A Holistic View on Access Right Management

TCCG Meeting #9

Frankfurt am Main, 06.02.2019

Summary

1. General principles
2. Access rights configuration process
3. Access to OTS-based components
4. Access of U2A-only users
5. T2 and TIPS specificities

1 – General principles

General principles

- With the T2-T2S Consolidation project, T2S SDMG will evolve into the Common Reference Data Management (CRDM) component.
- A first portion of CRDM functionalities (CRDM$^{\text{TIPS}}$) was delivered already, in order to support the reference data setup and maintenance for TIPS.
- CRDM is therefore the *to-be* application for common reference data management, built using T2S SDMG as *as-is* application, which also means all common reference data are stored in the same (logical and physical) database.
- Consequently, the Access Rights Management (ARM) model for CRDM follows the one that is already in place today in T2S.

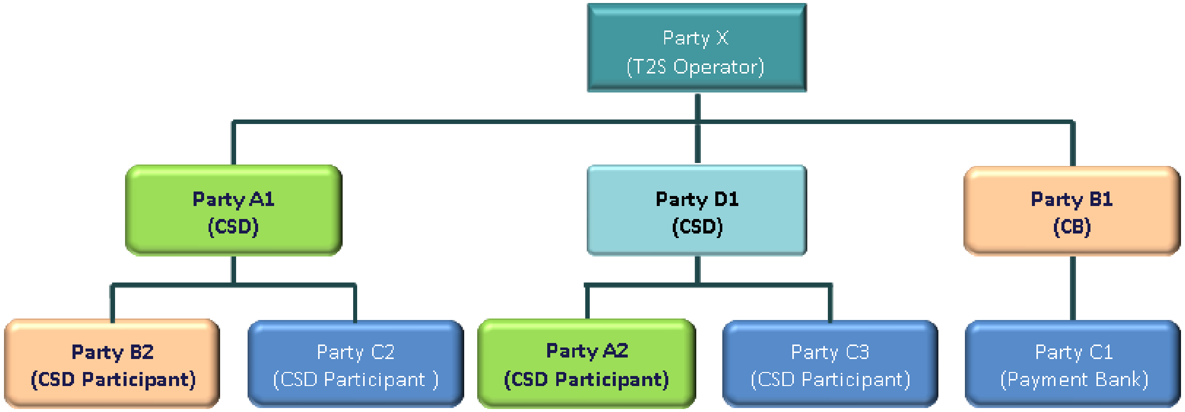
General principles

1. The party model is based on a hierarchical structure.
2. Each piece of information belongs to one system entity.
3. Functions are granted following the hierarchical party model.
4. Data are granted by the owner system entity.
5. Access rights are based on a RBAC* model.
6. Access rights management is decentralized.
7. Access rights granularity is function-based and object-based.
8. The data scope is determined by the hierarchical party model.
9. The data scope can be altered (extended and/or reduced) in some cases.

*Role-Based Access Control

General principles

1. The party model is based on a hierarchical structure.
   - Relationships between parties determine a hierarchical party model based on a three-level structure.
   - The Operator is the only party on the top level of the hierarchy and it is in a relationship with each party of the second level, i.e. each CSD (in T2S only) and each CB.
   - Relationships also exist between each party belonging to the second level of the hierarchy (i.e. a CSD or a CB) and all its participants (i.e. CSD participants for the CSDs and payment banks for the CBs).



*The same colour corresponds to the same legal entity.

General principles

2. Each piece of information belongs to one system entity.
   - System entity management in CRDM defines all functionality needed to support a participating CSD's or CB's segregation of processing capabilities and data across its participants.
   - A system entity defines the entity by which CRDM must segregate the data and access rights of the CSDs, CBs and the Operator.

General principles

3. Functions are granted following the hierarchical party model.
   - This principle establishes the requirement that each parent party is responsible for assigning the set of functions within each service or common component that each of its participants will be allowed to use.
   - On this basis, the Operator assigns functions to CSDs and CBs, each CSD assigns functions to each of its CSD participants, each CB assigns functions to each of its payment banks.
   - Different parties within the same system entity may be assigned different functions of different services or common components on the basis of specific contractual agreements.

General principles

4. Data are granted by the owner system entity.
    - Most of the functions of the different services and common components apply to data objects.
    - In this case, an access right refers to the privilege to use a given functional capability of a service or common component (function scope) on a given object (data scope).
    - In a typical scenario, one system entity is granting both the function scope and the data scope, according to the hierarchical party model (business requirement 9 later on describes exceptions for this typical scenario).

