



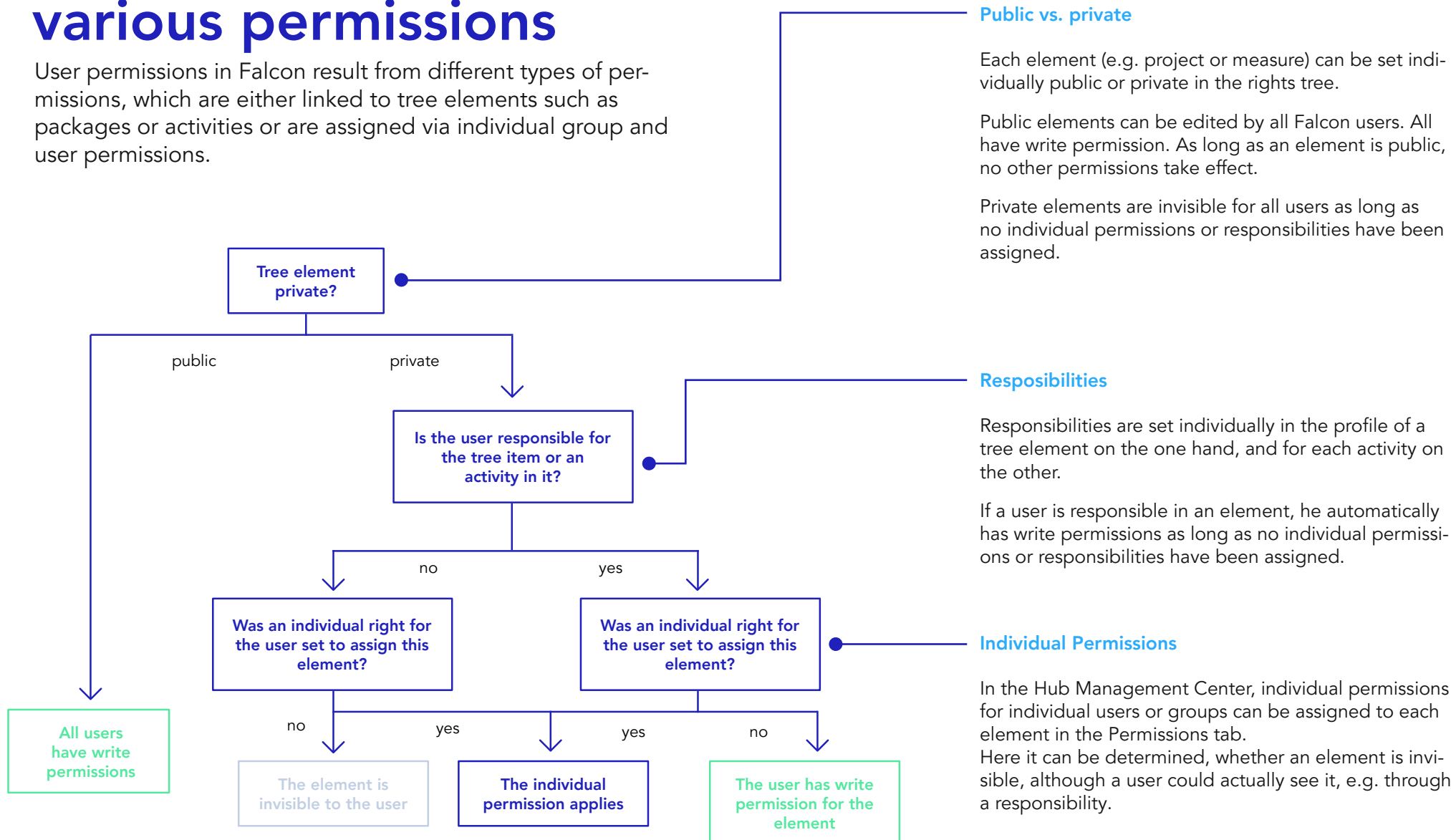
Falcon-Permissions explained



Falcon makes it possible for you to adapt the permission structure specifically to your project. We will explain in detail the logic on which Falcon's sophisticated permission system is based.

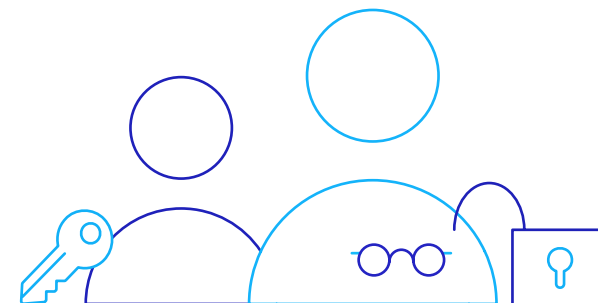
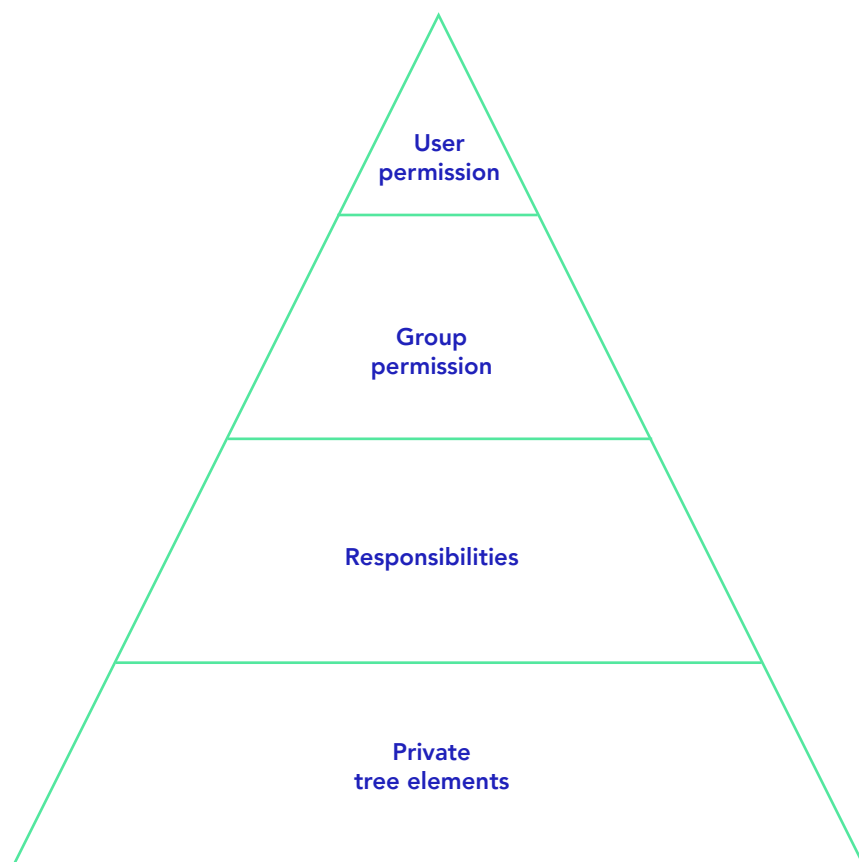
Overview of the various permissions

