iProConference:

SAP HCM Best Practise
London, 8th November 2012

HR Authorisations

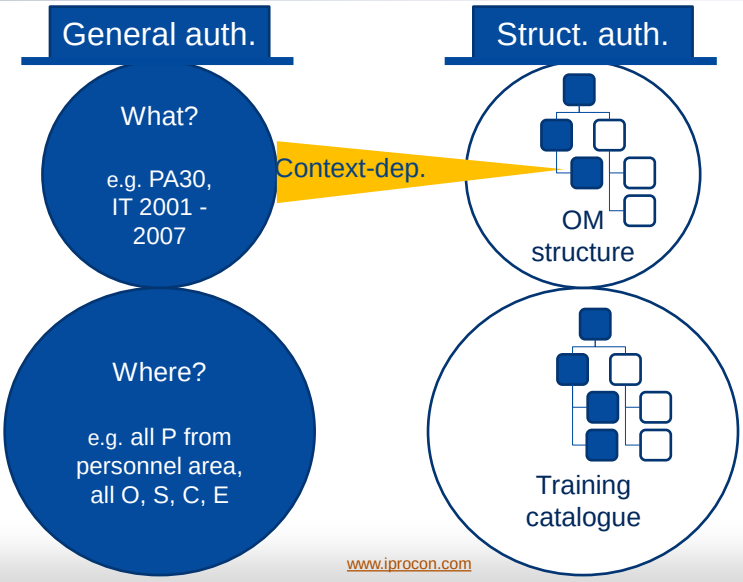
Anja Marxsen
Sven Ringling

#HCMBP2012

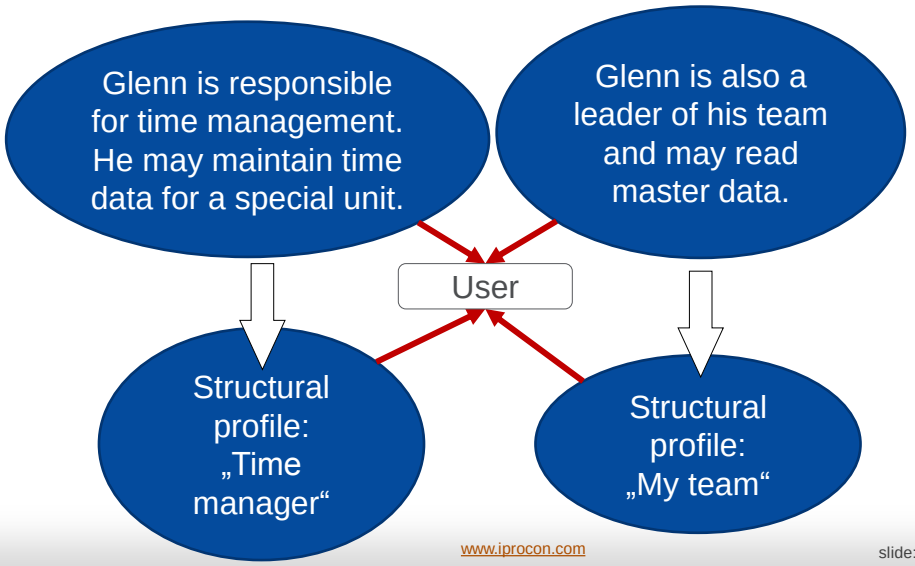
Agenda

- Overview: General / Structural / Context Authorisation Check
- How to reduce the number of roles
- Avoid these pitfalls
- How to approach a redesign

Overview



Example context-dependent auth.



Overlapping of authorisation

Maintain time data
+
Read master data

A special org unit
+
His own team

Solution: context-dependent auth.

