

# Archival Storage Gateway API

No description provided (generated by Swagger Codegen <https://github.com/swagger-api/swagger-codegen>)

More information: <https://inqool.cz/>

Contact Info: [info@inqool.cz](mailto:info@inqool.cz)

Version: v1.1

GNU GPL v3

<http://apache.org/licenses/LICENSE-2.0.html>

## Access

1. HTTP Basic Authentication

## Methods

[ Jump to [Models](#) ]

## Table of Contents

### [AipApi](#)

- [DELETE /api/storage/{objId}](#)
- [DELETE /api/storage/{objId}/forget](#)
- [DELETE /api/storage/{aipId}/forgetXml/{xmlVersion}](#)
- [GET /api/storage/{aipId}](#)
- [POST /api/storage/{aipId}/files-specified](#)
- [POST /api/storage/{aipId}/files-reduced](#)
- [GET /api/storage/{aipId}/info](#)
- [POST /api/storage/{aipId}/aip-with-files-specified](#)
- [POST /api/storage/{aipId}/aip-with-files-reduced](#)
- [GET /api/storage/{aipId}/state](#)
- [GET /api/storage/{aipId}/xml](#)
- [GET /api/storage/{aipId}/xml/{xmlVersion}/state](#)
- [PUT /api/storage/{objId}/remove](#)
- [PUT /api/storage/{objId}/renew](#)
- [DELETE /api/storage/{objId}/rollback](#)
- [DELETE /api/storage/{aipId}/rollbackXml/{xmlVersion}](#)
- [POST /api/storage/save](#)
- [POST /api/storage/{aipId}/update](#)

### [StorageAdministrationApi](#)

- [POST /api/administration/storage](#)
- [POST /api/administration/storage/check\\_reachability](#)
- [POST /api/administration/storage/sync/{id}](#)
- [DELETE /api/administration/storage/{id}](#)
- [GET /api/administration/storage](#)
- [GET /api/administration/storage/basic](#)
- [GET /api/administration/storage/{id}](#)
- [GET /api/administration/storage/{id}/state](#)
- [GET /api/administration/storage/sync/{id}](#)
- [POST /api/administration/storage/update](#)

### [SystemAdministrationApi](#)

- [GET /api/administration/backup](#)
- [POST /api/administration/cleanup](#)

- [POST /api/administration/cleanup/object/{objId}](#).
- [GET /api/administration/config](#)
- [POST /api/administration/recover\\_db](#)
- [POST /api/administration/config](#)

## User

- [POST /api/user/activate/{dataSpace}](#).
- [POST /api/user/register](#)

## AipApi

[Up](#)

DELETE /api/storage/{objId}

Physically deletes object and sets its state to DELETED. (**delete**)

Not allowed for AIP XML objects.

### Path parameters

**object id (required)**  
*Path Parameter* –

### Responses

**200**

object successfully deleted

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

operation forbidden with respect to the current object state

**500**

no logical storage attached, or other internal server error

**503**

some attached logical storage is currently not reachable or system is in readonly state

[Up](#)

DELETE /api/storage/{objId}/forget

Forgets the object. (**forget**)

If the object is AIP, it has to contain exactly one AIP XML which is also forget. All Metadata and DB records are deleted together with file content. If the object to forget does not exist, then do nothing.

### Path parameters

**object id (required)**  
*Path Parameter* –

### Responses

**200**

object successfully forgotten or forget call skipped because the object was not found

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

forget feature not allowed or trying to forget AIP XML (there is a special endpoint for that case) OR trying to forget AIP which has more than one AIP XML linked

**500**

no logical storage attached, or other internal server error

**503**

some attached logical storage is currently not reachable or system is in readonly state

[Up](#)

DELETE /api/storage/{aipId}/forgetXml/{xmlVersion}

Deletes the latest AIP XML. Version set in the path variable must be the latest AIP XML of the AIP. (**forgetXml**)

