Systems - DB: Roles provisioning

This tutorial is intended as a guide for administrators that want to provision roles from CzechIdM to another Database.

You will learn

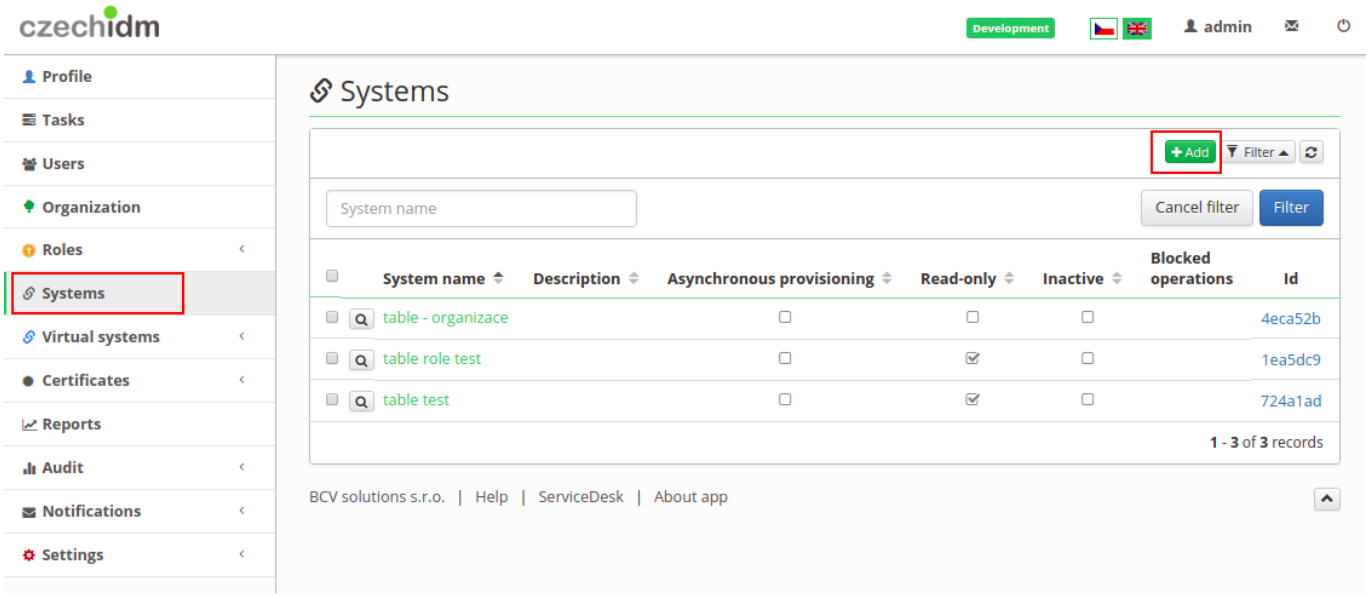
- how to connect an DB system for role provisioning
- how to propagate just those roles we want