Glenn is responsible for time mgmt. He may maintain time data for a special unit.

context

Structural profile „Time manager“

Authorization Profile

PROFL

Glenn is also a leader of his team and may read master data.

context

Structural profile „own team“

Context-dependent authorisation

2 roles and 2 profiles together lead to a **mix** of objects and authorisations

Context-dependent authorisation can assign a profile to a special role

No more mix. Everybody can only do what he is supposed to do.

e.g. P_ORGINCON

Authorization Fld	Short Descriptio...
INFTY	Infotype
SUBTY	Subtype
AUTHC	Authorization level
PERSA	Personnel Area
PERSG	Employee Group
PERSK	Employee Subgroup
VDSK1	Organizational Key
PROFL	Authorization Profile

Agenda

- Overview: General / Structural / Context Authorisation Check
- How to reduce the number of roles
- Avoid these pitfalls
- How to approach a redesign

Why do we have so many roles?

- General authorisation can restrict both: access to data and access to persons
- Functional range of SAP HCM applications increases
- Functions are becoming decentralised – more users need different access

Possible solutions

- ✓ Implement structural authorisation with dynamic start object
- ✓ Stay with general authorisation but
  - ► use object P_NNNNN
  - ► use custom object + BAdI
- ✓ Reduce maintenance effort using reference roles

N structural profiles - 1 for each location

Authoriz. profile	Auth.profile name
Z001	London
Z002	
Z003	
Z004	Birmingham
Z005	Liverpool
Z006	Bristol
Z007	Coventry
Z008	Leeds
Z009	Glasgow
Z010	Edinburgh

Responsible for your own location:

Auth.profile	No.	Plan...	Ob...	Object I	Maint.	Eval.path	Statu...	Depth	Sign	Period
Z001	1	01	0	50000100	☐	0-0-S-P	1			D

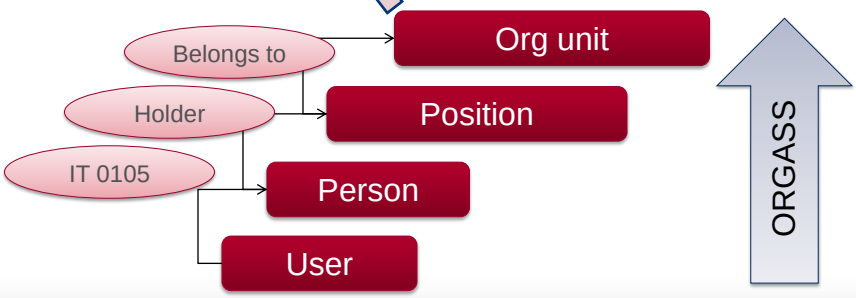
Responsible for 2 locations:

User name	Auth.profile	Start date	End date
IPROCON_AJ	Z001	01.01.2012	31.12.9999
IPROCON_AJ	Z002	01.01.2012	31.12.9999

How to create dynamic profiles

Auth.profile	N..	Pla...	Obj....	Object I	Maint.	Eval.p...	S..	D..	P	Function module
Z020	1	01	0		☐	0-0-S-P	12		D	RH_GET_ORG_ASSIGNMENT

Standard function module RH_GET_ORG_ASSIGNMENT dynamically identifies the assigned org unit.



Get more out of dynamic profiles

Many users stop at standard options

- Org unit: user is line manager of
- Org unit: user is staff member of

Real life requirements are more diverse

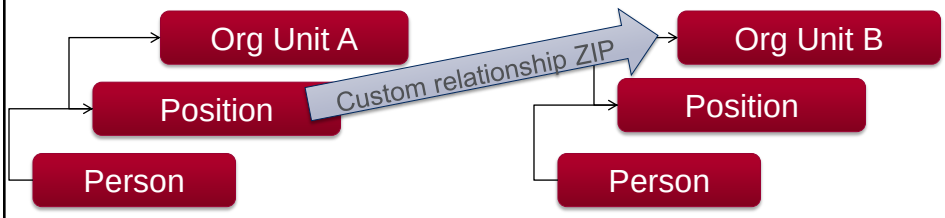
- PAs capturing data for managers or whole teams
- Managers not having access more than 2 levels down (“grandfather principle”)
- Other roles like resource planners, event managers,…

You can achieve much with little custom programming

- … and a good deal of analysis and conceptual thinking
- Nevertheless: always try to avoid complexity via pragmatic processes

