

# LDG MDC-WS Operations (1.3)

D. Pleiter, D. Melkumyan

07 November 2006

This document describes the LDG-specific operations implemented in the MDC-WS.

## Contents

<b>1 Terminology</b>	<b>2</b>
<b>2 Status Codes</b>	<b>2</b>
<b>3 Types</b>	<b>3</b>
3.1 Response . . . . .	3
3.2 ResponseString . . . . .	3
<b>4 Operations: Description and Interfaces</b>	<b>3</b>
4.1 Operation doEnsembleInsert . . . . .	3
4.2 Operation doMetadataInsert . . . . .	4
4.3 Operation doEnsembleUpdate . . . . .	5
4.4 Operation doMetadataUpdate . . . . .	6
4.5 Operation doEnsembleDelete . . . . .	7
4.6 Operation doMetadataDelete . . . . .	8
4.7 Operation doEnsembleValidate . . . . .	8
4.8 Operation doMetadataValidate . . . . .	9
4.9 Operation doCertMapUpdate . . . . .	10
4.10 Operation doCertMapDelete . . . . .	11
4.11 Operation doCertMapList . . . . .	12
4.12 Operation doPrjMapInsert . . . . .	13
4.13 Operation doPrjMapUpdate . . . . .	14
4.14 Operation doPrjMapDelete . . . . .	15
4.15 Operation doPrjMapList . . . . .	16
4.16 Operation doGrpMapInsert . . . . .	17
4.17 Operation doGrpMapUpdate . . . . .	18
4.18 Operation doGrpMapDelete . . . . .	19

4.19	Operation doGrpMapList	20
4.20	Operation doEnsembleMapInsert	21
4.21	Operation doEnsembleMapUpdate	22
4.22	Operation doEnsembleMapDelete	23
4.23	Operation doEnsembleMapList	24
4.24	Operation doAdmInsert	25
4.25	Operation doAdmDelete	26
4.26	Operation doAdmList	27
4.27	Operation doManagerInsert	28
4.28	Operation doManagerDelete	29
4.29	Operation doManagerList	30
4.30	Operation doGroupInsert	31
4.31	Operation doGroupDelete	32
4.32	Operation doGroupList	33
4.33	Operation doAclInsert	34
4.34	Operation doAclDelete	35
4.35	Operation doAclList	36
4.36	Operation doDump	37
4.37	Operation doRestore	38
4.38	Operation doSetConfig	39
4.39	Operation doGetConfig	40
<b>5</b>	<b>Activity diagrams</b>	<b>42</b>
5.1	Operation doEnsembleUpdate	42
5.2	Operation doMetadataUpdate	43
5.3	Operation doAdmInsert	44
5.4	Operation doAclDelete	45
<b>6</b>	<b>Changelog</b>	<b>45</b>
6.1	Changes since 1.2	45
6.2	Changes since 1.1	45
6.3	Changes since 1.0	45

## 1 Terminology

We assume two types of XML documents: ensemble and metadata XML documents. These documents can be identified through the `ensembleURI` (in ILDG this is equivalent to the `markovChainURI`) and the `dataLFN` (also used in ILDG).

## 2 Status Codes

The MDC webservice will return the status codes listed below. Note that the status codes with prefix `MDC_LDG_` are not ILDG compliant and specific for the LDG MDC:

MDC_SUCCESS	Operation has been completed successfully
MDC_INVALID_REQUEST	Badly formulated query
MDC_NO_DATA	No matching data
MDC_FAILURE	Operation failed
MDC_LDG_INVALID_XML	XML document does not conform to schema
MDC_LDG_INVALID_ID	Invalid document ID (ensembleURI, dataLFN)
MDC_LDG_NO_PRIVILEGE	Insufficient privileges for requested operation
MDC_LDG_INVALID_STATE	MDC-Core is a busy performing dump/restore

## 3 Types

### 3.1 Response

```
<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

### 3.2 ResponseString

```
<complexType name="ResponseString">
  <complexContent>
    <extension base="tns2:Response">
      <sequence>
        <element name="string" nillable="true" type="xsd:string"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

## 4 Operations: Description and Interfaces

### 4.1 Operation doEnsembleInsert

**Description:** Insert a new ensemble XML document. Operation fails if `ensembleURI` does already exist or if an entry for `ensembleURI` does not exist in the table `project` or if document does not conform to schema.

**Access:** secure [group privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_INVALID\_XML, MDC\_LDG\_INVALID\_ID, MDC\_LDG\_NO\_PRIVILEGE

**Input:** doEnsembleInsertRequest

xmlString	Ensemble XML document
-----------	-----------------------

```
<wsdl:message name="doEnsembleInsertRequest">
  <wsdl:part name="xmlString" type="xsd:string"/>
</wsdl:message>
```

**Output:** doEnsembleInsertResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doEnsembleInsertResponse">
  <wsdl:part name="doEnsembleInsertReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

## 4.2 Operation doMetadataInsert

**Description:** Insert a new configuration XML document. Operation fails if `dataLFN` does already exist or if `ensembleURI` does not exist or if document does not conform to schema. In case of an successful operation the element `<archiveHistory>` is patched.

**Access:** secure [group privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_INVALID\_XML, MDC\_LDG\_INVALID\_ID, MDC\_LDG\_NO\_PRIVILEGE

**Input:** doMetadataInsertRequest

xmlString	Metadata XML document
-----------	-----------------------

```
<wsdl:message name="doMetadataInsertRequest">
  <wsdl:part name="xmlString" type="xsd:string"/>
</wsdl:message>
```

**Output:** doMetadataInsertResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doMetadataInsertResponse">
  <wsdl:part name="doMetadataInsertReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

### 4.3 Operation doEnsembleUpdate

**Description:** Server checks for existence of the ensemble XML document and updates all updates elements except the element <archiveHistory>. Otherwise an error is returned.

