

AtlasMap Design Time Service API Reference

Version 1.35-SNAPSHOT, 2018-05-08

Design Time Service

1. Core	1
1.1. Paths.....	1
1.1.1. List FieldActions	1
1.1.2. Create Mapping	1
1.1.3. Validate Mapping	2
1.1.4. Update Mapping	2
1.1.5. Get Mapping	3
1.1.6. Remove Mapping.....	3
1.1.7. List Mappings.....	4
1.1.8. Ping	4
1.2. Definitions	5
1.2.1. ActionDetail	5
1.2.2. ActionDetails	6
1.2.3. ActionParameter	6
1.2.4. ActionParameters	6
1.2.5. AtlasMapping	7
1.2.6. BaseMapping	7
1.2.7. Constant	7
1.2.8. Constants.....	8
1.2.9. DataSource	8
1.2.10. LookupEntry	8
1.2.11. LookupTable	9
1.2.12. LookupTables.....	9
1.2.13. Mappings	10
1.2.14. Properties	10
1.2.15. Property	10
1.2.16. StringMap.....	10
1.2.17. StringMapEntry.....	10
1.2.18. Validation	11
1.2.19. Validations	11
2. Java	12
2.1. Paths	12
2.1.1. Inspect Class	12
2.1.2. Inspect Class.....	12
2.1.3. Generate Maven Classpath	13
2.1.4. Simple	13
2.2. Definitions	14
2.2.1. Actions.....	14

2.2.2. JavaClass	14
2.2.3. JavaEnumField	16
2.2.4. JavaEnumFields	18
2.2.5. JavaField	18
2.2.6. JavaFields	19
2.2.7. MavenClasspathResponse	20
2.2.8. ModifierList	20
2.2.9. StringList	20
3. JSON	21
3.1. Paths	21
3.1.1. Inspect JSON	21
3.1.2. Inspect JSON via URI	21
3.1.3. Simple	22
3.2. Definitions	22
3.2.1. Actions	22
3.2.2. Field	23
3.2.3. Fields	24
3.2.4. JsonDocument	24
3.2.5. JsonInspectionResponse	24
4. XML	25
4.1. Paths	25
4.1.1. Inspect XML	25
4.1.2. Inspect XML via URI	25
4.1.3. Simple	26
4.2. Definitions	26
4.2.1. Actions	26
4.2.2. Field	26
4.2.3. Fields	27
4.2.4. XmlDocument	28
4.2.5. XmlInspectionResponse	28
4.2.6. XmlNamespace	28
4.2.7. XmlNamespaces	29

Chapter 1. Core

1.1. Paths

1.1.1. List FieldActions

```
GET /v2/atlas/fieldActions
```

Description

Retrieves a list of available field action

Responses

HTTP Code	Description	Schema
200	Return a list of field action detail	ActionDetails

Produces

- `application/json`

1.1.2. Create Mapping

```
PUT /v2/atlas/mapping
```

Description

Save a mapping file on the server

Responses

HTTP Code	Description	Schema
200	Succeeded	No Content

Consumes

- `application/json`

Produces

- `application/json`

1.1.3. Validate Mapping

PUT /v2/atlas/mapping/validate

Description

Validate mapping file

Responses

HTTP Code	Description	Schema
200	Return a validation result	Validations

Consumes

- `application/json`

Produces

- `application/json`

1.1.4. Update Mapping

POST /v2/atlas/mapping/{mappingId}

Description

Update existing mapping file on the server

Parameters

Type	Name	Description	Schema
Path	mappingId <i>required</i>	Mapping ID	string

Responses

HTTP Code	Description	Schema
200	Succeeded	No Content

Consumes

- `application/json`

Produces

- `application/json`

1.1.5. Get Mapping

```
GET /v2/atlas/mapping/{mappingId}
```

Description

Retrieve a mapping file saved on the server

Parameters

Type	Name	Description	Schema
Path	mappingId <i>required</i>	Mapping ID	string

Responses

HTTP Code	Description	Schema
200	Return a mapping file content	AtlasMapping
204	Mapping file was not found	No Content

Produces

- `application/json`

1.1.6. Remove Mapping

```
DELETE /v2/atlas/mapping/{mappingId}
```

Description

Remove a mapping file saved on the server

Parameters

Type	Name	Description	Schema
Path	mappingId <i>required</i>	Mapping ID	string

Responses

HTTP Code	Description	Schema
200	Specified mapping file was removed successfully	No Content
204	Mapping file was not found	No Content

Produces

- `application/json`

1.1.7. List Mappings

```
GET /v2/atlas/mappings
```

Description

Retrieves a list of mapping file name saved on the server

Parameters

Type	Name	Schema
Query	filter <i>optional</i>	string

Responses

HTTP Code	Description	Schema
200	Return a list of a pair of mapping file name and content	StringMap

Produces

- `application/json`

1.1.8. Ping

GET /v2/atlas/ping

Description

Simple liveness check method used in liveness checks. Must not be protected via authentication.

