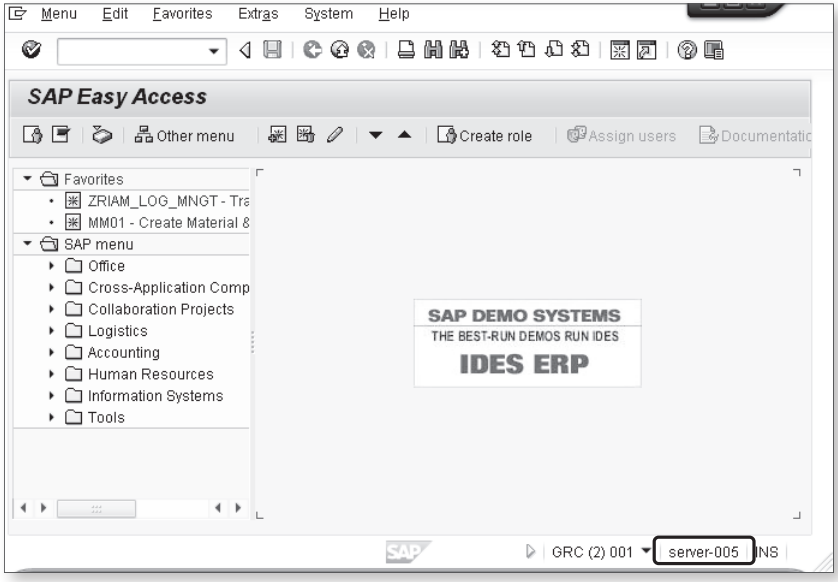


Figure 5 Discover the Application Server a User Is Logged In To

Tip 45

Customizing Your Tree Menu Settings to Avoid Duplicate Structures

You can customize the tree structure in your role menu with Transaction PFCG to avoid creating duplicate structures.

During role menu maintenance in Transaction PFCG, you may have to redefine or update the menu, which may cause you to insert an already existing tree structure from the SAP standard menu. Each time you insert the same menu tree, the SAP system doesn't merge the tree path, but reinserts all of the already existing menu tree to cause a duplicate structure. You can avoid this behavior by customizing the Transaction PFCG menu tree.

And Here's How ...

When you define a new role, the first thing you need to do is define the role menu if this role contains transactions. Figure 1 shows the Transaction PFCG role menu. The circled From SAP Menu button on the bottom left of the figure allows you to import a transaction or insert a tree with all transactions inserted into it from the SAP standard menu.

After you've clicked on this button, you can flag which transaction or menu tree to import and then click on the Transfer button (Figure 2).

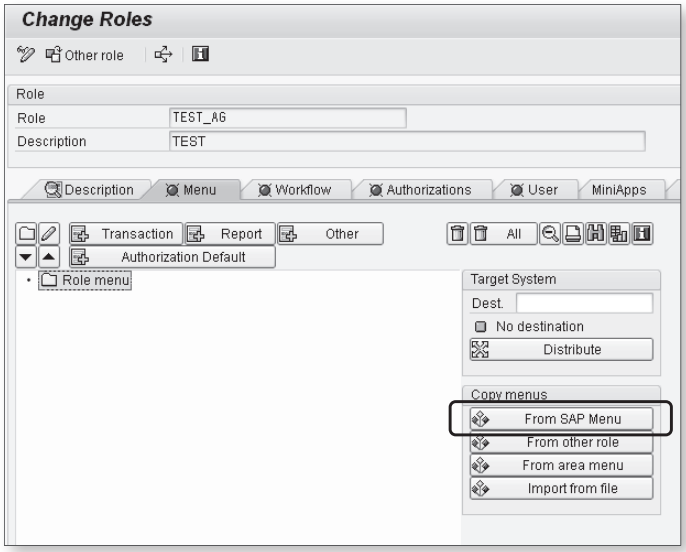


Figure 1 Copy Menus from the SAP Menu Button in the Transaction PFCG Menu Tab

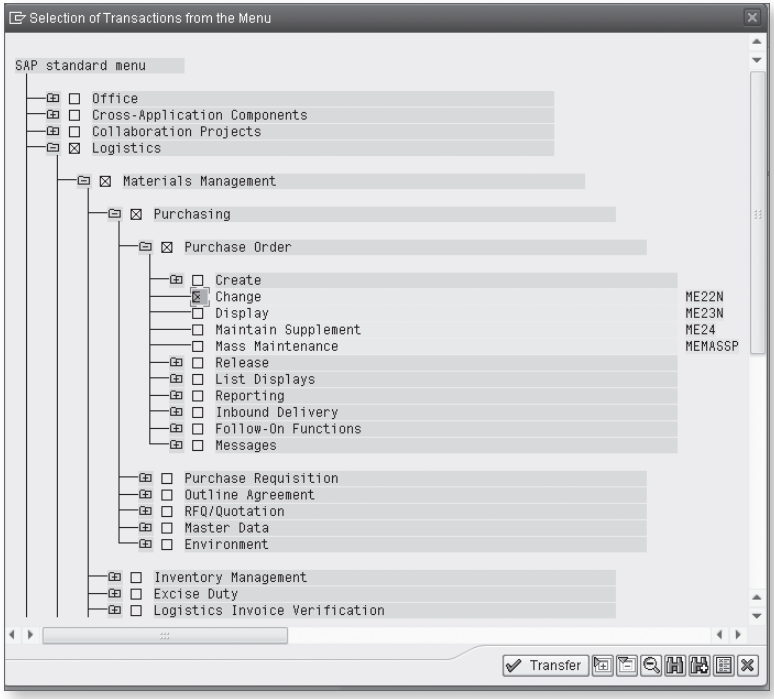


Figure 2 Flagging the Transaction or Menu Tree

The result of this import is shown in Figure 3. The menu tree and Transaction ME22N have been imported.

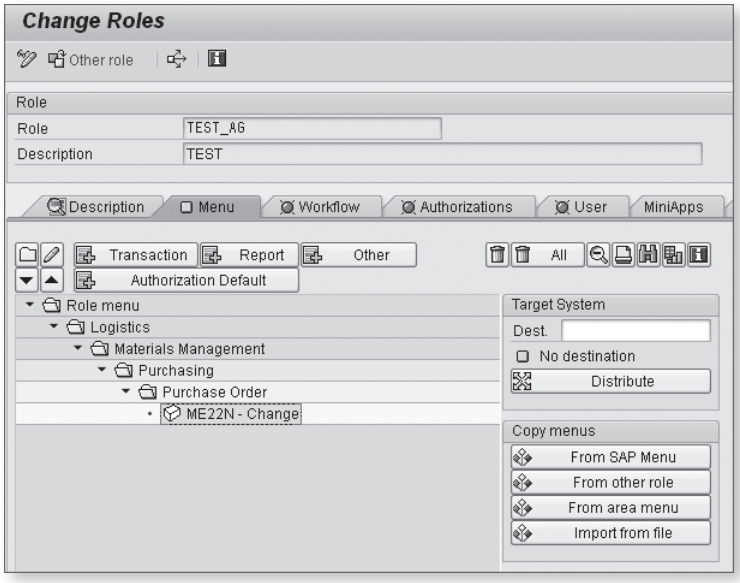


Figure 3 Transaction PFCG Menu Tree after the Import from the Standard Menu Tree

If you try to insert Transaction ME22N again or to insert a different transaction with the same path, the Transaction PFCG menu replicates the entry, and the logistics path is shown twice (see Figure 4). To avoid these duplicated record menus, set the menu inserting option by clicking on Utilities • Settings.

In the Settings: Role Maintenance screen, click on the Menu: Do Not Insert Existing Entries. Standard: NO checkbox. This setting overrides the global setting for this behavior due to the `CONDENSE_MENU_PFCG` switch in Table SSM_CUST. In this way, the path will not be replicated when you import an SAP menu.

Figure 5 shows the result of inserting a new Transaction ME23N menu when an already existing transaction of the same path was already present. The menu path is not replicated as shown in Figure 4.

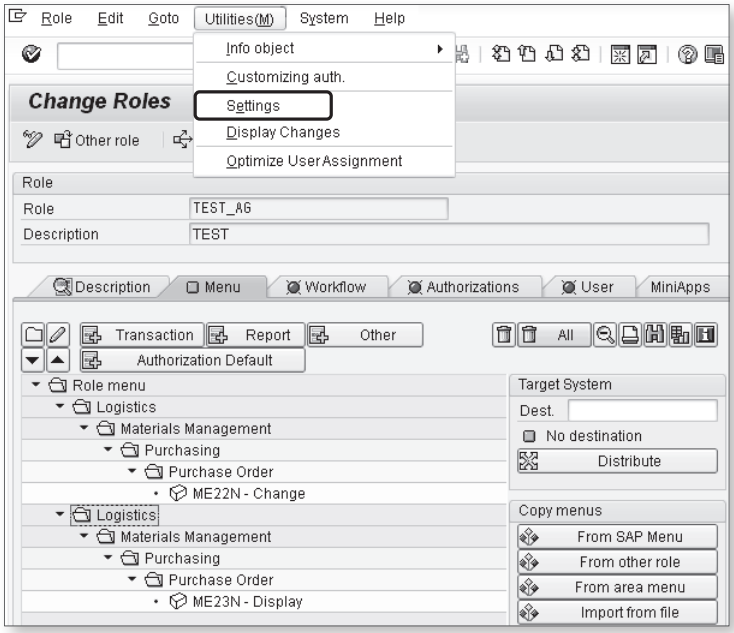


Figure 4 Transaction PFCG Menu Maintenance

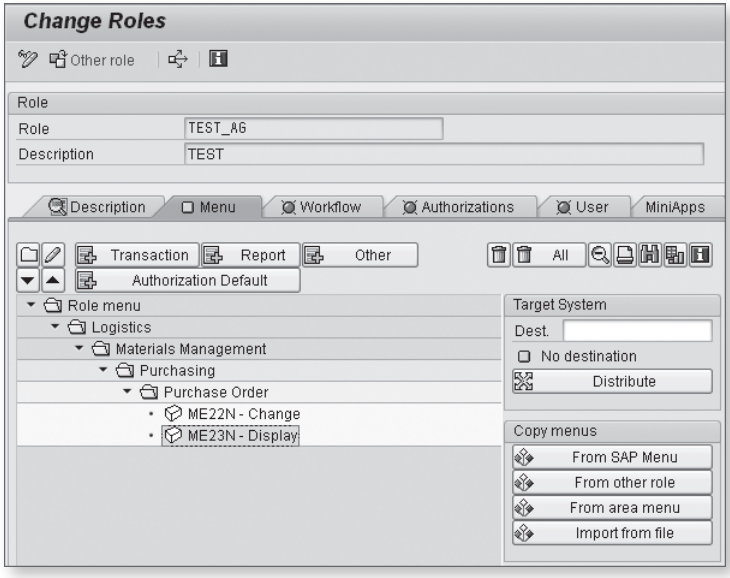


Figure 5 Result after Flagging the Menu Existent Entry Setting

Tip 46

Automatically Populating the Authorization Objects Transaction Link When Performing a Developer Trace

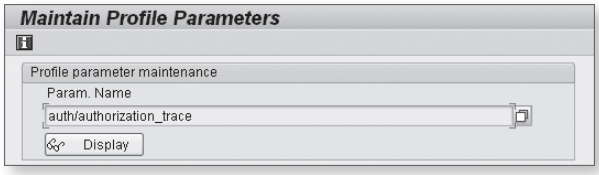
You can automatically link authorization objects to a transaction by using an instance profile.

In a few situations, you'll need to be aware of all authorization objects that are checked from custom transactions. The first is when you define a new custom transaction code that contains one or more authorization objects, which is useful when you're inserting this transaction in a role menu. When you perform this task, you'll find all authorization objects checked by the transaction inserted into the menu in Transaction PFCG's AUTHORIZATION tab.

The second situation is when you're checking custom transactions (mostly for old and not-well documented transactions), where you may not know all authorization objects checked from a custom transaction at first glance. In both of these cases case, you'll use Transaction SU24.

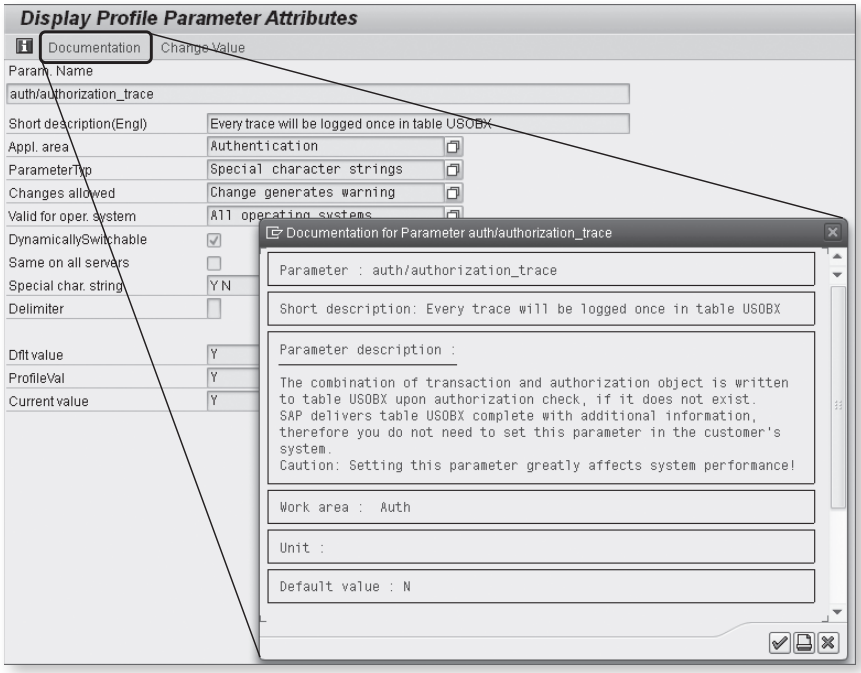
And Here's How ...

You can display all instance parameters documentation through Transaction RZ11, one of which is the authorization trace parameter `auth/authorization_trace` (see Figure 1).



« *Figure 1*
Transaction RZ11 on the auth/authorization_trace Profile Parameter

Click on the Display button, and then click on Documentation in the resulting screen (Figure 2), where you can read the SAP documentation of this profile.



» *Figure 2 Transaction RZ11 Documentation Button*

To set the `auth/authorization_trace` parameter, you have to involve your system administrator. This parameter is different from several others that require a reboot of the system. In this case, when you set the value to "Y" the recording is activated and you can perform your activities immediately.

The differences between Transactions SU24 and SU22 are:

- Transaction SU24 represents the link between transactions and authorization objects that are maintained directly from the customer. The SAP tables involved when you use Transaction SU24 are USOBT_C and USOBX_C (the last character in the table names means customer).
- Transaction SU22 represents the same concept as Transaction SU24, but is only used to see the standard values delivered by SAP. When you enable the authorization trace profile, the authorization objects aren't inserted into Table USOB*_C but only into Table USOBX.

Example

Imagine that you have a custom transaction code named ZBC002, and you want to automatically populate the authorization objects transaction link.

1. Set the AUTH/AUTHORIZATION_TRACE parameter to "Y".
2. Execute and use the custom transaction; in this example Transaction ZBC002. The system will insert it automatically when checking an authorization object.
3. Verify and check Transaction SU22 for Transaction ZBC002.
4. Upload the value in Transaction SU24 for Transaction ZBC002 (see Tip 42).

Figure 3 shows the Transaction SU22 interface. If you press F8 before performing the trace, Transaction SU22 on ZBC002 won't contain any authorization objects (see Figure 4), although there are some authorization check statements in the ABAP program of Transaction ZBC002.

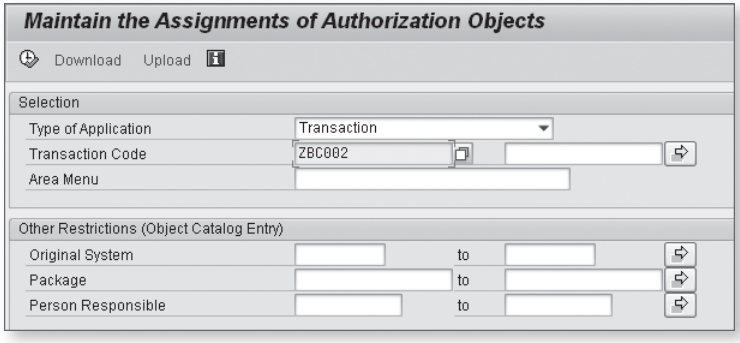
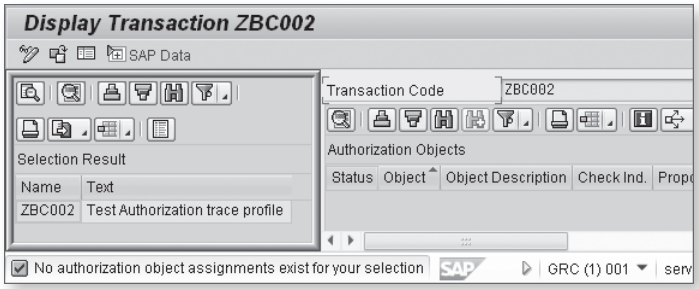
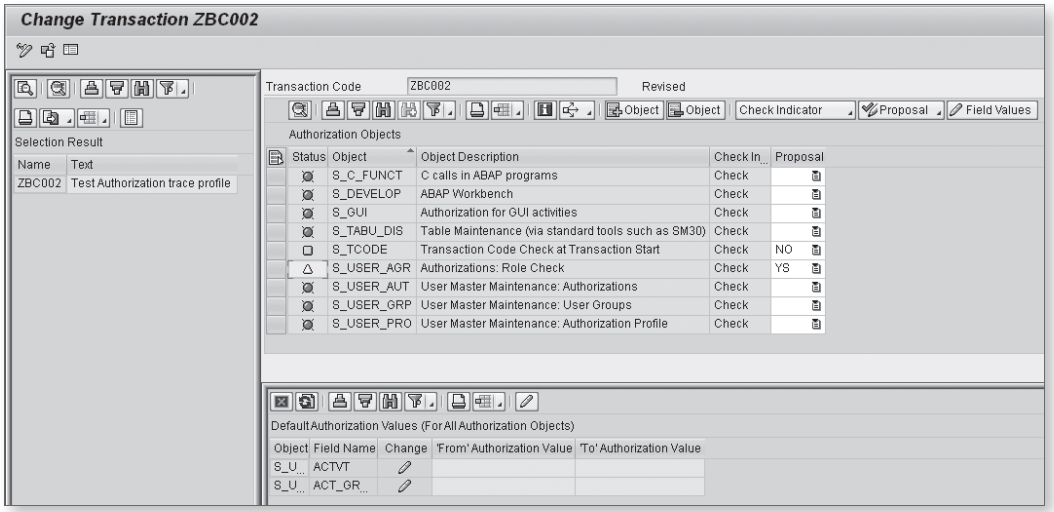


Figure 3 Transaction SU22 Interface: Maintain the Assignments of Authorization Objects



« Figure 4
Transaction ZBC002 in Transaction SU24

After you've executed and tested Transaction ZBC002, you'll see all authorization objects checked in Transaction SU22 of Transaction ZBC002 (see Figure 5). The authorization objects are inserted in Transaction SU22 as is, which means you have to decide how to maintain the transaction.[1]



» Figure 5 Transaction SU22 on Transaction ZBC002 after the Authorization Trace

After you've found the objects linked to a transaction and have adjusted these authorization objects, you can download the values (from Transaction SU22) and upload them in Transaction SU24 (see Tip 42 for upload and download capability in Transactions SU22 and SU24). Alternatively, access Transaction SU24 in change

1 This tip's focus is not on the behavior of Transactions SU24 or SU22; these two functionalities are the heart of Transaction PFCG (Profile Generator) and require a specific explanation outside the scope of the tip.

mode, click on the SAP DATA button, and then click on COPY SAP DATA to synchronize the SAP values just traced to the customer values.

Following are some further remarks about the use of this functionality:

- This kind of trace only works on custom transactions.
- As suggested in the SAP documentation, this parameter can decrease system performance, so you need to use it in a targeted manner.
- This parameter (developer trace) can be useful when you have to find the authorization objects list from a lot of unknown or undocumented custom transactions that are in use.
- You need to know how to test and use these custom transactions. If you only partially use/test the custom transaction, some authorization checks might not be traced.
- This kind of trace (trace only the authorization objects), doesn't insert the authorization objects values that are checked. To do so, you can also use Transaction ST01 (see Tip 44) in addition to this trace. This allows you to populate the authorization objects as well as know the values to enter.

Tip 47

Adjusting Query Maintenance to Avoid Security Problems

You can easily simplify query maintenance and security governance by understanding some specific guidelines and query strategies.

Let's say that a business asks to see the business data and elaborate on it, but the administrators are unwilling to release some capabilities, such as query at the end-user level, mainly due to performance reasons and the difficulty of segregating the data access into an SAP ERP system. This is a common business and administrator pain point.

This tip classifies all common query security and organizational problems and provides an exit strategy to save time and avoid the problem.

And Here's How ...

In an SAP ERP system, the SAP Query tool allows you to design and extract data from the system by directly reading SAP tables. From a security point of view, this can represent a security hole. A user might bypass the transactions allowed and browse the database tables directly (e.g., a user is not authorized to view sales document prices, but this user can directly browse the SAP table that contains this data via SAP Query).

There are some ways to protect these SAP tables from being accessed by users through specific authorization objects such as `S_TABU_DIS` to protect the view or change of a group of tables, `S_TABU_CLI` to protect the maintenance of client-dependent or independent tables, and `S_TABU_NAM` (the latest table authorization objects released from SAP in December 2010) to protect a single table.

However, it's essential to keep in mind that these authorization objects are widely used in SAP transactions, which means that if you decide to manage these authorization objects, you have to invest a lot of time in setting up all segregations, and further time for daily maintenance. For these reasons, consider adopting a query strategy developed from our experience as explained here:

- Evaluate whether the query will be heavily used. In some companies, very few queries are developed; in others, queries are at the front of the workload. Here, it's essential to evaluate if the SAP query in an SAP ERP system is sufficient to satisfy your requirements or if it's necessary to adopt a specific tool such as SAP BusinessObjects or business intelligence solutions. In other words, your company might develop a lot of queries to try to satisfy your requirements but may not be using the tools properly, which can cause an overload of maintenance work.
- If the SAP queries in your SAP ERP system satisfy your company needs, don't allow the definition of the query (Transactions SQ01, SQ02, and SQ03) at the end-user level. Query development should be performed by the IT department or a delegated business user. This is mainly important for two reasons:
  - A poorly designed query can cause performance decay of the system.
  - Validating and sharing the query being developed is important to avoid an excessive growth of duplicated/similar queries and unused queries.
- After the query is developed, it should be linked to a transaction code (see the end of Tip 25 in Part 2). It's best to avoid directly assigning the ability to use Transaction SQ00 for executing the queries at the end-user level. By assigning Transaction SQ00, you have to maintain authorization objects above all authorizations. Defining a transaction code that recalls a query gives you the opportunity to manage the authorization through Transaction SU24 in order to maintain the link transaction and authorization objects necessary to execute it.
- A further method to save time is to avoid the maintenance of query user groups. If you assign Transaction SQ00 with authorization object `S_QUERY`, a user can execute a query of all groups (Financial, HR, Purchasing, etc.). To avoid this situation, deactivate authorization object `S_QUERY`, and maintain query user groups, to assign a group of queries to one or more users. If you manage all queries through a transaction code, you don't need to maintain query user groups.
- From a governance point of view, it's easier to verify and control the transactions a user is allowed to execute rather than investigate the allowed query and data managed at all transactions.

Tip 48

Cleaning Up Unused Batch Jobs

You can delete unused jobs and verify whether all batch jobs are correctly scheduled.

If a job is periodic, it should be scheduled through a specific user ID. Furthermore, when you delete a user ID, you should verify and check whether this user has a periodic job assigned. The potential problem here is that if you delete a user ID that belongs to one or more periodic jobs, you risk introducing an error to the job. Imagine having a job to process and send the salary each month end. For an oversight, this job has been scheduled using a business (not technical) user ID. If you delete this user ID, this job won't process and send the salary.

You can easily check these situations by browsing two standard SAP tables.

And Here's How ...

To view a scheduled job, use Transaction SM37. When you schedule a job, you can decide which user ID to use and the related authorization this job should execute. To define a job, follow these steps:

1. Enter the job name; for example, "ZBCJOB001".
2. Click on the Step icon, and the Create Step 1 pop-up appears.
3. In this pop-up, you can enter the user ID in the User field. All job authorization needs will be checked against this user.
4. In the ABAP Program section, enter the Name of the program to execute as a job (e.g., "RHAUTUPD_NEW").
5. Click on the Check button, and then click on the Save icon.
6. Set up the start condition by clicking on the Start condition button at the top of the screen; for example, to set up whether the job should start immediately,

periodically, and so on. The focus of this example is on the periodic job, so select the PERIODIC JOB checkbox.

7. Click on the SAVE button to save the periodic job just defined. When you save a job, the fields are populated by the following SAP tables:
   - Table TBTCP: Background Job Step Overview
   - Table TBTCS: Background Processing: Time Schedule Table

In some situations, you may have to delete a user. However, first you need to verify that this user hasn't been assigned to a periodic job—if you delete the user, the job won't work.

Follow these steps:

1. Browse Table TBTCP to determine whether a user has a job assigned. Figure 1 shows the selections criteria to find all jobs scheduled with user ID MMANARA in the AUTHCKNAM field. Click on EXECUTE.

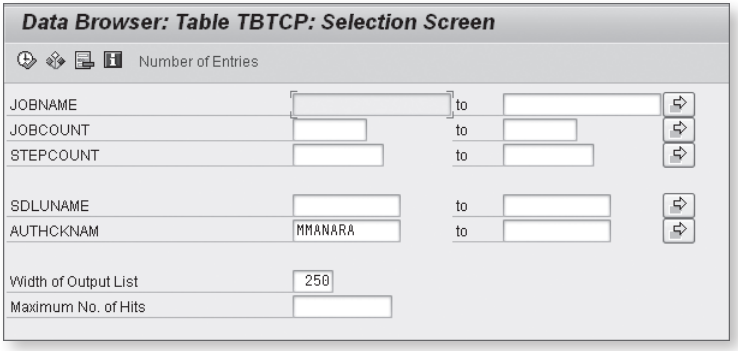


Figure 1 Table TBTCP with AUTHCKNAM Selection Criteria

2. Figure 2 shows the selection criteria. Enter the list of jobs in the JOBNAME field, and set "X" in the PERIODIC field to find which jobs retrieved from the first query on Table TBTCP are periodic. If this query returns some result, then deleting user MMANARA will cause some jobs to be in error.

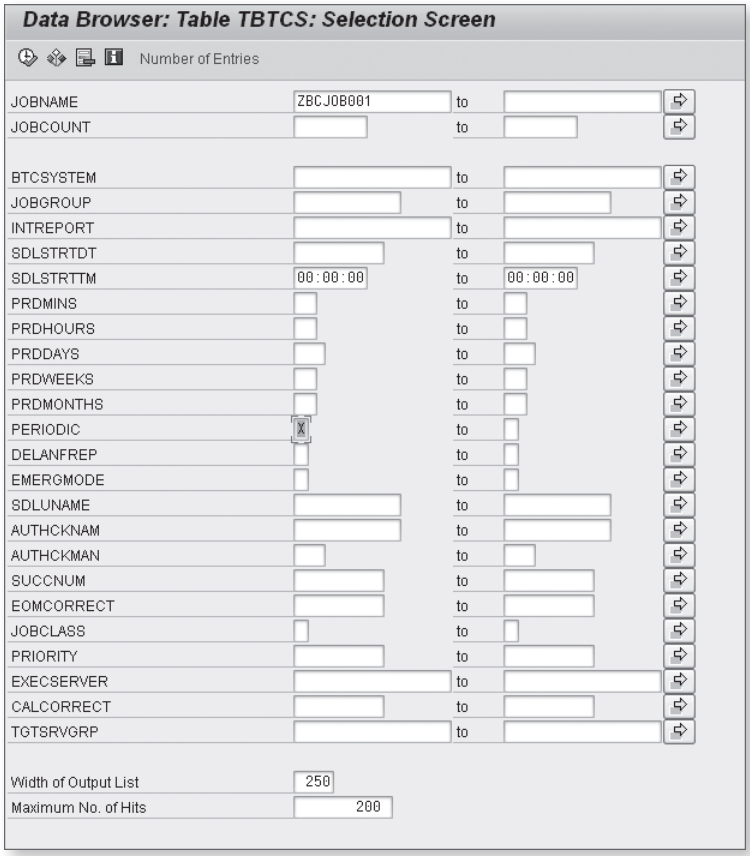


Figure 2 Table TBTCS Selection Criteria to Determine Whether Jobs Are Periodic

In this first step, you found all jobs, not only the periodic ones. To figure out which of these jobs are periodic, you need to get the output (all job names) of this first query and put the result in Table TBTCS.

To maintain the governance of the jobs scheduled, use a specific batch user ID. Using a specific batch user ID allows you to determine whether there are different users from your batch user in the AUTHCKNAM field for the periodic job.

Tip 49

Setting Up Authorizations to Allow Internet Service

You can define authorization roles for web service as well as classical authorization roles.

A company may use external software programs that interface with their SAP system through the network (e.g., the Internet). To allow users to access these HTTP services, you can use the Internet Communication Framework (ICF) service (Transaction SICF). To allow this access, you have to define a user ID and give the proper authorizations for the external program. A common approach is to define an authorization role without exploiting the authorization features of Transaction SICF. Consequently, many people waste a lot of time chasing the missing authorization problems.

And Here's How ...

Through Transaction SICF, you can view all enabled or disabled HTTP services. To see all services, press F8; otherwise, if you know the service name fill in the SERVICE NAME field.

You can now see the services tree by pressing F8 (EXECUTE button). If you explode the tree, you can see all of the service names. Figure 1 shows the exploded tree.

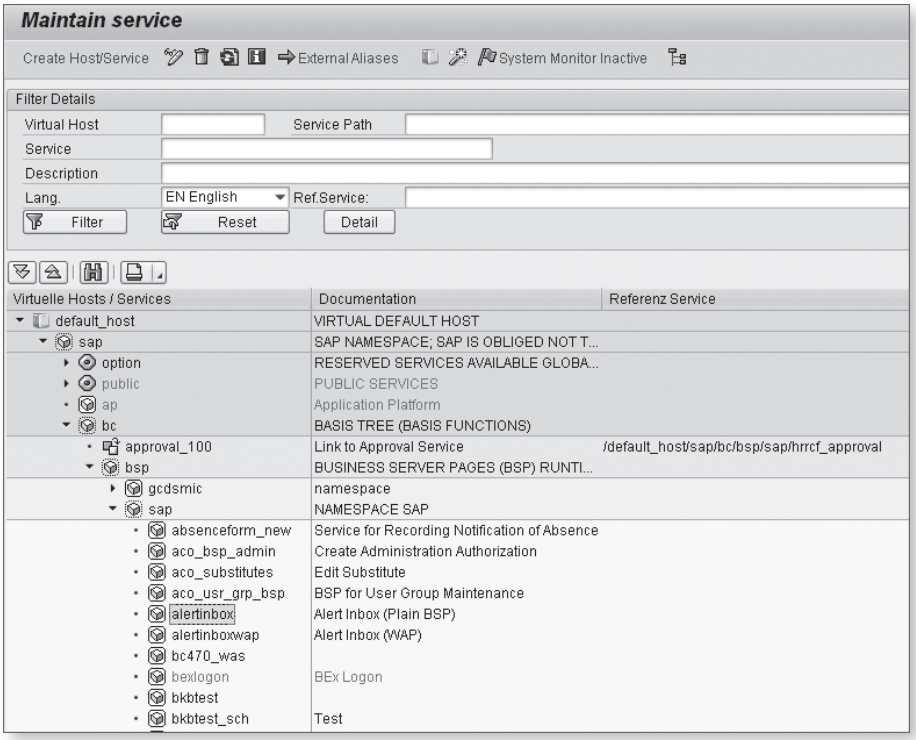
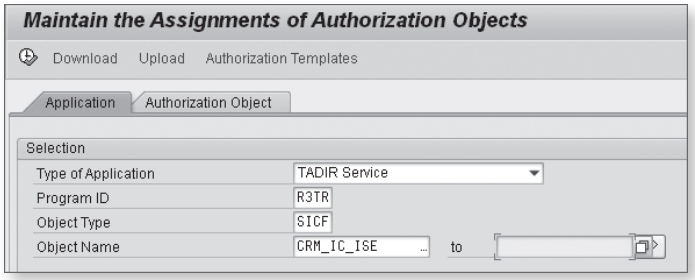


Figure 1 Transaction SICF Services Tree

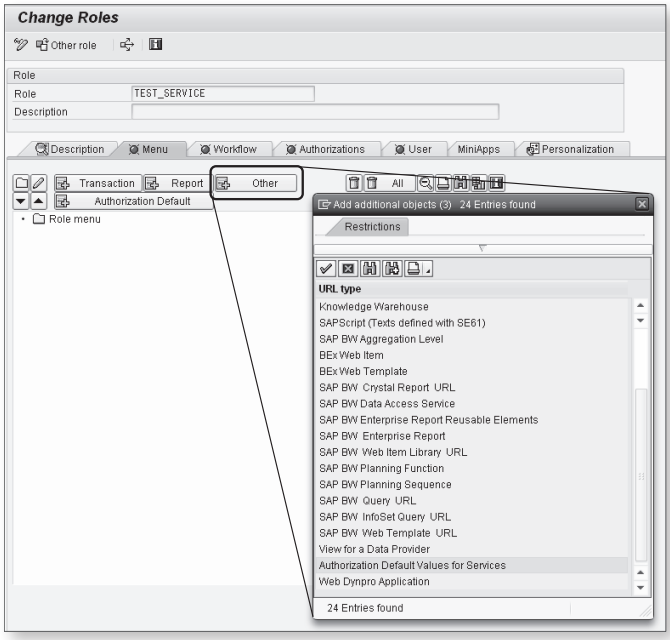
To define and design an authorization role to allow the execution of the HTTP service, select the service by highlighting it, and then click on EDIT • AUTHORIZATION PROFILE. This brings you to the screen where you can maintain all authorization objects that this service needs to execute by clicking on the CHECK INDICATOR button.

The maintenance of these authorization objects is the same as Transaction SU24 as shown in Figure 2. Select TADIR SERVICE in the TYPE OF APPLICATION field, and then enter the service program name and object type/name. You'll find this data in Table USOBHASH.



« *Figure 2 Maintain Service Authorization Objects Assignments*

After maintaining the service authorization objects assignments, you can enter it into a role. Access Transaction PFCG, go to the MENU tab, and enter the service name as shown in Figure 3. Create a role, click on the OTHER menu button, and choose AUTHORIZATION DEFAULT VALUES FOR SERVICES.



» *Figure 3 Define a Role with a Service in the Menu*

In the screen that appears you can enter the service details. At the end of this process, you can go to the AUTHORIZATIONS tab and generate the role just created.

If you maintain and design the role for SAP services in this way, these kinds of roles will be managed by the security administrator as well as all other type of roles, without any exception.

Tip 50

Avoiding Security Holes during SAP Menu Role Maintenance

You can create your role menu by using the SAP standard menu tree, but you need to be aware of potential missing entries that could open a breach in your roles.

When you define a new role that contains transaction codes, you can decide to enter these transactions by choosing them from the SAP standard menu. When you decide to use this functionality, you need to avoid setting the `S_TCODE` equal to an asterisk, which will give the user full authorization. Asterisks on `S_TCODE` objects give the user that receives this role the opportunity to potentially start any transaction code in total contrast with any Segregation of Duties (SoD) policy. This tip explains when this happens and how to avoid this situation.

And Here's How ...

As shown in Figure 1, when you are in the Transaction PFCG MENU tab you can insert a transaction or a menu tree by clicking on the FROM SAP MENU button and then clicking on the transaction to insert.

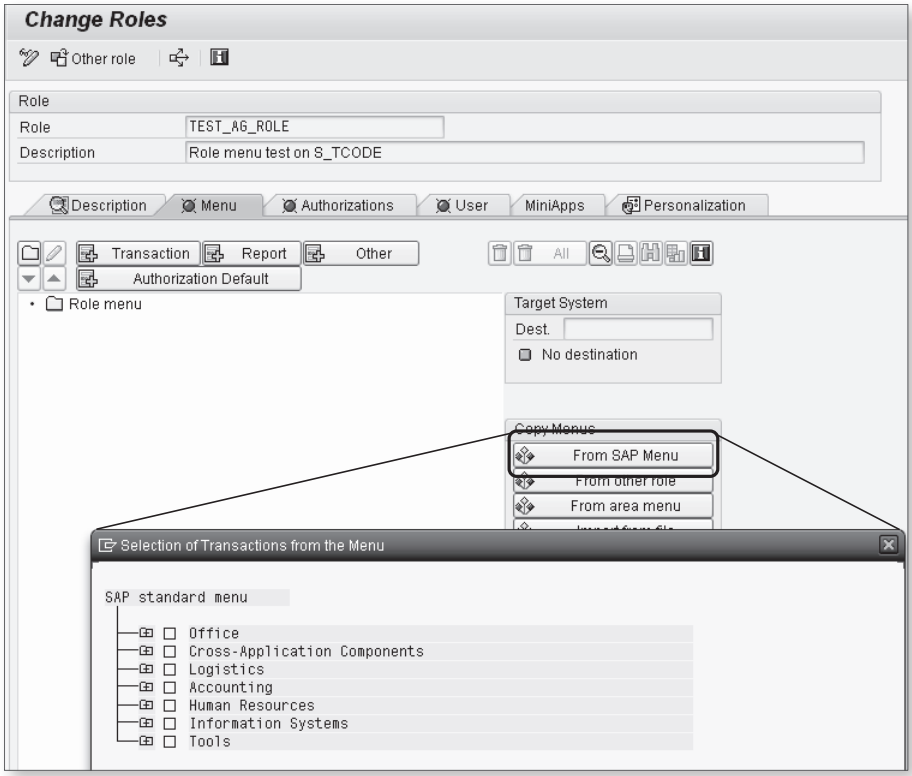


Figure 1 Insert a Transaction or a Menu Tree in the Transaction PFCG Role Menu Tab

When you perform this activity, you have to check whether the SAP menu contains a dummy transaction entry. This dummy entry could cause the system to insert an asterisk in the S_TCODE during the maintenance of the role in the Authorizations tab, giving a user full authorization. A dummy entry is shown in Figure 2, under the folder Bill of Exchange/Postdated Checks Enhancement for the transaction texts Bill of Exchange / Postdated Check Transactions and Reversal of Bill of Exchange / Postdated Check Transactions. You can tell that an entry is a dummy entry because there's a missing entry in the role menu, and the S_TCODE object in the Authorizations tab in the role is open.

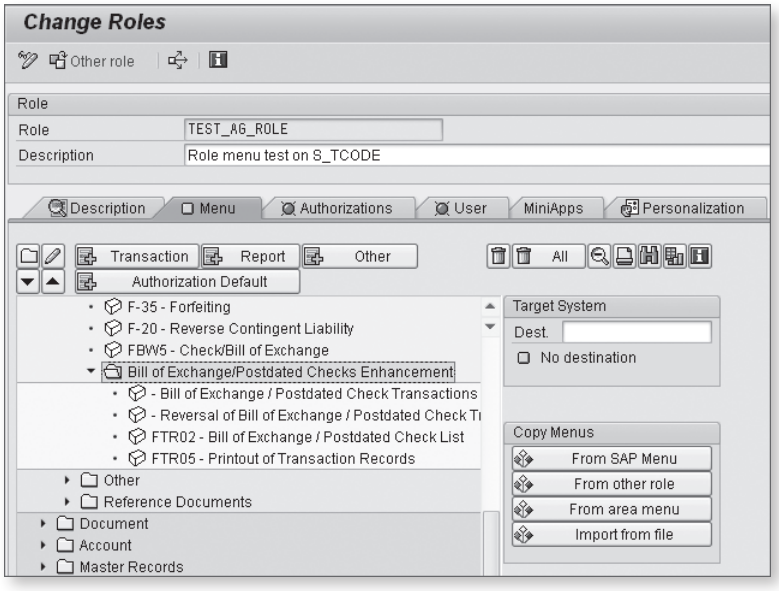


Figure 2 Dummy Transaction Entry in the SAP Role Menu

When you go through the AUTHORIZATIONS tab, the authorization object S_TCODE is open. If you click on the traffic light in Figure 3 to set the asterisk value on all authorization objects in the open status, you can risk transferring the asterisk to the S_TCODE object.

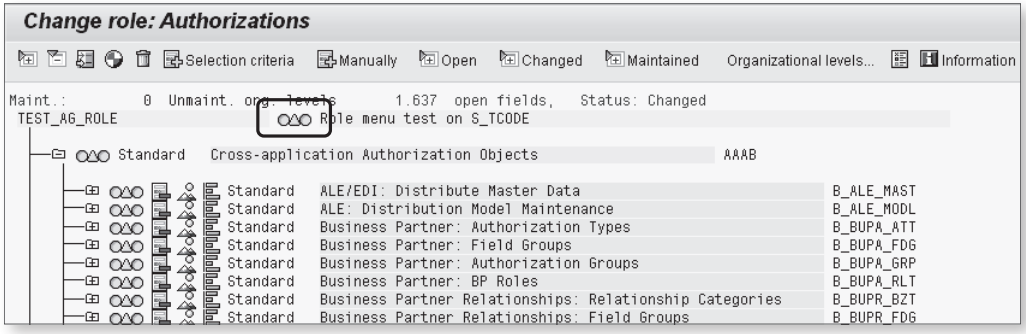


Figure 3 Traffic Light in the Role Authorizations Tab

When you use the SAP menu, you can easily find these situations by browsing Table AGR_HIER.

Perform a query (through Transaction SE16) on this table (shown in Figure 4). To look for a role that has dummy entries in the menu and a missing transaction code, enter the role name in the AGR_NAME field. Then enter the value "TR" in the REPORTTYPE field. The REPORT field should be equal to null.

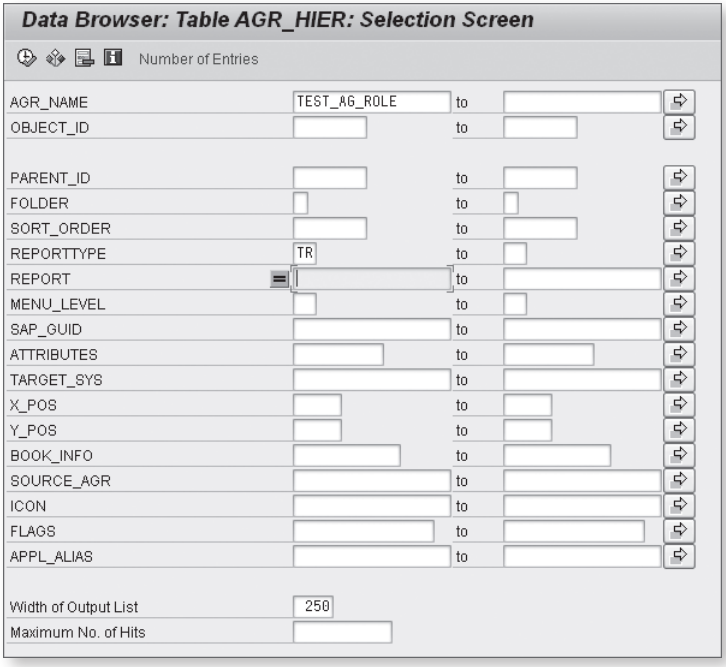


Figure 4 Finding a Missing Transaction in the Role Menu Entry

After you've identified the missing entry (in other words the transaction text without the transaction code, e.g., "Bill of Exchange / Postdated Check Transactions"), you can use the binoculars icon (top right in Figure 2) in the role menu to find the exact location of this dummy entry in the menu tree and then manually delete the entry.

Tip 51

Changing the Rules to Generate Profile Names

You can customize the naming conventions for authorization profiles to avoid inconsistencies and duplicate names.

Inconsistencies can occur when you move a role from a development system to a production system by using the transport management system due to two factors: the standard generated naming of an authorization profile and role maintenance in a system (quality or directly in production) that is opposed to the development system.

Because the authorization profile naming depends on the system name, if the development system has the same ID as that of a production system, a profile override can happen during the transport. In most cases you can't change the system's ID, but you can bypass this by changing the profile naming rule generation.

This tip is useful when the following factors are present:

- Your system landscapes (e.g., development system and quality system) have the same ID.
- You've defined several roles directly in the production system.
- Your profile incremental numbers are different from the system's numbers.
- You want to reestablish the correct flow of role change requests using SAP's transport management system, so you create the roles in the development system and then move them to production.

