Magda Access Control Sample Story

Date	Editor	Comment
07/03/2019	Jacky	Initial Draft
11/03/2019	Jacky	- Removed \`username\` & \`password\` from User \`resource\` - Added more details of data Catalogue
21/03/2019	Jacky	Renamed resource \`Data Catalogue\` to \`Dataset\` to avoid confusion
22/03/2019	Jacky	- \`Pre-Authorised Permission\` to Section 2.3 Resource - \`Pre-Authorised Constraint\` to Section 2.5 Permission - Added more explanation for Ad-hoc permissions & roles - Updated section 2.3.1.2 Dataset - Update operation list - Add new sections on Draft property & Approval Process / Workflow
25/03/2019	Jacky	Updated: - Built-in Permission Definition - Built-in Roles - Sample Client Org Structure Setup
29/03/2019	Jacky	Fixed Mismatched IDs

1. Overview

This is a sample access control story for Magda. It’s intended to be a demonstration of describing compulsory conceptual elements required for configuring any typical client organisation structure & access control, rather than a guideline. This document will attempt to cover:

- Definition of the compulsory conceptual elements that will be available as primitive building blocks for configuring any supported client org structure levels & access level requirements

- An example story of how to configure a sample client organisation structure using predefined built-in blocks / options.

- Discussion of any possible limitation of this story. i.e. the possible type of org structures / access controls it can support

This sample story will not cover all aspects of access controls and will focus only one area i.e. datasets / resource access control. This sample story will stay at conceptual level and will not attempt to cover any technical implementation details.

2. Definitions

This section defines compulsory conceptual elements required to for configuring any typical client organisation structure & access control.

2.1 Org Units

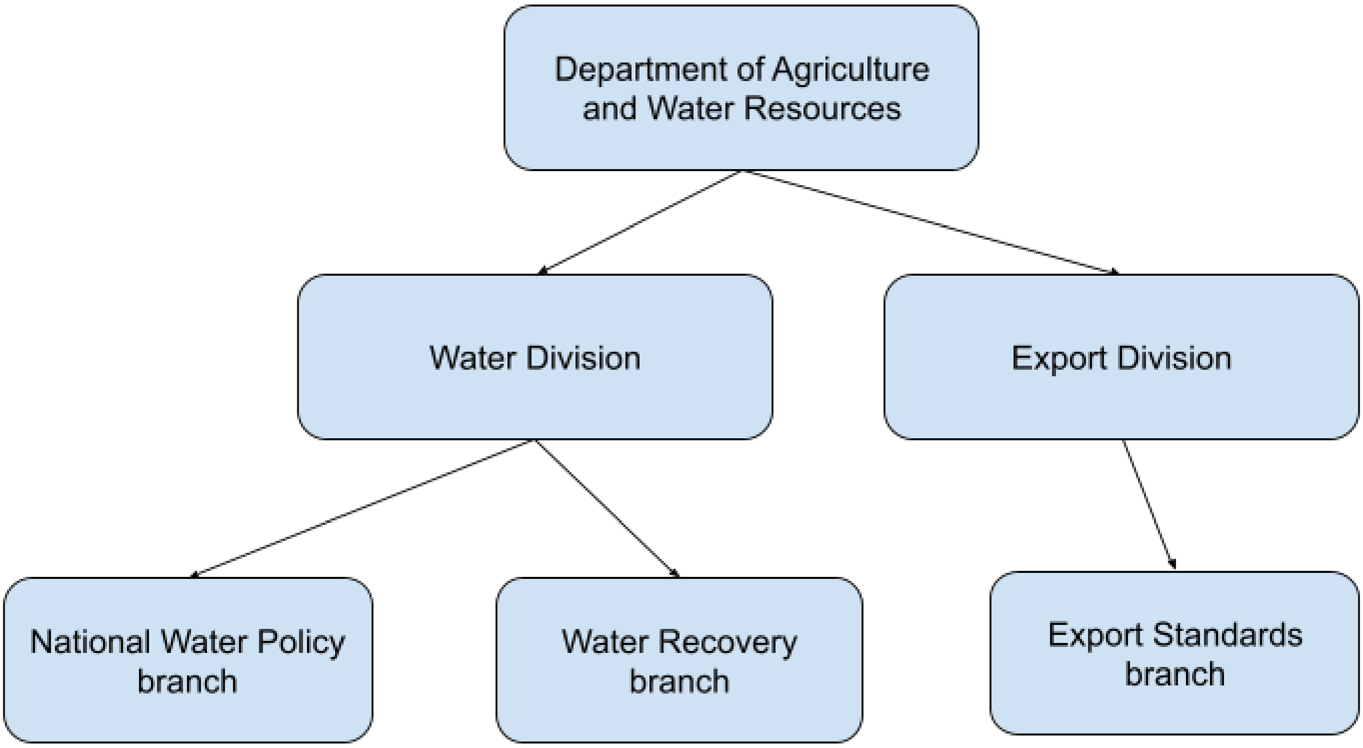
A organization unit presents the minimum building block of any supported organisation hericary / structure.

A org unit supports the followings properties:

- Id: the ID of the org unit
- Name: the name of the org unit.
    - E.g. Department of Agriculture and Water Resources OR Wild-life Task Group A
- Managing Org Units
    - A list of the lower level org units that are directly owned & managed by this org unit
    - This property is empty if the org unit doesn’t manage any other org units
- User Accounts
    - A list of User Accounts that are directly owned by this org units
    - Can be empty initially

There is no system built-in Org Units are required.

2.1.1 Example Setup


Department of Agriculture and Water Resources
Water Division
Export Division
National Water Policy branch
Water Recovery branch
Export Standards branch

The org structure above can be setup as the followings:

ID:	OU01
Name:	Department of Agriculture and Water Resources
Managing Org Units	Water Division (OU02), Export Division (OU03)
User Accounts	Empty

ID:	OU02
Name:	Water Division
Managing Org Units	Water Recovery branch (OU04), Export Standards branch (OU05)
User Accounts	Empty

ID:	OU03
Name:	Export Division
Managing Org Units	Export Standards branch (OU06)
User Accounts	Empty

ID:	OU04
Name:	National Water Policy branch
Managing Org Units	Empty
User Accounts	Empty

ID:	OU05
Name:	Water Recovery branch
Managing Org Units	Empty
User Accounts	Empty

ID:	OU06
Name:	Export Standards branch
Managing Org Units	Empty
User Accounts	Empty

2.2 User Accounts

A user account present a user in the system. A user must have a user account before he can login to the system. All user accounts must be assigned at least with one Role (See section 2.6 Role). Before a user is logged into the system, he is act as a user account with the built-in `Anonymous Users` role.