Responses

HTTP Code	Description	Schema
200	Return 'pong'	string

1.2. Definitions

1.2.1. ActionDetail

Name	Schema
className <i>optional</i>	string
method <i>optional</i>	string
name <i>optional</i>	string
parameters <i>optional</i>	ActionParameters
sourceCollectionType <i>optional</i>	enum (ALL, ARRAY, LIST, MAP, NONE)
sourceType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
targetCollectionType <i>optional</i>	enum (ALL, ARRAY, LIST, MAP, NONE)

Name	Schema
targetType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)

1.2.2. ActionDetails

Name	Schema
actionDetail <i>optional</i>	< ActionDetail > array

1.2.3. ActionParameter

Name	Schema
description <i>optional</i>	string
displayName <i>optional</i>	string
fieldType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
name <i>optional</i>	string
values <i>optional</i>	< string > array

1.2.4. ActionParameters

Name	Schema
parameter <i>optional</i>	< ActionParameter > array

1.2.5. AtlasMapping

Name	Schema
constants <i>optional</i>	Constants
dataSource <i>required</i>	< DataSource > array
lookupTables <i>optional</i>	LookupTables
mappings <i>required</i>	Mappings
name <i>optional</i>	string
properties <i>optional</i>	Properties

1.2.6. BaseMapping

Name	Schema
alias <i>optional</i>	string
description <i>optional</i>	string
mappingType <i>optional</i>	enum (ALL, COLLECTION, COMBINE, LOOKUP, MAP, SEPARATE, NONE)

1.2.7. Constant

Name	Schema
fieldType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
name <i>optional</i>	string
value <i>optional</i>	string

1.2.8. Constants

Name	Schema
constant <i>optional</i>	< Constant > array

1.2.9. DataSource

Name	Schema
dataSourceType <i>optional</i>	enum (SOURCE, TARGET)
id <i>optional</i>	string
uri <i>optional</i>	string

1.2.10. LookupEntry

Name	Schema
sourceType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
sourceValue <i>optional</i>	string
targetType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
targetValue <i>optional</i>	string

1.2.11. LookupTable

Name	Schema
description <i>optional</i>	string
lookupEntry <i>optional</i>	< LookupEntry > array
name <i>optional</i>	string

1.2.12. LookupTables

Name	Schema
lookupTable <i>optional</i>	< LookupTable > array

1.2.13. Mappings

Name	Schema
mapping <i>optional</i>	< BaseMapping > array

1.2.14. Properties

Name	Schema
property <i>optional</i>	< Property > array

1.2.15. Property

Name	Schema
fieldType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
name <i>optional</i>	string
value <i>optional</i>	string

1.2.16. StringMap

Name	Schema
stringMapEntry <i>optional</i>	< StringMapEntry > array

1.2.17. StringMapEntry

Name	Schema
name <i>optional</i>	string

Name	Schema
value <i>optional</i>	string

1.2.18. Validation

Name	Schema
id <i>optional</i>	string
message <i>optional</i>	string
scope <i>optional</i>	enum (ALL, DATA_SOURCE, MAPPING, LOOKUP_TABLE, CONSTANT, PROPERTY)
status <i>optional</i>	enum (ALL, INFO, WARN, ERROR, NONE)

1.2.19. Validations

Name	Schema
validation <i>optional</i>	< Validation > array

Chapter 2. Java

2.1. Paths

2.1.1. Inspect Class

```
POST /v2/atlas/java/class
```

Description

Inspect a Java Class with specified fully qualified class name and return a Document object

Responses

HTTP Code	Description	Schema
200	Return a Document object represented by JavaClass	JavaClass

Consumes

- `application/json`

Produces

- `application/json`

2.1.2. Inspect Class

```
GET /v2/atlas/java/class
```

Description

Inspect a Java Class with specified fully qualified class name and return a Document object

Parameters

Type	Name	Description	Schema
Query	<code>className</code> <i>optional</i>	The fully qualified class name to inspect	string

Responses

HTTP Code	Description	Schema
200	Return a Document object represented by JavaClass	JavaClass

Produces

- `application/json`

2.1.3. Generate Maven Classpath

POST /v2/atlas/java/mavenclasspath

Description

Retrieve a maven classpath string

Responses

HTTP Code	Description	Schema
200	Return a MavenClasspathResponse object which contains classpath string	MavenClasspathResponse