Dynamic – but different start object I

1. Create custom relationship between position and other org unit:



OT	Object type text	A/B	Rel	Relationship name	RelObjType
O	Organizational unit	B	ZIP	Is responsibility of	S
S	Position	A	ZIP	Responsible for location	O

Dynamic – but different start object II

2. Copy evaluation path ORGASS and replace your own relationship:

Evaluation Path | ZORGASS | Determine organizational assignment

No.	Obj.Ty...	A/B	Relat'ship	Relationship name	Priority	Rel.obj.type
1	P	B	008	Holder	*	S
2	US	A	208	Is identical to	*	P
3	S	A	ZIP	Responsible for location	*	O

Dynamic – but different start object III

Function module | Z_RH_GET_ORG_ASSIGNMENT | Inactive

Attributes | Import | Export | Changing | Tables | Exceptions

2. Copy function module and replace your own evaluation path:

```
23    IF sy-subrc = 0.
24      lv_wegid = 'ZORGASS'.
25    ENDIF.
26
27    CALL FUNCTION 'RH_PBUILD_WORKFLOW'
28      EXPORTING
29        iv_evaluation_path = lv_wegid
30      IMPORTING
31        ev_mod_ev          = lv_wegid
32        ev_is_internal     = lv_wegint.
33
34    CALL FUNCTION 'RH_STRUC_GET'
35      EXPORTING
36        act_otype       = 'US'
37        act_objid       = uname
```

Tip for enhanced use

- If the access to persons can‘t be determined from org structure you can also develop a custom function module that may identify relevant persons by
  - user parameter
  - master data
  - customizing
  - …
- In this case evaluation path and start object remain empty.

How to reduce number of roles

- ✓ **Dynamic start object**
- **Dynamic in general auhorization through custom object or P_NNNNN or BAdI**
- **Reference role**

P_NNNNN

You need access to all persons of your own cost center.

Standard authorisations don‘t provide cost center.

Using the organisational key leads to 1 role for each user.

You may also use

P_NNNNN with additional coding.

Field Name	Short Description
INFTY	Infotype
SUBTY	Subtype
AUTHC	Authorization level

+ all fields of infotype1 (incl. custom fields)

1 role for all users

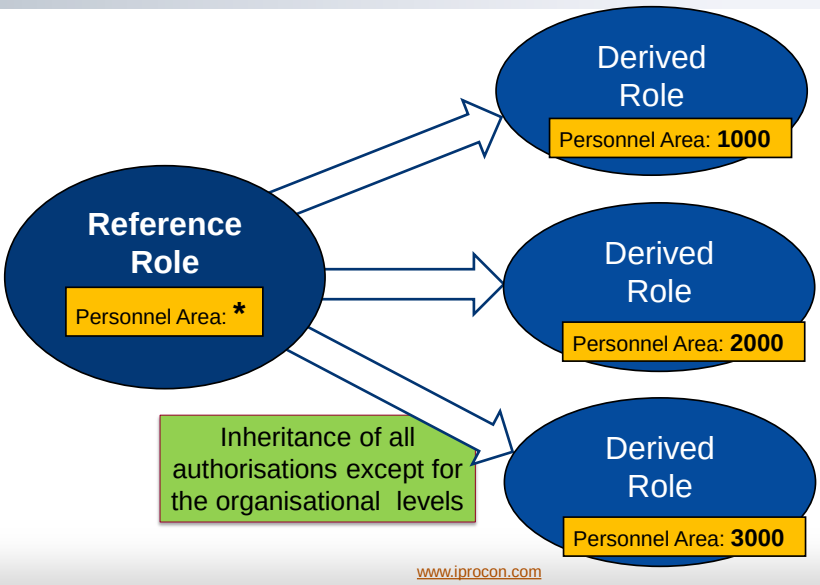
Additional coding for P_NNNNN

- The report RPUACG00 generates coding in program MPPAUTZZ
- Here you can add your own coding.
- Note! After every regeneration the custom code gets lost.