A user account supports the followings properties:

- ID: user id (e.g. U001)
- Name: user’s name. .e.g James Blogs
- Job Title: e.g. Senior Manager or I.T specialist
    - This property will be copied from `Account Template` if the account is created from an `Account template` (see section 2.7)
- Roles: A list of roles this user is assigned.
    - This property will be copied from `Account Template` if the account is created from an `Account template` (see section 2.7)
- Org Unit:
    - The org unit that the user account belongs to
    - Optional. A user account could have empty value for this property (i.e. not belongs to any org units). .e.g. System used account or User Accounts with the built-in `Anonymous Users` role

2.2.1 Built-in User Accounts

System should comes with the following built-in User Accounts:

2.2.1.1 Anonymous User

System should assume all accessing users are a `Anonymous User` until they are authenticated as other users.

ID:	U001	
Property	Value	Comments
Name	Anonymous User	
Username	Empty	
Password	Empty	
Job Title	Empty	
Roles	Anonymous Users (R01)	The definition of R01 can be found from section 2.6
Org Unit	Empty	

2.3 Resources

A resource is a type of target objects that an operation (see section 2.4) can be performed on. E.g. `Users` is a resource that operation `Create User`, `Update User` or `Delete User` can be performed on. It’s necessary to define a list of built-in resource before we can define operations & permissions.

Please note: a resource doesn’t represent a individual object. Instead, it represent a group of objects that the same operation can be performed on. e.g `User Joe blogs` is a User Account that belongs to resource `Users`

Unlike other elements, system only support built-in resources rather than configurable resources. i.e. the definition of the supported resources must be defined at the time of the system design.

A resource support the following property:

- ID: the ID the of the resource. E.g. RES01
- Name: the resource
- Definition: the resource definition.
- Operation: A list of Operations (see section 2.4) support by this resource
- User Ownership Support: Yes / No
    - Whether the resource can be owned by a user account
    - The resource object creator will be auto set to the object owner by default
    - Other details see section 2.5
- Org Unit Ownership Support: Yes /No
    - Whether the resource can be owned by a org unit
    - The resource object creator’s directly org unit will be auto set to the object Org Unit owner by default
    - Other details see section 2.5
- Pre-Approved Permissions:
    - A list of `Pre-Approved Permissions`. A `Pre-Approved Permissions` is a special type of permission with `Pre-Authorised Constraint`. i.e. Permissions that will only be valid for a resource if the permission has been added to the resource’s `Pre-Approved Permissions` list. (See section 2.5)
    - A system may choose to add a Permission to this list in order to grant unconditional access of all operations included by the permission to any users who are assigned this permission.
    - This was designed for use case where you want to refine-tune the access control cross organisationally & regardless its ownership.
        - E.g. A user can may:
            - Create a `Black Forest Project Read Only` permission
            - Added this permission to all `Black Forest Project` datasets’ `Pre-Approved Permissions` list
            - Assign this permission to `Black Forest Project Read Only Users` Role
            - Assign this role to all people who should be able to view all `Black Forest Project` datasets

2.3.1 System Built-in Resources

2.3.1.1 Users

ID:	RES01
Name:	Users
User Ownership Support	YES
Org Unit Ownership Support	YES
Definition:	All Non built-in User Accounts in system
Operation:	\`Create User\` (OP001) \`View User\` (OP002) \`Update User\` (OP003) \`Delete User\` (OP004) \`Disable User\` (OP005)

2.3.1.2 Dataset

A dataset recourse includes any data related to a Dataset including

- Dataset metadata
- Any Distribution meta-data belongs to this dataset

A system may choose to implement the actual datasets data storage in many different ways. However, It is worth it to mention that the current Magda system stores the same dataset data in two different data storage:

- Traditional Database: For handling transactional operation
- Elasticsearch: For searching dataset data

Although, there are two copies of dataset data stored in two different physical storages. The two copies as a whole are considered as `one` dataset Resource.

For more details of Dataset resource definition, please see next section 2.3.1.2.1

ID:	RES02
Name:	Dataset
User Ownership Support	YES
Org Unit Ownership Support	YES
Operation:	`Create Draft Dataset` (OP006) `Read Draft Dataset` (OP007) `Read Published Dataset` (OP008) `Update Draft Dataset` (OP009) `Delete Draft Dataset` (OP010) `Publish Dataset` (OP011) `Unpublish Dataset` (OP012) `Update Published Dataset (Licence Info Only)` (OP029) * `Update Published Dataset (Non-Licence Info)` (OP030) *

* To implement those operations (OP029 & OP030), a system needs to validate all **published** dataset update request against the following rules to determine the type of the operations:

- If update request includes License info Only, it'll be considered as operation `Update Published Dataset (Licence Info Only)` (OP029)
- Otherwise, it'll be considered as operation `Update Published Dataset (Non-Licence Info Only)` (OP030)

The implementation of those operations (OP029 & OP030) should include a UI to highlight the difference between update data and current data in front of users in order to make sure users be aware of the permission may be required.

Please note: A system implementation should NOT provide an interface (API or UI) that allows any users to perform OP029 & OP030 directly. Instead, a system implementation should only allow users to update the published dataset from a newly created draft. Details see next section.

2.3.1.2.1 `Draft` vs `Published` Dataset

Every `Dataset` comes with a `Draft` property to indicate whether a `Dataset` is a draft or published. All `Datasets` created by default will be `Draft` `Dataset`.

When converting a Draft dataset to a `Non-Draft` dataset (i.e. Publish it), there should be no additional independent Dataset resource to be created and it should behave like status transit.