**Access:** secure [group privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_INVALID\_XML, MDC\_LDG\_INVALID\_ID, MDC\_LDG\_NO\_PRIVILEGE

**Input:** doEnsembleUpdateRequest

xmlString	Ensemble XML document
-----------	-----------------------

```
<wsdl:message name="doEnsembleUpdateRequest">
  <wsdl:part name="xmlString" type="xsd:string"/>
</wsdl:message>
```

**Output:** doEnsembleUpdateResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doEnsembleUpdateResponse">
  <wsdl:part name="doEnsembleUpdateReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.4 Operation doMetadataUpdate

**Description:** Server checks for existence of metadata XML document and updates all updates elements except the element <archiveHistory>. Otherwise an error is returned.

**Access:** secure [group privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_INVALID\_XML, MDC\_LDG\_INVALID\_ID, MDC\_LDG\_NO\_PRIVILEGE

**Input:** doMetadataUpdateRequest

xmlString	Metadata XML document
-----------	-----------------------

```
<wsdl:message name="doMetadataUpdateRequest">
  <wsdl:part name="xmlString" type="xsd:string"/>
</wsdl:message>
```

**Output:** doMetadataUpdateResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doMetadataUpdateResponse">
  <wsdl:part name="doMetadataUpdateReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

## 4.5 Operation doEnsembleDelete

**Description:** Remove ensemble XML document from MDC.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doEnsembleDeleteRequest

ensembleURI	URI for ensemble document
-------------	---------------------------

```

<wsdl:message name="doEnsembleDeleteRequest">
  <wsdl:part name="ensembleURI" type="xsd:string"/>
</wsdl:message>

```

**Output:** doEnsembleQueryResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doEnsembleDeleteResponse">
  <wsdl:part name="doEnsembleDeleteReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

## 4.6 Operation doMetadataDelete

**Description:** Remove configuration XML document from MDC. If dataLFN is not known or if user has no right to perform requested operation, status code MDC\_LDG\_NO\_PRIVILEGE is returned.

**Access:** secure [group privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doMetadataDeleteRequest

metadataLFN	Identifier of XML document
-------------	----------------------------

```
<wsdl:message name="doMetadataDeleteRequest">
  <wsdl:part name="metadataLFN" type="xsd:string"/>
</wsdl:message>
```

**Output:** doMetadataQueryResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doMetadataDeleteResponse">
  <wsdl:part name="doMetadataDeleteReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

## 4.7 Operation doEnsembleValidate

**Description:** Validate XML document with respect to ensemble schema. If the operation of validation succeeds the status code MDC\_SUCCESS is returned (even if the document does not conform to the schema).

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_INVALID\_XML, MDC\_INVALID\_REQUEST



**Input:** doEnsembleValidateRequest

xmlString	Ensemble XML document
-----------	-----------------------

```
<wsdl:message name="doEnsembleValidateRequest">
  <wsdl:part name="xmlString" type="xsd:string"/>
</wsdl:message>
```

**Output:** doEnsembleValidateResponse

statusCode	Return status string
string	Validation messages

```
<wsdl:message name="doEnsembleValidateResponse">
  <wsdl:part name="doEnsembleValidateReturn" type="tns1:ResponseString"/>
</wsdl:message>
```

```
<complexType name="ResponseString">
  <complexContent>
    <extension base="tns2:Response">
      <sequence>
        <element name="string" nillable="true" type="xsd:string"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

## 4.8 Operation doMetadataValidate

**Description:** Validate XML document with respect to configuration schema. Response similar to doEnsembleValidate.

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_INVALID\_XML, MDC\_INVALID\_REQUEST

**Input:** doMetadataValidateRequest

xmlString	Metadata XML document
-----------	-----------------------

```
<wsdl:message name="doMetadataValidateRequest">
  <wsdl:part name="xmlString" type="xsd:string"/>
</wsdl:message>
```

**Output:** doMetadataValidateResponse

statusCode	Return status string
string	Validation messages

```
<wsdl:message name="doMetadataValidateResponse">
  <wsdl:part name="doMetadataValidateReturn" type="tns1:ResponseString"/>
</wsdl:message>

<complexType name="ResponseString">
  <complexContent>
    <extension base="tns2:Response">
      <sequence>
        <element name="string" nillable="true" type="xsd:string"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

#### 4.9 Operation doCertMapUpdate

**Description:** Update certID in table certmap for given cid after having checked that cid exists and certID is member of the VO. Otherwise the update is not performed and an error returned. Updating cid is not possible.

**Access:** secure [administrator privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doCertMapUpdateRequest

cid	(Existing) numerical identifier
certID	New distinguished name of certificate

```
<wsdl:message name="doCertMapUpdateRequest">
  <wsdl:part name="cid" type="xsd:int"/>
  <wsdl:part name="certID" type="xsd:string"/>
</wsdl:message>
```

**Output:** doCertMapUpdateResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doCertMapUpdateResponse">
  <wsdl:part name="doCertMapUpdateReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.10 Operation doCertMapDelete

**Description:** Delete certificate from table certmap after having checked that cid is not used in any other table. Otherwise delete is not performed and the status code MDC\_FAILED is returned. If cid does not exist or the remove operation is succesful, MDC\_SUCCESS is returned.

**Access:** secure [administrator privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doCertMapDeleteRequest

cid	Numerical certificate identifier
-----	----------------------------------

```
<wsdl:message name="doCertMapDeleteRequest">
  <wsdl:part name="cid" type="xsd:int"/>
</wsdl:message>
```

**Output:** doCertMapDeleteResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doCertMapDeleteResponse">
  <wsdl:part name="doCertMapDeleteResponseReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

#### 4.11 Operation doCertMapList

**Description:** Return content of table certmap.

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE

**Input:** doCertMapListRequest

Operation requires no arguments.

```

<wsdl:message name="doCertMapListRequest">
</wsdl:message>

```

**Output:** doCertMapListResponse

CID	Numerical identifier
certID	Distinguished name of certificate

```

<wsdl:message name="doCertMapListResponse">
  <wsdl:part name="doCertMapListReturn" type="tns1:ResponseCertList"/>
</wsdl:message>

<complexType name="ResponseCertList">
  <complexContent>
    <extension base="tns1:Response">
      <sequence>
        <element name="certList" nillable="true" type="impl:ArrayOf_tns3_Certificate"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="ArrayOf_tns3_Certificate">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="tns2:Certificate[]" />
    </restriction>
  </complexContent>
</complexType>

<complexType name="Certificate">
  <sequence>
    <element name="CID" type="xsd:int"/>
    <element name="certID" nillable="true" type="soapenc:string"/>
  </sequence>
</complexType>

```

## 4.12 Operation doPrjMapInsert

**Description:** Insert new project with name `prjName` into table `prjmap`. The `prjid` is automatically assigned by MDC.