Consumes

- `application/json`

Produces

- `application/json`

2.1.4. Simple

GET /v2/atlas/java/simple

Description

Simple hello service

Parameters

Type	Name	Description	Schema
Query	from <i>optional</i>	From	string

Responses

HTTP Code	Description	Schema
200	Return a response	string

Produces

- `text/plain`

2.2. Definitions

2.2.1. Actions

Type : object

2.2.2. JavaClass

Name	Schema
actions <i>optional</i>	Actions
anonymous <i>optional</i>	boolean
annotation <i>optional</i>	boolean
annotations <i>optional</i>	StringList
arrayDimensions <i>optional</i>	integer (int32)
arraySize <i>optional</i>	integer (int32)
className <i>optional</i>	string
collectionClassName <i>optional</i>	string

Name	Schema
collectionType <i>optional</i>	enum (ALL, ARRAY, LIST, MAP, NONE)
docId <i>optional</i>	string
enumeration <i>optional</i>	boolean
fieldType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
format <i>optional</i>	string
getMethod <i>optional</i>	string
index <i>optional</i>	integer (int32)
interface <i>optional</i>	boolean
javaEnumFields <i>required</i>	JavaEnumFields
javaFields <i>required</i>	JavaFields
localClass <i>optional</i>	boolean
memberClass <i>optional</i>	boolean
modifiers <i>required</i>	ModifierList

Name	Schema
name <i>optional</i>	string
packageName <i>optional</i>	string
parameterizedTypes <i>optional</i>	StringList
path <i>optional</i>	string
primitive <i>optional</i>	boolean
required <i>optional</i>	boolean
setMethod <i>optional</i>	string
status <i>optional</i>	enum (SUPPORTED, UNSUPPORTED, CACHED, ERROR, NOT_FOUND, BLACK_LIST)
synthetic <i>optional</i>	boolean
uri <i>optional</i>	string
value <i>optional</i>	object

2.2.3. JavaEnumField

Name	Schema
actions <i>optional</i>	Actions
arrayDimensions <i>optional</i>	integer (int32)

Name	Schema
arraySize <i>optional</i>	integer (int32)
className <i>optional</i>	string
collectionType <i>optional</i>	enum (ALL, ARRAY, LIST, MAP, NONE)
docId <i>optional</i>	string
fieldType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
format <i>optional</i>	string
index <i>optional</i>	integer (int32)
name <i>optional</i>	string
ordinal <i>optional</i>	integer (int32)
path <i>optional</i>	string
required <i>optional</i>	boolean
status <i>optional</i>	enum (SUPPORTED, UNSUPPORTED, CACHED, ERROR, NOT_FOUND, BLACK_LIST)
value <i>optional</i>	object

2.2.4. JavaEnumFields

Name	Schema
javaEnumField <i>optional</i>	< JavaEnumField > array

2.2.5. JavaField

Name	Schema
actions <i>optional</i>	Actions
annotations <i>optional</i>	StringList
arrayDimensions <i>optional</i>	integer (int32)
arraySize <i>optional</i>	integer (int32)
className <i>optional</i>	string
collectionClassName <i>optional</i>	string
collectionType <i>optional</i>	enum (ALL, ARRAY, LIST, MAP, NONE)
docId <i>optional</i>	string
fieldType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
format <i>optional</i>	string

Name	Schema
getMethod <i>optional</i>	string
index <i>optional</i>	integer (int32)
modifiers <i>required</i>	ModifierList
name <i>optional</i>	string
parameterizedTypes <i>optional</i>	StringList
path <i>optional</i>	string
primitive <i>optional</i>	boolean
required <i>optional</i>	boolean
setMethod <i>optional</i>	string
status <i>optional</i>	enum (SUPPORTED, UNSUPPORTED, CACHED, ERROR, NOT_FOUND, BLACK_LIST)
synthetic <i>optional</i>	boolean
value <i>optional</i>	object

2.2.6. JavaFields

Name	Schema
javaField <i>optional</i>	< JavaField > array

2.2.7. MavenClasspathResponse

Name	Schema
classpath <i>optional</i>	string
errorMessage <i>optional</i>	string
executionTime <i>optional</i>	integer (int64)

2.2.8. ModifierList

Name	Schema
modifier <i>optional</i>	< enum (ALL, ABSTRACT, FINAL, INTERFACE, NATIVE, PACKAGE_PRIVATE, PUBLIC, PROTECTED, PRIVATE, STATIC, STRICT, SYNCHRONIZED, TRANSIENT, VOLATILE, NONE) > array

2.2.9. StringList

Name	Schema
string <i>optional</i>	< string > array

Chapter 3. JSON

3.1. Paths

3.1.1. Inspect JSON

```
POST /v2/atlas/json/inspect
```

Description

Inspect a JSON schema or instance and return a Document object

Responses

HTTP Code	Description	Schema
200	Return a Document object represented by JsonDocument	JsonInspectionResponse

Consumes

- `application/json`

Produces

- `application/json`

3.1.2. Inspect JSON via URI

```
GET /v2/atlas/json/inspect
```

Description

NOT IMPLEMENTED Inspect a JSON schema or instance located at specified URI and return a Document object

