

Access Restrictions in the Expert Mode

Introduction to Expert Rules

In the tabs "Visibility" and "Access", you can configure additional preferences for many course elements. For instance, you can block a course element for learners, grant access only for certain groups or unlock it depending on the date.

In case of more complicated visibility or access rules you can use the expert mode, thus enabling you to configure visibility and access of course elements as required. You can e.g. limit access to a course element to specific user names, link several types of restrictions to each other or work with relative date values. The following example illustrates this:

Sie wollen einen Kursfragebogen erst in der letzter letzten Kurswoche frei schalten, möchten diese Option aber schon mal einrichten, damit Sie es später nicht vergessen.

Sie schalten also den Kursbaustein „Fragebogen“ datumsabhängig frei, damit Sie sich im Kursverlauf nicht mehr darum kümmern müssen. In den Tabs *Sichtbarkeit* und *Zugang* des „Fragebogens“ können Sie hierfür im einfachen Modus das Anfangs- und Enddatum eingeben. Sie können Ihren Fragebogen auch nur für eine bestimmte Teilnehmergruppe zugänglich machen. Wählen Sie hierfür unter Sichtbarkeit bzw. Zugang ergänzend "Gruppenabhängig". So könnten Sie z.B. bei institutionsübergreifenden Online-Kursen zwei unterschiedliche Fragebögen verwenden. Voraussetzung ist lediglich, dass Sie die Kursteilnehmenden in (zwei) unterschiedliche Gruppen geteilt haben die sie nun zuweisen können.

Ein etwas komplexeres Beispiel mit genauer Angabe der Expertenregel finden Sie [hier](#).

Expert rules primarily serve to save you time and work or just simplify it. Therefore, it is worth the effort to take a closer look at them. Just like any language, expert rules follow a syntax. OpenOLAT will indicate an error should you make a syntactic mistake. This is very helpful, especially at the beginning if one does not have any or just little programming skills. Expert rules verify if a certain attribute is TRUE or FALSE.

As an introduction to the syntax of expert rules, you should at first define a rule in the simple mode. For example, you may generate a "single page", and click on "Blocked for learners" in the "Access" tab.

Then click on "Display expert mode" and see your first expert rule:

```
( ( isCourseCoach(0) | isCourseAdministrator(0) ) )
```

The whole term is enclosed in double brackets. The two outer brackets can be omitted in this case. Just try it out. The vertical line in the center "|" is the Boolean operator OR and connects the course coach with the course administrator. Both of them have exclusive access to the "single page".

Now change the Boolean Operator into "&":

```
isCourseCoach(0) & isCourseAdministrator(0)
```

This rule grants access exclusively to those course coaches who are also course administrators. This preference is only possible in the expert mode.

You can try out any number of scenarios and insert further attributes and operators. In this chapter you will find further attributes and examples illustrating their meaning to help you become more acquainted with expert rules.

Configuration of Expert Rules

Expert rules certify if there is an attribute with a specific value.

Attribute	Description	Example Expert rule
isGuest	accessible only for guests	isGuest(0)
isCourseCoach	available only for coach	isCourseCoach(0)
isUser	available only for one specific user	isUser("pmuster")

Working with the Constants "TRUE" and "FALSE"

By using the constants "true" and "false", the existence ("true" = "1") or non-existence ("false" = "0") of an attribute can be verified. In this case, we refer to a so-called Boolean Variable (named after George Boole, the father of the Boolean Algebra). These variables can only take a limited number of values or states. In our specific case, the variable can only take the two values ("true" = "1" = existing or "false" = "0" = non-existing).

To give a practical example in our OLAT context, we will use a simple expert rule for managing the access to a course:

Case 1: Only guest-users get access to the course. The respective user therefore only gets access if the attribute "isGuest" is true. There are three alternatives for this expert rule:

isGuest(0) oder isGuest(0)=1 oder isGuest(0)=true

Case 2: In this case we want guest-users not to have access. The respective user therefore only gets access if the attribute "isGuest" is false. There are two alternatives for this expert rule:

isGuest(0)=0 oder isGuest(0)=false

An extensive list of all relevant components needed for applying expert rules can be found in the following box.