DB's Table

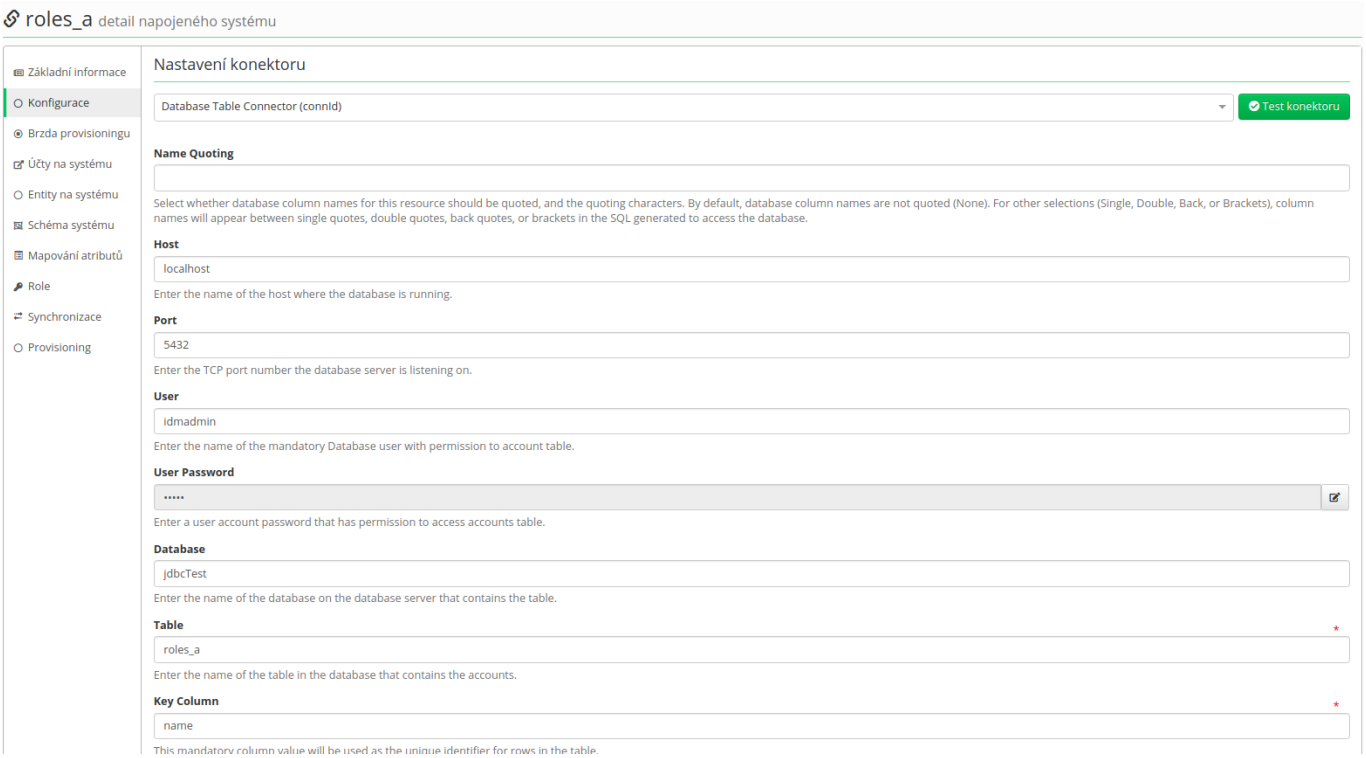
For this example I created simple table with just one column 'name' as varchar, which will be name of role and identifier.

Create system

- Go to **Systems** in the left menu and then click on **Add**.



- Fill in name of a system. And click on **Save and continue**.
- In tab **Configuration** choose Database table connector