And Here's How ...

Suppose your development system has an identifier of PS8, your quality system has an identifier of PQ8, and the production system has an identifier of PP8.

The generated authorization profile naming system is formed by a T-string—the first character of the ID system, the last character of the ID, and then an incremental number (e.g., T-P8000000089). If you create a role with this profile name that has been generated (ending with 89), and then import it into another system (e.g., if you create another role with different transactions from the previous role in the quality system, the incremental profile number may be the same). When you move the role created in the development system to the quality system, an overriding profile situation can arise.

To avoid this situation, you can adopt one of two solutions.

Define Different Number Ranges for Each Client

You can do this by setting the start number in Table AGR_NUM_2 with Transactions SM30 or SE16. Alternatively, you can use the standard Program PFCG_SET_PROFILE_NAMERANGE to set the start number by executing Transaction SE38. For example, in system PS8, the incremental number will start with 100xxx; in system PQ8, it will start with 200xxx; and in system PP8, it will start with 300xxx. In this way, you can ensure that during role number creation, you create a gap between the SAP landscapes.

Define a Client-Dependent Key

You can set up this key by using Table USR_CUST with the parameter `PRGN_PROF_PREFIX`. You can customize this table through Transaction SM30. The value of this parameter replaces the ID that is generated automatically based on the ID of the system where the profile is created. For example, if your system ID name is PQ8, the standard profile name will start with T-P8xxx (where xxx represents the incremental number). If you enter "PZ" in Table USR_CUST, your profile name generated will be T-PZxxx instead of T-P8xxx.

In each of these solutions you have to make sure there are no changes to these tables during the system copy; otherwise, the custom profile naming setting defined in a system will be replicated in another, overriding the values.

Tip 52

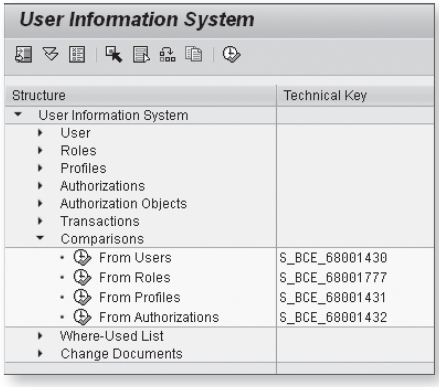
Comparing Authorization Roles to Check for Alignment Between Systems

You can easily compare two roles in different systems and check for alignment by using Transaction SUIM.

Comparing authorization roles is not a simple task if these roles contain several authorizations. Sometimes you may need to compare the same role if they have different values in the development system and quality system. This misalignment by the system is not in compliance using the transport management system. Through Transaction SUIM, you can easily compare two roles in a system or cross system.

And Here's How ...

Access Transaction SUIM to display the COMPARISONS tree shown in Figure 1.



« *Figure 1 Transaction SUIM Comparisons Tree*

Click on From Roles S_BCE_68001777. The screen shown in Figure 2 appears. Enter the role names you want to compare in the Role A and Role B fields.

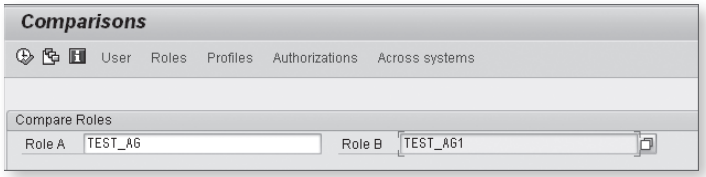


Figure 2 Role Comparison

Specify the system where you want the comparison performed by clicking on the Across systems button and then choosing the system ID name from the system list. For example, you can compare a role in the development system and a role in the production system. Note that to perform this functionality correctly, it's essential that the system's remote function call (RFC) destinations are defined.

The result of the comparison is shown in Figure 3. The Comparison column can display three different colors: Red means that different authorization objects exist between the roles, yellow indicates that the different roles have the same authorization objects but different values, and green indicates that the roles have the same authorization objects and same values. If the comparison is all green, that means the roles are equal. If you're seeing red or yellow, you can use Transaction PFCG to correct it.

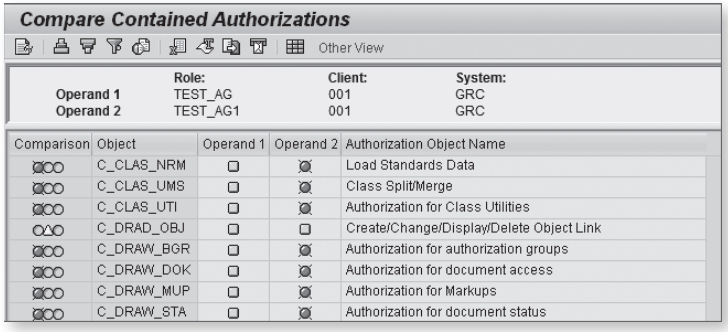


Figure 3 Comparison Result Example

Unfortunately, this transaction doesn't support a mass comparison between several roles at one time. However, you can go to *www.sdn.sap.com*, where you can find information on an enhancement that works around this limitation.

Tip 53

Replacing the Parent Role of a Derived Role en Masse

When you need to relink previously derived roles to a new father role, it's easy to set up a procedure to do this en masse.

Suppose you have a father role with several thousands of derived roles. You want to change the technical role name of this father role, but the authorization data should remain the same.

To accomplish this, you have to detach the relationship between all of the derived roles and the father role, and then relink these derived roles to the new father roles. Although this sounds like an error-prone and time-consuming process, there's a way to automate this procedure to avoid performing several manual tasks.

And Here's How ...

This solution shows just one example that will serve as your basis in all other situations. The example is formed by four roles:

- **ROLE_TEMPLATE:** The previously existing father role
- **ROLE_TEMPLATE_NEW:** The new father role
- **ROLE_DERIVED1:** Derived role
- **ROLE_DERIVED2:** Derived role

Note that the last two roles are the derived roles that you will unlink from ROLE_TEMPLATE and relink to ROLE_TEMPLATE_NEW.

Your first step is to go to Transaction PFCG and enter the edit mode on the first derived role—ROLE_DERIVED1—and then click on the Delete Inheritance Relationships button (see Figure 1).

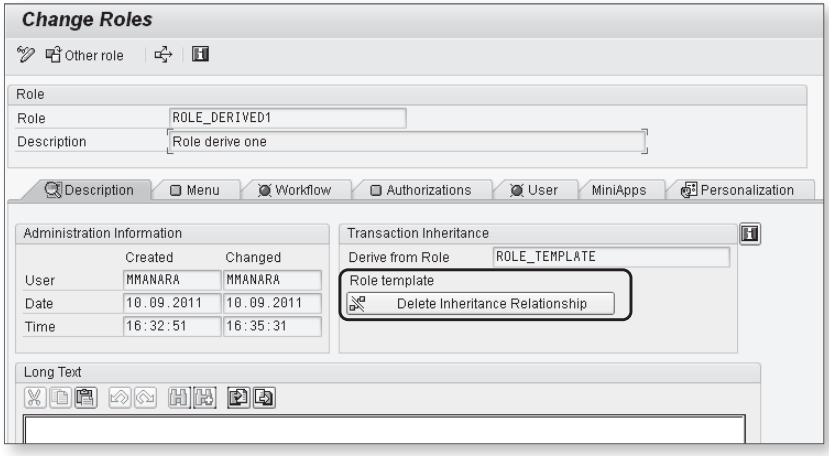


Figure 1 Delete Inheritance Relationships Button on a Derived Role

Click OK on the confirmation pop-up, which allows you to remove the relationship of this role from the father role ROLE_TEMPLATE. You'll now see a screen indicating that you've removed the relationships but cannot enter a new role father (see Figure 2).

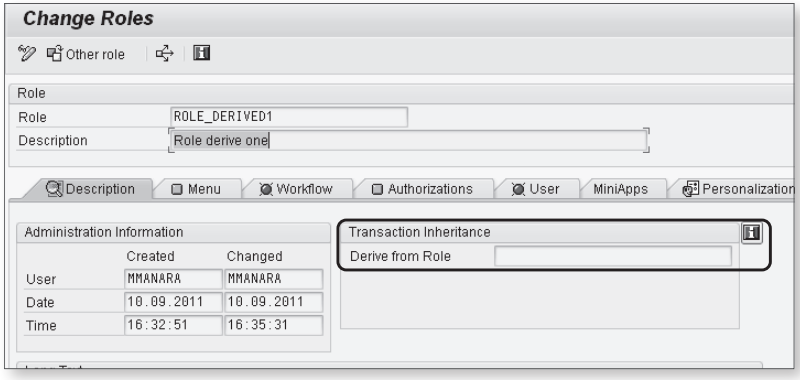


Figure 2 Derive from Role Is Not Editable

To be able to enter a new father role, first navigate to the Menu tab and remove the relationship menu entry. Delete all menu entries in the Menu tab by clicking on the Delete All button (see Figure 3).

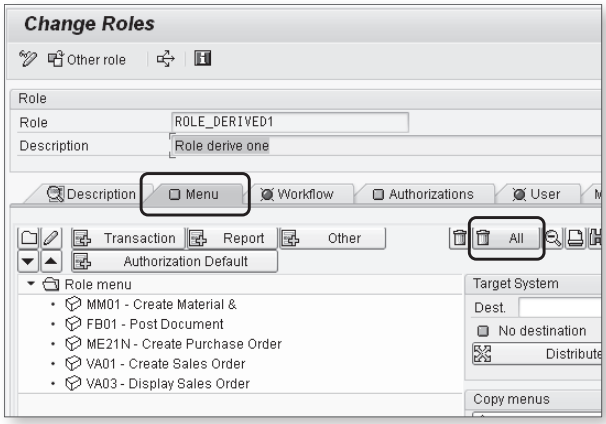


Figure 3 The Delete All Button in the Menu Tab

Save the role and go back to the Transaction PFCG interface. When you go into edit mode on the role ROLE_DERIVED1 now, you can manually enter the new father role, in our example, ROLE_TEMPLATE_NEW for ROLE_DERIVED1 as shown in Figure 4.

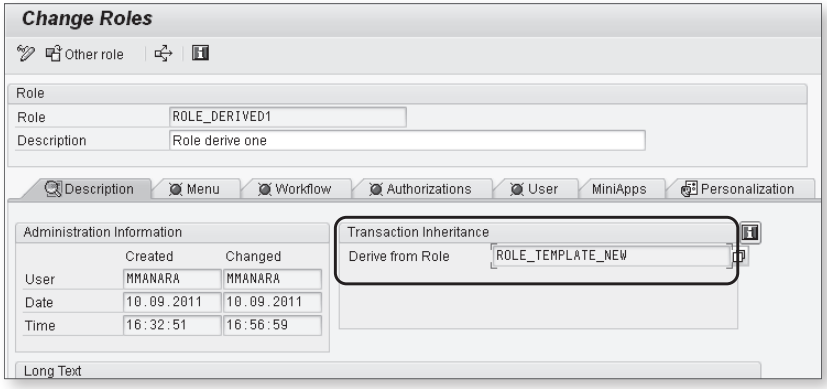


Figure 4 The Derive from Role Box Is Now Editable

To perform a mass change to a father role, you can create a Computer Aided Test Tool test script (through Transactions SECATT or SCAT), to perform these steps on all derived roles involved.

Tip 54

Generating Large Quantities of Profiles for Roles in a Single Transaction

When you need to generate a huge number of roles, you can use Transaction SUPC to avoid working with these roles manually.

Every time an authorization role is created, it's only useful if the respective profile has been generated as well. However, when you have many thousands of roles to generate, it's impossible to generate the profiles manually. Let's say you've created a new security concept based on many new roles, and you need to transport them to the production system. If you create a change request (CR) for all roles that include the generated profiles, you run the risk of timeout errors and disk space problems during the export phase of the release of the CR.

As a workaround, you can transport all roles without profiles and mass-generate the profile after the import. This tip goes into deeper detail of the transaction you need to use to accomplish this: Transaction SUPC (Mass Generation of Role Profiles).

And Here's How ...

When you launch Transaction SUPC you'll see the screen shown in Figure 1. In the Which roles do you want to output? section, define the selection for the roles you want to mass output.

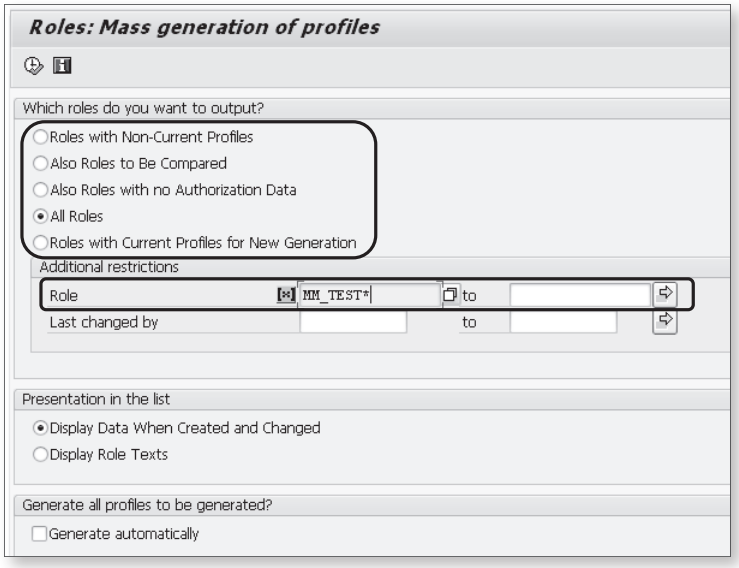


Figure 1 Transaction SUPC Initial Screen

In this example, the ALL ROLES radio button is selected, and the restriction to look for all roles starting with MM_TEST has been added. Figure 2 shows the result after executing the transaction by pressing F8.

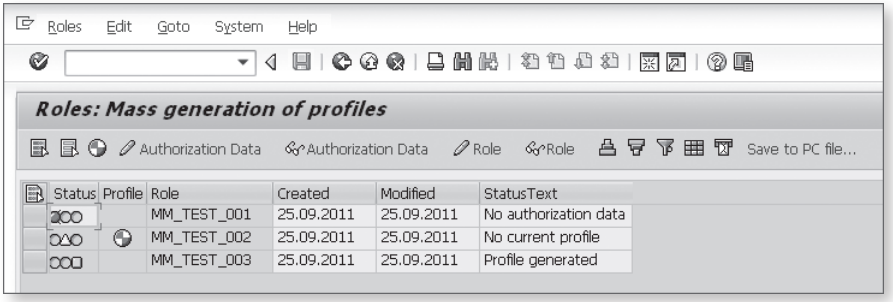


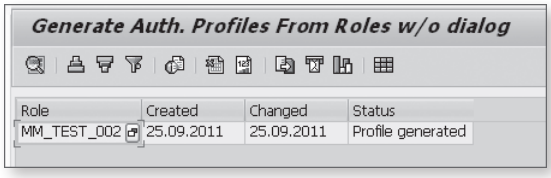
Figure 2 Transaction SUPC Role Selection

There are three potentially different situations, as you can see from the STATUSTEXT column:

- NO AUTHORIZATION DATA
  The corresponding role is available in the system, but the authorizations are not yet defined.

- No current profile
  In the second case, the authorizations have been defined, but the generated profile has never been generated, or an authorization value has been updated afterward without regenerating the profile. These are the only roles that will have profiles generated for them in this tip.
- Profile generated
  The third role is related to a role that is technically correct.

Now, highlight and select all of the roles involved in the generation (in this case, all of the roles shown in Figure 2), and click on the Generate button ( ) or press F8. The final result is shown in Figure 3.



« *Figure 3 Mass Generation Result*

Only roles with the No Current Profile status will be generated.

In the example, the first role (MM_TEST_001) cannot be generated because the authorization values are not yet defined. The third role (MM_TEST_003) will not be generated because it's correct and valid.

Tip 55

Using SAP BAPIs to Manage Roles with an External Program

You can reduce the implementation effort by creating roles through standard SAP BAPIs.

A classic project approach is composed of the analysis, implementation, test, and go-live phases. The go-live date is (normally) a date that cannot be postponed. During the analysis, many business requirements arise, and often not all answers are delivered in a timely manner. For this reason, the implementation and test phases will not be long enough. To help speed up the process, if the analysis phase delivers a clear input for the implementation phase by using BAPIs to create the roles defined in the analysis phase in the system, this can dramatically reduce the effort of this phase.

And Here's How ...

Suppose you have to implement all security elements (users and roles) for a new SAP system. You'll have to create many users (refer to Part 1, Tip 4), and many simple roles and composite roles need to be linked to the users.

First, find the possible BAPIs related to Transaction PFCG (Roles Maintenance). You can look for all BAPIs related to the groups starting with "PRGN" (which stands for "Profile Generator") by using the match code (F4) in Transaction SE37 (ABAP Function Modules).

Note that not all BAPIs can be called by external program; only the ones that have the Remote-Enable Module flag active in the Processing type frame can be called. Figure 1 shows this setting along with the attributes for the BAPI you can use to create a new role (simple or composite). To display BAPI attributes, enter the function name and press F7.

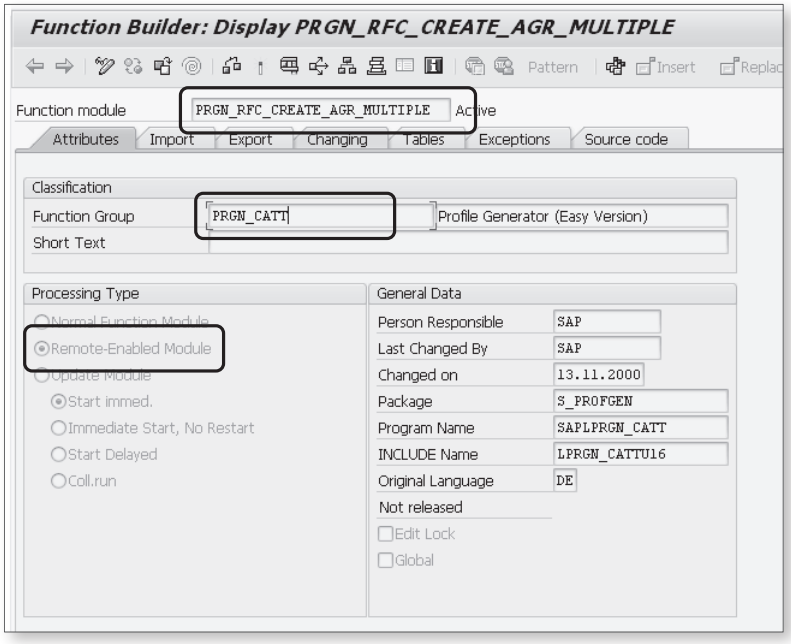


Figure 1 Transaction SE37

Calling a BAPI through Transaction SE37 involves three simple steps:

1. Set the import data.
2. Call the BAPI.
3. Check the output data.

You should test the correct BAPI to familiarize yourself with the input data.

Create a New Role Manually

Suppose you want to create a new role. As shown in Figure 2, you can test the BAPI by clicking on the TEST/EXECUTE button (or pressing F8). Figure 2 shows where you have to specify the input data to create the new role. You can enter the data in the VALUE box and then launch the BAPI.

If every input field has been filled correctly, you'll see a result screen with a valid runtime value (in microseconds). You should check that everything is working properly by accessing Transaction PFCG. As shown in Figure 3, you can verify that the new role has been created.

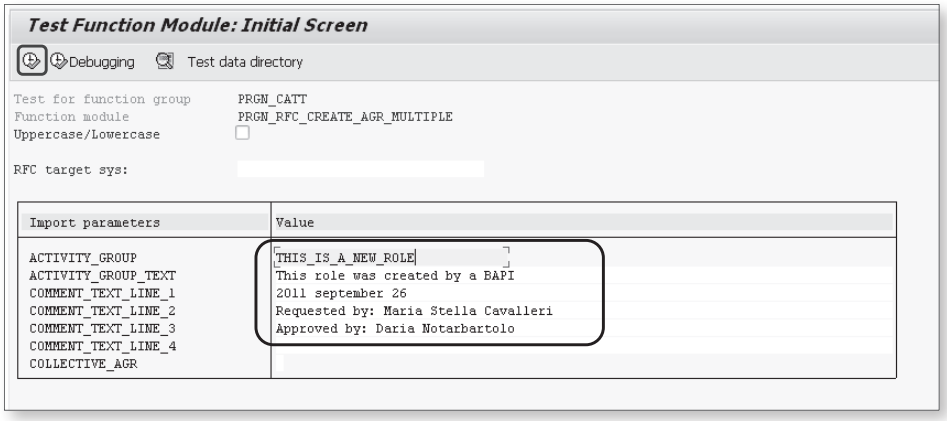


Figure 2 Transaction SE37: BAPI Execution

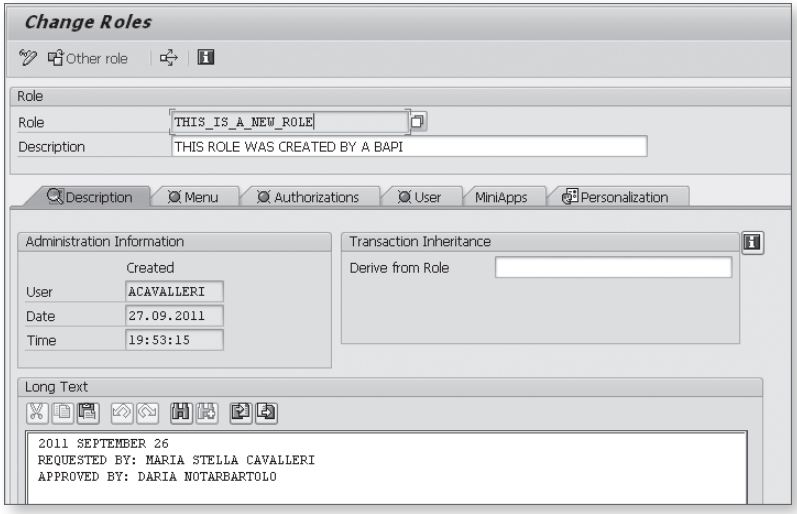


Figure 3 Display the New Role Created in Transaction PFCG

Mass-Manage Roles

Calling a BAPI and inputting the data can also be done (if the BAPI can be called via RFC) with external programs.

Imagine you want to create a Microsoft Access database that contains all information for the roles. You can write a small piece of Visual Basic source code to easily call the BAPI and create the 50,000 roles you need:


```
'*****************************************
'* Function RoleCreate
'*****************************************
Function RoleCreate(ROLE As String, ROLE_DESCRIPTION As String, ROLE_
COMPOSITE As String) As Boolean

    Dim rfcFunction As Object
    Dim paramActivityGroup As Object
    Dim paramActivityGroupDescription As Object
    Dim paramTableActivityGroups As Object
    Dim paramActivityCollectiveAgr As Object

    RoleCreate = False
    If connection_open Then
        ROLE = UCase(ROLE)
        Set rfcFunction = sapConnection.Add("PRGN_RFC_CREATE_AGR_
MULTIPLE")
        '***********************************
        ' paramActivityGroup
        '***********************************
        Set paramActivityGroup = rfcFunction.Exports("ACTIVITY_GROUP")
        paramActivityGroup.VALUE = ROLE
        '***********************************
        ' paramActivityGroupDescription
        '***********************************
        Set paramActivityGroupDescription = rfcFunction.
Exports("ACTIVITY_GROUP_TEXT")
        paramActivityGroupDescription.VALUE = ROLE_DESCRIPTION
        '***********************************
        ' paramActivityCollectiveAgr
        '***********************************
        Set paramActivityCollectiveAgr = rfcFunction.
Exports("COLLECTIVE_AGR")
        paramActivityCollectiveAgr.VALUE = ROLE_COMPOSITE
        If rfcFunction.Call = False Then
            Exit Function
        Else
            RoleCreate = True
        End If
    End If
End Function
```

Tip 56

Using Manual Composite Profiles to Bypass the Profile Technical Limit of 312

The SAP system has a technical limit to assign a certain number of profiles to a user. You can bypass this limit using the composite profile type.

There are technical constraints you must adhere to when designing a role in the analysis phase. The most critical of these restricts each user to 312 profiles. Unfortunately, there is no system parameter you can use to avoid this limit; your option is to use the composite profile type to work around this limit.

All security managers should read OSS Note 841612: Maximum Number of Profiles for Each User. This note describes how to increase the maximum number of profiles for each user from 300 to 312. This tip goes into deeper detail to describe some ways to work around this SAP limitation.

And Here's How ...

Every time you create an authorization role, you'll have to generate the corresponding profile(s) to make it usable. Then, as shown in Figure 1, you will assign the role to a user.

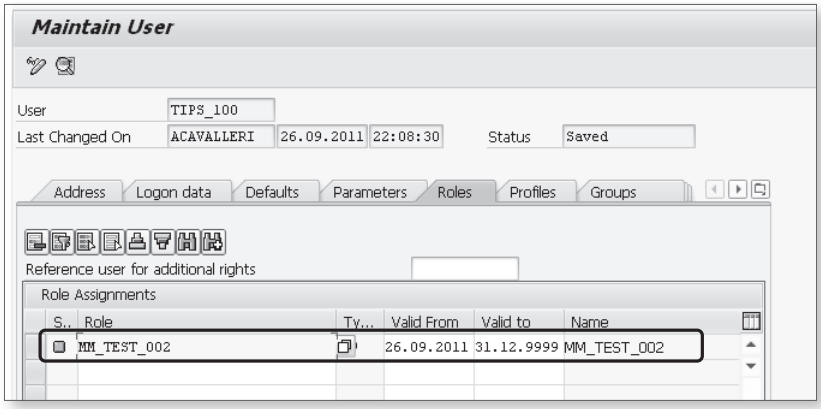


Figure 1 User Master Record: Role Assignment

After the "comparison" action available in Transaction PFCG, the profile generated from the role will be linked to the user (see Figure 2).

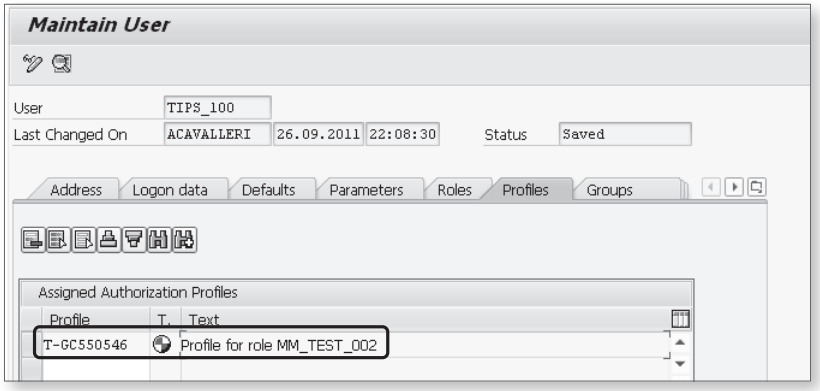


Figure 2 User Master Record: Profile Assignment

The relationship between a user ID and authorization profiles is of type one to many (1:*n*, meaning one user can have many profiles assigned). From a technical point of view, you expect (behind the scene) that there's a table in which there are many records that show the number of profiles assigned to the user. Unfortunately there is only one record in Table USR04, and the profiles are all concatenated in the PROFS field. If you divide the length of field PROFS by the length of a single profile, you'll come up with the number 312.

The most critical part of the roles design is the creation of the simple roles. There are two opposite strategies. The first is to create a few simple roles with many transactions. The second is to create many smaller roles so your library will be much more modulate. The 312 profiles limit will influence the simple roles creation strategy. Without this constraint, having many little roles is better because after the first setup of authorization roles, most of the maintenance will be at the composite roles level.

The Segregation of Duties (SoD) requirement is also involved in the authorization roles strategies. If you need to split a simple role because there is an SoD conflict, the effort could be relevant. When you have a simple roles library with many roles and each having a few transactions, the probability of split roles will be reduced dramatically. For maintenance reasons, the goal is to add or remove simple roles from composite roles instead of adjusting the simple roles.

From our experience, we decided to adopt the strategy of having many simple roles with few (but homogeneous) transactions inside.

Let's make an example of the 312 limit. Suppose that you've created a job role called "Buyer." Because a job role is a composite role, it will contain simple roles. Imagine now that this job role is composed of 40 simple roles. Then, if you have 30 purchasing groups (one for each buyer), you'll need to create 30 composite roles (one for each buyer). Theoretically, each "buyer" composite role will contain 40 derived roles to grant the correct purchasing group.

In many companies, there are some persons called "responsible buyers" who are supervisors of a set of purchasing groups (let's say 10). This means that they should have the summary of the authorizations related to their set of purchasing groups. Because you haven't yet created a specific composite role for each purchasing group, if you assign to a supervisor all composite roles of his set, he'll have the requested authorizations. If you assign to a single user 10 composite roles, and each composite role is composed of 40 simple roles, the total amount of roles (and generated profiles) will be 400 (10 * 40), which is over the 312 limit.

Grouping Derived Roles Values and Entering Them in a Composite Role

In this case, instead of assigning 10 composite roles to the supervisor, you can create an additional composite role in which each derived role contains all involved purchasing groups. This action will shrink the number of profiles assigned to the supervisor.

One other classical situation is related to the key users. In many companies, a key user must have all composite roles of his area. Again, the 312 profile limit will be exceeded. From an SoD point of view, it makes no sense for the key user to have all composite roles of his area, so the solution is simply to deny providing the roles to the user.

Form the technical point of view, it makes sense to create an additional composite role, which contains the really necessary simple roles for the key users. In other words, this means to create a specific job role for the key user.

Sometimes the assertion that the role's design is wrong makes sense. In our experience, the composite roles contain too many simple roles because the business asked for them. With a good analysis, you'll be able to remove many simple roles that contain transactions that are never used.

Create a Manual Composite Profile

As a workaround, you can manage the 312 profile limit with a simple trick, although this approach isn't recommended for the following reasons:

- An SAP best practice is to use only the "roles concept" thorough Transaction PFCG (Role Maintenance), and Transaction SU02 usage is no longer recommended.
- The generated profile is only a part of the role concept. The role's menu is helpful but will be lost using manual composite profiles.

As shown in Figure 3, you can create a manual composite profile with Transaction SU02 (Maintain Authorization Profiles) as a container of generated profiles (from simple roles). Follow these steps:

1. Execute Transaction SU02, enter your profile name, and press Enter.
2. Press F6 to decide whether the profile should be single or composite. In this example, the profile is composite (Figure 3).
3. Enter the single profile in this composite profile (see Figure 4).

In this case, the field PROFL of Table USR04 will contain only one profile. This method has created a sort of composite role by using the manual composite profile.

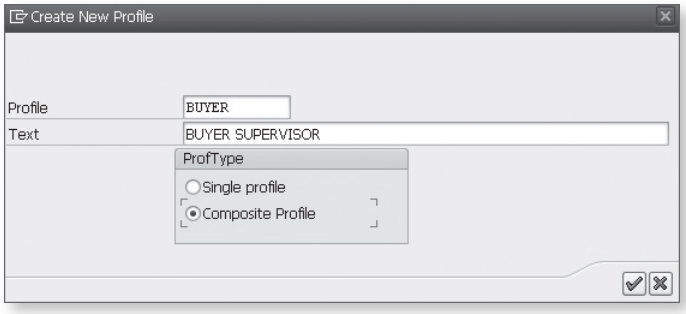


Figure 3 Transaction SU02 Manual Composite Profile

Figure 4 shows a generated profile from Transaction PFCG inserted in the manual composite profile.

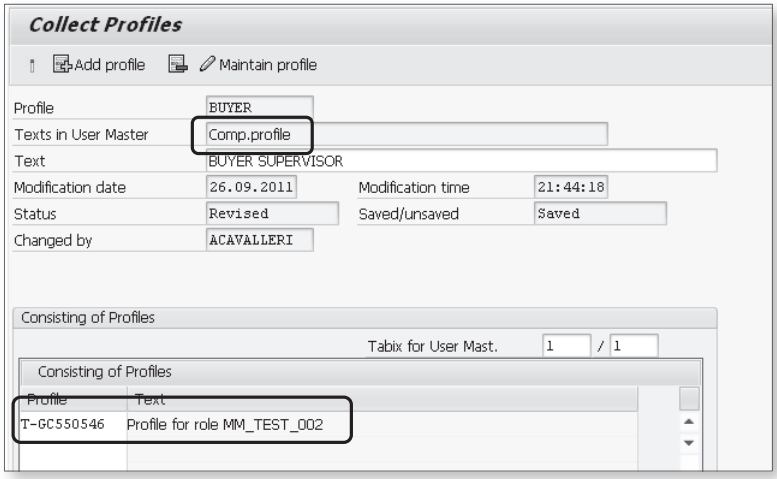


Figure 4 Generated Profile in a Manual Composite Profile

Tip 57

Using Parameter IDs and Customizing Transactions to Manage Authorizations

You can solve authorization problems by employing customizing transactions rather than authorization objects to solve a business requirement.

Not all security aspects can be managed through roles, profiles, and authorizations. In SAP, there are many transactions that restrict a user's possibilities, rather than using the ABAP statement `AUTHORITY-CHECK` (meaning a check over an authorization object; for example, customizing Transaction OMET).

This tip describes other ways to manage authorizations.

And Here's How ...

Imagine that you always want a purchase request reference linked whenever a user creates a new purchase order. You'll find that classic authorizations won't provide you with the means to make this scenario feasible. To manage this business scenario, SAP provides Transaction OMET (Settings for Function Authorizations). With this transaction, you can create a new "rule" and then assign it to the end user by using a parameter value in the user master record. This action can be done in just a few minutes by following these instructions:

1. Launch Transaction OMET (see Figure 1).

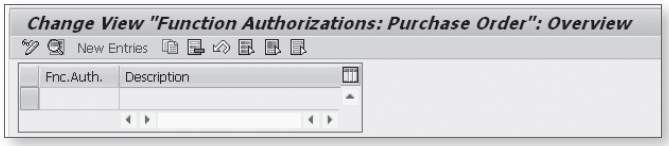


Figure 1 Transaction OMET Initial Screen

2. Create the rule by clicking on the NEW ENTRIES button (or press F5). A new screen will open, as shown in Figure 2, in which you can specify a rule name and the additional settings. In this example, the rule is named "AA" and the REF. TO PURCHASE REQUISITION checkbox is selected.

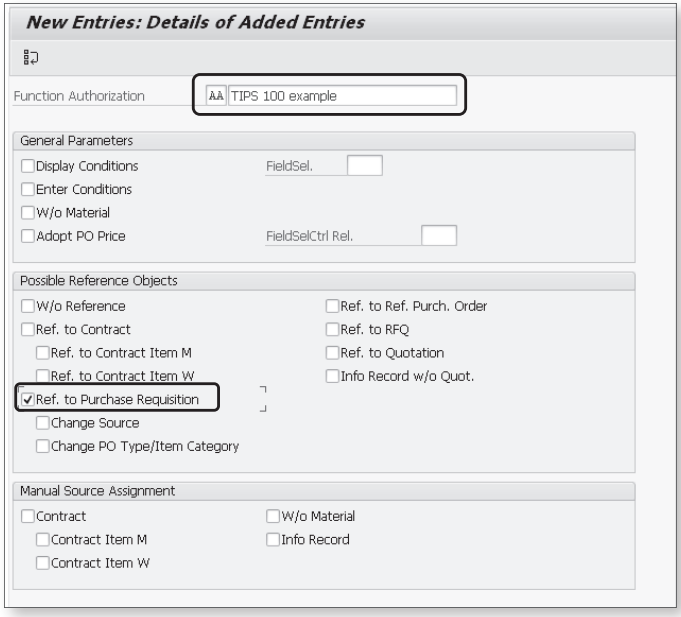


Figure 2 Rule Creation Screen in Transaction OMET

3. Now that the rule has been created, assign it to a user by using the Transaction SU01 (User Maintenance) PARAMETER tab as shown in Figure 3. Type "EFB" in the PARAMETER ID field (also shown as FUNCTION AUTHORIZATION: PURCHASE ORDER in the SHORT DESCRIPTION column).

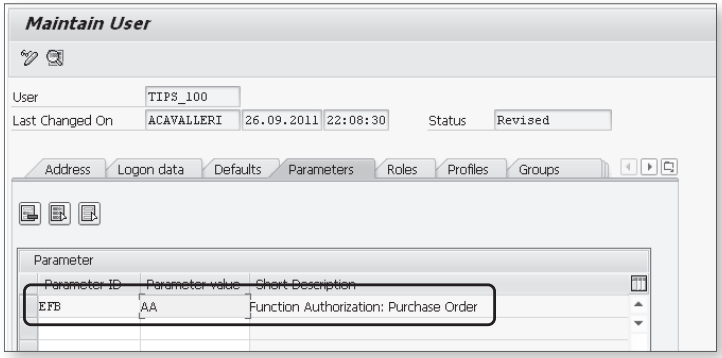


Figure 3 User's Parameters in Transaction SU01

Figure 4 shows the final result: An error message is displayed on the bottom line when the user tries to enter a new order without a purchase request field populated, indicating that a purchase request is mandatory.

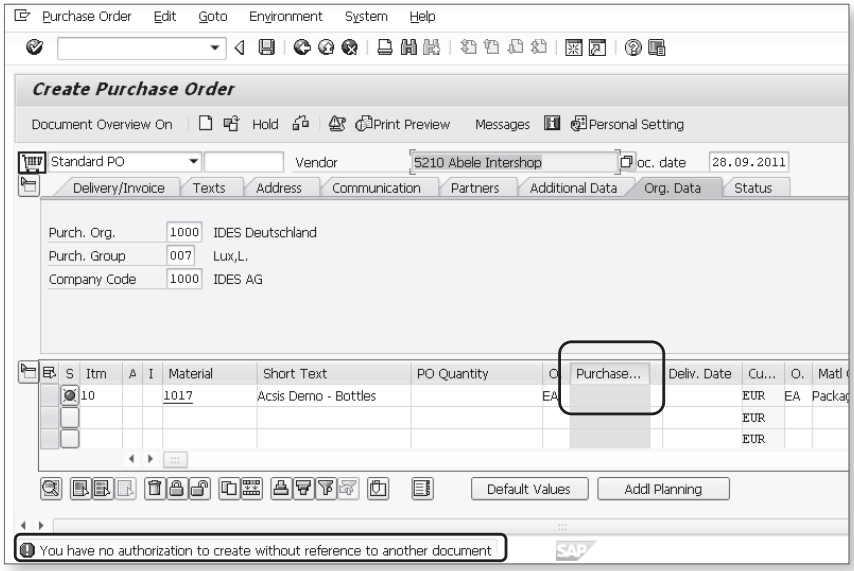


Figure 4 Purchase Order Creation (Transaction ME21N) without a Purchase Request Reference

Now you need to make sure that the user is not able to modify the parameter settings himself. Transaction SU3 is available in all SAP systems, which allows each user to change the main data of his user master record. As you can see in Figure 5, with this transaction the user can remove the EFP parameter you just activated. In such a case, the limitation on Transaction ME21N (Create Purchase Order) will be lost.

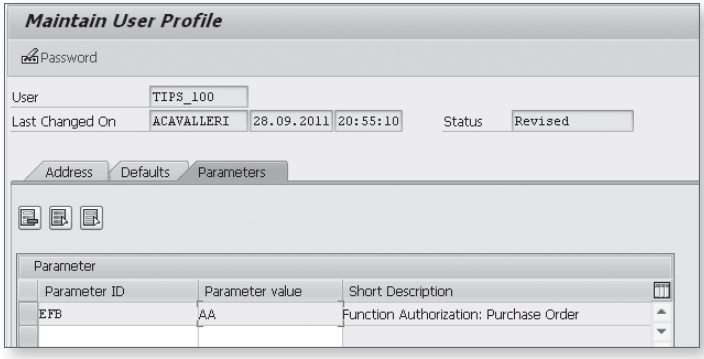


Figure 5 Transaction SU3: Maintain Users Own Data

We suggest authorizing end users with Transaction SU50 (Own Data) instead of Transaction SU3 because it's very similar, but the users won't be able to modify the PARAMETERS tab.

Tip 58

Removing Expired User-Role Links

You can easily mass-remove invalid and expired user-role links with a standard report.

You can assign a role to a user using a validity date concept, meaning the user will be authorized in a role for a specified period of time only. Unfortunately, the SAP system doesn't automatically remove the roles that have expired at that point—the authorization isn't available, but the link is still visible. Through a standard report, you can mass-remove the expired roles and avoid seeing these assigned roles in the user master record that can cause oversight. When you see all roles assigned to a user, often you can't keep an eye on the validity date, which may cause an error.

And Here's How ...

You can remove the expired roles that are assigned to users through Program PRGN_COMPRESS_TIMES. Execute this report by accessing Transaction PFCG, go to a role in the User tab, and choose the following:

Utilities • Optimize User Assignment

In this report (see Figure 1), enter a role name to be optimized or a user ID under User Assignments. You can also use a wildcard in the Role field.

It's also possible to perform a test execution by flagging the Simulation Run box. If you don't flag the Remove Validity Periods That Have Already Expired, the report will remove all roles duplicated on different periods on user master data. For example, if a user has been assigned the same role from 18.08.2011 to 25.08.2011, and in the second case from 19.08.2011 to 26.01.2011, the system will discard the

first role and will replace the date from 18.08.2011 to 26.01.2011. In other words, the system will maintain recent assignments but discard old ones.

If you flag Remove Validity Periods That Have Already Expired, the report will remove the roles that have expired.

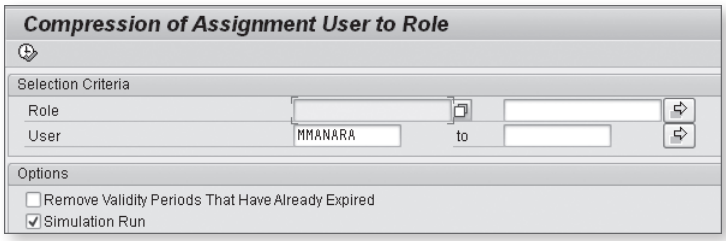


Figure 1 Compression of Assignment User to Role Interface

Delete Expired Roles en Masse

Another way to identify expired roles is to access Transaction SU01 by clicking on the circled button (Select Obsolete Roles) in Figure 2. If a user has expired roles, the system will highlight all roles assigned but expired by clicking on the Select Obsolete Roles button, and you can delete all expired roles from the user master record on analysis.

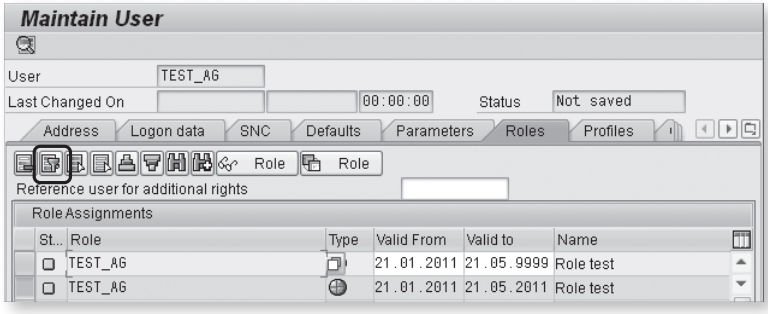


Figure 2 Select Obsolete Roles Button

Tip 59

Filtering Roles by Their Status

You can quickly obtain a role's status via a single standard report.

When you import a role from one system to another you have to regenerate the role profile. You also need to identify whether a role has a user assigned or not, as well as whether a role contains transaction codes or not. You can do this through Transaction PFCG—each tab has a traffic light symbol that shows you if the tab is complete and correct (green) or if it contains an error or warning (red or yellow). However, this process requires you to browse and access different SAP tables to retrieve these results. To avoid navigating between different reports, you can easily obtain the same information by using a single standard report.

And Here's How ...

You first need to access Transaction PFCG and then execute the standard Overview Status report by following this path:

Utilities • Overview Status

This report allows you to enter a set of roles (you can also use a wildcard asterisk to show the status of the roles involved). Figure 1 shows the role selection interface where you can enter the names of the roles and flag whether you want to display only the roles that contain errors and warnings (i.e., a red or yellow traffic light icon in the profile generator transaction).

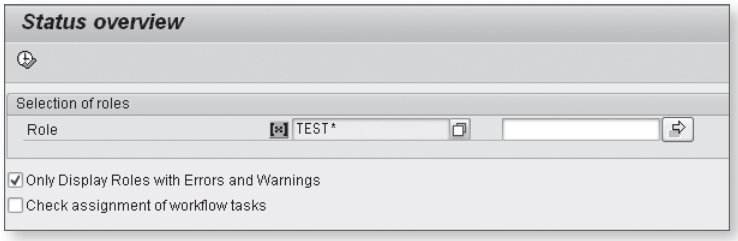


Figure 1 Status Overview Report

After you've maintained your fields, press F8 to see the result (see Figure 2). If you have a long list of selected roles and you want to know which of these don't have a generated profile, look for the red icon in the PROFILE STATUS column. To sort the list, highlight the PROFILE STATUS column (❶, see Figure 2), and then click on the funnel icon (❷).