**Access:** secure [administrator privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** `doPrjMapInsertRequest`

<code>collaboration</code>	Name of collaboration
<code>prjName</code>	Name of project

```

<wsdl:message name="doPrjMapInsertRequest">
  <wsdl:part name="collaboration" type="xsd:string"/>
  <wsdl:part name="prjName" type="xsd:string"/>
</wsdl:message>

```

**Output:** doPrjMapInsertResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doPrjMapInsertResponse">
  <wsdl:part name="doPrjMapInsertReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

### 4.13 Operation doPrjMapUpdate

**Description:** Update (collaboration,prjName) in table prjmap for given prjid after having checked that prjid exists. Otherwise the update is not performed and an error returned. Updating prjid is not possible.

**Access:** secure [administrator privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doPrjMapUpdateRequest

prjid	Numerical project identifier
collaboration	Name of collaboration
prjName	Name of project

```

<wsdl:message name="doPrjMapUpdateRequest">
  <wsdl:part name="prjid" type="xsd:int"/>
  <wsdl:part name="collaboration" type="xsd:string"/>
  <wsdl:part name="prjName" type="xsd:string"/>
</wsdl:message>

```

**Output:** doPrjMapUpdateResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doPrjMapUpdateResponse">
  <wsdl:part name="doPrjMapUpdateReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.14 Operation doPrjMapDelete

**Description:** Delete project from table `prjmap` after having checked that `prjid` is not used in any other table. Otherwise delete is not performed and an error returned.

**Access:** secure [administrator privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doPrjMapDeleteRequest

prjid	Numerical project identifier
-------	------------------------------

```
<wsdl:message name="doPrjMapDeleteRequest">
  <wsdl:part name="prjid" type="xsd:int"/>
</wsdl:message>
```

**Output:** doPrjMapDeleteResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doPrjMapDeleteResponse">
  <wsdl:part name="doPrjMapDeleteReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

#### 4.15 Operation doPrjMapList

**Description:** Return content of table prjmap.

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE

**Input:** doPrjMapListRequest

Operation requires no arguments.

```

<wsdl:message name="doPrjMapListRequest">
</wsdl:message>

```

**Output:** doPrjMapListResponse

prjid	Numerical identifier
collaboration	Name of collaboration
prjName	Name of project



```

<wsdl:message name="doPrjMapListResponse">
  <wsdl:part name="doPrjMapListReturn" type="tns1:ResponsePrjMap"/>
</wsdl:message>

<complexType name="ResponsePrjMap">
  <complexContent>
    <extension base="tns1:Response">
      <sequence>
        <element name="prjMap" nillable="true" type="impl:ArrayOf_tns3_Prj"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="ArrayOf_tns3_Prj">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="tns2:Prj []"/>
    </restriction>
  </complexContent>
</complexType>

<complexType name="Prj">
  <sequence>
    <element name="collaboration" nillable="true" type="soapenc:string"/>
    <element name="prjID" type="xsd:int"/>
    <element name="prjName" nillable="true" type="soapenc:string"/>
  </sequence>
</complexType>

```

#### 4.16 Operation doGrpMapInsert

**Description:** Insert new group with name `grpName` for project identified by `prjid` into table `grpmap`. The `gid` is automatically assigned by MDC. For implementation reasons the group name has to be unique. If insertion is not possible because `grpName` does already exist or `prjid` does not exist, the status code `LDG_MDC_FAILED` is returned.

**Access:** secure [manager privileges]

**Status codes:** `MDC_SUCCESS`, `MDC_FAILURE`, `MDC_LDG_NO_PRIVILEGE`, `MDC_INVALID_REQUEST`

**Input:** doGrpMapInsertRequest

prjid	Numerical project identifier
grpName	Name of group

```
<wsdl:message name="doPrjMapInsertRequest">
  <wsdl:part name="prjid" type="xsd:int"/>
  <wsdl:part name="grpName" type="xsd:string"/>
</wsdl:message>
```

**Output:** doGrpMapInsertResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doGrpMapInsertResponse">
  <wsdl:part name="doGrpMapInsertReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.17 Operation doGrpMapUpdate

**Description:** Update grpName in table grpmap for given gid after having checked that gid does exist and grpName does not exist. Otherwise the update is not performed and status code LDG\_MDC\_FAILED is returned. Updating gid is not possible.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doGrpMapUpdateRequest

gid	Numerical group identifier
prjid	Numerical project identifier
grpName	New name of group

```

<wsdl:message name="doGrpMapInsertRequest">
  <wsdl:part name="gid" type="xsd:int"/>
  <wsdl:part name="prjid" type="xsd:int"/>
  <wsdl:part name="grpName" type="xsd:string"/>
</wsdl:message>

```

**Output:** doGrpMapUpdateResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doGrpMapUpdateResponse">
  <wsdl:part name="doGrpMapUpdateReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

#### 4.18 Operation doGrpMapDelete

**Description:** Delete group from table grpmap after having checked that gid is not used in any other table. Otherwise delete is not performed and an error returned.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doGrpMapDeleteRequest

gid	Numerical project identifier
-----	------------------------------

```

<wsdl:message name="doGrpMapDeleteRequest">
  <wsdl:part name="gid" type="xsd:int"/>
</wsdl:message>

```

**Output:** doGrpMapDeleteResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doGrpMapDeleteResponse">
  <wsdl:part name="doGrpMapDeleteReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.19 Operation doGrpMapList