General principles

5. Access rights are based on a RBAC model.
   - Access rights configuration is based on the following concepts: privileges, roles, system users, objects and group of objects.
   - Privileges are segregated per service and component. This means that each privilege allows triggering a specific function of one service (e.g. T2S, TIPS), service component (e.g. RTGS, CLM) or common component (e.g. CRDM).
   - Privileges can be granted as system privileges (i.e. without narrowing the scope to a single or a homogeneous group of certain reference data objects) or as object privileges (i.e. in relation to a single reference data object or a group of reference data objects).
   - Roles and privileges can be granted as well at object group level (secured group). Each secured group includes objects of the same type. The same object may belong to more than one secured group.
   - Privileges related to the same service or component can be grouped into roles.
   - The access rights profile of a given system user is determined by the set of roles and privileges granted to it.

General principles

6. Access rights management is decentralized.
   - The administrator of the Operator (a) creates roles including the available privileges, (b) manages users of the Operator, (c) assigns roles and privileges to these users, (d) creates the administrators of CSDs and CBs, (e) assigns the relevant roles and privileges to CSDs and CBs.
   - The administrator of a CSD/CB (a) creates new roles including the available roles and privileges, (b) manages users of the same CSD/CB, (c) assigns the available roles and privileges to these users, (d) creates the administrators of the CSD participants/payment banks of the same CSD/CB, (e) assigns the available roles and privileges to participants/payment banks of the same CSD/CB.
   - The administrator of a CSD participant/payment bank (a) creates new roles including the available roles and privileges, (b) manages users of the same CSD participant/payment bank, (c) assigns the available roles and privileges to these users.

General principles

7. Access rights granularity is function-based and object-based.
   - A privilege defines a specific functional capability of a given service or common component, e.g. the settlement of an instruction, a query, a report, a reference data update.
   - Privileges can be granted with specific reference to a given object (e.g. a TIPS account) or group of objects (a group of T2S dedicated cash accounts).

General principles

8. The data scope is determined by the hierarchical structure.
   - Each system user is linked to a party.
   - This link determines the set of objects falling in the data scope of the system user, regardless the service which the different objects refer to.
   - CRDM uses this information to restrict the system user's access to the relevant data.

General principles

9. The data scope can be altered (extended and/or reduced) in some cases.
   - The default data scope of a given system user, determined by its link to the relevant party is not enough to cover some business scenarios. For example:
     - CSD participants may need to grant other parties with the privilege to instruct all their securities accounts or a subset of them.
     - System administrators may need to grant their users with access (display, instruct, etc.) on a subset of securities accounts or dedicated cash accounts only.
     - CSDs and CSD participants may grant CBs and payment banks with access (display) on all their securities accounts or a subset of them.
     - CBs and payment banks may grant CSD participants with access (display) on all their dedicated cash accounts or a subset of them.
     - The Operator may grant its users with privileges related to business functions on a subset of CSDs and CBs only.
   - In all these cases, the default data scope of a given system user for a given function can be extended and/or reduced by means of object privileges.