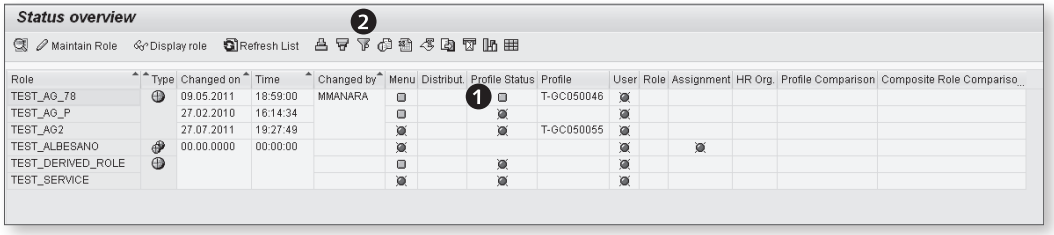


Figure 2 Overview Report: All Roles that Start with the TEST Name*

After highlighting the column and clicking on the funnel icon, you'll see the DETERMINE VALUES FOR FILTER CRITERIA pop-up in Figure 3. Click on the PROFILE STATUS field, and then press F4 to see the match code values. Then select one value (e.g., @5C\QNO AUTHORIZATION DATA EXISTS@) to find which roles don't contain data in their profile.

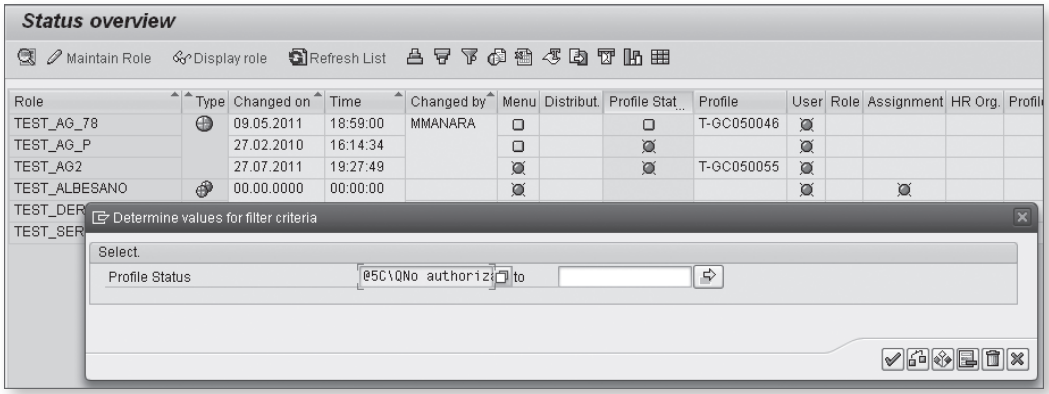


Figure 3 Determine Values for Filter Criteria Pop-Up

At the end of this selection criteria process you can see the list filtered by your selection criteria and find the status of all of your identified roles. By typing Ctrl + Shift + F9, you can also export the list in a spreadsheet format, such as Microsoft Excel.

Part 4

Segregation of Duties

Things You'll Learn in this Section

Segregation of Duties (SoD) is an essential part in the authorization process. To manage this mandatory matter, which is required by national and international law, SAP supplies SAP BusinessObjects governance, risk, and compliance solutions

(which we'll refer to as SAP GRC). This software suite is formed by an ecosystem of systems that are mostly composed of SAP GRC Access Control, SAP GRC Process Control, and SAP GRC Risk Management.

This part of the book focuses on SAP GRC Access Control, which is formed by four capabilities: Risk Analysis and Remediation (RAR), Compliant User Provisioning (CUP), Enterprise Role Management (ERM), and Superuser Privilege Management (SPM). Since July 29th, 2011, SAP has released a new and enhanced version of this software (release 10.0). Here, we'll teach you how to exploit the SAP GRC AC tool. Each tip will contain a specific release detail to clarify the release. Some tips will also give you advice on how to manage SoD without SAP GRC.

Tip 60

Tailoring Your Ad-Hoc Analysis by Using Custom Groups in RAR and ARA

You can use custom groups to select only a specific set of users in your risk analysis.

Overall, it isn't easy to perform an ad-hoc risk analysis on only a subset of users in SAP GRC Access Control Risk Analysis and Remediation (SAP GRC AC RAR 5.3). This gap is solved instead in SAP GRC Access Control Access Risk Analysis (SAP GRC AC ARA 10.0).

This tip shows you how to simply define and use a custom group to choose only a selected set of users in SAP GRC AC RAR release 5.3.

And Here's How ...

In SAP GRC 5.3, during an ad-hoc risk analysis there's no way to automatically insert or select several users. Figure 1 shows the selection box in ad-hoc analysis. You can only insert users one by one or by defining a range. Unfortunately, it isn't always possible to split the users up in a range or use the standard user group defined in the LOGON DATA tab of Transaction SU01. See SAP Note 1493683 for an explanation of how user group data is accumulated in SAP GRC 5.3.

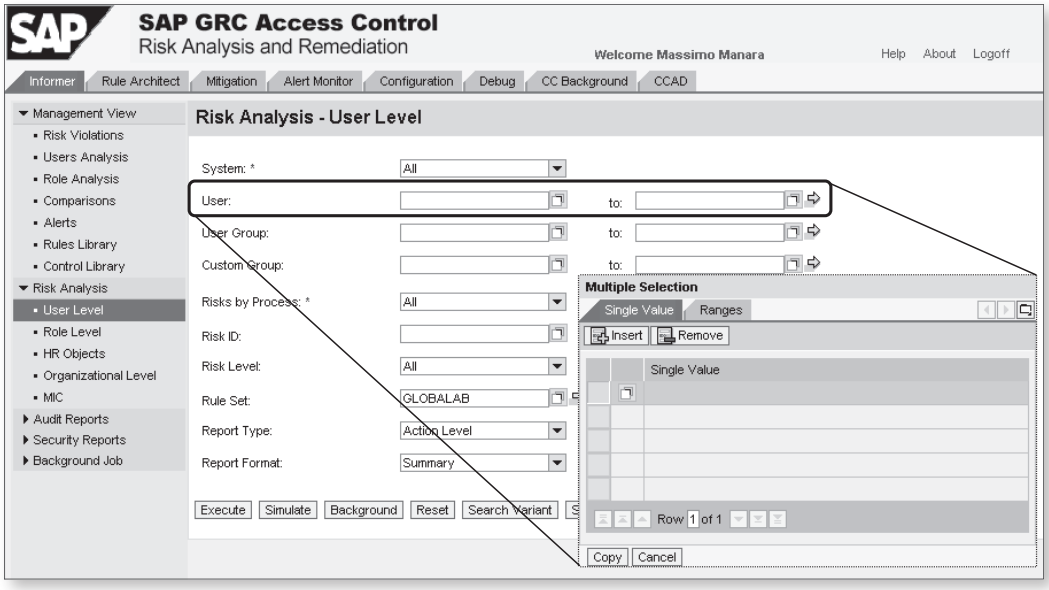


Figure 1 SAP GRC AC RAR: User Level Risk Analysis on the Informer Tab

To allow people who run ad-hoc risk analysis to select a predefined group of users, you can exploit the custom user groups functionality. To do this, go to the CONFIGURATION tab, and select the CUSTOM USER GROUPS item in the left-hand menu. Here, an administrator can create the link GROUP – USER manually, or import a plain text with predefined groups with the IMPORT button. You can create a plain text file with the group name tab delimiter and user ID.

Select a file in the BROWSE field to retrieve the text file, and then click on IMPORT (Figure 2 shows the custom user groups import interface with the BROWSE and IMPORT buttons). Make sure that your text file has been saved in UTF8 code (you can do this through the Microsoft Notepad program).

After the text file is loaded, you can search the groups that you just loaded by using the SEARCH functionality under the CUSTOM USER GROUPS ITEM (an example result is shown in Figure 3).

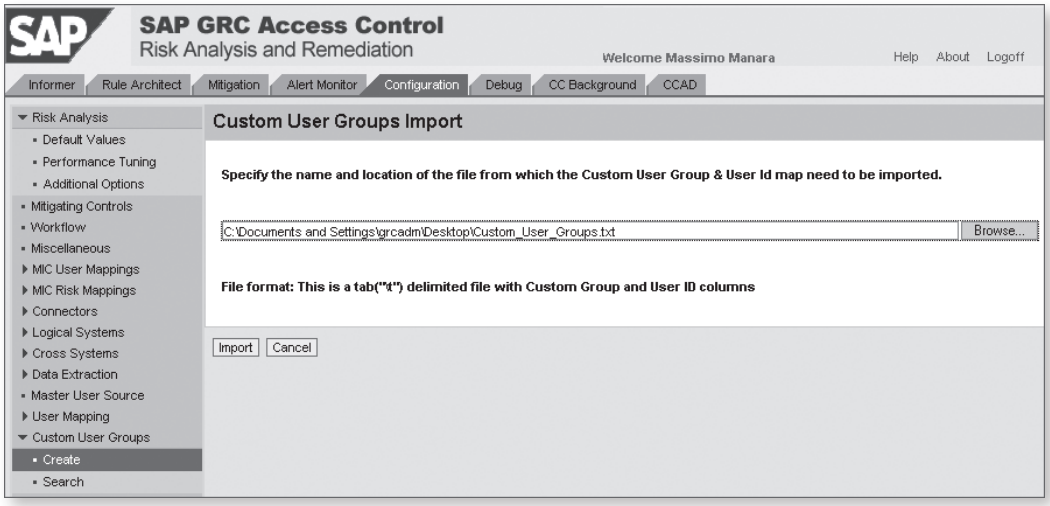


Figure 2 Import Phase from a Text File

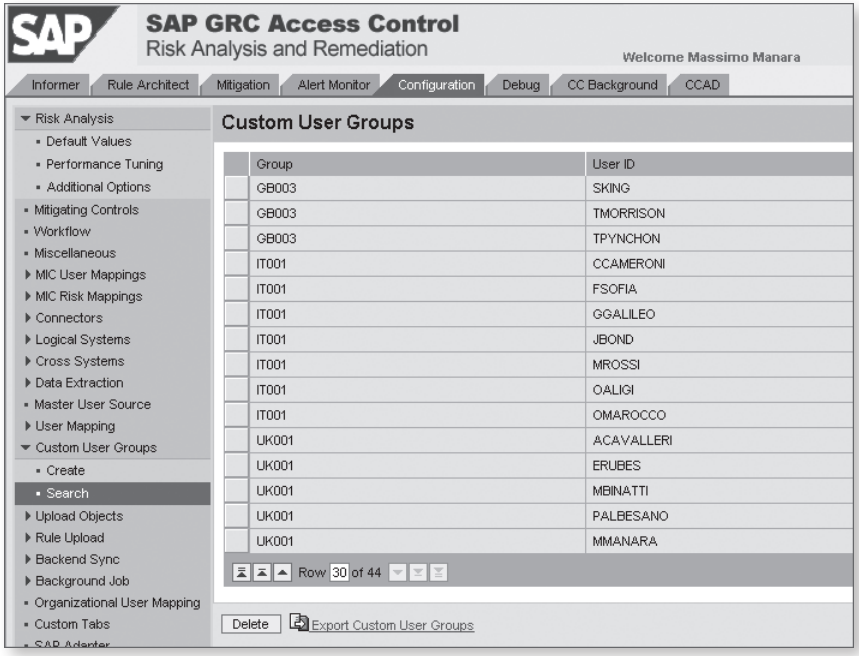


Figure 3 Search Custom User Groups Just Imported and Defined

Now, during an ad-hoc risk analysis an end user can select a predefined custom user group by clicking on the CUSTOM GROUPS match code in the USER RISK ANALYSIS interface and then selecting the custom user groups that you need.

This approach allows you to select a subset of users on which to perform a risk analysis. However, keep in mind the following disadvantages in SAP GRC AC RAR 5.3:

- Custom groups must be maintained manually. So if you want to add or remove a new user from a group, you have to insert/import it manually. Further imports of this text file append the data to the existing data. See Sap Note 1504689 for more information.
- The CONFIGURATION tab should be accessible only to an administrator, which means you have to predefine these user groups so that they're visible to the end users using SAP GRC AC RAR.
- You can't mass-delete the obsolete custom user groups. Instead, you need to ask your system administrator to clear the Java Table VIRSA_CC_CGROUP.

In SAP GRC AC ARA 10.0, this workaround is not necessary anymore. Instead, you can load your text file that contains a set of users to put under risk analysis during an ad-hoc risk analysis.

Tip 61

Modifying Your Selection Criteria for User/Roles Analysis in SAP GRC 10.0

You can use the new selection options available in SAP GRC 10.0 to enhance selection criteria during your analysis.

Due to the web interface in SAP GRC 5.3 it wasn't always simple to perform some types of selection criteria. In SAP GRC 10.0, new features have been introduced to enhance and improve this previous gap. However, if you are a release 5.3 user, some of these new features may not be readily apparent or easy to understand. This tip will explain these new selection options.

And Here's How ...

While selecting criteria for risk analysis in SAP GRC 10.0, you can follow this path in the SAP NetWeaver Business Client for user and roles analysis:

Access Management • Access Risk Analysis

Open the Risk Analysis: User Level interface as shown in Figure 1. Next to each selection criteria there's a Boolean dropdown box (circled in Figure 1). You can enhance your selection by selecting specific criteria (e.g., User) and selecting a condition.

You'll be able to understand all options by reading the text. However, note that an especially useful smart feature is the Multiple Selections option, which allows you to upload a text file where all your selection entries are listed from your frontend

system. If you select MULTIPLE SELECTIONS conditions, a new button will appear named ADD SELECTION, which will allow you to upload a text file or perform exclusion selection of the list loaded. Figure 2 shows the MULTIPLE SELECTION window.

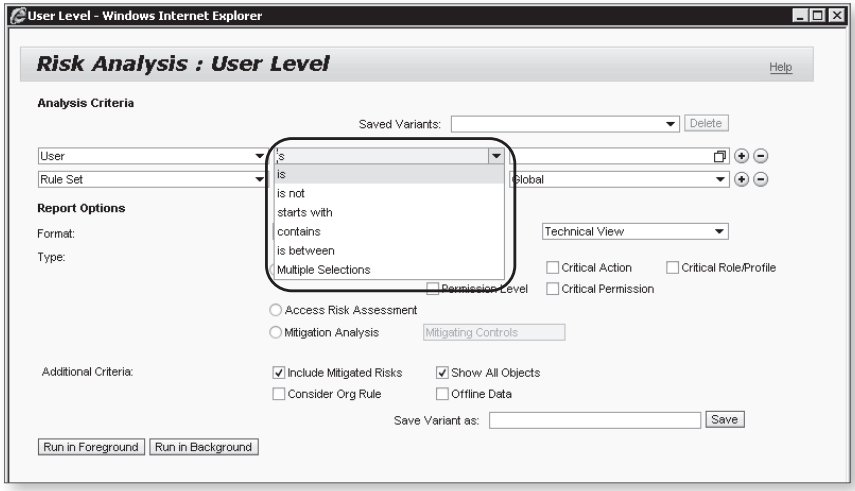


Figure 1 Selection Criteria Options in User-Level Risk Analysis

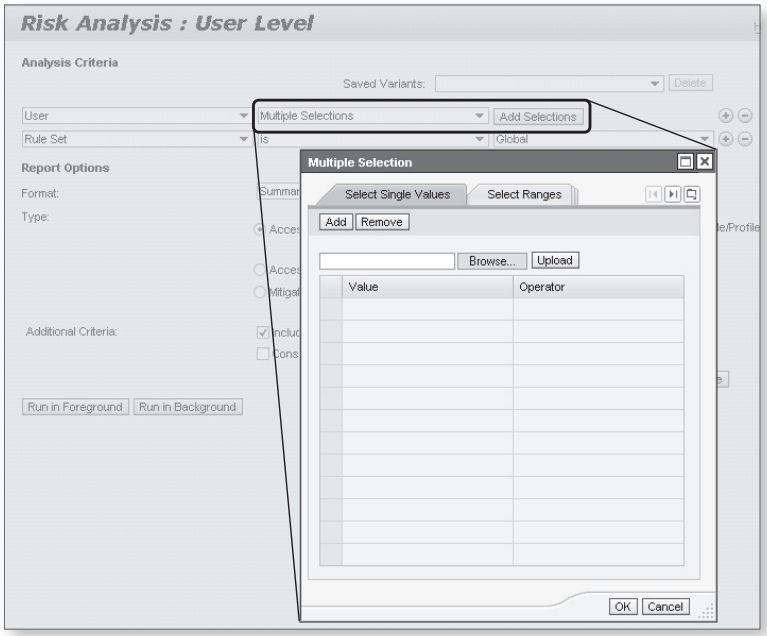


Figure 2 Multiple Selection Pop-Up

With the Browse button, you can search your text file where you have listed your data and then use the Upload button to load these data. Figure 3 shows the loading result. If you load the file, SAP GRC puts the data in append mode so if there is previously loaded data, it won't be overwritten.

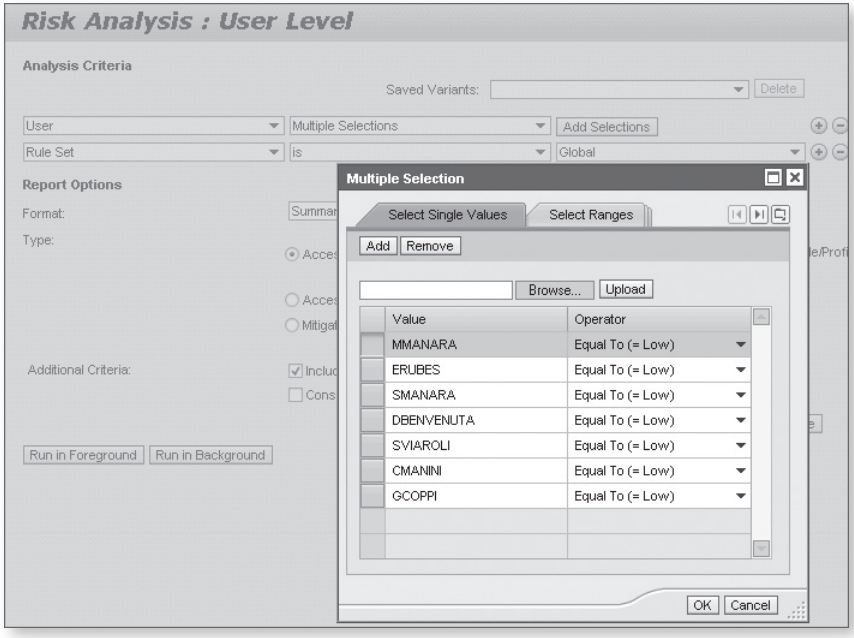


Figure 3 Loading Results of Multiple Selections

This way, you can easily and quickly perform sophisticated selections criteria to retrieve your tailored results.

Tip 62

Clustering Data to Enhance Your RAR Reporting for Easier Consumption

You can customize your risk analysis reporting to make it easier to understand for decision makers in your business.

You may find it difficult to involve business process owners during the first risk analysis because it's possible to flood the business process owner with several thousand rows of risk analysis output. You therefore need to decide, together with the HR department and the business unit, how to cluster this data. For example, each business process owner might receive the output based on your cost center or company user ID.

This tip shows you how to set up a reporting framework to involve the business process owners in these decisions.

And Here's How ...

To avoid an excessive growth of data that you need to manage, a fundamental point in reporting is to exclude specific objects from risk analysis.

Exclude Objects

Traditionally excluded objects are the profiles `SAP_ALL/SAP_NEW`, all SAP standard roles (if unused), and all your custom roles with extensive privileges. This is necessary to prevent decreasing database performance and to streamline the reporting database. Consider that a user with an `SAP_ALL` profile (using the SAP GRC standard rule set) generates about 300,000 rows of conflicts in the database.

There's a simple way to exclude the millions of rows of critical objects from the analysis—not by managing them but simply by treating them in a different way. Generally, these users are in the IT department; for example, technical users, consultants, or IT people. See Sap Note 986997 for additional information on excluding critical roles.

Classify Output Results

It's important to know how to classify the results of the output. If you have several thousands of users, you have to find a way to classify or cluster the users: by company, by cost center, by department, and by user IDs.

In SAP GRC RAR 5.3 from support package 9, SAP has introduced a new functionality called *data mart* to enable you to read the results of risk analysis and elaborate on them through an external reporting tool such as SAP BusinessObjects, business intelligence tools, or Microsoft Access.

This functionality gives you the opportunity to correlate your risk analysis data with HR or Lightweight Directory Access Protocol (LDAP) information to make more business-oriented and tailored reporting.

In SAP GRC 5.3, there are several guides and tutorials about configuring and using the data mart functionality. Follow these steps to configure the data mart in your system:

1. Enable offline risk analysis (use the customizing switch in the Configuration tab, and then choose Risk Analysis • Additional Options • Enable Offline Risk Analysis).
2. Schedule all synchronization, risk analysis, and management report jobs (under the Configuration tab, choose Background Job • Schedule Job).
3. Define your Java Database Connectivity (JDBC) connector data source (in Visual Administrator tools). This step requires a system administrator's support.
4. Enable the data mart customizing switch (under the Configuration tab, choose Risk Analysis • Additional Options • Enable Data Mart Job).
5. Schedule a data mart job (under the Configuration tab, choose Background Job • Schedule Job • Data Mart).

The process is basic: After the risk analysis tables are filled with risk analysis results, the data mart job takes a snapshot of the data (risk analysis result) into a set of other tables that are accessible by the data source that's been defined. You can read

the data through your external reporting tool. Note that all data mart tables in 5.3 start with GRC_DM. In this way, you can correlate the results of the risk analysis, of a certain moment, with other information and data.

SAP GRC 10.0 provides the data mart functionality as well, but only for the SAP Process Control and Risk Management system. You can use the ABAP tables listed in Table 1 for reporting.

SAP GRC AC ARA Tables	Description
GRACROLEPRMVL	Role Permission Violation Table
GRACROLEUSAGE	Role Usage
GRACUSER	User Table
GRACUSERACTVL	User Action Violation Table
GRACUSERCRPVL	User Critical Role/Profile Violation Table
GRACUSERMGMTSUM	User Management Total
GRACUSERMGVCOUNT	Management User Violation Count
GRACUSERORG	User ORG Level
GRACUSERPRMVL	User Permission Violation Table
GRACUSERROLE	User Role Assignment Table

Table 1 Mainly Risk Analysis Result Tables in SAP GRC Access Control 10.0

By using the tables listed in Table 1, you can use a reporting tool such as business intelligence options or Microsoft Access to develop your query and report. By relating these tables, you can define and streamline your own report to give the business process owners a better understanding of your organization view.

Tip 63

Performing a User Impact Risk Analysis

You can simulate changes at the role level and show the potential impact at the user level.

During continuous compliance and daily maintenance, it's important to discover whether a change at the role level will generate one or more conflicts for all users assigned to the role. As a quick example, let's say you insert Transaction ME51N in the buyer job role. If Transaction ME51N is SoD relevant (inserted in your SoD matrix/rule set), you have to check whether this change to the buyer role could cause some risk at the role level and for all users assigned to this role. User impact analysis features are available in both SAP GRC 5.3 and 10.0, and you can easily perform this check and ensure compliance, reducing the remediation time spent simulating it first.

And Here's How ...

In SAP GRC 10.0, you can perform a role analysis simulation by following this path:

Access Management • Access Risk Analysis Role Level Simulation

In SAP GRC 5.3, you can perform a role analysis by following this path:

Informer • Risk Analysis • Role Level • Simulate

You can also perform a role-level simulation. Figure 1 shows the simulation interface for a role level on System ECC_G10_810 and Role TEST_AG.

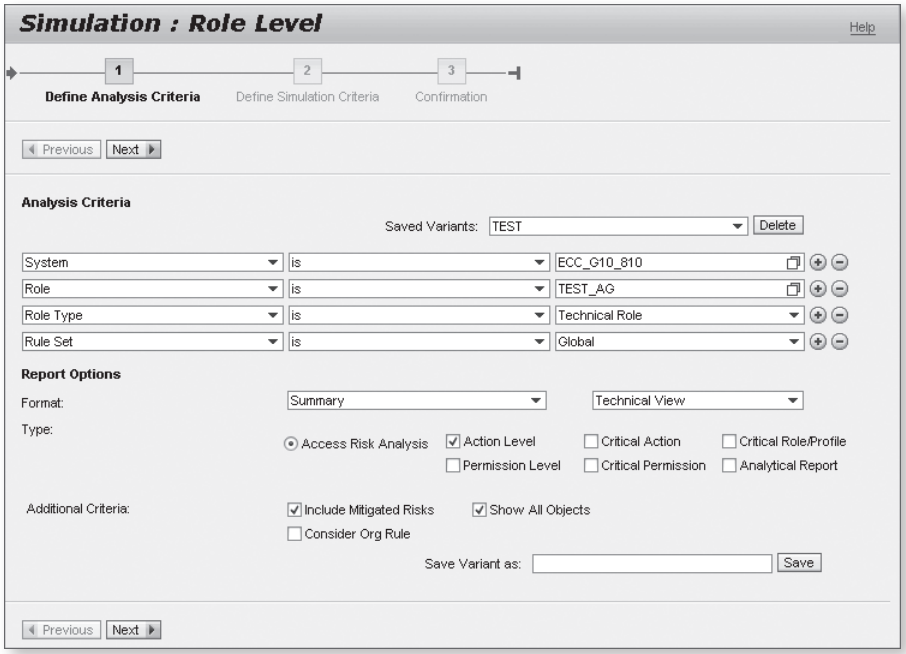


Figure 1 Simulation Role Level Interface

You have to follow two steps, which you can see at the top of Figure 1. The first is Define Analysis Criteria, where you insert the role name, role type, system, and rule set; you can also define the analysis type at the action level or permission level by selecting the respective flag in Figure 1.

The second step is Define Simulation Criteria shown in Figure 2. Through Additional Criteria flags, you can decide if the simulation should include or exclude values. If Exclude Values is checked, you're indicating that you want to remove Actions, Roles, or Profiles tabs, circled in Figure 2. If you want to include the users in your simulation, the impact user analysis is performed if the Include Users checkbox is checked.

In the example shown in Figure 2, you want to simulate what would happen if you remove (by flagging Exclude Values) Transactions MM01 and FB01 (click on the Actions tab, and then click on the Add button to insert the transaction codes)

from role TEST_AG (previously selected in the Define Analysis Criteria step), as well as the impact analysis on all users assigned to this role.

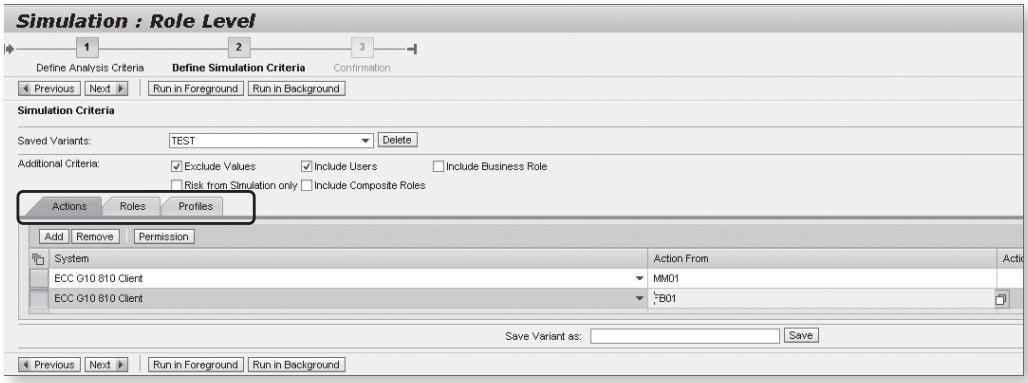


Figure 2 Define Simulation Criteria

After you've defined the first and second step by inserting the analysis criteria and simulation criteria, click on the Run in background button. SAP GRC automatically generates two jobs: The first represents the analysis on TEST_AG selected roles, and the second represents the user impact analysis.

By accessing Access Management • Scheduling • Background Jobs, you can see your two jobs scheduled as shown in Figure 3. The job name IMP_7 in Figure 3 represents the name defined by the user during the scheduling, this job contain the risk analysis result on the role inserted in the Define Analysis Criteria step. The second job's name is generated by SAP GRC and represents the user impact analysis.

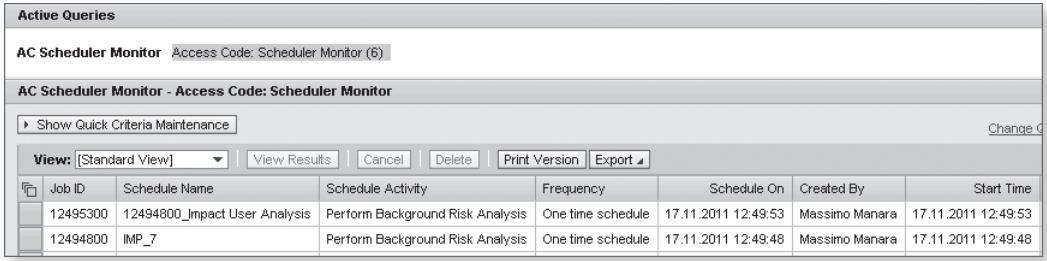


Figure 3 Job Overview

It's important that all SAP GRC synchronization jobs are performed as described in the SAP GRC installation and operation guide.

Tip 64

Setting Selection Criteria for the Web Interface as a Default Value

You can use a commonly used value as a default value in the SAP GRC web interface to avoid errors and save time during your selection analysis.

SAP GRC 10.0 allows you to define a default value during the selection criteria for the web interface. When you only have the name of a system or an often used value to use as the search criteria, you can save time by setting it as a default value. By using default values and variants, you can avoid manually inserting the selection criteria time and time again with the risk of mistakenly producing a report that is not in line with your requirements.

And Here's How ...

During risk analysis, suppose you want to set a system ID value as the default value at the user level. Figure 1 shows the Risk Analysis: User Level screen with the System field circled. Let's walk through the necessary steps to set a default value for this field.

First, enter your default value in the field; for example, ECC_G10_810. Right-click on the field involved; in this case, it's the System field circled in Figure 1, which will open the User Settings contextual menu as shown in Figure 2.

If you click on Use Current Value as Default, when you go in an interface with the same field, this field will be prepopulated with your default value.

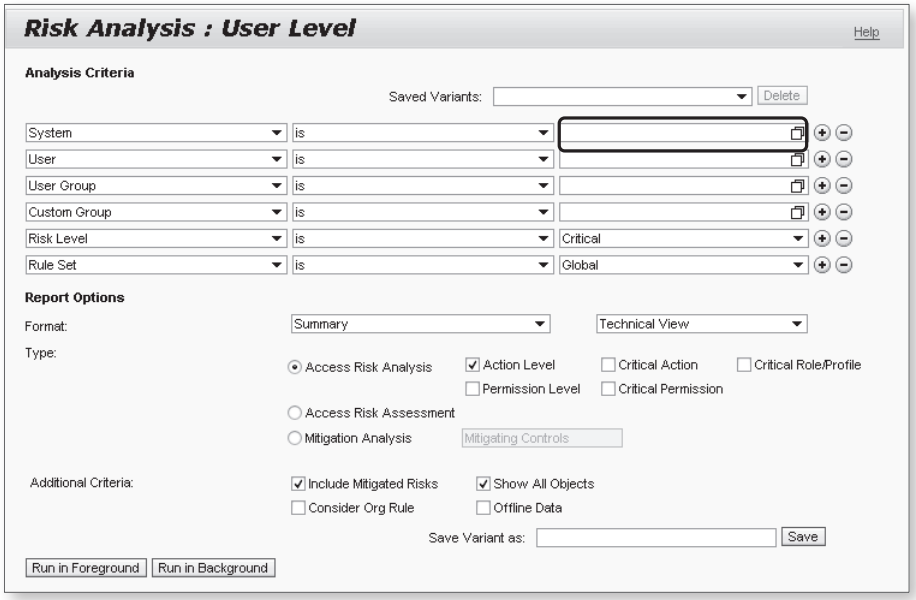


Figure 1 Set a Default Value for the System Field

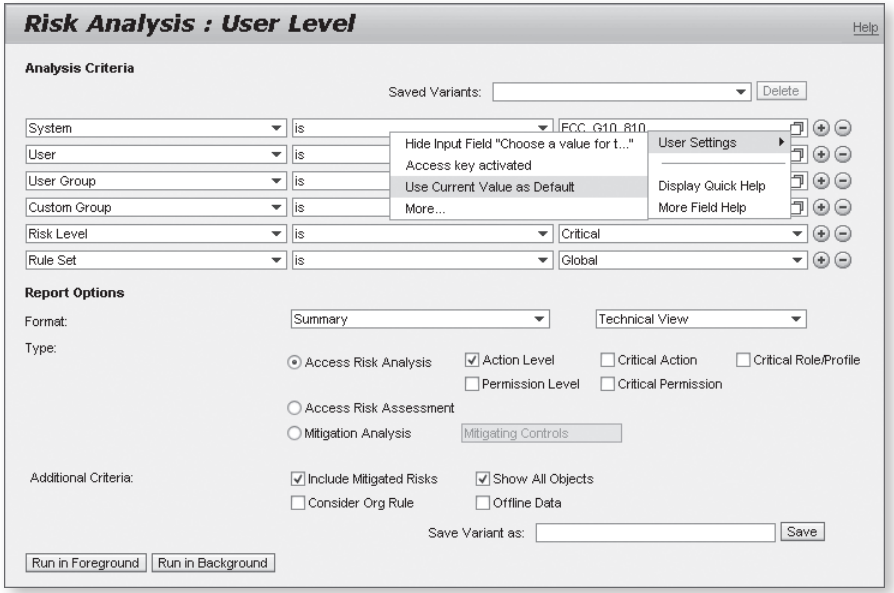


Figure 2 User Setting Contextual Menu

Tip 65

Defining a Firefighter User ID Naming Method

You can define a firefighter user ID naming method to better identify your firefighter user ID and to improve system performance.

Neither SAP GRC Access Control 5.3 (Superuser Privilege Management [SPM]) nor 10.0 (Emergency Access Management) release a predefined best practice to define a firefighter user ID (FF ID). Note that each customer can define his own firefighter naming conventions. However, during the daily use of this capability, you'll discover that if the firefighter naming is smartly defined, it can improve reporting and system performance. Defining FF IDs allows you to better manage the firefighter user master data, better identify firefighter users, and in some cases improve the performance of this SAP GRC Access Control (SAP GRC AC) capability.

And Here's How ...

Both SAP GRC AC 5.3 with the SPM capability or SAP GRC AC with the EAM capability allow you to define a set of emergency users to assign critical activity duties to specific end users. Each of these emergency users (FF IDs) are traced using the standard trace tools provided by SAP. This tool allows you also to define super roles instead superuser IDs; generally this tool is used more for user IDs than for roles.

Depending on your company size and SoD process, you have to define a small set of these user IDs, or several FF IDs. In SAP GRC 5.3, you have to define and configure each system to put your FF ID under SPM. In SAP GRC 10.0, you instead have to configure the SAP GRC EAM system centrally; SAP GRC will use the connectors (RFC connections) defined to connect to the backend system where the FF ID has

been defined and created. You can find detailed steps about these post-installation tasks at *www.sdn.sap.com*.

You have to define your FF ID by using Transaction SU01. FF IDs can be a maximum length of 12 characters from a SAP system. A user ID becomes an FF ID when it's assigned in the firefighter table in SAP GRC EAM. Figure 1 shows the Firefighter IDs assignment table. Access this table by following this path and then clicking on the Assign button:

Access Management • Super User Assignment • Firefighter ID

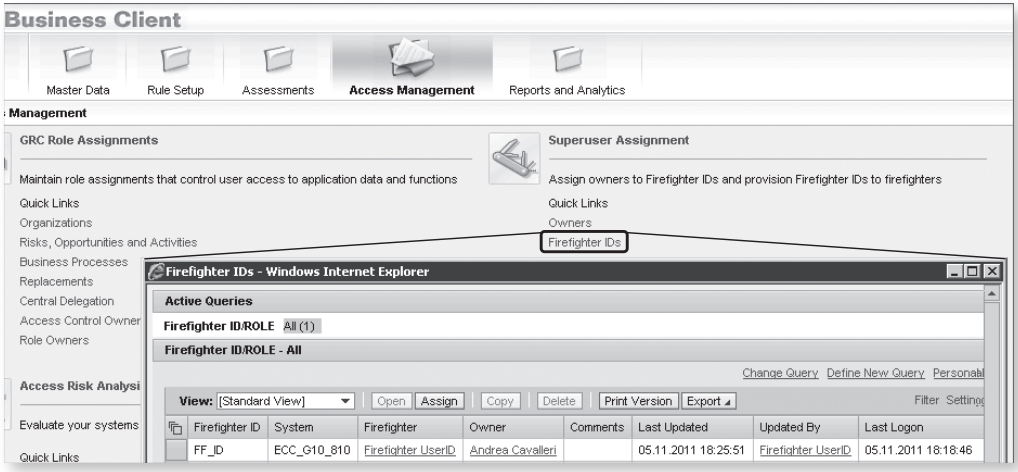


Figure 1 Firefighter IDs Assignment in the SAP NetWeaver Business Client Interface

Naming Conventions

A suggested naming convention for FF IDs is a set of 12 characters (12 characters represent the SAP ABAP maximum length).

1	2	3	4	5	6	7	8	9	10	11	12
F	F	_	S	I	D	C	L	N	0	0	0

The numbers in the convention are described here:

- Slots 1 and 2 represent the pattern for FF ID. With the first two characters, you can better identify your FF IDs defined in the backend system as well as improve

the job log used by EAM during the log collection. You should group the FF IDs by using a specific user group for authorization management purposes or simply for better grouping of these kinds of users.

- Slot 3 represents a separator character.
- Slots 4–6 represent the system ID (SID) of the backend system where the FF ID is defined. Through the SID in Table 1 from characters 4–6, you can at first identify where the FF ID is defined in your system landscape. You can also use three further characters in Table 1 from 7–8 to define the system's client. For systems with multiple clients, you can identity not only the system where the FF ID is defined but also the client.
- Slots 10–12 represent a numerical progressive, if there is more than one FF ID in the same system/client. You can also use these three last characters for further definition of the FF ID; for example, if you've defined a FF ID to cover some critical action in the Financial Accounting component process, you can use FI as the last characters; for an action in the Sales and Distribution component, you can use SD; and so on.

Tip 66

Using Organizational-Level Mapping in Business Role Management to Improve Role Derivation

Using SAP GRC 5.3 or 10.0, you can save a lot of time when deriving, defining, and maintaining roles by using organizational-level mapping.

There are 34 predefined organizational levels in the SAP ERP system. Imagine that one company has several thousand plants. In this case, if your business needs require you to segregate each plant, you have to create thousands of roles derived by each plant. This task is very time consuming using standard Transaction PFCG. Through Business Role Management (BRM) in SAP GRC 10.0 (also available in SAP GRC 5.3 Enterprise Role Management [ERM]), you can save a lot of time and ensure the compliance of your internal policy by using the organizational-level mapping features.

And Here's How ...

Manually Create Roles Using Transaction PFCG

Role derivation in Transaction PFCG is particularly useful when you have to split the data domain across two job roles with the same activities. For example, a job role can create a financial document only on company code IT10 and the other job role can create a document on company code IT20. You have to create a parent role that contains Transaction FB01 (Financial Document Creation) and then define two

derived roles. The first role is on the company code IT10 organizational level, and the second role is on the company code IT20 organizational level. If you only have a few organizational levels to maintain, you can do this manually. But if you have several thousand organizational levels and combinations, it's time consuming and tedious to define and manage these roles.

Using BRM Mapping

In BRM, the organizational-level mapping feature can help you in this derivation definition. Basically, you use the organizational-level mapping feature to define a set of primary organizational levels where you insert all linked child organizational levels. In this way, you can define a cluster of values to the naming of your branch or company. For example, if you define the primary organizational level named ITALY with BUKRS equal to the value 1000, you can insert all child levels such as the following:

- KORKS = 1500
- VKORG = 2000

Then, when you define a derived role in BRM with ITALY as the primary organizational value, all child values will be populated automatically.

You configure organizational-level mapping in the Transaction SPRO tree by following this path:

> Governance, Risk and Compliance • Access Control • Role Management • Define Organizational Value Maps

You now have to define your primary and secondary organizational levels.

Define Your Primary Organizational Level

Click on New Entries in Figure 1 and then insert the primary organizational level. This example uses ITALY for the Org Value Mapping field, BUKRS for the Organizational Level field, and then the value 1000.

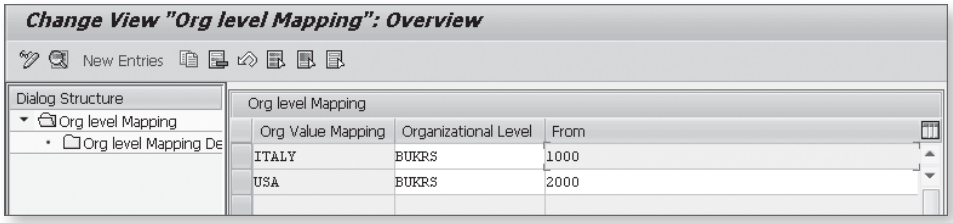


Figure 1 Primary Organizational-Level Mapping

Define Your Secondary Organizational Level

Click on Org. level Mapping Detail (under Dialog Structure in Figure 1), and insert all child organizational levels (in this example, set GSBER equal to 10 and KOART equal to 15) as shown in Figure 2. All child organizational levels—GSBER and KOART for ITALY Org. Value Mapping in Figure 1—are now defined.

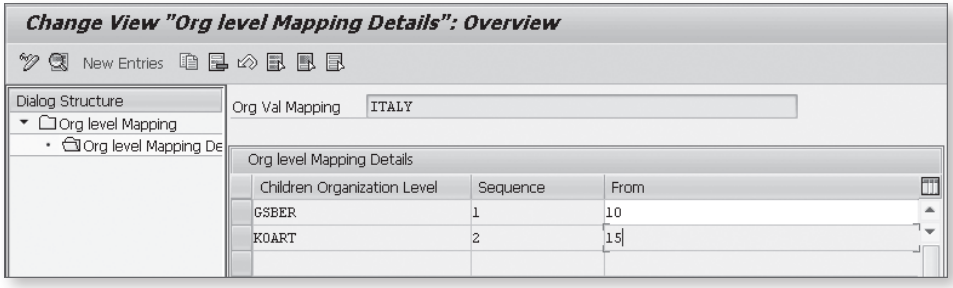


Figure 2 Children Organization Level of Org. Value Mapping for ITALY

Define Derived Roles

After this configuration setup, define your first derived role by using the organizational-level mapping feature. Follow this path to create an authorization role with BRM:

Access Management • Role Management • Role Maintenance

If your methodology includes the derivation step, you can use the features discussed previously.

Figure 3 shows the derivation step. If you insert the primary organizational-level value during the role definition, BRM will propose all mapping that is available to consequently generate all derived roles with the mapped values.

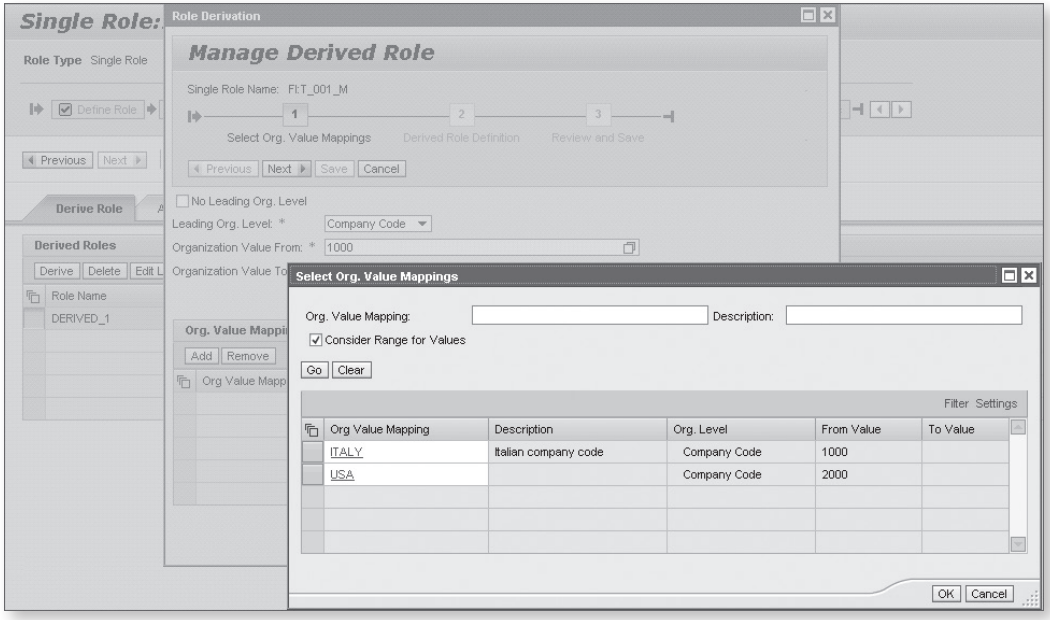


Figure 3 Select Org. Values Mappings

Tip 67

Using Business Role Management to Define Business Roles in Place of Composite Roles

When you need to use or combine composite roles but are using SAP ERP, you can use SAP GRC 10.0 to define a container of technical roles called business roles.