**Description:** Return content of table grpmap.

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE

**Input:** doGrpMapListRequest

Operation requires no arguments.

```
<wsdl:message name="doGrpMapListRequest">
</wsdl:message>
```

**Output:** doGrpMapListResponse

GID	Numerical group identifier
collaboration	Name of collaboration
grpName	Name of group
prjID	Numerical project identifier
prjName	Name of project

```

<wsdl:message name="doGrpMapListResponse">
  <wsdl:part name="doGrpMapListReturn" type="tns1:ResponseGrpMap"/>
</wsdl:message>

<complexType name="ResponseGrpMap">
  <complexContent>
    <extension base="tns1:Response">
      <sequence>
        <element name="grpMap" nillable="true" type="impl:ArrayOf_tns3_GrpMap"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="ArrayOf_tns3_GrpMap">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="tns2:GrpMap[]" />
    </restriction>
  </complexContent>
</complexType>

<complexType name="GrpMap">
  <sequence>
    <element name="GID" type="xsd:int"/>
    <element name="collaboration" nillable="true" type="soapenc:string"/>
    <element name="grpName" nillable="true" type="soapenc:string"/>
    <element name="prjID" type="xsd:int"/>
    <element name="prjName" nillable="true" type="soapenc:string"/>
  </sequence>
</complexType>

```

## 4.20 Operation doEnsembleMapInsert

**Description:** Add new ensemble with name `ensembleURI` to project identified by `prjid` into table `ensemblemap` after having checked that no ensemble XML document exists with same `ensembleURI`. The numerical identifier `eid` is automatically assigned by MDC.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doEnsembleMapInsertRequest

prjid	Numerical project identifier
ensembleURI	Name of ensemble

```
<wsdl:message name="doEnsembleMapInsertRequest">
  <wsdl:part name="prjid" type="xsd:int"/>
  <wsdl:part name="ensembleURI" type="xsd:string"/>
</wsdl:message>
```

**Output:** doEnsembleMapInsertResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doEnsembleMapInsertResponse">
  <wsdl:part name="doEnsembleMapInsertReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.21 Operation doEnsembleMapUpdate

**Description:** Update ensembleURI in table ensemblemap for given eid after having checked that eid exists and no ensemble XML document exists with same ensembleURI. Otherwise the update is not performed and an error returned. Updating eid or prjid is not possible.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doEnsembleMapUpdateRequest

eid	Numerical ensemble identifier
ensembleURI	New URI for ensemble

```

<wsdl:message name="doEnsembleMapUpdateRequest">
  <wsdl:part name="eid" type="xsd:int"/>
  <wsdl:part name="ensembleURI" type="xsd:string"/>
</wsdl:message>

```

**Output:** doEnsembleMapUpdateResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doEnsembleMapUpdateResponse">
  <wsdl:part name="doEnsembleMapUpdateReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

## 4.22 Operation doEnsembleMapDelete

**Description:** Delete ensemble from table `ensemblemap` after having checked that `eid` is not used in table `acl`. Otherwise delete is not performed and an error returned.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doEnsembleMapDeleteRequest

eid	Numerical ensemble identifier
-----	-------------------------------

```

<wsdl:message name="doEnsembleMapDeleteRequest">
  <wsdl:part name="eid" type="xsd:int"/>
</wsdl:message>

```

**Output:** doEnsembleMapDeleteResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doEnsembleMapDeleteResponse">
  <wsdl:part name="doEnsembleMapDeleteReturn" type="tns1:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

### 4.23 Operation doEnsembleMapList

**Description:** Return content of table grpmap.

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE

**Input:** doEnsembleMapListRequest

Operation requires no arguments.

```
<wsdl:message name="doEnsembleMapListRequest">
</wsdl:message>
```

**Output:** doEnsembleMapListResponse

EID	Numerical ensemble identifier
ensembleURI	URI for ensemble
prjID	Numerical project identifier



```

<wsdl:message name="doEnsembleMapListResponse">
  <wsdl:part name="doEnsembleMapListReturn" type="tns1:ResponseEnsembleMap"/>
</wsdl:message>

<complexType name="ResponseEnsembleMap">
  <complexContent>
    <extension base="tns1:Response">
      <sequence>
        <element name="ensembleMap" nillable="true" type="impl:ArrayOf_tns3_Ensemble"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="ArrayOf_tns3_Ensemble">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="tns2:Ensemble[]" />
    </restriction>
  </complexContent>
</complexType>

<complexType name="Ensemble">
  <sequence>
    <element name="EID" type="xsd:int"/>
    <element name="ensembleURI" nillable="true" type="soapenc:string"/>
    <element name="prjID" type="xsd:int"/>
  </sequence>
</complexType>

```

#### 4.24 Operation doAdmInsert

**Description:** Add a new administrator identified by his/her `certID` to the table of administrators `adm`. Nothing is done if `certID` has already been added. If `certID` is not found in table `certmap` it will be added after checking VO membership. Operation fails if this check fails.

**Access:** secure [administrator privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doAdmInsertRequest

certID	Distinguished name of administrator's certificate
--------	---

```
<wsdl:message name="doAdmInsertRequest">
  <wsdl:part name="certID" type="xsd:string"/>
</wsdl:message>
```

**Output:** doAdmInsertResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doAdmInsertResponse">
  <wsdl:part name="doAdmInsertReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.25 Operation doAdmDelete

**Description:** Remove an administrator identified by the cid from the table of administrators adm. If operation succeeds or cid does not exist, status code MDC\_SUCCESS is returned, otherwise LDG\_MDC\_FAILED.

**Access:** secure [administrator privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doAdmDeleteRequest

cid	Numerical identifier for administrator's certificate
-----	--

```
<wsdl:message name="doAdmDeleteRequest">
  <wsdl:part name="cid" type="xsd:int"/>
</wsdl:message>
```

**Output:** doAdmDeleteResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doAdmDeleteResponse">
  <wsdl:part name="doAdmDeleteReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.26 Operation doAdmList

**Description:** List the numerical identifier and certificate's distinguished names of all administrators currently entered to the table of administrators adm.

