This reference describes the RESTful domain runtime resources that are used to manage WebLogic Server Multitenant.
Contents

Preface .................................................................................................................................................................. xi

1 Resources

/management/weblogic......................................................................................................................... 1-2
/management/weblogic/{version} ........................................................................................................ 1-6
/management/weblogic/{version}/currentUser .................................................................................... 1-10
/management/weblogic/{version}/domainRuntime .............................................................................. 1-13
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes ...... 1-16
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}... 1-18

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/
appRuntimeStateRuntime 1-20

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/
appRuntimeStateRuntime/getCurrentState 1-23

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/
appRuntimeStateRuntime/getIntendedState 1-26

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/
appRuntimeStateRuntime/getModuleIds 1-28

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/
appRuntimeStateRuntime/getModuleTargets 1-30

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/
appRuntimeStateRuntime/getModuleType 1-33

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/
appRuntimeStateRuntime/getMultiVersionStateOnDemand 1-35

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/
appRuntimeStateRuntime/getRetireTimeMillis 1-37
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects 1-82

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects/{name} 1-84

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects/{name}/cancel 1-86

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects/{name}/getExceptions 1-88

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/distribute 1-90

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes 1-94

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes/{name} 1-96

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes/{name}/redeploy 1-98

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes/{name}/undeploy 1-101

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/purgeCompletedDeploymentProgressObjects 1-104

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/redeploy 1-106

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/undeploy 1-109

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/update 1-112

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager 1-114

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurationCreateForm 1-116

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations 1-118
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/getState 1-161

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/purgeTasks 1-163

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/resume 1-165

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/shutdown 1-168

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/start 1-176

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/startInAdmin 1-179

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/suspend 1-182

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks 1-187

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name} 1-189

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/cancel 1-191

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks 1-193

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name} 1-195

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name}/cancel 1-197

/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name}/subTasks 1-199
## 2 Entities

<table>
<thead>
<tr>
<th>Class</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>java.lang.Throwable</td>
<td>2-1</td>
</tr>
<tr>
<td>java.util.Properties</td>
<td>2-1</td>
</tr>
<tr>
<td>weblogic.health.HealthState</td>
<td>2-1</td>
</tr>
<tr>
<td>weblogic.health.Symptom</td>
<td>2-2</td>
</tr>
<tr>
<td>weblogic.management.rest.bean.model.CurrentUser</td>
<td>2-2</td>
</tr>
<tr>
<td>weblogic.management.rest.bean.model.Version</td>
<td>2-3</td>
</tr>
<tr>
<td>weblogic.management.runtime.AppDeploymentRuntimeMBean</td>
<td>2-3</td>
</tr>
<tr>
<td>weblogic.management.runtime.AppRuntimeStateRuntimeMBean</td>
<td>2-4</td>
</tr>
<tr>
<td>weblogic.management.runtime.DeploymentManagerMBean</td>
<td>2-4</td>
</tr>
<tr>
<td>weblogic.management.runtime.DeploymentProgressObjectMBean</td>
<td>2-5</td>
</tr>
<tr>
<td>weblogic.management.runtime.DomainPartitionRuntimeMBean</td>
<td>2-7</td>
</tr>
<tr>
<td>weblogic.management.runtime.DomainRuntimeMBean</td>
<td>2-8</td>
</tr>
<tr>
<td>weblogic.management.runtime.EditSessionConfigurationManagerMBean</td>
<td>2-8</td>
</tr>
<tr>
<td>weblogic.management.runtime.EditSessionConfigurationRuntimeMBean</td>
<td>2-8</td>
</tr>
<tr>
<td>weblogic.management.runtime.LibDeploymentRuntimeMBean</td>
<td>2-10</td>
</tr>
<tr>
<td>weblogic.management.runtime.PartitionLifeCycleRuntimeMBean</td>
<td>2-11</td>
</tr>
<tr>
<td>weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean</td>
<td>2-12</td>
</tr>
<tr>
<td>weblogic.management.runtime.PartitionUserFileSystemManagerMBean</td>
<td>2-14</td>
</tr>
<tr>
<td>weblogic.management.runtime.ResourceGroupLifeCycleRuntimeMBean</td>
<td>2-14</td>
</tr>
<tr>
<td>weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean</td>
<td>2-15</td>
</tr>
<tr>
<td>weblogic.management.runtime.RolloutServiceRuntimeMBean</td>
<td>2-17</td>
</tr>
<tr>
<td>weblogic.management.runtime.ServerRuntimeMBean</td>
<td>2-17</td>
</tr>
<tr>
<td>weblogic.management.runtime.TaskRuntimeMBean</td>
<td>2-19</td>
</tr>
<tr>
<td>weblogic.management.runtime.WorkflowTaskRuntimeMBean</td>
<td>2-20</td>
</tr>
</tbody>
</table>
This preface describes the document accessibility features and conventions used in this guide, RESTful Domain Runtime Reference for Oracle WebLogic Server Multitenant.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>boldface</td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td>italic</td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
This documentation describes the Oracle WebLogic Server Multitenant RESTful domain runtime resources that are used to monitor the specified WLS domain partition.

For a complete listing of the WLS REST reference documents and descriptions of their use, see *Administering Oracle WebLogic Server with RESTful Management Services*. 
This resource is used to access the different versions of the resources in this domain. The latest version is 12.2.1.3.0. The other versions, which are deprecated, are 12.2.1.0.0, 12.2.1.1.0, 12.2.1.2.0.

For future releases, latest always refers to the most recent release.

All new and non-deprecated MBean features are available in all versions. Any MBean features that were deprecated in a version will not be available in that version or later versions, but will still be available using earlier versions.

To always use the latest resources, use latest in the URLs. Note that this might require updating a client when moving to a newer version of WebLogic Server. To fix the version so that an update is not required, use 12.2.1.3.0 in the URLs.

The resource supports the following methods:

- GET
GET

Get this collection of weblogic.management.rest.bean.model.Version instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of weblogic.management.rest.bean.model.Version entities. This method can return the following links:

- uri=/management/weblogic rel=canonical
  This resource.
- uri=/management/weblogic/12.2.1.3.0 rel=current
  Access the latest version of the WLS resources in the this domain.
- uri=/management/weblogic rel=self
  This resource.

Example

Example 1 View the versions collection.

This example uses the GET method to display the version instances in this collection.

Example Request

curl -v \
--user admin:admin123 \
-H X-Requested-By:MyClient \
-H Accept:application/json \

Example Response

HTTP/1.1 200 OK

Response Body:

```json
{
  'links': [
    {
      'rel': 'self',
      'href': "http://localhost:7001/management/weblogic"
    },
    {
      'rel': 'canonical',
      'href': "http://localhost:7001/management/weblogic"
    },
    {
      'rel': 'current',
      'href': "http://localhost:7001/management/weblogic/12.2.1.3.0"
    }
  ],
  'items': [
    {
```
GET

"links": [
{
  "rel": "self",
  "href": "http:\/\/localhost:7001/management/weblogic/12.2.1.0.0"
},
{
  "rel": "canonical",
  "href": "http:\/\/localhost:7001/management/weblogic/12.2.1.0.0"
}],
"version": "12.2.1.0.0",
"isLatest": false,
"lifecycle": "deprecated"
},
{
  "links": [
  {
    "rel": "self",
    "href": "http:\/\/localhost:7001/management/weblogic/12.2.1.1.0"
  },
  {
    "rel": "canonical",
    "href": "http:\/\/localhost:7001/management/weblogic/12.2.1.1.0"
  }
  ],
  "version": "12.2.1.1.0",
  "isLatest": false,
  "lifecycle": "deprecated"
},
{
  "links": [
  {
    "rel": "self",
    "href": "http:\/\/localhost:7001/management/weblogic/12.2.1.2.0"
  },
  {
    "rel": "canonical",
    "href": "http:\/\/localhost:7001/management/weblogic/12.2.1.2.0"
  }
  ],
  "version": "12.2.1.2.0",
  "isLatest": false,
  "lifecycle": "deprecated"
},
{
  "links": [
  {
    "rel": "self",
    "href": "http:\/\/localhost:7001/management/weblogic/12.2.1.3.0"
  },
  {
    "rel": "canonical",
    "href": "http:\/\/localhost:7001/management/weblogic/12.2.1.3.0"
  }
  ],
  "version": "12.2.1.3.0",
  "isLatest": false,
  "lifecycle": "deprecated"
}
"http:\/localhost:7001/management/weblogic/12.2.1.3.0"
}
},
"version": "12.2.1.3.0",
"isLatest": true,
"lifecycle": "active"
}
This resource describes the resource trees that are used to manage a WebLogic Server domain and its managed servers. The resource trees mimic the bean trees in WLST and allow you to view and edit configuration and monitor runtime behavior.

The resource supports the following methods:

- **GET**
GET

Get this `weblogic.management.rest.bean.model.Version` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.rest.bean.model.Version` entity.

This method can return the following links:

- `uri=/management/weblogic/{version} rel=canonical`
  
  This resource.

- `uri=/management/weblogic/{version}/currentUser rel=currentUser`

- `uri=/management/weblogic/{version}/domainConfig rel=domainConfig`

- `uri=/management/weblogic/{version}/domainRuntime rel=domainRuntime`

- `uri=/management/weblogic/{version}/edit rel=edit`

- `uri=/management/weblogic rel=parent`
  
  The parent resource of this resource.

- `uri=/management/weblogic/{version} rel=self`
  
  This resource.

- `uri=/management/weblogic/{version}/serverConfig rel=serverConfig`

- `uri=/management/weblogic/{version}/serverRuntime rel=serverRuntime`

Example

**Example 1 View the latest version.**

This example uses the GET method to display the latest version.

Example Request

```bash
curl -v \
--user admin:admin123 \
-H X-Requested-By:MyClient \
-H Accept:application/json \
```

Example Response

HTTP/1.1 200 OK

Response Body:

```
{
  'links': [
    {
      'rel': 'parent',
      'href': "http://localhost:7001/management/weblogic"
    },
    {
```
Example 2  View a specific version.
This example uses the GET method to display a specific version.

Example Request

curl -v
--user admin:admin123
-X Requested-By:MyClient
-A Accept:application/json
-X GET http://localhost:7001/management/weblogic/12.2.1.3.0

Example Response

HTTP/1.1 200 OK

Response Body:
{
  "links": [
/management/weblogic/{version}/currentUser

This resource represents an authenticated user in the domain. The resource supports the following methods:

- GET
GET

Get this weblogic.management.rest.bean.model.CurrentUser instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a weblogic.management.rest.bean.model.CurrentUser entity.

This method can return the following links:

- uri="/management/weblogic/{version}/currentUser" rel=canonical
  This resource.
- uri="/management/weblogic/{version}" rel=parent
  The parent resource of this resource.
- uri="/management/weblogic/{version}/currentUser" rel=self
  This resource.

Example

Example 1  Viewing the weblogic.management.rest.bean.model.CurrentUser Instance

This example uses the GET method to view this weblogic.management.rest.bean.model.CurrentUser instance.

Example Request

curl -v \
  --user admin:admin123 \
  -H X-Requested-By:MyClient \
  -H Accept:application/json \

Example Response

HTTP/1.1 200 OK

Response Body:

```json
{
  "links": [
    {
      "rel": "parent",
      "href": "http:\//localhost:7001/management/weblogic/latest"
    },
    {
      "rel": "self",
      "href": "http:\//localhost:7001/management/weblogic/latest/currentUser"
    },
    {
      "rel": "canonical",
      "href": "http:\//localhost:7001/management/weblogic/latest/currentUser"
    }
  ]
}
```
GET

},
"partitionId": "0",
"partitionName": "DOMAIN",
"user": "admin"
}
/management/weblogic/{version}/domainRuntime

This resource is used to monitor the entire WebLogic Server domain.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.DomainRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.DomainRuntimeMBean` entity.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime` `rel=canonical`
  This resource.

- `rel=currentDomainPartitionRuntime`
  This resource's currentDomainPartitionRuntime reference to a `weblogic.management.runtime.DomainPartitionRuntimeMBean` resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes` `rel=domainPartitionRuntimes`
  This resource's domainPartitionRuntimes collection resource.

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService` `rel=rolloutService`
  This resource's rolloutService singleton resource.

- `uri=/management/weblogic/{version}/domainRuntime/search` `rel=search`
  This resource's search resource.

- `uri=/management/weblogic/{version}/domainRuntime` `rel=self`
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/serverRuntimes` `rel=serverRuntimes`
  This resource's serverRuntimes resource.

Example

Example 1  Viewing the `weblogic.management.runtime.DomainRuntimeMBean` Instance

This example uses the GET method to view this `weblogic.management.runtime.DomainRuntimeMBean` instance.

Example Request

```
curl -v \
  --user BUCAadmin:BUCAadmin123 \
  -H X-Requested-By:MyClient \
  -H Accept:application/json \
```
Example Response

HTTP/1.1 200 OK

Response Body:

```json
{
    "links": [
        {
            "rel": "parent",
            "href": "http://localhost:7001/BUC/management/weblogic/latest"
        },
        {
            "rel": "currentDomainPartitionRuntime",
            "href": "http://localhost:7001/BUC/management/weblogic/latest/domainRuntime/domainPartitionRuntimes/BUCPartition"
        },
        {
            "rel": "self",
            "href": "http://localhost:7001/BUC/management/weblogic/latest/domainRuntime"
        },
        {
            "rel": "canonical",
            "href": "http://localhost:7001/BUC/management/weblogic/latest/domainRuntime"
        },
        {
            "rel": "domainPartitionRuntimes",
            "href": "http://localhost:7001/BUC/management/weblogic/latest/domainRuntime/domainPartitionRuntimes"
        },
        {
            "rel": "serverRuntimes",
        },
        {
            "rel": "search",
            "href": "http://localhost:7001/BUC/management/weblogic/latest/domainRuntime/search"
        }
    ],
    "identity": [],
    "name": "bean",
    "type": "DomainRuntime",
    "currentDomainPartitionRuntime": [
        "domainPartitionRuntimes",
        "BUCPartition"
    ]
}
```
This resource manages a collection of
weblogic.management.runtime.DomainPartitionRuntimeMBean instances.

The resource supports the following methods:

- GET
GET

Get this collection of weblogic.management.runtime.DomainPartitionRuntimeMBean instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of weblogic.management.runtime.DomainPartitionRuntimeMBean entities.

This method can return the following links:

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes rel=canonical
  This resource.
- uri=/management/weblogic/{version}/domainRuntime rel=parent
  The parent resource of this resource.
- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes rel=self
  This resource.
This resource manages a `weblogic.management.runtime.DomainPartitionRuntimeMBean` instance.

The resource supports the following methods:

- **GET**
GET

Get this `weblogic.management.runtime.DomainPartitionRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.DomainPartitionRuntimeMBean` entity.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime rel=appRuntimeStateRuntime`
  This resource's `appRuntimeStateRuntime` singleton resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name} rel=canonical`
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager rel=deploymentManager`
  This resource's `deploymentManager` singleton resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager rel=editSessionConfigurationManager`
  This resource's `editSessionConfigurationManager` singleton resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/parent rel=parent`
  The parent resource of this resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime rel=partitionLifeCycleRuntime`
  This resource's `partitionLifeCycleRuntime` singleton resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionUserFileSystemManager rel=partitionUserFileSystemManager`
  This resource's `partitionUserFileSystemManager` singleton resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/rel=self`
  This resource.
This resource manages a `weblogic.management.runtime.AppRuntimeStateRuntimeMBean` instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.AppRuntimeStateRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.AppRuntimeStateRuntimeMBean` entity.