#### Path parameters

**AIP id (required)**

*Path Parameter* –

**XML version (required)**

*Path Parameter* – format: int32

#### Responses

**200**

object successfully forgotten or forget call skipped because the object was not found

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

forget feature not allowed or the XML is version 1 (whole AIP should be forgotten instead) or if this is not the latest XML version of the AIP (newer versions exist)

**500**

no logical storage attached, or other internal server error

**503**

some attached logical storage is currently not reachable or system is in readonly state

[Up](#)

GET /api/storage/{aipId}

Return specified AIP as a ZIP package (**getAip**)

#### Path parameters

**AIP ID (required)**

*Path Parameter* –

#### Query parameters

true to return all XMLs, otherwise only the latest is returned  
(optional)

*Query Parameter* – default: false

#### Responses

**200**

AIP successfully returned

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

operation forbidden with respect to the current AIP state

**500**

file is corrupted at all storages, no logical storage attached, or other internal server error

**503**

all attached logical storages are currently unreachable

[Up](#)

POST /api/storage/{aipId}/files-specified

Return specified files of AIP packed in ZIP (**getAipFilesReducedByListOfPaths**)

DOES NOT validate the AIP.. does not return any AIP XML, requires some local storage to be reachable

### Path parameters

**AIP ID (required)**

*Path Parameter* –

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

**body [string](#) (required)**

*Body Parameter* –

### Responses

**200**

AIP successfully returned

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

operation forbidden with respect to the current AIP state

**500**

file is corrupted at all storages, no logical storage attached, or other internal server error

**503**

all attached logical storages are currently unreachable

[Up](#)

## POST /api/storage/{aipId}/files-reduced

Return specified files of AIP packed in ZIP (**getAipFilesReducedByRegex**)

DOES NOT validate the AIP, does not return any AIP XML, requires some local storage to be reachable

### Path parameters

**AIP ID (required)**

*Path Parameter* –

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

**body [DataReduction](#) (required)**

*Body Parameter* –

### Responses

**200**

AIP successfully returned

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

operation forbidden with respect to the current AIP state

500

file is corrupted at all storages, no logical storage attached, or other internal server error

503

all attached logical storages are currently unreachable



## GET /api/storage/{aipId}/info

Verifies AIP consistency at given storage and retrieves result. (**getAipInfo**)

If the AIP is not in some final, consistent state even in DB the storage is not checked and only DB data are returned.

### Path parameters

**AIP id (required)**

*Path Parameter* –

### Query parameters

**id of the logical storage (required)**

*Query Parameter* –

### Return type

[AipConsistencyVerificationResultDto](#)