Only operations `Publish Dataset` (OP011) or `Unpublish Dataset` (OP012) can change the value of the `Draft` property:

- `Publish Dataset` (OP011): make `Draft` property set to false (i.e. Publish it)
- `Unpublish Dataset` (OP012): make `Draft` property set to true

Generally, a system should make sure the followings applied to `Draft` `Datasets`:

- A `Draft` dataset should be considered as more `private` data. Its visibility should be controlled within smaller scope.
    - E.g. A system may choose, by default, only people within the same org unit can see `draft` datasets. Or another system may choose only owner by default can see it (plus a list of selected users when create the dataset can see it).
    - I.e. At least, it shouldn't be public available

- A `Draft` dataset should be considered as more temporary records (i.e.may be updated or disposed before it turns into `Non-draft` / `Published`). Consequently, `loose` access control should be applied to them.
    - E.g. By default, the permission of creating a `Draft` dataset should be available widely. I.e. Most people should be able to create a Draft dataset.
    - By default, at least, the person who created the Draft dataset (Dataset Owner) should be able to update it or delete it (before it has been converted to Published dataset)

Furthermore, a system should make sure the followings applied to `Published` (Non-Draft) `Datasets`:

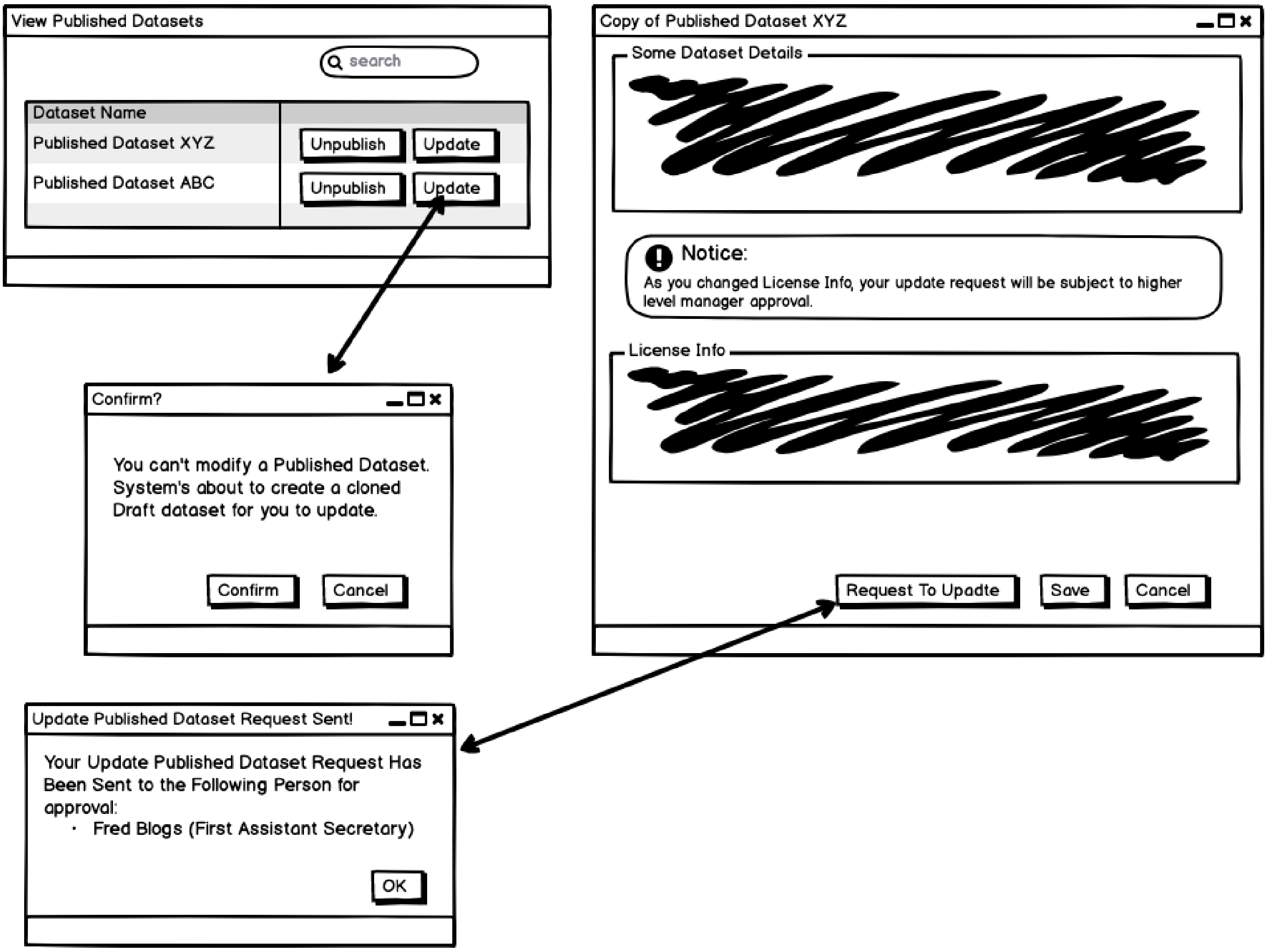
- A `Published` Dataset cannot be updated without going through a approval process.
- A `Published` Dataset cannot be updated directly. Instead, system should only allows user to update a published dataset from a draft dataset.

2.3.1.2.2 Approval Processing for Updating `Published` Datasets

A `Draft` Dataset also serve an important role through the approval / update process. A system implementation should NOT provide an interface (API or UI) that allows any users to perform OP029 & OP030 directly. Instead, a system implementation should only allow users to update the published dataset from a newly created draft.

i.e. In order to update a published dataset, it should require a user to:

- Create a draft dataset by cloning an existing published dataset.
  - Require operation `OP008` (Create Draft Dataset) & `OP006` (Create Draft Dataset)
- Update a draft dataset. Require operation `OP009` (Update Draft Dataset)
- User requests a published dataset to be updated based on his draft.
  - Require operation `Read Draft Dataset` (OP007) & `Read Published Dataset` (OP008)
- Depends on the difference between the `Draft dataset ` and `Published Dataset`, the system may consider the requested operation is either a OP029 (Update license Info of Published Dataset) or OP030 (Update Non-license Info of Published Dataset)
- Depends on the required operation type, system should lookup
  - All users who have either permission OP029 or OP030 to this dataset
  - And send them approval requests
- Any user who receives this approval request, may choose to approve this request by
  - Perform operation either OP029 or OP030



You can't modify a Published Dataset. System's about to create a cloned Draft dataset for you to update.