You can use different types of roles in the SAP ERP system or in SAP GRC 5.3 such as simple, derived, and composite to define your authorization concept. You can obtain the maximum level of abstraction by using composite roles. This type of role allows you to define a company job role formed by several simple roles.

In some cases, due to role complexity, it may be necessary to create a further level of abstraction by grouping several composite roles. However, this solution in an SAP ERP system isn't possible. To circumvent this issue, you can use the role mapping features as a workaround in SAP GRC 5.3; however, the real solution is given by SAP GRC 10.0. SAP GRC 10.0 has introduced a new type of role called the *business role*. This kind of role can contain all other types of roles, as well as itself. A business role can also contain several other business roles.

And Here's How ...

You first need to verify that this type of role is enabled in your SAP GRC Business Role Management (BRM) capability. You can check this through Transaction SPRO under the path:

Access Management • Role Management • Maintain Role Type Settings • Deactivate Role Type

Ensure there are no business role types in the table that appears. If a role is in this table, it's deactivated and you can't use it.

To create a business role, go to the SAP NetWeaver Business Client and access the Access Management work center. Click on the Role Maintenance application link, as shown in Figure 1.

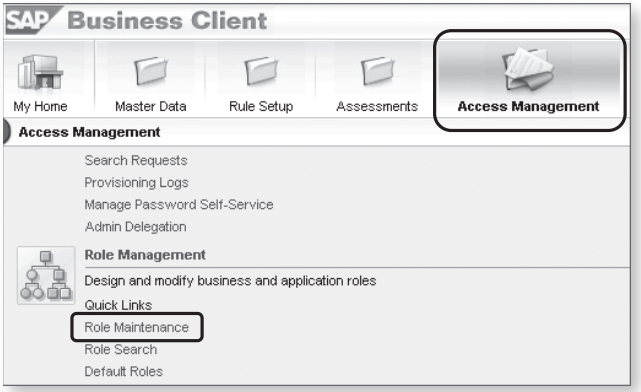


Figure 1 Access Management Work Center and Role Maintenance Application Link

In the resulting Role Maintenance screen, create a new business role by clicking on the Create button shown in Figure 2.

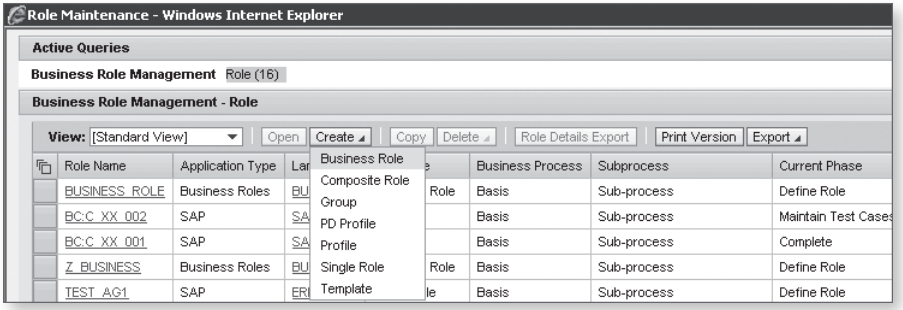


Figure 2 Create a Business Role

After you create a new business role, define the role attribute details, landscape, business process, subprocess, role name, and so on. Figure 3 shows the business role definition step in the Roles subtab.

Here, you can search for and insert the role you just defined. You can also insert simple roles, composite roles, or other business roles.

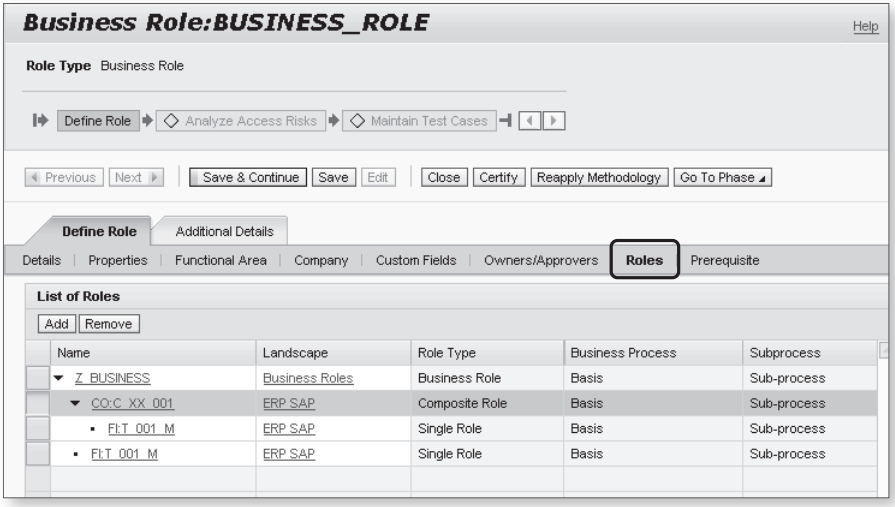


Figure 3 Business Role: Definition Step

This new type of role can enhance and simplify the role structure and concept, but it's essential that you don't abuse it. If you create too many levels of abstraction, you run the risk of losing the governance of the authorization model.

Tip 68

Setting Up Data Segregation in SAP GRC ARA

You can restrict data access through the authorization objects in SAP GRC 10.0 to ensure that each administrator works only on his own data.

In SAP GRC 10.0 (unlike SAP GRC 5.3), you can easily use the standard authorization objects to segregate some SAP GRC activities (e.g., by system ID, risk ID, or rule sets). Suppose you've set up a two-layer rule set. The layers are defined by using a risk or function naming convention. The first layer is valid for all geography and should be changed only from company headquarters. The second level of the same rule set is a layer to manage the local risks for each country. This last layer should only be changed by the one local rule set administrator. Restricting user access for a scenario like this can be very tricky in SAP GRC 5.3, but it is easily achieved in SAP GRC 10.0.

And Here's How ...

SAP GRC 10.0 provides authorization objects to manage and segregate the rule set maintenance. In SAP GRC, the rule set represents the containers that hold all conflicting rules defined in your company.

Imagine that you have a common rule set released from headquarters that is valid for all countries around the world. Each country, through a specific shared naming convention, could enhance and improve its own risks by adding risks and functions based on its specific local process, law, and policy.

Through the `GRAC_RISK` and `GRAC_FUNC` authorization objects you can segregate the access risk and the function maintenance, respectively.

Figure 1 shows a Transaction PFCG role with these two authorization objects. On GRAC_FUNC there are two authorizations. T-G055101300 allows you to maintain all function IDs that start with TE*. The second authorization, T-G055101301, allows you to view all functions that are ID defined. You can insert more than one authorization on an authorization object by clicking on the MANUAL button when you're in the Transaction PFCG authorization tree.

In the same way, GRAC_RISK is formed by two authorizations. The first allows you to manage all risks of business process basis and APO (BS00 and APO) in the AGLEA rule set. The second allows you to display all risks defined in all rule sets that are available.

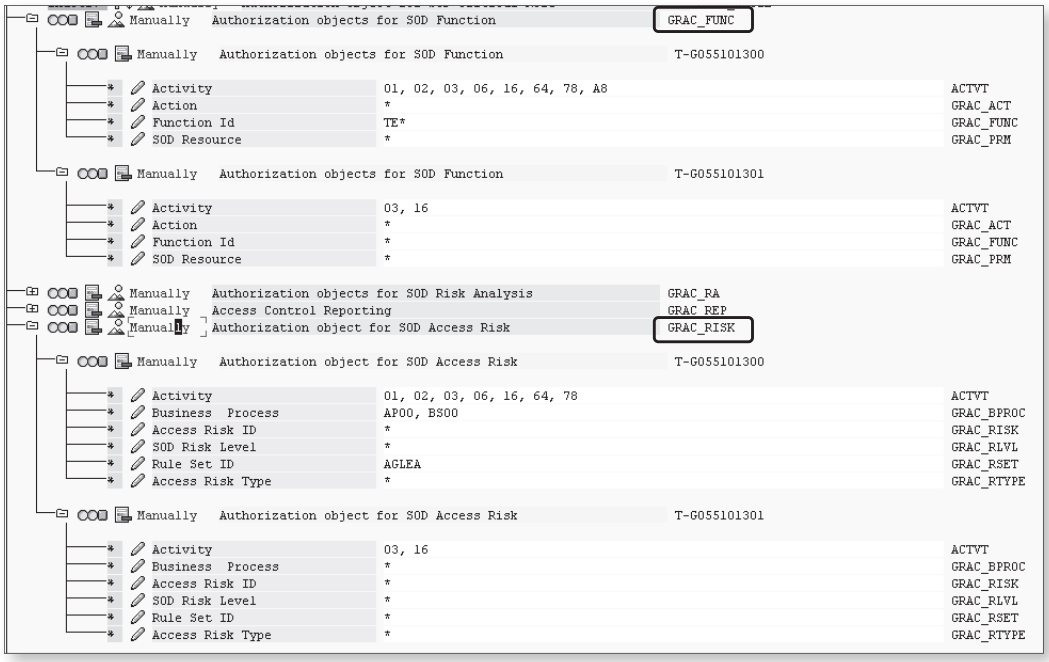


Figure 1 GRAC_FUNC and GRCAC_RISK Authorization Objects

After a user receives this role (with the authorization shown in Figure 1), he will only be able to display access risk and function ID links on the RULE SETUP page shown in Figure 2.

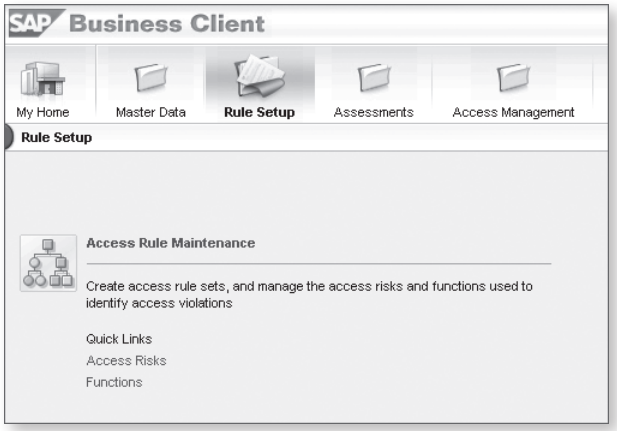


Figure 2 Only Access Risks and Functions Links Allowed

By following this example, if you try to manage an access risk that isn't contained in the AGLEA rule set, an authorization error message will appear (see Figure 3). By clicking on Access Risks, you go into the Access Risk maintenance interface (❶, see Figure 3). After you are in this interface, select the risk (❷), and click on the Open button (❸) if you have the correct authorization; otherwise, a warning message appears (❹), so you can only display the risks.

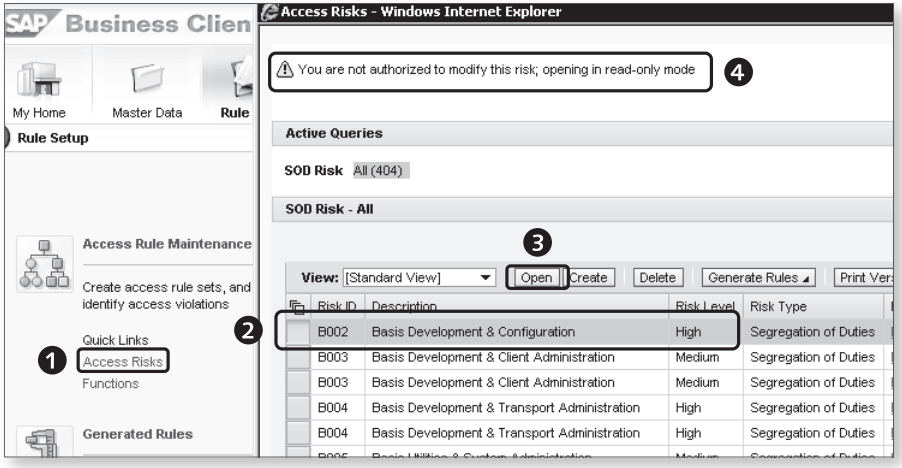


Figure 3 Manage Access Risks Not in the AGLEA Rule Set

You can see the authorization error message through an authorization trace Transaction ST01 or by analyzing the application system message log.

In SAP GRC 10.0, you can read the application message log through Transaction SLG1. Figure 4 shows the selection criteria to retrieve the log application authorization check message. You need to fill in the OBJECT and SUBOBJECT fields, enter the user ID or the time restriction, and press F8.

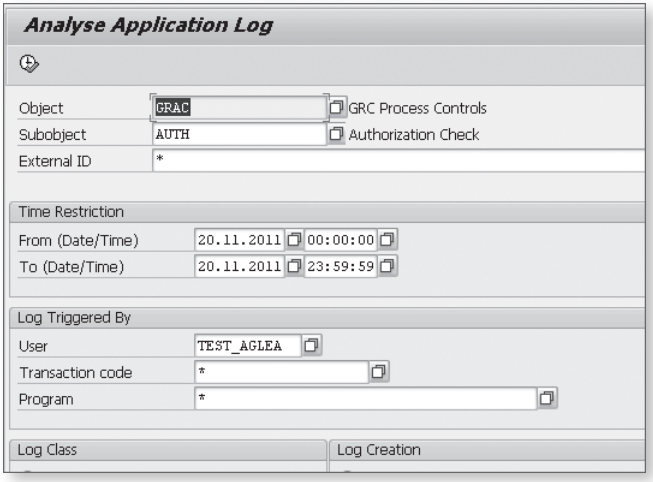


Figure 4 Transaction SLG1 Screen

Tip 69

Keeping Your Mitigation Tables Clean and Accurate with the Invalid Mitigation Report

You can keep your mitigation tables clean and updated with a standard SAP GRC report.

If you mitigate a role or a user and these objects types are removed from the back-end system, the mitigation on these objects remains in the mitigation tables of SAP GRC. Through an SAP GRC standard report, which is available in both the 5.3 and 10.0 releases, you can prevent the system from having these types of entries in mitigation tables to maintain a clean system. This report helps you identify these cases and decide whether to maintain or remove these dummy entries. See SAP Note 1492227 for information about the logic of this report.

And Here's How ...

Mitigation control is defined on some objects such as roles, users, profiles, or organization units. A mitigation control has a validity date. If the objects on the validity date are defined, the control is deleted, but the mitigation control isn't automatically removed from the mitigation tables. The same holds true if a mitigation control expires.

In that case, you can use the standard report Invalid Mitigation Controls in the Risk Analysis interface to identify and remove these instances. Let's go through the steps to use this report.

In both SAP GRC 5.3. and 10.0, you can execute this report within the Risk Analysis interface at the user and role level. Figure 1 shows the Invalid Mitigating Con-

trols report in SAP GRC 10.0 during a risk analysis on the role level. In this example the system, role name, and rule set have been entered. You want to know if there are invalid mitigations defined on role TEST_AG in system ECC_G10_810. To do this, check the Mitigation Analysis check box and select Invalid Mitigating Controls.

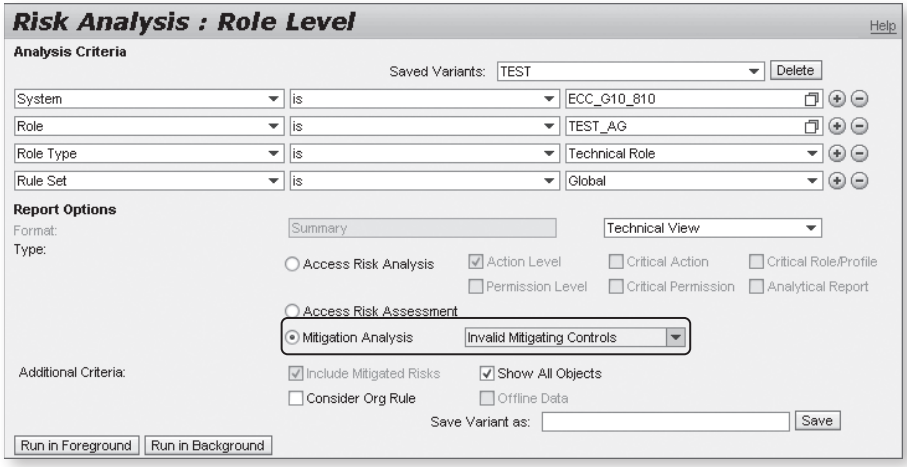


Figure 1 Risk Analysis Role Level: Invalid Mitigating Controls Checkbox

Figure 2 shows the same report in User Risk Analysis in SAP GRC 5.3. It's similar to the SAP GRC 10.0 version, where you can enter the user name and then specify the System, the Validity Date, and the Report Type (in this case, Invalid Mitigation Controls). Click on Execute.

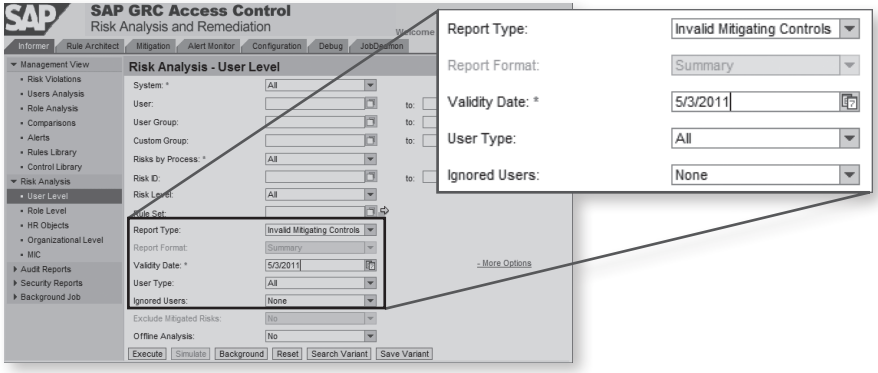


Figure 2 Invalid Mitigating Controls in SAP GRC 5.3

You can see the result of the executed report from SAP GRC 10.0 in Figure 3. You can decide to perform this report monthly to keep your SAP GRC system clean.

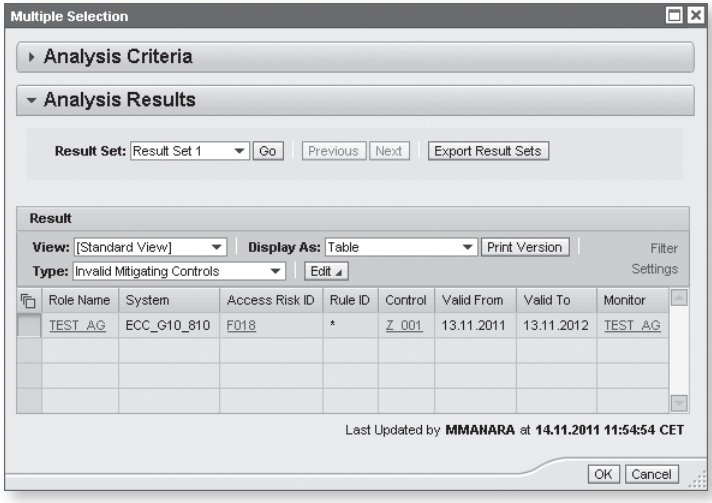


Figure 3 Invalid Mitigation Controls on Role TEST_AG

In this way, you can quickly determine whether this role (or generally, a SAP GRC object) has been removed or whether the control is expired. Through the Edit button, you can decide whether you want to change the control or remove it. Bear in mind that in SAP GRC 10.0, mitigation controls are defined at the system ID level. On the same user or role you could have some invalid mitigation controls on a system and others valid on other systems.

Part 5

Upgrades

Things You'll Learn in this Section

A release upgrade project is not a day-to-day task, but it's necessary in some scenarios to update and improve the system. During an upgrade of one release to another, some transactions in program functions may change, and the authority check in the programs may introduce new authorization objects. Consequently, you'll also have to update your authorization roles.

Depending on the time between your system implementation and target release to upgrade, a project could be relatively simple or difficult (if there's a long time between the start and target release). An unexpected pain point during an SAP authorization upgrade is the time needed to update the roles. If you've defined a certain authorization concept by following smart tips and best practices, you can save a lot of time; otherwise, you may have difficulty when updating your roles and decide to redefine all your roles rather than update.

The goal of this part of the book is to give you some advice to follow during an SAP authorization upgrade process—basically to use Transaction SU25. Knowing a possible problem that may arise during an upgrade could influence you to make a different decision about how to define your authorization concept, saving you time, money, and many headaches.

Tip 70

Making Your Roles Compliant with Transaction SU25

Because Transaction SU25 works on all of your roles, you must first understand the logic behind Transaction SU25 and prepare your roles in advance so that they are compliant with a release upgrade.

Every SAP system will be involved with an upgrade. Authorizations in particular, however, are often not fully considered by project leaders in technical upgrades. The most frequent question that the customer asks is "Why should we modify authorizations?" The answer is simple—new authorization objects have been introduced, and new transaction codes are now available.

Transaction SU25 is dedicated to the roles upgrade. Before you can use it, however, you must check your roles to be sure that they are ready to be updated. Otherwise, the result could be worse than before.

And Here's How ...

This tip assumes that your goal is to upgrade roles. To do this, you will perform the four subactions in step 2 of Transaction SU25. Here is a short description of all four steps (note that we will go into the details of step 2c in this tip):

- **Step 2a**
  Normally no interaction is necessary. In this subaction, the system will perform internal checks mandatory for the following subactions.
- **Step 2b**
  The customer tables are updated with the new SAP default values. It's important

to understand that this action will preserve the previous maintenance of the customer tables.

- **Step 2c**
  This is the most important subaction, and it will adjust all roles according to the new customer table values. The authorization objects are considered in this subaction.
- **Step 2d**
  In this subaction, you can manage the new transaction codes.

Step 2c is the most critical because it will consider all roles in the system, and you can't perform a roll-back action. For this reason, you should perform a mass download of all roles (using Transaction PFCG) before starting Transaction SU25.

Transaction SU25: Step 2c

The logic of step 2c is simple: For each transaction specified in the role, all default authorizations (according to the updated customer tables) not yet present will be inserted.

Even though the logic is simple, it makes two important assumptions:

- The transaction codes specified in the role that will be considered by Transaction SU25 are only the ones inserted in the role's menu. This means that all transaction codes inserted manually into the authorizations of the `S_TCODE` object will not be considered.
- Authorizations with a status of CHANGED will not be considered by the system as defaults, and after Transaction SU25 step 2c processing, you will have two (or more) authorizations on the same objects because the default values will be introduced again.

To better understand the first assumption, Figure 1 shows a role with Transaction MM01 inserted in the menu.

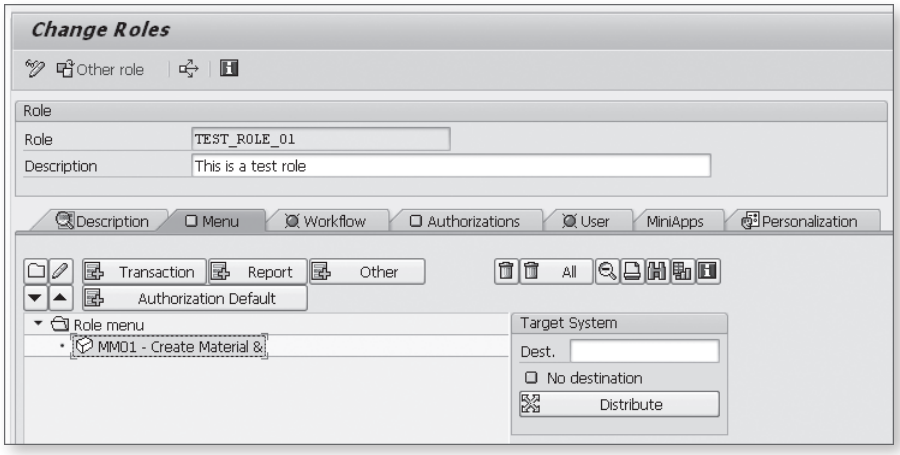


Figure 1 Transaction MM01 in the Menu Role

Figure 2 shows that there are more transactions specified in the S_TCODE authorizations (and we considered this an incorrect action). In the highlighted area, there are other transactions specified through the single value MM02 and the range MM03–MM09. It's important to note that this authorization is marked as MANUALLY.

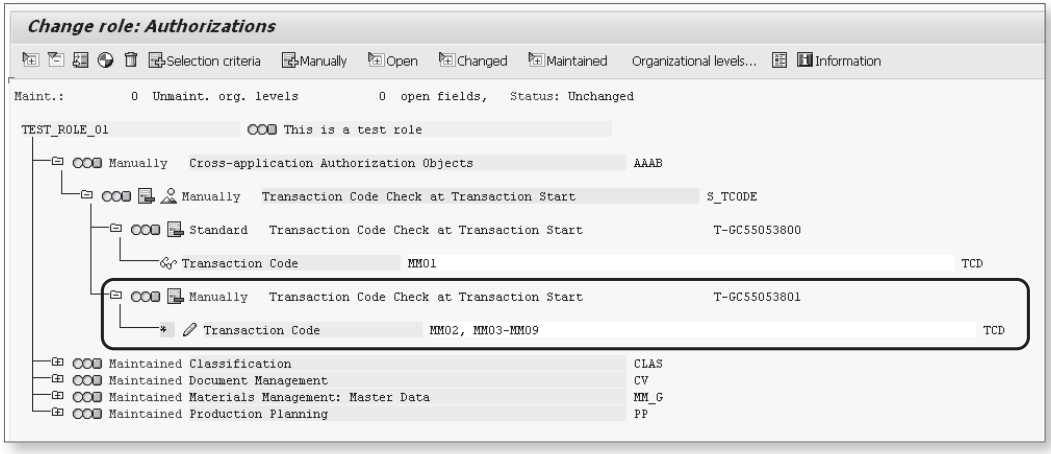


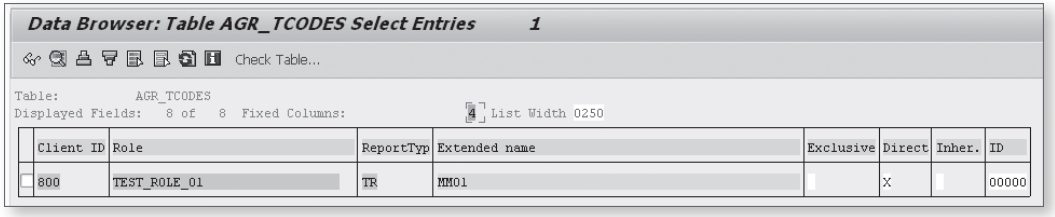
Figure 2 Authorization Tree of Role TEST_ROLE_01 with Manual Authorization on S_TCODE

During the processing of step 2c in Transaction SU25, only Transaction MM01 will be considered. Therefore, you have to adjust the roles before running Transaction SU25.

Make and Check the Compliance of Your Roles for Transaction SU25 Step 2c at the Transaction Code Level (S_TCODE)

To repair this erroneous situation, you must look in Tables AGR_TCODES (Assignment of Roles to Transaction Code in Role Menu) and AGR_1251 (Authorization Data for the Activity Group) to find the differences.

In Figure 3, you can see MM01 in the column Extended Name in the Role menu of the TEST_ROLE_01 role.



Data Browser: Table AGR_TCODES Select Entries 1

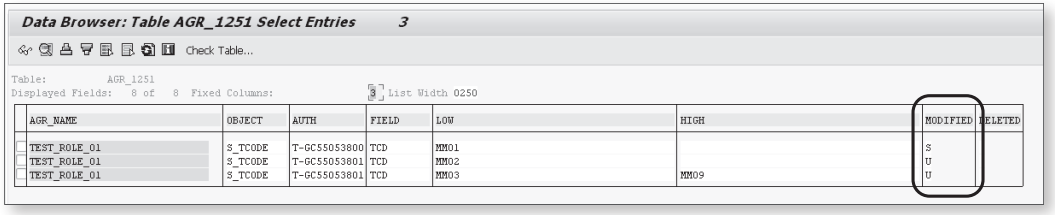
Check Table...

Table: AGR_TCODES
Displayed Fields: 8 of 8 Fixed Columns: 4 List Width 0250

Client ID	Role	ReportTyp	Extended name	Exclusive	Direct	Inher.	ID
800	TEST_ROLE_01	TR	MM01		X		00000

Figure 3 Browsing Table AGR_TCODES to Find the Transaction in the Role's Menu

In Figure 4, you can see the same data from the authorization point of view. In the first row, there is the same record you saw in Table AGR_TCODE. In the second and third rows, you can see the transactions that will not be considered by step 2c of Transaction SU25.



Data Browser: Table AGR_1251 Select Entries 3

Check Table...

Table: AGR_1251
Displayed Fields: 8 of 8 Fixed Columns: 3 List Width 0250

AGR_NAME	OBJECT	AUTH	FIELD	LOW	HIGH	MODIFIED	DELETED
TEST_ROLE_01	S_TCODE	T-GC55053800	TCD	MM01		S	
TEST_ROLE_01	S_TCODE	T-GC55053801	TCD	MM02		U	
TEST_ROLE_01	S_TCODE	T-GC55053801	TCD	MM03	MM09	U	

Figure 4 Using Table AGR_1251 to Find the Transactions Code Specified in the S_TCODE Authorizations

Look for all authorizations on the S_TCODE object that contain a value different from "S" (which means standard) in the MODIFIED field. After you have the list of the transaction codes, you have to verify that they are not yet present in Table AGR_TCODES. Remember that you have to explode each range with the possible values (use Transaction SE16 with Table TSTC). All remaining transactions should be inserted using Transaction PFCG in the role's menu, and consequently, the authorization values must be adjusted.

Make and Check the Compliance of Your Roles for Transaction SU25 Step 2c at the Transaction Authorization Object Status Level

The second assumption noted previously is the most at issue in the roles upgrades. Remember that there are four possible statuses for each authorization as shown in Table 1.

STATUS	VALUE	TEXT
OK	S	Standard
	G	Maintained
NOK	M	Changed
	U	Manual

Table 1 Best Practice Authorization Statuses

If the authorization in the role is marked as "S" (Standard) or "G" (Maintained), during processing of step 2c in Transaction SU25, a new authorization on the same authorization object will be introduced only if the updated customer Table USOBT_C is changed (this doesn't occur frequently).

If the authorization is marked as "M" (Changed) and there are no other authorizations marked as "standard" or "maintained" on the same authorization object, after step 2c of Transaction SU25, you will find many new authorizations that correspond to the default values of the updated customer Table USOBT_C (this occurs frequently).

Many times, the authorization values in the roles are wrong or aren't related to the transaction specified in the role's menu. Many times the organizational fields updated directly from the authorization tree are responsible for the "changed" authorizations and not those updated through the specific pop-up.

Figure 5 shows this error. Here you can see that the field WERKS has the value 1000, and the field description (PLANT) is in dark blue.

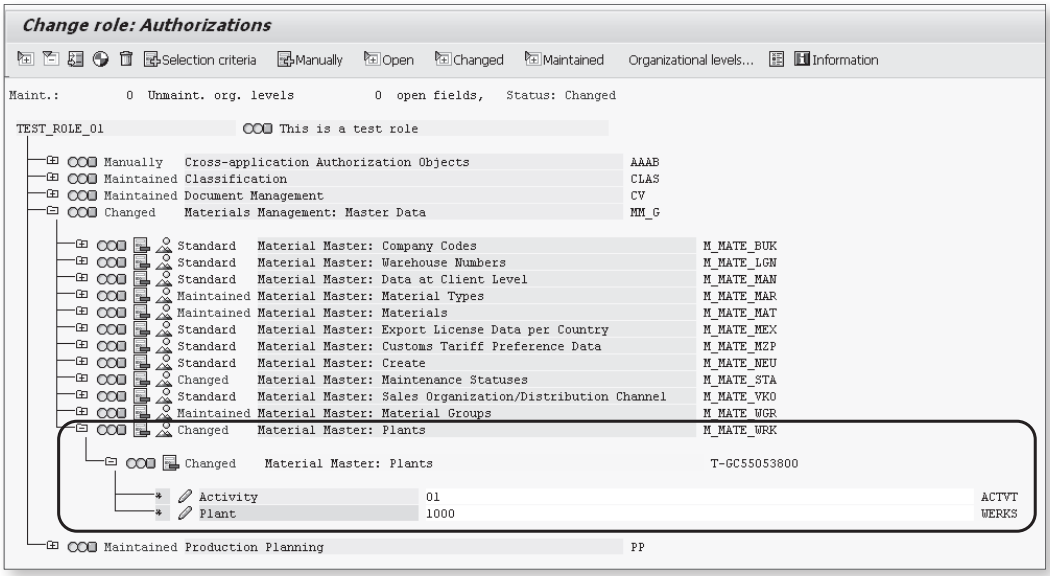


Figure 5 Changed Authorization after Organizational Field Update

You can check the authorizations to have (as much as possible) only authorizations marked as "S" (Standard) or "G" (Maintained) by following these steps:

1. Remove all organizational fields maintained directly in the authorization tree through the Expert Mode for Profile Generation maintenance available in the Authorization tab of Transaction PFCG (Role Maintenance).
2. Review the authorizations values to reflect the standard values or update the standard values using Transaction SU24 (Auth. Obj. Check Under Transactions).

During this analysis, click on the  icon, which allows you to see which authorizations values are really necessary (according to the default values).

Tip 71

Deciding How to Set Up Your Authorization Upgrade

You can avoid overriding customer customizations by understanding how to use the first two steps in the Upgrade Tool for Profile Generator user interface.

Transaction SU25 (Upgrade Tool for Profile Generator) is necessary to perform authorization upgrades and manage customer tables. Although there are six actions (or steps) in the user interface, using all of the actions is not recommended, depending on your business situation. This tip helps you recognize when you should use the first two Transaction SU25 steps. Understanding the difference between these two steps is essential in order to avoid overriding some customer customization that is already in place.

And Here's How ...

Let's discuss when to use the first two actions in the Transaction SU25 interface:

- **Step 1:** Initially Fill the Customer Tables
- **Step 2:** Postprocess the Settings After Upgrade to a Higher Release (this step is formed by four substeps [2a, 2b, 2c, 2d])

Before going into the goal of this tip, it's essential to define how the tables used from Transaction SU25 are managed by SAP. This is important because if you perform step 1 of Transaction SU25 outside the installation of the SAP system or during an upgrade, you can override the customer settings performed by the customer.

ABAP Statement AUTHORITY-CHECK and Customer Tables Switch

When a user starts a transaction code, the application server will start to process the ABAP code associated with it. If the application server finds an `AUTHORITY-CHECK` statement in the source code, it will check Table USOBX_C to see if the function `AUTHORITY-CHECK` has to be performed or not.

How the Customer Tables Are Managed by the Customer

Customer tables are managed through Transaction SU24 (Auth. Obj. Check Under Transactions). Figure 1 shows a print screen of Transaction SU24 in case you decide that Transaction MM03 (Display Material) does not have to perform the `AUTHORITY-CHECK` on the `M_MATE_STA` authorization object.

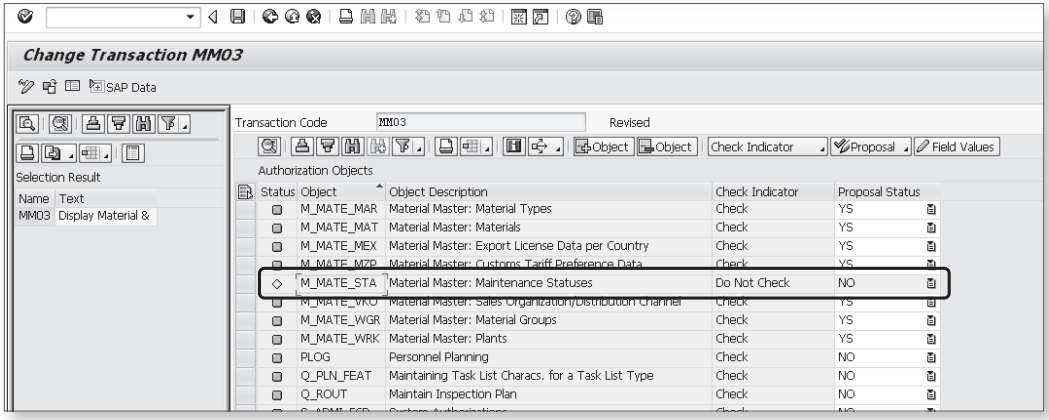


Figure 1 Transaction SU24 on Transaction MM03 Authorization Values

Figure 2 shows the result of the previous activity from the table point of view (Table USOBX_C). As you can see, field OKFLAG is now set to N (previously, it was set to Y).

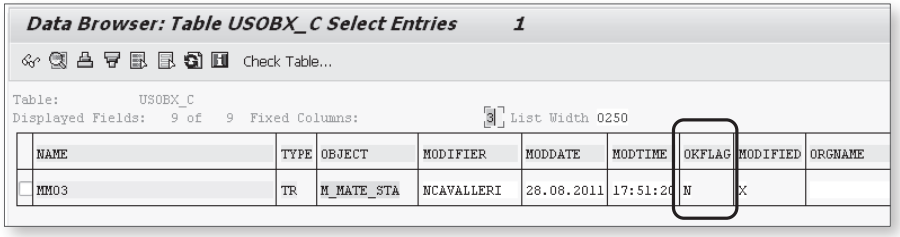
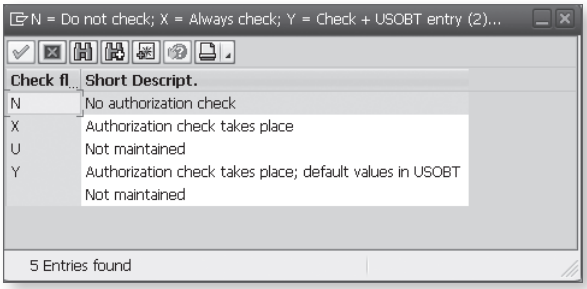


Figure 2 Table USOBX_C Record on Transaction MM03 with M_MATE_STA Authorization Object and OKFLAG Set to X

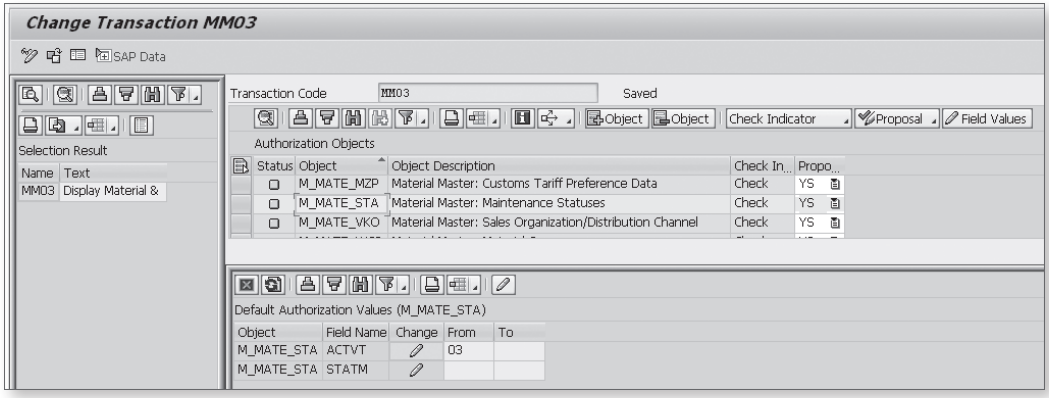
Figure 3 shows the possible values for the OKFLAG field of customer Table USOBX_C.



« *Figure 3 OKFLAG Values*

There are five entries in Figure 3. The fourth record, Y, is very important because it's related to Table USOBT_C. In fact, when the value is set to Y (AUTHORIZATION CHECK TAKES PLACE; DEFAULT VALUES IN USOBT), Transaction PFCG (Role Maintenance) will create a default authorization for the related "transaction/authorization object" with the default values retrieved in Table USOBT_C (after you add a transaction in the role's menu).

Figure 4 shows the default values for coordinate MM03 – M_MATE_STA (before you deactivate it). This data is contained in Table USOBT_C.



» *Figure 4 Transaction SU24: Default Values Maintenance*

Now let's consider the difference between Transaction SU25 step 1 and step 2.

Step 1: Initially Fill the Customer Table

Step 1 is mandatory only once (normally in the post-installation phase and just before you create a new role). When you run this step, the system will copy the SAP default values from Tables USOBX and USOBT to the corresponding customer tables (Tables USOBX_C and USOBT_C). Only after this action will Transaction PFCG (Role Maintenance) be ready to be used because it will create the authorizations with the updated default values.

Every time you use Transaction SU24 for adjusting some values, you'll update the customer tables. For this reason, you shouldn't run step 1 again because this action will start from scratch, and all maintained values will be deleted.

Step 2: Postprocess the Settings after Upgrade to a Higher Release

When you need to upgrade the roles to a newer SAP release, you'll use step 2 of Transaction SU25 (directly in the new SAP system), which is composed of four substeps (a, b, c, and d). Step 2b is related to the authorization customer tables (Tables USOBX_C and USOBT_C) because it will perform the same action as step 1 (i.e., copy Tables USOBX and USOBT to customer tables) but will preserve your maintained values (unlike step 1 of Transaction SU25). See Tip 70, which is related to step 2c, and Tips 74 and 75 for step 2d.

Tip 72

Managing Derived Roles during an Upgrade

When using Transaction SU25 and exploiting the role derivation, you can reduce the time and effort needed to upgrade the derived roles.

Many authorization concepts are based on the composite->simple role architecture, where the simple roles can be derived. Transaction SU25 (Upgrade Tool for Profile Generator) step 2 (a, b, c, d) is recommended when the roles are compliant with the prerequisites (see Tip 70). Although there is no effort required to upgrade the composite roles, you have to process (check) all "impacted" simple roles in step 2c of Transaction SU25. The amount of roles to be processed could be very huge. To save a lot of time, you can process only the parent role of derived roles in Transaction SU25. Then avoid processing all derived roles in a singular manner.

And Here's How ...

Step 2c in Transaction SU25 will list all roles to be checked to add new authorizations (see Figure 1). When derived roles are used, parent roles and all derived roles are listed in step 2c. This list may be very long if you use several derived roles.

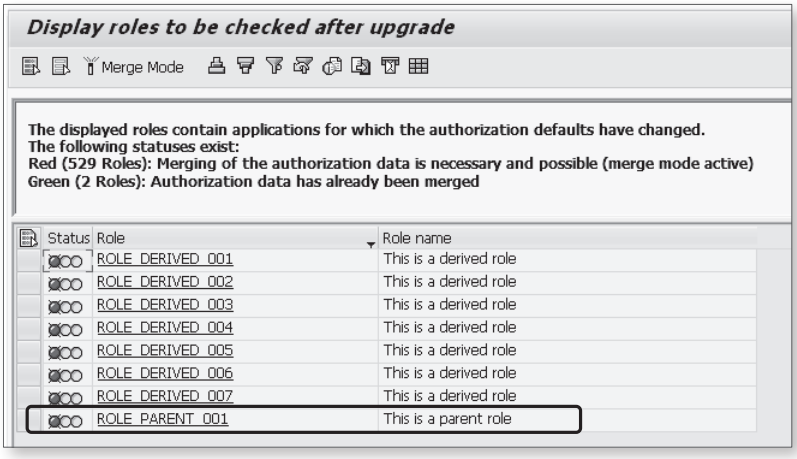


Figure 1 Transaction SU25 Step 2c Output

You don't need to process all derived roles in step 2c. You can process only parent roles and then adjust the derived roles. Click on the GENERATE DERIVED ROLES button in the parent roles, as shown in Figure 2.