How to decrease amount of roles

- ✓ Dynamic start object
- ✓ Dynamic in general auhorization through custom object or P_NNNNN or BAdI
- ■ Reference role

Concept of reference roles



Agenda

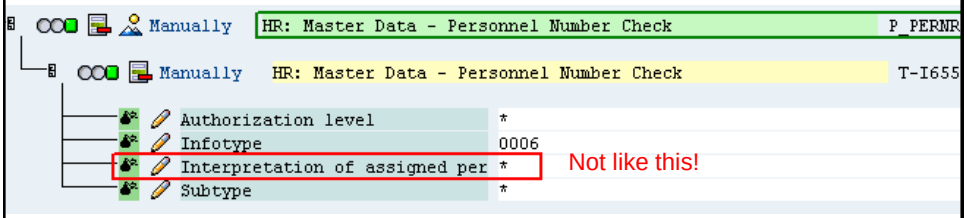
- Overview: General / Structural / Context Authorisation Check
- How to reduce the number of roles
- Avoid these pitfalls
- How to approach a redesign

Avoid these pitfalls

- P_PERNR
- P_ABAP
- time dependent check - T582a
- Adding rights from different roles, particularly backend and XSS
- BAdI: all methods!

P_PERNR

Possible values:
E = exclude own personnel number
I = include own personnel number



Rule:
Basis is always ‚normal' authorisation – P_PERNR is checked last
E: less rights for own personnel number (e.g. Change IT0008)
I: more rights for own personnel number (e.g. ESS)

Authorisation Object P_ABAP

Often difficult to provide access to non-critical reports (e. g. phone list)

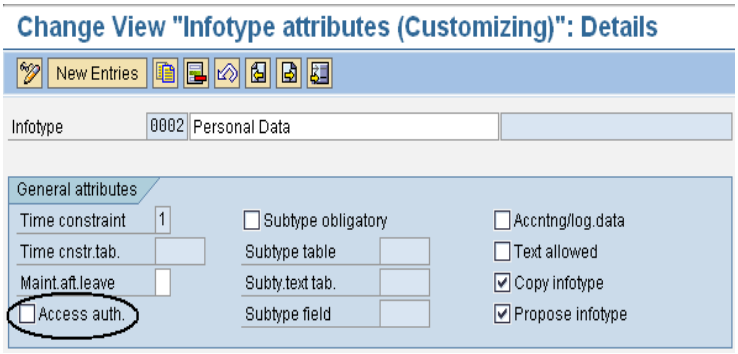
Authorization Fld	Short Descriptio...
REPID	ABAP Program Name
COARS	Degree of simplification for authorization check

P_ABAP **deactivates HR authorisation check** (COARS = 2) but doesn't replace the basic authorisation to start a report!

Recommendation: 1 role with non-critical reports for all users

Time dependent check

The date-dependent check **is not carried out for each infotype** by default. You can change the setting in table view V_T582A.



Rights from different roles adding up

- **It is a common misconception that authorisations are only used together, when in the same role**
  - E.g.: if one role allows to read infotype 0002 and a different role holds rights for transaction PA20, then the user cannot access infotype 0002 in PA20 → WRONG!
  - When a user wants to perform any action, authorisations from all roles assigned are applied
- **Example: HR team leader**
  - Role „HR Manager UK“ gives access to transaction PA30 and HR infotypes only for personnel areas in the UK
  - Role „Manager for MSS“ gives access to all HR infotypes without restrictions (assumption: MSS assigns right people only)
  - Problem: combining both roles gives access to all HR data globally

BAdI for general auth. checks

Definition Name	HRPAD00AUTH_CHECK
Definition Short Text	HR: Authorization Check

You must **consider all these methods** during implementation to ensure that the standard authorization check continues to work! Otherwise, you deactivate the complete authorization check.