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE

**Input:** doAdmListRequest

Operation requires no arguments.

```
<wsdl:message name="doAdmListRequest">
</wsdl:message>
```

**Output:** doAdmListResponse

statusCode	Return status string
CID	Numerical certificate identifier
certID	Certificate's distinguished name

```

<wsdl:message name="doAdmListResponse">
  <wsdl:part name="doAdmListReturn" type="tns1:ResponseCertList"/>
</wsdl:message>

<complexType name="ResponseCertList">
  <complexContent>
    <extension base="tns1:Response">
      <sequence>
        <element name="certList" nillable="true" type="impl:ArrayOf_tns3_Certificate"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="ArrayOf_tns3_Certificate">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="tns2:Certificate[]" />
    </restriction>
  </complexContent>
</complexType>

<complexType name="Certificate">
  <sequence>
    <element name="CID" type="xsd:int"/>
    <element name="certID" nillable="true" type="soapenc:string"/>
  </sequence>
</complexType>

```

## 4.27 Operation doManagerInsert

**Description:** Add a new manager for project prjName identified by his/her certID to the table of project managers. Nothing is done if certID has already been added for this project.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doManagerInsertRequest

certID	Distinguished name of manager's certificate
prjid	Number project identifier

```
<wsdl:message name="doManagerInsertRequest">
  <wsdl:part name="certID" type="xsd:string"/>
  <wsdl:part name="prjid" type="xsd:int"/>
</wsdl:message>
```

**Output:** doManagerInsertResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doManagerInsertResponse">
  <wsdl:part name="doManagerInsertReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.28 Operation doManagerDelete

**Description:** Remove an manager identified by the `certID` from the list of managers for the project identified by `prjName`.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doManagerDeleteRequest

cid	Numerical identifier for manager's certificate
prjid	Number project identifier

```
<wsdl:message name="doManagerDeleteRequest">
  <wsdl:part name="cid" type="xsd:int"/>
  <wsdl:part name="prjid" type="xsd:int"/>
</wsdl:message>
```

**Output:** doManagerDeleteResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doManagerDeleteResponse">
  <wsdl:part name="doManagerDeleteReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.29 Operation doManagerList

**Description:** List for all projects entered in table `manager` the `certID` for all managers.

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE

**Input:** doManagerListRequest

Operation requires no arguments.

```
<wsdl:message name="doManagerListRequest">
</wsdl:message>
```

**Output:** doManagerListResponse

statusCode	Return status string
cid	Numerical identifier of manager's certificate
certID	Distinguished name of manager's certificate
prjid	Numerical identifier of project
collaboration	Collaboration name
prjName	Project name

```

<wsdl:message name="doManagerListResponse">
  <wsdl:part name="doManagerListReturn" type="tns1:ResponseMngList"/>
</wsdl:message>

<complexType name="ResponseMngList">
  <complexContent>
    <extension base="tns1:Response">
      <sequence>
        <element name="mngMap" nillable="true" type="impl:ArrayOf_tns3_Mng"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="ArrayOf_tns3_Mng">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="tns2:Mng[]"/>
    </restriction>
  </complexContent>
</complexType>

<complexType name="Mng">
  <sequence>
    <element name="cert" nillable="true" type="tns2:Certificate"/>
    <element name="prj" nillable="true" type="tns2:Prj"/>
  </sequence>
</complexType>

```

### 4.30 Operation doGroupInsert

**Description:** Add a new person identified by `certID` to the group identified by `gid` after having checked that this group exists. Otherwise the insert operation is not performed and an error returned. If `certID` is not in table `certmap` it will be added after VO membership has been checked. Otherwise an error is returned.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doGroupInsertRequest

gid	Numerical group identifier
certID	Distinguished name of certificate

```
<wsdl:message name="doGroupInsertRequest">
  <wsdl:part name="gid" type="xsd:int"/>
  <wsdl:part name="certID" type="xsd:string"/>
</wsdl:message>
```

**Output:** doGroupInsertResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doGroupInsertResponse">
  <wsdl:part name="doGroupInsertReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

### 4.31 Operation doGroupDelete

**Description:** Delete a person identified by cid from a group identified by gid.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doGroupDeleteRequest

gid	Numerical group identifier
cid	Numerical certificate identifier

```
<wsdl:message name="doGroupDeleteRequest">
  <wsdl:part name="gid" type="xsd:int"/>
  <wsdl:part name="cid" type="xsd:int"/>
</wsdl:message>
```



**Output:** doGroupDeleteResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doGroupDeleteResponse">
  <wsdl:part name="doGroupDeleteReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

### 4.32 Operation doGroupList

**Description:** Return content of table grp.

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE

**Input:** doGroupListRequest

Operation requires no arguments.

```
<wsdl:message name="doGroupListRequest">
</wsdl:message>
```

**Output:** doGroupListResponse

statusCode	Return status string
gid	Numerical group identifier
cid	Numerical certificate identifier

```

<wsdl:message name="doGroupListResponse">
  <wsdl:part name="doGroupListReturn" type="tns1:ResponseGrpCertList"/>
</wsdl:message>

<complexType name="ResponseGrpCertList">
  <complexContent>
    <extension base="tns1:Response">
      <sequence>
        <element name="grpCertList" nillable="true" type="impl:ArrayOf_tns3_GrpCert"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="ArrayOf_tns3_GrpCert">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="tns2:GrpCert[]" />
    </restriction>
  </complexContent>
</complexType>

<complexType name="GrpCert">
  <sequence>
    <element name="CID" type="xsd:int"/>
    <element name="GID" type="xsd:int"/>
  </sequence>
</complexType>

