

CzechIdM - extras

CzechIdM - extras contains various features, which are not suited to be in any other module. List of the currently supported features is below.

Currently supported CzechIdM version : 11.0.x

<https://github.com/bcvsolutions/czechidm-extras>

Table of compatible versions

Extras version	Min dependency versions	Max dependency versions	Note
2.5.0	idm-core-10.3.3	UNKNOWN	
2.6.0	idm-core-10.4.4	UNKNOWN	
2.6.1	idm-core-10.4.8	UNKNOWN	This version is needed for Automatic roles on tree nodes - import data from CSV
2.7.0	idm-core-10.4.9	UNKNOWN	

Developing and releasing

How to develop a new feature in extras:

1. Create a specification page in private section and consult it with module owner and other colleagues. Specification page should contain:
2. Usecases - why we or user need this feature, what problem does it solve?
3. Why is it in extras and not in core, or other module?
4. Functional specification - how should it work, edge cases
5. Ask a module owner, if this feature can be a part of extras and in which version it will be published
6. Create ticket in Redmine with final requirements and with correct target version
7. Implement the feature in a separate GIT branch
8. Create merge request to develop or LTS version
9. Get someone from product team, or module owner to review your changes
10. After successful review, ask module owner to merge your code

Rules for code review:

- All new features have at least 80 percent test coverage
- All features are documented (tutorials are welcomed and may even be required if the feature is complicated)
- There are no sonar issues in committed code
- Changelog is updated
- Feature is by default turned off (can be enabled either by processor, or configuration property)



If your code does not meet the requirements mentioned above, it may be rejected.

- When developing, use our standard gitflow:
 - Branch per feature. Branch naming as usual.
 - Develop on top of the develop.
 - Master branch contains tagged releases.
 - The **only** way for code to get into master is by pull request develop → master.
- Release process
 1. After merging all features wanted for the release, one (selected) developer builds the develop. If it builds fine and tests are also OK, the developer edits module version in pom.xml and package.json files and sets it to X.Y.Z.
 2. Developer creates pull request on GitHub to merge develop → master.
 3. Repo admin (or any other authorized user) reviews the pull request, can request changes if necessary. Unresolved TODOs, missing comments, bad codestyle or documentation, suspected bugs, etc. - all those things can be grounds for change request.
 4. If the pull request is OK, repo admin merges it.
 5. Repo admin creates a new release in GitHub interface, version is set to X.Y.Z to correspond with version set in project sources.
 6. Repo admin pushes release into BCV Nexus.
 7. After release, repo admin makes changes on the develop branch: upps module version to X(+1).Y(+1).Z(+1) - SNAPSHOT in the pom.xml and package.json. What the next version of the module will be is up to discussion preceding the release.

Automatic role definitions - Import all rules LRT

Since module version **1.9.0**. Documentation is available here: [Automatic role definitions - Import all rules](#)

Assign roles to contract EAV - Import of data from CSV LRT

Documentation is available here: [Assign roles to contract EAV - Import of data from CSV](#)

Roles - Import of data from CSV LRT

Documentation is available here: [Roles - Import of data from CSV](#)

Automatic roles - adding role by node in structure

Documentation is available here: [Automatic roles - adding role by node in structure](#)

Create automatic roles by tree structure, based on user's roles LRT

Since module version **2.8.0** Documentation is available here: [Create automatic roles by tree structure, based on user's roles](#)

Status task

Documentation is available here: [Status task - How to prepare the task](#) Information about content is here: [Notification content](#)

SSO authenticate

Documentation is available here: [sso_authenticate](#)

Role force provisioning to particular system

The tutorial is available here: [Provisioning - how to force provisioning for roles](#)

Guarantees of roles can assign their roles to everybody

This feature enable that if you are guarantee at least for one role then you will see all users and you can assign/delete/edit roles for which you are guarantee. You can see all user's roles but you can't change the others for which you are not guarantee

For correct behavior you need to configure three new evaluators to userRole:

- IdentityAccessForRoleGuaranteeEvaluator
- IdentityRoleAccessForRoleGuaranteeEvaluator
- RoleRequestAccessForRoleGuaranteeEvaluator

Other thing you need to do is to enable service ExtrasIdmConceptRoleRequestService. This service is by default turned off in extras module. Go to your project modul and create new service which will inherit from ExtrasIdmConceptRoleRequestService and add annotation Primary and Service.