This method can return the following links:

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getCurrentState`  
  **rel** = action  
  **title** = getCurrentState

  This resource's getCurrentState action resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getIntendedState`  
  **rel** = action  
  **title** = getIntendedState

  This resource's getIntendedState action resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getModuleIds`  
  **rel** = action  
  **title** = getModuleIds

  This resource's getModuleIds action resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getModuleTargets`  
  **rel** = action  
  **title** = getModuleTargets

  This resource's getModuleTargets action resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getModuleType`  
  **rel** = action  
  **title** = getModuleType

  This resource's getModuleType action resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getMultiVersionStateOnDemand`  
  **rel** = action  
  **title** = getMultiVersionStateOnDemand

  This resource's getMultiVersionStateOnDemand action resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getRetireTimeMillis`  
  **rel** = action  
  **title** = getRetireTimeMillis

  This resource's getRetireTimeMillis action resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getRetireTimeoutSeconds`  
  **rel** = action  
  **title** = getRetireTimeoutSeconds

  This resource's getRetireTimeoutSeconds action resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getSubmoduleIds`  
  **rel** = action  
  **title** = getSubmoduleIds
This resource's getSubmoduleIds action resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/isActiveVersion rel=action
title=isActiveVersion

This resource's isActiveVersion action resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/isAdminMode rel=action
title=isAdminMode

This resource's isAdminMode action resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime rel=canonical

This resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name} rel=parent

The parent resource of this resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime rel=self

This resource.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getCurrentState

The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST

POST

Aggregate state for the application. This is defined as the most advanced state of the application's modules on the named target.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

- **appid**
  - appid
  - Type: string

- **target**
  - logical target where the app is deployed
  - Type: string

Response Body

Contains a JSON object with a string return field.

POST

Aggregate state for the module. This is defined as the most advanced state of the module on all servers associated with the named target.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

- **appid**
  - appid
  - Type: string

- **moduleid**
  - moduleid
  - Type: string
**target**
logical target where module is deployed

**Type:** string

**Response Body**
Contains a JSON object with a string return field.

**POST**
Aggregate state for a submodule. This is defined as the most advanced state of the submodule on all servers associated with the named target.

**Roles**
Admin, Deployer, Monitor, Operator

**Request Body**
Must contain a JSON object with the following fields:

- **appid**
  appid
  **Type:** string

- **moduleid**
  moduleid
  **Type:** string

- **subModuleId**
  subModuleId
  **Type:** string

- **target**
  logical target where module is deployed
  **Type:** string

**Response Body**
Contains a JSON object with a string return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

The state the application should be in.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

appid

appid

Type: string

Response Body

Contains a JSON object with a string return field.

POST

The state the application should be in on a specific target.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

appid

appid

Type: string

target

target

Type: string

Response Body

Contains a JSON object with a string return field.
The resource supports the following methods:

- POST
POST

Names of the modules contained in the application. This does not include submodules.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

- **appid**
  - appid
  - Type: string

Response Body

Contains a JSON object with a array of string return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:
- POST
- POST

POST

Logical targets where the submodule is deployed.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

- `appid`
  - Type: string
- `moduleid`
  - Type: string
- `subModuleId`
  - Type: string

Response Body

Contains a JSON object with a array of string return field.

POST

Logical targets where the module is deployed.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

- `appid`
  - Type: string
- `moduleid`
  - Type: string
Response Body
Contains a JSON object with an array of string return field.
The resource supports the following methods:

- POST
POST

Indicates the type of module: EAR, WAR, etc.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

appid
appid
Type: string

moduleid
moduleid
Type: string

Response Body

Contains a JSON object with a string return field.
The resource supports the following methods:

- POST
For each configured ids provided, identify all inferred ids on each of managed server to which the configured ids are targeted.

Admin Server contacts all relevant servers identified by target on demand. Some of the managed servers might be slow in responding or may not respond at all. The use of timeout ensures an upper limit on the time Admin Server will wait for response from managed servers. Though not recommended, a value of 0 timeout may be used to indefinitely wait for response from all managed servers. Admin Server only contacts the managed server that are known to be running at this time.

This result also includes servers that did not respond. But it does not include servers that were known to be shutdown at the time of request and were never contacted.

Here is an example result:

```xml
<unresponsive><target>ms5</target><target>ms10</target></unresponsive>
<configured-id id="landscapedesign#V2.0">
  <inferred-id id="landscapedesign#V2.0.2">
    <state value="STATE_ACTIVE">
      <target>ms1</target>
      <target>ms2</target>
      <target>ms3</target>
      <target>ms4</target>
    </state>
  </inferred-id>
</configured-id>
<configured-id id="soilmanagement">
  <inferred-id id="soilmanagement#1">
    <state value="STATE_ACTIVE"/>
    <target>ms1</target>
    <target>ms2</target>
    <target>ms3</target>
    <target>ms4</target>
    <target>ms6</target>
    <target>ms7</target>
    <target>ms8</target>
    <target>ms9</target>
  </inferred-id>
</configured-id>
```

**Roles**

Admin, Deployer, Monitor, Operator

**Request Body**

Must contain a JSON object with the following fields:

- **configuredIds**
  A list of application ids or library’s deployment ids as configured in config.xml. This is the list for which state is being requested
  
  **Type**: array of string

- **timeout**
  amount of time in milliseconds before the command times out. Though not recommended, a value of 0 may be used to wait indefinitely
  
  **Type**: integer/int64

**Response Body**

Contains a JSON object with a string return field.
The resource supports the following methods:

- POST
POST

The time when the application will be retired.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

appid
appid

Type: string

Response Body

Contains a JSON object with a integer/int64 return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/getRetireTime-

The resource supports the following methods:

- POST
POST

The amount of time the application is given to retire.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

appid
appid

Type: string

Response Body

Contains a JSON object with a integer/int32 return field.
The resource supports the following methods:

- POST
POST

Submodules associated with this module.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

appid
Type: string

moduleid
Type: string

Response Body

Contains a JSON object with an array of string return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/appRuntimeStateRuntime/isActiveVersion

The resource supports the following methods:

- POST
POST

Indicates if the application is the active version, the one that new sessions will use.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

appid
is the application id

Type: string

Response Body

Contains a JSON object with a boolean return field.
The resource supports the following methods:

- POST
POST

Indicates if the application should only be available through the administration port. This is the desired state of the application.

Roles

Admin, Deployer, Monitor, Operator

Request Body

Must contain a JSON object with the following fields:

- **appid**
  - is the application id
  - Type: string

- **target**
  - target
  - Type: string

Response Body

Contains a JSON object with a boolean return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager

This resource manages a weblogic.management.runtime.DeploymentManagerMBean instance.

The resource supports the following methods:

- GET
- POST
GET

Get this `weblogic.management.runtime.DeploymentManagerMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.DeploymentManagerMBean` entity.

This method can return the following links:

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager/appendToExtensionLoader rel=action title=appendToExtensionLoader`
  This resource's appendToExtensionLoader action resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager/confirmApplicationName rel=action title=confirmApplicationName`
  This resource's confirmApplicationName action resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager/deploy rel=action title=deploy`
  This resource's deploy action resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager/distribute rel=action title=distribute`
  This resource's distribute action resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager/purgeCompletedDeploymentProgressObjects rel=action title=purgeCompletedDeploymentProgressObjects`
  This resource's purgeCompletedDeploymentProgressObjects action resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager/redeploy rel=action title=redeploy`
  This resource's redeploy action resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager/undeploy rel=action title=undeploy`
  This resource's undeploy action resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager/update rel=action title=update`
  This resource's update action resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager/appDeploymentRuntimes rel=appDeploymentRuntimes`
  This resource's appDeploymentRuntimes collection resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes[/name]/deploymentManager rel=canonical`
This resource.

- **uri**=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects
  
  This resource’s deploymentProgressObjects collection resource.

- **uri**=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes
  
  This resource’s libDeploymentRuntimes collection resource.

- **uri**=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name} rel=parent
  
  The parent resource of this resource.

- **uri**=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager rel=self
  
  This resource.
POST

Update this `weblogic.management.runtime.DeploymentManagerMBean` instance.

Roles

Admin, Deployer

Request Body

Must contain a `weblogic.management.runtime.DeploymentManagerMBean` entity.
This resource manages a collection of `weblogic.management.runtime.AppDeploymentRuntimeMBean` instances.

The resource supports the following methods:

- GET
GET

Get this collection of `weblogic.management.runtime.AppDeploymentRuntimeMBean` instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of `weblogic.management.runtime.AppDeploymentRuntimeMBean` entities.

This method can return the following links:

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes` *rel*=canonical
  
  This resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager` *rel*=parent
  
  The parent resource of this resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes` *rel*=self
  
  This resource.
This resource manages a `weblogic.management.runtime.AppDeploymentRuntimeMBean` instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.AppDeploymentRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.AppDeploymentRuntimeMBean` entity. This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}/getState`
  rel=action title=getState
  This resource’s getState action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}/redeploy`
  rel=action title=redeploy
  This resource’s redeploy action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}/start`
  rel=action title=start
  This resource’s start action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}/stop`
  rel=action title=stop
  This resource’s stop action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}/undeploy`
  rel=action title=undeploy
  This resource’s undeploy action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}/update`
  rel=action title=update
  This resource’s update action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}`
  rel=canonical
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes`
  rel=parent
  The parent resource of this resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}`
  rel=self
  This resource.
The resource supports the following methods:

- POST
POST

The state of the application for a target. Notifications will be generated for this attribute on state changes. Valid states are those supported by the weblogic.management.runtime.AppRuntimeStateRuntimeMBean. The notification types are appdeployment.created, appdeployment.deleted, appdeployment.state.new, appdeployment.state.prepared, appdeployment.state.admin, appdeployment.state.active, appdeployment.state.retired, appdeployment.state.failed, appdeployment.state.update.pending, and appdeployment.state.unknown. The userdata is a Map where the keys are target names and the values are the application state for that target.

Roles

Admin, Deployer, Operator

Request Body

Must contain a JSON object with the following fields:

- target
  the target for the application state
  Type: string

Response Body

Contains a JSON object with a string return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}/redeploy

The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST

POST

Redeploy the application in the background for the targets specified with the options specified. This is an asynchronous operation that returns immediately. The returned DeploymentProgressObjectMBean can be used to determine when the operation is completed.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer, Operator

Request Body

Must contain a JSON object with the following fields:

- **targets**
  The targets on which to redeploy the application. This would be server names, cluster names, or module names in a similar format to weblogic.Deployer (i.e. module1@server1). If null, the application will be redeployed on all configured targets.
  
  **Type:** array of string

- **applicationPath**
  A supported Java EE application or module archive, or a root directory for the application or module. May be null.

  **Type:** string

- **plan**
  The deployment plan for this application. May be null.

  **Type:** string

- **deploymentOptions**
  Allows for overriding the deployment options. If null, default options will be used. The values should all be of type The keys, units and default values for options are clusterDeploymentTimeout milliseconds 3600000, gracefulIgnoreSessions boolean false, gracefulProductionToAdmin boolean false, retireGracefully boolean true, retireTimeout seconds -1 (no timeout), adminMode boolean false, timeout milliseconds 0 (no timeout)

  **Type:** null
Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST

Redeploy the application in the background for the targets specified with the options specified. This is an asynchronous operation that returns immediately. The returned DeploymentProgressObjectMBean can be used to determine when the operation is completed.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer, Operator

Request Body

Must contain a JSON object with the following fields:

**targets**
The targets on which to redeploy the application. This would be server names, cluster names, or module names in a similar format to weblogic.Deployer (i.e. module1@server1). If null, the application will be redeployed on all configured targets.

Type: array of string

**plan**
The deployment plan for this application. May be null.

Type: string

**deploymentOptions**
Allows for overriding the deployment options. If null, default options will be used. The values should all be of type The keys, units and default values for options are clusterDeploymentTimeout milliseconds 3600000, gracefulIgnoreSessions boolean false, gracefulProductionToAdmin boolean false, retireGracefully boolean true, retireTimeout seconds -1 (no timeout), adminMode boolean false, timeout milliseconds 0 (no timeout)

Type: null

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST

Redeploy the application using the default options and configured targets. This is a synchronous operation that returns when the redeploy operation has completed. The default options are clusterDeploymentTimeout: 3600000 milliseconds, gracefulIgnoreSessions: false, gracefulProductionToAdmin: false, retireGracefully: true, retireTimeout: no timeout, adminMode: false, timeout: no timeout
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Deployer, Operator

**Response Body**

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appDeploymentRuntimes/{name}/start

The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

Start the application in the background for the targets specified with the options specified. This is an asynchronous operation that returns immediately. The returned DeploymentProgressObjectMBean can be used to determine when the operation is completed.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer, Operator

Request Body

Must contain a JSON object with the following fields:

- **targets**
  
The targets on which to start the application. This would be server names, cluster names, or module names in a similar format to weblogic.Deployer (i.e. module1@server1). If null, the application will be started on all configured targets.
  
  **Type:** array of string

- **deploymentOptions**

  Allows for overriding the deployment options. If null, default options will be used. The values should all be of type The keys, units and default values for options are clusterDeploymentTimeout milliseconds 3600000, gracefulIgnoreSessions boolean false, gracefulProductionToAdmin boolean false, retireGracefully boolean true, retireTimeout seconds -1 (no timeout), adminMode boolean false, timeout milliseconds 0 (no timeout)

  **Type:** null

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST

Start the application using the default options and configured targets. This is a synchronous operation that returns when the start operation has completed. The default options are clusterDeploymentTimeout: 3600000 milliseconds, gracefulIgnoreSessions: false, gracefulProductionToAdmin: false, retireGracefully: true, retireTimeout: no timeout, adminMode: false, timeout: no timeout
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Deployer, Operator

**Response Body**

Contains a JSON object with a `weblogic.management.runtime.DeploymentProgressObjectMBean` reference return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

Stop the application in the background for the targets specified with the options specified. This is an asynchronous operation that returns immediately. The returned DeploymentProgressObjectMBean can be used to determine when the operation is completed.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer, Operator

Request Body

Must contain a JSON object with the following fields:

targets
The targets on which to stop the application. This would be server names, cluster names, or module names in a similar format to weblogic.Deployer (i.e. module1@server1). If null, the application will be stopped on all configured targets.

Type: array of string

deploymentOptions
Allows for overriding the deployment options. If null, default options will be used. The values should all be of type The keys, units and default values for options are clusterDeploymentTimeout milliseconds 3600000, gracefulIgnoreSessions boolean false, gracefulProductionToAdmin boolean false, retireGracefully boolean true, retireTimeout seconds -1 (no timeout), adminMode boolean false, timeout milliseconds 0 (no timeout)

Type: null

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST

Stop the application using the default options and configured targets. This is a synchronous operation that returns when the stop operation has completed. The default options are clusterDeploymentTimeout: 3600000 milliseconds, gracefulIgnoreSessions: false, gracefulProductionToAdmin: false, retireGracefully: true, retireTimeout: no timeout, adminMode: false, timeout: no timeout
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Deployer, Operator

**Response Body**

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.
The resource supports the following methods:

- POST
The resource supports the following methods:

- POST
- POST

**POST**

Undeploy the application using the default options and configured targets. This is a synchronous operation that returns when the undeploy operation has completed. The default options are clusterDeploymentTimeout: 3600000 milliseconds, gracefulIgnoreSessions: false, gracefulProductionToAdmin: false, retireGracefully: true, retireTimeout: no timeout, adminMode: false, timeout: no timeout

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Deployer, Operator

**Response Body**

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

**POST**

Undeploy the application in the background for the targets specified with the options specified. This is an asynchronous operation that returns immediately. The returned DeploymentProgressObjectMBean can be used to determine when the operation is completed.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Deployer, Operator

**Request Body**

Must contain a JSON object with the following fields:

- **targets**
  The targets on which to undeploy the application. This would be server names, cluster names, or module names in a similar format to weblogic.Deployer (i.e. module1@server1). If null, the application will be undeployed on all configured targets.

  **Type**: array of string
**deploymentOptions**

 Allows for overriding the deployment options. If null, default options will be used. The values should all be of type. The keys, units and default values for options are:

- clusterDeploymentTimeout milliseconds 3600000
- gracefulIgnoreSessions boolean false
- gracefulProductionToAdmin boolean false
- retireGracefully boolean true
- retireTimeout seconds -1 (no timeout)
- adminMode boolean false
- timeout milliseconds 0 (no timeout)

**Type:** null

**Response Body**

Contains a JSON object with a
weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.
The resource supports the following methods:

- POST
POST

Allows for updating an application configuration using an changed deployment plan. A configuration update is equivalent to replacing the application's deployment plan. The deployment plan is redeployed in-place. A new version of the application is not started, even if the new deployment plan has a different version.