Parameters

Type	Name	Description	Schema
Query	<code>type</code> <i>optional</i>		string

Type	Name	Description	Schema
Query	uri <i>optional</i>	URI for JSON schema or instance	string

Responses

HTTP Code	Description	Schema
200	Return a Document object represented by JsonDocument	JsonDocument

Produces

- `application/json`

3.1.3. Simple

```
GET /v2/atlas/json/simple
```

Description

Simple hello service

Parameters

Type	Name	Description	Schema
Query	from <i>optional</i>	From	string

Responses

HTTP Code	Description	Schema
200	Return a response	string

Produces

- `text/plain`

3.2. Definitions

3.2.1. Actions

Type : object

3.2.2. Field

Name	Schema
actions <i>optional</i>	Actions
arrayDimensions <i>optional</i>	integer (int32)
arraySize <i>optional</i>	integer (int32)
collectionType <i>optional</i>	enum (ALL, ARRAY, LIST, MAP, NONE)
docId <i>optional</i>	string
fieldType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
format <i>optional</i>	string
index <i>optional</i>	integer (int32)
path <i>optional</i>	string
required <i>optional</i>	boolean
status <i>optional</i>	enum (SUPPORTED, UNSUPPORTED, CACHED, ERROR, NOT_FOUND, BLACK_LIST)
value <i>optional</i>	object

3.2.3. Fields

Name	Schema
field <i>optional</i>	< Field > array

3.2.4. JsonDocument

Name	Schema
fields <i>required</i>	Fields

3.2.5. JsonInspectionResponse

Name	Schema
errorMessage <i>optional</i>	string
executionTime <i>optional</i>	integer (int64)
jsonDocument <i>optional</i>	JsonDocument

Chapter 4. XML

4.1. Paths

4.1.1. Inspect XML

```
POST /v2/atlas/xml/inspect
```

Description

Inspect a XML schema or instance and return a Document object

Responses

HTTP Code	Description	Schema
200	Return a Document object represented by XmlDocument	XmlInspectionResponse

Consumes

- `application/json`

Produces

- `application/json`

4.1.2. Inspect XML via URI

```
GET /v2/atlas/xml/inspect
```

Description

Inspect a XML schema or instance located at specified URI and return a Document object

Parameters

Type	Name	Schema
Query	type <i>optional</i>	string
Query	uri <i>optional</i>	string

Responses

HTTP Code	Description	Schema
200	Return a Document object represented by XmlDocument	XmlDocument

Produces

- `application/json`

4.1.3. Simple

```
GET /v2/atlas/xml/simple
```

Description

Simple hello service

Parameters

Type	Name	Schema
Query	from <i>optional</i>	string

Responses

HTTP Code	Description	Schema
200	Return a response	string

Produces

- `text/plain`

4.2. Definitions

4.2.1. Actions

Type : object

4.2.2. Field

Name	Schema
actions <i>optional</i>	Actions
arrayDimensions <i>optional</i>	integer (int32)
arraySize <i>optional</i>	integer (int32)
collectionType <i>optional</i>	enum (ALL, ARRAY, LIST, MAP, NONE)
docId <i>optional</i>	string
fieldType <i>optional</i>	enum (ANY, ANY_DATE, BIG_INTEGER, BOOLEAN, BYTE, BYTE_ARRAY, CHAR, COMPLEX, DATE, DATE_TIME, DATE_TIME_TZ, DATE_TZ, DECIMAL, DOUBLE, FLOAT, INTEGER, LONG, NONE, NUMBER, SHORT, STRING, TIME, TIME_TZ, UNSIGNED_BYTE, UNSIGNED_INTEGER, UNSIGNED_LONG, UNSIGNED_SHORT, UNSUPPORTED)
format <i>optional</i>	string
index <i>optional</i>	integer (int32)
path <i>optional</i>	string
required <i>optional</i>	boolean
status <i>optional</i>	enum (SUPPORTED, UNSUPPORTED, CACHED, ERROR, NOT_FOUND, BLACK_LIST)
value <i>optional</i>	object

4.2.3. Fields

Name	Schema
field <i>optional</i>	< Field > array

4.2.4. XmlDocument

Name	Schema
fields <i>required</i>	Fields
xmlNamespaces <i>required</i>	XmlNamespaces

4.2.5. XmlInspectionResponse

Name	Schema
errorMessage <i>optional</i>	string
executionTime <i>optional</i>	integer (int64)
xmlDocument <i>optional</i>	XmlDocument

4.2.6. XmlNamespace

Name	Schema
alias <i>optional</i>	string
locationUri <i>optional</i>	string
targetNamespace <i>optional</i>	boolean
uri <i>optional</i>	string

4.2.7. XmlNamespaces

Name	Schema
xmlns <i>optional</i>	< XmlNamespace > array