```

### 4.33 Operation doAclInsert

**Description:** Insert ACL for a group identified by `gid` and an ensemble identified by `eid` after having checked that `gid` and `eid` do exist. Otherwise the insert operation is not performed and an error returned.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doAclInsertRequest

eid	Numerical ensemble identifier
gid	Numerical group identifier
writeRight	If true write access otherwise read access

```
<wsdl:message name="doAclInsertRequest">
  <wsdl:part name="eid" type="xsd:int"/>
  <wsdl:part name="writeRight" type="xsd:boolean"/>
  <wsdl:part name="gid" type="xsd:int"/>
</wsdl:message>
```

**Output:** doAclInsertResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doAclInsertResponse">
  <wsdl:part name="doAclInsertReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

#### 4.34 Operation doAclDelete

**Description:** Delete ACL entry for group gid and ensemble eid.

**Access:** secure [manager privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST

**Input:** doAclDeleteRequest

eid	Numerical ensemble identifier
gid	Numerical group identifier
writeRight	Write right flag

```

<wsdl:message name="doAclDeleteRequest">
  <wsdl:part name="grpName" type="xsd:string"/>
  <wsdl:part name="ensembleURI" type="xsd:string"/>
</wsdl:message>

```

**Output:** doAclDeleteResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doAclDeleteResponse">
  <wsdl:part name="doAclDeleteReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

#### 4.35 Operation doAclList

**Description:** Return content of table grp.

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE

**Input:** doAclListRequest

Operation requires no arguments.

```

<wsdl:message name="doAclListRequest">
</wsdl:message>

```

**Output:** doAclListResponse

statusCode	Return status string
EID	Numerical ensemble identifier
GID	Numerical group identifier
writeRight	Write access right

```

<wsdl:message name="doAclListResponse">
  <wsdl:part name="doAclListReturn" type="tns1:ResponseAcl"/>
</wsdl:message>

<complexType name="ResponseAcl">
  <complexContent>
    <extension base="tns1:Response">
      <sequence>
        <element name="acl" nillable="true" type="impl:ArrayOf_tns3_Acl"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="ArrayOf_tns3_Acl">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="tns2:Acl []"/>
    </restriction>
  </complexContent>
</complexType>

<complexType name="Acl">
  <sequence>
    <element name="EID" type="xsd:int"/>
    <element name="GID" type="xsd:int"/>
    <element name="writeRight" type="xsd:boolean"/>
  </sequence>
</complexType>

```

#### 4.36 Operation doDump

**Description:** Dump all documents (ensemble and metadata) to file.

**Access:** secure [administrator privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_LDG\_INVALID\_STATE

**Input:** doDump

Operation requires no arguments.

```
<wsdl:message name="doDumpRequest">
</wsdl:message>
```

**Output:** doDumpResponse

statusCode	Return status string
------------	----------------------

```
<wsdl:message name="doDumpResponse">
  <wsdl:part name="doDumpReturn" type="tns2:Response"/>
</wsdl:message>
```

#### 4.37 Operation doRestore

**Description:**

**Access:** secure [administrator privilege]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_LDG\_INVALID\_STATE

**Input:** doRestoreRequest

Operation requires no arguments.

```
<wsdl:message name="doRestoreRequest">
</wsdl:message>
```

**Output:** doRestoreResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doRestoreResponse">
  <wsdl:part name="doRestoreReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

### 4.38 Operation doSetConfig

**Description:**

**Access:** secure [administrator privileges]

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE, MDC\_LDG\_NO\_PRIVILEGE, MDC\_INVALID\_REQUEST, MDC\_LDG\_INVALID\_STATE

**Input:** doSetConfigRequest

name	Name of configuration variable
value	Value of configuration variable

```

<wsdl:message name="doSetConfigRequest">
  <wsdl:part name="name" type="xsd:string"/>
  <wsdl:part name="value" type="xsd:string"/>
</wsdl:message>

```

**Output:** doSetConfigResponse

statusCode	Return status string
------------	----------------------

```

<wsdl:message name="doSetConfigResponse">
  <wsdl:part name="doSetConfigReturn" type="tns2:Response"/>
</wsdl:message>

<complexType name="Response">
  <sequence>
    <element name="statusCode" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

### 4.39 Operation doGetConfig

**Description:**

**Access:** open

**Status codes:** MDC\_SUCCESS, MDC\_FAILURE

**Input:** doGetConfigRequest

Operation requires no arguments.

```

<wsdl:message name="doGetConfigRequest">
</wsdl:message>

```

**Output:** doGetConfigResponse

statusCode	Return status string
readEnabled	Reading enabled if true
status	Status string
writeEnabled	Writing enabled if true

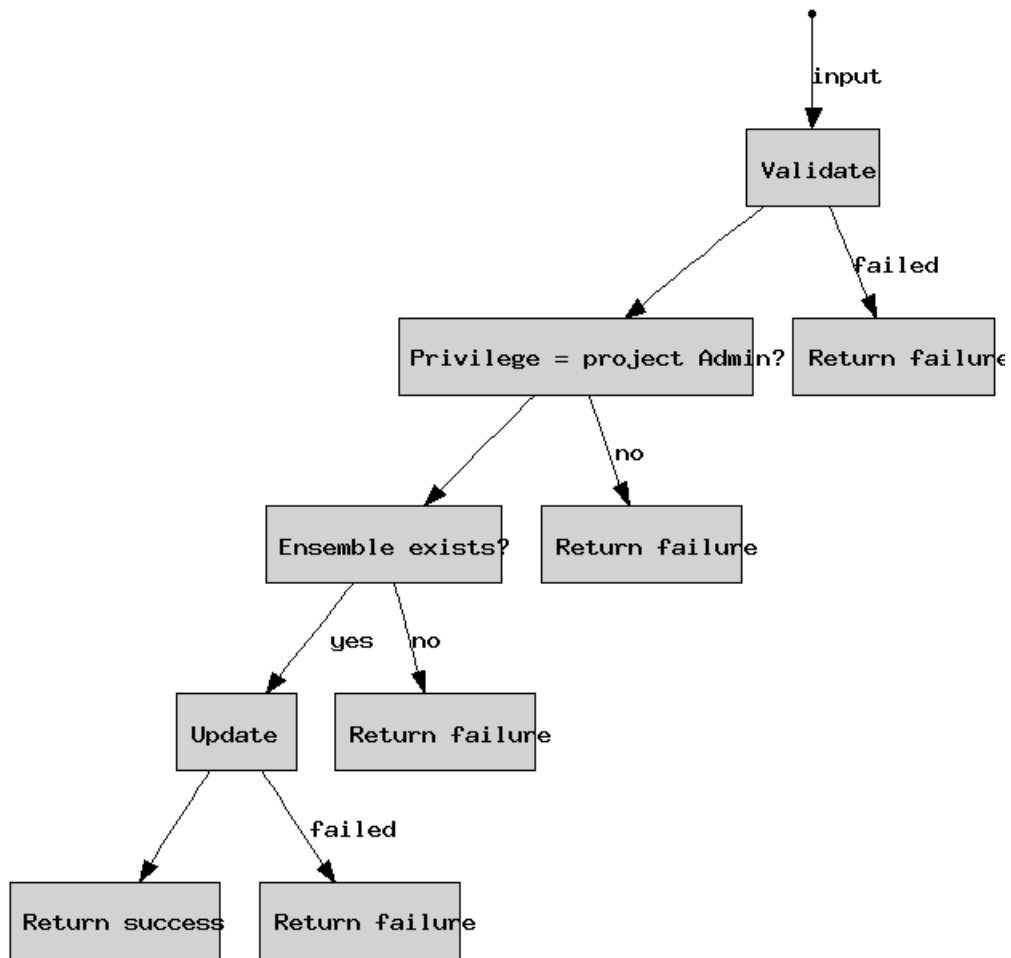