User permissions in Falcon result from different types of permissions, which are either linked to tree elements such as packages or activities or are assigned via individual group and user permissions.



Permissions hierarchy

The different types of permissions in Falcon allow for extremely precise permissions assignment. You can find out here how the different types of permissions work together:



User permission

The user permission is the strongest permission in Falcon and overrides all other permissions that are created by group rights or responsibilities.

Group permission

The group permission overwrites permissions from responsibilities, but can be overridden by user permissions.

Responsibilities

Responsibilities give the responsible users write permissions, even if the tree item is protected and no individual permissions have been assigned. Write permissions resulting from responsibilities can be overwritten by individual permissions.

Private tree elements

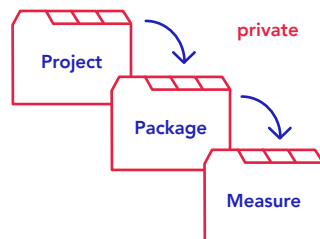
User and group permissions, as well as write permissions from responsibilities only apply to private tree items. In public elements, every user has write permission, no matter which additional permissions have been assigned.

Permissions are passed on

Permissions are passed on through the complete project tree. This applies to the „public / private“ setting of the tree items, as well as write permissions from responsibilities and individual group and user permissions. The different types of permissions and their hierarchies among each other allow you to assign user permissions that fit exactly to your project. The following examples will show you how this sophisticated permission assignment works.

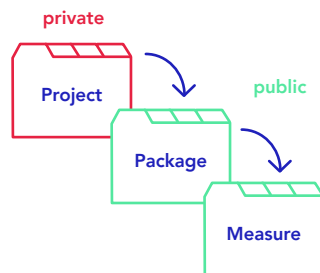
Private element is passed on

The project is protected. By default, the subordinate elements (packages and measures) are also protected.



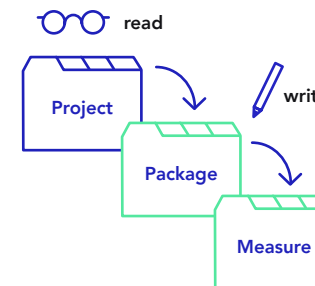
Child elements can also be passed on unprivate

The project is private. The package is made public. The setting „public“ is passed on to the measure.



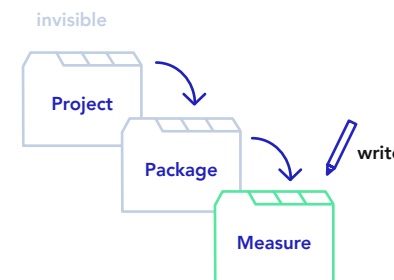
User permission beats group permission

The group was granted read-only permission for the project. The group right is not passed on, because it is overwritten by the user permission on package level.



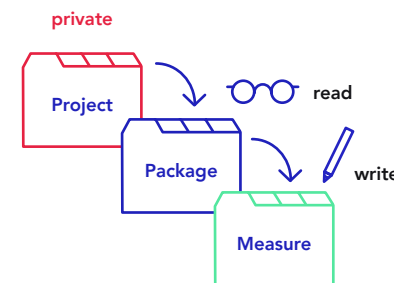
Combined user permissions

The project was made invisible to the user. The setting „invisible“ is passed on to the package. The write permission for the measure overwrites the passed on „invisible“ setting.

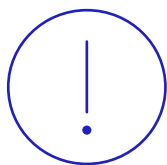


Combined permission types

The project is „private“. At package level, the group has read permission. On measure level, the user additionally receives write permission.



Tips and tricks



The admin permission

The admin permission beats all other individual permissions given to the user in the admin element and in the subordinate tree elements.



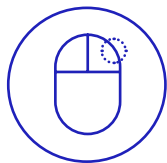
Detailed permissions overview

[HERE](#) you can download a detailed overview, which lists which actions can be performed with the different permissions.



Setting permissions in Falcon

If you would like to learn how to make the necessary settings in Falcon, [follow this article](#).



Check user permissions

In the Permissions tab of the hub administration you can check the user permissions with a right click. This allows you to check quickly and easily whether you grant the selected user the required permissions.