The update succeeds only if changes to the deployment plan do not require the application to be restarted. Configuration changes in the new deployment plan must be limited to tuning the application. Changes to resource bindings causes the update to fail. Use redeploy() to apply resource binding changes to a production application.

This method targets only root modules. Module level targeting is not supported.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer, Operator

Request Body

Must contain a JSON object with the following fields:

- **targets**
  represent the targets on which to update the application.
  Type: array of string

- **plan**
  The deployment plan containing the new deployment configuration information.
  Type: string

- **deploymentOptions**
  Allows for overriding the deployment options. May be null.
  Type: null

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/appendToExtensionLoader

The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

Appends a code source to the search path of the WebLogic Extension Loader, which is also referred to as the WebLogic Domain Loader. The code source file is distributed to the targets using the options specified. When the operation is complete, classes and resources in the jar file will be visible for class loading at the extension loader. Since the jar is now in place on targeted servers, the classes and resources will continue to be available even after a targeted server restarts. This is an asynchronous operation that returns immediately.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

**codeSourcePath**
A jar file to add to the extension loader

*Type: string*

**targets**
The targets on which to distribute the code source. This would be server names or cluster names. If null, the code source will be distributed on all configured targets.

*Type: array of string*

**deploymentOptions**
Allows for overriding the deployment options. If null, default options will be used. The values should all be of type string. The keys, units and default values for options are:

- clusterDeploymentTimeout milliseconds 3600000, retireTimeout seconds -1 (no timeout), timeout milliseconds 0 (no timeout)

*Type: null*

Response Body

Contains a JSON object with a

weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST

Appends a code source to the search path of the WebLogic Extension Loader, which is also referred to as the WebLogic Domain Loader. The code source file is distributed to all targets using default options. When the operation is complete, classes and resources in the jar file will be visible for class loading at the extension loader. Since the jar is now in place on targeted servers, the classes and resources will continue to be
available even after a targeted server restarts. This is an asynchronous operation that returns immediately.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

- **codeSourcePath**
  A jar file to add to the extension loader

  **Type:** string

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/confirmApplicationName

The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

Used by the deployment subsystem to confirm an application name.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

- **isRedeployment**
  - should be true in the case where this function is called for a redeployment.
  - Type: boolean

- **appSource**
  - is a the path to the application.
  - Type: string

- **optional**
  - alternate app descriptor.
  - Type: string

- **tentativeName**
  - The name proposed for the application (provided by the user, or through the manifest). Null if no name was specified by the user or through the manifest.
  - Type: string

- **tentativeApplicationId**
  - The tentative application ID. May only be null if the application is not versioned. If tentativeName is null and the application is versioned, this should be just the version string (which includes both the implementation and specification versions).
  - Type: string

- **deploymentOptions**
  - deployment options
  - Type: null

Response Body

Contains a JSON object with a string return field.

POST

Used by the deployment subsystem to confirm an application name.
Roles
Admin, Deployer

Request Body
Must contain a JSON object with the following fields:

**isRedeployment**
should be true in the case where this function is called for a redeployment.
Type: boolean

**appSource**
is a the path to the application.
Type: string

**optional**
alternate application descriptor path
Type: string

**tentativeName**
The name proposed for the application (provided by the user, or through the manifest). Null if no name was specified by the user or through the manifest.
Type: string

**tentativeApplicationId**
The tentative application ID. May only be null if the application is not versioned. If tentativeName is null and the application is versioned, this should be just the version string (which includes both the implementation and specification versions).
Type: string

Response Body
Contains a JSON object with a string return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

A convenience method for distributing and starting an application.

The application source files and deployment plan are distributed to the deployment targets using the default options. This is a synchronous operation that returns when the deploy operation has completed. If distribution is successful, the application is then started on each of the targets. Successful deployment requires that all targets receive both the archive and deployment plan.

The default options are

- clusterDeploymentTimeout: 3600000 milliseconds
- gracefulIgnoreSessions: false
- gracefulProductionToAdmin: false
- retireGracefully: true
- retireTimeout: no timeout
- adminMode: false
- timeout: no timeout

If any modules are currently running, this method immediately restarts those modules using the newer files. Restarting modules in this manner is not the same as redeployment, because session state is not preserved for the active modules.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deploier

Request Body

Must contain a JSON object with the following fields:

- **name**
  The name of the application
  Type: string

- **applicationPath**
  A supported Java EE application or module archive, or a root directory for the application or module.
  Type: string

- **plan**
  The deployment plan for this application. May be null.
POST

Type: string

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST

A convenience method for distributing and starting an application. The application source files and deployment plan are distributed to the deployment targets using the options specified. This is an asynchronous operation that returns immediately. If distribution is successful, the application is then started on each of the targets. Successful deployment requires that all targets receive both the archive and deployment plan.

If any modules are currently running, this method immediately restarts those modules using the newer files. Restarting modules in this manner is not the same as redeployment, because session state is not preserved for the active modules.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

name
The name of the application
Type: string

applicationPath
A supported Java EE application or module archive, or a root directory for the application or module.
Type: string

targets
The targets on which to deploy the application. This would be server names, cluster names, or module names in a similar format to weblogic.Deployer (i.e. module1@server1). If null, the application will be redeployed on all configured targets.
Type: array of string

plan
The deployment plan for this application. May be null.
Type: string

deploymentOptions
Allows for overriding the deployment options. If null, default options will be used. The values should all be of type The keys, units and default values for options are clusterDeploymentTimeout milliseconds 3600000, gracefulIgnoreSessions boolean
false, gracefulProductionToAdmin boolean false, retireGracefully boolean true, retireTimeout seconds -1 (no timeout), adminMode boolean false, timeout milliseconds 0 (no timeout)

Type: null

**Response Body**

Contains a JSON object with a `weblogic.management.runtime.DeploymentProgressObjectMBean` reference return field.
This resource manages a collection of `weblogic.management.runtime.DeploymentProgressObjectMBean` instances.

The resource supports the following methods:

- **GET**
GET

Get this collection of weblogic.management.runtime.DeploymentProgressObjectMBean instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of weblogic.management.runtime.DeploymentProgressObjectMBean entities.

This method can return the following links:

- **uri**=management/weblogic/\{version\}/domainRuntime/domainPartitionRuntimes/\{name\}/deploymentManager/deploymentProgressObjects **rel**=canonical
  
  This resource.

- **uri**=management/weblogic/\{version\}/domainRuntime/domainPartitionRuntimes/\{name\}/deploymentManager **rel**=parent
  
  The parent resource of this resource.

- **uri**=management/weblogic/\{version\}/domainRuntime/domainPartitionRuntimes/\{name\}/deploymentManager/deploymentProgressObjects **rel**=self
  
  This resource.
This resource manages a
weblogic.management.runtime.DeploymentProgressObjectMBean instance.

The resource supports the following methods:

- GET
GET

Get this weblogic.management.runtime.DeploymentProgressObjectMBean instance.

Roles

Admin, Deplyeer, Monitor, Operator

Response Body

Contains a weblogic.management.runtime.DeploymentProgressObjectMBean entity.

This method can return the following links:

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects/{name}/cancel
  rel=action title=cancel
  This resource's cancel action resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects/{name}/getExceptions
  rel=action title=getExceptions
  This resource's getExceptions action resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects/{name}
  rel=canonical
  This resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects
  rel=parent
  The parent resource of this resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects/{name}
  rel=self
  This resource.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/deploymentProgressObjects/{name}/cancel

The resource supports the following methods:

- POST
POST

Attempt to cancel the deployment operation. Any actions which have yet to start will be inhibited. Any completed actions will remain in place.

Roles

Admin, Deployer

Response Body

Contains an empty JSON object.
The resource supports the following methods:

- POST
POST

If the current deployment operation has failed, this method may return zero or more exception(s) which represent the errors for this target. The array will not contain WLS exception classes; instead they will be new RuntimeExceptions containing the stack traces and messages from the original WLS Exceptions.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

target
the target where exceptions might have occurred.

Type: string

Response Body

Contains a JSON object with a array of null return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

Distribute an application.

The application source files and deployment plan are distributed to the deployment targets using the default options. This is a synchronous operation that returns when the distribute operation has completed.

The default options are:

- clusterDeploymentTimeout: 3600000 milliseconds
- gracefulIgnoreSessions: false
- gracefulProductionToAdmin: false
- retireGracefully: true
- retireTimeout: no timeout
- adminMode: false,
- timeout: no timeout

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

- **name**
  The name of the application
  Type: string

- **applicationPath**
  A supported Java EE application or module archive, or a root directory for the application or module.
  Type: string

- **plan**
  The deployment plan for this application. May be null.
  Type: string
**Response Body**

Contains a JSON object with a 
weblogic.management.runtime.DeploymentProgressObjectMBean reference return 
field.

**POST**

Distribute an application. The application source files and deployment plan are 
distributed to the deployment targets using the options specified. This is an 
asynchronous operation that returns immediately.

This method supports both asynchronous and best effort synchronous invocations. 
The behavior is controlled by the 'Prefer' header. See About Synchronous and 
Asynchronous Operations.

**Roles**

Admin, Deployer

**Request Body**

Must contain a JSON object with the following fields:

- **name**
The name of the application
  Type: string

- **applicationPath**
A supported Java EE application or module archive, or a root directory for the 
application or module.
  Type: string

- **targets**
The targets on which to deploy the application. This would be server names, cluster 
names, or module names in a similar format to weblogic.Deployer (i.e. 
module1@server1). If null, the application will be distributed on all configured targets.
  Type: array of string

- **plan**
The deployment plan for this application. May be null.
  Type: string

- **deploymentOptions**
Allows for overriding the deployment options. If null, default options will be used.
The values should all be of type The keys, units and default values for options are 
clusterDeploymentTimeout milliseconds 3600000, gracefulIgnoreSessions boolean 
false, gracefulProductionToAdmin boolean false, retireGracefully boolean true, 
retireTimeout seconds -1 (no timeout), adminMode boolean false, timeout milliseconds 
0 (no timeout)
  Type: null
Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.
This resource manages a collection of weblogic.management.runtime.LibDeploymentRuntimeMBean instances.

The resource supports the following methods:

- GET
GET

Get this collection of `weblogic.management.runtime.LibDeploymentRuntimeMBean` instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of `weblogic.management.runtime.LibDeploymentRuntimeMBean` entities.

This method can return the following links:

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes` **rel**=canonical
  
  This resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager` **rel**=parent
  
  The parent resource of this resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes` **rel**=self
  
  This resource.
This resource manages a `weblogic.management.runtime.LibDeploymentRuntimeMBean` instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.LibDeploymentRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.LibDeploymentRuntimeMBean` entity.

This method can return the following links:

- **uri**: `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes/{name}/redeploy`
  - rel: `action` title: redeploy
    - This resource’s redeploy action resource.

- **uri**: `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes/{name}/undeploy`
  - rel: `action` title: undeploy
    - This resource’s undeploy action resource.

- **uri**: `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes/{name}`
  - rel: `canonical`
    - This resource.

- **uri**: `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes/{name}`
  - rel: `parent`
    - The parent resource of this resource.

- **uri**: `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/libDeploymentRuntimes/{name}`
  - rel: `self`
    - This resource.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

Redeploy the library in the background for the targets specified with the options specified. This is an synchronous operation that returns immediately. The returned DeploymentProgressObjectMBean can be used to determine when the operation is completed.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

- **targets**
  
  The targets on which to redeploy the library. This would be server names, cluster names, or module names in a similar format to weblogicDeployer (i.e. module1@server1). If null, the library will be redeployed on all configured targets.
  
  Type: array of string

- **deploymentOptions**
  
  Allows for overriding the deployment options. If null, default options will be used. The keys, units and default values for options are adminMode boolean false, timeout milliseconds 0 (no timeout)
  
  Type: null

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST

Redeploy the library using the default options and configured targets. This is a synchronous operation that returns when the redeploy operation has completed. The default options are adminMode: false, timeout: no timeout

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer
Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

Undeploy the library using the default options and configured targets. This is a synchronous operation that returns when the redeploy operation has completed. The default options are adminMode boolean false timeout: no timeout

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST

Undeploy the library in the background for the targets specified with the options specified. This is an asynchronous operation that returns immediately. The returned DeploymentProgressObjectMBean can be used to determine when the operation is completed.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

- **targets**
  The targets on which to undeploy the library. This would be server names, cluster names, or module names in a similar format to weblogic.Deployer (i.e. module1@server1). If null, the library will be undeployed on all configured targets.
  
  **Type**: array of string

- **deploymentOptions**
  Allows for overriding the deployment options. If null, default options will be used. The values should all be of type The keys, units and default values for options are adminMode boolean false timeout milliseconds 0 (no timeout)
  
  **Type**: null
Response Body

Contains a JSON object with a
weblogic.management.runtime.DeploymentProgressObjectMBean reference return
field.
The resource supports the following methods:

- POST
POST

Removes progress objects for completed operations.

Roles

Admin, Deployer

Response Body

Contains an empty JSON object.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/deploymentManager/redeploy

The resource supports the following methods:
  
  - POST
POST

The resource supports the following methods:

- POST
- POST

POST

Redeploy the application that is deployed to a resource group template in the background for the targets specified with the options specified. This is an asynchronous operation that returns immediately. The returned DeploymentProgressObjectMBean can be used to determine when the operation is completed.

Note that this API is only for redeploying applications or libraries that are deployed to a resource group template, and a resourceGroupTemplate property value must be provided in deploymentOptions. AppDeploymentRuntimeMBean should be used for redeploying applications that are not deployed to a resource group template, and LibDeploymentRuntimeMBean should be used for redeploying libraries that are not deployed to a resource group template.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

name
The application name
Type: string

applicationPath
A supported Java EE application or module archive, or a root directory for the application or module. May be null.
Type: string

plan
The deployment plan for this application. May be null.
Type: string

deploymentOptions
Allows for overriding the deployment options. The keys, units and default values for options are clusterDeploymentTimeout milliseconds 3600000, gracefulIgnoreSessions boolean false, gracefulProductionToAdmin boolean false, retireGracefully boolean true, retireTimeout seconds -1 (no timeout), adminMode boolean false, timeout milliseconds 0 (no timeout) resourceGroupTemplate this value is required
Type: null
POST

Response Body
Contains a JSON object with a
weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST
Redeploy the application that is deployed to a resource group template in the
background for the targets specified with the options specified. This is an
asynchronous operation that returns immediately. The returned
DeploymentProgressObjectMBean can be used to determine when the operation is
completed.

Note that this API is only for redeploying applications or libraries that are deployed to
a resource group template, and a resourceGroupTemplate property value must be
provided in deploymentOptions. AppDeploymentRuntimeMBean should be used for
redeploying applications that are not deployed to a resource group template, and
LibDeploymentRuntimeMBean should be used for redeploying libraries that are not
deployed to a resource group template.

This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and
Asynchronous Operations.

Roles
Admin, Deployer

Request Body
Must contain a JSON object with the following fields:

name
The application name
Type: string

plan
The deployment plan for this application. May be null.
Type: string

deploymentOptions
Allows for overriding the deployment options. The keys, units and default values for
options are clusterDeploymentTimeout milliseconds 3600000, gracefulIgnoreSessions
boolean false, gracefulProductionToAdmin boolean false, retireGracefully boolean
true, retireTimeout seconds -1 (no timeout), adminMode boolean false, timeout
milliseconds 0 (no timeout) resourceGroupTemplate this value is required
Type: null

Response Body
Contains a JSON object with a
weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

Undeploy the application that is deployed to a resource group template using the default options. This is a synchronous operation that returns when the undeploy operation has completed. The default options are clusterDeploymentTimeout: 3600000 milliseconds, gracefulIgnoreSessions: false, gracefulProductionToAdmin: false, retireGracefully: true, retireTimeout: no timeout, adminMode: false, timeout: no timeout

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

- name
  The application name
  Type: string

- template
  The resource group template
  Type: string

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.