As you changed License Info, your update request will be subject to higher level manager approval.

Your Update Published Dataset Request Has Been Sent to the Following Person for approval:

- Fred Blogs (First Assistant Secretary)

2.3.1.3 Org Units

ID:	RES03
Name:	Org Units
User Ownership Support	Yes
Org Unit Ownership Support	No
Definition:	Org Unit defined in section 2.1
Operation:	`Create Org Unit` (OP013) `View Org Unit` (OP014) `Update Org Unit` (OP015) `Delete Org Unit` (OP016)

2.3.1.4 Permissions

ID:	RES04
Name:	Permissions
User Ownership Support	NO
Org Unit Ownership Support	NO
Definition:	Permissions defined in section 2.5
Operation:	`Create Permission` (OP017) `View Permission` (OP018) `Update Permission` (OP019) `Delete Permission` (OP020)

2.3.1.5 Roles

ID:	RES05
Name:	Roles
User Ownership Support	No
Org Unit Ownership Support	No
Definition:	Roles defined in section 2.6
Operation:	\`Create Role\` (OP021) \`View Role\` (OP022) \`Update Role\` (OP023) \`Delete Role\` (OP024)

2.3.1.6 Account Templates

ID:	RES06
Name:	Account Template
User Ownership Support	No
Org Unit Ownership Support	No
Definition:	Roles defined in section 2.7
Operation:	\`Create Account Template\` (OP025) \`View Account Template\` (OP026) \`Update Account Template\` (OP027) \`Delete Account Template\` (OP028)

2.4 Operations

An operation is an specific action that can be performed on a resource. E.g. `Update User` (OP002) operation is an operation can be performed on resource `Users` (RES01).

An operation supports the following properties:

- ID: operation ID. e.g. OP001
- Name: operation name
- Resource: supported resource. I.e. Which resource that the operation can be performed on

Unlike other elements, system only support built-in operations rather than configurable operations. i.e. the definition of the supported operations must be defined at the time of the system design.

2.4.1 System Built-in Operations

2.4.1.1 Create User

ID:	OP001
Name:	Create Users
Resource:	`Users` (RES01)
Description:	Create a user account. This operation is possibly to be performed by a system account.

2.4.1.2 View User

ID:	OP002
Name:	View Users
Resource:	`Users` (RES01)
Description:	Review a user account details.

2.4.1.3 Update User

ID:	OP003
Name:	Update Users
Resource:	\`Users\` (RES01)
Description:	Update a user account. This operation is possibly to be performed by a system account.

2.4.1.4 Delete User

ID:	OP004
Name:	Delete Users
Resource:	\`Users\` (RES01)
Description:	Delete a user account. This operation is possibly to be performed by a system account.

2.4.1.5 Disable User

ID:	OP005
Name:	Disable Users
Resource:	\`Users\` (RES01)
Description:	Disable a user account. Once a user account is disabled, no one can act on behalf of this user.

2.4.1.6 Create Dataset

ID:	OP006
Name:	Create Draft Dataset
Resource:	`Datasets` (RES02)
Description:	Create a Draft dataset including creating all metadata records e.g. dataset record & distribution records.

2.4.1.7 View Draft Dataset

ID:	OP007
Name:	Read Draft Dataset
Resource:	`Datasets` (RES02)
Description:	Read Draft data Dataset for a dataset including all metadata records e.g. dataset record & distribution records.

2.4.1.8 View Non-Draft Dataset

ID:	OP008
Name:	Read Published Dataset
Resource:	`Datasets` (RES02)
Description:	Read Non Draft data Dataset for a dataset including all metadata records e.g. dataset record & distribution records.

2.4.1.9 Update Dataset

ID:	OP009
Name:	Update Draft Dataset
Resource:	`Datasets` (RES02)
Description:	Update the data Dataset for a dataset including all metadata records e.g. dataset record & distribution records.

2.4.1.10 Delete Dataset

ID:	OP010
Name:	Delete Draft Dataset
Resource:	`Datasets` (RES02)
Description:	Delete the data Dataset for a dataset including all metadata records e.g. dataset record & distribution records.

2.4.1.11 Publish Dataset

ID:	OP011
Name:	Publish Dataset
Resource:	`Datasets` (RES02)
Description:	Publish the data Dataset & change its status from draft to non-draft.

2.4.1.12 Unpublish Dataset

ID:	OP012
Name:	Unpublish Dataset
Resource:	`Datasets` (RES02)
Description:	Unpublish the data Dataset & change its status from Non-draft to draft.

2.4.1.13 Create Org Unit

ID:	OP013
Name:	Create Org Unit
Resource:	`Org Units` (RES03)
Description:	Create a Org Unit record

2.4.1.14 View Org Unit

ID:	OP014
Name:	View Org Unit
Resource:	`Org Units` (RES03)
Description:	View a Org Unit record

2.4.1.15 Update Org Unit

ID:	OP014
Name:	Update Org Unit
Resource:	`Org Units` (RES03)
Description:	Update a Org Unit record

2.4.1.16 Delete Org Unit

ID:	OP016
Name:	Delete Org Unit
Resource:	`Org Units` (RES03)
Description:	Delete a Org Unit record

2.4.1.17 Create Permission

ID:	OP017
Name:	Create Permission
Resource:	`Permissions` (RES04)
Description:	Create a Permission record

2.4.1.18 View Permission

ID:	OP018
Name:	View Permission
Resource:	\`Permissions\` (RES04)
Description:	View a Permission record

2.4.1.19 Update Permission

ID:	OP019
Name:	Update Permission
Resource:	\`Permissions\` (RES04)
Description:	Update a Permission record

2.4.1.20 Delete Permissions

ID:	OP020
Name:	Delete Permissions
Resource:	\`Permissions\` (RES04)
Description:	Delete a Permission record

2.4.1.21 Create Role

ID:	OP021
Name:	Create Role
Resource:	\`Roles\` (RES05)
Description:	Create a Role record

2.4.1.22 View Role

ID:	OP022
Name:	View Role
Resource:	\`Roles\` (RES05)
Description:	View a Role record

2.4.1.23 Update Role

ID:	OP023
Name:	Update Role
Resource:	\`Roles\` (RES05)
Description:	Update a Role record

