ParentOwnerLimitation

A limitation to specify that only users that owns all parent locations of content gets a certain access right, used for `content/create` permission.

Identifier	ParentOwner
Value Class	eZ\Publish\API\Repository\Values\User\Limitation\ParentOwnerLimitation
Type Class	eZ\Publish\Core\Limitation\ParentOwnerLimitationType
Criterion used	n/a
Role Limitation	no

Possible values

Value	UI value	Description
1	"self"	Only user which is owner of all parent locations gets access (see "*Legacy compatibility notes*" #2)
2	"session"	Same as "self" (see "*Legacy compatibility notes*" #1)

Legacy compatibility notes:

1. Owner("session") is deprecated and works exactly like "self" in Public API since it has no knowledge of user Sessions
2. User is no longer auto assumed to be owner of himself and get access create children of own user content object when `Owner` limitation is used in Public API*

* Workaround for `ParentOwner` limitation on user object (when user object is the parent of the content being created):

To make sure user gets access to himself when using (Parent)Owner limitation across 4.x and 5.x, the solution is to change the user to be owner of himself.
This is accomplished using a privileged user to do the following API calls:

```
$user = $userService->loadUser( $userId );
$contentMetadataUpdateStruct = $contentService->newContentMetadataUpdateStruct();
$contentMetadataUpdateStruct->ownerId = $user->id;
$contentService->updateContentMetadata( $user->getVersionInfo()->getContentInfo(),
$contentMetadataUpdateStruct );
```