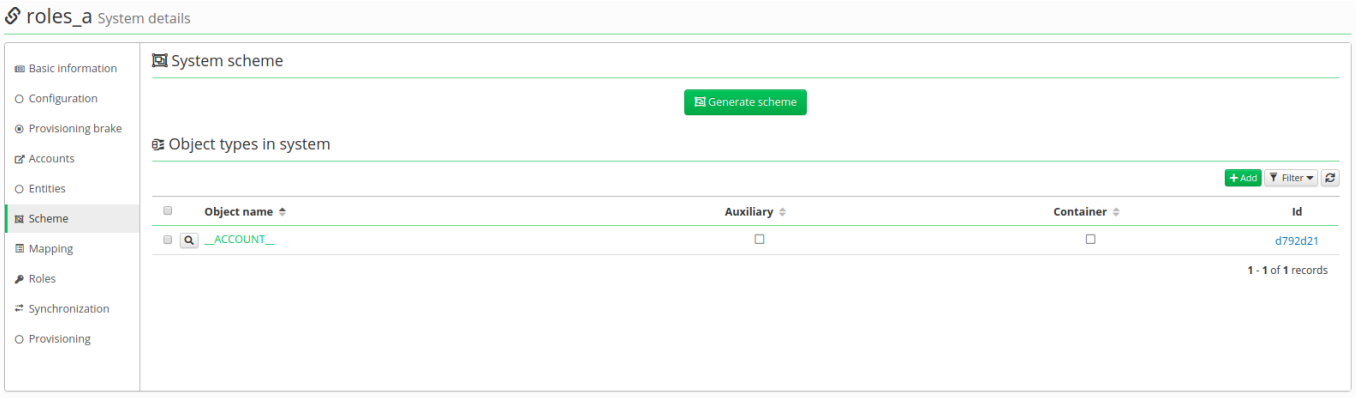
Connector configuration

On this page fill in these important values:

- **Host** - IP address of ad server or hostname
- **Port** - on this port will server listen
- **User** - with this username connector will connect to the DB system, this user has to have enough rights to write.
- **User password** - password of the "user" account
- **Database** - name of database, we want to connect
- **Table** - name of table
- **Key column** - column as an identifier in table
- **JDBC Driver** - it is based on type of database, e.g. for Postgres it is "org.postgresql.Driver"
- **JDBC Connection URL** - it is based on type of database

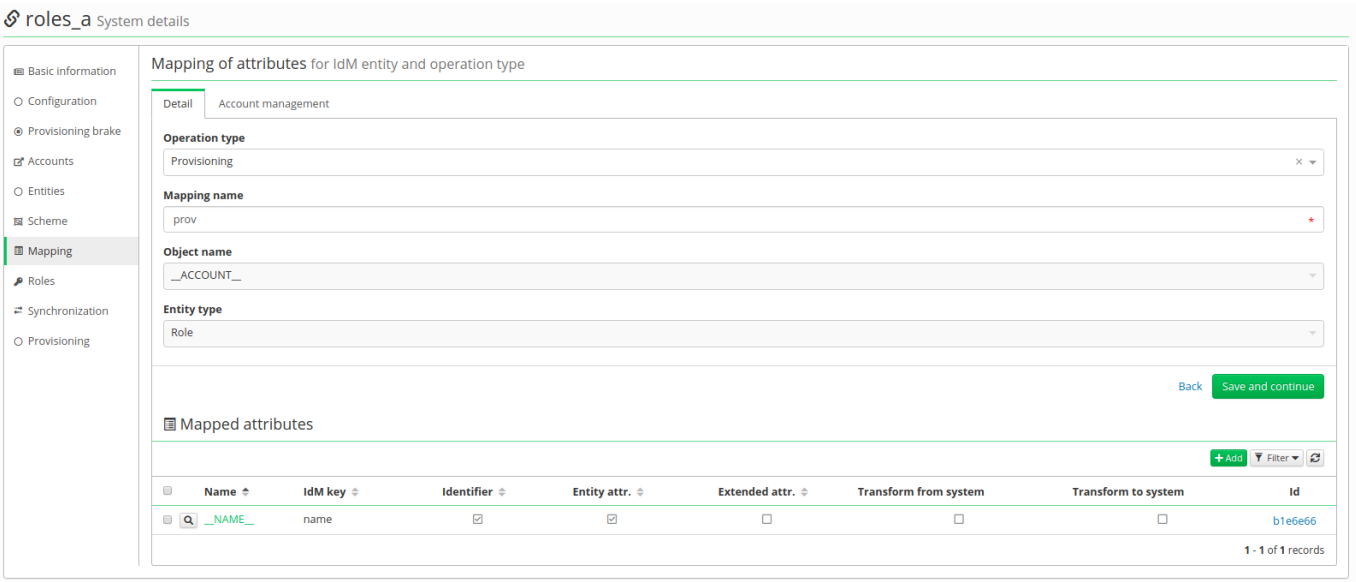
Connector's mapping

- Firstly in **Scheme** tab generate a schema with a green button. If there is some exception, you have probably mistake in the configuration of the connector.



