

AtgSolidRdb

Solid database service

Table of contents

1 Overview.....	2
2 Design.....	2
3 Constraints.....	2
3.1 Allowed Child Dependencies.....	2
3.2 Allowed Parent Dependencies.....	3
4 Attributes.....	3
4.1 Exported Attributes.....	3
4.2 Defaults for Imported Attributes.....	3
5 Commands.....	3
5.1 Docs-Generate.....	3
5.2 assertServiceIsDown.....	4
5.3 assertServiceIsUp.....	4
5.4 startService.....	4
5.5 stopService.....	5

1. Overview

AtgSolidRdb: *Solid database service*

2. Design

Super Type

[AtgRdb](#)

Role	Concrete. (Objects can be created.)
Instance Names	Unique
Notification	false
Template Directory	
Data View	Children, proximity: 1
Logger Name	

3. Constraints

3.1. Allowed Child Dependencies

- [AtgCatalogRdbSchema](#) 1
- AtgDynamoHome1
- AtgDynamoRoot1
- AtgJavaHome1
- [AtgLicenseZip](#) 1
- AtgLocalConfig1
- [AtgPlatformInstaller](#) 1
- AtgRdbConnection1
- [AtgRdbData](#)
- AtgRdbDriver1
- AtgRdbInstanceName1
- AtgRdbPassword1
- AtgRdbPort1
- AtgRdbUserName1
- [AtgSolidRdbSchema](#)
- JavaBin1

1: These types have a *Singleton* constraint. Only one instance may be added as a resource.

3.2. Allowed Parent Dependencies

- [AtgAppServer](#)
- AtgCatalogBuilder
- [AtgDataBuilder](#)
- [AtgSite](#)
- Node

4. Attributes

4.1. Exported Attributes

Name	Property
atgrdb_install_root	deployment-install-root

4.2. Defaults for Imported Attributes

Name	Default
atgLocalConfig	\${entity.attribute.atg_home}/home/localconfig
atgRdbDriver	solid.jdbc.SolidDriver

5. Commands

Note:

Commandline options displayed in square brackets "["] are optional. If an option expects arguments, then angle brackets are shown after the option "<>". Any default value is shown within the brackets.

5.1. Docs-Generate

Generates the Solid database initialization file

Usage

```
Docs-Generate [-archivedir <${entity.instance.dir}/var>]
               [-buildstamp <>]
```

Options

Option	Description
archivedir	<i>dir to store current output docs</i>

buildstamp	<i>build identifier. currently ignored</i>
------------	--

5.2. assertServiceIsDown

Checks that the Solid database server process is stopped

Usage

`assertServiceIsDown [-basedir <>]`

Execution	bash
Arguments	if [\${os.arch} = sparc]; then PS=/usr/ucb/ps; else PS=ps; fi ; \$PS auxww grep "\${opts.basedir}\\$" grep -v grep && exit 1 exit 0

Options

Option	Description
basedir	<i>Base directory of the Solid database instance</i>

5.3. assertServiceIsUp

Checks that the Solid database server process is running

Usage

`assertServiceIsUp [-basedir <>]`

Execution	bash
Arguments	if [\${os.arch} = sparc]; then PS=/usr/ucb/ps; else PS=ps; fi ; \$PS auxww grep "\${opts.basedir}\\$" grep -v grep && exit 0 exit 1

Options

Option	Description
basedir	<i>Base directory of the Solid database instance</i>

5.4. startService

Starts the Solid database server process

Usage

`startService [-dynamoroot <>]`

Execution	bash
Arguments	<code>\${opts.dynamoroot}/home/bin/startSolid</code>

Options

Option	Description
dynamoroot	<i>ATG home directory</i>

5.5. stopService

Stops the Solid database server process

Usage

`stopService [-basedir <>]`

Execution	bash
Arguments	if [\${os.arch} = sparc]; then PS=/usr/ucb/ps; else PS=ps; fi ; \$PS auxww grep "\${opts.basedir}\\$" grep -v grep awk '{print \$2}' xargs kill -9

Options

Option	Description
basedir	<i>Base directory of the Solid database instance</i>