POST

Undeploy the application that is deployed to a resource group template in the background with the options specified. This is an asynchronous operation that returns immediately. The returned DeploymentProgressObjectMBean can be used to determine when the operation is completed.

Note that this API is only for undeploying applications that are deployed to a resource group template, and a resourceGroupTemplate property value must be provided in deploymentOptions. AppDeploymentRuntimeMBean should be used for undeploying applications that are not deployed to a resource group template.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.
Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

**name**
The application name
Type: string

**deploymentOptions**
Allows for overriding the deployment options. The keys, units and default values for options are:
- `clusterDeploymentTimeout` milliseconds: 3600000
- `gracefulIgnoreSessions` boolean: false
- `gracefulProductionToAdmin` boolean: false
- `retireGracefully` boolean: true
- `retireTimeout` seconds: -1 (no timeout)
- `adminMode` boolean: false
- `timeout` milliseconds: 0 (no timeout)
- `resourceGroupTemplate` this value is required

Type: null

Response Body

Contains a JSON object with a reference return field.
The resource supports the following methods:

- POST
POST

Allows for updating an application configuration using an changed deployment plan for applications deployed to a resource group template.

A configuration update is equivalent to replacing the application's deployment plan. The deployment plan is redeployed in-place. A new version of the application is not started, even if the new deployment plan has a different version.

The update succeeds only if changes to the deployment plan do not require the application to be restarted. Configuration changes in the new deployment plan must be limited to tuning the application. Changes to resource bindings causes the update to fail. Use redeploy(String, String, Properties) to apply resource binding changes to a production application.

This method targets only root modules. Module level targeting is not supported.

Note that this API is only for updating applications that are deployed to a resource group template, and a resourceGroupTemplate property value must be provided in deploymentOptions. AppDeploymentRuntimeMBean should be used for updating applications that are not deployed to a resource group template.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

name
The application name
Type: string

plan
The deployment plan for this application. May be null.
Type: string

deploymentOptions
Allows for overriding the deployment options. The keys, units and default values for options are clusterDeploymentTimeout milliseconds 3600000, gracefulIgnoreSessions boolean false, gracefulProductionToAdmin boolean false, retireGracefully boolean true, retireTimeout seconds -1 (no timeout), adminMode boolean false, timeout milliseconds 0 (no timeout) resourceGroupTemplate this value is required
Type: null

Response Body

Contains a JSON object with a weblogic.management.runtime.DeploymentProgressObjectMBean reference return field.
This resource manages a
weblogic.management.runtime/EditSessionConfigurationManagerMBean instance.
The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime/EditSessionConfigurationManagerMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime/EditSessionConfigurationManagerMBean` entity.

This method can return the following links:

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/forceDestroyEditSessionConfiguration`  
  **rel**=action  
  **title**=forceDestroyEditSessionConfiguration

  This resource's forceDestroyEditSessionConfiguration action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager`  
  **rel**=canonical

  This resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurationCreateForm`  
  **rel**=editSessionConfigurationCreateForm

  The create form resource for this resource's editSessionConfigurations collection resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations`  
  **rel**=editSessionConfigurations

  This resource's editSessionConfigurations collection resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager`  
  **rel**=parent

  The parent resource of this resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager`  
  **rel**=self

  This resource.
This resource manages the create forms for the editSessionConfigurations collection resource.

The resource supports the following methods:

- GET
GET

This resource returns a `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` entity populated with default values that can be customized then posted (using the POST method) to the editSessionConfigurations collection resource to create a new instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` entity.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurationCreateForm` rel=canonical
  
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations` rel=create
  
  The collection resource for this create form resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager` rel=parent
  
  The parent resource of this resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurationCreateForm` rel=self
  
  This resource.
This resource manages a collection of `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` instances.

The resource supports the following methods:

- GET
- POST
GET

Get this collection of `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` entities.

This method can return the following links:

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations`
  
  rel = canonical
  
  This resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurationCreateForm`
  
  rel = create-form
  
  The create form resource for this collection resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager`
  
  rel = parent
  
  The parent resource of this resource.

- **uri** = `/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations`
  
  rel = self
  
  This resource.
POST

Add a new `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` instance to this collection.

Roles

Admin, Deployer

Request Body

Must contain a `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` entity.
This resource manages a
weblogic.management.runtime.EditSessionConfigurationRuntimeMBean instance.

The resource supports the following methods:

- DELETE
- GET
DELETE

Delete this `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` instance.

Roles

Admin, Deployer
GET

Get this `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.EditSessionConfigurationRuntimeMBean` entity.

This method can return the following links:

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations/{name}/containsUnactivatedChanges` **rel**=action **title**=containsUnactivatedChanges
  This resource's containsUnactivatedChanges action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations/{name}` **rel**=canonical
  This resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations` **rel**=parent
  The parent resource of this resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations/{name}` **rel**=self
  This resource.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/editSessionConfigurationManager/editSessionConfigurations/{name}/containsUnactivatedChanges

The resource supports the following methods:

- POST
POST

Returns true if there are changes saved in the pending directory or there are changes in memory. To apply these changes, activate the session.

Roles

Admin, Deployer

Response Body

Contains a JSON object with a boolean return field.
The resource supports the following methods:

- POST
POST

Destroys an existing named edit session configuration. This will destroy the associated edit session specific MBeanServer and remove the <domain>/edit/<name> or <domain>/partitions/<name>/edit/<name> directory and subdirectories.

Unlike destroyEditSessionConfiguration(EditSessionConfigurationRuntimeMBean), this operation does not end with error in case that session contains unactivated changes, and allows to destroy even edit sessions not owned by the caller (requires admin privileges).

Roles

Admin, Deployer

Request Body

Must contain a JSON object with the following fields:

**editSession**
Edit session to destroy.

*Type: weblogic.management.runtime.EditSessionConfigurationRuntimeMBean reference*

Response Body

Contains an empty JSON object.
This resource manages a `weblogic.management.runtime.PartitionLifeCycleRuntimeMBean` instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.PartitionLifeCycleRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.PartitionLifeCycleRuntimeMBean` entity.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/boot rel=action title=boot`
  This resource's boot action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/forceShutdown rel=action title=forceShutdown`
  This resource's forceShutdown action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/forceSuspend rel=action title=forceSuspend`
  This resource's forceSuspend action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/getState rel=action title=getState`
  This resource's getState action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/getSubState rel=action title=getSubState`
  This resource's getSubState action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/halt rel=action title=halt`
  This resource's halt action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/purgeTasks rel=action title=purgeTasks`
  This resource's purgeTasks action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resume rel=action title=resume`
  This resource's resume action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/shutdown rel=action title=shutdown`
  This resource's shutdown action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/start rel=action title=start`
  This resource's start action resource.
GET

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime/startInAdmin rel=action title=startInAdmin
  This resource's startInAdmin action resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime/suspend rel=action title=suspend
  This resource's suspend action resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime rel=canonical
  This resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name] rel=parent
  The parent resource of this resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes
  rel=resourceGroupLifeCycleRuntimes
  This resource's resourceGroupLifeCycleRuntimes collection resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime rel=self
  This resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime/tasks rel=tasks
  This resource's tasks collection resource.
POST

The resource supports the following methods:

- POST
- POST
- POST

POST

Begins the transition of the partition from the \texttt{SHUTDOWN} state and \texttt{HALTED} substate to the \texttt{SHUTDOWN} state and \texttt{BOOTED} substate on the specified targets.

This operation has the effect of starting the partition's administrative resource groups on those targets, changing their states on those targets from \texttt{SHUTDOWN} to \texttt{RUNNING}.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion.

Please use \texttt{PartitionLifeCycleRuntimeMBean.boot(String[] targetNames)}

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

\textbf{targets}
the specific targets on which to boot this partition

\textbf{Type}: array of \texttt{weblogic.management.configuration.TargetMBean} references

Response Body

Contains a JSON object with a \texttt{weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean} reference return field.

POST

Begins the transition of the partition from the \texttt{SHUTDOWN} state and \texttt{HALTED} substate to the \texttt{SHUTDOWN} state and \texttt{BOOTED} substate on the specified targets.

This operation has the effect of starting the partition's administrative resource groups on those targets, changing their states on those targets from \texttt{SHUTDOWN} to \texttt{RUNNING}.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator
Request Body

Must contain a JSON object with the following fields:

**targetNames**
the specific servers on which to boot this partition

**Type:** array of string

Response Body

Contains a JSON object with a weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean reference return field.

POST

Begins the transition of the partition from the **SHUTDOWN** state and **HALTED** substate to the **SHUTDOWN** state and **BOOTED** substate on all targeted servers.

This operation has the effect of starting the partition's administrative resource groups, changing their states from **SHUTDOWN** to **RUNNING**

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean reference return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/forceShutdown

The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST
- POST

POST

Begins to force the shutdown of this partition with the specified timeout on all targeted servers.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

```
timeout
```

Type: integer/int32

Response Body

Contains a JSON object with a

POST

Begins to force the shutdown of this partition on all targeted servers.

On each targeted server the partition is shut down and then restored to its current state on that server.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator
**Response Body**

Contains a JSON object with a

**POST**

Begins to force the shut down of this partition on the specified targets.
On each target the partition is shut down and then restored to its current state on that server.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

- **targetNames**
  the specific servers on which to shut down this partition

**Type**: array of string

**Response Body**

Contains a JSON object with a

**POST**

Begins to force the shut down of this partition on the specified targets.
On each target the partition is shut down and then restored to its current state on that server.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
Please use PartitionLifeCycleRuntimeMBean.forceShutdown(String[] targetNames)
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:
targets
the specific targets on which to shut down this partition
Type: array of weblogic.management.configuration.TargetMBean references

Response Body
Contains a JSON object with a
The resource supports the following methods:

- POST
The resource supports the following methods:

- POST
- POST
- POST

**POST**

Begins the forceful transition of the partition from the **RUNNING** state to the **ADMIN** state on all targeted servers.

Use the returned `PartitionLifeCycleTaskRuntimeMBean` value to check for completion.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Operator

**Response Body**

Contains a JSON object with a `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` reference return field.

**POST**

Begins the forceful transition of the partition from the **RUNNING** state to the **ADMIN** state on the specified targets.

Use the returned `PartitionLifeCycleTaskRuntimeMBean` value to check for completion.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

- **targetNames**
  
  the specific servers on which to suspend this partition

  **Type:** array of string

**Response Body**

Contains a JSON object with a `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` reference return field.
POST

Begins the forceful transition of the partition from the **RUNNING** state to the **ADMIN** state on the specified targets.

Use the returned `PartitionLifeCycleTaskRuntimeMBean` value to check for completion.

Please use `PartitionLifeCycleRuntimeMBean.forceSuspend(String[] targetNames)`

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

- **targets**
  the specific targets on which to suspend this partition

  Type: array of `weblogic.management.configuration.TargetMBean` references

**Response Body**

Contains a JSON object with a `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` reference return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/getState

The resource supports the following methods:

- POST
The resource supports the following methods:

- POST
- POST

POST

The state of the partition on a specific server.

This method has been deprecated in favor of retrieving partition state correctly on dynamic servers. ServerMBean does not exist for dynamic servers in the domain runtime tree. So, although invoking this method from domain runtime tree works for static servers, does not work for dynamic servers.

Please use `PartitionLifeCycleRuntimeMBean.getState(String serverName)`

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- `serverMBean`
  
  serverMBean

  Type: `weblogic.management.configuration.ServerMBean` reference

Response Body

Contains a JSON object with a string return field.

POST

The current state of the partition on a specific server.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- `serverName`
  
  name of the server on which the substate to check

  Type: `string`

Response Body

Contains a JSON object with a string return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST

POST

The current state of the partition on a specific server.

This method has been deprecated in favor of retrieving partition state correctly on dynamic servers. ServerMBean does not exist for dynamic servers in the domain runtime tree. So, although invoking this method from domain runtime tree works for static servers, does not work for dynamic servers.

Please use
PartitionLifeCycleRuntimeMBean.getSubState(ServerLifeCycleRuntimeMBean serverLifeCycleRuntimeMBean)

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

serverMBean

the server on which the state to check

Type: weblogic.management.configuration.ServerMBean reference

Response Body

Contains a JSON object with a string return field.

POST

The current state of the partition on a specific server.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

serverName

name of the server on which the state to check

Type: string

Response Body

Contains a JSON object with a string return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/halt

The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST

POST

Begins the forceful transition of the partition from the **RUNNING** state or the **ADMIN** state to the **SHUTDOWN** state and **HALTED** substate on a specific set of targets.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion.

This method supports both asynchronous and best effort synchronous invocations.

The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- **targetNames**
  the specific servers on which to halt this partition
  
  Type: array of string

Response Body

Contains a JSON object with a weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean reference return field.

POST

Begins the forceful transition of the partition from the **RUNNING** state or the **ADMIN** state to the **SHUTDOWN** state and **HALTED** substate on all targeted servers.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion.

This method supports both asynchronous and best effort synchronous invocations.

The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean reference return field.
POST

Begins the forceful transition of the partition from the \texttt{RUNNING} state or the \texttt{ADMIN} state to the \texttt{SHUTDOWN} state and \texttt{HALTED} substate on a specific set of targets.

Use the returned \texttt{PartitionLifeCycleTaskRuntimeMBean} value to check for completion

Please use \texttt{PartitionLifeCycleRuntimeMBean.halt(String[] targetNames)}

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- \texttt{targets}
  the specific targets on which to halt this partition

  Type: array of \texttt{weblogic.management.configuration.TargetMBean} references

Response Body

Contains a JSON object with a \texttt{weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean} reference return field.
The resource supports the following methods:

- POST
POST

Clears the retained task collection.

Roles

Admin, Operator

Response Body

Contains an empty JSON object.
This resource manages a collection of `weblogic.management.runtime.ResourceGroupLifeCycleRuntimeMBean` instances.

The resource supports the following methods:

- `GET`
GET

Get this collection of

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of

This method can return the following links:

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes
  /{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes
  rel=canonical
  This resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes
  /{name}/partitionLifeCycleRuntime rel=parent
  The parent resource of this resource.

- uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes
  /{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes rel=self
  This resource.
This resource manages a weblogic.management.runtime.ResourceGroupLifeCycleRuntimeMBean instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.ResourceGroupLifeCycleRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body


This method can return the following links:

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/forceShutdown rel=action title=forceShutdown`
  
  This resource's forceShutdown action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/forceSuspend rel=action title=forceSuspend`
  
  This resource's forceSuspend action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/getState rel=action title=getState`
  
  This resource's getState action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/purgeTasks rel=action title=purgeTasks`
  
  This resource's purgeTasks action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/resume rel=action title=resume`
  
  This resource's resume action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/shutdown rel=action title=shutdown`
  
  This resource's shutdown action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/start rel=action title=start`
  
  This resource's start action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/startInAdmin rel=action title=startInAdmin`
  
  This resource's startInAdmin action resource.
- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/suspend rel=action title=suspend
  This resource's suspend action resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]
  rel=canonical
  This resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes
  rel=parent
  The parent resource of this resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]
  rel=self
  This resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks rel=tasks
  This resource's tasks collection resource.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

■ POST
■ POST
■ POST

POST

Begins to force the shutdown of this resource group on all targeted servers, changing its state to SHUTDOWN

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

POST

Begins to force the shutdown of this resource group on the specified targets, changing its state to SHUTDOWN

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**targetNames**
the specific servers on which to force a shutdown on this resource group

**Type:** array of string
Response Body

Contains a JSON object with a

POST

Begins to force the shutdown of this resource group on the specified targets, changing its state to \textit{SHUTDOWN}

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

\textbf{Please use} ResourceGroupLifeCycleTaskRuntimeMBean.forceShutdown(String[] targetNames)

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

\textbf{targets}
the specific targets on which to force a shutdown on this resource group

\textbf{Type}: array of weblogic.management.configuration.TargetMBean references

Response Body

Contains a JSON object with a
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST

POST

Begins the forceful transition of this resource group from the **RUNNING** state to the **ADMIN** state on all targeted servers.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

POST

Begins the forceful transition of this resource group from the **RUNNING** state to the **ADMIN** state on the specified targets.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**targetNames**
the specific servers on which to force a suspend of this resource group

**Type:** array of string
**Response Body**
Contains a JSON object with a

**POST**

Begins the forceful transition of this resource group from the **RUNNING** state to the **ADMIN** state on the specified targets.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion.