Method	Description
CHECK_MAX_INFTY_AUTHORIZ...	Maximum Check for Infotype Authorization
CHECK_MAX_LEVEL_AUTHORIZ...	Maximum Check for Authorization Level
CHECK_MAX_SUBTY_AUTHORIZ...	Maximum Check for Subtype Authorization
CHECK_MIN_INFTY_AUTHORIZ...	Minimum Check for Infotype Authorization
CHECK_MIN_LEVEL_AUTHORIZ...	Minimum Check for Authorization Level
CHECK_MIN_SUBTY_AUTHORIZ...	Minimum Check for Subtype Authorization
SET_ORG_ASSIGNMENT	Set Organizational Assignment

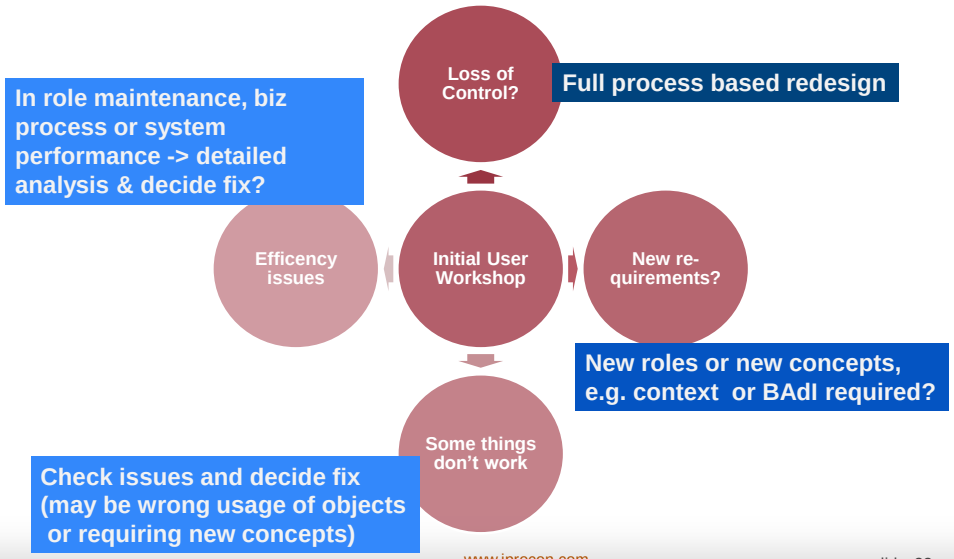
Agenda

- Overview: General / Structural / Context Authorisation Check
- How to reduce the number of roles
- Avoid these pitfalls
- How to approach a redesign

Redesigning HR Authorisations

- General approach / test
- Performance improvement of structural authorisation
- Composite roles
- OM assignment?
- Assign structural profiles via BAdI
- Further improvements through BAdIs?
- Performance improvement through object P_ABAP
- Migrating to context-dependent authorisation
- Amend profile generator for better defaults in transaction PFCG

General approach



Tips for Test and Cut-Over

4 elements of authorisation test

- Can users do, what they need to? → key users test their own process
- Can users do more than they should? → Key users and tech experts test others' process
- Performance → tech experts perform mass test together with key users
- User maintenance process → end to end acceptance test with user admin and business users

Cut-Over

- Keep old roles as a contingency and allow them to be assigned for a limited period of time in case of issues
- Do not tell key users before test is completed

Improve performance for struct. auth.

Evaluation Path	SBESX	Staff assignments along organizational structure

No.	Obj.Ty...	A/B	Relat'ship	Relationship name	Priority	Rel.obj.type
10	O	B	003	Incorporates	*	S
20	O	B	002	Is line supervisor of	*	O
30	S	A	008	Holder	*	*
35	S	B	088	Dotted Line Supervises	*	S

better: P

Evaluation path with nonspecified target object reduces performance

Change View "User Table for Batch Input"