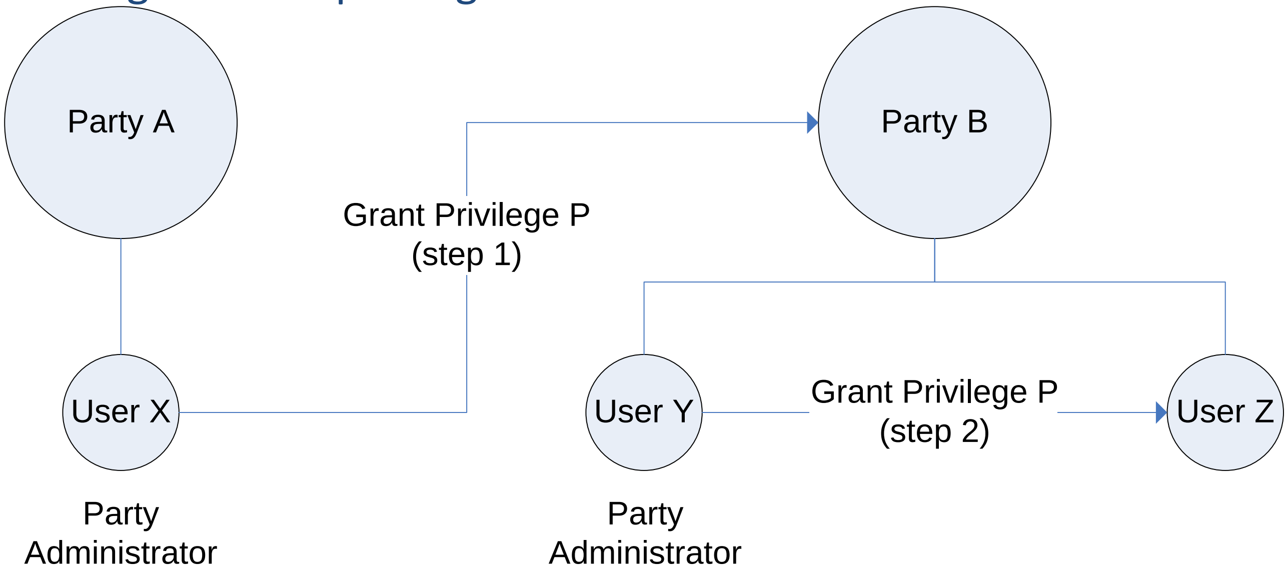
2 – Access rights configuration process

Users and parties

- Links between users and parties
  - Each new user is linked to the same party which the creator user belongs to.
  - An exception takes place when creating the first user of a party (i.e. when the Operator, a CSD or a CB create a new system administrator for one of its participants). In all these cases the created user is linked to the party this user is going to administer.
  - Through the link with the relevant party, each user inherits a default data scope.
  - The link between a user and a party can not be changed, i.e. a user is always linked to the same party.
- Party administrators
  - Each party must have at least one party administrator, i.e. a user being granted a specific system privilege that allows its grantee to grant any roles and privileges previously granted to the grantee’s party.

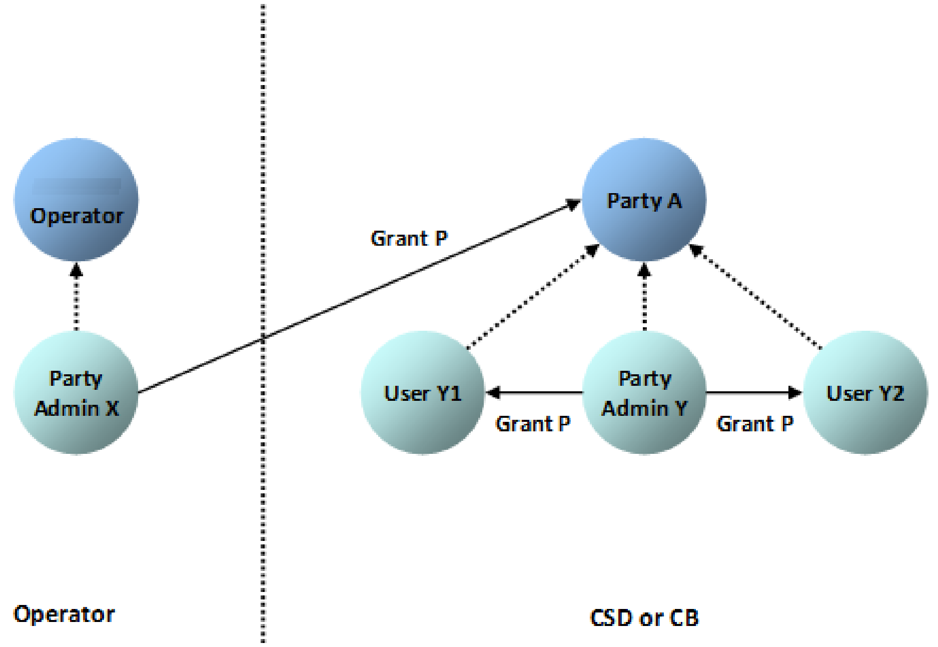
Privileges and users

- Each privilege is originally available to the party administrator(s) of the Operator only. This means that party administrators of all the other parties can not grant this privilege to their users.
- A privilege becomes available to a party administrator of a party different from the Operator only after this privilege has been granted to this party. From this moment on, the party administrator can grant this privilege.
- This implies that a two-step process is required in order to grant a specific privilege to a user belonging to a party different from the Operator. In the first step, the privilege is granted to the relevant party (so that it becomes available to the party administrator(s) of this party). With the second step, one of the party administrators grants the privilege to the relevant user.