- Then in **Mapping** tab create new mapping - provisioning (__ACCOUNT__ (Object name), Role (Entity type)). {If you are using Active Directory, select __GROUP__ as Object name }
- Now we will map just 1 attribute. Click on green add button like on picture below and this fill in:

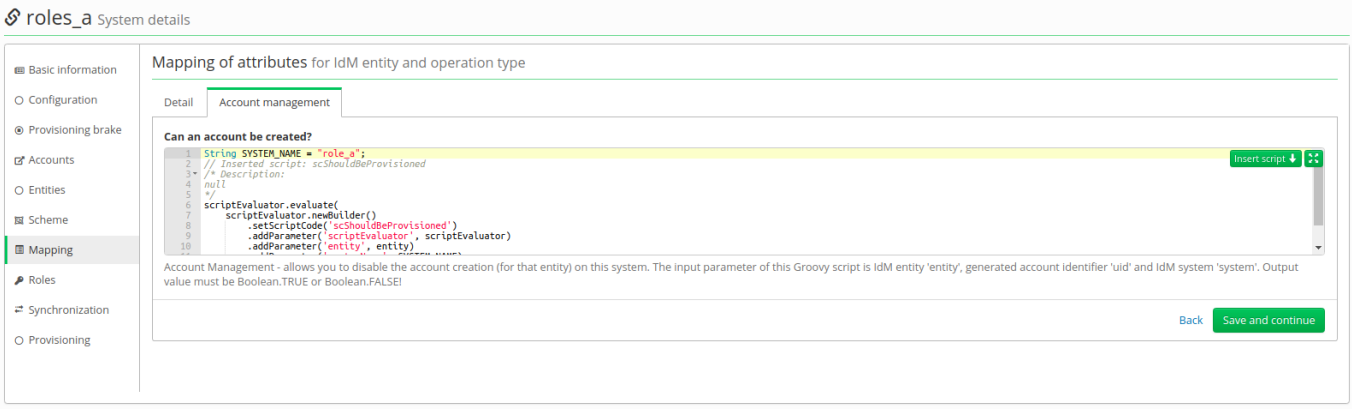
```
| Attribute in schema    | Name          | Attribute          | IdM key
|
| __NAME__ (__ACCOUNT__) | name          | identifier, entity | name
|
```



Make a script

At this point, provisioning of roles is active and if we create a role or re-save already existing, role will be provisioned to database. But we probably do not want propagate all of roles.

Select our system and then agenda **Mapping**. Select just created provisioninig mapping. On this page there is another tab **Account Management**. Here you can write a script or add one with green button **Insert script**. For example you can specify which roles will be propagated based on role name (roles_a:roleToBeProvisioned)



or if role is in specified role catalogue (roles_a catalogue).

```
// Inserted script: IsRoleInCatalogue
/* Description:
Is role in the catalogue? Script return "true" if given (input parameter
"role") IdmRoleDto is in supported catalogue (given in the parameter
"catalogueCode"). Search is recursively.
*/
scriptEvaluator.evaluate(
    scriptEvaluator.newBuilder()
        .setScriptCode('IsRoleInCatalogue')
        .addParameter('scriptEvaluator', scriptEvaluator)
        .addParameter('uid', uid)
        .addParameter('entity', entity)
        .addParameter('system', system)
        .addParameter('role', entity)
        .addParameter('catalogueCode', '123') // '123' represents a catalog
code
    .build());
```

Beware: If you add first (roles_a:roleToBeProvisioned) script after provisioning of a role. This script will not prevent future provisioning of this role. You have to remove role's account on this system. In agenda **Roles** on left menu you can find the role, click on magnifying glass. In tab **Accounts** you can see all accounts of this role (there could be more items, if role was synchronized from system or provisioned to more systems). Here if you remove account, role will be erased on end system. Future provisioning of the role to this system based on script mentioned above.