### Example data

Content-Type: application/json

```
{
  "storageType" : "FS",
  "reachable" : true,
  "xmlStates" : [ {
    "contentConsistent" : true,
    "created" : "2000-01-23T04:56:07.000+00:00",
    "metadataConsistent" : true,
    "state" : "PRE_PROCESSING",
    "databaseId" : "databaseId",
    "version" : 0,
    "storageId" : "storageId"
  }, {
    "contentConsistent" : true,
    "created" : "2000-01-23T04:56:07.000+00:00",
    "metadataConsistent" : true,
    "state" : "PRE_PROCESSING",
    "databaseId" : "databaseId",
    "version" : 0,
    "storageId" : "storageId"
  } ],
  "storageName" : "storageName",
  "aipState" : {
    "contentConsistent" : true,
    "created" : "2000-01-23T04:56:07.000+00:00",
    "metadataConsistent" : true,
    "state" : "PRE_PROCESSING",
    "storageChecksum" : {
      "type" : "MD5",
      "value" : "value"
    }
  },
  "databaseId" : "databaseId",
```

```
    "storageId" : "storageId"
  }
}
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

## Responses

200

AIP successfully deleted [AipConsistencyVerificationResultDto](#)

400

bad request, e.g. the specified id is not a valid UUID [AipConsistencyVerificationResultDto](#)

403

not authorized or the storage is just synchronizing [AipConsistencyVerificationResultDto](#)

500

internal server error [AipConsistencyVerificationResultDto](#)

503

storage is not reachable [AipConsistencyVerificationResultDto](#)

POST /api/storage/{aipId}/aip-with-files-specified

Up

Return AIP with specified files packed in ZIP ([getAipReducedByListOfFiles](#))

validates the AIP and if it is invalid tries to recover it from other storage.. does not return only files but also AIP XML(s)

## Path parameters

AIP ID (required)

*Path Parameter* –

## Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

## Request body

body [string](#) (required)

*Body Parameter* –

## Query parameters

true to return all XMLs, otherwise only the latest is returned  
(optional)

*Query Parameter* – default: false

## Responses

200

AIP successfully returned

400

bad request, e.g. the specified id is not a valid UUID

403

operation forbidden with respect to the current AIP state

500

file is corrupted at all storages, no logical storage attached, or other internal server error

503

all attached logical storages are currently unreachable

## POST /api/storage/{aipId}/aip-with-files-reduced

Return AIP with specified files packed in ZIP ([getAipReducedByRegex](#))

validates the AIP and if it is invalid tries to recover it from other storage.. does not return only files but also AIP XML(s)

### Path parameters

**AIP ID (required)**

*Path Parameter* —

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [DataReduction](#) (required)

*Body Parameter* —

### Query parameters

true to return all XMLs, otherwise only the latest is returned  
(optional)

*Query Parameter* — default: false

### Responses

**200**

AIP successfully returned

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

operation forbidden with respect to the current AIP state

**500**

file is corrupted at all storages, no logical storage attached, or other internal server error

**503**

all attached logical storages are currently unreachable

## GET /api/storage/{aipId}/state

State of AIP. ([getAipState](#))

Retrieves the state of AIP stored in database.

### Path parameters

**AIP id (required)**

*Path Parameter* —

### Return type

String

### Example data

Content-Type: application/json

```
"PRE_PROCESSING"
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

## Responses

200

state of AIP successfully retrieved [String](#)

404

AIP with the id not found [String](#)

[Up](#)

GET /api/storage/{aipId}/xml

Return specified AIP XML ([getXml](#))

### Path parameters

**AIP ID (required)**

*Path Parameter* –

### Query parameters

version number of XML, if not set the latest version is returned  
(optional)

*Query Parameter* – format: int32

## Responses

200

AIP XML successfully returned

400

bad request, e.g. the specified id is not a valid UUID

403

operation forbidden with respect to the current AIP state

500

file is corrupted at all storages, no logical storage attached, or other internal server error

503

all attached logical storages are currently unreachable

[Up](#)

GET /api/storage/{aipId}/xml/{xmlVersion}/state

State of XML. ([getXmlState](#))

Retrieves the state of XML stored in database.