Type	Syntax	Meaning
Constants	<i>TRUE</i> or <i>1</i>	True
	<i>FALSE</i> or <i>0</i>	False
	<i>ANY_COURSE</i>	Query should be applied to every course (only for isCourseAdministrator(), isCourseCoach(), isCourseParticipant())
Variable	<i>now</i>	Actual time of server system
Functions	<i>date("[date]")</i>	Retrieve date
	<i>inLearningGroup("[string]")</i>	Generates TRUE for all members of the learning group [string]
	<i>inRightGroup("[string]")</i>	Generates TRUE for all group members with the same rights [string]
	<i>isLearningGroupFull("[string]")</i>	Generates the boolean TRUE (= full) or FALSE (= vacancies) for the learning group indicated.
	<i>isUser("[string]")</i>	Results in TRUE for users with the user name [string]
	<i>inLearningArea("[string]")</i>	Generates TRUE for all group members in the learning area [string]
	<i>isGlobalAuthor(0)</i>	Generates TRUE for all members of the OLAT author group
	<i>isCourseAdministrator(0)</i>	Generates TRUE for all owners of a course (learning resource)
	<i>isCourseAdministrator(ANY_COURSE)</i>	Generates TRUE for all users which have owner rights on at least one course on the system
	<i>isCourseCoach(0)</i>	Generates TRUE for all users supervising a learning group or are supervising the course
	<i>isCourseCoach(ANY_COURSE)</i>	Generates TRUE for all users supervising at least one learning group of a course or are supervising at least one course on the system
	<i>isCourseParticipant(0)</i>	Generates TRUE for all participants of this course
	<i>isCourseParticipant(ANY_COURSE)</i>	Generates TRUE for all users on the system that participate in at least one course
	<i>isGuest(0)</i>	Generates TRUE for all users visiting OLAT as guests
	<i>hasAttribute("[AttributeName]", [string])</i>	Generates TRUE, if [string] corresponds to the relevant user's value of the AAI attribute [AttributeName] .
	<i>isInAttribute("[AttributeName]", [substring])</i>	Generates TRUE, if [substring] corresponds to part of the relevant user's value of the AAI attribute [AttributeName]. General information on AAI AAI attributes Specification of AAI attributes (pdf file)
	<i>getUserProperty("userPropertyName")</i>	Generates the value of a specific user attribute. By means of "=" this value can be compared to another fixed value.
	<i>getPassed("[integer]")</i>	Generates the Boolean TRUE (=Passed) or FALSE (=Failed) from a course element with specified ID

	<i>getScore</i> ("[integer] ")	Generates the score from a course element with specified ID
	<i>getAttempts</i> ("[integer] ")	Generates the number of completed attempts from a course element with specified ID. Can be applied to course elements of the type <i>Test</i> , <i>Self-test</i> , <i>Questionnaire</i> (possible return values 0 or 1) and <i>Task</i> (return value = number of files handed in).
	<i>getLastAttemptDate</i> ("[integer] ")	Generates the date of the last attempt from a course element with the specified ID. Can be applied like the <i>getAttempts</i> method.
	<i>getInitialEnrollmentDate</i> ("[integer] ")	Generates the date of the first registration of the relevant course participant from the course element <i>Enrollment</i> with specified ID.
	<i>getRecentEnrollmentDate</i> ("[integer] ")	Generates the date of the last registration of the relevant course participant from the course element <i>Enrollment</i> with specified ID.
	<i>getInitialCourseLaunchDate</i> (0)	Generates the date of a course participant's first course attendance.
	<i>getRecentCourseLaunchDate</i> (0)	Generates the date of a course participant's last course attendance.
	<i>getPassedWithCourseId</i> ("[integer-1] ", "[integer-2] ")	Generates the Boolean TRUE (=Passed) or FALSE (=Failed) from the course element ID=[integer-2] of the course ID=[integer-1]
	<i>getScoreWithCourseId</i> ("[integer-1] ", "[integer-2] ")	Generates the score from the course element ID=[integer-2] of the course ID=[integer-1]
	<i>hasUserProperty</i> ("[userPropertyname]", "[string] ")	Generates TRUE, if [string] corresponds to the relevant user's value of the userproperty [userPropertyname].
	<i>userPropertyStartswith</i> ("[userPropertyname]", "[substring] ")	Generates TRUE, if the userproperty [userPropertyname] starts with [substring].
	<i>userPropertyEndswith</i> ("[userPropertyname]", "[substring] ")	Generates TRUE, if the userproperty [userPropertyname] ends with [substring].
	<i>isInUserProperty</i> ("[userPropertyname]", "[substring] ")	Generates TRUE, if [substring] corresponds to part of the relevant user's value of the userproperty [userPropertyname].
	<i>isNotInUserProperty</i> ("[userPropertyname]", "[substring] ")	Generates TRUE, if [substring] does not show up in the value of the userproperty [userPropertyname].
	<i>hasNotUserProperty</i> ("[userPropertyname]", "[string] ")	Generates TRUE, if [string] does not correspond to the relevant user's value of the userproperty [userPropertyname].
Units	<i>min</i>	Minutes
	<i>h</i>	Hours
	<i>d</i>	Days
	<i>w</i>	Weeks
	<i>m</i>	Months
Operators	=	equal
	>	greater than
	<	less than
	>=	greater/equal
	<=	less/equal
	*	Multiplication
	/	Division
	+	Addition
	-	Subtraction