2.4.1.24 Delete Roles

ID:	OP024
Name:	Delete Roles
Resource:	\`Roles\` (RES05)
Description:	Delete a Roles record

2.4.1.25 Create Account Template

ID:	OP025
Name:	Create Account Template
Resource:	\`Account Templates\` (RES06)
Description:	Create a Account Template record

2.4.1.26 View Account Templates

ID:	OP026
Name:	Update Account Template
Resource:	\`Account Templates\` (RES06)
Description:	View a Account Template record

2.4.1.27 Update Account Templates

ID:	OP027
Name:	Update Account Template
Resource:	\`Account Templates\` (RES06)
Description:	Update a Account Template record

2.4.1.28 Delete Account Templates

ID:	OP028
Name:	Delete Account Template
Resource:	\`Account Templates\` (RES06)
Description:	Delete a Account Template record

2.4.1.29 Update Published Dataset (Licence Info Only)

ID:	OP029
Name:	Update Published Dataset (Licence Info Only)
Resource:	\`Datasets\` (RES02)
Description:	Update Published Dataset (Licence Info Only)

2.4.1.30 Update Published Dataset (Licence Info Only)

ID:	OP030
Name:	Update Published Dataset (Non-Licence Info)
Resource:	`Datasets` (RES02)
Description:	Update Published Dataset (Non-Licence Info)

2.5 Permissions

A permission defines a right that allows a user to perform a list of (supported) operations (see section 2.4) on a resource (see section 2.3) with optional three types of constraints. Each of the constraint limits when or where the permission is valid in certain way:

- `User Ownership Constraint`:
    - The permission is only valid when current user owns the target resource.
- `Org Unit Ownership Constraint`
    - The permission is only valid when current user's org unit (lower level org unit) owns the target resource.
- `Pre-Authorised Constraint`
    - The permission is only valid when this permission has been added to the resource's `Pre-Authorised Permission` List (see section 2.3)

System comes with a list of built-in permission (see section 2.5.1) which can used as the foundational building blocks of generic access control. System built-in permission can't be modified.

System may choose to create ad-hoc permission in order to define more fine-grained access controls cross organisationally & regardless its ownership.

E.g. A user can may:

- Create a `Black Forest Project Read Only` permission
- Added this permission to all `Black Forest Project` datasets' `Pre-Approved Permissions` list
- Assign this permission to `Black Forest Project Read Only Users` Role
- Assign this role to all people who should be able to view all `Black Forest Project` datasets Only

A permission supports the following properties:

- ID: permission ID. e.g. P001
- Name: e.g. Edit Own User
- Resource: Which resource this permission is about. E.g. `Users` (RES01)
- Operations:
    - A list of operations that are included in this permission
    - E.g. `Create User` OP001
- User Ownership Constraint:
    - Possible value Yes or No
    - Only can be Yes when the relevant resource supports `User Ownership Support` (see section 2.3.1)
    - If yes, the permission is only valid for Resource objects when current user is the owner
- Org Unit Ownership Constraint:
    - Possible value Yes or No
    - Only can be Yes if the relevant resource supports `Org Unit Ownership Support` (see section 2.3.1)
    - If yes, the permission is only valid for Resource objects when:
        - The current user belongs to a org unit
        - The current user's org unit or any lower level org units is the owner of the Resource object
- Pre-Authorised Constraint:
    - Possible value Yes or No
    - If yes. The permission will only be valid for a resource if the permission has been added to the resource's `Pre-Approved Permissions` list (see section 2.3).
    - This was designed for use case where you want to refine-tune the access control cross organisationally & regardless its ownership.

2.5.1 System Built-in Permissions

System allows customised permission to be created by users. Here, we need to define a list of system built-in permission to support other system built-in elements.

2.5.1.1 View Non-Draft Dataset

ID:	P001
Name:	View Non-Draft Dataset
Resource:	\`Data Datasets\` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	No
Operations:	\`View Non-Draft Dataset\` (OP008)
Description:	This permission allows users to view non-draft Dataset.

2.5.1.2 View Dataset (Within Org Unit)

ID:	P002
Name:	View Draft Dataset
Resource:	\`Data Datasets\` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	No
Operations:	\`View Draft Dataset\` (OP007)
Description:	This permission allows users to view draft Dataset.

2.5.1.3 Create Draft Dataset (Within Org Unit)

ID:	P003
Name:	Create Draft Dataset (Within Org Unit)
Resource:	\`Data Datasets\` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	\`Create Draft Dataset\` (OP006)
Description:	This permission allows users to view Dataset within their own org units

2.5.1.4 Update Draft Dataset (Within Org Unit)

ID:	P004
Name:	Update Draft Dataset (Within Org Unit)
Resource:	\`Data Datasets\` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	\`Update Draft Dataset\` (OP009)
Description:	This permission allows users to update Dataset within their own org units. They can also update catalog not created by themselves within the org unit.

2.5.1.5 Update Draft Dataset (Own Dataset)

ID:	P005
Name:	Update Draft Dataset (Own Dataset)
Resource:	`Data Datasets` (RES02)
User Ownership Constraint	Yes
Org Unit Ownership Constraint	NO
Operations:	`Update Draft Dataset` (OP009)
Description:	This permission allows users to update Datasets that created by themselves only. Can be used as a general permission to allow most user to be able to update the Dataset created by themselves only.

2.5.1.6 Delete Draft Dataset (Within Org Unit)

ID:	P006
Name:	Delete Draft Dataset (Within Org Unit)
Resource:	`Data Datasets` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	`Delete Draft Dataset` (OP010)
Description:	This permission allows users to delete the Dataset within their own org units. They can also delete catalog not created by themselves within the org unit.

2.5.1.7 Delete Draft Dataset (Own Dataset)

ID:	P007
Name:	Delete Draft Dataset (Own Dataset)
Resource:	`Data Datasets` (RES02)
User Ownership Constraint	Yes
Org Unit Ownership Constraint	No
Operations:	`Delete Draft Dataset` (OP010)
Description:	This permission allows users to delete Datasets that created by themselves only. Can be used as a general permission to allow most user to be able to delete the Dataset created by themselves only.