### Path parameters

**AIP ID (required)**

*Path Parameter* –

**XML version (required)**

*Path Parameter* – format: int32

### Return type

String

### Example data

Content-Type: application/json



```
"PRE_PROCESSING"
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

### Responses

**200**

state of XML successfully retrieved [String](#)

**404**

XML not found [String](#)

[Up](#)

## PUT /api/storage/{objId}/remove

Logically removes object by setting its state to REMOVED. (**remove**)

Not allowed for AIP XML objects.

### Path parameters

**object id (required)**

*Path Parameter* —

### Responses

**200**

object successfully removed

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

operation forbidden with respect to the current object state

**500**

no logical storage attached, or other internal server error

**503**

some attached logical storage is currently not reachable or system is in readonly state

[Up](#)

## PUT /api/storage/{objId}/renew

Renews logically removed object by setting its state to ARCHIVED. (**renew**)

Not allowed for AIP XML objects.

### Path parameters

**object id (required)**

*Path Parameter* —

### Responses

**200**

object successfully renewed

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

operation forbidden with respect to the current object state

**500**

no logical storage attached, or other internal server error

**503**

some attached logical storage is currently not reachable or system is in readonly state

[Up](#)

## DELETE /api/storage/{objId}/rollback

Deletes the object data. (**rollback**)

If the object is AIP, it has to contain exactly one AIP XML which is also rolled back. Metadata and DB records remains persistent with state set to ROLLED\_BACK. If the object to rollback does not exists, then do nothing.

### Path parameters

**object id (required)**  
*Path Parameter* –

### Responses

**200**

object successfully rolled back or rollback skipped because the object was not found

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

trying to rollback AIP XML (there is a special endpoint for that case) OR trying to rollback AIP which has more than one AIP XML linked

**500**

no logical storage attached, or other internal server error

**503**

some attached logical storage is currently not reachable or system is in readonly state

[Up](#)

## DELETE /api/storage/{aipId}/rollbackXml/{xmlVersion}

Deletes data of the latest AIP XML. Version set in the path variable must be the latest AIP XML of the AIP. (**rollbackXml**)

### Path parameters

**AIP id (required)**  
*Path Parameter* –

**XML version (required)**  
*Path Parameter* – format: int32

### Responses

**200**

object successfully rolled back or rollback skipped because the object was not found

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

if the XML is version 1 (whole AIP should be rolled back instead) or if this is not the latest XML version of the AIP (newer versions exist)

**500**

no logical storage attached, or other internal server error

**503**

some attached logical storage is currently not reachable or system is in readonly state

[Up](#)

## POST /api/storage/save

Stores AIP parts (SIP and AIP XML) into Archival Storage and returns the AIP ID. (**saveAip**)

## Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

## Request body

body [storage\\_save\\_body](#) (optional)  
*Body Parameter* –

## Query parameters

value of the SIP checksum (required)  
*Query Parameter* –

type of the SIP checksum (required)  
*Query Parameter* –

value of the AIP XML checksum (required)  
*Query Parameter* –

type of the AIP XML checksum (required)  
*Query Parameter* –

UUID of the AIP, generated if not specifies (optional)  
*Query Parameter* –

## Return type

String

## Example data

Content-Type: application/json

```
""
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

## Responses

**200**

AIP successfully stored [String](#)

**400**

bad request, e.g. the specified id is not a valid UUID [String](#)

**422**

the checksum computed after the transfer does not match the provided checksum [String](#)

**500**

no logical storage attached, or other internal server error [String](#)

**503**

some attached logical storage is currently not reachable or system is in readonly state [String](#)

POST /api/storage/{aipId}/update

[Up](#)

Versioning of the AIP XML: stores new AIP XML into Archival Storage. ([saveXml](#))

Sync flag is set to false for batch AIP XML updates from Ingest. Sync flag is set to true for AIP XML updates invoked from user's interaction with the AIP XML editor in GUI.

## Path parameters

**AIP id (required)**  
*Path Parameter* –

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [aipId\\_update\\_body](#) (optional)  
*Body Parameter* –

### Query parameters

**AIP XML checksum value (required)**  
*Query Parameter* –

**AIP XML checksum type (required)**  
*Query Parameter* –

**synchronous/asynchronous processing flag (optional)**  
*Query Parameter* – default: false

**version number of the AIP XML, is automatically set to the lastVersion+1 if not specified (optional)**  
*Query Parameter* – format: int32

### Responses

**200**

AIP XML successfully stored

**400**

bad request, e.g. the specified id is not a valid UUID

**403**

operation forbidden with respect to the current AIP state

**409**

bad XML version number provided (not following the sequence)

**422**

the checksum computed after the transfer does not match the provided checksum

**500**

no logical storage attached, or other internal server error

**503**

some attached logical storage is currently not reachable or system is in readonly state

## StorageAdministrationApi

[Up](#)

POST /api/administration/storage

Attaches new logical storage and starts synchronization. (**attachStorage**)

After the initial checks are done, process continues asynchronously. See documentation for example JSON configurations

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [Storage](#) (required)

*Body Parameter* –

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

### Responses

**400**

invalid configuration [Storage](#)

**409**

storage already exists [Storage](#)

**503**

storage to be synchronized is not reachable [Storage](#)

**566**

synchronization initialization timeout because there are still processing objects at the archival storage [Storage](#)

**567**

cant create some dataspace at the storage [Storage](#)

POST /api/administration/storage/check\_reachability

[Up](#)

Check reachability of all storages ([checkReachability](#))

### Responses

**200**

OK

POST /api/administration/storage/sync/{id}

[Up](#)

Continues with failed (stopped) synchronization. ([continueSync](#))

After the initial checks are done, system asynchronously continues from the last successfully synced object/operation.