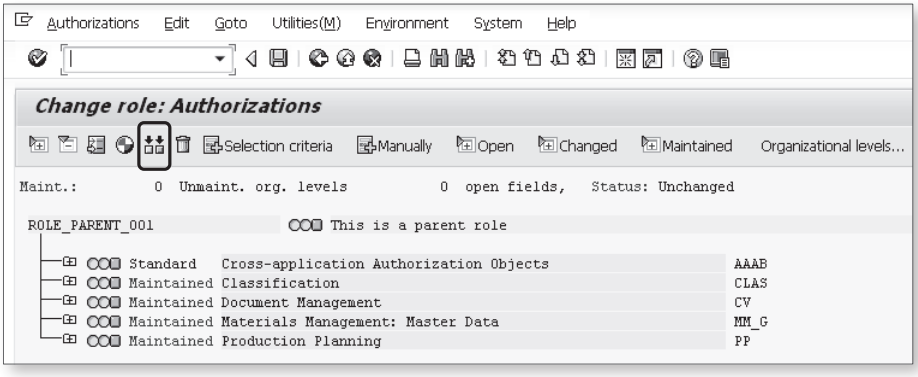


Figure 2 Generate Derived Roles Feature

After you've clicked on the button, the system considers the roles "processed." The role is then marked as green in the step 2c list as shown in Figure 3.

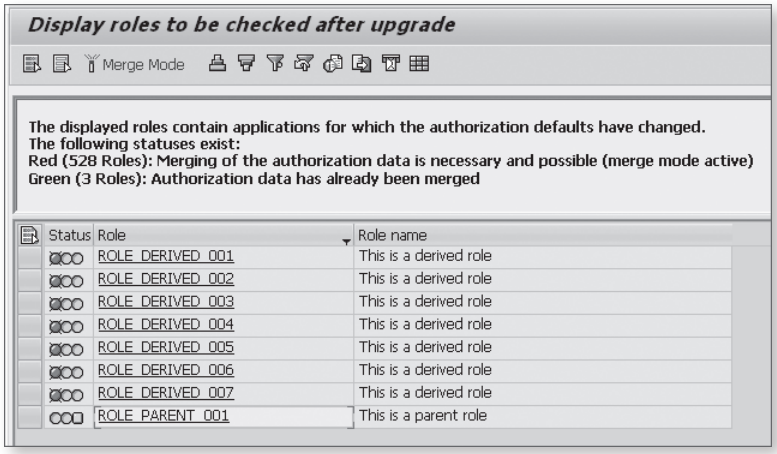


Figure 3 Checked Role's Status

All derived roles are still marked as red, but if you run step 2c again, they will disappear.

In the example, we were able to check eight roles (one parent and seven derived) just by working on one role.

In our experience, many companies use the derived roles but only for transaction inheritance. This means that each derived role has its own authorization tree that is different from the parent role not only for the organizational values. In such cases, you shouldn't use this technique because the parent's authorizations will be propagated to all derived roles (related to this parent role) when the GENERATE DERIVED ROLES button is clicked on, and all previous authorization values will be overwritten.

Tip 73

Converting a Manually Created Profile into a Role

You can use the "Optimized" or "Identical to profile" options in Transaction SU25 to convert manually created profiles into roles.

Before Transaction PFCG (Role Maintenance; also called Profile Generator) was introduced, authorizations were manually managed with Transaction SU03 (Maintain Authorizations), and the profiles were manually managed with Transaction SU02 (Maintain Authorization Profiles). The main issue in manual management was that the security managers didn't know which authorization objects were related to a specific transaction code.

SAP has delivered a special feature, available in Transaction SU25 (Upgrade Tool for Profile Generator), which is able to convert a profile that was manually created into a role. This feature is available through step 6, and gives you two different approaches to convert a profile into a role.

And Here's How ...

You can display a manually created profile, as shown in Figure 1, by executing Transaction SU02.

This profile just contains an authorization on object `S_TCODE` for Transaction MM01 (Material Master Data) and nothing else. All other authorizations (related to Transaction MM01) are probably distributed in other profiles.

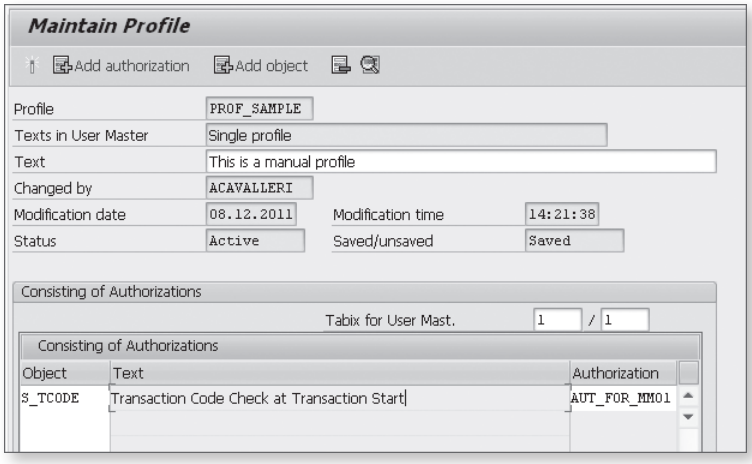


Figure 1 Transaction SU02: Display Manual Profile

Access Transaction SU25 and go to step 6: Create Roles from Manually Created Profiles. If you start step 6, you'll see all manual profiles available in the system. Figure 2 shows the first screen (after applying a filter to find the profile name, which in this example, is only one) of step 6. Notice that there are two possible options to convert the profiles: Optimized and Identical to profile.

Figure 2 Step 6 Methods

The best option for you will depend on your security concept (in this tip, it's based on manual profiles), but we suggest using Optimized instead of Identical to profile because you will benefit from the Transaction SU24 (Auth. Obj. Check Under Transactions) logic.

Optimized Method

If you click on the Optimized button to convert the profile, you'll see a role's menu (see Figure 3). The new role now contains all default authorizations related to Transaction MM01.

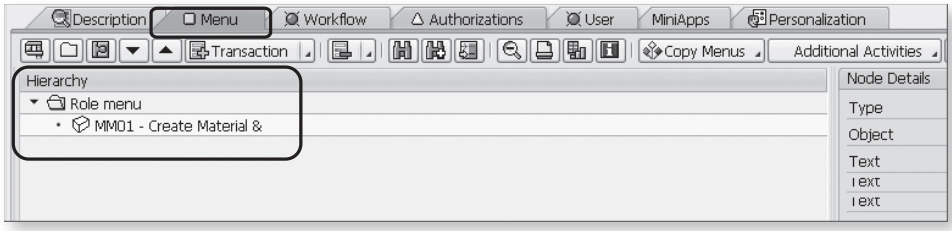


Figure 3 Role's Menu of Converted Manual Profile

In Figure 4, notice all of the branches for the newly introduced authorizations.

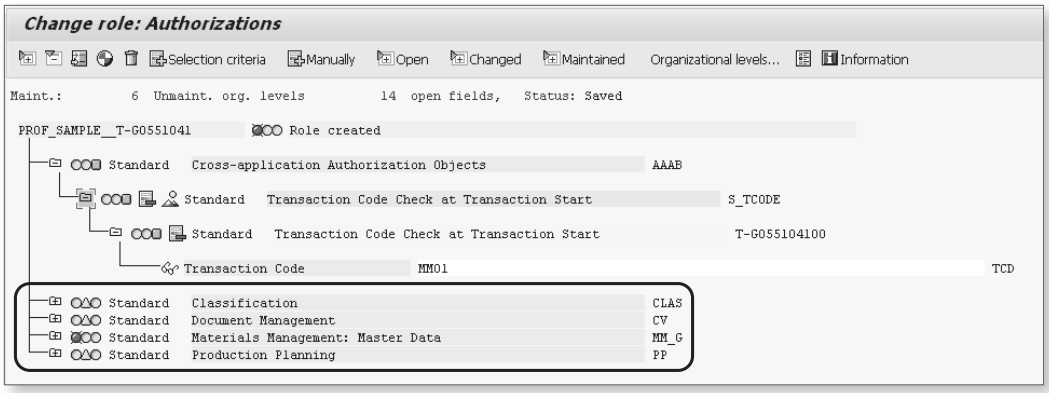


Figure 4 New Authorizations in the Authorization Tab of Transaction PFCG

This method is useful when you want to switch your authorization concept from a manual profile to a roles-based profile.

Identical to Profile Method

The IDENTICAL TO PROFILE option does not introduce anything. The role's menu will be empty, and the authorization tree will just contain the unique authorization found in the manual profile. Only authorization S_TCODE is present in the role's authorization tree. This method could be useful when you want to convert a profile to a role but maintain the same authorization data. This option is generally used to convert a profile assigned to a technical user ID with particular authorization requirements.

Tip 74

Avoid Maintaining a Role's Authorization Tree Twice When New Transaction Codes Are Added

You can use the steps of Transaction SU25 out of order if you want to avoid maintaining the authorization tree twice.

Every time a new transaction code is inserted into a role's menu, the authorization values must be adjusted. To avoid maintaining the authorizations tree for the role twice, you can run step 2d prior to step 2c. In this way, you'll reduce the total effort of using step 2 in Transaction SU25.

And Here's How ...

Occasionally, SAP introduces new transaction codes. The frequency depends on the distance between the old release and the new release. For instance, if you upgrade an SAP R/3 4.0b system to an SAP ERP 6.0 EHP5 system, the number of transaction codes introduced is bigger than what you'll see in an upgrade from SAP ERP 6.0 EHP5 to EHP6.

Access Transaction SU25 to see the description Display Changed Transaction Codes for step 2d (see Figure 1). Note that rather than a "display," this step will change the role's content.

Click on step 2d, and after few moments, a list of involved roles will appear. In this example, as you can see in Figure 2, there is a status indicator in the first column: red is for roles that haven't been processed, and green is for processed roles. If you

select a role, you can decide which action must be performed by clicking on one of the three buttons marked in the upper part of Figure 2.

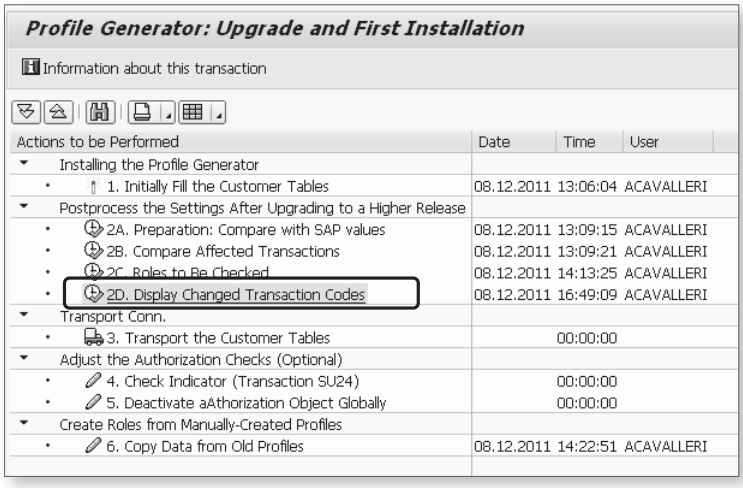


Figure 1 Step 2d of Transaction SU25

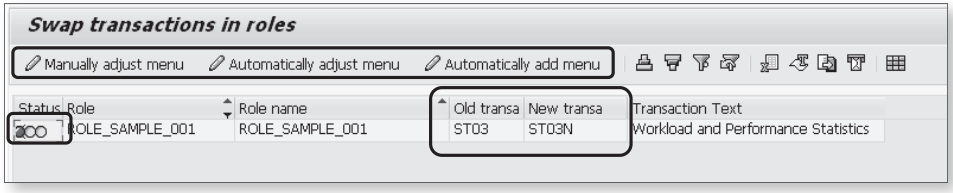


Figure 2 Step 2d Output

Now look at the OLD/NEW TRANSACTIONS columns. In the example, the old Transaction ST03 code has been replaced by the new Transaction ST03N code. This means that the SAP system has replaced a transaction with a new one.

Independently from which action you will perform, by using all of the circled buttons at the top of Figure 2, you can make the system open Transaction PFCG (Role Maintenance) with the selected role loaded. Because you will change the role's menu, you should also go to the authorization tree to maintain the new authorization values.

You can now avoid the maintenance of the authorization tree here because you can do the same action (later) in step 2c. Of course, this makes sense only if you postpone step 2c after step 2d.

Tip 75

Identifying New Transactions in a Role's Menu

You can easily share new transaction codes with the functional analysts so they can decide whether to introduce a new transaction to the users.

From a technical point of view, you can use step 2d of Transaction SU25 to introduce new transaction codes in a role's menu. Keep in mind, however, that it's not always recommended that you introduce new transaction codes because there are many different aspects of this process that need to be considered. For example, if the users don't know how to use the new transactions (due to lack of training), they will be in trouble.

Because an upgrade project doesn't just involve people who deal with authorizations, you should share (in advance) the new transaction codes list with the functional analyst teams to let them verify whether the new transaction code can be introduced. This tip will show you how to easily access and share this list.

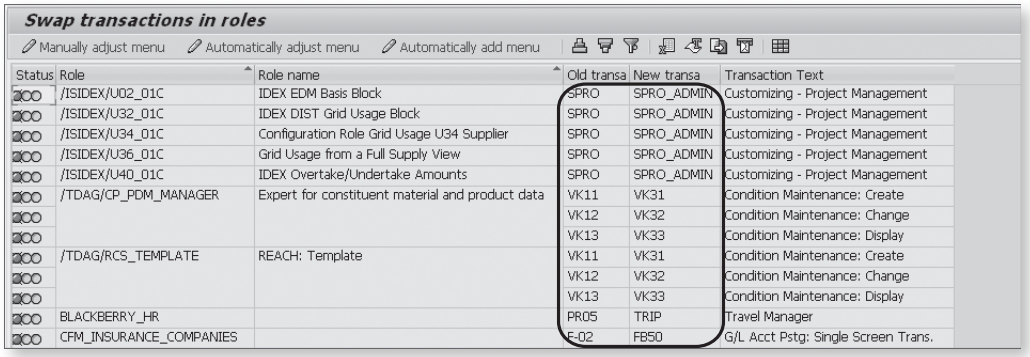
And Here's How ...

New transaction codes are introduced for three reasons:

- An old transaction code has been replaced by a new one (1:1).
- Many old transaction codes have been replaced by a new one (*n*:1).
- An old transaction code has been replaced by many new transactions (1:*n*).

Step 2d of Transaction SU25 (Upgrade Tool for Profile Generator) is simple: For each transaction found in a role's menu (i.e., Table AGR_TCODES), the program will look into Table PRGN_CORR2 if there is a mapping transaction.

As you can see in Figure 1, in the proposal of step 2d, there are two columns for the transactions: the old and the new.



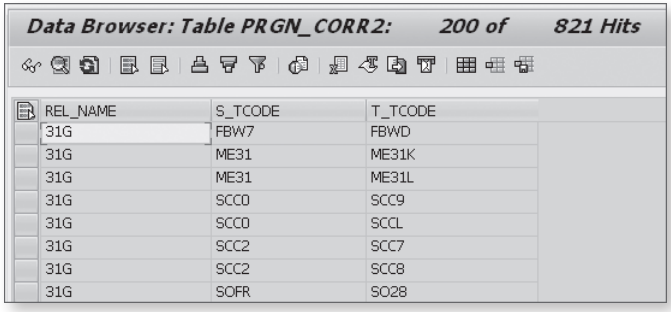
Swap transactions in roles

Manually adjust menu | Automatically adjust menu | Automatically add menu

Status	Role	Role name	Old transa	New transa	Transaction Text
	/ISIDEX/U02_01C	IDEX EDM Basis Block	SPRO	SPRO_ADMIN	Customizing - Project Management
	/ISIDEX/U32_01C	IDEX DIST Grid Usage Block	SPRO	SPRO_ADMIN	Customizing - Project Management
	/ISIDEX/U34_01C	Configuration Role Grid Usage U34 Supplier	SPRO	SPRO_ADMIN	Customizing - Project Management
	/ISIDEX/U36_01C	Grid Usage from a Full Supply View	SPRO	SPRO_ADMIN	Customizing - Project Management
	/ISIDEX/U40_01C	IDEX Overtake/Undertake Amounts	SPRO	SPRO_ADMIN	Customizing - Project Management
	/TDAG/CP_PDM_MANAGER	Expert for constituent material and product data	VK11	VK31	Condition Maintenance: Create
			VK12	VK32	Condition Maintenance: Change
			VK13	VK33	Condition Maintenance: Display
	/TDAG/RCS_TEMPLATE	REACH: Template	VK11	VK31	Condition Maintenance: Create
			VK12	VK32	Condition Maintenance: Change
			VK13	VK33	Condition Maintenance: Display
	BLACKBERRY_HR		PR05	TRIP	Travel Manager
	CFM_INSURANCE_COMPANIES		F-02	FB50	G/L Acct Pstg: Single Screen Trans.

Figure 1 Transaction SU25 Step 2d

If you browse Table PRGN_CORR2, as shown in Figure 2 you can see the mapping from the old to the new transaction codes. Note that the first column contains the SAP release in which the transaction code was changed. You can now share this list of transactions with the appropriate groups.



Data Browser: Table PRGN_CORR2: 200 of 821 Hits

REL_NAME	S_TCODE	T_TCODE
31G	FBW7	FBWD
31G	ME31	ME31K
31G	ME31	ME31L
31G	SCC0	SCC9
31G	SCC0	SCCL
31G	SCC2	SCC7
31G	SCC2	SCC8
31G	SOFR	SO28

Figure 2 Table PRGN_CORR2

Make sure that before you start step 2d of Transaction SU25, you update Table PRGN_CORR2 according to the OSS Notes at *http://service.sap.com/notes*.

Even though the logic is simple, there are situations in which the old transaction code is linked to a new ABAP program (and the previous ABAP program is linked to a new transaction code). For example, Transaction PPOM (Maintain Organizational Plan) is the Enjoy SAP version of Transaction PPOM_OLD (Maintain Organizational Plan).

Tip 76

Communicating Password Requirement Changes During SAP Upgrades

In the first days of go-live with a new release, password issues represent most of the help-desk calls. You can avoid this issue by communicating the changed requirements to end users.

New password rules started from SAP NetWeaver 2004s ABAP stacks and SAP ERP 6.0 EHP 5. Passwords can be from 3 to 40 characters in length, support case-sensitivity (uppercase and lowercase characters are distinguished), and use a SHA1 algorithm for hashing instead of MD5. No changes on system profile parameters for password rules are necessary, but many users are not able to log in to the upgraded system anyway. This tip shows you how to alert your end users to changing rules so they can maintain their passwords in a timely manner.

And Here's How ...

In previous SAP systems, when a user changed the password he could insert a string with more than eight characters and with lowercase and uppercase characters even if the system considered only non-case-sensitive passwords of eight characters maximum length. In other words, the end user thought that the new password had been created successfully, but it would be converted to uppercase and truncated at eight characters. For example, a user may enter "morningsun" as a password with the result of MORNINGS.

In the first days of go-live with the upgraded system, the new logic becomes valid (case sensitive and up to 40-character in length); however, for these smart end

users, the previous valid password stored in the system is still uppercase and 8 characters in length.

Figure 1 shows the SAP logon screen; from the end user point of view, nothing is changed.

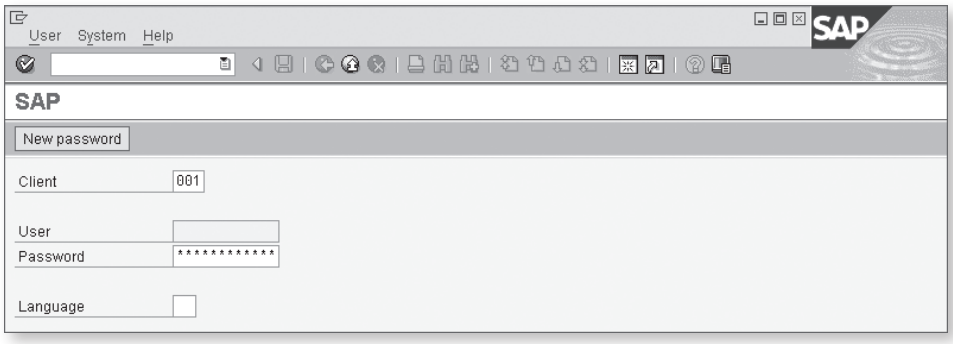
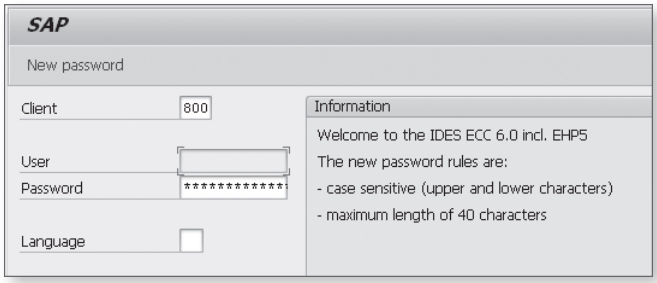


Figure 1 SAP Logon Screen

With the upgraded system, the end user will try to log in to the system with a password having (in his mind) lowercase and uppercase characters and more than eight characters, but he will not be able to access his account because the previous password is different. Probably, he will not understand why and will try again until the user ID is locked.

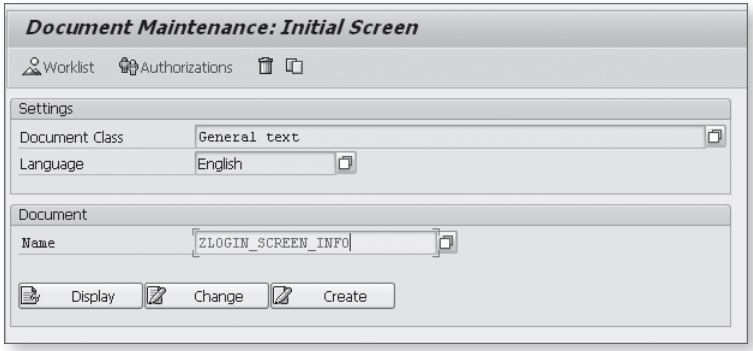
You can bypass the user's problem using the password profile parameters to set a minimum number of uppercase characters and a minimum length—`login/min_password_uppercase` and `login/min_password_lng`—but this will not benefit the new security opportunity.

Our suggestion is to clearly communicate to end users in advance the new password rules. The best way is to add a static message on the SAP GUI logon screen as shown in Figure 2 or publish the new password rules on your intranet if you use a Single Sign-On authentication method.



« *Figure 2 Display Password Rules on the Logon Screen*

To insert an information message in the SAP logon screen like the one shown in Figure 2, use Transaction SE61. You can customize the element ZLOGIN_SCREEN_INFO in the NAME field, and enter "General text" in the DOCUMENT CLASS field as shown in Figure 3. After that, you have to make the change by clicking on the CHANGE button and entering your own text.



» *Figure 3 Customizing the Logon Screen Message*

However, remember that not all users are subjected to password rules, such as service and system user types.

Part 6

Auditing

Things You'll Learn in this Section

The authorization process is always ongoing. After you've defined a good authorization concept, you need to periodically check role authorizations and roles assignments to prevent authorization and security holes or errors. By doing this, you'll be able to find where and when the weaknesses of your authorization concept arise to improve and fortify the authorization concept. You can often accomplish this by using the SAP change documents.

In this part of the book, we'll show you how to improve and control the authorization concept that's in place and avoid some common problems during a security and authorization audit phase of an internal or external audit. You'll also find ways to speed up your audit phase and make some self-audits.

Tip 77

Searching for Roles or Users Using Transaction SUIM with Asterisk Searching

You can use Transaction SUIM to perform several types of searches on users and roles.

Using Transaction SUIM, you can perform several types of searches mainly for users and roles; for example, you can find all executable transactions for a user or a role. However, you have to pay attention when you use the special character asterisk (*). When using this transaction with asterisk values, a common problem is retrieving data not related to your search. In this tip, you will learn how to avoid this issue.

And Here's How ...

Execute Transaction SUIM by following this path in the SAP standard menu:

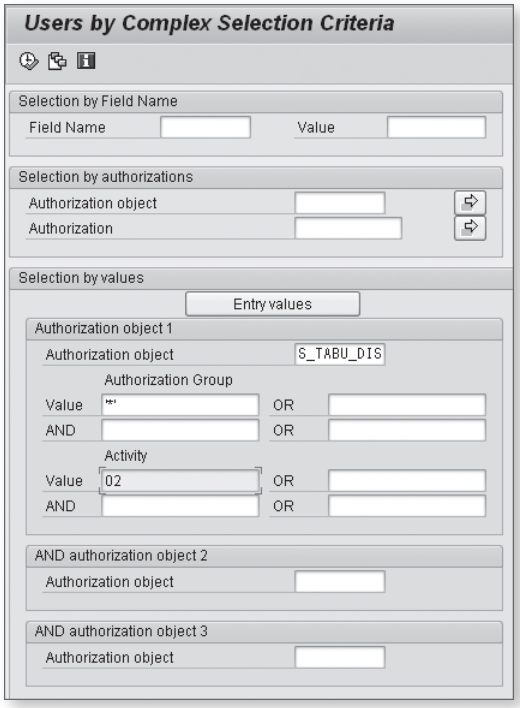
Tools • Administration • User Maintenance • Information System

Because Transaction SUIM is a menu tree formed by several other transactions, this path allows you to execute all of these subtransactions.

For the purpose of this tip, we're executing Transaction S_BCE_68001400, which is located in the Users According to Complex Criteria tree. In this transaction, you can find all users according to your criteria search.

Suppose you want to find all users that contain the authorization object `S_TABU_DIS` with ACTVT field value set to "02" (edit) and with the DICBERCLS field

(authorization group) set with an asterisk. To accomplish this, first maintain the fields for Transaction S_BCE_68001400 as shown in Figure 1.



« *Figure 1*
Transaction S_BCE_68001400: Selection by Values Form

Enter the authorization object that you're looking for in the AUTHORIZATION OBJECT field (for this tip, enter "S_TABU_DIS"). Press Enter, and the system proposes the field defined for this authorization object where you have to enter your search values.

Note that if you're looking for the exact value of the asterisk, you have to enter the value * between single quotation marks like this: '*' in an AUTHORIZATION OBJECT field. If you enter only the asterisk, the search displays all values.

As another example, let's say you're looking for the exact value of S*. In this case, these characters should always be placed in single quotation marks, such as 'S*'.

You can also use this search to retrieve all roles or users that contain an asterisk in the value searched.

Tip 78

Using the Security Audit Log to Manage Your Super Users' Access

During emergency activities you can temporarily allow user access and trace critical user IDs' activity with the SAP Security Audit Log.

If your company doesn't use a tool such as SAP BusinessObjects governance, risk, and compliance solution (SAP GRC), you can use the Security Audit Log to put critical or emergency access users under a trace. In this way, you log the activities performed during the use of these super user IDs. Because of Segregation of Duties, the IT department/people should generally not be allowed to perform business activities. But in some emergencies and isolated cases, the IT team should correct or unlock a business-critical issue. In those cases, you can define a set of critical users who are allowed to be logged in to the system to improve your audit during these emergency activities.

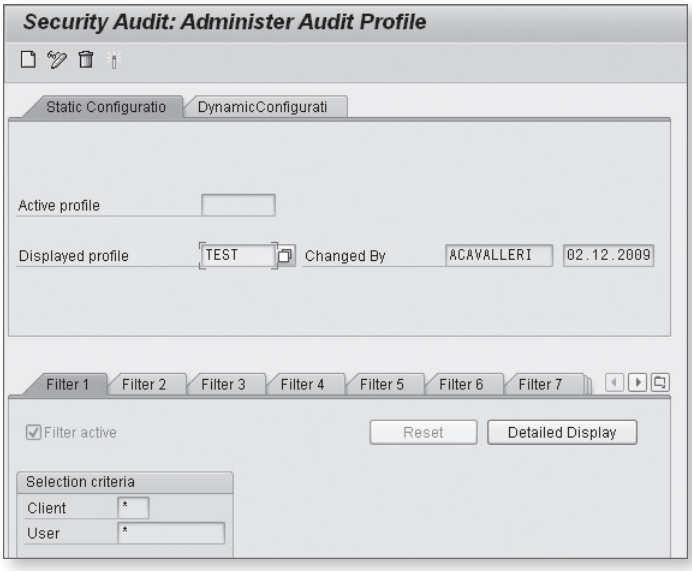
And Here's How ...

To enable the Security Audit Log, you must set some instance profile parameters. These activities are performed by your system administrator. Some of the most important parameters are:

- `rsau/enable`: Allows you to enable the Security Audit Log.
- `rsau/selection_slots`: Represents the number of available slots to insert users under trace.
- `rsau/user_selection`: Allows you to define the user selection method through a wildcard.

After enabling these parameters and rebooting your system, use Transaction SM19 to configure the Security Audit Log (see Figure 1). You need to decide how many slots to activate and which type of filter to use (dynamic or static). There are two tabs at the top of the screen: Static Configuration and Dynamic Configuration. The main difference is that the static configuration remains active when you reboot the system. Dynamic configuration is useful only for spot-tracing activities.

All Filter tabs in the middle of Figure 1 represent all of the available slots where you can enter a specific user ID to put under trace for a client or a wildcard user.



« *Figure 1*
Transaction SM19 Interface

You can also trace data items such as dialog logon, RFC logon, transaction and report start, user master data change, and other system and security events. During the filter definition, you can flag which of these logs you want to trace in Transaction SM19.

With further system administration activity and support, you can enable an alert message when a Security Audit Log event happens (e.g., when a user logs on, an alert e-mail, or SMS is sent to a security administrator).

Logged Data

You can display the logged data with Transaction SM20N. This transaction collects your log file when more than one application server is defined (see the left side of Figure 2). You can also define a time period restriction and an event base selection.

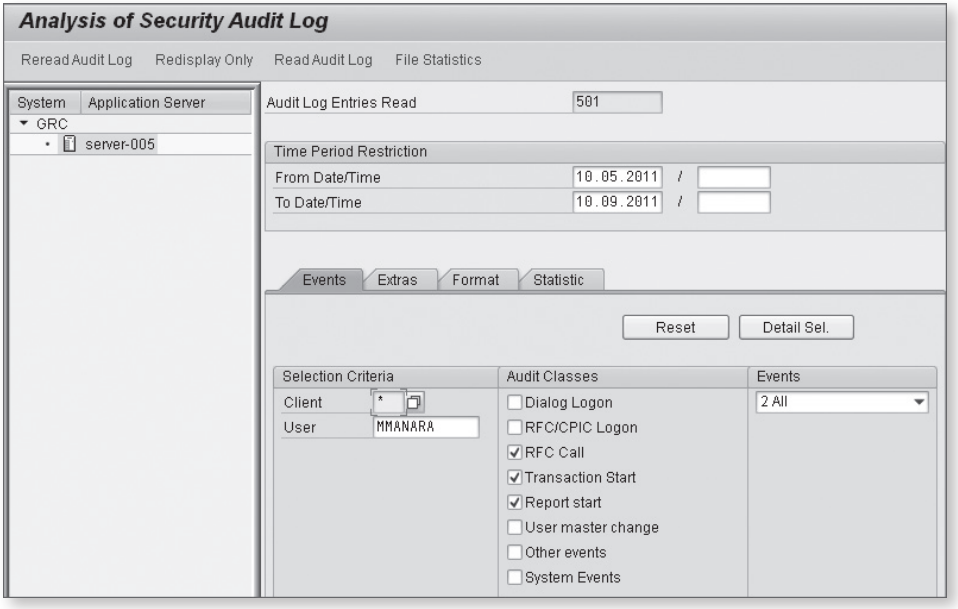
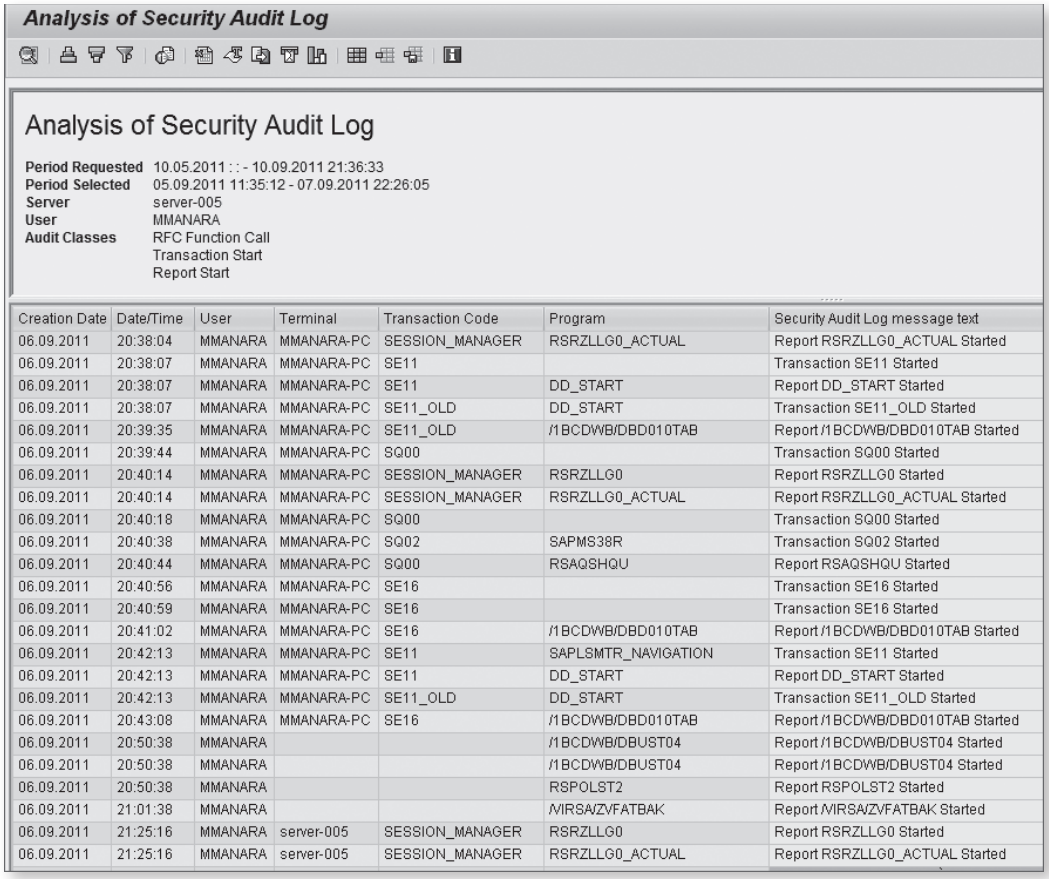


Figure 2 Transaction SM20N: Selection Screen

If you execute the information shown in Figure 2, the data will be exported into a spreadsheet as shown in Figure 3. You can then see and store all activities performed during a session for auditing purposes.

Sometimes a warning message may appear such as "Output terminated by reaching maximum number of pages" during the analysis log evaluation due to the size of the data. In this case, you can augment the Maximum Number of Pages parameters in the Format tab shown in Figure 2 or download the raw text file on the application server with the help of your Basis team.



Analysis of Security Audit Log

Analysis of Security Audit Log

Period Requested 10.05.2011 : : - 10.09.2011 21:36:33
Period Selected 05.09.2011 11:35:12 - 07.09.2011 22:26:05
Server server-005
User MMANARA
Audit Classes RFC Function Call
Transaction Start
Report Start

Creation Date	Date/Time	User	Terminal	Transaction Code	Program	Security Audit Log message text
06.09.2011	20:38:04	MMANARA	MMANARA-PC	SESSION_MANAGER	RSRZLLG0_ACTUAL	Report RSRZLLG0_ACTUAL Started
06.09.2011	20:38:07	MMANARA	MMANARA-PC	SE11		Transaction SE11 Started
06.09.2011	20:38:07	MMANARA	MMANARA-PC	SE11	DD_START	Report DD_START Started
06.09.2011	20:38:07	MMANARA	MMANARA-PC	SE11_OLD	DD_START	Transaction SE11_OLD Started
06.09.2011	20:39:35	MMANARA	MMANARA-PC	SE11_OLD	/1BCDWB/DBD010TAB	Report /1BCDWB/DBD010TAB Started
06.09.2011	20:39:44	MMANARA	MMANARA-PC	SQ00		Transaction SQ00 Started
06.09.2011	20:40:14	MMANARA	MMANARA-PC	SESSION_MANAGER	RSRZLLG0	Report RSRZLLG0 Started
06.09.2011	20:40:14	MMANARA	MMANARA-PC	SESSION_MANAGER	RSRZLLG0_ACTUAL	Report RSRZLLG0_ACTUAL Started
06.09.2011	20:40:18	MMANARA	MMANARA-PC	SQ00		Transaction SQ00 Started
06.09.2011	20:40:38	MMANARA	MMANARA-PC	SQ02	SAPMS38R	Transaction SQ02 Started
06.09.2011	20:40:44	MMANARA	MMANARA-PC	SQ00	RSAQSHQU	Report RSAQSHQU Started
06.09.2011	20:40:56	MMANARA	MMANARA-PC	SE16		Transaction SE16 Started
06.09.2011	20:40:59	MMANARA	MMANARA-PC	SE16		Transaction SE16 Started
06.09.2011	20:41:02	MMANARA	MMANARA-PC	SE16	/1BCDWB/DBD010TAB	Report /1BCDWB/DBD010TAB Started
06.09.2011	20:42:13	MMANARA	MMANARA-PC	SE11	SAPLSMTR_NAVIGATION	Transaction SE11 Started
06.09.2011	20:42:13	MMANARA	MMANARA-PC	SE11	DD_START	Report DD_START Started
06.09.2011	20:42:13	MMANARA	MMANARA-PC	SE11_OLD	DD_START	Transaction SE11_OLD Started
06.09.2011	20:43:08	MMANARA	MMANARA-PC	SE16	/1BCDWB/DBD010TAB	Report /1BCDWB/DBD010TAB Started
06.09.2011	20:50:38	MMANARA			/1BCDWB/DBUST04	Report /1BCDWB/DBUST04 Started
06.09.2011	20:50:38	MMANARA			/1BCDWB/DBUST04	Report /1BCDWB/DBUST04 Started
06.09.2011	20:50:38	MMANARA			RSPOLST2	Report RSPOLST2 Started
06.09.2011	21:01:38	MMANARA			/VIRSA/ZVFATBAK	Report /VIRSA/ZVFATBAK Started
06.09.2011	21:25:16	MMANARA	server-005	SESSION_MANAGER	RSRZLLG0	Report RSRZLLG0 Started
06.09.2011	21:25:16	MMANARA	server-005	SESSION_MANAGER	RSRZLLG0_ACTUAL	Report RSRZLLG0_ACTUAL Started

Figure 3 Transaction SM20N: Audit Log Selection

By using this tool a few times, you can set up a way to control and monitor your super users' access and use. Before enabling this tool, make sure you're in compliance with your national privacy laws.

Tip 79

Changing the Classification of an Audit Log Message

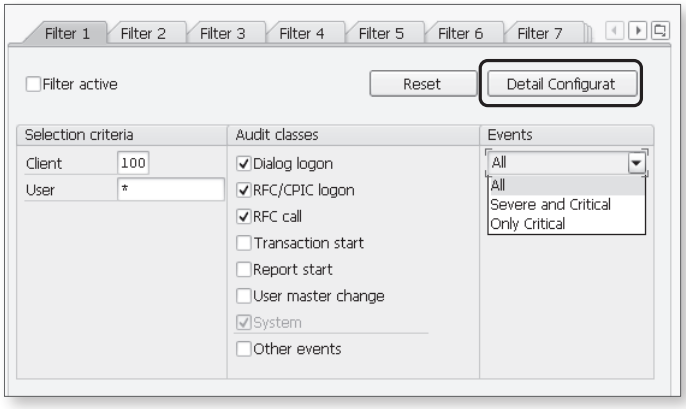
You can customize the audit class and security levels used in the Security Audit Log tool to help monitor a wide range of events.

All logging events are classified in audit classes (e.g., logon, transaction start) and security levels (e.g., critical, important, non-critical). You can customize the classification if the standard doesn't fit your needs (using Transaction SE92). For example, some third-party tracing tools can catch only critical events. This is the case with the user logoff event, which is classified as a logon class and has a security level of non-critical. In this tip, you'll learn how to customize these events using the Security Audit Log feature available through Transactions SM19 (Security Audit Configuration) and SM20N (Analysis of Security Audit Log). You'll use Transaction SM19 to configure the logs and Transaction SM20 to analyze the log content.

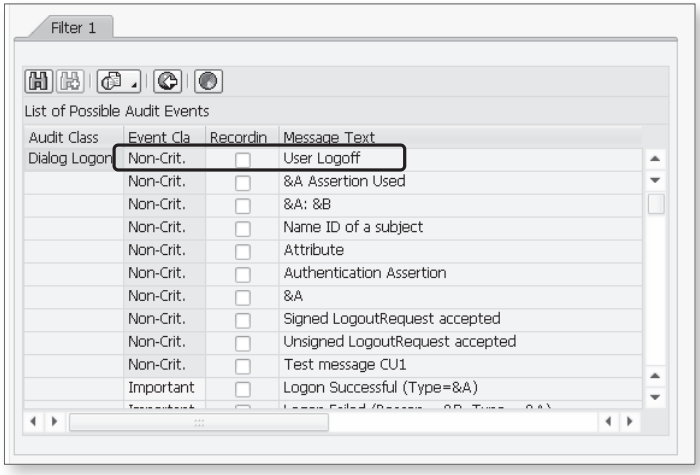
And Here's How ...

If you launch Transaction SM19, you can set up the static or dynamic Security Audit Logs. In the filter section, define your settings through a high-level view or a detail view. In Figure 1, you can see the high-level view and the Detail Configuration button.

If you click on the Detail Configuration button, you can see all events in the last column as shown in Figure 2. The audit classes appear in the first column, Audit Classes, and the event classifications appear in the second column, Event Cla. If you want to record an event, select it in the third column, Recordin.



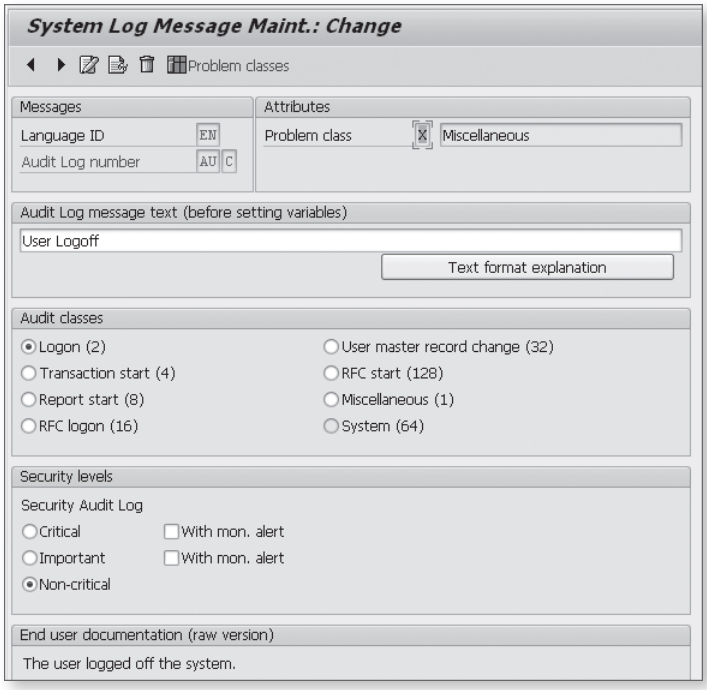
« *Figure 1 Transaction SM19: Security Audit Filter Detail Configuration*



« *Figure 2 Transaction SM19: Security Audit Events with Details*

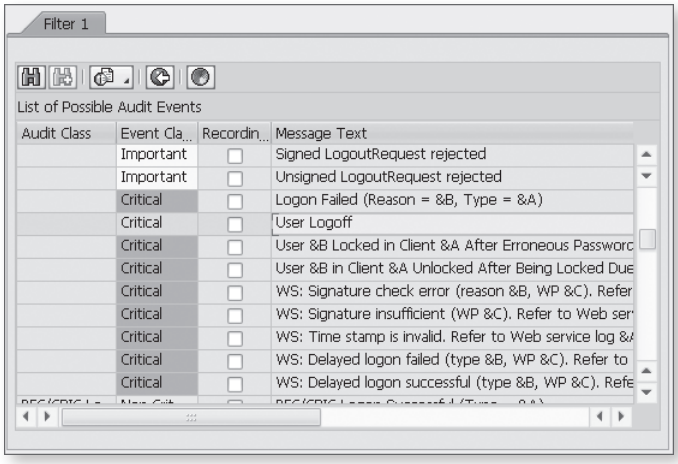
Transaction SE92 (New SysLog Msg Maintenance as of 46A) can be used to change SAP default values. After you access the transaction, specify a language, maintain the LANGUAGE ID field, and enter a system log number in the SYSTEM LOG NO. field. The user logoff message is defined as SYSTEM LOG NUMBER AU – C. Save your changes by pressing F8.