Please use `ResourceGroupLifeCycleTaskRuntimeMBean.forceSuspend(String[] targetNames)`

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**
Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

*targets*
the specific targets on which to start this resource group

**Type**: array of weblogic.management.configuration.TargetMBean references

**Response Body**

Contains a JSON object with a
The resource supports the following methods:

- POST
**POST**

The resource supports the following methods:

- **POST**
- **POST**

**POST**

The state of resource group on a specific server.

Please use `ResourceGroupLifeCycleTaskRuntimeMBean.getState(String serverName)`

**Roles**

Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

- `serverMBean`
  - `serverMBean`
  - **Type**: `weblogic.management.configuration.ServerMBean reference`

**Response Body**

Contains a JSON object with a string return field.

**POST**

The current state of the resource group on a specific server.

**Roles**

Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

- `serviceName`
  - `name of the server to check`
  - **Type**: `string`

**Response Body**

Contains a JSON object with a string return field.
The resource supports the following methods:

- POST
POST

Clears the retained task collection.

Roles

Admin, Operator

Response Body

Contains an empty JSON object.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST

POST

Begins the transition of this resource group from the ADMIN state to the RUNNING state on all targeted servers.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

POST

Begins the transition of this resource group from the ADMIN state to the RUNNING state on the specified targets.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion.

Please use ResourceGroupLifeCycleTaskRuntimeMBean.resume(String[] targetNames)

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- **targets**
  the specific targets on which to start this resource group

  Type: array of weblogic.management.configuration.TargetMBean references
Response Body
Contains a JSON object with a
weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference
return field.

POST
Begins the transition of this resource group from the ADMIN state to the RUNNING state
on the specified targets.
Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for
completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and
Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

targetNames
the specific servers on which to resume this resource group
Type: array of string

Response Body
Contains a JSON object with a
weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference
return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST
- POST
- POST
- POST
- POST
- POST
- POST

POST

Begins to shut down this resource group gracefully on the specified targets, changing its state to `SHUTDOWN`. (equivalent to shutdown(targets, 0, false)).

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

Please use `ResourceGroupLifeCycleTaskRuntimeMBean.shutdown(String[] targetNames)`

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- targets
  the specific targets on which to start this resource group

Type: array of `weblogic.management.configuration.TargetMBean` references

Response Body

Contains a JSON object with a `weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean` reference return field.

POST

Begins to shut down this resource group gracefully on the specified targets, changing its state to `SHUTDOWN`. (equivalent to shutdown(timeout, ignoreSessions, false, targets)).

Use the returned `ResourceGroupLifeCycleTaskRuntimeMBean` value to check for completion
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**timeout**
Number of seconds to wait before aborting inflight work and shutting down the server.

*Type: integer/int32*

**ignoreSessions**
true indicates ignore pending HTTP sessions during inflight work handling.

*Type: boolean*

**targetNames**
the specific servers on which to shut down this resource group

*Type: array of string*

Response Body

Contains a JSON object with a

POST

Begins to shut down this resource group gracefully on the specified targets, changing its state to SHUTDOWN.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

Please use ResourceGroupLifeCycleTaskRuntimeMBean.shutdown(int timeout, boolean ignoreSessions, boolean waitForAllSessions, TargetMBean[] targets)

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**timeout**
Number of seconds to wait before aborting inflight work and force shutting down the server.
Type: integer/int32

ignoreSessions
Set to true to ignore pending HTTP sessions during inflight work handling.
Type: boolean

waitForAllSessions
Set to true to wait for all HTTP sessions during inflight work handling; false to wait for non-persisted HTTP sessions only.
Type: boolean

targets
the specific targets on which to shut down this resource group
Type: array of weblogic.management.configuration.TargetMBean references

Response Body
Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

POST
Begins to shut down this resource group gracefully on all targeted servers, changing its state to SHUTDOWN.
Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

timeout
Number of seconds to wait before aborting inflight work and force shutting down the server.
Type: integer/int32

ignoreSessions
Set to true to ignore pending HTTP sessions during inflight work handling.
Type: boolean

waitForAllSessions
Set to true to wait for all HTTP sessions during inflight work handling; false to wait for non-persisted HTTP sessions only.
Type: boolean
POST

Response Body
Contains a JSON object with a
weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference
return field.

POST
Begins to shut down this resource group gracefully on the specified targets, changing
its state to SHUTDOWN. (equivalent to shutdown(timeout, ignoreSessions, false, targets)).
Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for
completion

Please use ResourceGroupLifeCycleTaskRuntimeMBean.shutdown(int timeout,
boolean ignoreSessions, String[] targetNames)

This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and
Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

timeout
Number of seconds to wait before aborting inflight work and shutting down the
server.
Type: integer/int32

ignoreSessions
ture indicates ignore pending HTTP sessions during inflight work handling.
Type: boolean

targets
the specific targets on which to shut down this resource group
Type: array of weblogic.management.configuration.TargetMBean references

Response Body
Contains a JSON object with a
weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference
return field.

POST
Begins to shut down this resource group gracefully on the specified targets, changing
its state to SHUTDOWN. (equivalent to shutdown(targets, 0, false)).
Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for
completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and
Asynchronous Operations.
Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**targetNames**
the specific servers on which to shut down this resource group
Type: array of string

Response Body

Contains a JSON object with a

POST

Begins to shut down the resource group gracefully on all targeted servers, changing its state to SHUTDOWN.
Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion.
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:
**timeout**
Number of seconds to wait before aborting inflight work and force shutting down the server.
Type: integer/int32

**ignoreSessions**
Set to true to ignore pending HTTP sessions during inflight work handling.
Type: boolean

**waitForAllSessions**
Set to true to wait for all HTTP sessions during inflight work handling; false to wait for non-persisted HTTP sessions only.
Type: boolean

**targetNames**
the specific servers on which to shut down this resource group
Type: array of string

**Response Body**
Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

**POST**
Begins to shut down this resource group gracefully on all targeted servers, changing its state to SHUTDOWN. (equivalent to shutdown(timeout, ignoreSessions, false)).
Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**
Admin, Operator

**Request Body**
Must contain a JSON object with the following fields:

**timeout**
Number of seconds to wait before aborting inflight work and shutting down the server.
Type: integer/int32

**ignoreSessions**
true indicates ignore pending HTTP sessions during inflight work handling.
Type: boolean
Response Body

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/start

The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST

POST

Begins to start this resource group on all targeted servers, changing its state to `RUNNING`.
Use the returned `ResourceGroupLifeCycleTaskRuntimeMBean` value to check for completion.
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a `weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean` reference return field.

POST

Begins to start this resource group on the specified targets, changing its state to `RUNNING`.
Use the returned `ResourceGroupLifeCycleTaskRuntimeMBean` value to check for completion.
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

`targetNames`
the specific servers on which to start this resource group

**Type:** array of string

Response Body

Contains a JSON object with a `weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean` reference return field.
POST

Begins to start this resource group on the specified targets, changing its state to RUNNING

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

Please use ResourceGroupLifeCycleTaskRuntimeMBean.start(String[] targetNames)

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**targets**
the specific targets on which to start this resource group

Type: array of weblogic.management.configuration.TargetMBean references

Response Body

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST

POST

Begins to start this resource group in the admin state on all targeted servers, changing its state to **ADMIN**

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

This method supports both asynchronous and best effort synchronous invocations.

The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Operator

**Response Body**

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

POST

Begins to start this resource group in the admin state on the specified targets, changing its state to **ADMIN**

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

This method supports both asynchronous and best effort synchronous invocations.

The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

**targetNames**
the specific servers on which to start this resource group in "admin" state

**Type:** array of string
Response Body
Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

POST
Begins to start this resource group in the admin state on the specified targets, changing its state to ADMIN
Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion
Please use ResourceGroupLifeCycleTaskRuntimeMBean.startInAdmin(String[] targetNames)
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

targets
the specific targets on which to start this resource group in "admin" state
Type: array of weblogic.management.configuration.TargetMBean references

Response Body
Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/suspend

The resource supports the following methods:

■ POST
The resource supports the following methods:

- POST
- POST
- POST
- POST
- POST
- POST
- POST

POST

Begins to transition this resource group from RUNNING to ADMIN state gracefully on the specified targets.

Applications are in admin mode. Inflight work is completed.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion.

Please use ResourceGroupLifeCycleTaskRuntimeMBean.suspend(timeout, boolean ignoreSessions, String[] targetNames)

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**timeout**

Seconds to wait for the resource group to transition gracefully. The resource group automatically calls forceSuspend() after timeout.

Type: integer/int32

**ignoreSessions**

drop inflight HTTP sessions during graceful suspend

Type: boolean

**targets**

the specific targets on which to start this resource group

Type: array of weblogic.management.configuration.TargetMBean references
Response Body

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

POST

Begins the graceful transition of this resource group from the **RUNNING** state to the **ADMIN** state on all targeted servers. (equivalent to suspend(0, false)) New requests are rejected and inflight work is allowed to complete.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

POST

Begins to transition the resource group from **RUNNING** to **ADMIN** state gracefully on all targeted servers.

Applications are in admin mode. Inflight work is completed.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- **timeout**
  Seconds to wait for the resource group to transition gracefully. The resource group automatically calls forceSuspend() after timeout.
  
  Type: integer/int32

- **ignoreSessions**
  Drop inflight HTTP sessions during graceful suspend
Type: boolean

Response Body

Contains a JSON object with a
weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference
return field.

POST

Begins the graceful transition of this resource group from the **RUNNING** state to the
**ADMIN** state on the specified targets. (equivalent to suspend(0, false, targets)) New
requests are rejected and inflight work is allowed to complete.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for
completion

This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and
Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**targetNames**
the specific servers on which to suspend this resource group

Type: array of string

Response Body

Contains a JSON object with a
weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference
return field.

POST

Begins to transition this resource group from **RUNNING** to **ADMIN** state gracefully on the
specified targets.

Applications are in admin mode. Inflight work is completed.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for
completion

This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and
Asynchronous Operations.

Roles

Admin, Operator
POST

Request Body

Must contain a JSON object with the following fields:

- **timeout**
  Seconds to wait for the resource group to transition gracefully. The resource group automatically calls forceSuspend() after timeout.
  
  **Type:** integer/int32

- **ignoreSessions**
  drop inflight HTTP sessions during graceful suspend
  
  **Type:** boolean

- **targetNames**
  the specific servers on which to suspend this resource group
  
  **Type:** array of string

Response Body

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.

POST

Begins the graceful transition of this resource group from the RUNNING state to the ADMIN state on the specified targets. (equivalent to suspend(0, false, targets)) New requests are rejected and inflight work is allowed to complete.

Use the returned ResourceGroupLifeCycleTaskRuntimeMBean value to check for completion

Please use ResourceGroupLifeCycleTaskRuntimeMBean.suspend(String[] targetNames)

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- **targets**
  the specific targets on which to start this resource group
  
  **Type:** array of weblogic.management.configuration.TargetMBean references

Response Body

Contains a JSON object with a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean reference return field.
This resource manages a collection of `weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean` instances.

The resource supports the following methods:

- **GET**
GET

Get this collection of
`weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean`
instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of

This method can return the following links:

- `uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks
  rel=canonical`
  This resource.

- `uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]
  rel=parent`
  The parent resource of this resource.

- `uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks
  rel=self`
  This resource.
This resource manages a weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean instance.

The resource supports the following methods:

- GET
GET


Roles

Admin, Deployer, Monitor, Operator

Response Body


This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/cancel` rel=action title=cancel
  This resource's cancel action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}` rel=canonical
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks` rel=parent
  The parent resource of this resource.

- rel=parentTask
  This resource’s parentTask reference to a `weblogic.management.runtime.TaskRuntimeMBean` resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks` rel=subTasks
  This resource’s subTasks collection resource.
The resource supports the following methods:

- POST
POST

Attempts to cancel this task.
An exception is thrown to indicate failure to cancel the task. Not all tasks can be cancelled.

Roles

Admin, Operator

Response Body

Contains an empty JSON object.
This resource manages a collection of `weblogic.management.runtime.TaskRuntimeMBean` instances.

The resource supports the following methods:

- GET
GET

Get this collection of `weblogic.management.runtime.TaskRuntimeMBean` instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of `weblogic.management.runtime.TaskRuntimeMBean` entities.

This method can return the following links:

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks/[name]/subTasks rel=canonical`

  This resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks/[name] rel=parent`

  The parent resource of this resource.

- `uri=management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks/[name]/subTasks rel=self`

  This resource.
This resource manages a `weblogic.management.runtime.TaskRuntimeMBean` instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.TaskRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.TaskRuntimeMBean` entity.

This method can return the following links:

- uri=`/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks/[name]/subTasks/[name]/cancel` rel=action title=cancel
  
This resource's cancel action resource.

- uri=`/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks/[name]/subTasks/[name]` rel=canonical
  
This resource.

- uri=`/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks/[name]/subTasks` rel=parent
  
The parent resource of this resource.

- rel=parentTask
  
This resource's parentTask reference to a `weblogic.management.runtime.TaskRuntimeMBean` resource.

- uri=`/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks/[name]/subTasks/[name]` rel=self
  
This resource.

- uri=`/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/[name]/tasks/[name]/subTasks/[name]/subTasks` rel=subTasks
  
This resource's subTasks collection resource.
The resource supports the following methods:

- POST
POST

Attempts to cancel this task.
An exception is thrown to indicate failure to cancel the task. Not all tasks can be cancelled.

Roles
Admin

Response Body
Contains an empty JSON object.
This resource manages a collection of `weblogic.management.runtime.TaskRuntimeMBean` instances.

The resource supports the following methods:

- GET
GET

Get this collection of `weblogic.management.runtime.TaskRuntimeMBean` instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of `weblogic.management.runtime.TaskRuntimeMBean` entities.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name}/subTasks rel=canonical`
  
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name} rel=parent`
  
  The parent resource of this resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name}/subTasks rel=self`
  
  This resource.
This resource manages a `weblogic.management.runtime.TaskRuntimeMBean` instance. The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.TaskRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.TaskRuntimeMBean` entity.

This method can return the following links:

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes
  /{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name}/subTasks/{name}/cancel` rel=action title=cancel

  This resource’s cancel action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes
  /{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name}/subTasks/{name}` rel=canonical

  This resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes
  /{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name}/subTasks` rel=parent

  The parent resource of this resource.

- rel=parentTask

  This resource’s parentTask reference to a `weblogic.management.runtime.TaskRuntimeMBean` resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes
  /{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name}/subTasks/{name}` rel=self

  This resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes
  /{name}/partitionLifeCycleRuntime/resourceGroupLifeCycleRuntimes/{name}/tasks/{name}/subTasks/{name}/subTasks` rel=subTasks

  This resource’s subTasks collection resource.
The resource supports the following methods:

- POST
POST

Attempts to cancel this task.
An exception is thrown to indicate failure to cancel the task. Not all tasks can be cancelled.

Roles

Admin

Response Body

Contains an empty JSON object.
The resource supports the following methods:

- POST
The resource supports the following methods:

- POST
- POST
- POST

POST

Begins the transition of the partition from the `ADMIN` state to the `RUNNING` state on all targeted servers.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean reference return` field.

POST

Begins the transition of the partition from the `ADMIN` state to the `RUNNING` state on the specified targets.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion.