### Path parameters

**id of the synchronization status entity (required)**

*Path Parameter* –

### Responses

**403**

synchronization is already in progress or not authorized

**503**

storage to be synchronized is not reachable

DELETE /api/administration/storage/{id}

[Up](#)

Removes a logical storage. ([delete1](#))

### Path parameters

**id of the logical storage (required)**

*Path Parameter* –

### Responses

403

not authorized or can't be deleted because count of logical storage would be less than the configured minimum

[Up](#)

## GET /api/administration/storage

Returns all attached logical storages. (**getAll**)

### Return type

array[[Storage](#)]

### Example data

Content-Type: application/json

```
[ {
  "note" : "note",
  "port" : 0,
  "synchronizing" : true,
  "name" : "name",
  "host" : "host",
  "storageType" : "FS",
  "id" : "id",
  "priority" : 6,
  "config" : "config",
  "reachable" : true
}, {
  "note" : "note",
  "port" : 0,
  "synchronizing" : true,
  "name" : "name",
  "host" : "host",
  "storageType" : "FS",
  "id" : "id",
  "priority" : 6,
  "config" : "config",
  "reachable" : true
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

### Responses

200

OK

[Up](#)

## GET /api/administration/storage/basic

Returns simple DTOs of all attached logical storages. (**getAllAsDtos**)

### Return type

array[[StorageBasicDto](#)]

### Example data

Content-Type: application/json

```
[ {
  "synchronizing" : true,
  "name" : "name",
  "storageType" : "storageType",
  "id" : "id"
}, {
  "synchronizing" : true,
  "name" : "name",
  "storageType" : "storageType",
  "id" : "id"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

### Responses

200  
OK

[Up](#)

GET /api/administration/storage/{id}

Returns logical storage with specified ID. (**getOne**)

### Path parameters

**id of the logical storage (required)**  
*Path Parameter* —

### Return type

[Storage](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "port" : 0,
  "synchronizing" : true,
  "name" : "name",
  "host" : "host",
  "storageType" : "FS",
  "id" : "id",
  "priority" : 6,
  "config" : "config",
  "reachable" : true
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

### Responses

200

OK [Storage](#)

[Up](#)

## GET /api/administration/storage/{id}/state

Returns state of logical storage. (**getStorageState**)

Data returned depends on the logical storage. E.g. self-healing data are returned only if self-healing is supported by the storage.

### Path parameters

**id of the logical storage (required)**  
*Path Parameter* —

### Return type

[StorageStateDto](#)

### Example data

Content-Type: application/json

```
{
  "storageStateData" : {
    "key" : { }
  },
  "storage" : {
    "note" : "note",
    "port" : 0,
    "synchronizing" : true,
    "name" : "name",
    "host" : "host",
    "storageType" : "FS",
    "id" : "id",
    "priority" : 6,
    "config" : "config",
    "reachable" : true
  }
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

### Responses

200

OK [StorageStateDto](#)

[Up](#)

## GET /api/administration/storage/sync/{id}

Retrieves sync status entity for the storage. (**getSyncStatusOfStorage**)

### Path parameters

**id of the storage (required)**  
*Path Parameter* —



## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

## Responses

404

no sync status for this storage [StorageSyncStatus](#)

[Up](#)

## POST /api/administration/storage/update

Updates a logical storage. (**update**)

Only name, priority and note flags are updatable, others are ignored.

## Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

## Request body

body [StorageUpdateDto](#) (required)

*Body Parameter* –

## Return type

[Storage](#)

## Example data

Content-Type: application/json