In the resulting screen shown in Figure 3, you can see all of the possible elements that can be changed. In this example, change the security level from NON-CRITICAL to CRITICAL.



« *Figure 3 System Message Maintenance Form*

After the message has been changed and saved (press Ctrl+S), go back to Transaction SM19 to verify the results of this change. Figure 4 shows that the User Logoff is classified as Critical instead of Non-critical after this setting.



« *Figure 4 User Logoff Classified as a Critical Event Class*

Tip 80

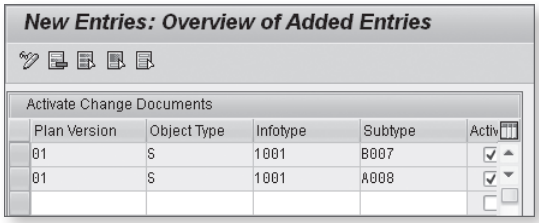
Configuring the SAP System to Log Activity in the Security Structure

You can trace all activity within your security organization structure by tracing authorization data assignments with change documents.

As we discussed in Part 1, Tip 10, you can use the HR structure to assign authorization to the users; however, by default the trace log on the organization structure isn't active. So if an error or an oversight occurs or you change or delete some data, you cannot easily restore the previous status. Let's look at a quick example: Suppose several security teams across all geographies of your company are using the organization structure. If one of these teams deletes or moves part of this structure due to an oversight, you can easily see what happened by consulting the log. By configuring a customizing table, you can enable logging on only the authorization data assignments.

And Here's How ...

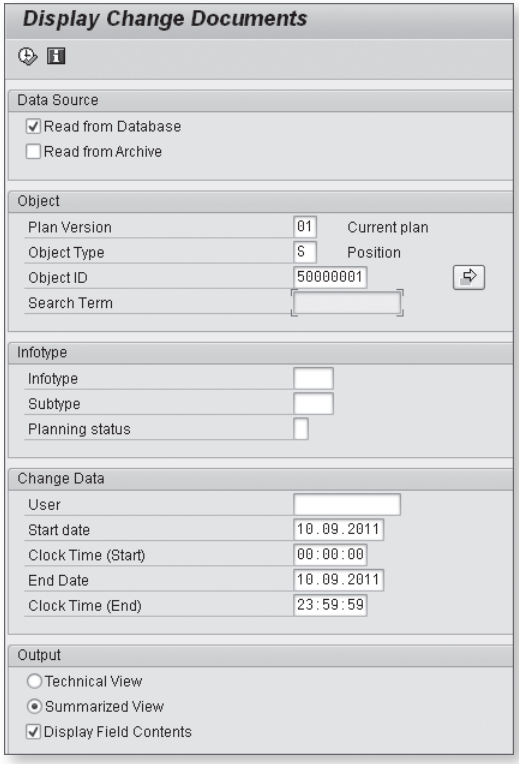
First, access Table T77CDOC_CUST to set up the data you want to trace. In this table, enter the Plan Version and Infotype for which you want to activate the logging by using the New Entry button. Figure 1 shows this table populated with two rows, which allow the trace to be run on the position level. All user and role assignments (add or remove) for a position are traced for their allocation relationship (1001 B007 for role and 1001 A008 for users) upon a position (S).



« *Figure 1 Table T77CDOC_CUST*

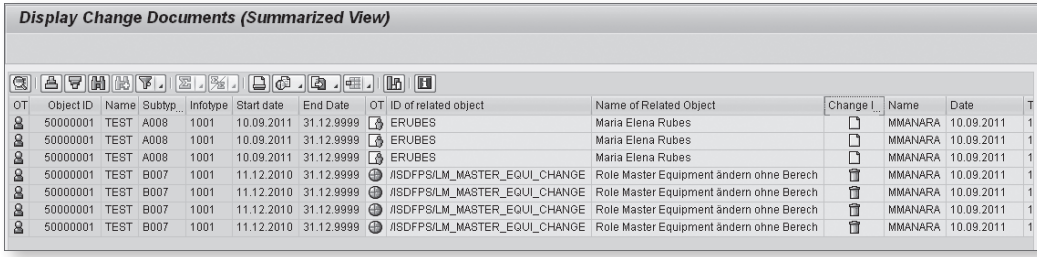
With this configuration, if you try to allocate a user to a position in the organization structure with Transaction PPOMW, for example, a change document record is written.

To see the change documents, use Report RHCDOC_DISPLAY (executable through Transaction SE38). Figure 2 shows how to execute the report. Enter "01" in the PLAN VERSION field, and fill in the OBJECT TYPE and the OBJECT ID fields. You can also restrict your selection criteria by infotype or user change data in their respective sections of this dialog box, for example, and then click on the EXECUTE button.



« *Figure 2 Report RHCDOC_ DISPLAY Selection Criteria to Display Organizational Structure Change Documents*

Figure 3 shows the result of a change document. For position 50000001, user ERUBES has been added and role /ISDFPS/LM_MASTER_EQUI_CHANGE has been removed. You can easily identify objects that have been added or removed by keeping an eye on the icon in the CHANGE column. A new blank page icon means an element has been added; a recycle bin icon means an element has been removed.



Display Change Documents (Summarized View)

OT	Object ID	Name	Subtyp...	Infotype	Start date	End Date	OT	ID of related object	Name of Related Object	Change I...	Name	Date	T
	50000001	TEST	A008	1001	10.09.2011	31.12.9999		ERUBES	Maria Elena Rubes		MMANARA	10.09.2011	1
	50000001	TEST	A008	1001	10.09.2011	31.12.9999		ERUBES	Maria Elena Rubes		MMANARA	10.09.2011	1
	50000001	TEST	A008	1001	10.09.2011	31.12.9999		ERUBES	Maria Elena Rubes		MMANARA	10.09.2011	1
	50000001	TEST	B007	1001	11.12.2010	31.12.9999		/ISDFPS/LM_MASTER_EQUI_CHANGE	Role Master Equipment ändern ohne Berech		MMANARA	10.09.2011	1
	50000001	TEST	B007	1001	11.12.2010	31.12.9999		/ISDFPS/LM_MASTER_EQUI_CHANGE	Role Master Equipment ändern ohne Berech		MMANARA	10.09.2011	1
	50000001	TEST	B007	1001	11.12.2010	31.12.9999		/ISDFPS/LM_MASTER_EQUI_CHANGE	Role Master Equipment ändern ohne Berech		MMANARA	10.09.2011	1
	50000001	TEST	B007	1001	11.12.2010	31.12.9999		/ISDFPS/LM_MASTER_EQUI_CHANGE	Role Master Equipment ändern ohne Berech		MMANARA	10.09.2011	1

Figure 3 Report RHCDOC_DISPLAY Output Result

Tip 81

Activating Table Tracing to Log the Details of Changes Made

You can activate table tracing (logging) to track who made certain changes and when the changes were made.

Suppose you want to create a relationship between a user and his manager to manage an approval step. To manage the mapping values, you can sometimes create a custom table. However, it's crucial that only specific users maintain the custom table to avoid any incorrect assignments. To make sure no unknown users are making the wrong changes, you can track them using the table log feature.[1]

And Here's How ...

To use table tracing (used interchangeably in this tip with table log), you must perform three steps, as discussed in the following subsections.

Activate the REC/CLIENT Parameter

The first step must be performed only once by the SAP system administrator. In Figure 1 (showing Transaction RZ10), you can see the relevant parameter: `rec/client`. Here you need to indicate the involved client; if you enter "ALL" in the PARAMETER VAL. field, the table trace is activated for all clients defined in the system.

1 Note that table tracing (also called change logs) can also be useful to monitor all customizing object changes in a production system.

Figure 1 rec/client System Parameter Setting

Activate Log Data Changes

After you've activated the `rec/client` parameter, select the LOG DATA CHANGES checkbox for the custom table. It's essential that this flag work on both custom and standard tables.

Let's go through an example where you need to activate the log parameter and then set a log flag in the table definition of a custom table. Custom Table ZCONVERT_USER has two fields: USER_1 and USER_2. In this table, there are records similar to those shown in Figure 2.

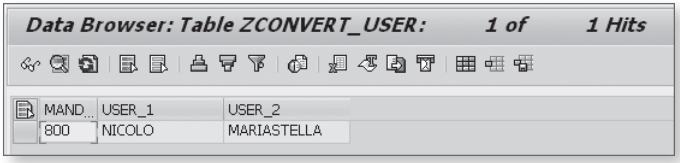


Figure 2 Transaction SE16: Display Content Data of Custom Table ZCONVERT _ USER

To activate the LOG DATA CHANGES flag, you must use Transaction SE13 (Maintain Technical Settings [Tables]) as shown in Figure 3. Enter the table name, press F8, and then flag the LOG DATA CHANGES checkbox.

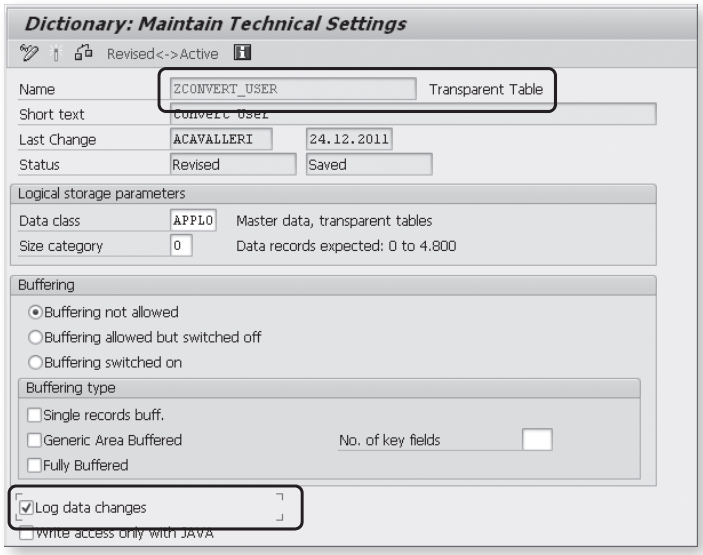


Figure 3 Transaction SE13: Log Data Changes Checkbox

Now you have to make a change to see the change log. In this example, you'll change the content of field USER_2 from MARIASTELLA to DARIA.

Analyze the Tables Changes

The last action is to look into the change logs (as shown in Figure 4) with Transaction SCU3 (Table History). If you execute Transaction SCU3 and enter the table name after you press F8 you can see all changes performed on this table during a period of time. Figure 4 shows that on 24.12.2011 at 17:10:54, user ACAVALLERI has changed a value from MARIASTELLA to DARIA on Table ZCONVERT_USER.

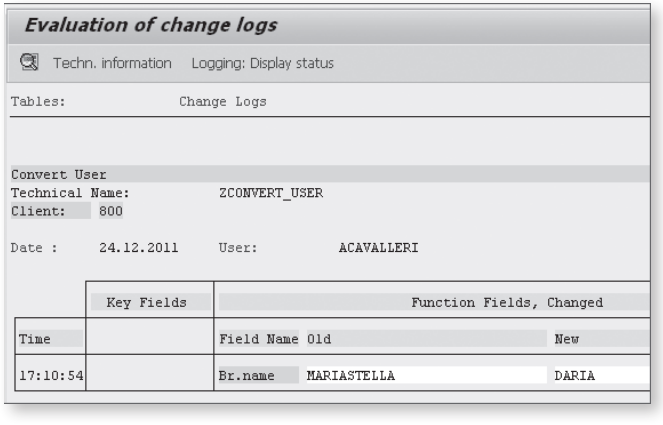


Figure 4 Transaction SCU3: Log Data Changes Output

Tip 82

Viewing All Instances of Profile Parameters

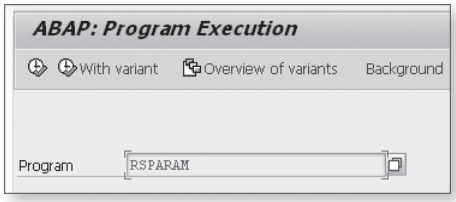
You can display all instance profile parameters to overcome an audit issue with a little-known report.

Instance profile parameters are responsible for many system settings. This is true in particular for security. For example, all password rules are managed though profile parameters.

If you're looking for a global overview of all (or a subset) of profile parameters, you probably usually use Transactions RZ10 and RZ11, which are very cumbersome because you can only display one parameter at a time. If you want to see all actual profile parameter settings of an SAP system, you can use standard ABAP Report RSPARAM.

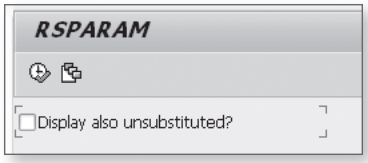
And Here's How ...

If you want to view more than one profile parameter at a time, SAP delivers standard ABAP Report RSPARAM. This report can be launched with Transaction SA38 (ABAP Execution) as shown in Figure 1.



« *Figure 1 Transaction SA38 with Program RSPARAM*

Launch Report RSPARAM by pressing F8, and the screen shown in Figure 2 appears.

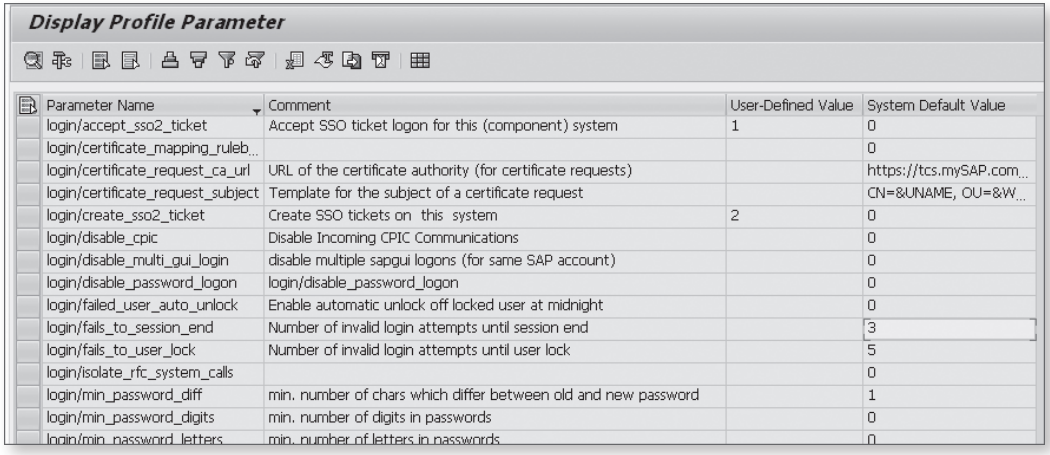


« *Figure 2 Report RSPARAM: Display Also Unsubstituted Flag*

Normally, the best output is with the DISPLAY ALSO UNSUBSTITUTED? flag inactive. An "unsubstituted value" means a value "before the substitute variables were replaced." For example, the SAP standard value for parameter `DIR_ATRA` is `$(DIR_DATA)`. During the installation phase, this value will be substituted with the real directory of the application server (e.g., *D:\usr\sap\DEV\D00\data*), and this will become the system default value.

In the output layout, you can apply filters (by using the funnel button). In Figure 3, you can see all profile parameters for the "login" suffix.

When an SAP system administrator changes a default value, it will be shown in Report RSPARAM in the USER-DEFINED VALUE column. This is the case for parameter LOGIN/ACCEPT_SSO2_TICKET in the first rows of Figure 3.



Parameter Name	Comment	User-Defined Value	System Default Value
login/accept_sso2_ticket	Accept SSO ticket logon for this (component) system	1	0
login/certificate_mapping_ruleb...			0
login/certificate_request_ca_url	URL of the certificate authority (for certificate requests)		https://tcs.mySAP.com...
login/certificate_request_subject	Template for the subject of a certificate request		CN=&UNAME, OU=&W...
login/create_sso2_ticket	Create SSO tickets on this system	2	0
login/disable_cpic	Disable Incoming CPIC Communications		0
login/disable_multi_gui_login	disable multiple sapgui logons (for same SAP account)		0
login/disable_password_logon	login/disable_password_logon		0
login/failed_user_auto_unlock	Enable automatic unlock off locked user at midnight		0
login/fails_to_session_end	Number of invalid login attempts until session end		3
login/fails_to_user_lock	Number of invalid login attempts until user lock		5
login/isolate_rfc_system_calls			0
login/min_password_diff	min. number of chars which differ between old and new password		1
login/min_password_digits	min. number of digits in passwords		0
login/min_password_letters	min. number of letters in passwords		0

» *Figure 3 Login Profile Parameters*

As an alternative to ABAP Report RSPARAM you can also use some standard transactions to display the parameter values filtered at first by family area, as shown in Table 1.

Transaction Code	Transaction Description
RSPFPAR	Display Profile Parameter
RSPFPAR_AUTH	Authorization All
RSPFPAR_CALLSYSTEM	Call System
RSPFPAR_GATEWAY	SAP Gateway
RSPFPAR_LOGIN	Logon Rules
RSPFPAR_PROFGEN	Profile Generator
RSPFPAR_RFC	Remote Function Call
RSPFPAR_SAPSTAR	Hardcoded SAP*
RSPFPAR_SNC	SNC
RSPFPAR_SPOOL	Spool Parameters
RSPFPAR_STATISTICS	Workload Statistics
RSPFPAR_SYSLOG	Syslog Parameters
RSPFPAR_TABLEREC	Table Recording
RSPFPAR_TABLESTAT	Table Access Statistics

Table 1 Transactions to Display Profile Parameters

These transactions are useful when an auditor asks to see the settings of these profile parameters. Indeed, you can enable some of these transactions instead of releasing Transaction SA38 into the external auditor's role.

Tip 83

Identifying Alias Transactions to Eliminate Unauthorized System Access

You can identify alias transactions to avoid a pitfall during an audit.

SAP authorization transactions are mainly divided into two groups: user management and authorizations management. From a Segregation of Duties (SoD) point of view, you must guarantee that a user manager isn't able to maintain authorizations, and that the authorization manager isn't able to maintain users. If you as a security administrator browse the SAP standard menu, you can find the main transactions involved in security. However, these transactions are not the only ones available. In this tip, you'll discover that there are many other transactions with different names (known as alias transactions) that can manage users and roles in the same way. It's essential to know how to identify these transactions and determine whether your role contains these improperly assigned transactions to avoid a pitfall during an audit. In this tip, you'll learn how to find alias transactions that relate to the classical Transactions SU01 (User Maintenance) and PFGG (User Maintenance) and how to keep unauthorized users from utilizing them.

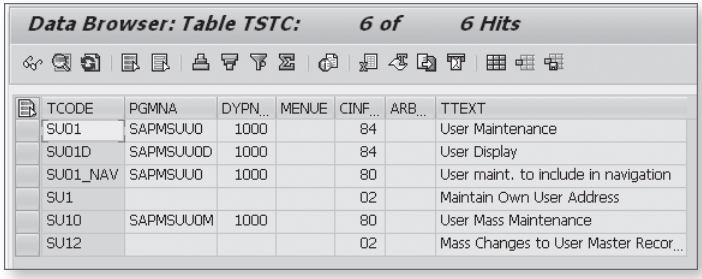
And Here's How ...

Working with User Management Transactions

The classical transactions for user management are the following:

- Transaction SU01 (User Maintenance)
- Transaction SU10 (User Mass Maintenance)

If you look (through Transaction SE16) at Table TSTC (SAP Transaction Codes) with "SU01*" and "SU1*" filters in the TCODE field, you'll see the output as shown in Figure 1. All transactions defined in the SAP system that start with SU01 and SU1 are the classical transactions used to maintain user master records.



Data Browser: Table TSTC: 6 of 6 Hits

TCODE	PGMNA	DYPN...	MENUE	CINF...	ARB...	TTEXT
SU01	SAPMSUU0	1000		84		User Maintenance
SU01D	SAPMSUU0D	1000		84		User Display
SU01_NAV	SAPMSUU0	1000		80		User maint. to include in navigation
SU1				02		Maintain Own User Address
SU10	SAPMSUU0M	1000		80		User Mass Maintenance
SU12				02		Mass Changes to User Master Recor...

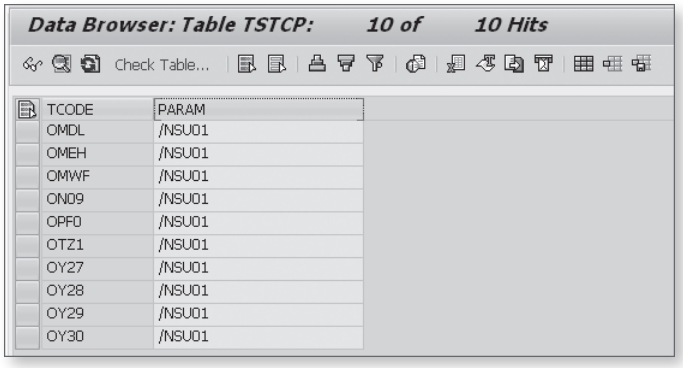
« *Figure 1 Transaction SE16: Main User Management Transactions in Table TSTC*

Transaction SU01_NAV (User Maintenance to Include in Navigation) is also available, which is very similar to Transaction SU01 because it can create and change user attributes. Instead, Transaction SU01D (User Display) is available in situations in which you want to provide only the display functionality of a user master record.

For mass maintenance, you can use Transaction SU12 (Mass Changes to User Master Records) because it has the same functionality as Transaction SU10.

Transaction SU1 (Maintain Own User Address) is similar to Transaction SU3 (Maintain Users Own Data) but only in the ADDRESS tab. Of this family, you can also consider Transactions SU50 (Own Data), SU51 (Maintain Own User Address, which is the same as Transaction SU1), and SU52 (Maintain Own User Parameters).

Sometimes, the same functionality (in this example, user master record maintenance) is available through a customizing transaction. These customizing transaction are parameter transactions. If you use Transaction SE16 to look into Table TSTCP (Parameters for Transactions) with the "*SU01*" filter in the PARAM field, you can see the output displayed in Figure 2.



Data Browser: Table TSTCP: 10 of 10 Hits

Check Table...

TCODE	PARAM
OMDL	/NSU01
OMEH	/NSU01
OMWF	/NSU01
ON09	/NSU01
OPF0	/NSU01
OTZ1	/NSU01
OY27	/NSU01
OY28	/NSU01
OY29	/NSU01
OY30	/NSU01

« *Figure 2 Transaction SE16: Table TSTCP Output for a Search of Transaction SU01*

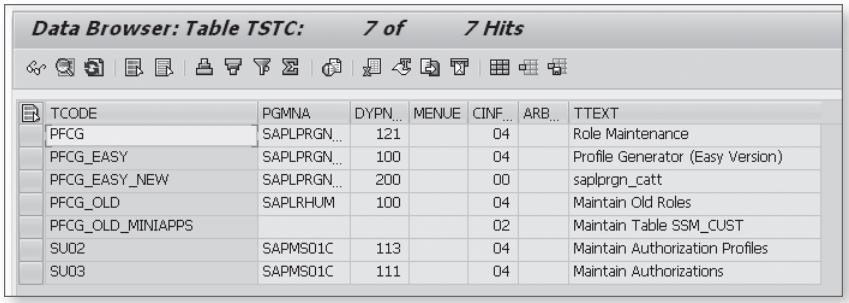
As you can see in Figure 2, many other transactions are equivalent to Transaction SU01 (User Maintenance).

Authorization Management Transaction

From the authorization side, the classical transactions for authorization management are the following:

- Transaction SU02 (Maintain Authorization Profiles)
- Transaction SU03 (Maintain Authorizations)
- Transaction PFCG (Role Maintenance)

If you look into Table TSTC, you'll discover the result shown in Figure 3.



Data Browser: Table TSTC: 7 of 7 Hits

TCODE	PGMNA	DYPN...	MENUE	CINF...	ARB...	TTEXT
PFCG	SAPLPRGN...	121		04		Role Maintenance
PFCG_EASY	SAPLPRGN...	100		04		Profile Generator (Easy Version)
PFCG_EASY_NEW	SAPLPRGN...	200		00		saplprgn_catt
PFCG_OLD	SAPLRHUM	100		04		Maintain Old Roles
PFCG_OLD_MINIAPPS				02		Maintain Table SSM_CUST
SU02	SAPMS01C	113		04		Maintain Authorization Profiles
SU03	SAPMS01C	111		04		Maintain Authorizations

» *Figure 3 Transaction SE16: Main Authorization Management Transactions in Table TSTC*

Except Transaction PFCG (Role Maintenance), most of the transactions with suffix PFCG have not been released by SAP. However, be sure to check again in future releases.

When you look in Table TSTCP (Parameters for Transactions), you can see the content shown in Figure 4. If you start these transactions, you'll discover that they are related to Transactions SUIM (User Information System) and PFUD (User Master Data Reconciliation). But, if you just consider the name of the last transaction (Transaction ZDUMMY), you'll probably ignore it. As you can see, this is exactly like Transaction PFCG (Role Maintenance). Now you can check to see if there are roles that have this transaction improperly assigned.

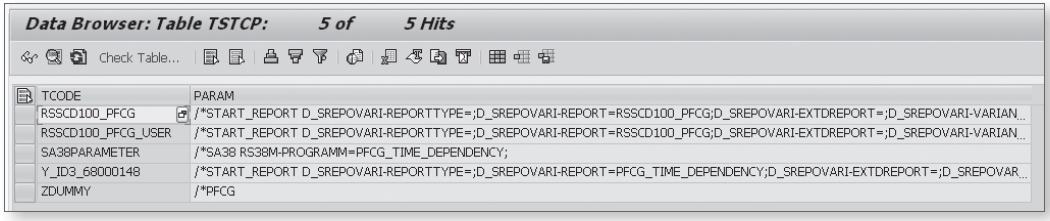


Figure 4 Transaction SE16: Table TSTCP Output for a Search of Transactions PFCG, SU02, and SU03

It's important to remember that a user authorized only to use transaction codes (with the `S_TCODE` authorization object) will not be able to create or maintain users and roles because he is missing many other mandatory authorizations.

Tip 84

Finding a Specific User Who Has Made Changes to Values

You can verify (and certify) that only selected users are authorized to maintain specific data by checking different repository tables.

Let's say you have users in the system who aren't allowed to perform financial accounting activities. However, when you perform a change document review, you see that these users have found a way to make changes to some of the financial documents. This tip shows you how to use two different tables to easily find changed documents and identify the users who have performed the changes.

And Here's How ...

SAP delivers many transactions that help you determine which user has changed a particular document (and when). Many of these transactions have "04" at the end of their names; for example, Transaction MM04 (Display Material Change Documents). In the backstage, there are two important tables that contain the history of changed (and created) documents. The tables we focus on in this tip are Table CDHDR (Change Document Header) and Table CDPOS (Change Document Items). The table's naming is quite clear: CD stands for "change document," HDR stands for "header," and POS stands for "position." Suppose that a user has changed the Material Group field value from "002" to "00101" with Transaction MM02 (Change Material) as shown in Figure 1.

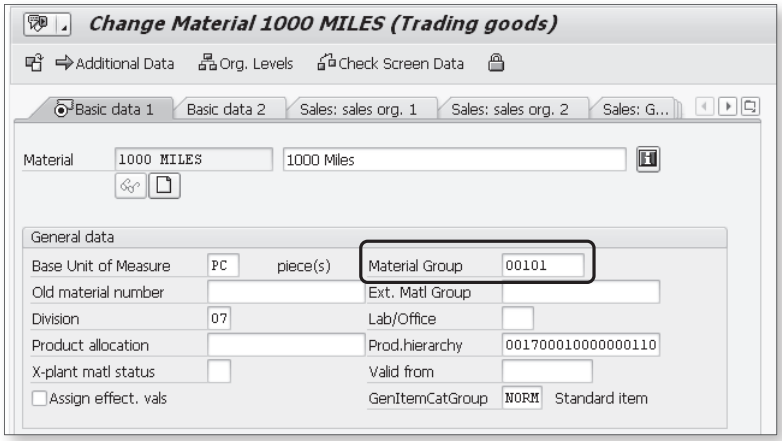


Figure 1 Transaction MM02: Change Material Group

As a security manager, you need to find who has changed the value in Transaction MM02, and you can do this with change documents. Start Transaction MM04 (Display Material Change Documents) to see a form like the one in Figure 2. Here you need to specify a material code. In this case, it will be difficult to find all materials changed by a specific user you want to monitor (e.g., an IT department should not modify business data).

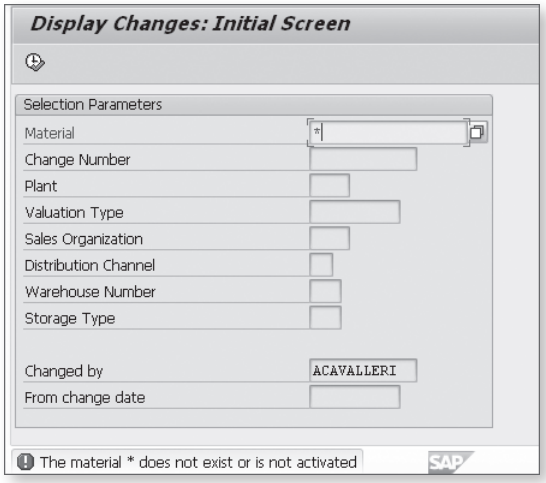


Figure 2 Transaction MM04 Parameters

When you execute the search, if you specify the material code, you will see the output shown in Figure 3.

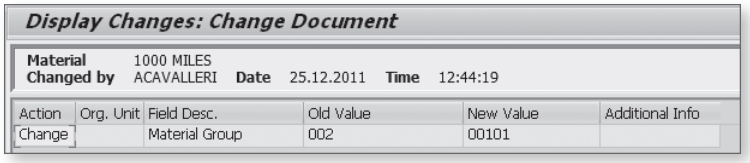


Figure 3 Transaction MM04 Output: Change Document Data

Suppose now that you really want to analyze all material master data changes performed by user ACAVALLERI since the previous year. In this case, it could be very helpful to use a set of transactions available through the Audit Information System (AIS). If you look at the SAP_AUDITOR_SA_BC_CUS_TOL role with Transaction PFCG (Role Maintenance), you'll discover the transactions highlighted in Figure 4.

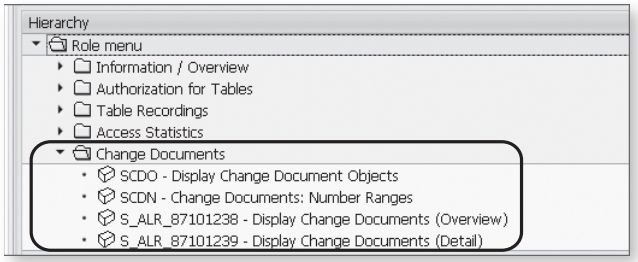


Figure 4 Change Documents Transactions in Role SAP_AUDITOR_SA_BC_CUS_TOL

Start Transaction S_ALR_87101238 (Display Change Documents Overview), and you can now specify all materials (enter the "*" value in the Object ID field) for the user ID you want to monitor in the period of interest.

The final result is a list (with more details), which will contain all activities performed by user ACAVALLERI as shown in Figure 5.

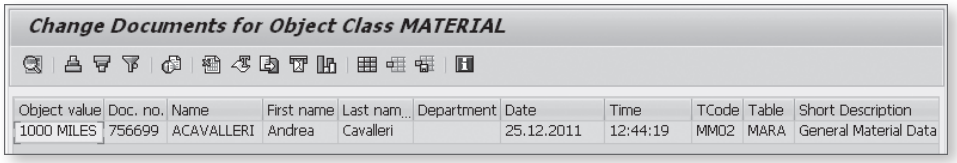


Figure 5 Transaction S_ALR_87101238 Output

Tip 85

Identifying Query Changes

You can perform an inquiry to determine who changed a query and when that change occurred.

In some cases it's useful to determine which user changed a program or a query. However, it isn't always simple or easy to look at the transaction interface and find the data you're looking for if you have to check several programs or queries. You can easily bypass this problem by accessing an SAP table to discover this information. This may be useful to verify that the proper people are allowed to change and maintain the query.

And Here's How ...

To discover which user changed a query, you first need to access Transaction SQ00 to find the query technical name. Figure 1 shows the following menu path after a query is highlighted to show the technical query name:

Query • More Functions • Display Report Name

After you've retrieved the query name and the query has been executed at least once, you can browse Table D010SINF to find the query creator and the author of the last change to this query. This view is based on Table REPOSRC. This table contains all SAP programs and the ABAP source code. This table is classified as critical; if a user is allowed to change some data in this table, the user can potentially change the SAP standard or custom source code.

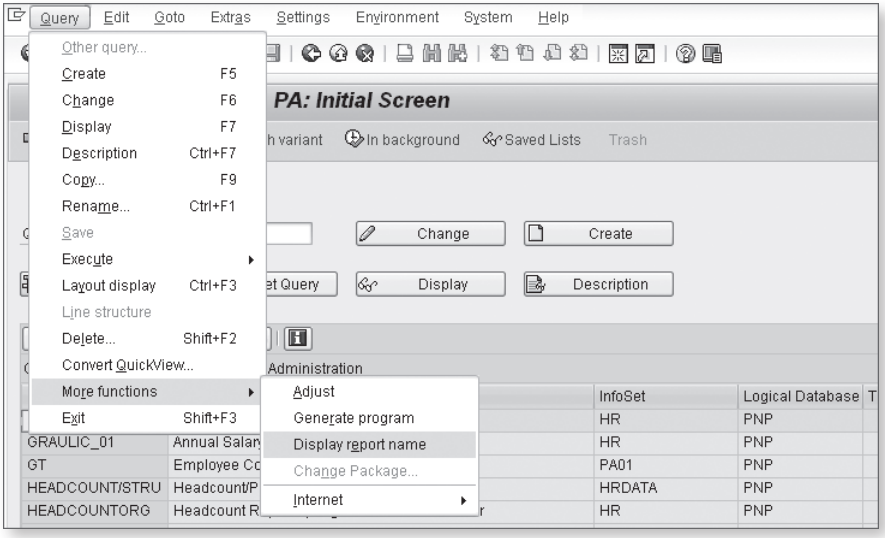


Figure 1 Transaction SQ00

However, if you put your query technical name in the PROG field, you can find the creator of the query through the CNAM field. You can also find who last changed this query through the UNAM field as shown in Figure 2.

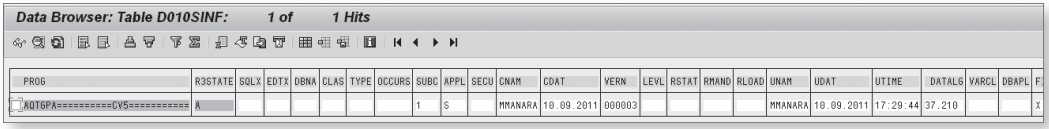


Figure 2 CNAM and UNAM Fields Used to Find the Creator and Last Change User of a Program

After you've retrieved the information, you can investigate whether the change is due to an authorization hole or other reasons.

Tip 86

Protecting and Auditing Your Remote Function Call

You can monitor users who have remote function call (RFC) authorization to make sure unauthorized users don't have remote access to the SAP system.

User types are divided into two main groups: dialog users and nondialog users. While security managers mainly define authorizations for dialog users, they don't maintain RFC authorizations. This means that for nondialog users, an SAP_ALL profile is often granted without a manager considering the risks. For example, if a user knows the nondialog user ID and password, that person could connect with an RFC connection and perform all activities by calling many thousands of functions. In other words, a user who shouldn't have authorization could remotely create purchase order documents or modify HR data. This tip shows you how to protect RFC connections and check that users have the proper RFC authorizations.

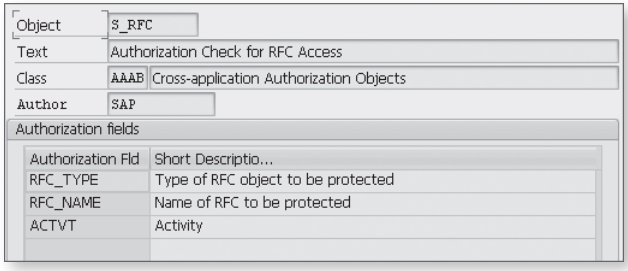
And Here's How ...

There are two steps you need to follow to maintain secure RFC connections.

Check RFC Authorizations

Almost all companies have defined a Basis role containing all noncritical transactions and authorizations to be given to all dialog users (see Part 3, Tip 38). Often a full RFC authorization is granted through authorization object `S_RFC` in the Basis role. This means that all users have full authorization on RFCs.

As shown in Figure 1, the S_RFC authorization object is composed of three fields, which you can see in the authorization tree of Transaction PFCG when the S_RFC authorization object is present.



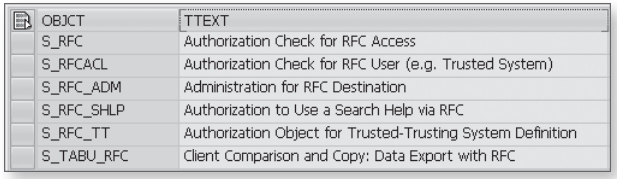
« *Figure 1 S_RFC Authorization Object Fields*

It's very important that you don't underestimate this object because it should be considered as important as the S_TCODE authorization object. Let's quickly go over each of the authorization fields to understand the meaning and the possible values of each one:

- The ACTVT field has a sort of Boolean value because it can only equal 16 (if a user has this authorization, he can execute an RFC function).
- The RFC_TYPE field is responsible for granting a function group or a function module. You must consider that in a function group, you may find many function modules.
- The RFC_NAME field is the most important because it permits you to define a very limited authorization. In fact, you could authorize a user just to use a unique function module.

You should limit the authorizations on the S_RFC object as much as possible. You must also consider that this object is one of the most relevant for external auditors when using the auditor critical authorization object checklist.

You should know also that the S_RFC object is not the unique RFC authorization. When you look in Table TOBJ (Authorization Objects) through Transaction SE16 with a "S_RFC*" filter in the OBJECT field, you can see that there are many others authorization objects related to RFC connections (see Figure 2). You should always manually grant these authorization objects.



OBJCT	TTEXT
S_RFC	Authorization Check for RFC Access
S_RFCACL	Authorization Check for RFC User (e.g. Trusted System)
S_RFC_ADM	Administration for RFC Destination
S_RFC_SHLP	Authorization to Use a Search Help via RFC
S_RFC_TT	Authorization Object for Trusted-Trusting System Definition
S_TABU_RFC	Client Comparison and Copy: Data Export with RFC

« *Figure 2 Transaction SE16: Table TOBJ on the S_RFC* Object*

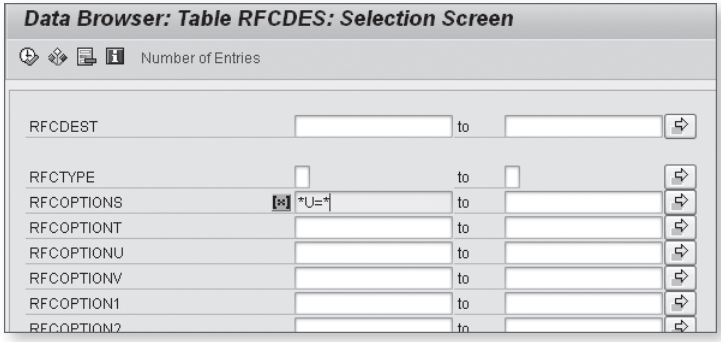
A more dangerous risk is represented by the system parameter `auth/rfc_authority_check`. In fact, if you change the default SAP value (from 1 to 0), the authorization check will no longer be performed (see OSS Note 931252).

To audit the roles yourself, you can quickly analyze the content of Table AGR_1251 by accessing Transaction SE16 and entering the value "S_RFC" in the OBJECT field.

Check Your RFC Destination

When working with RFC connections, you also have to pay attention to the RFC destinations that are defined in the system. You can verify that the RFC destination defined should not contain the user ID credential (user ID and password), which allows the user to use the RFC destination to log on to another system.

To verify whether the RFC destination contains a specific user ID credential, access Transaction SE16, browse Table RFCDES, and find the destination with the U= string in the RFCOPTIONS field (see Figure 3).



» *Figure 3 Table RFCDES to Identify the RFC Destination with the User ID Credential*

Alternatively, you can use the standard Report RSRFCCHK.

Part 7

Security Templates

Things You'll Learn in this Section

Due to the nature of information and structures, SAP authorization can be complicated to gather and analyze for security administrators and business users alike. Therefore, it's essential to use the correct tool and authorization template during an authorization project to avoid lost time. This part of the book will present information to help the security administrator and key users to collect, analyze, and represent the authorization of an SAP system more easily by using common and available tools. We present practical advice and instructions where necessary to help you learn how to use standard templates to collect authorization data during a project and streamline your processes. This part may also be useful when your company uses tools for business process modeling.

Tip 87

Using a Spreadsheet to Collect Authorization Data

You can use several kinds of office software tools to process and document your authorization data.

There are many office software tools that you can use during the analysis, development, and documentation phases of your authorization projects. However, you should know that not all of these tools are useful for processing authorization data. It's also possible that you may use the right tool, but one that doesn't help you save time and money. This tip explores how you can manage authorization data through a spreadsheet, while at the same time avoiding errors during data processing and exploiting all features to save time and reach your goal.

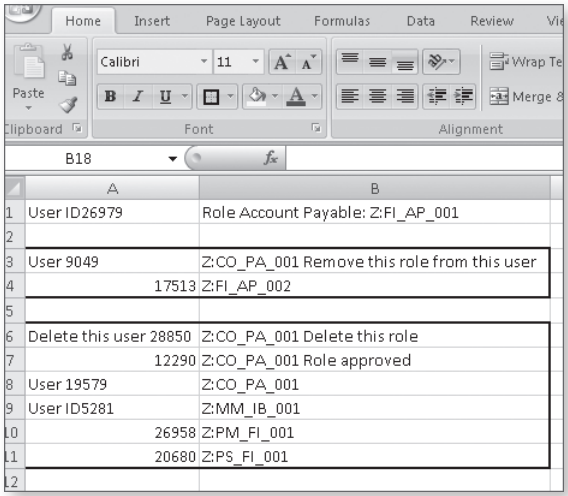
And Here's How ...

The most commonly used tool for documenting and processing authorization data is a spreadsheet. Another tool, most used in conjunction with or in addition to the spreadsheet, is a relational database management system (RDBMS), such as Microsoft Office or Open Office. These tools give you the ability to manage your data in a database form and then create several kinds of formulas and charts for extracting and representing your authorization data.

Although the tool used for managing and documenting the authorization is a spreadsheet, unfortunately, misuse of the tool often results in lost time and productivity. Let's walk through the most common errors and how to avoid them.

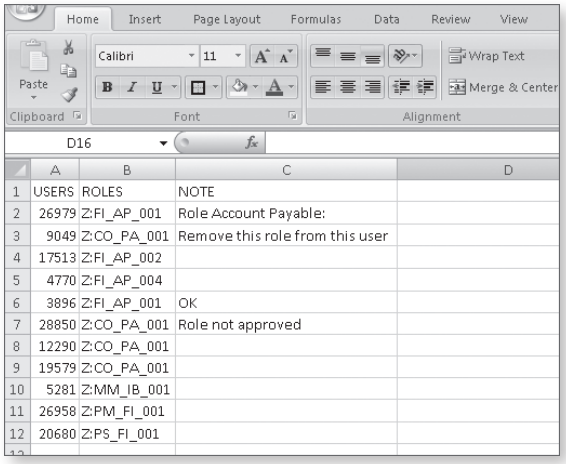
Using a Spreadsheet Like a Word Processor

Here, you insert several types of data in a spreadsheet, and you simply insert the spreadsheet in the document without structuring it. An example is shown in Figure 1: Cells A3 and B3 contain several different pieces of data in the same cell. A3 contains the user ID and the label user; in the same way, cell B3 contains the technical role name and a note about it. In this way, you've spent more time extracting the correct data from the cell.



« *Figure 1 Unstructured Data in a Spreadsheet*

A more efficient way of organizing your data is shown in Figure 2, where a header defines the data in the column, and each cell contains a unique value.



« *Figure 2 Structured Data Spreadsheet*

Condensing Information

Avoid creating and defining many sheets (tabs) in your spreadsheet. Typically, use no more than five sheets so you don't lose control of your spreadsheet file.

Using Several Colors and Fonts

You can use color to comment or highlight data in a spreadsheet; however, you can only create filters that are based on colors in certain releases of spreadsheet processors. Use multiple fonts and colors sparingly in your spreadsheets.

Avoid Using Several Types of Formatting: Wrap Text and Merge & Center Functionality

Wrap Text and Merge & Center commands are useful for reading purposes but tend to disrupt macros that try to extract the correct information. Avoid using this type of formatting when you have to manually/automatically process the data.