Booleans	&	Logical AND
		Logical OR

Various expert rules require the use of user attributes in order to filter eligible users for access content. Those rules enable authors to limit access rights depending on name, gender, address, field of studies and so on. Those user attributes are usually visible in the user profile.

OpenOLAT provides standardized terms for those attributes. The following expert rules require the use of user attributes:

- `getUserProperty("[userPropertyname]")`
- `hasUserProperty("[userPropertyname]", "[string]")`
- `userPropertyStartswith("[userPropertyname]", "[substring]")`
- `userPropertyEndswith("[userPropertyname]", "[substring]")`
- `isInUserProperty("[userPropertyname]", "[substring]")`
- `isNotInUserProperty("[userPropertyname]", "[substring]")`
- `hasNotUserProperty("[userPropertyname]", "[string]")`

The following user attributes are available in OpenOLAT. Please note that access restrictions using user attributes can only be successful if those user attributes are used and generally filled in throughout your system. Simply check your user profile in the the personal menu in Configuration /Profile for available user attributes. For questions, please contact your system administrator.

Benutzerdaten		Kontaktdaten		Adressdaten	
userName	User name	telPrivate	Phone number private	street	Street
firstName	First name	telMobile	Phone number mobile	extendedAddress	Extra address line
lastName	Last name	telOffice	Phone number office	poBox	P.O.Box
email	E-mail address	skype	Skype ID	zipCode	Zip code
creationDateDisplayProperty	User creation date	xing	Xing	region	Region / Canton
lastloginDateDisplayProperty	User last login	homepage	Homepage	city	City
birthDay	Date of birth			country	Country
gender	Gender			countryCode	Country code
Organisation		Berufliche Kontaktdaten		Verschiedenes	
institutionalName	Institution	department	Department / Company	typeOfUser	Type of user
institutionalUserIdentifier	Institution identifier (registration number)	officeStreet	Address / P.O. box	rank	Service grade / employment title
institutionalEmail	Institutional e-mail	extendedOfficeAddress	Extended office address	socialSecurityNumber	Social security number
orgUnit	Organizational unit / study group	officeZipCode	Office ZIP	degree	Academic degree
studySubject	Field of studies	officeCity	Office city	position	Role / position
graduation	Graduation year	officeCountry	Office country	userInterests	Expertise
		officeMobilePhone	Office mobile phone		

Examples on how to apply "getUserProperty":

- Only course participants of a specific field of study should be granted access:

```
getUserProperty("studySubject") = "Mechanical Engineering"
```

Now anybody who needs access must first complete the field "field of study" in their profile and state it as "Mechanical Engineering".

- The other way round, should you intend to grant access only to those who have not stated their field of study in their profile, you can express the corresponding rule as follows:

```
getUserProperty("studySubject") = ""
```
- Should you want to grant access only course participants who have completed the field of study in their profile (no matter what the study subjects are), the rule can be defined as follows:

```
getUserProperty("studySubject") = "" = false
```

or

```
getUserProperty("studySubject") = "" = 0
```

There are various options to interrelate single rules to each other. The two most important operators to combine attributes are:

- AND conjunction: &
- OR conjunction: |

Please note that an OR conjunction precedes an AND conjunction. In order to handle an AND conjunction first you have to use brackets.

Example: The expert rule (`inGroup("Participants IntensiveCourse") | isCourseCoach(0)`) means that either participants of an intensive course or all coaches of groups will have access to a course element.