2.5.1.8 Dataset Publish / Unpublish (Within Org Unit)

ID:	P008
Name:	Dataset Publish / Unpublish (Within Org Unit)
Resource:	`Data Datasets` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	`Publish Dataset` (OP011) `Unpublish Dataset` (OP012)
Description:	This permission allows users to publish / unpublish Dataset within their own org units.

2.5.1.9 Data Administration (System)

ID:	P009
Name:	Data Administration (System)
Resource:	`Data Datasets` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	No
Operations:	`Create Draft Dataset` (OP006) `View Draft Dataset` (OP007) `View Non-Draft Dataset` (OP008) `Update Draft Dataset` (OP009) `Delete Draft Dataset` (OP010) `Publish Dataset` (OP011) `Unpublish Dataset` (OP012) `Update Published Dataset (Licence Info Only)` (OP029) `Update Published Dataset (Non-Licence Info Only)` (OP030)
Description:	This permission is a short-cut to allow special users has system wide all data Datasets access. Could be useful for user account created for Magda team (client side admin team) for internal use

2.5.1.10 User Administration (Within Org Unit)

ID:	P010
Name:	User Administration (Within Org Unit)
Resource:	\`Users\` (RES01)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	\`Create User\` (OP001) \`View User\` (OP002) \`Update User\` (OP003) \`Delete User\` (OP004) \`Disable User\` (OP005)
Description:	This permission is a short-cut to allow special users (within the org unit) to manage all users

2.5.1.11 User Administration (System wide)

ID:	P011
Name:	User Administration (System wide)
Resource:	\`Users\` (RES01)
User Ownership Constraint	No
Org Unit Ownership Constraint	No
Operations:	\`Create User\` (OP001) \`View User\` (OP002) \`Update User\` (OP003) \`Delete User\` (OP004) \`Disable User\` (OP005)
Description:	This permission is a short-cut to allow special users (system wide) to manage all users

2.5.1.12 Permission Administration (System wide)

ID:	P012
Name:	Permission Administration (System wide)
Resource:	\`Permissions\` (RES04)
User Ownership Constraint	No
Org Unit Ownership Constraint	No
Operations:	\`Create Permission\` (OP017) \`View Permission\` (OP018) \`Update Permission\` (OP019) \`Delete Permission\` (OP020)
Description:	This permission is a short-cut to allow special users (system wide) to setup custom permission

2.5.1.13 Role Administration (System wide)

ID:	P013
Name:	Role Administration (System wide)
Resource:	\`Roles\` (RES05)
User Ownership Constraint	No
Org Unit Ownership Constraint	No
Operations:	\`Create Role\` (OP021) \`View Role\` (OP022) \`Update Role\` (OP023) \`Delete Role\` (OP024)
Description:	This permission is a short-cut to allow special users (system wide) to setup custom roles

2.5.1.14 Account Template Administration (System wide)

ID:	P014
Name:	Account Template Administration (System wide)
Resource:	\`Account Templates\` (RES06)
User Ownership Constraint	No
Org Unit Ownership Constraint	No
Operations:	\`Create Account Template\` (OP025) \`View Account Template\` (OP026) \`Update Account Template\` (OP027) \`Delete Account Template\` (OP028)
Description:	This permission is a short-cut to allow special users (system wide) to setup custom account templates

2.5.1.15 Org Unit Administration

ID:	P015
Name:	Org Unit Administration
Resource:	\`Org Units\` (RES03)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	\`Create Org Unit\` (OP013) \`View Org Unit\` (OP014) \`Update Org Unit\` (OP015) \`Delete Org Unit\` (OP016)
Description:	This permission is a short-cut to allow special users to manage all org units

2.5.1.16 View Non-Draft Dataset (Within Org Unit)

ID:	P016
Name:	View Non-Draft Dataset (Within Org Unit)
Resource:	\`Data Datasets\` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	\`View Non-Draft Dataset\` (OP008)
Description:	This permission allows users to view non-draft Dataset (within Org Unit).

2.5.1.17 View Draft Dataset (Within Org Unit)

ID:	P017
Name:	View Draft Dataset (Within Org Unit)
Resource:	`Data Datasets` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	`View Draft Dataset` (OP007)
Description:	This permission allows users to view draft Dataset (within Org Unit).

2.5.1.18 Update Published Dataset (Licence Info Only)

ID:	P018
Name:	Update Published Dataset (Licence Info Only)
Resource:	`Data Datasets` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	`Update Published Dataset (Licence Info Only)` (OP029)
Description:	This permission allows user to update a Published dataset (Licence Info Only). A user with this permission indicates he can authorise any `Licence Info` updates to a published dataset as part of the approval process. More details see section 2.3.1.2.2

2.5.1.19 Update Published Dataset (Non-Licence Info Only)

ID:	P019
Name:	Update Published Dataset (Non-Licence Info Only)
Resource:	`Data Datasets` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	Yes
Operations:	`Update Published Dataset (Non-Licence Info Only)` (OP030)
Description:	This permission allows user to update a Published dataset (Non-Licence Info Only). A user with this permission indicates he can authorise any `Non-Licence Info` updates to a published dataset as part of the approval process. More details see section 2.3.1.2.2

2.5.1.20 Anonymous Users Published Dataset Read-Only

ID:	P020
Name:	Anonymous Users Published Dataset Read-Only
Resource:	`Data Datasets` (RES02)
User Ownership Constraint	No
Org Unit Ownership Constraint	No
Pre-Authorised Constraint	YES
Operations:	`Read Published Dataset` (OP008)
Description:	This permission is a permission with Pre-Authorised Constraint. I.e. User can only access the resources that explicitly authorises this permission. See section 2.3 & 2.5. Resources authorises this permission will grant Anonymous Users Read only access

2.6 Roles

A role defines a group of permission. i.e. It's a container of permission and allows a group of permissions to be grouped together in order to assign to a user to grant him particular type of access. A permission can not be directly assign to a user.

System comes with a list of built-in roles (see section 2.6.1) which can used as the foundational building blocks of generic access control. System built-in roles can't be modified.

System may choose to create ad-hoc roles to serve as the containers of ad-hoc permissions (see 2.5) as system built-in roles can't be modified --- subject to system implementation decisions.

E.g. A system may choose to create a default ad-hoc role (e.g. named as `Default Adhoc Role`) for every users on their creation. And later always assign any adhoc permission to this role for this user.