```
{
  "note" : "note",
  "port" : 0,
  "synchronizing" : true,
  "name" : "name",
  "host" : "host",
  "storageType" : "FS",
  "id" : "id",
  "priority" : 6,
  "config" : "config",
  "reachable" : true
}
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

## Responses

200

OK [Storage](#)

# SystemAdministrationApi

## GET /api/administration/backup

Exports all objects which have changed in the specified time range. (**backup**)

Once the reachability of export location is verified, the response is returned and process continues asynchronously. If since/until is not filled, MIN/MAX is used.

### Query parameters

**since (optional)**

*Query Parameter* – format: date-time

**until (optional)**

*Query Parameter* – format: date-time

### Responses

**500**

Export location unreachable or other internal error.

## POST /api/administration/cleanup

Cleans storage. In order to succeed, all storages must be reachable. (**cleanup**)

By default only failed package are cleaned (i.e. rolled back/deleted from all storages). If 'all' is set to true, also currently processing/stucked packages and tmp folder is cleaned.

### Query parameters

**all (optional)**

*Query Parameter* – default: false

### Responses

**403**

some logical storage is synchronizing

**503**

some logical storage unreachable

## POST /api/administration/cleanup/object/{objId}

Cleans one particular object. In order to succeed, all storages must be reachable. (**cleanupOne**)

Object must be in failed or processing state. If the object is in processing state, caller must ensure that the object is not actually processing but the state was left processing because some unexpected error has occurred (e.g. system crashed).

### Path parameters

**database id of the object (required)**

*Path Parameter* –

### Responses

**403**

some logical storage is synchronizing

**503**

some logical storage unreachable

## GET /api/administration/config

Returns configuration (**get**)

### Return type

[SystemState](#)