Update IdmConceptRoleRequestDto is allowed everybody that will change only audited fields or systemState field (this is for update state of whole request after retry mechanism or approving virtual request).

Report Compare values in IdM with values in system

Report will compare value of attributes with connected system. Connected system does not need to be in read only. More information is available here: [Report - Compare values in IdM to system \(extras, before CzechIdM 12\)](#)

Notification about the end of identity's last contract

A notification about the end of identity's last contract will be sent to those who have a specified role assigned and optionally the manager of the user. A different notification can be sent before the contract ends and when it ends. More information is available here: [Notification - the end of identity's last contract](#)

Edit: full IdmIdentityDto was added for use in a template in 1.7.0

Edit: Support for technical identities added for use in version 1.9.0

Notification about the start of identity's contract

Since version 3.2.3

A notification about start of identity's contract will be send to specific persons. You can also set the topic which will be used for notification. More information is available here: [Notification - the start of identity's contract](#)

Notification about the change of identity's contract

Since version 3.4.1

A notification about a change of identity's contract will be sent to specific persons. You can also set the topic which will be used for notification. More information is available here: [Notification - the change of identity's contract](#)

Script split titles before and after

Almost every project receive all titles in one string and IdM allow separates titles before and after. For this case was created in **ExtrasUtils** two methods **getTitlesAfter** and **getTitlesBefore**. And transformation scripts **extrasGetTitlesBefore** and **extrasGetTitlesAfter**, transformation scripts calls method from utils.

Dictionary with titles can be setup by configuration properties. Default values exists in [BE](#)

```
idm.sec.extras.configuration.titlesAfter=Ph.D.,Th.D.,CSc.,DrSc.,dr. h.
c.,DiS.,MBA,LL.M.,FESC,MHA,FEB0,FES0,FEBU,FACC
idm.sec.extras.configuration.titlesBefore=Bc.,BcA.,Ing.,Ing.
arch.,MUDr.,MVDr.,MgA.,Mgr.,JUDr.,PhDr.,RNDr.,PharmDr.,ThLic.,ThDr.,prof.,do
c.,PaedDr.,Dr.,PhMr.,MDDr.
```

in version 2.8.0 was added the option to select a separator for source string and for titles before and after values in IdM.

```
idm.sec.extras.configuration.titlesSourceSeparator - a character, that
separates titles in the source. Default" "
idm.sec.extras.configuration.titlesBeforeSeparator - titles before separator
for IdM. Default", "
idm.sec.extras.configuration.titlesAfterSeparator - titles before separator
for IdM. Default", "
```

Import automatic roles on tree nodes

You can use this tool to create automatic roles which are assigned based on the position within the organization structure using a CSV file as a source. More information is available here: [Automatic roles on tree nodes - import data from CSV](#)

Groups synchronization workflow

Since module version **1.4.0** there exists improved workflow in the extras module for groups synchronization than in the core module. The workflow **Extras Synchronization - Roles from LDAP** (extrasSyncRoleLdap) has more features and some fixed bugs than the product workflow syncRoleLdap. The workflow in the product is not developed anymore, because we plan to add more user-friendly configuration of the feature in future versions of the product.

Documentation for configuration is available in [Group synchronization workflow - Application properties](#).

Note: the workflow extrasSyncRoleLdap depends on some services implemented in the extras module, so to use this workflow, you must deploy the whole module to CzechIdM.

Workflow to disable contract on MISSING_ACCOUNT

Setting this workflow (extrasDisableMissingContract) as workflow for action in contract reconciliation will disable contract, when its being synchronized. It can be used for example, in situations when contracts are being deleted from source data after expiration and they keep being stuck in MISSING_ACCOUNT state.

Note: When using this workflow, please consider the possibility that the contracts may "reappear" in the source data. In such case, it would make sense to enable the contracts again, e.g. by mapping the attribute "state" and filling it by value null (= active contract).