Alternatively, a system may choose to create an adhoc role per adhoc permission when adhoc permission is created. And then, assign this ad-hoc role to the user.

A role supports the followings properties:

- ID: id of the role: e.g. R01
- Name: the name of the role. .e.g Anonymous Users
- Permissions:
    - A list of permissions included in this role

2.6.1 System Built-in Roles

System allows customised role to be created by users. Here, we need to define a list of system built-in role to support other system built-in elements.

2.6.1.1 Anonymous Users

ID:	R01
Name:	Anonymous Users
Permissions:	Anonymous Users Published Dataset Read-Only (P020)
Description:	This role allows all users (even non-logged in users) view selected Non-Draft Datasets.

2.6.1.2 Standard Users

ID:	R02
Name:	Standard Users / Data Producer
Permissions:	View Non-Draft Dataset (Within Org Unit) (P016) View Draft Dataset (Within Org Unit) (P017) Create Draft Dataset (Within Org Unit) (P003) Update Draft Dataset (Own Dataset) (P005) Delete Draft Dataset (Own Dataset) (P007)
Description:	This role allows users to: - View Non-Draft Dataset within his org unit - View Draft Dataset within his org unit - Create Dataset within his org unit - Edit Dataset that created by his own - Delete Dataset that created by his own

2.6.1.3 Data Master Users

ID:	R03
Name:	Data Steward Users
Permissions:	View Non-Draft Dataset (Within Org Unit) (P016) View Draft Dataset (Within Org Unit) (P017) Create Dataset (Within Org Unit) (P003) Update Dataset (Within Org Unit) (P004) Delete Dataset (Within Org Unit) (P006) Dataset Publish / Unpublish (Within Org Unit) (P008) Update Published Dataset (Non-Licence Info Only) (P019)
Description:	This role allows users to: - View Non-Draft Dataset within his org unit - View Draft Dataset within his org unit - Create Dataset within his org unit - Edit Dataset within his org unit - Delete Dataset within his org unit - Publish dataset within his org unit - Authorise non-license changes to Published datasets

2.6.1.4 Managers

ID:	R04
Name:	Managers
Permissions:	View Non-Draft Dataset (Within Org Unit) (P016) View Draft Dataset (Within Org Unit) (P017) Create Dataset (Within Org Unit) (P003) Update Dataset (Within Org Unit) (P004) Delete Dataset (Within Org Unit) (P006) User Administration (Within Org Unit) (P010)
Description:	This role allows users to: - View Non-Draft Dataset within his org unit - View Draft Dataset within his org unit - Create Dataset within his org unit - Edit Dataset within his org unit - Delete Dataset within his org unit - Create & Manage all user accounts within his org unit

2.6.1.5 Senior Managers

ID:	R05
Name:	Senior Managers
Permissions:	View Non-Draft Dataset (Within Org Unit) (P016) View Draft Dataset (Within Org Unit) (P017) Create Dataset (Within Org Unit) (P003) Update Dataset (Within Org Unit) (P004) Delete Dataset (Within Org Unit) (P006) Dataset Publish / Unpublish (Within Org Unit) (P008) Update Published Dataset (Non-Licence Info Only) (P018) Update Published Dataset (Licence Info Only) (P019) User Administration (Within Org Unit) (P010)
Description:	This role allows users to: - View Non-Draft Dataset within his org unit - View Draft Dataset within his org unit - Create Dataset within his org unit - Edit Dataset within his org unit - Delete Dataset within his org unit - Create & Manage all user accounts within his org unit - Authorise non-license changes to Published datasets - Authorise license changes to Published datasets

2.6.1.6 Data Administrators

ID:	R06
Name:	Data Administrators
Permissions:	Data Administration (System) (P009)
Description:	This role allows users to do any system wide data Dataset operation

2.6.1.7 System Administrators

ID:	R07
Name:	System Administrators
Permissions:	Data Administration (System) (P009) User Administration (System) (P011) Permission Administration (System wide) (P12) Role Administration (System wide) (P13) Account Template Administration (System wide) (P14) Org Unit Administration (System wide) (P15)
Description:	This role allows users to do any system wide: - data Dataset operation - User administration - Permission, role related setup - Setup org structure

2.7 Account Template

A account template defines the default setup (e.g. roles, job title, default org unit etc.) based on known information from an account creation request. This is useful for handling automatically account creation through SSO (single sign-on) interface.

When a SSO account creation request comes in, system should take the incoming authenticated foreign user information to run through the `Match Rules` of each `Account Template` until find a match. The matched account template will be used for creating a new Magda account for the user.

It supports the following properties:

- ID: id of the account template. E.g. T01
- Name: name of the template. E.g. Department A Manager Template
- Job Title: Specify the `Job Title` property of the created `User Account` (see section 2.2)
- Roles: Specify the `Roles` property of the created `User Account` (see section 2.2)
- Org Unit: Specify the `Org Unit` property of the created `User Account` (see section 2.2)
- Match Rules: Specify what kind of SSO account creation request can use this account template for new user account creation

There is no system built-in Account Templates are required.

2.7.1 Example Setup

ID:	T04
Name:	Managers
Job Title:	Senior Managers of Water Division
Roles:	Managers (R04) See built-in object defined in section 2.6.1.4
Org Unit:	Water Division (U02) See example object defined in section 2.1
Match Rules:	Match SSO account requests that: - For a Manager - For a person from Water Division & Department of Agriculture and Water Resources

3. Constraints & Limitations

The model described in this document has the following limitations:

3.1 No Multiple Position Support

This model doesn’t support user accounts have more than one position. E.g.

User A is a Manager at Water Division and also a Data Master at Export Division at the same time.

For this situation, user A is required to use two separate user accounts to access the two separate positions.