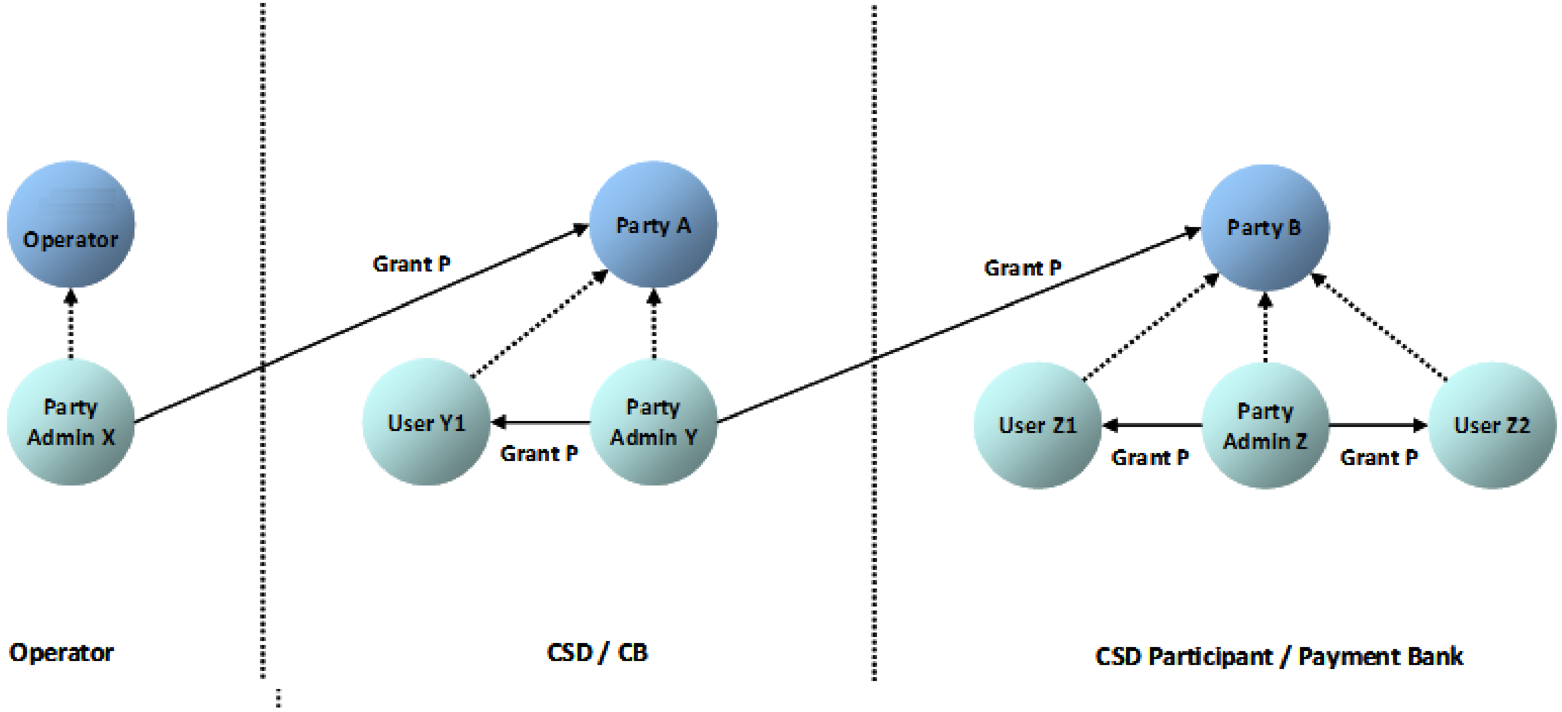
Configuration process

- Before the party administrator of a given party can grant a privilege to a user of the same party, the same privilege has to be granted to the same party, so that it becomes available to the party administrator(s) of the party.
- On this basis, two steps are needed for granting a given privilege P to the users of a CSD or of a CB:



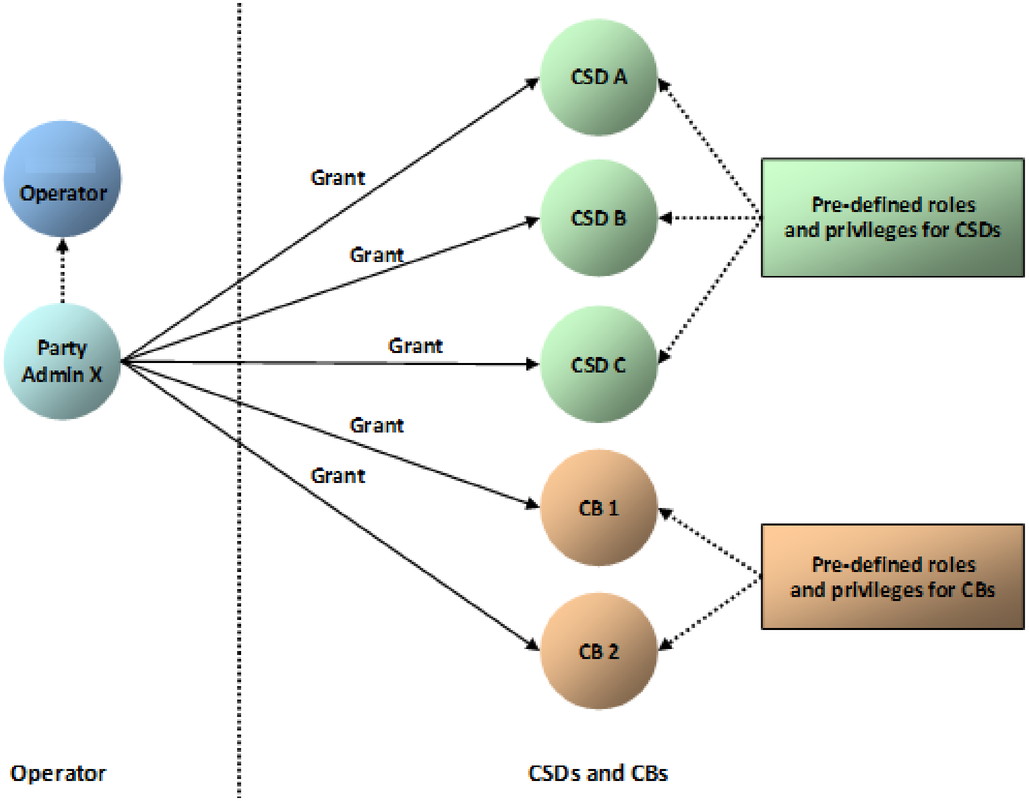
Configuration process

- A two-step process also applies when a CSD or a CB needs to configure access rights for their CSD participants or for their payment banks, respectively:



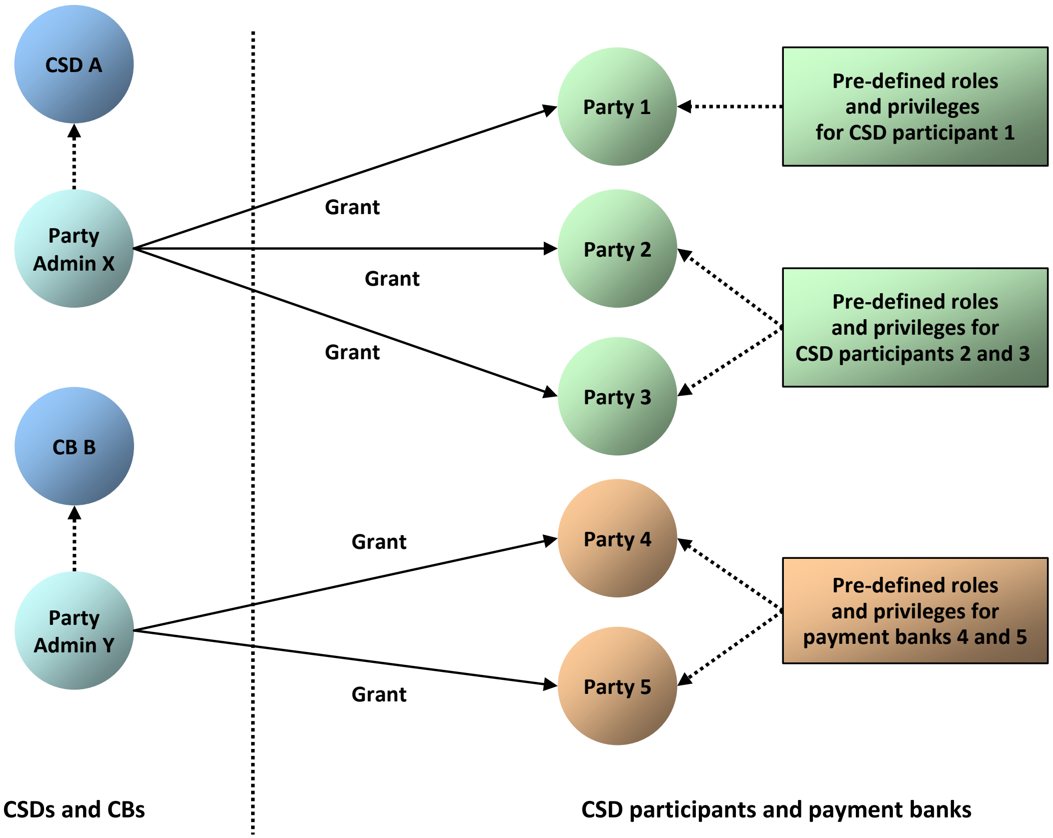
Configuration for parties

- This task consists in the assignment of the relevant set of roles and privileges to a given party in CRDM.
- A party administrator of the Operator performs this task for the configuration of access rights of CSDs and CBs.