New Entries

User name	Days	Gen.day
IPROCON_AJ	5	01.01.2012

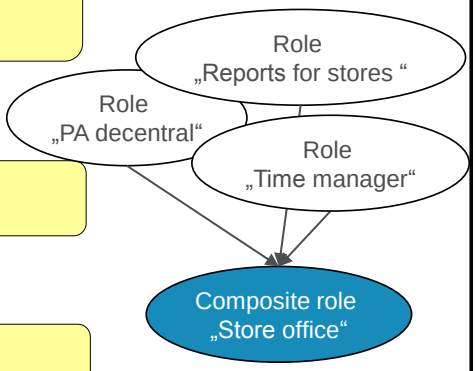
Save user data in SAP memory

Use composite roles

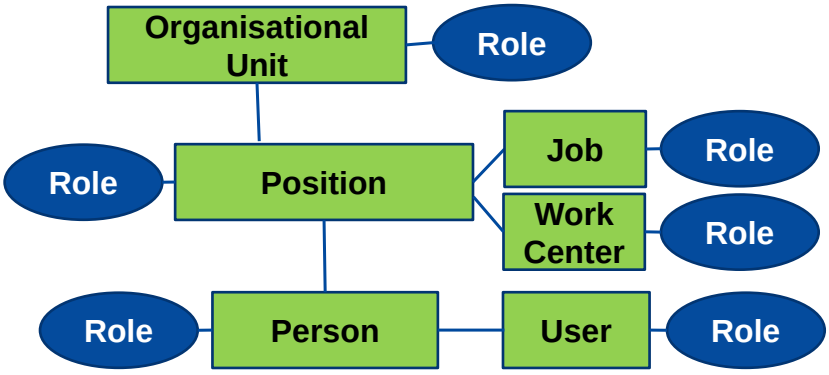
The more roles you have
the more maintenance effort
you need for user assignment

Combine single roles into
composite roles

Modular design of role concept
reduces maintenance effort



Assign roles via org management



Assign structural profiles via BAdI

Maintenance of table T77UA takes too much effort
or doesn't fulfill the requirements

User name	Auth.profile	Start date	End date	Exclusion	Display Objects
CHICAGO	CHICAGO	01.01.1900	31.12.9999	☐	
COMMCLERK_A	COMMCLERK_A	01.01.1900	31.12.9999	☐	
IPROCON_TS	ALL	01.01.2012	31.12.9999	☐	
SAP*	ALL	01.01.1900	31.12.9999	☐	
SMITH	MANAGER	07.01.2000	31.12.9999	☐	

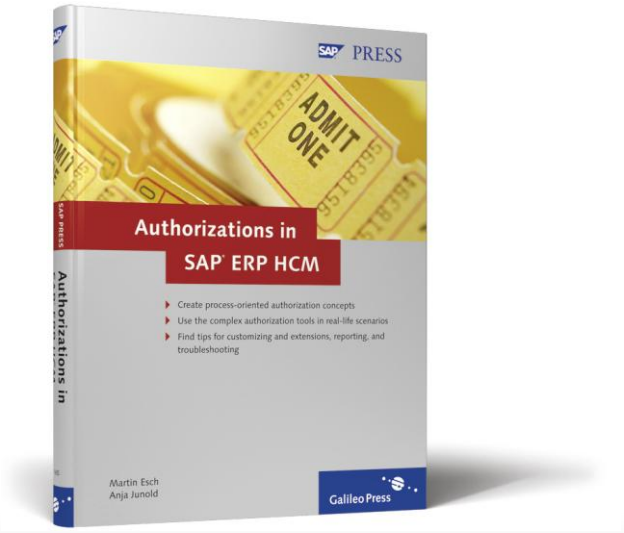
Assignment of structural profiles either from the field
PROFL or following your own logic
via **BAdI HRBAS00_ GET_PROFL**

No need of maintaining table T77UA.
Dynamic assignment of structural profiles.

Further improvements through BAdIs

- **The BAdIs available are very powerful**
  - You may find ways to improve performance or usability by making good use of them
  - Risk: users / data security team learn that "everything is possible somehow" → you end up reinventing the system
- **Examples**
  - Allow access to some infotypes only in specific transactions, e.g. access to IT0002 fields for reporting, but not in transactions, where NI number is shown
  - Capture additional payments up to certain limit
  - Rights to change HR data for most users "switched on/off", if central team wants to avoid changes at certain times