Groups membership in multi domain (cross domain) AD environment

Since module version **1.8.0**

Documentation is available [Systems - AD: Manage groups membership in multi domain \(cross domain\) AD environment](#)

Evaluator (permissions) for identities that has relationship on defined organization unit

Since module version **1.9.0. Available only on LTS version!**

Documentation is available [there](#)

Evaluator (permissions) for roles that is inside defined role catalogue

Since module version **1.9.0. Available only on LTS version!**

Documentation is available [there](#).

Import code list and it's items

Since module version **1.9.0** and **2.4.0**

Documentation is available [there](#)

Synchronize code list items from end system

Since module version **3.2.0**

Documentation is available [there](#)

Evaluators (permissions) for Identities and Contracts which are both subordinate and have given projection

Since module version **2.4.0**.

Documentation is available [there](#).

Workflows for approval of role assignment

Since module version **2.3.0**

Documentation is available [there](#)

Script for soft abbreviate strings (extrasAbbreviateString)

Since module version **3.2.0**

Script called **extrasAbbreviateString** abbreviate given string by default with maximum length 255. Maximum length can be defined by adding parameter **maximum**. For example:

- `.addParameter('maximum', 60)`

Result examples:

- "Hi how are you" - maximum: 9 - "Hi how"
- "Ing. Mgr. at Mgr. MBA" - maximum: 20 - "Ing. Mgr. at Mgr."
- "Ing." - maximum: 20 - "Ing."

Copying assigned roles from the main contract to a contract valid in the future

This feature allows you to automatically copy roles from the existing main contract to a contract valid in the future. This serves to handle the issue with cases when contract validity is not extended but rather a new contract is created altogether.

Since module version **3.3.0**

Documentation is available [there](#)

Deduplication for automatic roles by organization structure

The feature process deduplication existing automatic roles that are assigned onto organization units.

Since module version **3.4.0**

Complete documentation is available [there](#).

Check whether a value is unique in a system

This feature enables you to check that a certain value is unique in a system. This is typically useful for generators creating emails or usernames which need to be unique.

Since module version **3.5.0**

Complete documentation is available [here](#).

Notify user of account creation

This feature enables users to receive notifications about account creation on a particular system or systems.

Since module version **3.6.0**

Complete documentation is available [there](#).

Create automatic role for organization group

This feature creates automatic organization role for group.

Since module version **3.7.0**

Complete documentation is available [there](#).

Notification about roles without guarantees

This feature sends notification about roles with expiring/expired/missing role guarantees.

Since module version **3.8.0**

Complete documentation is available [there](#).

Resize profile photo to defined size

This feature allows you to resize user profile photos to a defined size.

Since module version **3.9.0**

Complete documentation is available [there](#).

Evaluator (permissions) for users with given form projection who do not have contract in denied organization

Since module version **4.3.0**.

Documentation is available [there](#).

Recalculate identities with roles

Since module version **4.5.0**.

Documentation is available [there](#).

Send custom notification to identities

Since module version **4.5.0**.

Documentation is available [there](#).

Notify about identity changes caused by synchronization

Since module version **3.11.0** .

Documentation is available [there](#).

Approval of externs created from Hub

Since module version **4.5.1**.

Documentation is available [there](#).

Missing related entity permissions report

Since module version **2.13.0, 3.12.0, 4.6.0**.

Documentation is available [there](#).

Virtual system import LRT

Documentation is available here: [Systems - Import of data from CSV](#)

Automatic role definitions - Import of data from CSV LRT

Documentation is available here: [Automatic role definitions - Import of data from CSV](#)

Identity temporary forced activation

Since module version **5.4.0**.

Documentation is available here: [Identity temporary forced activation \(since 5.4.0\)](#)

Assign roles to accounts - Import of data from CSV

Since module version **5.5.0**.

Documentation is available here: [Assign roles to accounts - Import of data from CSV \(since 5.5.0\)](#)

From:

<https://wiki.czechidm.com/> - IdStory Identity Manager

Permanent link:

https://wiki.czechidm.com/devel/documentation/modules_extras

Last update: **2025/03/04 00:40**