Configuration for parties

- A party administrator of each CSD assigns the relevant set of roles and privileges to all its CSD participants, whereas a party administrator of each CB assigns the relevant set of roles and privileges to all its payment banks.


CSD A
Party Admin X
CB B
Party Admin Y
Grant
Grant
Grant
Grant
Grant
Party 1
Party 2
Party 3
Party 4
Party 5
Pre-defined roles and privileges for CSD participant 1
Pre-defined roles and privileges for CSD participants 2 and 3
Pre-defined roles and privileges for payment banks 4 and 5
CSDs and CBs
CSD participants and payment banks

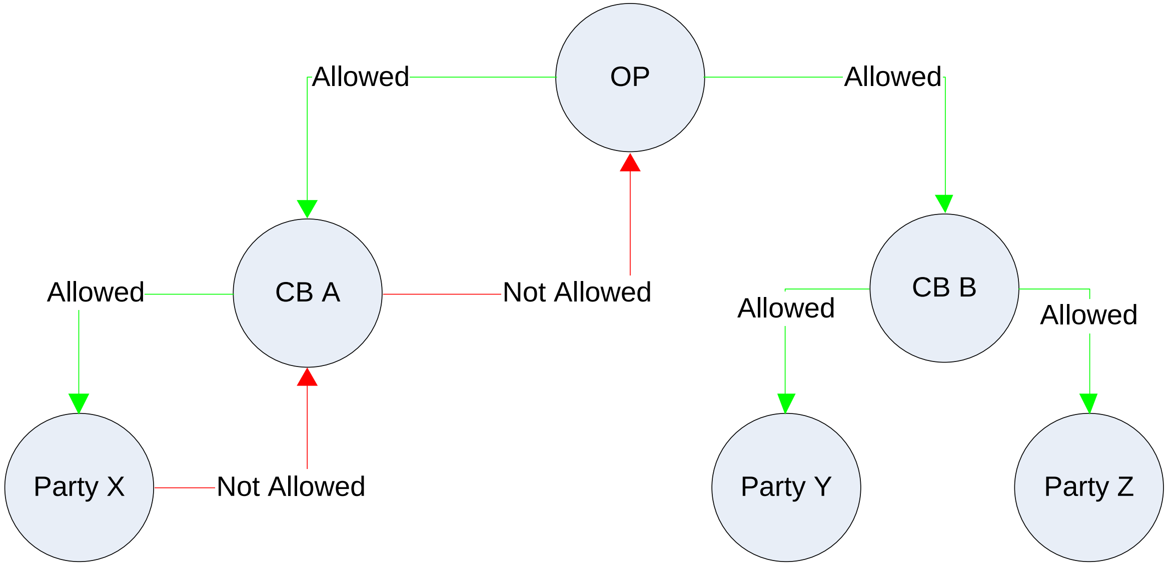
Configuration for parties

- A party administrator can propagate a privilege to other parties only if the same privilege was previously granted to its party with Admin option (i.e. Admin="True").
- When a privilege is granted in 4-Eyes mode, the grantee cannot use or propagate the same privilege in 2-Eyes, whereas the opposite is possible (i.e. restricting the usage of privilege from 2-Eyes to 4-Eyes).
- On this basis:
  - The Operator can specify what privileges each CSD/CB is allowed to propagate to other parties.
  - Each CSD/CB can specify what privileges their respective participants can propagate to other parties.
- System privileges can only be propagated top-down, i.e. following the structure of the hierarchical party model. Consequently:
  - The Operator grants system privileges to CSDs and CBs.
  - Each CSD grants system privileges to its CSD participants.
  - Each CB grants system privileges to its payment banks.
  - CSD participants and payment banks can not grant system privileges.