3.2. Only Pre-Defined Resource & Operations

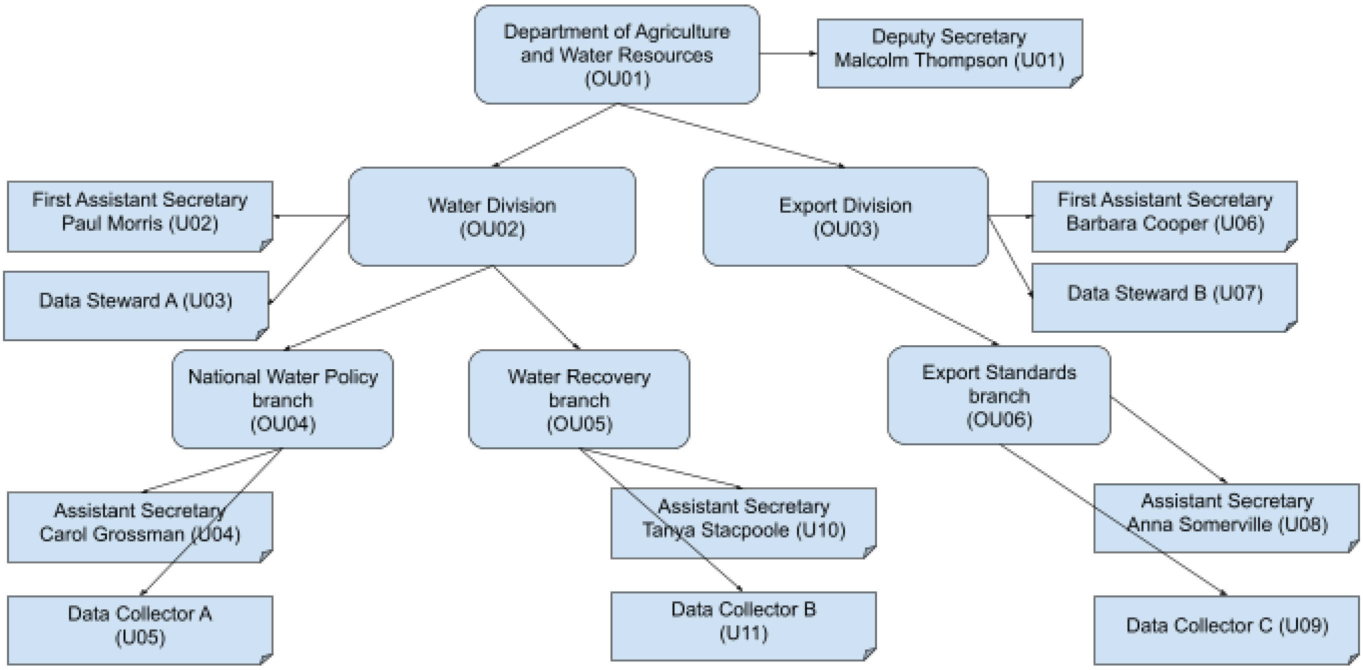
This model requires supported Resources & Operations are pre-defined.

i.e. At system design stage, we need to have the clear vision of all supported functionality and security implication of all functions. We need to list:

- all possible actionable objects
- all possible actions for each of actionable object

4. Sample Client Org Structure Setup (A)

Suppose a client has the following org structure:


Department of Agriculture and Water Resources (OU01)
Deputy Secretary Malcolm Thompson (U01)
First Assistant Secretary Paul Morris (U02)
Data Steward A (U03)
Water Division (OU02)
Export Division (OU03)
First Assistant Secretary Barbara Cooper (U06)
Data Steward B (U07)
National Water Policy branch (OU04)
Water Recovery branch (OU05)
Export Standards branch (OU06)
Assistant Secretary Carol Grossman (U04)
Assistant Secretary Tanya Stacpoole (U10)
Assistant Secretary Anna Somerville (U08)
Data Collector A (U05)
Data Collector B (U11)
Data Collector C (U09)

The required the org structure & access control can be setup as followings:

4.1 Org Units

ID:	OU01
Name:	Department of Agriculture and Water Resources
Managing Org Units	Water Division (OU02), Export Division (OU03)
User Accounts	Empty

ID:	OU02
Name:	Water Division

Managing Org Units	Water Recovery branch (OU04), Export Standards branch (OU05)
User Accounts	Empty

ID:	OU03
Name:	Export Division
Managing Org Units	Export Standards branch (OU06)
User Accounts	Empty

ID:	OU04
Name:	National Water Policy branch
Managing Org Units	Empty
User Accounts	Empty

ID:	OU05
Name:	Water Recovery branch
Managing Org Units	Empty
User Accounts	Empty

ID:	OU06
Name:	Export Standards branch
Managing Org Units	Empty
User Accounts	Empty

4.2 User Accounts

ID:	U01
Name:	Malcolm Thompson
Job Title:	Deputy Secretary
Roles:	Senior Managers (R05) Data Master Users (R03)
Org Unit:	Department of Agriculture and Water Resources (U01)

ID:	U02
Name:	Paul Morris
Job Title:	First Assistant Secretary
Roles:	Senior Managers (R05) Data Master Users (R03)
Org Unit:	Water Division (U02)

ID:	U03
Name:	Data Steward A
Job Title:	Data Steward
Roles:	Data Steward Users (R03)
Org Unit:	Water Division (U02)

ID:	U04
Name:	Carol Grossman
Job Title:	Assistant Secretary
Roles:	Managers (R04) Data Steward Users (R03)
Org Unit:	National Water Policy branch (U04)

ID:	U05
Name:	Data Collector A
Job Title:	Data Collector
Roles:	Standard Users (R02)
Org Unit:	National Water Policy branch (U04)

ID:	U06
Name:	Barbara Cooper
Job Title:	First Assistant Secretary
Roles:	Senior Managers (R05) Data Steward Users (R03)
Org Unit:	Export Division (U03)

ID:	U07
Name:	Data Steward B
Job Title:	Data Steward
Roles:	Data Steward Users (R03)
Org Unit:	Export Division (U03)

ID:	U08
Name:	Anna Somerville
Job Title:	Assistant Secretary
Roles:	Managers (R04) Data Steward Users (R03)
Org Unit:	Export Standards branch (U06)

ID:	U09
Name:	Data Collector C
Job Title:	Data Collector
Roles:	Standard Users (R02)
Org Unit:	Export Standards branch (U06)

ID:	U10
Name:	Tanya Stacpoole
Job Title:	Assistant Secretary
Roles:	Managers (R04) Data Steward Users (R03)
Org Unit:	Water Recovery branch (U05)

ID:	U11
Name:	Data Collector B
Job Title:	Data Collector
Roles:	Standard Users (R02)
Org Unit:	Water Recovery branch (U05)