Signals Reference

This page references **all available signals** that you can listen to, triggered by ("Public") Repository API in eZ Platform.

> For more information, check the SignalSlot section and the Listening to Core events recipe.

> All signals are relative to `eZ\Publish\Core\SignalSlot\Signal` namespace.

> **Transactions**
> Signals are sent after transactions are executed, making signals transaction safe.

ContentService

Signal type	Properties	Triggered by	When
`ContentService\AddRelationSignal`	`srcContentId` (source contentId, aka referrer) `srcVersionNo` `dstContentId` (destination contentId, aka target)	`ContentService::addRelation()`	After
`ContentService\AddTranslationInfoSignal`	N/A	`ContentService::addTranslationInfo()`	After
`ContentService\CopyContentSignal`	`srcContentId` (original content ID) `srcVersionNo` `dstContentId` (contentId of the copy) `dstVersionNo` `dstParentLocationId` (locationId where the content has been copied)	`ContentService::copyContent()`	After
`ContentService\CreateContentDraftSignal`	`contentId` `versionNo` `userId` (Id of user used to create the draft, or null - current user)	`ContentService::createContentDraft()`	After
`ContentService\CreateContentSignal`	`contentId` `versionNo`	`ContentService::createContent()`	After
`ContentService\DeleteContentSignal`	`contentId`	`ContentService::deleteContent()`	After
`ContentService\DeleteRelationSignal`	`srcContentId` `srcVersionNo` `dstContentId`	`ContentService::deleteRelation()`	After
`ContentService\DeleteVersionSignal`	`contentId` `versionNo`	`ContentService::deleteVersion()`	After
`ContentService\PublishVersionSignal`	`contentId` `versionNo`	`ContentService::publishVersion()`	After
`ContentService\TranslateVersionSignal`	`contentId` `versionNo` `userId`	`ContentService::translationVersion()`	After

Signal type	Properties	Triggered by	When
`ContentService\Upd ateContentMetadata Signal`	• contentId	`ContentService::up dateContentMetadat a()`	After
`ContentService\Upd ateContentSignal`	• contentId • versionNo	`ContentService::up dateContent()`	After

ContentTypeService

Signal type	Properties	Triggered by	When
`ContentTypeService \AddFieldDefinitio nSignal`	• contentTypeDraftId	`ContentTypeService ::addFieldDefiniti on()`	After
`ContentTypeService \AssignContentType GroupSignal`	• contentTypeId • contentTypeGro upId	`ContentTypeService ::assignContentTyp eGroup()`	After
`ContentTypeService \CopyContentTypeSi gnal`	• contentTypeId • userId	`ContentTypeService ::copyContentType( )`	After
`ContentTypeService \CreateContentType DraftSignal`	• contentTypeId	`ContentTypeService ::createContentTyp eDraft()`	After
`ContentTypeService \CreateContentType GroupSignal`	• groupId	`ContentTypeService ::createContentTyp eGroup()`	After
`ContentTypeService \CreateContentType Signal`	• contentTypeId	`ContentTypeService ::createContentTyp e()`	After
`ContentTypeService \DeleteContentType GroupSignal`	• contentTypeGro upId	`ContentTypeService ::deleteContentTyp eGroup()`	After
`ContentTypeService \DeleteContentType Signal`	• contentTypeId	`ContentTypeService ::deleteContentTyp e()`	After
`ContentTypeService \PublishContentTyp eDraftSignal`	• contentTypeDra ftId	`ContentTypeService ::publishContentTy peDraft()`	After
`ContentTypeService \RemoveFieldDefini tionSignal`	• contentTypeDra ftId • fieldDefinitio nId	`ContentTypeService ::removeFieldDefin ition()`	After
`ContentTypeService \UnassignContentTy peGroupSignal`	• contentTypeId • contentTypeGro upId	`ContentTypeService ::unassignContentT ypeGroup()`	After
`ContentTypeService \UpdateContentType DraftSignal`	• contentTypeDra ftId	`ContentTypeService ::updateContentTyp eDraft()`	After
`ContentTypeService \UpdateContentType GroupSignal`	• contentTypeGro upId	`ContentTypeService ::updateContentTyp eGroup()`	After
`ContentTypeService \UpdateFieldDefini tionSignal`	• contentTypeDra ftId • fieldDefinitio nId	`ContentTypeService ::updateFieldDefin ition()`	After

LanguageService

Signal type	Properties	Triggered by	When
`LanguageService\Cr eateLanguageSignal`	• languageId	`LanguageService::c reateLanguage()`	After

Signal type	Properties	Triggered by	When
LanguageService\DeleteLanguageSignal	• languageId	LanguageService::deleteLanguage()	After
LanguageService\DisableLanguageSignal	• languageId	LanguageService::disableLanguage()	After
LanguageService\EnableLanguageSignal	• languageId	LanguageService::enableLanguage()	After
LanguageService\UpdateLanguageNameSignal	• languageId • newName	LanguageService::updateLanguageName()	After

LocationService

Signal type	Properties	Triggered by	When
LocationService\CopySubtreeSignal	• subtreeId (top locationId of the subtree to be copied) • targetParentLocationId	LocationService::copySubtree()	After
LocationService\CreateLocationSignal	• contentId • locationId	LocationService::createLocation()	After
LocationService\DeleteLocationSignal	• contentId • locationId	LocationService::deleteLocation()	After
LocationService\HideLocationSignal	• contentId • locationId	LocationService::hideLocation()	After
LocationService\UnhideLocationSignal	• contentId • locationId	LocationService::unhideLocation()	After
LocationService\MoveSubtreeSignal	• subtreeId • newParentLocationId	LocationService::moveSubtree()	After
LocationService\SwapLocationSignal	• content1Id • location1Id • content2Id • location2Id	LocationService::swapLocation()	After
LocationService\UpdateLocationSignal	• contentId • locationId	LocationService::updateLocation()	After