Please use `PartitionLifeCycleRuntimeMBean.resume(String[] targetNames)`.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- **targets**
  the specific targets on which to resume this partition

  Type: array of `weblogic.management.configuration.TargetMBean references`
Response Body
Contains a JSON object with a

POST
Begins the transition of the partition from the ADMIN state to the RUNNING state on the specified targets.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

targetNames
the specific servers on which to resume this partition
Type: array of string

Response Body
Contains a JSON object with a
The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST
- POST
- POST
- POST
- POST
- POST
- POST

POST

Begins to shut down this partition gracefully on the specified targets.
(Equivalent to shutdown(targets, 0, false)).

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion

Please use PartitionLifeCycleRuntimeMBean.shutdown(String[] targetNames)

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

targets
the specific targets on which to shut down this partition

Type: array of weblogic.management.configuration.TargetMBean references

Response Body

Contains a JSON object with a

POST

Begins to shut down this partition gracefully on the specified targets.
(Equivalent to shutdown(timeout, ignoreSessions, false, targets)).

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

**timeout**
Number of seconds to wait before aborting inflight work and shutting down the server.

**Type**: integer/int32

**ignoreSessions**
true indicates ignore pending HTTP sessions during inflight work handling.

**Type**: boolean

**targetNames**
the specific servers on which to shut down this partition

**Type**: array of string

Response Body
Contains a JSON object with a weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean reference return field.

POST
Begins to shut down this partition gracefully on the specified targets.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
Please use PartitionLifeCycleRuntimeMBean.shutdown(int timeout, boolean ignoreSessions, boolean waitForAllSessions, String[] targetNames)
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

**timeout**
Number of seconds to wait before aborting inflight work and force shutting down the server.

**Type**: integer/int32
ignoreSessions
Set to true to ignore pending HTTP sessions during inflight work handling.
Type: boolean

waitForAllSessions
Set to true to wait for all HTTP sessions during inflight work handling; false to wait for non-persisted HTTP sessions only.
Type: boolean

targets
the specific targets on which to shut down this partition
Type: array of weblogic.management.configuration.TargetMBean references

Response Body
Contains a JSON object with a

POST
Begins to shut down this partition gracefully on all targeted servers.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

timeout
Number of seconds to wait before aborting inflight work and force shutting down the server.
Type: integer/int32

ignoreSessions
Set to true to ignore pending HTTP sessions during inflight work handling.
Type: boolean

waitForAllSessions
Set to true to wait for all HTTP sessions during inflight work handling; false to wait for non-persisted HTTP sessions only.
Type: boolean
**Response Body**
Contains a JSON object with a

**POST**
Begins to shut down this partition gracefully on the specified targets.
(Equivalent to shutdown(timeout, ignoreSessions, false, targets)).
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
Please use PartitionLifeCycleRuntimeMBean.shutdown(int timeout, boolean ignoreSessions, String[] targetNames)
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**
Admin, Operator

**Request Body**
Must contain a JSON object with the following fields:

- **timeout**
  Number of seconds to wait before aborting inflight work and shutting down the server.
  **Type**: integer/int32

- **ignoreSessions**
  true indicates ignore pending HTTP sessions during inflight work handling.
  **Type**: boolean

- **targets**
  the specific targets on which to shut down this partition
  **Type**: array of weblogic.management.configuration.TargetMBean references

**Response Body**
Contains a JSON object with a

**POST**
Begins to shut down this partition gracefully on the specified targets.
(Equivalent to shutdown(targets, 0, false)).
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.
Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**targetNames**
the specific servers on which to shut down this partition

Type: array of string

Response Body

Contains a JSON object with a

POST

Begins to shut down this partition gracefully on all targeted servers.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a

POST

Begins to shut down this partition gracefully on the specified targets.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

**timeout**
Number of seconds to wait before aborting inflight work and force shutting down the server.
**POST**

Type: integer/int32

**ignoreSessions**
Set to true to ignore pending HTTP sessions during inflight work handling.

Type: boolean

**waitForAllSessions**
Set to true to wait for all HTTP sessions during inflight work handling; false to wait for non-persisted HTTP sessions only.

Type: boolean

**targetNames**
the specific servers on which to shut down this partition

Type: array of string

**Response Body**
Contains a JSON object with a weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean reference return field.

**POST**
Begins to shut down this partition gracefully on all targeted servers.  
(Equivalent to shutdown(timeout, ignoreSessions, false, targets)).

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**
Admin, Operator

**Request Body**
Must contain a JSON object with the following fields:

**timeout**
Number of seconds to wait before aborting inflight work and shutting down the server.

Type: integer/int32

**ignoreSessions**
false indicates ignore pending HTTP sessions during inflight work handling.

Type: boolean

**Response Body**
Contains a JSON object with a weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean reference return field.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/start

The resource supports the following methods:

- POST
The resource supports the following methods:

- POST
- POST
- POST
- POST
- POST

Begins to start this partition on all targeted servers.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Response Body
Contains a JSON object with a weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean reference return field.

Begins to start this partition in the specified initialState within the specified timeOut on all targeted servers.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

- initialState
can be ADMIN or RUNNING, default is RUNNING.
Type: string

- timeOut
Specifies the number of milliseconds to start a Partition. Throws InterruptedException if Partition is unable to start during that duration and leaves the Partition in UNKNOWN state. Default is 60000ms
Type: integer/int32
Response Body
Contains a JSON object with a

POST
Begins to start this partition on the specified servers.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

**targetNames**
the specific servers on which to start this partition
Type: array of string

Response Body
Contains a JSON object with a

POST
Begins to start this partition on the specified targets.
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion
Please use PartitionLifeCycleRuntimeMBean.start(String[] targetNames)
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles
Admin, Operator

Request Body
Must contain a JSON object with the following fields:

**targets**
the specific targets on which to start this partition
Type: array of weblogic.management.configuration.TargetMBean references
Response Body

Contains a JSON object with a `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` reference return field.
The resource supports the following methods:

- POST
The resource supports the following methods:

- POST
- POST
- POST
- POST

**POST**

Begins to start this partition in the **ADMIN** state on all targeted servers.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**
Admin, Operator

**Response Body**

Contains a JSON object with a `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` reference return field.

**POST**

Begins to start this partition in the **ADMIN** state on the specified targets.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**
Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

- **targetNames**
  the specific servers on which to start this partition in "admin" state
  
  **Type**: array of string

**Response Body**

Contains a JSON object with a `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` reference return field.
POST

Begins to start this partition in the ADMIN state on the specified targets.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion

Use the method `PartitionLifeCycleRuntimeMBean.startInAdmin(String[] targetNames)`

This method supports both asynchronous and best effort synchronous invocations.

The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- **targets**
  the specific targets on which to start this partition in "admin" state

**Type**: array of `weblogic.management.configuration.TargetMBean` references

Response Body

Contains a JSON object with a

The resource supports the following methods:

- POST
POST

The resource supports the following methods:

- POST
- POST
- POST
- POST
- POST
- POST
- POST

POST

Begins the graceful transition of the partition from the **RUNNING** state to the **ADMIN** state on the specified targets.

Use the returned **PartitionLifeCycleTaskRuntimeMBean** value to check for completion.

Applications are in admin mode. Inflight work is completed.

Please use

```java
PartitionLifeCycleRuntimeMBean.suspend(int timeout, boolean ignoreSessions, String[] targetNames)
```

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

- **timeout**
  seconds to wait for the partition to transition gracefully. The partition automatically calls `forceSuspend()` after timeout.

  *Type: integer/int32*

- **ignoreSessions**
  drop inflight HTTP sessions during graceful suspend

  *Type: boolean*

- **targets**
  the specific targets on which to suspend this partition

  *Type: array of weblogic.management.configuration.TargetMBean references*

Response Body

Contains a JSON object with a

POST

Begins the graceful transition of the partition from the \texttt{RUNNING} state to the \texttt{ADMIN} state on all targeted servers.

(Equivalent to \texttt{suspend(0, false)})

Use the returned \texttt{PartitionLifeCycleTaskRuntimeMBean} value to check for completion.

New requests are rejected and inflight work is allowed to complete.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Response Body

Contains a JSON object with a \texttt{weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean} reference return field.

POST

Begins the graceful transition of the partition from the \texttt{RUNNING} state to the \texttt{ADMIN} state on all targeted servers.

Use the returned \texttt{PartitionLifeCycleTaskRuntimeMBean} value to check for completion.

Applications are in admin mode. Inflight work is completed.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, Operator

Request Body

Must contain a JSON object with the following fields:

\texttt{timeout}

Seconds to wait for the partition to transition gracefully. The partition automatically calls \texttt{forceSuspend()} after timeout.

\textbf{Type}: integer/int32

\texttt{ignoreSessions}

\texttt{drop} inflight HTTP sessions during graceful suspend

\textbf{Type}: boolean

Response Body

Contains a JSON object with a \texttt{weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean} reference return field.
POST

Begins the graceful transition of the partition from the **RUNNING** state to the **ADMIN** state on the specified targets. (Equivalent to `suspend(0, false, targets)`)

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion

New requests are rejected and inflight work is allowed to complete.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

- **targetNames**
  the specific servers on which to suspend this partition
  Type: array of string

**Response Body**

Contains a JSON object with a `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` reference return field.

---

POST

Begins the graceful transition of the partition from the **RUNNING** state to the **ADMIN** state on the specified targets.

Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion

Applications are in admin mode. Inflight work is completed.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

**Roles**

Admin, Operator

**Request Body**

Must contain a JSON object with the following fields:

- **timeout**
  Seconds to wait for the partition to transition gracefully. The partition automatically calls `forceSuspend()` after timeout.
  Type: integer/int32

- **ignoreSessions**
  drop inflight HTTP sessions during graceful suspend
POST  

Type: boolean

**targetNames**  
the specific servers on which to suspend this partition  
Type: array of string

**Response Body**  
Contains a JSON object with a  

POST  

Begins the graceful transition of the partition from the **RUNNING** state to the **ADMIN** state on the specified targets. (Equivalent to suspend(0, false, targets))  
Use the returned PartitionLifeCycleTaskRuntimeMBean value to check for completion  
New requests are rejected and inflight work is allowed to complete.  
Please use PartitionLifeCycleRuntimeMBean.suspend(String[] targetNames)  
This method supports both asynchronous and best effort synchronous invocations.  
The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

**Roles**  
Admin, Operator

**Request Body**  
Must contain a JSON object with the following fields:  

**targets**  
the specific targets on which to suspend this partition  
Type: array of weblogic.management.configuration.TargetMBean references

**Response Body**  
Contains a JSON object with a  
This resource manages a collection of `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` instances.

The resource supports the following methods:

- GET
GET

Get this collection of 

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of 
weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean entities.

This method can return the following links:

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime/tasks rel=canonical
  This resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime rel=parent
  The parent resource of this resource.

- uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes
  /[name]/partitionLifeCycleRuntime/tasks rel=self
  This resource.
This resource manages a

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean` entity.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/cancel` `rel=action` `title=cancel`
  This resource’s cancel action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}` `rel=canonical`
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/parent` `rel=parent`
  The parent resource of this resource.

- `rel=parentTask`
  This resource’s parentTask reference to a `weblogic.management.runtime.TaskRuntimeMBean` resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/self` `rel=self`
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks` `rel=subTasks`
  This resource’s subTasks collection resource.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/cancel

The resource supports the following methods:

- POST
POST

Attempts to cancel this task.
An exception is thrown to indicate failure to cancel the task. Not all tasks can be cancelled.

Roles
Admin, Operator

Response Body
Contains an empty JSON object.
This resource manages a collection of 
The resource supports the following methods:

- GET
GET

Get this collection of `weblogic.management.runtime.TaskRuntimeMBean` instances.

**Roles**

Admin, Deployer, Monitor, Operator

**Response Body**

Contains a list of `weblogic.management.runtime.TaskRuntimeMBean` entities.

This method can return the following links:

- uri=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks rel=canonical`
  This resource.

- uri=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name} rel=parent`
  The parent resource of this resource.

- uri=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks rel=self`
  This resource.
This resource manages a `weblogic.management.runtime.TaskRuntimeMBean` instance. The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.TaskRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.TaskRuntimeMBean` entity.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks/{name}/cancel`
  `rel=action` `title=cancel`
  This resource’s cancel action resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks/{name}`
  `rel=canonical`
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks` `rel=parent`
  The parent resource of this resource.

- `rel=parentTask`
  This resource’s parentTask reference to a `weblogic.management.runtime.TaskRuntimeMBean` resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks/{name}/subTasks` `rel=self`
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks/subTasks`
  This resource’s subTasks collection resource.
/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks/{name}/cancel

The resource supports the following methods:

- POST
POST

Attempts to cancel this task.
An exception is thrown to indicate failure to cancel the task. Not all tasks can be cancelled.

Roles

Admin

Response Body

Contains an empty JSON object.
This resource manages a collection of weblogic.management.runtime.TaskRuntimeMBean instances.

The resource supports the following methods:

- GET
GET

Get this collection of `weblogic.management.runtime.TaskRuntimeMBean` instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of `weblogic.management.runtime.TaskRuntimeMBean` entities.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks/{name}/subTasks` `rel=canonical`
  
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks/{name}` `rel=parent`
  
  The parent resource of this resource.

- `uri=/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionLifeCycleRuntime/tasks/{name}/subTasks/{name}/subTasks` `rel=self`
  
  This resource.
This resource manages a `weblogic.management.runtime.TaskRuntimeMBean` instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.TaskRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.TaskRuntimeMBean` entity.

This method can return the following links:

- `uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/tasks/[name]/subTasks/[name]/subTasks/[name]/cancel rel=action title=cancel`
  This resource’s cancel action resource.

- `uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/tasks/[name]/subTasks/[name]/subTasks/[name]/canonical`
  This resource.

- `uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/tasks/[name]/subTasks/[name]/subTasks/[name]/parent`
  The parent resource of this resource.

- `rel=parentTask`
  This resource’s parentTask reference to a `weblogic.management.runtime.TaskRuntimeMBean` resource.

- `uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/tasks/[name]/subTasks/[name]/subTasks/[name]/self`
  This resource.

- `uri=/management/weblogic/[version]/domainRuntime/domainPartitionRuntimes/[name]/partitionLifeCycleRuntime/tasks/[name]/subTasks/[name]/subTasks/[name]/subTasks`
  This resource’s subTasks collection resource.
The resource supports the following methods:

- POST
POST

Attempts to cancel this task.
An exception is thrown to indicate failure to cancel the task. Not all tasks can be cancelled.

Roles

Admin

Response Body

Contains an empty JSON object.
This resource manages a `weblogic.management.runtime.PartitionUserFileSystemManagerMBean` instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.PartitionUserFileSystemManagerMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.PartitionUserFileSystemManagerMBean` entity.

This method can return the following links:

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionUserFileSystemManager/deleteFile rel=action title=deleteFile`
  
  This resource’s deleteFile action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionUserFileSystemManager/exists rel=action title=exists`
  
  This resource’s exists action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionUserFileSystemManager/list rel=action title=list`
  
  This resource’s list action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionUserFileSystemManager/mkdir rel=action title=mkdir`
  
  This resource’s mkdir action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionUserFileSystemManager/rmdir rel=action title=rmdir`
  
  This resource’s rmdir action resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionUserFileSystemManager rel=canonical`
  
  This resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name} rel=parent`
  
  The parent resource of this resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/domainPartitionRuntimes/{name}/partitionUserFileSystemManager rel=self`
  
  This resource.
The resource supports the following methods:

- POST
POST

Deletes specified file

Roles

Admin

Request Body

Must contain a JSON object with the following fields:

- **filePath**
  - file path relative to partition user file system root
  - **Type**: string

Response Body

Contains a JSON object with a boolean return field.
The resource supports the following methods:

- POST
POST

Checks whether specified file exists

Roles

Admin

Request Body

Must contain a JSON object with the following fields:

- **filePath**
  file path relative to partition user file system root
  - Type: string

Response Body

Contains a JSON object with a boolean return field.
The resource supports the following methods:

- POST
POST

Lists files from partition user file system directory

Roles

Admin

Request Body

Must contain a JSON object with the following fields:

- **dirPath**
  directory path relative to partition user file system root
  Type: string

Response Body

Contains a JSON object with a array of string return field.
The resource supports the following methods:

- POST
POST

Creates partition user file system subdirectory

Roles

Admin

Request Body

Must contain a JSON object with the following fields:

- **dirPath**
  directory path relative to partition user file system root
  
- **Type**: string

Response Body

Contains a JSON object with a boolean return field.
The resource supports the following methods:

- **POST**
POST

Deletes specified directory

Roles

Admin

Request Body

Must contain a JSON object with the following fields:

- **dirPath**
  - directory path relative to partition user file system root
  - **Type**: string

Response Body

Contains a JSON object with a boolean return field.
This resource manages a weblogic.management.runtime.RolloutServiceRuntimeMBean instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.RolloutServiceRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.RolloutServiceRuntimeMBean` entity.