Configuration for parties

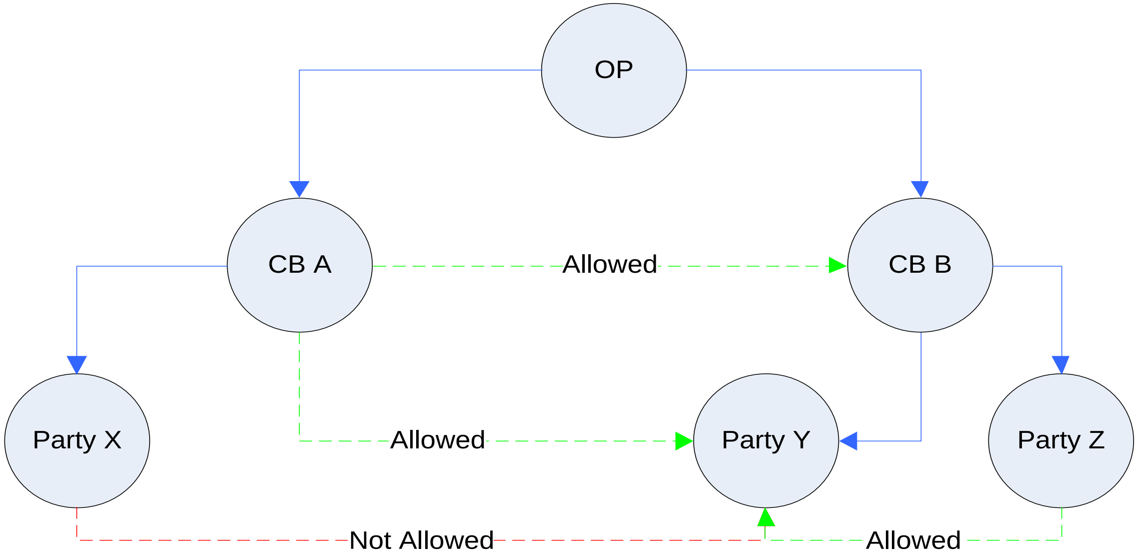
- Object privileges can be propagated both top-down (i.e. following the structure of the hierarchical party model) and transversally (i.e. cross-system entity).
- Top-down propagation of object privileges is used to set up reduced data scope scenarios. It follows the same rules of top-down propagation of system privileges.
- For transversal propagation:
  - Each CSD/CB can grant object privileges to other CSDs/CBs.
  - Each CSD/CB can grant object privileges to CSD participants belonging to other CSDs or to payment banks belonging to other CBs.
  - CSD participants and payment bank can grant object privileges only within the same system entity.
  - In all cases, the grantee party must be granted in advance with the same privileges, at system level, by its parent party.