ObjectStateService

Signal type	Properties	Triggered by	When
ObjectStateService\CreateObjectStateGroupSignal	• objectStateGroupId	ObjectStateService::createObjectStateGroup()	After
ObjectStateService\CreateObjectStateSignal	• objectStateGroupId • objectStateId	ObjectStateService::createObjectState()	After
ObjectStateService\DeleteObjectStateGroupSignal	• objectStateGroupId	ObjectStateService::deleteObjectStateGroup()	After
ObjectStateService\DeleteObjectStateSignal	• objectStateId	ObjectStateService::deleteObjectState()	After
ObjectStateService\SetContentStateSignal	• contentId • objectStateGroupId • objectStateId	ObjectStateService::setContentState()	After
ObjectStateService\SetPriorityOfObjectStateSignal	• objectStateId • priority	ObjectStateService::setPriorityOfObjectState()	After

Signal type	Properties	Triggered by	When
ObjectStateService\UpdateObjectStateGroupSignal	• objectStateGroupId	ObjectStateService::updateObjectStateGroup()	After
ObjectStateService\UpdateObjectStateSignal	• objectStateId	ObjectStateService::updateObjectState()	After

RoleService

Signal type	Properties	Triggered by	When
RoleService\AddPolicySignal	• roleId • policyId	RoleService::addPolicy()	After
RoleService\AssignRoleToUserGroupSignal	• roleId • userGroupId • roleLimitation	RoleService::assignRoleToUserGroup()	After
RoleService\AssignRoleToUserSignal	• roleId • userId • roleLimitation	RoleService::assignRoleToUser()	After
RoleService\CreateRoleSignal	• roleId	RoleService::createRole()	After
RoleService\DeleteRoleSignal	• roleId	RoleService::deleteRole()	After
RoleService\RemovePolicySignal	• roleId • policyId	RoleService::removePolicy()	After
RoleService\UnassignRoleFromUserGroupSignal	• roleId • userGroupId	RoleService::unassignRoleFromUserGroup()	After
RoleService\UnassignRoleFromUserSignal	• roleId • userId	RoleService::unassignRoleFromUser()	After
RoleService\UpdatePolicySignal	• policyId	RoleService::updatePolicy()	After
RoleService\UpdateRoleSignal	• roleId	RoleService::updateRole()	After

SectionService

Signal type	Properties	Triggered by	When
SectionService\AssignSectionSignal	• contentId • sectionId	SectionService::assignSection()	After
SectionService\CreateSectionSignal	• sectionId	SectionService::createSection()	After
SectionService\DeleteSectionSignal	• sectionId	SectionService::deleteSection()	After
SectionService\UpdateSectionSignal	• sectionId	SectionService::updateSection()	After

TrashService

Signal type	Properties	Triggered by	When
TrashService\DeleteTrashItemSignal	• trashItemId	TrashService::deleteTrashItem()	After
TrashService\EmptyTrashSignal	N/A	TrashService::emptyTrash()	After

Signal type	Properties	Triggered by	When
TrashService\RecoverSignal	• `trashItemId` • `newParentLocationId` • `newLocationId`	`TrashService::recover()`	After
TrashService\TrashSignal	• `locationId`	`TrashService::trash()`	After

URLAliasService

Signal type	Properties	Triggered by	When
URLAliasService\CreateGlobalUrlAliasSignal	• `urlAliasId`	`URLAliasService::createGlobalUrlAlias()`	After
URLAliasService\CreateUrlAliasSignal	• `urlAliasId`	`URLAliasService::createUrlAlias()`	After
URLAliasService\RemoveAliasesSignal	• `aliasList`	`URLAliasService::removeAliases()`	After

URLWildcardService

Signal type	Properties	Triggered by	When
URLWildcardService\CreateSignal	• `urlWildcardId`	`URLWildcardService::create()`	After
URLWildcardService\RemoveSignal	• `urlWildcardId`	`URLWildcardService::remove()`	After
URLWildcardService\TranslateSignal	• `url`	`URLWildcardService::translate()`	After

UserService

Signal type	Properties	Triggered by	When
UserService\AssignUserToUserGroupSignal	• `userId` • `userGroupId`	`UserService::assignUserToUserGroup()`	After
UserService\CreateUserGroupSignal	• `userGroupId`	`UserService::createUserGroup()`	After
UserService\CreateUserSignal	• `userId`	`UserService::createUser()`	After
UserService\DeleteUserGroupSignal	• `userGroupId`	`UserService::deleteUserGroup()`	After
UserService\DeleteUserSignal	• `userId`	`UserService::deleteUser()`	After
UserService\MoveUserGroupSignal	• `userGroupId` • `newParentId`	`UserService::moveUserGroup()`	After
UserService\UnAssignUserFromUserGroupSignal	• `userId` • `userGroupId`	`UserService::unAssignUserFromUserGroup()`	After
UserService\UpdateUserGroupSignal	• `userGroupId`	`UserService::updateUserGroup()`	After
UserService\UpdateUserSignal	• `userId`	`UserService::updateUser()`	After