```
<wsdl:message name="doGetConfigResponse">
  <wsdl:part name="doGetConfigReturn" type="tns2:ResponseConfig"/>
</wsdl:message>

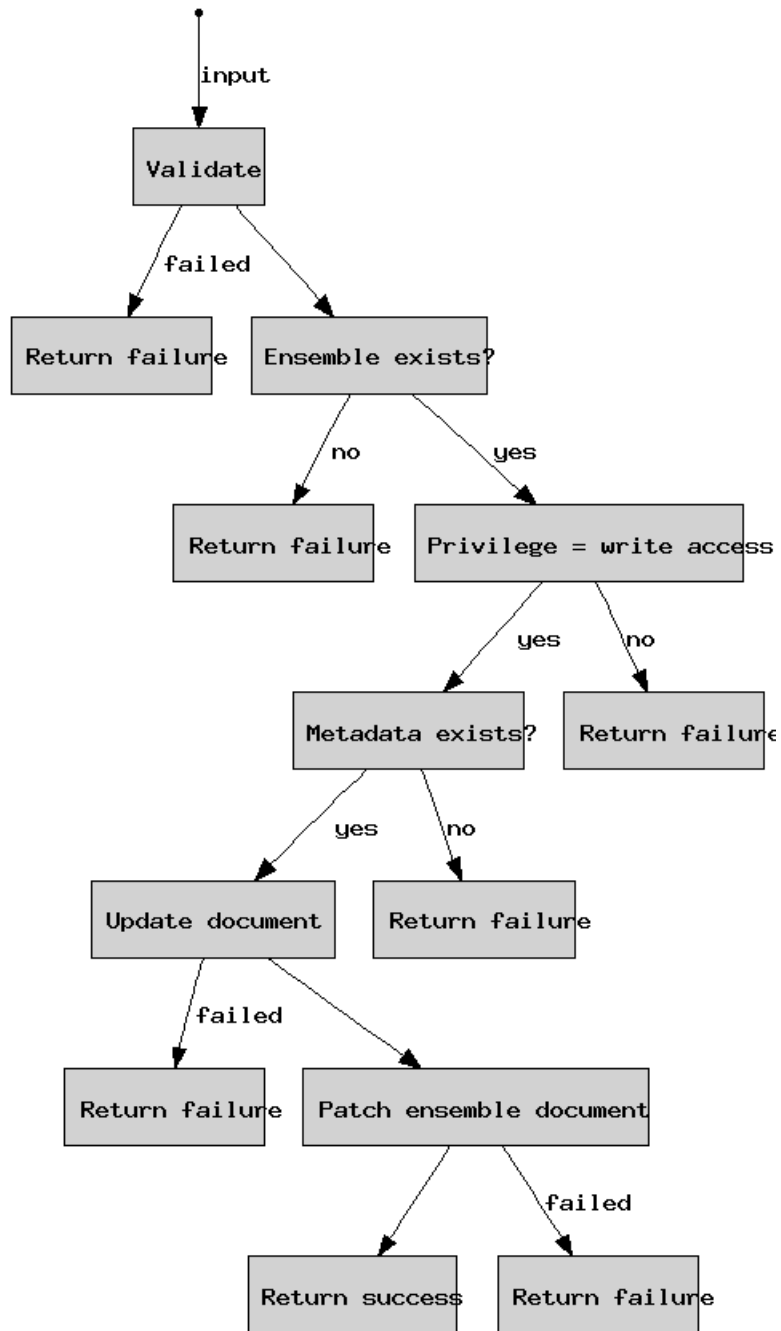
<complexType name="ResponseConfig">
  <complexContent>
    <extension base="tns1:Response">
      <sequence>
        <element name="readEnabled" type="xsd:boolean"/>
        <element name="statusDumpRestore" nillable="true" type="soapenc:string"/>
        <element name="writeEnabled" type="xsd:boolean"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