### Example data

Content-Type: application/json

```
{
  "lastReachabilityCheck" : "2000-01-23T04:56:07.000+00:00",
  "readOnly" : true,
  "reachabilityCheckIntervalInMinutes" : 6,
  "id" : "id",
  "minStorageCount" : 0,
  "lastVerifiedObjectCreation" : "2000-01-23T04:56:07.000+00:00"
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

### Responses

200

OK [SystemState](#)

## POST /api/administration/recover\_db

[Up](#)

Recovers database from the storage. (**recoverDbFromStorage**)

Synchronous call. System has to be set to read-only mode. Currently the recovery is implemented only for local FS/ZFS. If the object metadata is present in DB and also at storage, the metadata are compared and if not equal, system logs (warn) the conflict and if override parameter is set to true, DB metadata are overridden by storage metadata. System also logs (error) all objects which are in DB but not at storage.

### Query parameters

**storage id (required)**

*Query Parameter* –

**whether to override conflicting metadata in DB with those from storage (optional)**

*Query Parameter* – default: false

### Responses

403

Not authorized or system is in read-write mode

503

logical storage unreachable

## POST /api/administration/config

[Up](#)

Updates systemState of the Archival Storage. (**save**)

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

## Request body

body [SystemState](#) (required)  
*Body Parameter* –

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

## Responses

**400**

the config breaks the basic LTP policy (e.g. count of storages) [SystemState](#)

**409**

provided config has different id than that stored in DB (only one config object is allowed) [SystemState](#)

# User

POST /api/user/activate/{dataSpace}

[Up](#)

Activates dataSpace i.e. creates folders/buckets etc. at all logical storages. (**activateDataSpace**)

DataSpace should be simple string, no dashes, slashes, uppercase letters, spaces or underscores.

## Path parameters

dataSpace (required)  
*Path Parameter* –

## Responses

**500**

no logical storage attached or internal server error

**503**

some logical storage not reachable or system is in readonly state

POST /api/user/register

[Up](#)

Register user. (**registerUser**)

DataSpace should be simple string, no dashes, slashes, uppercase letters, spaces or underscores. If the user has READ/READ-WRITE role and the dataSpace is new, the dataSpace should be activated by another endpoint.

## Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

## Request body

body [User](#) (required)  
*Body Parameter* –

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- \*/\*

## Responses

400

Wrong JSON | username/password/role missing | role!=ADMIN and dataSpace missing | role==ADMIN and dataSpace not null [User](#)

409

username already exists [User](#)

## Models

[ [Jump to Methods](#) ]

### Table of Contents

1. [AipConsistencyVerificationResultDto](#)
2. [Checksum](#)
3. [DataReduction](#)
4. [ObjectConsistencyVerificationResultDto](#)
5. [Storage](#)
6. [StorageBasicDto](#)
7. [StorageStateDto](#)
8. [StorageSyncStatus](#)
9. [StorageSyncStatus\\_exceptionInfo](#)
10. [StorageSyncStatus\\_exceptionInfo\\_cause](#)
11. [StorageSyncStatus\\_exceptionInfo\\_cause\\_stackTrace](#)
12. [StorageSyncStatus\\_exceptionInfo\\_cause\\_suppressed](#)
13. [StorageUpdateDto](#)
14. [SystemState](#)
15. [User](#)
16. [XmlConsistencyVerificationResultDto](#)
17. [aipId\\_update\\_body](#)
18. [storage\\_save\\_body](#)

### AipConsistencyVerificationResultDto

[Up](#)

storageName (optional)

[String](#)

storageType (optional)

[String](#)

Enum:

FS

ZFS

CEPH

reachable (optional)

[Boolean](#)

aipState (optional)

[ObjectConsistencyVerificationResultDto](#)

xmlStates (optional)

[array\[XmlConsistencyVerificationResultDto\]](#)

### Checksum

[Up](#)

type (optional)

[String](#)

Enum:

*MD5*

*SHA512*

value (optional)

[String](#)

## DataReduction

[Up](#)

regexes

[array\[String\]](#)

mode

[String](#)

Enum:

*INCLUDE*

*EXCLUDE*

## ObjectConsistencyVerificationResultDto

[Up](#)

databaseId (optional)

[String](#)

storageId (optional)

[String](#)

state (optional)

[String](#)

Enum:

*PRE\_PROCESSING*

*PROCESSING*

*ARCHIVED*

*ROLLED\_BACK*

*DELETED*

*REMOVED*

*ARCHIVAL\_FAILURE*

*DELETION\_FAILURE*

*ROLLBACK\_FAILURE*

*FORGOT*

contentConsistent (optional)

[Boolean](#)