Book recommendation



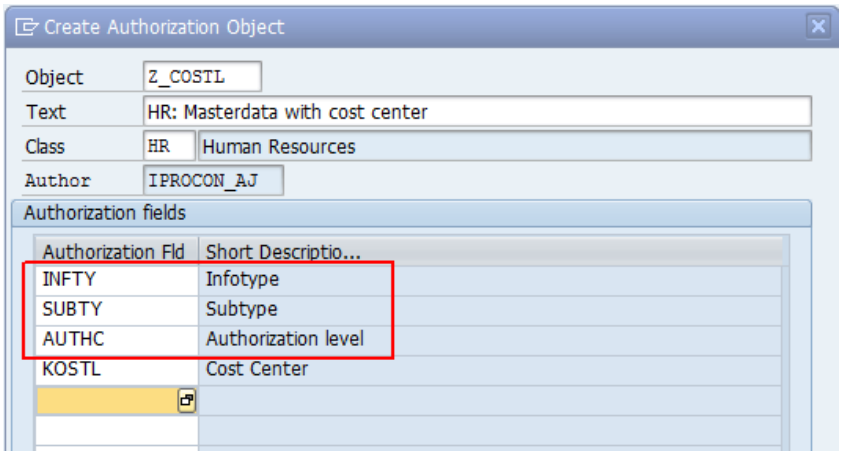
Appendix

- P_NNNNN
- Reference role

Step by Step

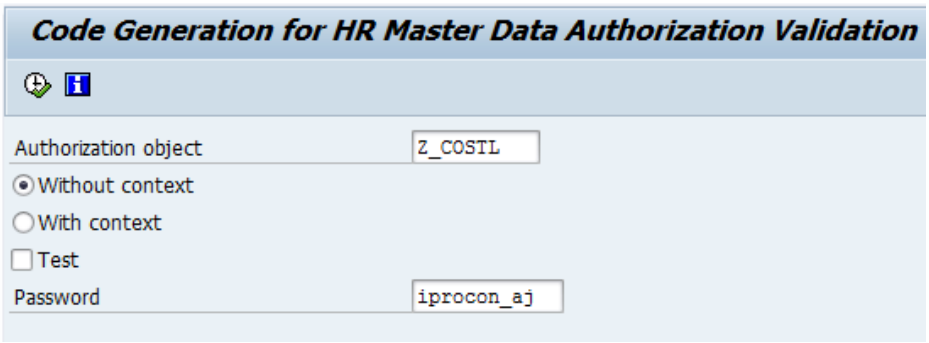
1. Create P_NNNNN
2. Take over P_NNNNN in standard authorisation check
3. Activate P_NNNNN

Create P_NNNNN



Take over in standard auth. check

Report RPUACG00

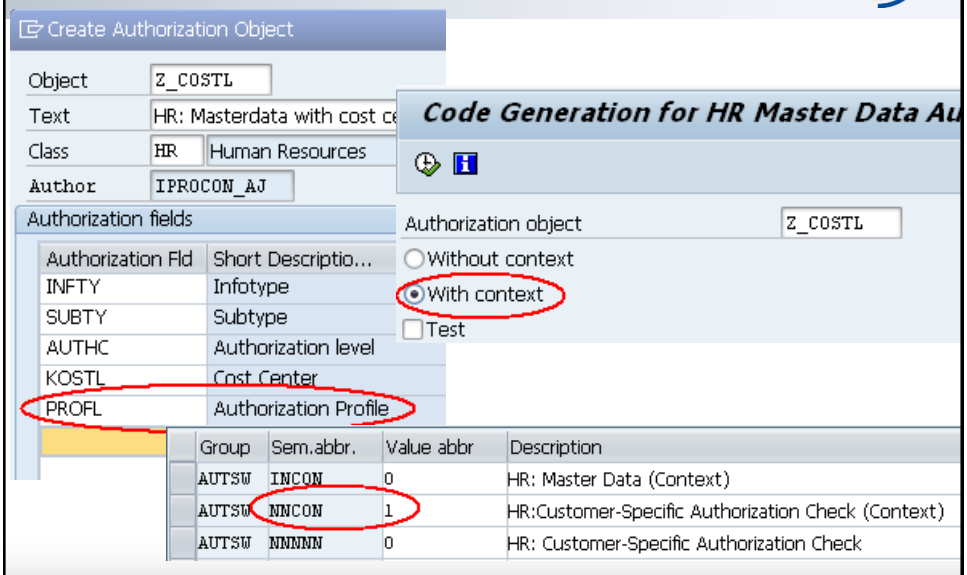


Activate P_NNNNN

Table T77S0

Group	Sem.abbr.	Value abbr	Description
AUTSW	INCON	0	HR: Master Data (Context)
AUTSW	NNCON	0	HR:Customer-Specific Authorization Check (Context)
AUTSW	NNNNN	1	HR: Customer-Specific Authorization Check
AUTSW	ORGIN	1	HR: Master Data
AUTSW	ORGPD	1	HR: Structural Authorization Check