## 5 Activity diagrams

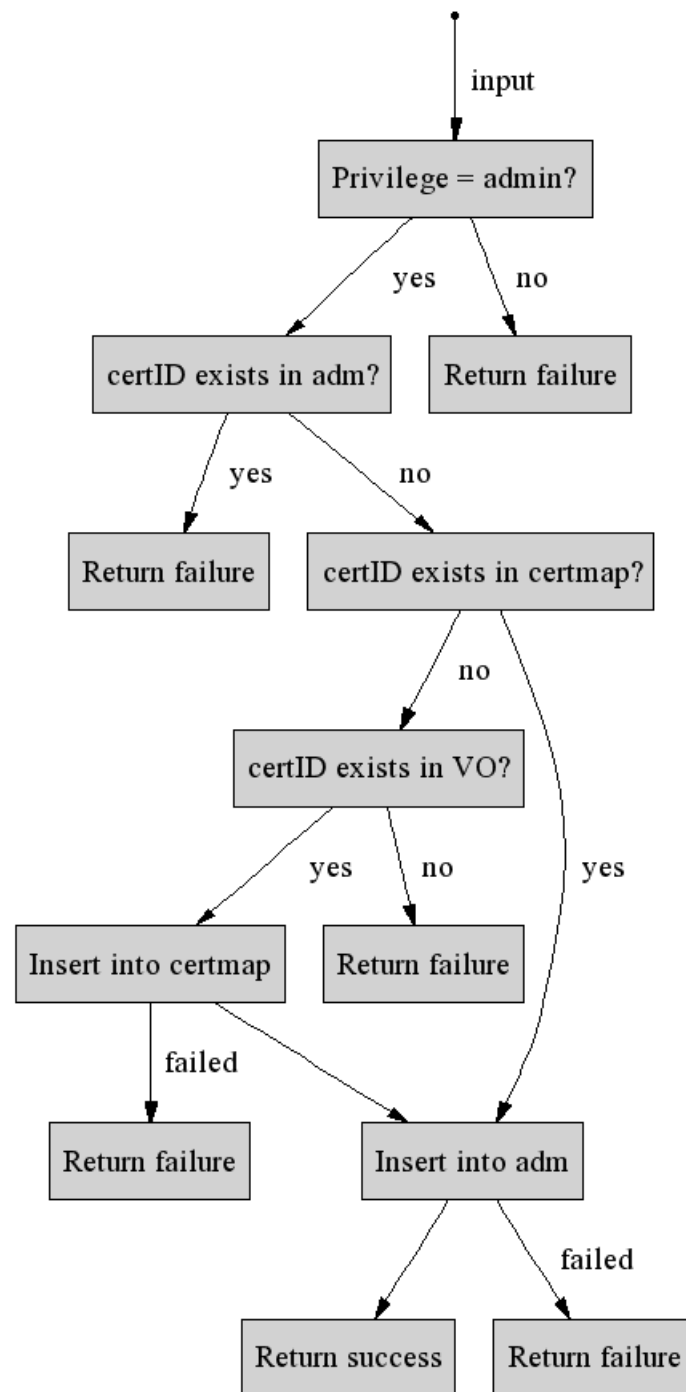
### 5.1 Operation doEnsembleUpdate



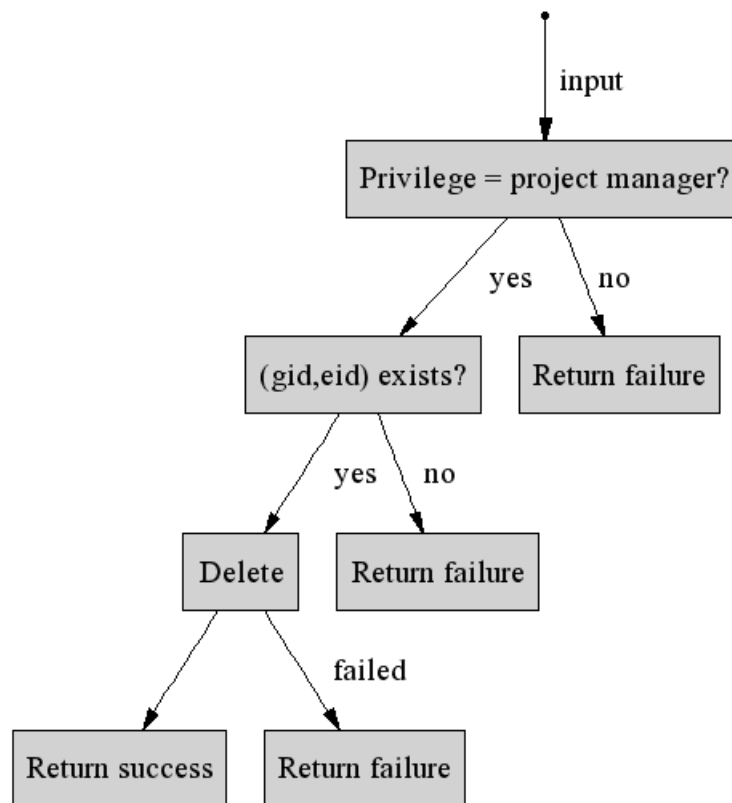
## 5.2 Operation doMetadataUpdate



### 5.3 Operation doAdmInsert



## 5.4 Operation doAclDelete



## 6 Changelog

### 6.1 Changes since 1.2

- Removed the services which have been standardized by ILDG.

### 6.2 Changes since 1.1

- Removed automatic insert feature from services doEnsembleUpdate and doMetadataUpdate
- Added new services doEnsembleInsert and doMetadataInsert
- Defined ILDG compliant return status codes

### 6.3 Changes since 1.0

- Added new services doCertMapUpdate, doCertMapDelete, doCertMapList
- Added new services doPrjMapInsert, doPrjMapUpdate, doPrjMapDelete, doPrjMapList

- Added new services `doGrpMapInsert`, `doGrpMapUpdate`, `doGrpMapDelete`, `doGrpMapList`
- Added new services `doEnsembleMapInsert`, `doEnsembleMapUpdate`, `doEnsembleMapDelete`, `doEnsembleMapList`
- Added new service `doAclList`
- Check for `ensembleURI` changed with respect to removed service `doProject*`
- Require additional check for `doAdmInsert`
- Changed arguments for `doAdmDelete`, `doManagerInsert`, `doManagerDelete`, `doGroupInsert`, `doGroupDelete`, `doAclDelete`
- Rename service `doAclUpdate` into `doAclInsert` and changed arguments.
- Changed return values for `doGroupList`
- Removed services `doProjectInsert`, `doAclGetByEnsemble`, `doAcsGet`