metadataConsistent (optional)

[Boolean](#)

storageChecksum (optional)

[Checksum](#)

databaseChecksum (optional)

[Checksum](#)

created (optional)

[Date](#) format: date-time

## Storage

[Up](#)

id (optional)

[String](#)

name  
[String](#)

host  
[String](#)

port (optional)  
[Integer](#) format: int32

priority (optional)  
[Integer](#) format: int32

storageType  
[String](#)  
Enum:  
FS  
ZFS  
CEPH

note (optional)  
[String](#)

config (optional)  
[String](#)

reachable (optional)  
[Boolean](#)

synchronizing (optional)  
[Boolean](#)

## StorageBasicDto

[Up](#)

id (optional)  
[String](#)

name (optional)  
[String](#)

storageType (optional)  
[String](#)

synchronizing (optional)  
[Boolean](#)

## StorageStateDto

[Up](#)

storage (optional)  
[Storage](#)

storageStateData (optional)  
[map\[String, Object\]](#)

## StorageSyncStatus

[Up](#)

id (optional)  
[String](#)

storage (optional)  
[Storage](#)

created (optional)  
[Date](#) format: date-time

updated (optional)  
[Date](#) format: date-time

phase (optional)  
[String](#)

Enum:  
INIT  
COPYING\_ARCHIVED\_OBJECTS  
PROPAGATING\_OPERATIONS  
POST\_SYNC\_CHECK  
DONE

totalInThisPhase (optional)  
[Long](#) format: int64

doneInThisPhase (optional)  
[Long](#) format: int64

exceptionMsg (optional)  
[String](#)

exceptionStackTrace (optional)  
[String](#)

stuckAt (optional)  
[Date](#) format: date-time

exceptionInfo (optional)  
[StorageSyncStatus\\_exceptionInfo](#)

## StorageSyncStatus\_exceptionInfo

[Up](#)

cause (optional)  
[StorageSyncStatus\\_exceptionInfo\\_cause](#)

stackTrace (optional)  
[array\[StorageSyncStatus\\_exceptionInfo\\_cause\\_stackTrace\]](#)

message (optional)  
[String](#)

suppressed (optional)  
[array\[StorageSyncStatus\\_exceptionInfo\\_cause\\_suppressed\]](#)

localizedMessage (optional)  
[String](#)

## StorageSyncStatus\_exceptionInfo\_cause

[Up](#)

stackTrace (optional)  
[array\[StorageSyncStatus\\_exceptionInfo\\_cause\\_stackTrace\]](#)

message (optional)  
[String](#)

suppressed (optional)  
[array\[StorageSyncStatus\\_exceptionInfo\\_cause\\_suppressed\]](#)

localizedMessage (optional)  
[String](#)

## StorageSyncStatus\_exceptionInfo\_cause\_stackTrace

[Up](#)



classLoaderName (optional)

[String](#)

moduleName (optional)

[String](#)

moduleVersion (optional)

[String](#)

methodName (optional)

[String](#)

fileName (optional)

[String](#)

lineNumber (optional)

[Integer](#) format: int32

className (optional)

[String](#)

nativeMethod (optional)

[Boolean](#)

## StorageSyncStatus\_exceptionInfo\_cause\_suppressed

[Up](#)

stackTrace (optional)

[array\[StorageSyncStatus\\_exceptionInfo\\_cause\\_stackTrace\]](#)

message (optional)

[String](#)

localizedMessage (optional)

[String](#)

## StorageUpdatedDto

[Up](#)

id

[String](#)

priority (optional)

[Integer](#) format: int32

name (optional)

[String](#)

note (optional)

[String](#)

## SystemState

[Up](#)

id (optional)

[String](#)

minStorageCount (optional)

[Integer](#) format: int32

readOnly (optional)

[Boolean](#)

reachabilityCheckIntervallInMinutes (optional)

[Integer](#) format: int32

lastReachabilityCheck (optional)

[Date](#) format: date-time

lastVerifiedObjectCreation (optional)

[Date](#) format: date-time

## User

[Up](#)

id (optional)

[String](#)

created (optional)

[Date](#) format: date-time

updated (optional)

[Date](#) format: date-time

deleted (optional)

[Date](#) format: date-time

username

[String](#)

password (optional)

[String](#)

email (optional)

[String](#)

dataSpace (optional)

[String](#)

role

[String](#)

Enum:

*ROLE\_ADMIN*

*ROLE\_READ*

*ROLE\_READ\_WRITE*

## XmlConsistencyVerificationResultDto

[Up](#)

databaseId (optional)

[String](#)

storageId (optional)

[String](#)

state (optional)

[String](#)

Enum:

*PRE\_PROCESSING*

*PROCESSING*

*ARCHIVED*

*ROLLED\_BACK*

*DELETED*

*REMOVED*

*ARCHIVAL\_FAILURE*

*DELETION\_FAILURE*

*ROLLBACK\_FAILURE*

*FORGOT*

contentConsistent (optional)

[Boolean](#)

metadataConsistent (optional)

[Boolean](#)

storageChecksum (optional)

[Checksum](#)

databaseChecksum (optional)

[Checksum](#)

created (optional)

[Date](#) format: date-time

version (optional)

[Integer](#) format: int32

aipId\_update\_body

[Up](#)

xml

[byte\[\]](#) format: binary

storage\_save\_body

[Up](#)

sip

[byte\[\]](#) format: binary

aipXml

[byte\[\]](#) format: binary