Processing and Inserting Data by Using Several Empty Rows

If you process and insert a spreadsheet with several empty rows, you won't be able to filter these data. Also, if you have to define a formula near these data, you'll have several errors or empty results in the corresponding empty row. Table 1 shows an example of this. If you have two columns in a spreadsheet, and you want to insert a formula (e.g., sum of the first value and the second) in the third column, this formula will not work in the corresponding empty rows.

First Column	Second Column	Sum of First and Second Column
1	1	2
		Error or zero
2	2	4
		Error or zero
5	3	8

Table 1 Example of Spreadsheet Formatting with Empty Rows

Tip 88

Defining a Template for Gathering and Defining Your Job Role Data

You can easily gather data to create the job role analysis file.

Collecting data for defining and analyzing a job role requirement is an essential part of the authorization process phase. After that's done, you need to find a way to avoid doing rework and data processing.

This tip gives you an efficient way to structure your data into a spreadsheet. Unfortunately, it isn't possible to cover all project cases using process design tools such as ARIS, SAP Solution Manager, and so on. However, this tip will help you avoid some common errors and use consolidated best practices.

And Here's How ...

To start, follow the rules to manage a spreadsheet as we discussed in the previous tip. After that's done, make sure you do the following in terms of the data and headers:

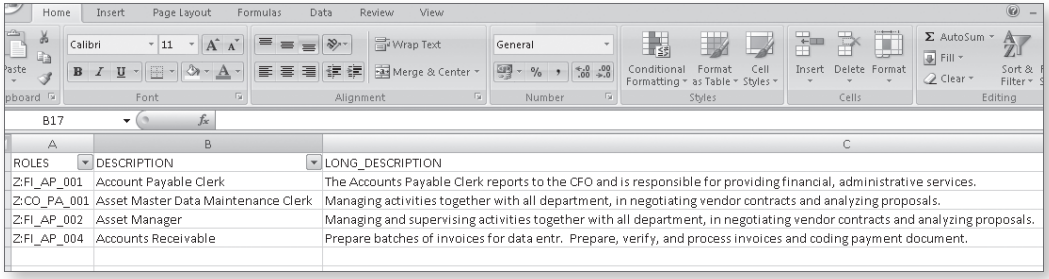
- List all job roles you want to define.
- Define the technical names of the job roles.
- Define a long and business-understandable description.
- Decide which SAP transactions should be contained in a job role with the help of the HR department or key users. Then insert this information in the template.

Keep in mind that gathering the authorization information is a step-by-step process. From a security administrator point of view, the path and steps are the following:

1. Retrieve the job role master list.
2. Retrieve the link job role master list and transaction codes.
3. Decide how to segregate at the organizational-level constraints.
4. Decide how to segregate at the nonorganizational-level constraints via business and internal policy decisions.

Merging this data in the same steps and same spreadsheet can cause difficulty in understanding and reading the template. Additionally, you can't perform an organizational constraints analysis if you haven't retrieved a good percentage of transaction codes in the job role master list.

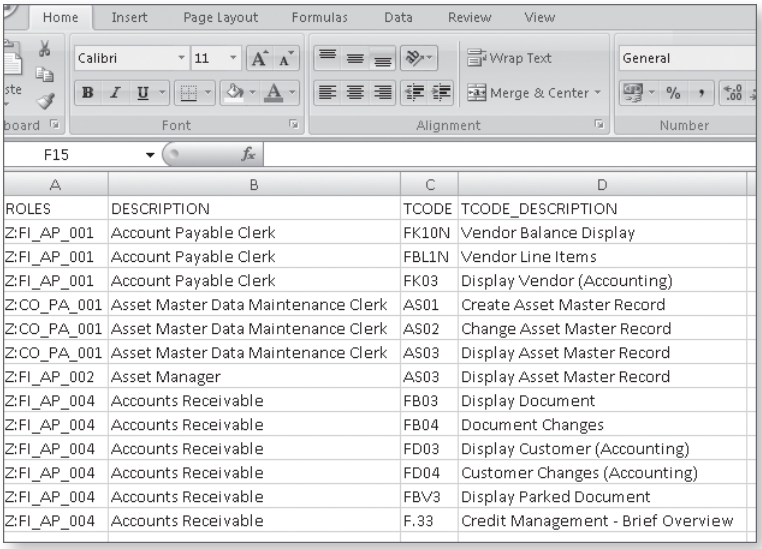
Figure 1 shows a master template example of a job role master list. The sheet is formed by three columns: technical role name, role name short job description, and job role long description. This template represents the minimum set of information to collect. You can also add other columns such as role approver and role owner. You can add these columns together with the business and HR department.



ROLES	DESCRIPTION	LONG_DESCRIPTION
Z:FI_AP_001	Account Payable Clerk	The Accounts Payable Clerk reports to the CFO and is responsible for providing financial, administrative services.
Z:CO_PA_001	Asset Master Data Maintenance Clerk	Managing activities together with all department, in negotiating vendor contracts and analyzing proposals.
Z:FI_AP_002	Asset Manager	Managing and supervising activities together with all department, in negotiating vendor contracts and analyzing proposals.
Z:FI_AP_004	Accounts Receivable	Prepare batches of invoices for data entr. Prepare, verify, and process invoices and coding payment document.

Figure 1 Job Role Master List Example of Template

The second sheet of this template shows a technical detail job role transaction code. For each job role and short description, you have to populate or receive a spreadsheet as shown in Figure 2.



A	B	C	D
ROLES	DESCRIPTION	TCODE	TCODE_DESCRIPTION
Z:FI_AP_001	Account Payable Clerk	FK10N	Vendor Balance Display
Z:FI_AP_001	Account Payable Clerk	FBL1N	Vendor Line Items
Z:FI_AP_001	Account Payable Clerk	FK03	Display Vendor (Accounting)
Z:CO_PA_001	Asset Master Data Maintenance Clerk	AS01	Create Asset Master Record
Z:CO_PA_001	Asset Master Data Maintenance Clerk	AS02	Change Asset Master Record
Z:CO_PA_001	Asset Master Data Maintenance Clerk	AS03	Display Asset Master Record
Z:FI_AP_002	Asset Manager	AS03	Display Asset Master Record
Z:FI_AP_004	Accounts Receivable	FB03	Display Document
Z:FI_AP_004	Accounts Receivable	FB04	Document Changes
Z:FI_AP_004	Accounts Receivable	FD03	Display Customer (Accounting)
Z:FI_AP_004	Accounts Receivable	FD04	Customer Changes (Accounting)
Z:FI_AP_004	Accounts Receivable	FBV3	Display Parked Document
Z:FI_AP_004	Accounts Receivable	F.33	Credit Management - Brief Overview

Figure 2 Job Role Master List with Transaction Data Detail

This method of collecting your job role data helps you save time and perform several consistency checks during the overall analysis phase, consequently helping you avoid more time in the other phases to correct a previous analysis gap. Processing this file is quick; you won't need to rebuild and rework it when you receive this kind of file.

Tip 89

Defining a Template for Gathering the Organizational Constraints of Job Role Data

You can set up and gather your organizational-level constraints during the analysis phase by using a spreadsheet, which will save you time in the development phase.

Organizational-level constraints are commonly used in a corporation with several plants, companies, purchasing organizations, and so on. Generally, a corporation's main request is that company 1 users shouldn't view and manage data of company 2. To give an accurate analysis during job role deployment in the analysis phase, it's essential to first discover all sensible organizational-level constraints and all values for each job role defined.

This step, while conceptually simple, is often an analysis pitfall during an authorization or roll-out projects. This tip shows you how to avoid some commonly occurring problems.

And Here's How ...

In the SAP ERP system, there are 34 standard organizational constraints (Table USORG_DB). Some of the most common constraints are used to restrict users from viewing company codes, plants, purchasing organizations, and commercial organizations. To retrieve this data, you need to join a job role and transaction code link with Table USOBT_C. This table contains the link between transaction codes and authorization objects. If an authorization object contains an organizational level that has been filed, you can determine whether a transaction code (and then the job role where this transaction is located) intercepts a certain organizational field.

The quickest way to relate these two tables is by using a tool such as Microsoft Access. Let's look at the concept behind this correlation data.

Figure 1 shows the query in Table USOBT_C. In the NAME field, enter any transaction that's involved; in the LOW field, enter your sensible organizational fields (e.g., in some companies, the commercial organization can be an important organizational level; in others that aren't commercial-oriented, this level may not be critical).

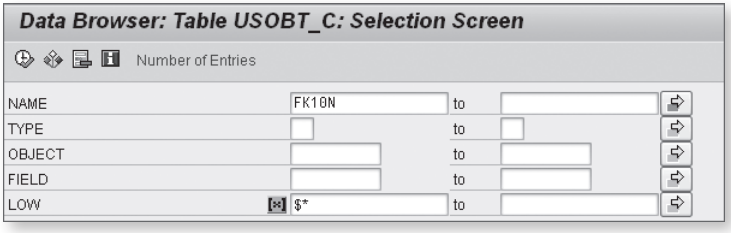
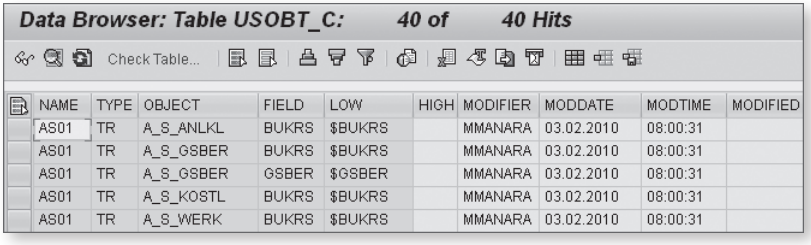


Figure 1 All Authorization Objects with at Least an Organizational Level Field of Transaction FK10N

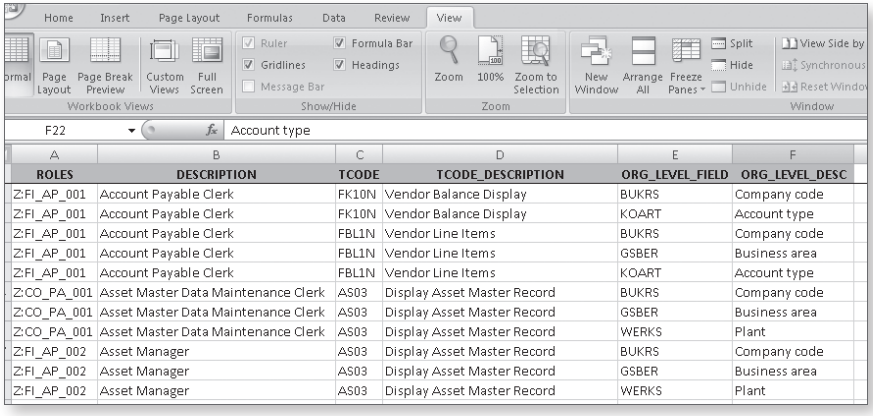
Execute the query in your management system database to see the screen in Figure 2, showing which transaction intercepts an authorization object with an organizational level field. You can tell that Transaction AS01 intercepts the company code field level BUKRS, the plant and the business area GSBER, and so on with all other transactions.



NAME	TYPE	OBJECT	FIELD	LOW	HIGH	MODIFIER	MODDATE	MODTIME	MODIFIED
AS01	TR	A_S_ANLKL	BUKRS	$BUKRS		MMANARA	03.02.2010	08:00:31	
AS01	TR	A_S_GSBER	BUKRS	$BUKRS		MMANARA	03.02.2010	08:00:31	
AS01	TR	A_S_GSBER	GSBER	$GSBER		MMANARA	03.02.2010	08:00:31	
AS01	TR	A_S_KOSTL	BUKRS	$BUKRS		MMANARA	03.02.2010	08:00:31	
AS01	TR	A_S_WERK	BUKRS	$BUKRS		MMANARA	03.02.2010	08:00:31	

Figure 2 Result of the Query to Find the Organizational Level for the Transaction Codes Involved

By relating these data retrieved with the job role and transaction code, you can see which organizational fields these job roles need to be constrained. Figure 3 shows you this detail, including the job role with all transactions involved and all organizational levels intercepted.



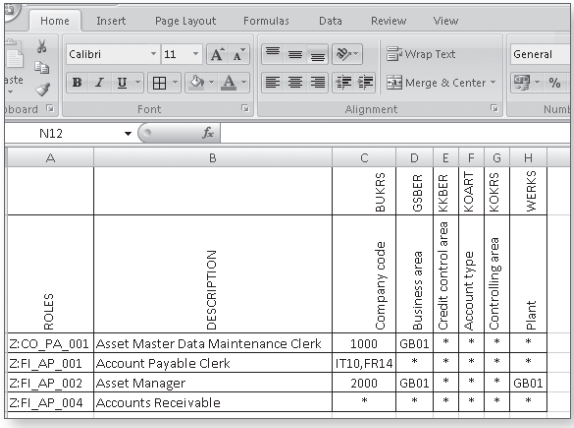
F22 | Account type

ROLES	DESCRIPTION	TCODE	TCODE_DESCRIPTION	ORG_LEVEL_FIELD	ORG_LEVEL_DESC
Z:FI_AP_001	Account Payable Clerk	FK10N	Vendor Balance Display	BUKRS	Company code
Z:FI_AP_001	Account Payable Clerk	FK10N	Vendor Balance Display	KOART	Account type
Z:FI_AP_001	Account Payable Clerk	FBL1N	Vendor Line Items	BUKRS	Company code
Z:FI_AP_001	Account Payable Clerk	FBL1N	Vendor Line Items	GSBER	Business area
Z:FI_AP_001	Account Payable Clerk	FBL1N	Vendor Line Items	KOART	Account type
Z:CO_PA_001	Asset Master Data Maintenance Clerk	AS03	Display Asset Master Record	BUKRS	Company code
Z:CO_PA_001	Asset Master Data Maintenance Clerk	AS03	Display Asset Master Record	GSBER	Business area
Z:CO_PA_001	Asset Master Data Maintenance Clerk	AS03	Display Asset Master Record	WERKS	Plant
Z:FI_AP_002	Asset Manager	AS03	Display Asset Master Record	BUKRS	Company code
Z:FI_AP_002	Asset Manager	AS03	Display Asset Master Record	GSBER	Business area
Z:FI_AP_002	Asset Manager	AS03	Display Asset Master Record	WERKS	Plant

Figure 3 Job Role, Transaction Code, and Detail of Which Organizational Level These Job Roles Require

After you know the organizational files that are used by each job role and where they are located, you can use a template to collect your organizational values. Copy the results shown in Figure 3 and paste them into the template shown in Figure 4. This approach gives you the opportunity to avoid missing information during the analysis phase, and drives the business to collect this kind of information in a structured and babysitting approach.

Figure 4 shows an example of templates that have been filled with organizational-level values for each job role. For each job role, in the rows, the business has to insert the values for each organizational level. So the role Asset Master Data Maintenance Clerk will work constrained only on Company code 1000 and Business area GB01. All other values show an asterisk, which means no restrictions.



ROLES	DESCRIPTION	BUKRS	GSBER	KKBER	KOART	KOKRS	WERKS
		Company code	Business area	Credit control area	Account type	Controlling area	Plant
Z:CO_PA_001	Asset Master Data Maintenance Clerk	1000	GB01	*	*	*	*
Z:FI_AP_001	Account Payable Clerk	IT10,FR14	*	*	*	*	*
Z:FI_AP_002	Asset Manager	2000	GB01	*	*	*	GB01
Z:FI_AP_004	Accounts Receivable	*	*	*	*	*	*

Figure 4 Organizational-Level Values Example of Filled Template

Tip 90

Defining a Template for Gathering the Nonorganizational Constraints of Job Role Data

You can set up and gather nonorganizational constraints by using a spreadsheet and simplifying the technical steps at the business level.

After the organizational-level constraints discussed in Tip 89, the next step is to identify the nonorganizational constraints. All authorization object fields, except the field in Table USORG_DB, are not at the organizational level. Nonorganizational constraints, in some cases, are very difficult and time-consuming to manage. Also, you may find that it's difficult to collaborate with the business leaders, who often are not technical people.

As a best practice, don't avoid expanding these nonorganizational authorization objects constraints. Some example of nonorganizational constraints during the analysis phase are segregating the document type in the purchase order, billing document, or material document, or segregating the material master data views by department, and so on. Each of these examples corresponds to an authorization object.

Due to the deep technical level of this analysis step, it's essential to set up a correct template to optimize the collection of these kinds of data and business requisites.

And Here's How ...

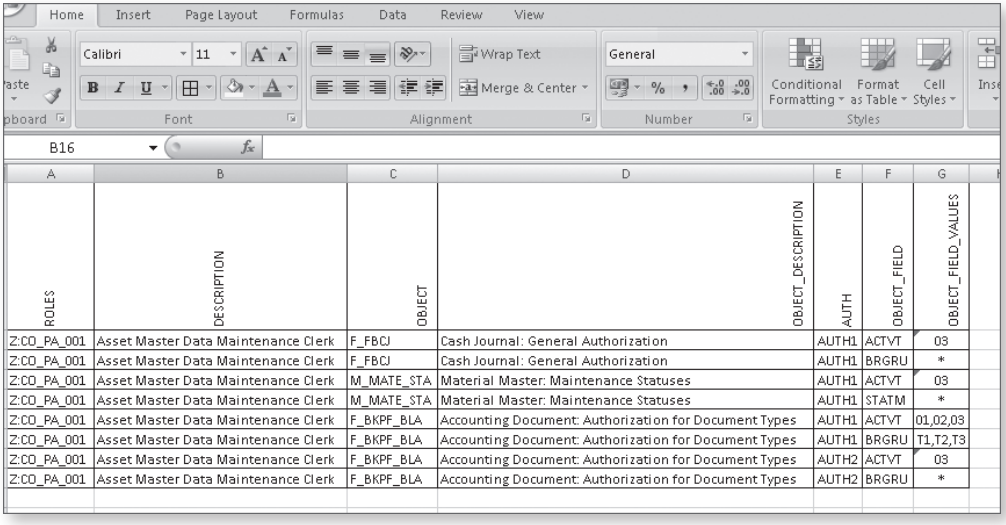
During the analysis phase, all finance job roles that are defined could display all financial document types, but only certain people should be editing the financial

documents. What's the best way to set up a gathering authorization document to set up this analysis and deploy?

After you've determined and selected the authorization objects to manage, you can find which job roles intercept these authorization objects. Using Table USOBT_C, you can correlate the transaction code and the authorization objects.

After you set up a proper template, gather the correct constraints for each job role defined, or for each user ID if there's a segregation based on user ID. Figure 1 shows the example template. The following list describes each column:

- **Column A: ROLES:** Represents the job role technical name.
- **Column B: DESCRIPTION:** Represents the job role short description.
- **Column C: OBJECT:** Represents the authorization object name selected as sensible, in other words, to manage which values should be included in the sensible authorization object.
- **Column D: OBJECT_DESCRIPTION:** Represents the authorization object description.
- **Column E: AUTH:** Represents the Transaction PFCG authorization concept. A role can be allowed to see all document types but is allowed only to change a particular type. The authorization object that protects the document type is the same but is necessary for defining two different authorizations.



ROLES	DESCRIPTION	OBJECT	OBJECT_DESCRIPTION	AUTH	OBJECT_FIELD	OBJECT_FIELD_VALUES
Z:CO_PA_001	Asset Master Data Maintenance Clerk	F_FBCJ	Cash Journal: General Authorization	AUTH1	ACTVT	03
Z:CO_PA_001	Asset Master Data Maintenance Clerk	F_FBCJ	Cash Journal: General Authorization	AUTH1	BRGRU	*
Z:CO_PA_001	Asset Master Data Maintenance Clerk	M_MATE_STA	Material Master: Maintenance Statuses	AUTH1	ACTVT	03
Z:CO_PA_001	Asset Master Data Maintenance Clerk	M_MATE_STA	Material Master: Maintenance Statuses	AUTH1	STATM	*
Z:CO_PA_001	Asset Master Data Maintenance Clerk	F_BKPF_BLA	Accounting Document: Authorization for Document Types	AUTH1	ACTVT	01,02,03
Z:CO_PA_001	Asset Master Data Maintenance Clerk	F_BKPF_BLA	Accounting Document: Authorization for Document Types	AUTH1	BRGRU	T1,T2,T3
Z:CO_PA_001	Asset Master Data Maintenance Clerk	F_BKPF_BLA	Accounting Document: Authorization for Document Types	AUTH2	ACTVT	03
Z:CO_PA_001	Asset Master Data Maintenance Clerk	F_BKPF_BLA	Accounting Document: Authorization for Document Types	AUTH2	BRGRU	*

Figure 1 Example of Template to Gather Nonorganizational-Level Constraints

- **Column F: OBJECT_FIELD:** Represents the authorization object field.
- **Column G: OBJECT_FIELD_VALUES:** Represents the authorization object values.

After the template is filled, all information and decisions are available for the Transaction PFCG operator. With this structure, there are no ambiguous colors to decipher any notes that are unnecessary

This template allows you to do the following:

- Process the template mechanically through a Microsoft Excel macro if necessary to simplify the definition of collected data in the system.
- Reduce the time needed to populate these kinds of data.
- Discover gaps in the analysis phase.
- Avoid starting with the development phase and discover missing or unnecessary information.

Tip 91

Using Pivot Tables and Authorization Reports to Customize Data for the Reader

You can use a pivot table to save time during authorization analysis and improve your authorization reporting/documentation.

Documenting authorizations can be a difficult task—it's not easy at first to understand technical data from a business point of view. Additionally, there are several ways to represent this kind of data, but it's essential to keep in mind who is reading this data: a security administrator, an auditor, key users, or a business process owner. Each one of these readers will read the data from a different point of view.

By using the pivot table tool, you can make the representation of these data individual to the reader's point of view.

And Here's How ...

The first step is to define a set of standard and shared (business process owners, auditors, key users, etc.) queries. In other words, you have to define which data you want to show up based on the decision of the future readers of these reports.

A security administrator will want to know all of the simple roles and all of the transaction codes inserted for each composite role defined in the system. There are some tables to relate for answering this request. The minimum tables to relate are the following:

- **Table AGR_AGRS:** Shows all simple roles for each composite role
- **Table AGR_TCODES:** Shows all transactions for each simple role
- **Table AGR_DEFINE:** Shows the definition of all roles in the system
- **Table AGR_TEXTS:** Shows the role description for each simple or composite role
- **Table TSTC:** Shows all transaction codes defined in the system
- **Table TSTCT:** Shows the description for each transaction

You can enhance this relationship by adding the SAP module and component for each transaction or by correlating the transaction with the statistics usage.

By using the SAP Query tool or exporting and correlating this table in a tool such as Microsoft Access, you can define a database as shown in Figure 1. This figure illustrates how the preceding tables are related to extract and create the database that contains ROLE COMPOSITE, ROLE SIMPLE, and TRANSACTION CODE.

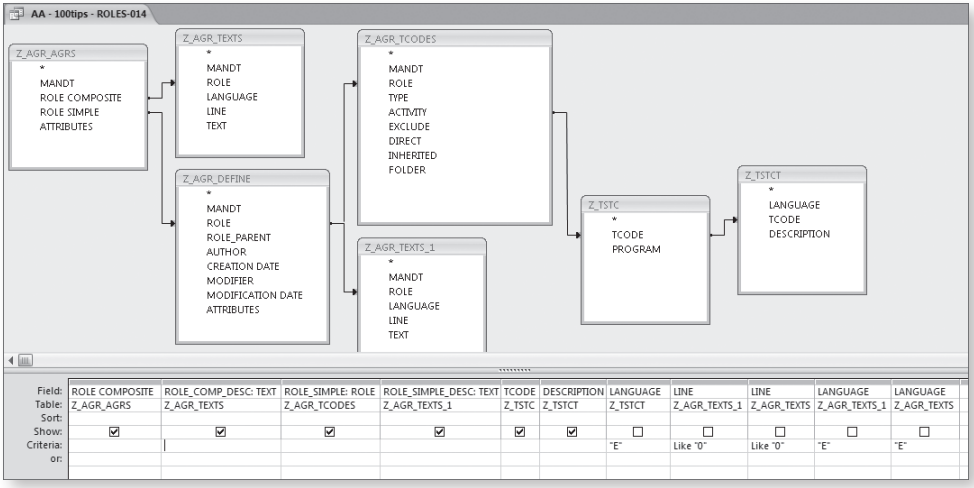
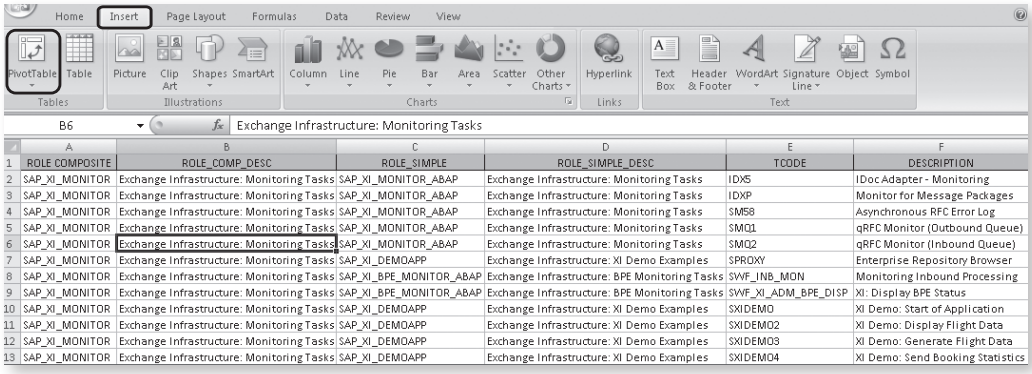


Figure 1 Relates Tables in Microsoft Access Database

When you perform the query shown in Figure 1, pay attention to the master language used in your roles. The query filters all text language in English.

After the query is performed, you have to export these data (in this example, we've only exported an SAP standard composite role) into a spreadsheet and make a pivot table (as shown in Figure 2) by clicking on the Insert ribbon button and then clicking on PivotTable.



	A	B	C	D	E	F
1	ROLE COMPOSITE	ROLE_COMP_DESC	ROLE_SIMPLE	ROLE_SIMPLE_DESC	TCODE	DESCRIPTION
2	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_MONITOR_ABAP	Exchange Infrastructure: Monitoring Tasks	IDX5	IDoc Adapter - Monitoring
3	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_MONITOR_ABAP	Exchange Infrastructure: Monitoring Tasks	IDXP	Monitor for Message Packages
4	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_MONITOR_ABAP	Exchange Infrastructure: Monitoring Tasks	SM58	Asynchronous RFC Error Log
5	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_MONITOR_ABAP	Exchange Infrastructure: Monitoring Tasks	SMQ1	qRFC Monitor (Outbound Queue)
6	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_MONITOR_ABAP	Exchange Infrastructure: Monitoring Tasks	SMQ2	qRFC Monitor (Inbound Queue)
7	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_DEMOAPP	Exchange Infrastructure: XI Demo Examples	SPROXY	Enterprise Repository Browser
8	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_BPE_MONITOR_ABAP	Exchange Infrastructure: BPE Monitoring Tasks	SWF_INB_MON	Monitoring Inbound Processing
9	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_BPE_MONITOR_ABAP	Exchange Infrastructure: BPE Monitoring Tasks	SWF_XI_ADM_BPE_DISP	XI: Display BPE Status
10	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_DEMOAPP	Exchange Infrastructure: XI Demo Examples	SXIDEMO	XI Demo: Start of Application
11	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_DEMOAPP	Exchange Infrastructure: XI Demo Examples	SXIDEMO2	XI Demo: Display Flight Data
12	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_DEMOAPP	Exchange Infrastructure: XI Demo Examples	SXIDEMO3	XI Demo: Generate Flight Data
13	SAP_XI_MONITOR	Exchange Infrastructure: Monitoring Tasks	SAP_XI_DEMOAPP	Exchange Infrastructure: XI Demo Examples	SXIDEMO4	XI Demo: Send Booking Statistics

Figure 2 Insert a Pivot Table from the Ribbon Menu in Microsoft Excel

The Microsoft Excel tool will automatically highlight the entire area of the sheet; click on OK to create a new sheet with the empty pivot table to fill as you want. Drag and drop the column header from the PivotTable Field List (box on the right) to the PivotTable (on the left) to set up your pivot table as shown in Figure 3.

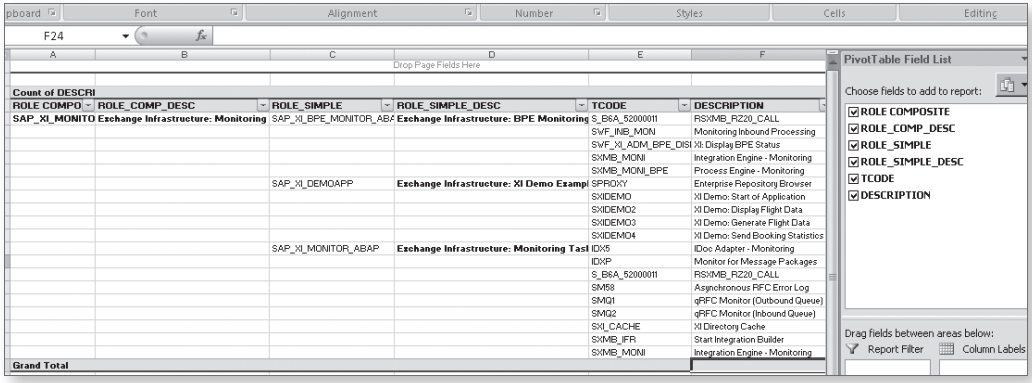


Figure 3 Pivot Table to Show for Each Composite Role All Simple Roles and All Transactions

You can create your standard report to set up your documentation for your target readers. This approach gives you the opportunity to set up a standard report to save time. After you've created a query, you can reuse it and provide a professional layout for all of your published documentation.

Part 8

Continuous Compliance and Governance

Things You'll Learn in this Section

The goal of the continuous compliance phase in the authorization field is mainly to ensure that all of your users and roles are validated for the respective owners. This phase is not always simple to prepare because you are documenting the authorization concept so that business owners can scrutinize the authorizations of their users. There are no tools that can accomplish these tasks as smoothly as SAP BusinessObjects governance, risk, and compliance solutions (SAP GRC). However, it's still essential to have a good authorization concept with few technical mistakes or oversights.

In this part of the book, you'll learn some tips to help prepare your authorization concept to be compliant and simple to review and validate from a business perspective.

Tip 92

Defining Data for User Revalidation

You can define which data should be delivered during a user revalidation and how by setting up user master record attributes and using a query to create the revalidation file.

Every day, user managers perform many actions on the user master record. To guarantee the appropriate compliance, you must periodically (at least once a year) perform a user revalidation. The goal of this activity is to keep the people involved in security processes (user data manager and business process owner) informed regarding the main users' data. Due to organizational changes, it's possible that a user's authorizations don't match the current organization. For example, a user might be currently assigned to an Accounts Payable Clerk role but the role for that user should be Accounts Receivable Clerk instead due to a promotion or job change. If the business owner is kept informed, he could follow up on this misalignment.

Every user responsible for a department has to confirm that the users assigned to him (and the corresponding authorizations) are still correct. When you have many hundreds or thousands of users, it's difficult to dispatch a huge amount of data to the people responsible for analyzing this. You have to organize your actions in advance. In this tip, you'll learn how to quickly set up user master records attributes and how to use the query to produce the user revalidation file.

And Here's How ...

A user revalidation should be easy: Every responsible user receives a list of assigned users (with related data) and has to confirm (or not) that the data are still correct. The first question to answer is which user's data are relevant for user revalidation?

Here are some suggestions:

- User ID
- Last name
- First name
- Last logon date
- Company
- Function
- Department
- Cost center
- Assigned authorizations
- HR personnel number (CID)
- Users responsible for a specific business area
- Job

Working with HR

Some of these data should be generally available in the HR department. Unfortunately, the relationship between HR data and user master record data is not so easy. In some companies, the HR system is not a SAP solution. In many companies, the HR system is a SAP solution but is not in the same instance as the other business processes (and related SAP modules—FI, CO, MM, and SD).

You should involve the HR department to make a link between their data and the user master record. For example, it's mandatory that the HR Infotype 0105 (subtype 0001) is maintained in the personnel master data (HR-PA) (see also Part 1, Tip 13).

One of the most important pieces of information for each user is the name of the person responsible for the department, which should come from the HR department. It will be mandatory to distribute this data for revalidation.

Job information is probably the most difficult to retrieve. A job is the link between the personnel number's position (in the organizational structure: HR-OM) and the assigned authorizations to the user ID. A good idea is to define a relationship between the job and the authorization roles. If this link is available, by enquiry into the HR data, you should be able to intercept incorrectly assigned roles in respect to the ones related to the job.

Manage User Master Record

Next you have to manage all information for each user ID in the user master record. A practical solution is to fill in the available fields in Transaction SU01 (Maintain User). Some fields are always present in Transaction SU01, but many others are available only if you click on the MORE FIELDS button circled in Figure 1.

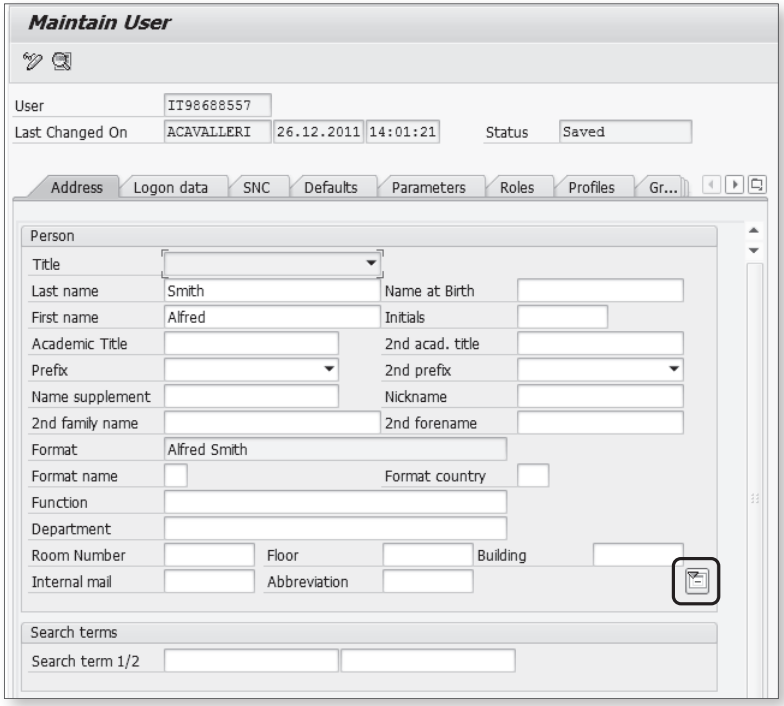


Figure 1 Fields Available in Transaction SU01

Our suggestion is that you analyze all available fields in all tabs of Transaction SU01 to map all of the necessary data for user revalidation. For example, because a field for the personnel number isn't available, many companies put it in the NICKNAME field.

Of course, you must guarantee that the data are aligned with the data in the HR department. The best solution is to create an interface (e.g., using HR triggers) that automatically aligns HR data with user master record data. A good solution is also to implement identity management.

After the user master record data are updated, you have to distribute all information to be confirmed to all responsible parties. A good, quick solution is to create specific queries with Transaction SQVI (Quick Viewer). After you execute Transaction SQVI, create a join query and then insert the table name to correlate (see Figure 2). You can easily create table joins and extract all necessary fields to a Microsoft Excel file. This example mass-extracted all assigned roles to users.

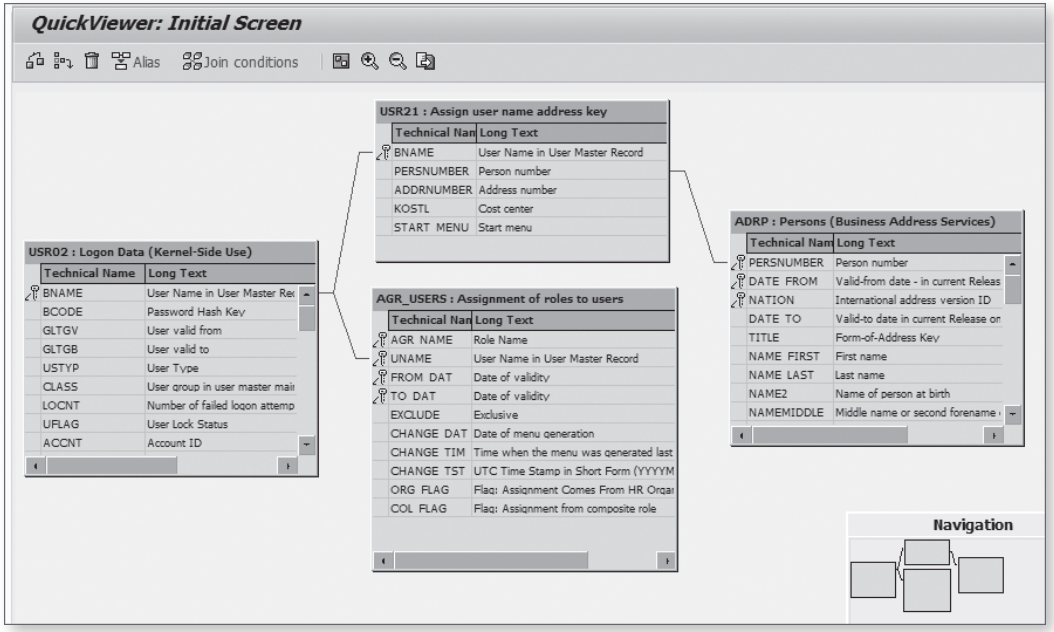


Figure 2 Transaction SQVI Sample

One of the biggest problems in user revalidation is related to authorizations. In many companies, a strong authorization concept based on roles is not well defined (see Part 3, Tip 37). Indeed, when there are many roles (not specifically manual profiles) assigned to users, it's crucial that they are somehow related to a job. In this case, composite roles are a good SAP standard solution.

Tip 93

Revalidating Roles and Providing Documentation for Analysis

To increase your authorization governance, you need to periodically revalidate all roles and then provide clear documentation that your target audience can analyze.

Many times, authorizations administrators limit their attention to a role name and description without going into a deeper level of analysis. At least once a year you have to perform a roles revalidation with the goal to confirm that the roles' content is still correct. This tip recaps the common technical errors to avoid and shows how to prepare and document role reporting to revalidate roles for the business.

And Here's How ...

Don't confuse user management with "authorization" management. In a roles-based access control (RBAC) approach, a role must be valid for an abstract user and not for a specific person.

A roles revalidation can be split into two different goals, which we look at in the following subsections.

Technical Revalidation

With this step, you want to certify that a role is correct from a technical point of view. This means that it's compliant with the authorization concept you've adopted. In many companies, each area (normally each SAP module: FI, CO, MM, SD, etc.) has the ownership of roles management. This not only in terms of content but also physically with Transaction PFCG (Role Maintenance). When many administrators are operative, there is a high probability that the global role set is

not harmonized. A different naming convention could be adopted and a different roles architecture too. A security and authorization guideline must be put in place to create a formalized model.

Let's look at some major problems that could occur during technical revalidation.

S_TCODE Critical Error

The most critical technical error (which occurs in the business revalidation) is in the maintenance of S_TCODE authorizations. A major concern is if you have a misalignment between transaction codes that have been inserted in the role menu and transaction codes in the Authorization tab of a role.

You must avoid this situation as much as possible because it will be difficult to document the real transactions granted with this role to the many involved owners (see also Part 3, Tip 40 to identify this type of misalignment).

Assign a User Directly to a Simple Role

One other typical error in a security concept based on composite roles and simple roles is to assign a user directly to a simple role. If the assumption is that a composite role is the technical implementation of a job, it will be very difficult during a user revalidation to determine which job a user is related to.

You can find the simple roles directly assigned to a user by using Transaction SE16 to browse Table AGR_USERS where the ORG_FLAG and COLL_FLAG fields are equal to null (blank). Exclude your composite role from the results of this query, and you will find the simple roles directly assigned.

Manually Maintain the Organizational Levels Directly in the Authorization Tree

A classic technical error is to manually maintain the organizational levels directly in the authorization tree instead of the specific pop-up form available in Transaction PFCG (Role Maintenance).

This action can be very dangerous when you use the derived roles. It will influence the documentation of the roles too because it will not be clear to which companies this role is granting authorization (see also Tip 97).

Business Revalidation

When all roles are correct from the technical point of view, you have to document them to be validated. Keep in mind that in a RBAC approach, the roles' content

will be used also in user management. In fact, the junction point between user processes and role processes is the role. When business area owners have to assign authorizations to their users, they will likely use the roles' documentation to find the best choice.

The goal of the business revalidation is to certify that each role's content is correct against the implemented process and organization.

The main issue in business revalidation is to determine the technical level of the details you'll deliver. If you document all of the authorization trees for all roles (available in Table AGR_1251), your business users will probably not understand because it's too technical.

Let's go over the steps needed to prepare the documentation to revalidate roles for the business.

First Level: Role Name and Description

All role descriptions are recorded in Table AGR_TEXTS (File Structure for Hierarchical Menu – Customer), and the SPRAS (language) field will determine the language used in the descriptions. Pay attention to the logon language field because it will be used as the value every time you maintain a role's description.

Second Level: Role Name and Transaction Code

One important level of detail is the transaction codes inserted in the roles. In this case, you must decide if you want to consider the transactions in the role's menu or the transactions in the `S_TCODE` authorizations. You should use the transactions in the role's menu because the user will see them in the SAP GUI. Table AGR_TCODES (Assignment of Roles to Tcodes) contains the role's menu transactions.

Third Level: Role and Organizational Values

Document the organizational-level values (company code, plant, sales organization, etc.) because these are the most important authorizations related to the business. This information is available through Table AGR_1252.

Also, make sure to document your composite roles. With Table AGR_AGRS (Roles in Composite Roles), you can find all simple roles inserted in a composite role.

Tip 94

Making Sure Users Are Assigned Only to the Roles and Transactions They Use

You can determine whether a user has too many roles assigned to them by exporting statistical data to a spreadsheet.

It's normal to find roles with hundreds (or even thousands) of transactions defined in a role's menu. If you ask your business users to tell you which transactions they need, they will probably ask for all available transactions. However, security guidelines indicate that each user should only be authorized for the minimum transactions he needs (the principle of least privilege). Therefore, you need a way to check if there are too many roles and transactions granted to your users.

And Here's How ...

While Transaction ST03N (Workload and Performance Statistics) is mainly used to verify an instance's performance, it's a fantastic resource for security managers as well.

After accessing the transaction as shown in Figure 1, you can inspect which transactions have been used in a certain period. On the top-left side, you can select the period of analysis (DAY, WEEK, or MONTH), and on the bottom-left side, you can select the TRANSACTION PROFILE to display all transactions that have been executed.

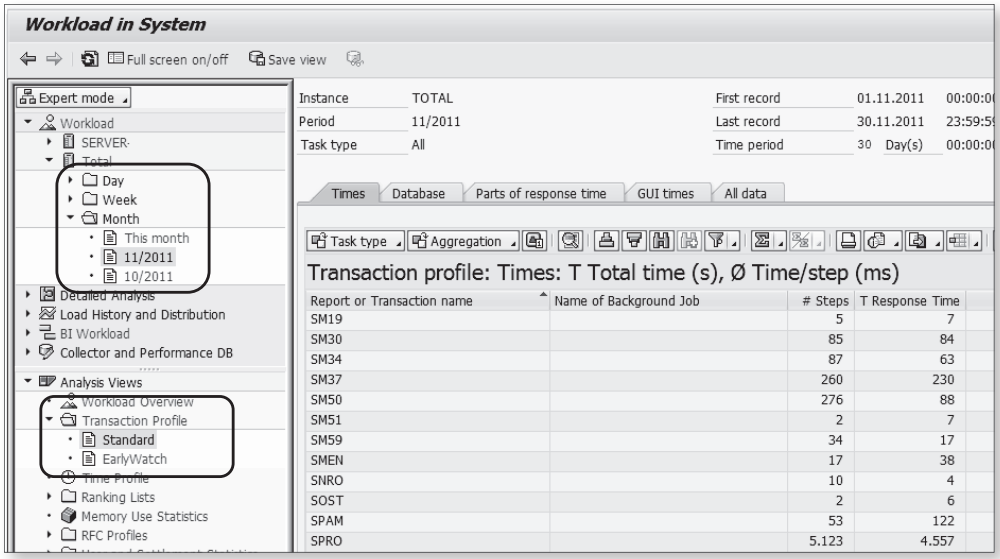


Figure 1 Transaction ST03N

Because there are normally many application servers in a production system, select Total in the Workload branch (top-left area in Figure 1). In the Analysis views -> Transaction Profile area, choose standard to view transactions statistics.