This method can return the following links:

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/deleteWorkflow` rel=action title=deleteWorkflow
  This resource’s deleteWorkflow action resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/executeWorkflow` rel=action title=executeWorkflow
  This resource’s executeWorkflow action resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/getWorkflowTask` rel=action title=getWorkflowTask
  This resource’s getWorkflowTask action resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/revertWorkflow` rel=action title=revertWorkflow
  This resource’s revertWorkflow action resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/rollingRestart` rel=action title=rollingRestart
  This resource’s rollingRestart action resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/rolloutApplications` rel=action title=rolloutApplications
  This resource’s rolloutApplications action resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/allWorkflows` rel=allWorkflows
  This resource’s allWorkflows collection resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/parent` rel=canonical
  This resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/inactiveWorkflows` rel=inactiveWorkflows
  This resource’s inactiveWorkflows collection resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService` rel=parent
  The parent resource of this resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService` rel=self
  This resource.
- **uri**=management/weblogic/{version}/domainRuntime/rolloutService/stoppedWorkflows  
  rel=stoppedWorkflows

  This resource's stoppedWorkflows collection resource.
This resource manages a collection of
weblogic.management.runtime.WorkflowTaskRuntimeMBean instances.

The resource supports the following methods:

■ GET
GET

Get this collection of `weblogic.management.runtime.WorkflowTaskRuntimeMBean` instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of `weblogic.management.runtime.WorkflowTaskRuntimeMBean` entities.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService/allWorkflows rel=canonical`
  
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService rel=parent`
  
  The parent resource of this resource.

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService/allWorkflows rel=self`
  
  This resource.
This resource manages a
`weblogic.management.runtime.WorkflowTaskRuntimeMBean` instance.

The resource supports the following methods:

- **GET**
GET

Get this `weblogic.management.runtime.WorkflowTaskRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.WorkflowTaskRuntimeMBean` entity.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService/allWorkflows/{name} rel=canonical`
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService/allWorkflows rel=parent`
  The parent resource of this resource.

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService/allWorkflows/{name} rel=self`
  This resource.
The resource supports the following methods:

- POST
POST

Delete workflow data of finished workflow

Roles

Admin, admin

Request Body

Must contain a JSON object with the following fields:

- `workflowId`  
  is ID of workflow to be deleted - `@code WorkflowTaskRuntimeMBean.getWorkflowId()`  
  Type: string

Response Body

Contains an empty JSON object.
The resource supports the following methods:

- POST
POST

Continue in workflow execution. Can be called only for terminated workflows. This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the 'Prefer' header. See About Synchronous and Asynchronous Operations.

Roles

Admin, admin

Request Body

Must contain a JSON object with the following fields:

- **wfTask**
  - which should be executed - continued. [@code delegate.isStopped()] must be [@code null]

  **Type:** weblogic.management.runtime.WorkflowTaskRuntimeMBean reference

Response Body

Contains a JSON object with a
weblogic.management.runtime.WorkflowTaskRuntimeMBean reference return field.
The resource supports the following methods:

- POST
POST

Returns [@code WorkflowTaskRuntimeMBean] for given id. Progress must NOT be yet deleted and must be started by this service.

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

Roles

Admin, admin

Request Body

Must contain a JSON object with the following fields:

- **id**
  - of executed workflow
  - Type: string

Response Body

Contains a JSON object with a [weblogic.management.runtime.WorkflowTaskRuntimeMBean] reference return field.
This resource manages a collection of
weblogic.management.runtime.WorkflowTaskRuntimeMBean instances.

The resource supports the following methods:

- GET
GET

Get this collection of `weblogic.management.runtime.WorkflowTaskRuntimeMBean` instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of `weblogic.management.runtime.WorkflowTaskRuntimeMBean` entities.

This method can return the following links:

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/inactiveWorkflows` rel=canonical
  This resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService` rel=parent
  The parent resource of this resource.

- uri=`/management/weblogic/{version}/domainRuntime/rolloutService/inactiveWorkflows` rel=self
  This resource.
This resource manages a `weblogic.management.runtime.WorkflowTaskRuntimeMBean` instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.WorkflowTaskRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.WorkflowTaskRuntimeMBean` entity.

This method can return the following links:

- **uri**=`/management/weblogic/{version}/domainRuntime/rolloutService/inactiveWorkflows/{name}` **rel**=canonical
  - This resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/rolloutService/inactiveWorkflows` **rel**=parent
  - The parent resource of this resource.

- **uri**=`/management/weblogic/{version}/domainRuntime/rolloutService/inactiveWorkflows/{name}` **rel**=self
  - This resource.
The resource supports the following methods:

- POST
POST

Reverts workflow execution. Can be called only for terminated workflows.
This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the 'Prefer' header. See About Synchronous and
Asynchronous Operations.

Roles

Admin, admin

Request Body

Must contain a JSON object with the following fields:

- **wfTask**
  which should be reverted. `delegate.isStopped()` must be `null`
  Type: `weblogic.management.runtime.WorkflowTaskRuntimeMBean` reference

Response Body

Contains a JSON object with a
`weblogic.management.runtime.WorkflowTaskRuntimeMBean` reference return field.
The resource supports the following methods:

- POST
POST

Creates and starts a workflow that can be used to update restart servers, one at a time.
rolloutUpdate(target, [patchedOracleHome, backupOracleHome, isRollback],
[javaHome], [applicationProperties], [options])

This method supports both asynchronous and best effort synchronous invocations.
The behavior is controlled by the ‘Prefer’ header. See About Synchronous and
Asynchronous Operations.

Roles

Admin, admin

Request Body

Must contain a JSON object with the following fields:

**target**

can be the name of a domain, clusters or servers

**Type:** string

**options**

comma separated list of <name>=<value> options

**Type:** string

Response Body

Contains a JSON object with

weblogic.management.runtime.WorkflowTaskRuntimeMBean reference return field.
The resource supports the following methods:

- POST
POST

Creates and starts a workflow that can be used to update Applications.
rolloutUpdate(target, [applicationProperties], [options])

This method supports both asynchronous and best effort synchronous invocations. The behavior is controlled by the ‘Prefer’ header. See About Synchronous and Asynchronous Operations.

Roles

Admin, admin

Request Body

Must contain a JSON object with the following fields:

**target**
can be the name of a domain, clusters or servers
Type: string

**applicationProperties**
the path of a formatted text file containing the information needed to upgrade applications
Type: string

**options**
comma separated list of <name>=<value> options
Type: string

Response Body

Contains a JSON object with a
weblogic.management.runtime.WorkflowTaskRuntimeMBean reference return field.
This resource manages a collection of `weblogic.management.runtime.WorkflowTaskRuntimeMBean` instances.

The resource supports the following methods:

- GET
GET

Get this collection of `weblogic.management.runtime.WorkflowTaskRuntimeMBean` instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of `weblogic.management.runtime.WorkflowTaskRuntimeMBean` entities.

This method can return the following links:

- `uri=`/management/weblogic/{version}/domainRuntime/rolloutService/stoppedWorkflows rel=canonical
  This resource.
- `uri=`/management/weblogic/{version}/domainRuntime/rolloutService rel=parent
  The parent resource of this resource.
- `uri=`/management/weblogic/{version}/domainRuntime/rolloutService/stoppedWorkflows rel=self
  This resource.
This resource manages a `weblogic.management.runtime.WorkflowTaskRuntimeMBean` instance.

The resource supports the following methods:

- GET
GET

Get this `weblogic.management.runtime.WorkflowTaskRuntimeMBean` instance.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a `weblogic.management.runtime.WorkflowTaskRuntimeMBean` entity.

This method can return the following links:

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService/stoppedWorkflows/{name}` rel=canonical
  This resource.

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService/stoppedWorkflows` rel=parent
  The parent resource of this resource.

- `uri=/management/weblogic/{version}/domainRuntime/rolloutService/stoppedWorkflows/{name}` rel=self
  This resource.
The resource supports the following methods:

- **POST**
POST

Search the domainRuntime tree.

Roles

Admin, Deployer, Monitor, Operator

Request Body

The search request body includes the search criteria.

Response Body

The search response includes the results of the search.
This resource manages a collection of weblogic.management.runtime.ServerRuntimeMBean instances.

Monitors all the running servers in the WebLogic Server domain from the administration server. From this resource, you can monitor each managed server.

The resource supports the following methods:

- GET
GET

Get this collection of weblogic.management.runtime.ServerRuntimeMBean instances.

Roles

Admin, Deployer, Monitor, Operator

Response Body

Contains a list of weblogic.management.runtime.ServerRuntimeMBean entities.

This method can return the following links:

- uri=/management/weblogic/{version}/domainRuntime/serverRuntimes
  rel=canonical
  This resource.

- uri=/management/weblogic/{version}/domainRuntime
  rel=parent
  The parent resource of this resource.

- uri=/management/weblogic/{version}/domainRuntime/serverRuntimes
  rel=self
  This resource.
The following sections describe the data models on which the REST resources operate. These data models describe the information exchanged in REST resources.

**java.lang.Throwable**

This entity represents errors and exceptions.

The properties of the java.lang.Throwable entity are as follows:

- **cause**
  This property contains the stack trace of errors and exceptions.
  Type: java.lang.Throwable

- **message**
  This property contains the reason why the error or exception was thrown.
  Type: string

**java.util.Properties**

This entity represents a JSONObject with a property for each property in the properties object, where the key is the property’s name the the value is a string containing the property’s value.

**weblogic.health.HealthState**

This entity represents the health of the services that are in a WebLogic server domain. Selected subsystems within each WebLogic Server instance monitor their health status based on criteria specific to the subsystem.

The properties of the weblogic.health.HealthState entity are as follows:

- **partitionName**
  This property indicates the partition name for which the health state is reported.
  Type: string

- **state**
  This property indicates the reported state as:
  - OK - Service is healthy.
  - WARN - Service could have problems in the future
- CRITICAL - Something must be done now to prevent service failure
- FAILED - Service has failed and must be restarted
- OVERLOADED - Service is functioning normally but there is too much work in it
- UNKNOWN - Service state cannot be determined

Type: string

subsystemName
This property indicates the service for which the health state is reported.
Type: string

symptoms
This property indicates the cause of the reported state.
Type: array of weblogic.health.Symptom

weblogic.health.Symptom
This entity represents an observation reported by a service with its health state. A symptom can be used to determine the reason for the current health state of a WebLogic server.

The properties of the weblogic.health.Symptom entity are as follows:

info
This property provides additional information about the symptom if available.
Type: string

instanceId
This property represents an identifier for the symptom.
Type: string

severity
This property indicates a low, medium, or high severity level.
Type: string

type
This property represents the cause of the symptom (low memory, network error, stuck threads, and so on).
Type: string

weblogic.management.rest.bean.model.CurrentUser
The authenticated user that is making the request.

The properties of the weblogic.management.rest.bean.model.CurrentUser entity are as follows:

partitionId
The ID of the partition where the component that is being invoked resides.
Type: string
**partitionName**
The human readable name of the partition ID.
*Type: string*

**user**
The login of the user making this request, if the user has been authenticated, or null if the user has not been authenticated.
*Type: string*

---

**weblogic.management.rest.bean.model.Version**
A Version entity describes a version of the WebLogic Server resources.
The properties of the weblogic.management.rest.bean.model.Version entity are as follows:

**isLatest**
True if this is the default version.
*Type: boolean*

**lifecycle**
The lifecycle of this version: active or deprecated
*Type: string*

**version**
The name of this version.
*Type: string*

---

**weblogic.management.runtime.AppDeploymentRuntimeMBean**
This MBean provides deployment operations for an application. Currently only start and stop are supported. In the future, this MBean may be enhanced with operations to support deployment applications to the domain as well as extended WLS deployment features such as production redeployment and partial deployment of modules in an enterprise application.
The properties of the weblogic.management.runtime.AppDeploymentRuntimeMBean entity are as follows:

**applicationName**
The application's name.
*Type: string*
*Constraints: Read Only*

**applicationVersion**
The application's version identifier.
*Type: string*
*Constraints: Read Only*

**modules**
The list of modules for the application. These modules can be used in module level targeting.
weblogic.management.runtime.AppRuntimeStateRuntimeMBean

Provides access to runtime state for deployed applications.

The properties of the weblogic.management.runtime.AppRuntimeStateRuntimeMBean entity are as follows:

applicationIds
Type: array of string
Constraints: Read Only

name
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.
Type: string
Constraints: Read Only

type
Returns the type of the MBean.
Type: string
Constraints: Read Only

weblogic.management.runtime.DeploymentManagerMBean

This MBean provides deployment operations.

A DeploymentManager object is a stateless interface into the WebLogic Server deployment framework. It currently provides access to the AppDeploymentRuntimeMBeans that allow the user to start and stop deployments. In the future, this MBean may be enhanced with operations to support deployment applications to the domain, as well as extended WebLogic Server deployment features,
such as production redeployment and partial deployment of modules in an enterprise application.

This MBean emits notifications when an application is created or removed and when the application state changes. The notification types are:

- appdeployment.created
- appdeployment.deleted
- appdeployment.state.new
- appdeployment.state.prepared
- appdeployment.state.admin
- appdeployment.state.active
- appdeployment.state.retired
- appdeployment.state.failed
- appdeployment.state.update.pending
- appdeployment.state.unknown

The userdata is the object name of the application.

The properties of the weblogic.management.runtime.DeploymentManagerMBean entity are as follows:

- **maximumDeploymentProgressObjectsCount**
  The maximum number of progress objects allowed.
  
  **Type**: integer/int32

- **name**
  The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.
  
  **Type**: string
  
  **Constraints**: Read Only

- **type**
  Returns the type of the MBean.
  
  **Type**: string
  
  **Constraints**: Read Only

**weblogic.management.runtime.DeploymentProgressObjectMBean**

This MBean is the user API for monitoring deployment operations and exists only on an Administration Server. Currently only start and stop operations initiated by AppDeploymentRuntimeMBean are supported.

The properties of the weblogic.management.runtime.DeploymentProgressObjectMBean entity are as follows:

- **applicationName**
  The name of the application for the current deployment operation.
  
  **Type**: string
Constraints: Read Only

completed
This property indicates whether an asynchronous job has completed.
Type: boolean

deploymentMessages
Provides an ordered array of status messages generated for the current deployment operation.
Type: array of string
Constraints: Read Only

d endTime
This property indicates the time when the asynchronous job ended.
Type: string/date

failedTargets
The targets on which the current deployment operation failed.
Type: array of string
Constraints: Read Only

id
The unique ID for the current deployment operation.
Type: string
Constraints: Read Only

name
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.
Type: string
Constraints: Read Only

operationType
The deployment operation type for the current deployment operation. Possible values are 1 (start) and 2 (stop).
Type: integer/int32
Constraints: Read Only

progress
This property indicates whether an asynchronous job has succeeded, failed, or is in progress.
Type: string

rootExceptions
If the current deployment operation has failed, this method may return zero or more exception(s) which represent the root cause of the failure. The array will not contain WLS exception classes; instead they will be new Exceptions containing the stack traces and messages from the original WLS Exceptions.
Type: array of java.lang.Throwable
weblogic.management.runtime.DomainPartitionRuntimeMBean

Constraints: Read Only

**startTime**
This property indicates the time when an asynchronous job was started.

Type: string/date

**startTimeAsLong**
The time that the current deployment operation began. The value is in milliseconds consistent with the system time.