Some examples are listed below in order to show you how to use the expert syntax.

`inLearningGroup("Amateur") = 0`

With the exception of the group «*Amateur*» this course element is visible for all participants.

`(now >= date("22.03.2018 12:00")) & (now <= date("23.08.2018 18:00")) | inLearningGroup("Tutor")`

This course element is visible for all participants between 22-3-2018 and 23-8-2018. For members of the learning group «*Tutor*» it is always visible.

`(now >= date("03.09.2018 00:00")) & (now <= date("13.10.2018 00:00")) & inRightGroup("Assessors") | isUser("Author")`

This course element is visible for all participants of the right group «*Assessors*» between 3-9-2018 and 13-10-2018. For the person with the user name «*Author*» it is always visible.

`hasAttribute("swissEduPersonStudyBranch3", "6200")`

Only students of human medicine have access to this course element.

See also:

[AAI attributes](#)

[Specification of AAI attributes \(pdf file\)](#)

`hasAttribute("swissEduPersonHomeOrganization", "uzh.ch")`

Only students of the University of Zurich have access to this course element.

See also:

[AAI attributes](#)

[Specification of AAI attributes \(pdf file\)](#)

`isInAttribute("surname", "Mue")`

Generates TRUE for all persons whose attribute *surname* contains the letter sequence "Mue". E.g. gives TRUE for the value "Mueller" or "Muehlebacher"

See also:

[AAI attributes](#)

[Specification of AAI attributes \(pdf file\)](#)

`isInAttribute("eduPersonEntitlement", "http://vam.uzh.ch")`

Generates TRUE for all persons whose attribute *eduPersonEntitlement* contains the value "http://vam.uzh.ch". E.g. gives TRUE for the value "http://vam.uzh.ch/surgery"

See also:

[AAI attributes](#)

[Specification of AAI attributes \(pdf file\)](#)

`(getUserProperty("orgUnit") = "Sales")`

Checks if a person is part of the organizational unit 'Sales.' This can be useful if e.g. data are automatically transferred from LDAP.

`(getPassed("69742969114730") | getPassed("69742969115733") | getPassed("69742969118009")) * 10`

This rule is set in the tab «*Score*» -> «*Processing score*» of the course element *Structure*. The course element *Structure* shows 10 points if one of the tests (course element IDs "69742969114730", "69742969115733" or "69742969118009") was passed. Otherwise 0 points.

`(getScore("69742969114730") + getScore("69742969115733") + getScore("69742969118009")) >= 140 | getPassed("69978845384688")`

This rule is set in the tab «*Score*» -> «*Passed if*» of the course element *Structure*. The course element *Structure* shows «*Passed*», if a minimum of 140 points in all tests is achieved or if «*Passed*» is entered manually. (Element *Assessment* with ID "69978845384688").

`getAttempts("70323786958847") > 0`

Generates TRUE, as soon as the relevant course participant has completed the test with specified ID for the first time.

`getAttempts("70323524635734") <= 3`

Generates FALSE, as soon as the relevant course participant has put more than 3 files into the storage folder of the course element *Task* (*deprecated*).

`getLastAttemptDate("70323524635734") + 24h < now`

Generates TRUE when the last test attempt is older than 24 hours

getInitialEnrollmentDate("70323786958847") <= date("26.5.2005 18:00")

Generates TRUE for those participants who enrolled in an available group before 6 p.m. on May 26th, 2005, by means of the course element *Enrollment* with specified ID.

getInitialEnrollmentDate("70323786958847") + 2h > now

Generates TRUE within two hours starting at the moment of registration for those participants who have enrolled in an available group by means of the course element *Enrollment* with specified ID. This way it is clear that every participant can only work on e.g. a script within a particular time frame.

(getInitialCourseLaunchDate(0) >= never) | (getInitialCourseLaunchDate(0) + 2h > now)

Generates TRUE if a course participant has not yet taken any courses or during the first two hours after taking a course. This way it is possible represent that each course participant can only see courses for a certain period of time.

(getRecentCourseLaunchDate(0) + 10min < now)

Generates TRUE if a user is active for more than 10 min within a course.