Configuration for parties



Top-down propagation

Configuration for parties



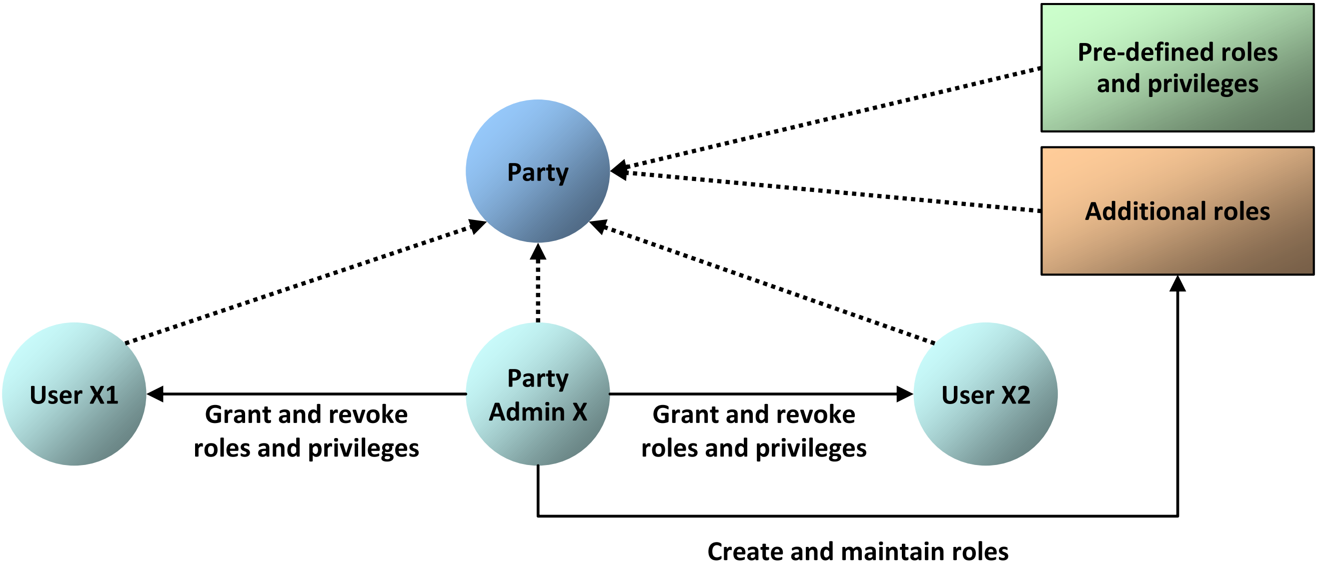
Transversal propagation

Configuration for parties

- On the basis of the propagation rules described so far, each CSD/CB can
  - use the functions granted by the Operator;
  - allow its participants to use (a sub-set of) these functions;
  - allow parties within other system entities to use (a sub-set of) these functions on (a sub-set of) its data, provided that those parties have been already authorised to use the same functions by their parent party;
  - allow its participants to use (a sub-set of) these functions on data belonging to other system entities, provided that these participants have been already authorised to use the same functions on those data by the relevant CSD/CB.
- Each CSD participant/payment bank can
  - use the functions granted by its CSD/CB;
  - allow parties within the same system entity to use (a sub-set of) these functions on (a sub-set of) its data, provided that these parties have been already authorised to use the same functions by their CSD/CB.

Configuration for parties

- After the configuration of access rights at party level has been set up for a given party, its party administrator(s) can perform the configuration of access rights at user level, in order to assign the appropriate roles and privileges to all the users of the given party.


Pre-defined roles and privileges
Party
Additional roles
User X1
Party Admin X
User X2
Grant and revoke roles and privileges
Grant and revoke roles and privileges
Create and maintain roles

3 – Access to OTS-based components

Access to OTS-based Components

- All services rely on a number of application components that are based on commercial Off-The-Shelf (OTS) software. For example:
  - TARGET2 CRSS and T2S LTSI are based on *Business Objects*
  - T2S and TIPS Trouble Management System is based on *BMC Remedy*
  - CSLD Data Warehouse (which will replace CRSS and LTSI) will be based on *Cognos*
- Each OTS software implements its own access rights management features, which are generally different (e.g. in terms of structure, functionality, granularity, etc.) from the ARM model implemented at CRDM level.

Access to OTS-based Components

- The ARM model implemented in CRDM is integrated with the one provided by the different OTS components as follows:
  - For each OTS component one (or a few) privilege(s) are defined in CRDM. These privilege(s) just allow the grantee user (defined in CRDM) to access the OTS component, without authorising the usage of any specific function of the component itself.
  - Different pieces of information retrieved from CRDM during the authorisation process (e.g. the relevant system user, the above mentioned privilege(s), the party type the system user belongs to) are provided to the relevant OTS component as login credentials.
  - The relevant OTS component performs the authentication and authorisation process based on the provided credentials, which eventually allow the usage of one or many specific functions of the OTS component.
- It is currently under analysis if and which detailed list of DWH privileges will be stored in CRDM.

4 – Access of U2A-only users

Access of U2A-only users

- The ARM model of CRDM does not distinguish between U2A and A2A privileges. In other words, if a user is granted with a specific privilege, they may gain access to the related user function in both U2A and A2A mode (if available).
- However, the A2A/U2A access can be regulated at Network Service Provider level, as the two access modes are normally based on different network services and, therefore, different Closed Groups of Users .
- This implies that a U2A-only user can be set up simply by assigning its certificate to a CGU related to a U2A network service. Such a set up would prevent the user making use of any A2A-related network service and, therefore, triggering any A2A functionality of any given service or common component.

Access of U2A-only users

- The same approach can also be used to setup users that may only work in A2A mode.
- Finally, users that must be authorized to work both in A2A and U2A mode shall have their certificates assigned to CGUs related to the different A2A and U2A network services, which would allow them triggering the granted functions both in A2A and in U2A mode.

5 – T2 and TIPS specificities

T2 and TIPS specificities

T2 (RTGS/CLM) and TIPS are based on the ARM model described so far, with two limitations only, related to principles 5 and 9:

5. Access rights are based on a RBAC model.
   - The limitation introduced for T2 and TIPS consists in the fact privileges can only be granted to roles (and not also directly to users). This implies the Operator assigns all T2 and TIPS privileges to CBs through pre-defined roles and the CBs then create their own roles and propagate them to their parties.
9. The data scope can be altered (extended and/or reduced) in some cases.
   - No object privileges are for TIPS and the current working assumption is not to have object privileges in T2 as well. This implies all privileges work on the basis of the default data scope defined by the grantee's party.