Keep in mind that the default retention period for statistics is the current month plus the previous two. If you need more time, you have to modify the system configuration.

Double-click on a transaction code to see all of the users who have used it. Figure 2 shows all of the users who have executed Transaction SE16.

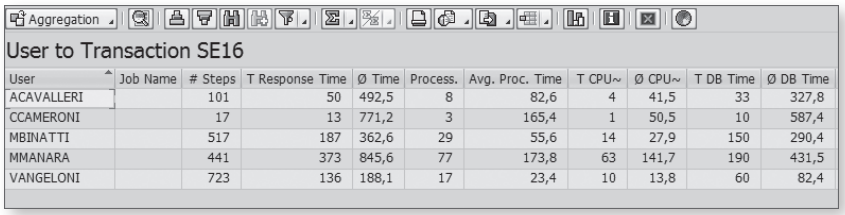
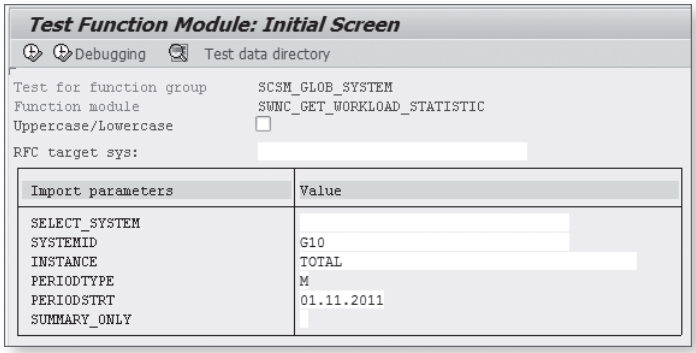


Figure 2 Transaction SE16 User Statistics

Although all information is available in Transaction ST03N, it's difficult to periodically extract, analyze, and compare all of the data with a role's transactions.

Unfortunately, the information stored by Transaction ST03N isn't easy to extract with Transaction SE16 (Data Browser) because the data are stored in Table MONI (Monitor), which is a "cluster" table. In other words, you can't see the same data in the format shown in Transaction ST03N.

If you want to extract all Transaction ST03N data, you can use a specific function module. Using Transaction SE37 (ABAP Function Modules), you can execute a function called `SWNC_GET_WORKLOAD_STATISTIC` (via the Execute button) as shown in Figure 3.



« *Figure 3 SWNC_GET_WORKLOAD_STATISTIC Function Module Import Parameters*

If the selection parameters are correct, a set of filled tables will be returned. If you open Table USERTCODE by clicking on the table name, you can see a result similar to the one shown in Figure 4. In this case, you can easily download all table content in a spreadsheet file by following this menu path:

System • List • Save • Local File • Spreadsheet

ACCOUNT	ENTRY_ID		COUNT	DC
ACAVALLERI	!OSA_01	R	3	
ACAVALLERI	!_SA_01	R	1	
ACAVALLERI	OKW7	T	5	
ACAVALLERI	CAPS	T	1	
ACAVALLERI	CDESK	T	1	
ACAVALLERI	CG3Y	T	1	
ACAVALLERI	GR23	T	3	
ACAVALLERI	GR33	T	9	
ACAVALLERI	GRAC_SPM	T	2	
ACAVALLERI	GRR1	T	6	
ACAVALLERI	GRR3	T	8	
ACAVALLERI	IH01	T	1	
ACAVALLERI	IH02	T	1	

» *Figure 4 SWNC_GET_WORKLOAD_STATISTIC Function Module Output*

Tip 95

Using Indirect Role Assignment to Simplify User Maintenance and Reporting

You can use the HR organizational structure (HR-OM) to distribute authorizations to users.

The classical approach to assigning a role to a user is direct assignment via Transactions SU01 (User Maintenance) or PFCG (Role Maintenance). However, there's a more powerful scenario you can consider: indirect assignment of roles to users using an organizational structure as a bridge between users and roles. With this scenario, it will be much easier to share security documentation with business contacts who technically own the data and users but often have very limited technical knowledge.

And Here's How ...

One of the most difficult decisions a user manager has to make is how to delegate duties among business contacts. This is because they speak a different language—security and authorization managers are often very technical, whereas business references are not necessarily SAP experts.

For instance, when you make a pivot table from data stored in Table AGR_USERS to document the link between the roles and users, the result will be similar to the one shown in Figure 1 (with roles in the rows and users in the columns). Business contacts will find it very difficult to match this output with their organization.

chette di riga	ACAVALLERI	AFAZIO	AGOMBA	APAPA	CCORONATO	FBERETTA	FCOLOMBO	FF_USER	FMORLEO	FPASTORE	FSOFIA	GCANNATA	GGABRIELE	ISANTANOCITA	JALBAMONTE	MABBA'	MARRIGONI	MPIROTTA	MTRESOLDI	NCAVALLERI	NPARRAVICINI	PALBESANO	PPERABONI	RMARELLI	RPULICI	TEST_200	TEST_IAM_04	VNAVA
BC:T_001_V															1		1		1									1
BC:T_999_V								1																				
CO:C_XX_0001																1							1	1				
CO:T_001_M																1							1	1				
CO:T_005_M																1							1	1				
CO:T_016_V																1							1	1				
FI:C_MD_MANAGER										1																		
FI:C_XX_0001			1		1	1								1														
FI:T_001_E		1																										
FI:T_001_M		1																										
FI:T_001_V		1																										
FI:T_002_E		1																										
FI:T_002_M		1																										
FI:T_002_V		1																										
FI:T_005_M										1																		
FI:T_007_M										1																		
FI:T_007_V			1		1	1								1		1							1	1				
FI:T_019_M										1																		

Figure 1 Table AGR_USERS Pivot Table

Instead of using a direct assignment (of roles to users), you can take advantage of the HR-OM component. As shown in Figure 2, you can assign the role (AG object) and users (US object) to the position (S object). (Refer to Part 1, Tip 10.)

Don't confuse this scenario with the classical HR approach which distributes "persons" (object type P) over the structure. The security approach doesn't need any specific HR implementation because it distributes "users" (object US) over the structure (see highlighted user in Figure 2). (See Part 1, Tip 10.)

One other big difference is that you can link many users to a single position without having to consider the percentage values (as it is for P objects).

When this scenario is implemented, you have a graphical way to communicate "who does what" for your business contacts. For them, it will be much easier to navigate in a tree instead of a pivot table. In fact, you can also assign Transaction PPOSW (Display Org. and Staffing [WF]) to your business contacts to let them analyze the security status at all times.

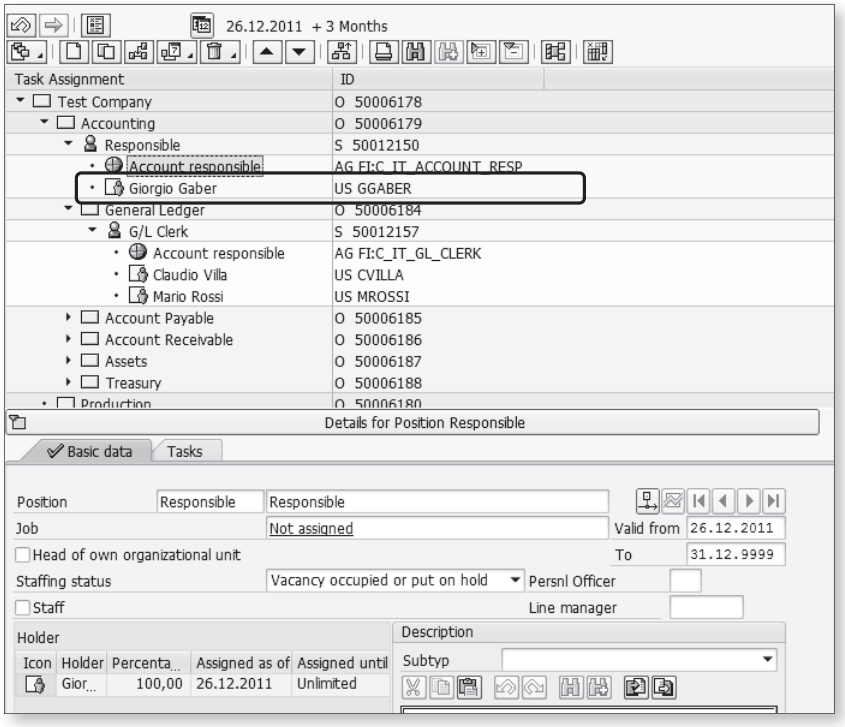


Figure 2 Organizational Structure for Security Management

One other way to make a printable documentation is using Transaction PPSS (Display Structure) as shown in Figure 3. In the parameters selection, the Evaluation Path value is very important because it will influence the final output that defines which elements must be displayed.

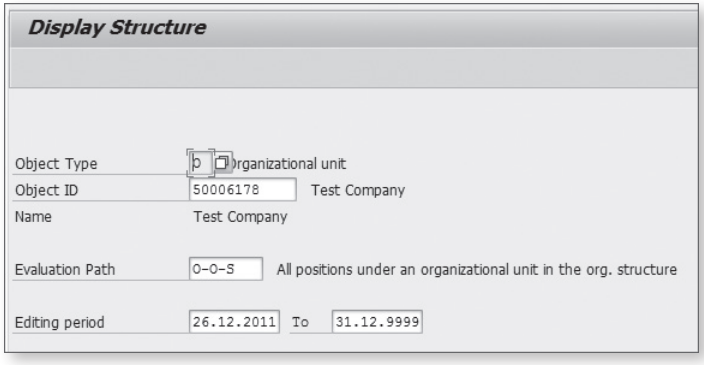
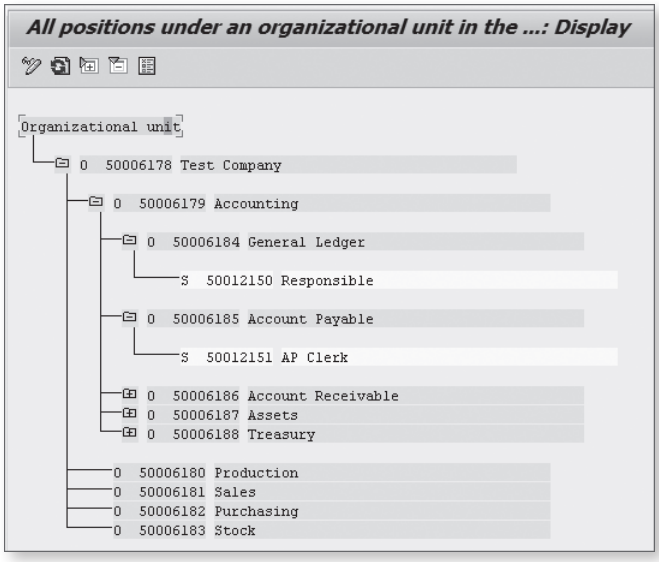


Figure 3 Transaction PPSS Parameters

Figure 4 shows an example of the organizational structure with only organizational units (object O) and positions (object S).



« *Figure 4*
Transaction PPSS Output

Tip 96

Defining Business Owners

By understanding each type of owner that's present in a business, you can easily determine which person manages what data or responsibility to ensure proper governance.

Defining the owners of a company is not very clear, especially during periodical revalidations (users and roles). Often an owner receives a document to be validated and he has difficulties because the goal and the responsibilities are not well defined.

In security processes, many different areas are involved. For each area, one or more owners have to be identified. This tip explains some of the possible owners that you can use in your company during the revalidation process.

And Here's How ...

Remember that the classical SAP security/authorization concepts are based on the RBAC (role-based access control) logic. This means that you should divide the authorization processes for owners into (at least) two main areas:

- User processes
- Role processes

To avoid confusion, you must always have a clear idea of whether a decision is related to users or roles. The typical situation in which it's not clear how to proceed occurs when a user asks for a new transaction. The requested transaction could be added to a role not yet assigned to the user, or a new role (containing the transaction) could be assigned to the user.

In recent years, the Segregation of Duties (SoD) logic has added a new level of complexity. When a new request causes a SoD conflict, a new set of owners are involved.

Unfortunately, when the responsibilities aren't well addressed, the security team becomes responsible for all mistakes and misunderstandings.

Following you will find a proposal of who should own which responsibilities:

- **Business area owner**
  Responsible for the users in his area or department. Every user must be assigned to a responsible person who will manage each change of the user's data.
- **Business process owner**
  Responsible for defining the sequence of all activities that are mandatory in his processes. The decisions made by a business process owner should be valid across all company departments. For each business process, there is one process owner.
- **Data owner**
  Responsible for the most important information of the process: the data. Each data owner must assure that the data are correctly created and maintained. Every time a data is involved, data owners must validate the request.
- **Role owner**
  Responsible for the content (transactions and authorizations) of roles. He must communicate with data owners and process owners to guarantee the final integrity for each role's change.
- **SoD rules owner**
  Responsible for physically maintaining the set of rules necessary to perform risk analysis.
- **SoD risk owner**
  Responsible for defining a risk in terms of content and level of severity (critical, high, medium, low).
- **SoD mitigation control owner**
  Responsible for mitigation actions.

Many other people can also be relevant to maintain a high level of governance such as internal controllers, business process analysts, and so on.

All security and authorization processes should be well designed and written with a clear indication of the actors' involvement and responsibilities.

When all owners have been identified, it's important to formally communicate this fact in the company to avoid misunderstanding.

Tip 97

Finding Misalignments between Organizational-Level Pop-Ups and Authorization Data in Derived Roles

You can optimize your authorization role concept and governance by using derived roles.

Derived roles can enhance your governance and simplify daily authorization maintenance. Without using derived roles, you have to manage several simple roles separately. By using derived roles, you can maintain father roles and specify the data domain into all derived roles. However, you can create a gap in governance if you don't use this type of role properly. This tip shows you how to properly maintain derived roles to avoid this pitfall.

And Here's How ...

When you create a father role and its transactions contain authorization objects with organizational-level fields, an organizational tab pop-up appears when you go into the Transaction PFCG Authorization tab.

For this tip, create a father role with Transaction MM01. When you click on an Authorization tab, an organizational pop-up will appear as shown in Figure 1.

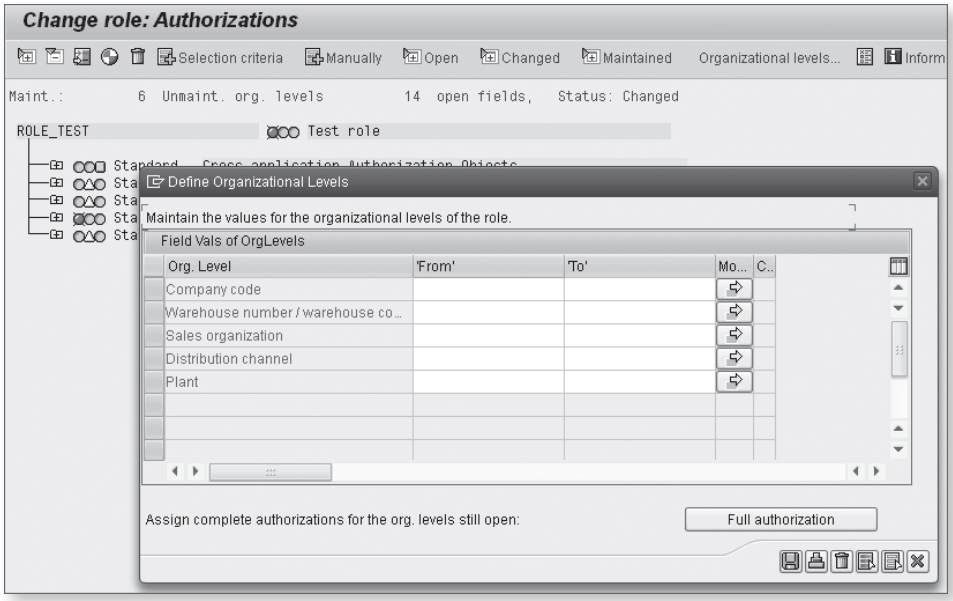


Figure 1 Define Organizational Levels Pop-Up in the Transaction PFCG Authorization Tab

Here, you insert the organizational-level values (e.g., enter an asterisk for all organizational levels), and the father roles is generally not assigned at the end-user level. Afterward, you create the derived role (child). In the same way, when you go in the AUTHORIZATION tab, you have to insert the organizational-level values. Inserting the value through the pop-up automatically fills all authorization fields with the value entered. Figure 2 shows all authorization objects (M_MATE_MZP and M_MATE_WRK) with the WERKS field populated.

From a governance point of view, you should never manually change an organizational-level field because you will overwrite the authorization value. The maintenance of organizational levels is not advised at the field level but only in the organizational-level panel, as shown in the DEFINE ORGANIZATIONAL LEVELS pop-up screen (see Figure 3).

Additionally, if you base your authorization analysis on an organizational-levels panel, you might make an oversight because there are different values between organizational-level pop-ups and organizational-level fields.

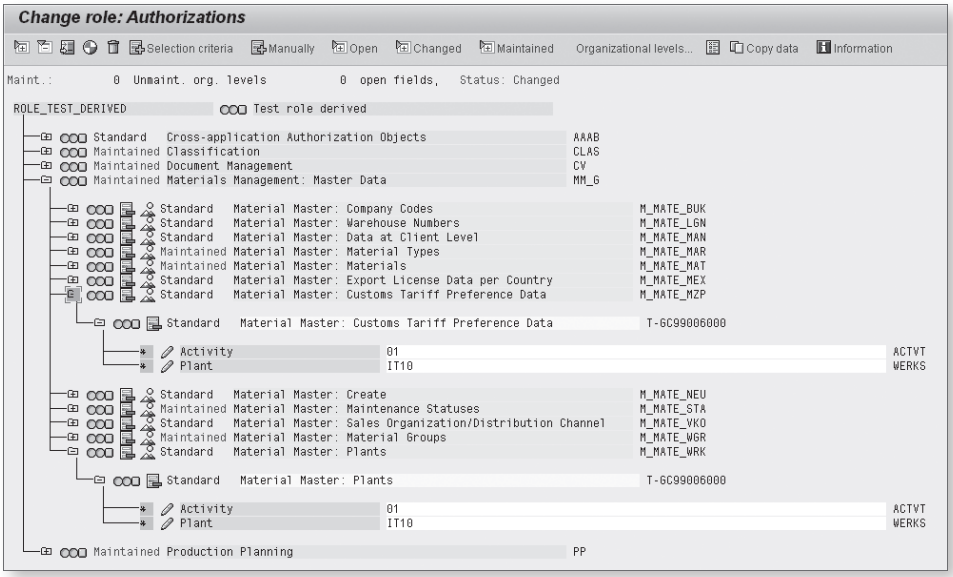


Figure 2 Organizational-Level Field WERKS Populated through Organizational Level Pop-Up

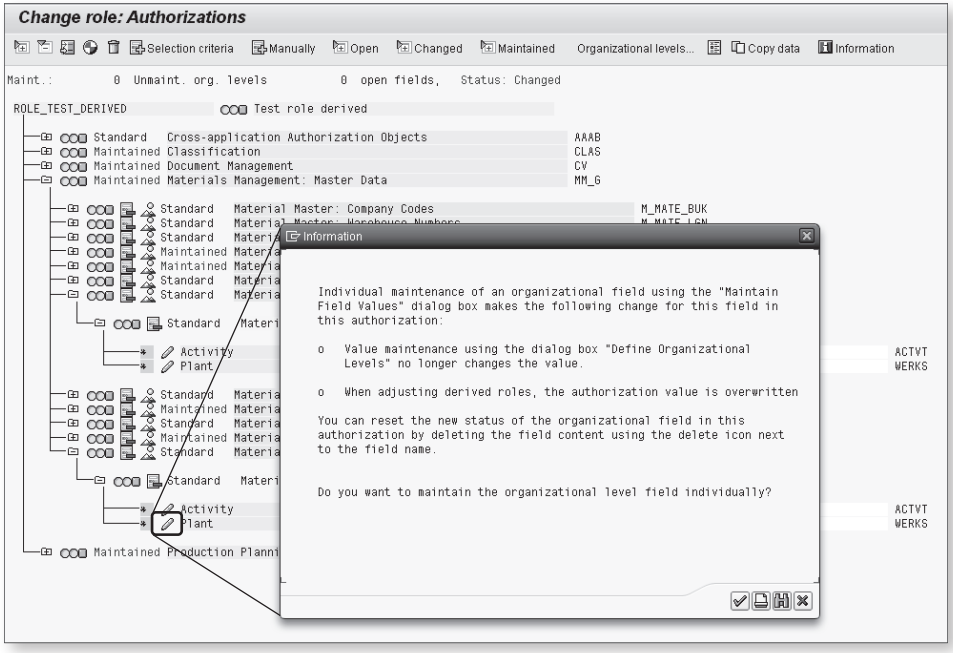


Figure 3 Field-Level Editing of an Organizational-Level Field

You can remediate this situation by easily finding these noncompliant situations through a targeted query.

All values entered through the organizational panel are stored in Table AGR_1252. If an organizational field is manually changed, this value is stored in Table AGR_1251 instead of Table AGR_1252. To find these values, you can perform the query on Table AGR_1251 as shown in Figure 4.

In the FIELD field, enter all of the organizational-level fields (you can retrieve these by browsing in Table USORG_DB). Afterward, insert the value $* in the LOW or HIGH fields if you're managing the range of exclusion.

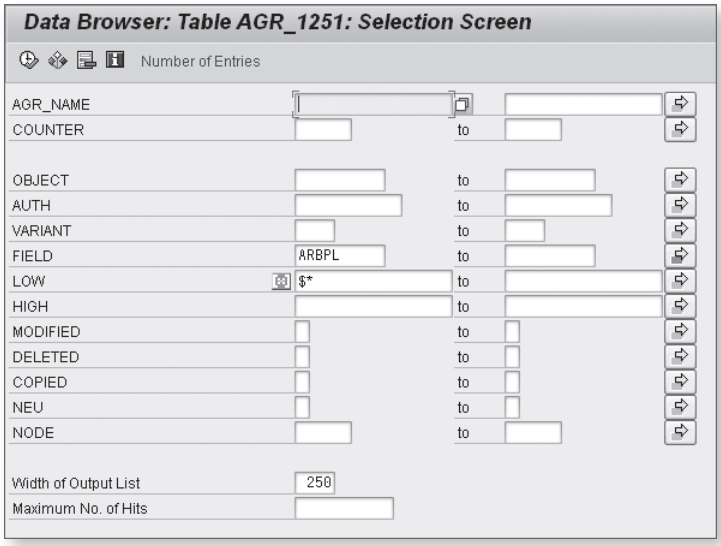


Figure 4 Table AGR_1251 Query to Find Misaligned Organizational-Level Values

Because all organizational values are stored in Table AGR_1251 with a variable that starts with $*, if you are looking for all organizational values that do not start with this character, you have found the misalignment.

Tip 98

Finding Manually Created Authorizations in a Role's Authorization Tree

You can find a role's authorizations that don't seem to be related to any transaction codes or that have been manually created, as indicated in the role's menu.

One of the most important goals of Transaction PFCG is to facilitate everyday role maintenance, maintaining a strong relationship between transaction codes and related authorization objects. It's important that all authorizations present in a role's authorization tree are logically related to the transaction codes granted with the role itself. Otherwise, you risk losing control of your security concept. Each manual authorization added into the authorization tree is not linked to the transaction in the role menu. That means when you remove a transaction from the role menu, the manual authorization objects present in the authorization tree will not automatically be removed. This tip shows you how to avoid and correct these misalignments.

And Here's How ...

Every time you insert a transaction code in a role's menu, Transaction PFCG (Role Maintenance) analyzes Table USOBT_C (Relation Transaction > Auth. Object [Customer]) to retrieve all mandatory authorization objects with SAP default values. As a result, when you go in the authorization tree of a role through Transaction PFCG, you'll find a set of created authorizations.

For example, every time you create a custom transaction code, you should establish the relationship between the transaction code and all authorization objects checked by the transaction's logic through Transaction SU24 (Auth. Obj. Check

Under Transactions). This is true for SAP standard transaction codes, too. If you find a missing authorization object, you have to link it with the related transaction code with Transaction SU24.

Instead of following this procedure, however, many security managers manually insert a new authorization directly in the role's authorization tree as shown in Figure 1.

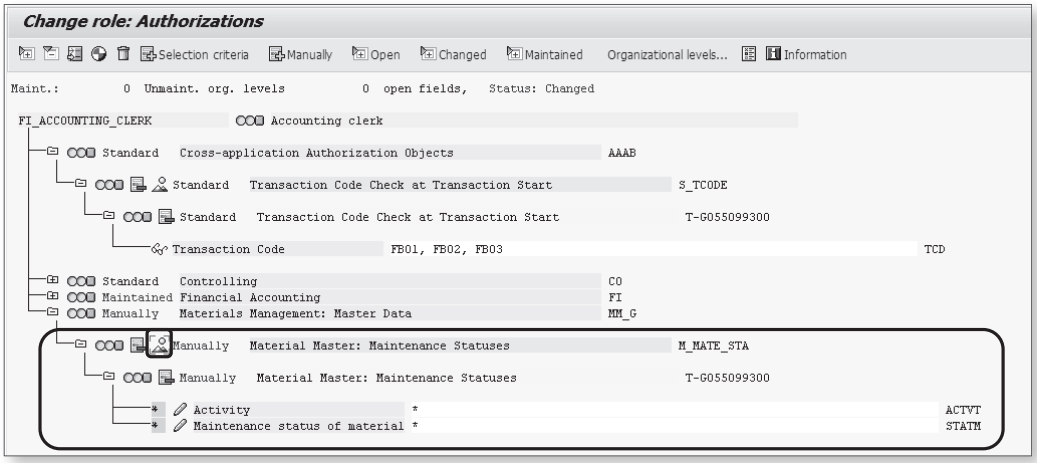


Figure 1 Manual Authorization

The last authorization, on authorization object M_MATE_STA (Material Master: Maintenance Statuses) (circled in Figure 1), in the authorization tree is indicated as manually.

Looking in the S_TCODE authorization you can see Transactions FB01, FB02, and FB03. All of these transaction codes can be used to create/maintain/display FI documents.

If you click on the highlighted icon with the sun on the mountains (), you'll see a pop-up form like the one shown in Figure 2.

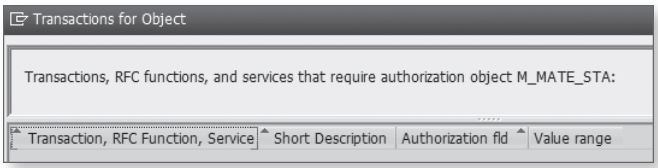


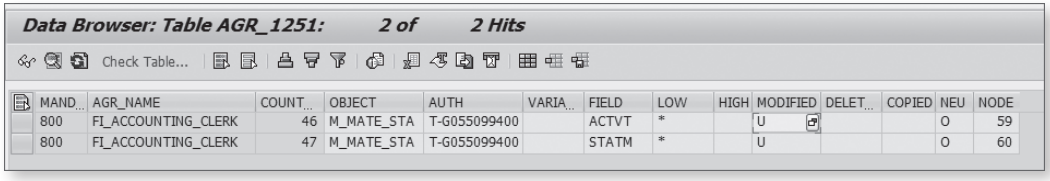
Figure 2 List of Transaction Codes Related to the Authorization Object

Because the output of Figure 2 doesn't display any transactions, this means that the authorization object M_MATE_STA, in this example, does not have any relationship. In such a case, there are only two possibilities:

- The relationship exists for at least one of the role's transactions, but Transaction SU24 was not "informed." In this case, you should update Transaction SU24 to create the link between the authorization object and the transaction.
- The authorization object M_MATE_STA is in an incorrect role, so you should remove it.

Finding Nonstandard or Manual Authorizations

If you want to retrieve all manually inserted authorizations, you can do so through Transaction SE16 by browsing Table AGR_1251 (Authorization Data for the Activity Group). As shown in Figure 3, manual authorizations are marked with the value "U" in the MODIFIED field. You have to enter your role in the AGR_NAME field and enter "U" or "M" in the MODIFIED field.



MAND...	AGR_NAME	COUNT...	OBJECT	AUTH	VARIA...	FIELD	LOW	HIGH	MODIFIED	DELET...	COPIED	NEU	NODE
800	FI_ACCOUNTING_CLERK	46	M_MATE_STA	T-G055099400		ACTVT	*		U			O	59
800	FI_ACCOUNTING_CLERK	47	M_MATE_STA	T-G055099400		STATM	*		U			O	60

Figure 3 Find Manual Authorizations

Tip 99

Substituting SAP Queries with Specific Transaction Codes

If you find that end users are creating or changing queries, you can easily substitute your SAP query with a specific transaction code.

SAP Query is a tool made for simple or complex data correlation. SAP Query doesn't substitute for a specific reporting tool, such as SAP Business Warehouse or SAP BusinessObjects. From a security point of view already discussed in Part 3, Tip 47, a query should only be created and defined by specific people and shouldn't be created or changed by end users.

However, if a company has allowed end users to use SAP Query for several years, they could be unwilling to convert and substitute a SAP query for a specific transaction. This will probably cause some end-user frustration but also may be difficult from a technical aspect. The first aspect is solvable mainly through training; the second is shown in this tip.

And Here's How ...

The solution for this issue requires two steps:

- To substitute an SAP query, convert the SAP query into a transaction code.
- Link the proper `S_TABU_DIS` authorization object values (and/or `S_TABU_NAM` if enabled in your system) through Transaction SU24. This authorization object protects the SAP table access through the SAP table authorization groups.

The common difficulty in the second step is finding all of the tables used in all SAP queries, which are defined to locate the correct authorization group values to insert into authorization object S_TABU_DIS through the Transaction SU24.

Link a Transaction Code to the Query (Program)

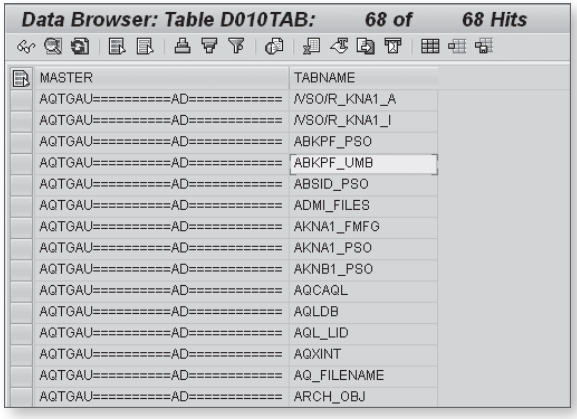
To convert a SAP query into a program, go to Transaction SQ01 and follow the menu path:

Query • More Functions • Generate Program

Alternatively, you can click on the Display report name menu entry to see the program names. You have now generated the program that will be assigned to a transaction through Transaction SE93.

Find All Tables Used in All SAP Queries

You can discover the link between a generated query program and all tables used from the query by browsing Table D010TAB as shown in Figure 1. This table has only two fields: MASTER and TABNAME. If you insert the program generated in the MASTER field, you find all related tables used from the query converted into a program.



« *Figure 2 Table D010TAB with the MASTER and TABNAME Fields*

After you've found all of the tables, highlight them and browse Table TDDAT to find the authorization group to insert in the S_TABU_DIS authorization objects. Table TDDAT gives you the link between a SAP table and the SAP table authorization group.

Tip 100

Using a Query to Find Manually Created Authorizations and Convert them to Roles

You can create a query to find all authorizations that were manually created and assigned to users.

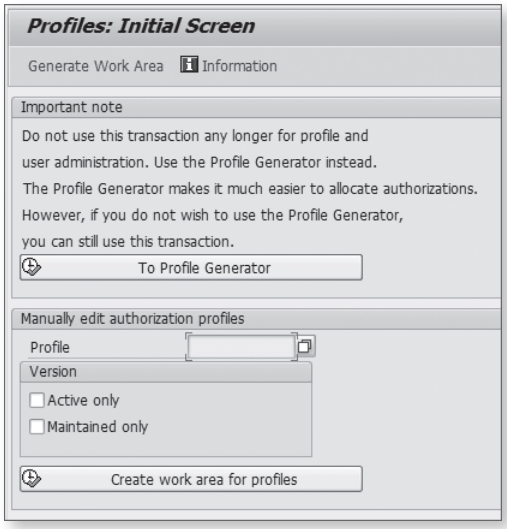
Now that the process of manually creating authorizations is obsolete, you need to increase your governance by making sure that your security concept (on SAP ABAP systems) is roles based. This means that you should only use Transaction PFCG (Role Maintenance) instead of the previously used Transactions SU02 (Maintain Authorization Profiles) and SU03 (Maintain Authorizations). If you have users with manually created profiles, the effort to maintain them is high, and it will be very difficult to create security documentation for the periodical revalidation.

This tip shows you how to discover all manually created authorizations assigned to users. In this way, you can use a unique way to define and then assign the authorizations as roles. This tip will be very useful because a mixed mode can became difficult to maintain in daily work and during the continuous compliance phase.

And Here's How ...

Defining Manual Authorization

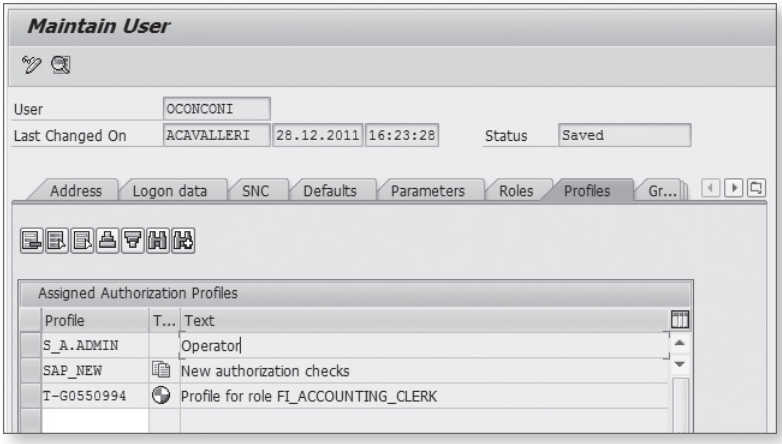
When you start Transaction SU02 (Maintain Authorization Profiles), you'll see a screen similar to the one shown in Figure 1.



« *Figure 1 Maintain Authorization Profiles Initial Screen*

If you carefully read the text in the IMPORTANT NOTE section of Figure 1, you'll understand that this transaction should not be used.

When you display the PROFILES tab in the user master record (Transaction SU01) for a certain user, you see profiles assigned to the user as shown in Figure 2.



» *Figure 2 Authorization Profiles Assigned to a User*

In the example shown in Figure 2, you can see three different profile types:

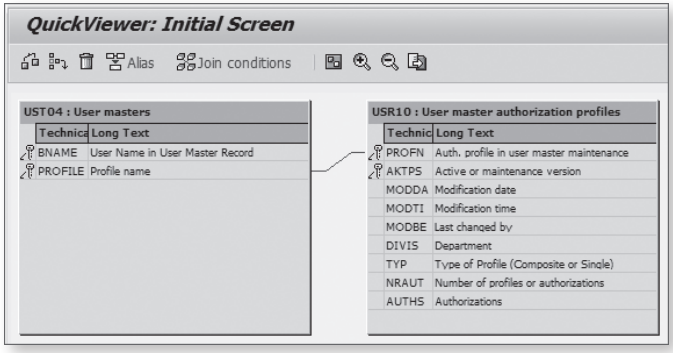
- S_A.ADMIN is a manual "single" profile (classified as an S character in the SAP table).
- SAP_NEW is a "composite" profile (classified as a C character in the SAP table).
- T-G0550994 is a "generated" (from Transaction PFCG) profile (classified as a G character in the SAP table).

In your SAP system, you should have (at least for end users) only generated profiles assigned to users.

It's simple to retrieve all users with manual or composite profiles assigned.

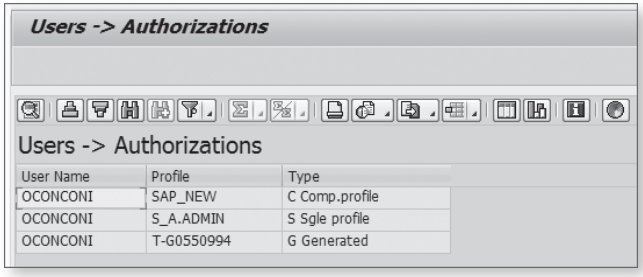
Finding Manual Profile Assigned

With Transaction SQVI (Quick Viewer), you can create a query joining Tables UST04 (User Masters) and USR10 (User Master Authorization Profiles) as shown in Figure 3. After you execute Transaction SQVI, you have to specify to create a join query and then insert Tables UST04 and USR10.



« *Figure 3 Quick Viewer Query Sample Creation*

The final result is shown in Figure 4. To convert a profile into a role, see Part 5, Tip 73.



« *Figure 4 Quick Viewer Query Sample Output*

Additional Resources

This appendix provides a set of useful tables and SAP OSS notes related to the tips given in this book. Instead of simply defining a list of useful tables, we have also indicated how you can reach your goal by using it.

Most Important Role Tables

Table	Description	How It Helps
AGR_1016	List of all generated profiles that are linked to a role	Provides the role's name by looking at the profile
AGR_1250	List of all authorization objects inserted into a role (without authorization value detail)	Provides all authorization objects inserted into a role without authorization objects' value detail
AGR_1251	List of all authorization objects inserted into a role (with authorization value detail)	Provides information on whether a critical authorization object is inactive in all roles except certain ones
AGR_1252	List of all organizational-level values inserted into a role	Provides the allowed organizational data domain (plant, company code, etc.) where a role can work
AGR_AGRS	List of all simple roles contained in a composite role	Provides how many simple roles are inserted in composite roles
AGR_DEFINE	List of all roles defined in the system	Provides a list of roles defined in the system
AGR_DATEU	Personal SAP GUI settings (see Part 1, Tip 1)	Tells whether all users have the technical name switched on in the SAP GUI
AGR_FAVOS	Personal Profile Generator roles favorites	Lists the favorite roles for a user in Transaction PFCG
AGR_FLAGS	Role attributes	Contains several flag attributes, including whether a role is collective and what the role master language is
AGR_NUM_2	Last number of generated profile	Tells the last number of the generated profile

Table	Description	How It Helps
AGR_TCODES	List of all transactions inserted into a role menu	Tells in which role menus Transaction MM03 is inserted
AGR_TEXTS	List of all descriptions of a role	Tells how many language descriptions a role has
AGR_USERS	List of all users assigned to a role	Tells how many users are assigned to a role

Most Important User Tables

Table	Description	How It Helps
USR01	Transaction SU01 DEFAULT tab data	Verifies the default data tab of users
USR02	User logon data: date of creation, last logon, user status (locked, unlocked), validity date (see Part 1, Tip 2)	Finds users not assigned to any user group (LOGON data tab)
USR05	Parameter ID and value for each user (see Part 1, Tip 6)	Provides list of parameter IDs, which can be restored in case of an accidental deletion
USR07	Last failed authorization check (Transaction SU53 content)	Enables you to inquire on all last failed authorization checks
ADR6	E-Mail Addresses (Business Address Services) (See Part 1, Tip 2)	Extracts the mail address
TPARA	User master data parameter ID tables (see Part 1, Tip 6)	Finds a parameter ID
SMEN_BUFFC	Users' favorites (see Part 2, Tip 19)	Views users' favorites

Most Important Tables in the SAP Menu and Profile Generator Customizing Switch

Table	Description	How It Helps
SSM_CID/SSM_CIDT	Contain all customizing switches and explanation of Profile Generator (Transaction PFCG) customization and Session Manager Customization	Discovers all customizing switches for Session Manager and Profile Generator

Table	Description	How It Helps
SSM_CUST	Session Manager customizing switch	Sets and customizes Session Manager
SSM_START	Startup logon transaction	Executes a specific transaction when you log on
USERS_SSM	Customize allowed menu at the user level	Enables users to use the SAP user menu instead of the SAP standard menu when you have disabled the SAP user menu in PRGN_CUST, but some users should still use it
PRGN_CUST	Customizing settings for authorization processes (see Part 1, Tip 5)	Enables or disables HR organizational assignment, customize profile and role transport, and customize SAP_ALL generation

Miscellaneous Useful Tables and Views

Table	Description	How It Helps
V_FDIRT	View for scanning function modules with texts	Extracts the function group AREA field when given a function module
DEVACCESS	List of all developers and developer keys	Lists all developers defined
ICFSECPASSWD	ICF: ICF Password Repository (Service)	Provides credentials and users in Internet Communication Framework (Transaction SICF)
RFCDES	Destination table for RFCs	Provides RFC destination credentials and users
D010SINF	Table REPOSRC view	Provides last change at the SAP program
D010TAB	Table for user report tables	Lists all tables used in a query
DD2526V	DD: Base tables of buffered database views	Finds a table view

Table	Description	How It Helps
TSTCP	List of all SAP parametrical transactions defined	Finds all parameters in parameter transactions
TSTC	List of all SAP transactions defined	Finds all transaction codes defined in a SAP system

OSS Notes

OSS Note Number/description	Tip
Note 1469961 - SU01-SU10 Revision to Password Dialog Box	Part 1, Tip 5
Note 1482619 - PRGN_CUST: Switches Are Missing from Value Help Note 662466 - SU01/SU10: Generated Password Contains Special Characters	Part 1, Tip 5
Note 380029 - FAQ SAP Easy Access Customizing	Part 1, Tip 9
Note 13202 - Security Aspects in ABAP Programming Note 358122 - Function Description of Transaction SE97 Note 35612 - Authorization Check for CALL TRANSACTION	Part 2, Tip 1
Note 77430 - Customizing: Current Settings	Part 2, Tip 12
Note 888889 - Automatic Checks for Security Notes Using RSECNOTE	Part 2, Tip 13
SAP Note 368496 - Check Indicators and Default Authorization Values	Part 3, Tip 42
Note 504006 - PFCG: New Functions, Corrections for Role Menu Maintenance	Part 3, Tip 45
Note 543164 - Significance of `auth-authorization_trace` Values	Part 3, Tip 46
Note 1380203 - PFCG FAQ: Naming Template for Generated Profile Names	Part 3, Tip 51
Note 841612 - Maximum Number of Profiles for Each User	Part 3, Tip 56
Note 1504689 - Custom User Group Upload in Risk Analysis and Remediation	Part 4, Tip 60
Note 991377 - Missing Entries in Table PRGN_CORR2	Part 5, Tip 75

The Authors

Andrea Cavalleri is an SAP-certified security and compliance consultant. He founded Aglea s.r.l. (*www.aglea.com*) in 2003. He has more than 12 years of experience in IT and more than 15 years of experience as a developer in C++ and Microsoft Access. Andrea has been a team leader in more than 30 SAP GRC, identity management, and security projects and has been a teacher for SAP Italy authorization and security courses since 1999.

Massimo Manara is an SAP-certified security and compliance consultant at Aglea s.r.l. (*www.aglea.com*), the only Italian company whose core business is SAP security and compliance. He has nearly 10 years of experience in IT security and a bachelor's and master's degree in security computer science. Massimo has been a team leader in several SAP security projects and has been a teacher for SAP Italy GRC courses since 2009.

Index

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

Q

R

S

T

U

V

W

Service Pages

The following sections contain notes on how you can contact us.

Praise and Criticism

We hope that you enjoyed reading this book. If it met your expectations, please do recommend it, for example, by writing a review on http://www.sap-press.com. If you think there is room for improvement, please get in touch with the editor of the book: publishing@galileo-press.com. We welcome every suggestion for improvement but, of course, also any praise!

You can also navigate to our web catalog page for this book to submit feedback or share your reading experience via Facebook, Google+, Twitter, email, or by writing a book review. Simply follow this link: http://www.sap-press.com/H3233.

Supplements

Supplements (sample code, exercise materials, lists, and so on) are provided in your online library and on the web catalog page for this book. You can directly navigate to this page using the following link: http://www.sap-press.com/H3233. Should we learn about typos that alter the meaning or content errors, we will provide a list with corrections there, too.

Technical Issues

If you experience technical issues with your e-book or e-book account at SAP PRESS, please feel free to contact our reader service: customer@sap-press.com.

About Us and Our Program

The website http://www.sap-press.com provides detailed and first-hand information on our current publishing program. Here, you can also easily order all of our books and e-books. For information on Galileo Press Inc. and for additional contact options please refer to our company website: http://www.galileo-press.com.

Legal Notes

This section contains the detailed and legally binding usage conditions for this e-book.

Copyright Note

Your Rights as a User

Digital Watermark

Trademarks

The common names, trade names, descriptions of goods, and so on used in this publication may be trademarks without special identification and subject to legal regulations as such.

Limitation of Liability

Regardless of the care that has been taken in creating texts, figures, and programs, neither the publisher nor the author, editor, or translator assume any legal responsibility or any liability for possible errors and their consequences.