(getCourseBeginDate(0) <= today) & (getCourseEndDate(0) >= today)

Returns the value TRUE if today's date lies in between the start and end date of the execution period.

isAssessmentMode(0)

Returns the value TRUE if the course is within an assessment.

hasUserProperty("email", "john.doe@openolat.org")

Generates TRUE, if the course participant is registered in OpenOLAT with the listed e-mail address.

userPropertyEndswith("email", "@openolat.org")

Generates TRUE, if the e-mail address of the course participant ends with *@openolat.org*.

isInUserProperty("email", "doe@openo")

Generates TRUE, if the term *doe@openo* is a part of the e-mail address of the course participant.

isNotInUserProperty("email", "doe@openo")

Generates FALSE, if the term *doe@openo* is a part of the e-mail address of the course participant.



Please note that the IDs of the course elements mentioned above are only examples. To create your course, you have to make reference to the relevant numbers available on the first tab «*Title and description*» of the favored course element.

Use of AAI Attributes

If you are enrolled at swiss academia or any other institution with access to an AAI infrastructure, by means of AAI attributes you can set access rules within a course to make sure that only course participants with specific user attributes (e.g. members of a certain organization) will have access to your course material. AAI means "Authentication and Authorization Infrastructure" and allows university members to use systems of other participating institutions with only one user name and password. For further information on AAI please go to e.g. [Switch](#) or to [Deutsches Forschungsnetz](#).

Available attributes and possible values are described in the AAI Attribute Specification on the [Switch](#) and the [DFN-AAI](#) site (in german). The two most common attributes at swiss universities can be found in the following table along with examples of their corresponding expert rules:

Attribute	Description	Example Expert rule and Explication
swissEduPerson-HomeOrganization	University or home organization	hasAttribute ("swissEduPersonHomeOrganization", "uzh.ch"): only members of the Zurich University will get access.
swissEduStudyBranch3	Field of study, 3rd classification	hasAttribute ("swissEduPersonStudyBranch3", "6400"): only veterinary medicine students will have access.

Utilization

You can retrieve AAI attributes by using the syntax **hasAttribute("[AttrName]", "[string]")** or **isInAttribute("[AttrName]", "[substring]")**.

Where:

[AttrName]	is the attribute name you can find in the following table and also in the Specification of AAI attributes (pdf file) (column <i>LDAP names</i>) on page 5.
[string]	is the value of the AAI attribute with the name [AttrName].
[substring]	is any part of [string] .

AAI retrievals for example: John Doe

Variable You can retrieve AAI attributes by using the syntax <i>hasAttribute("["AttrName]"; ["string] ")</i> or <i>isInAttribute("["AttrName]"; ["substring] ")</i> .	Example value [string]	Description
swissEduPersonUniqueID	845938727494@uzh.ch	Unambiguous personal identification number
surname	Doe	Last name
givenName	John	First name
mail	john.doe@uzh.ch	Preferred e-mail address
swissEduPersonHomeOrganization	uzh.ch	Home organisation/university
swissEduPersonHomeOrganizationType	university	Type of home organisation
eduPersonAffiliation	student	Position within this organisation
swissEduPersonStudyBranch1	4	Field of study 1st classification
swissEduPersonStudyBranch2	42 (=Natural sciences)	Field of study 2nd classification
swissEduPersonStudyBranch3	4600 (=Chemistry)	Field of study 3rd classification
swissEduPersonStudyLevel	15	Description of study level
eduPersonEntitlement	http://vam.uzh.ch/surgery	Access right to resource
employeeNumber	01-234-567	Registration number (only for students at Zurich university)
organizationalUnit	1	Unity of home organisation e.g. faculty (only for employees)

isInAttribute("surname","ust")	true
hasAttribute("swissEduPersonStudyBranch3","4600")	true
hasAttribute("swissEduPersonStudyBranch3","1200")	false
isInAttribute("eduPersonEntitlement"," http://vam.uzh.ch ")	true
isInAttribute("eduPersonEntitlement"," http://vam.uzh.ch/ophthalmology ")	false
hasAttribute("employeeNumber","01-234-567")	true

You will find the link to a list of possible attribute values in the Specification of AAI attributes (pdf file) appendix, as of page 20. [Specification of AAI attributes \(pdf file\)](#)

For further information on attribute values or the application of AAI attributes in Switzerland please go to [Switch](#), and for Germany go to [Deutsches Forschungsnetz](#).



Only use the AAI attributes if you are sure that all participants of your course are dialing in via an AAI structure. Otherwise the parameters do not apply!