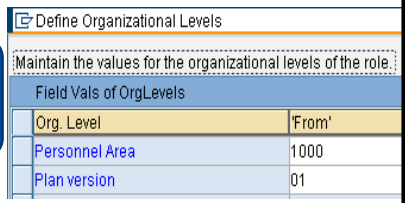
P_NNNNN with context



Reference role

You might have roles for decentralised use that only differ in one or few org level fields (e.g. personnel area).
In the standard, the plan version is the only org level

You can change existing fields to org levels via Report PFCG_ORGFIELD_CREATE

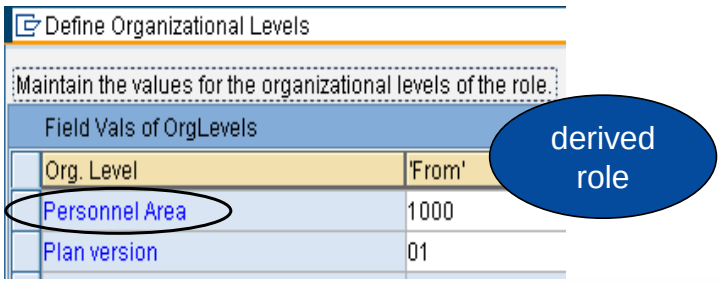


Only the reference role needs to be maintained.

1. Create an org level field

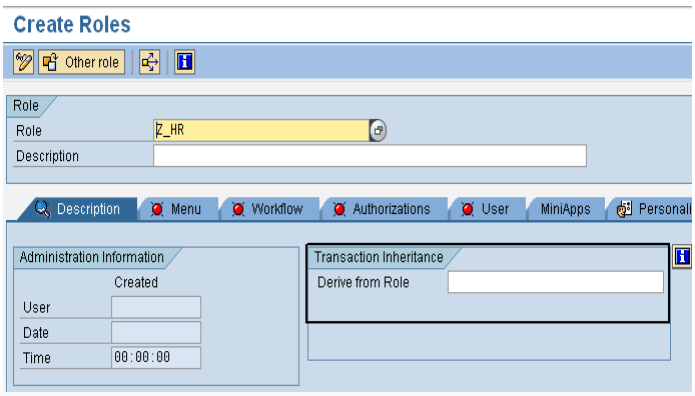
Use report PFCG_ORGFIELD_CREATE to create a new org level because the standard provides only the plan version as an org level.

Result:



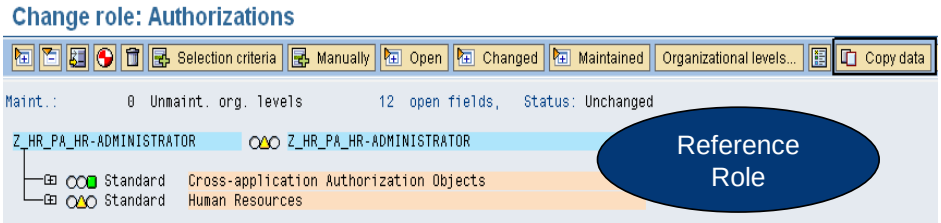
2. Derive role from reference role

A role becomes a reference role as soon as another role has been derived from that role.



3. Maintain the reference role

Transfer the authorisations of the reference role to the derived roles via button „Copy data“ – except for the organisational levels.



Ask for our in-house workshops

SAP HR Authorisations design

- **Full (re)design**
- **Additional modules / processes**

Reviewing your HR Authorisations system

Preparing for a rollout

- **Incl. international rollout**

Switching to structural authorisations

- **…or context sensitive authorisations**

s.ringling@iprocon.com