Type: integer/int64

Constraints: Read Only

**state**
The state of the current deployment operation. Possible values are STATE_INITIALIZED, STATE_RUNNING, STATE_COMPLETED, STATE_FAILED and STATE_DEFERRED.

Type: string

Constraints: Read Only

**targets**
The targets specified for the current deployment operation.

Type: array of string

Constraints: Read Only

**type**
Returns the type of the MBean.

Type: string

Constraints: Read Only

weblogic.management.runtime.DomainPartitionRuntimeMBean

This class is used for domain level partition handling.

The properties of the weblogic.management.runtime.DomainPartitionRuntimeMBean entity are as follows:

**name**
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.

Type: string

Constraints: Read Only

**partitionID**
Returns partition ID value for corresponding partition.

Type: string

Constraints: Read Only

**type**
Returns the type of the MBean.

Type: string
weblogic.management.runtime.DomainRuntimeMBean

This class is used for monitoring a WebLogic domain. A domain may contain zero or more clusters. A cluster may be looked up by a logical name.

The properties of the weblogic.management.runtime.DomainRuntimeMBean entity are as follows:

**currentDomainPartitionRuntime**
Returns the domain partition runtime MBean for the "current" partition

*Type:* weblogic.management.runtime.DomainPartitionRuntimeMBean reference

*Constraints:* Read Only

**name**
The user-specified name of this MBean instance.

This name is included as one of the key properties in the MBean's javax.management.ObjectName

*Name= user-specified-name*

*Type:* string

*Constraints:* Read Only

**type**
Returns the type of the MBean.

*Type:* string

*Constraints:* Read Only

weblogic.management.runtime.EditSessionConfigurationManagerMBean

The properties of the weblogic.management.runtime.EditSessionConfigurationManagerMBean entity are as follows:

**name**
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.

*Type:* string

*Constraints:* Read Only

**type**
Returns the type of the MBean.

*Type:* string

*Constraints:* Read Only

weblogic.management.runtime.EditSessionConfigurationRuntimeMBean

The properties of the weblogic.management.runtime.EditSessionConfigurationRuntimeMBean entity are as follows:
**creator**
The original creator of the edit session.
*Type*: string
*Constraints*: Read Only

**currentEditor**
The current editor of the edit session. The editor owns the lock.
*Type*: string
*Constraints*: Read Only

**description**
The description of the edit session configuration.
*Type*: string
*Constraints*: Read Only

**editSessionName**
The name of the edit session configuration.
*Type*: string
*Constraints*: Read Only

**editSessionServerJndi**
The JNDI name of the related `EditSessionServer`
*Type*: string
*Constraints*: Read Only

**mergeNeeded**
Returns `true` if another edit session activates its changes after the last activation of this edit session or after its creation but before this edit session was ever activated.

It indicates that this edit session configuration is probably not the actual runtime configuration. Use the resolve method to merge changes to this configuration.

If this edit session configuration was also modified then there can be conflicts between this session and the runtime configuration. Use the resolve method to remove possible inconsistencies.
*Type*: boolean
*Constraints*: Read Only

**name**
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.
*Type*: string
*Constraints*: Read Only

**partitionName**
The partition name to which this edit session belongs.
*Type*: string
*Constraints*: Read Only
**weblogic.management.runtime.LibDeploymentRuntimeMBean**

This MBean provides deployment operations for a library.

The properties of the `weblogic.management.runtime.LibDeploymentRuntimeMBean` entity are as follows:

- **implementationVersion**
  The library's implementation version, null if none is set
  
  **Type:** string
  
  **Constraints:** Read Only

- **libraryIdentifier**
  The library's identifier.
  
  **Type:** string
  
  **Constraints:** Read Only

- **libraryName**
  The library's name.
  
  **Type:** string
  
  **Constraints:** Read Only

- **name**
  The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.
  
  **Type:** string
  
  **Constraints:** Read Only

- **partitionName**
  The name of the partition the library is in.
  
  **Type:** string
  
  **Constraints:** Read Only

- **specificationVersion**
  The library's specification version, null if none is set
  
  **Type:** string
  
  **Constraints:** Read Only

- **type**
  Returns the type of the MBean.
  
  **Type:** string
  
  **Constraints:** Read Only
weblogic.management.runtime.PartitionLifeCycleRuntimeMBean

Provides methods that transition partitions from one state to another. This class is instantiated once for each partition but only on the Administration Server. Even so, you can use a partition’s instance to change the state of the partition on Managed Servers as well as on the Administration Server.

Detecting when Operations Complete

Methods representing operations on partitions return a PartitionLifeCycleTaskRuntimeMBean object. The caller should monitor the returned instance, invoking its PartitionLifeCycleTaskRuntimeMBean#isRunning() method to detect when the operation has completed. In particular, the caller should not rely on the value returned from getState() to indicate when an operation on a partition that is targeted to multiple servers has finished. That value will be accurate as described in detail on getState() but can change before the operation has finished.

Desired States

The system tracks the desired state for each partition, which defaults to the state SHUTDOWN and substate HALTED. Whenever a lifecycle operation method is called the system not only performs the requested action on the partition but also records the resulting state as the new desired state for the partition.

The system uses the desired state for a partition to bring the partition to the correct state during server restart. It can also be used when the getState() method computes the overall state for the partition.

The properties of the weblogic.management.runtime.PartitionLifeCycleRuntimeMBean entity are as follows:

name

The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.

Type: string

Constraints: Read Only

state

The current overall state of the partition.

The value returned is the highest state for the partition on all servers where it is targeted. As a result, the overall partition state can change (for example, to RUNNING) as soon as the operation (such as start) has finished on one server, even while the operation is still in progress elsewhere.

If the partition is not currently targeted to any running server then its reported state is the desired state indicated by the most recent partition life cycle operation on the partition.

Type: string

Constraints: Read Only

subState

The current substate of the partition. It is mostly useful for knowing whether the partition is completely shutdown and the administrative resource groups are not running, in which case this method returns HALTED, or if the partition is in the SHUTDOWN. BOOTED state and the administrative resource groups are running, in which case this method returns BOOTED.
weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean

Exposes information about a long-running task related to the partition life cycle.

The properties of the weblogic.management.runtime.PartitionLifeCycleTaskRuntimeMBean entity are as follows:

**completed**
This property indicates whether an asynchronous job has completed.

*Type: boolean*

**description**
A description of this task.

*Type: string*

**endTime**
This property indicates the time when the asynchronous job ended.

*Type: string/date*

**name**
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.

*Type: string*

**operation**
Returns the name of the current operation.

*Type: string*

**parentTask**
The task of which this task is a part.

A value of `null` indicates that this task is not a subtask.

*Type: weblogic.management.runtime.TaskRuntimeMBean reference*

**partitionName**
Returns the name of the partition being operated on.

*Type: string*
Constraints: Read Only

**progress**
The progress of this task.
**Type:** string
**Constraints:** Read Only

**running**
Indicates whether the task is still running.
**Type:** boolean
**Constraints:** Read Only

**serverName**
Returns the name of the server on which operation happens.
**Type:** string
**Constraints:** Read Only

**startTime**
This property indicates the time when an asynchronous job was started.
**Type:** string/date

**startTimeAsLong**
The time at which this task was started.
**Type:** integer/int64
**Constraints:** Read Only

**systemTask**
Indicates whether this task was initiated by the server versus a user.
**Type:** boolean
**Constraints:** Read Only

**taskError**
Returns an exception describing the error, if any, that occurred while performing this task.
**Type:** java.lang.Throwable
**Constraints:** Read Only

**taskStatus**
The status of this task.
**Type:** string
**Constraints:** Read Only

**type**
Returns the type of the MBean.
**Type:** string
**Constraints:** Read Only
weblogic.management.runtime.PartitionUserFileSystemManagerMBean

This class is used for partition user file system handling.

The properties of the
weblogic.management.runtime.PartitionUserFileSystemManagerMBean entity are as follows:

name
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.

Type: string
Constraints: Read Only

type
Returns the type of the MBean.

Type: string
Constraints: Read Only

weblogic.management.runtime.ResourceGroupLifeCycleRuntimeMBean

Provides methods that transition resource groups from one state to another. This class is instantiated once for each resource group but only on the Administration Server. Even so, you can use a resource group’s instance to change the state of the resource group on Managed Servers as well as on the Administration Server.

Detecting when Operations Complete

Methods representing operations on resource groups return a ResourceGroupLifeCycleTaskRuntimeMBean object. The caller should monitor the returned instance, invoking its ResourceGroupLifeCycleTaskRuntimeMBean#isRunning() method to detect when the operation has completed. In particular, the caller should not rely on the value returned from getState() to indicate when an operation on a resource group that is targeted to multiple servers has finished. That value will be accurate as described in detail on getState() but can change before the operation has finished.

Desired States

The system tracks the desired state for each resource group, which defaults to the state RUNNING. Whenever a lifecycle operation method is called the system not only performs the requested action on the resource group but also records the resulting state as the new desired state for the resource group.

The system uses the desired state for a resource group to bring the resource group to the correct state during server and partition start-up. The desired can also be used when the getState() method computes the overall state for the resource group.

The properties of the
weblogic.management.runtime.ResourceGroupLifeCycleRuntimeMBean entity are as follows:

name
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.

Type: string
Constraints: Read Only

state
The current overall state of the resource group.

The value returned is the highest state for the resource group on all servers where it is targeted. As a result, the overall resource group state can change (for example, to RUNNING) as soon as the operation (such as start) has finished on one server, even while the operation is still in progress elsewhere.

If the resource group is not currently targeted to any running server then its reported state is the desired state indicated by the most recent resource group life cycle operation on the resource group.

Type: string
Constraints: Read Only

type
Returns the type of the MBean.

Type: string
Constraints: Read Only

weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean

The properties of the weblogic.management.runtime.ResourceGroupLifeCycleTaskRuntimeMBean entity are as follows:

completed
This property indicates whether an asynchronous job has completed.

Type: boolean

description
A description of this task.

Type: string
Constraints: Read Only

endTime
This property indicates the time when the asynchronous job ended.

Type: string/date

name
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.

Type: string
Constraints: Read Only

operation
Returns the name of the current operation.

Type: string
Constraints: Read Only
**parentTask**
The task of which this task is a part.
A value of null indicates that this task is not a subtask.
**Type:** weblogic.management.runtime.TaskRuntimeMBean reference
**Constraints:** Read Only

**progress**
The progress of this task.
**Type:** string
**Constraints:** Read Only

**resourceGroupName**
Returns the name of the resource group being operated on.
**Type:** string
**Constraints:** Read Only

**running**
Indicates whether the task is still running.
**Type:** boolean
**Constraints:** Read Only

**serverName**
Returns the name of the server on which operation happens.
**Type:** string
**Constraints:** Read Only

**startTime**
This property indicates the time when an asynchronous job was started.
**Type:** string/date

**startTimeAsLong**
The time at which this task was started.
**Type:** integer/int64
**Constraints:** Read Only

**systemTask**
Indicates whether this task was initiated by the server versus a user.
**Type:** boolean
**Constraints:** Read Only

**taskError**
Returns an exception describing the error, if any, that occurred while performing this task.
**Type:** java.lang.Throwable
**Constraints:** Read Only

**taskStatus**
The status of this task.
weblogic.management.runtime.ServerRuntimeMBean

Provides methods for retrieving runtime information about a server instance and for transitioning a server from one state to another.

The properties of the weblogic.management.runtime.ServerRuntimeMBean entity are as follows:

**adminServerHost**
The address on which the Administration Server is listening for connections. For example, this might return the string: santiago

Type: string
Constraints: Read Only

**adminServerListenPort**
The port on which the Administration Server is listening for connections.

Type: integer/int32
Constraints: Read Only

**adminServerListenPortSecure**
Indicates whether the port that the server uses for administrative traffic is configured to use a secure protocol.

Type: boolean
Constraints: Read Only

**defaultURL**
The URL that clients use to connect to this server’s default network channel.

The returned value indicates the default protocol, listen address and listen port:

```
protocol://listen-address:listen-port
```

Note: The default protocol, listen address and listen port are persisted in the domain’s config.xml file, however when a server instance is started, command-line options can
override these persisted values. This `getDefaultURL` method returns the URL values that are currently being used, not necessarily the values that are specified in `config.xml`.

The returned URL will be consistent with dynamic channel updates.

**Type:** string  
**Constraints:** Read Only

**healthState**
The health state of the server as reported by the server’s self-health monitoring.

For example, the server can report if it is overloaded by too many requests, if it needs more memory resources, or if it will soon fail for other reasons.

**Type:** `weblogic.health.HealthState`  
**Constraints:** Read Only

**name**
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.

**Type:** string  
**Constraints:** Read Only

**overallHealthState**
Determine the overall health state of this server, taking into account the health of each of its subsystems.

**Type:** `weblogic.health.HealthState`  
**Constraints:** Read Only

**restartRequired**
Indicates whether the server must be restarted in order to activate configuration changes.

**Type:** boolean  
**Constraints:** Read Only

**state**
The current life cycle state of this server.

For example, a server can be in a RUNNING state in which it can receive and process requests or in an ADMIN state in which it can receive only administrative requests.

**Type:** string  
**Constraints:** Read Only

**stateVal**
Returns current state of the server as in integer. `weblogic.management.runtime.ServerStates` has more information about the available server states.

**Type:** `integer/int32`  
**Constraints:** Read Only

**type**
Returns the type of the MBean.
weblogic.management.runtime.TaskRuntimeMBean

Exposes monitoring information about an ongoing and potentially long-running administrative process. This should be taken to mean, at minimum, any OA&M operation involving I/O. Examples include starting and stopping servers, deploying and undeploying applications, or migrating services.

An MBean operation of this sort should fork a separate thread to perform the actual work and immediately return an instance of TaskRuntimeMBean to the caller. The caller can then use this to track the task's progress as desired. Users can also query for all instances of TaskRuntimeMBean to get a summary of both currently-running and recently-completed tasks.

Instance of TaskRuntimeMBean continue to exist in the MBeanServer after the completion of the work they describe. They will eventually either be explicitly deregistered by the user, or removed by a scavenger process which periodically purges instances of TaskRuntimeMBean that have been completed for some time.

The properties of the weblogic.management.runtime.TaskRuntimeMBean entity are as follows:

completed
This property indicates whether an asynchronous job has completed.
Type: boolean

description
A description of this task.
Type: string
Constraints: Read Only

date
This property indicates the time when the asynchronous job ended.
Type: string/date

name
The name of this configuration. WebLogic Server uses an MBean to implement and persist the configuration.
Type: string
Constraints: Read Only

parentTask
The task of which this task is a part.
A value of null indicates that this task is not a subtask.
Type: weblogic.management.runtime.TaskRuntimeMBean reference
weblogic.management.runtime.WorkflowTaskRuntimeMBean

Constraints: Read Only

**progress**
The progress of this task.
Type: string
Constraints: Read Only

**running**
Indicates whether the task is still running.
Type: boolean
Constraints: Read Only

**startTime**
This property indicates the time when an asynchronous job was started.
Type: string/date

**startTimeAsLong**
The time at which this task was started.
Type: integer/int64
Constraints: Read Only

**systemTask**
Indicates whether this task was initiated by the server versus a user.
Type: boolean
Constraints: Read Only

**taskError**
Returns an exception describing the error, if any, that occurred while performing this task.
Type: java.lang.Throwable
Constraints: Read Only

**taskStatus**
The status of this task.
Type: string
Constraints: Read Only

**type**
Returns the type of the MBean.
Type: string
Constraints: Read Only

**weblogic.management.runtime.WorkflowTaskRuntimeMBean**

TaskRuntimeMBean for any workflow (WorkflowProgress) executed using management orchestration framework.

The properties of the weblogic.management.runtime.WorkflowTaskRuntimeMBean entity are as follows:
**completed**
This property indicates whether an asynchronous job has completed.
*Type:* boolean

**endTime**
This property indicates the time when the asynchronous job ended.
*Type:* string/date

**progress**
This property indicates whether an asynchronous job has succeeded, failed, or is in progress.
*Type:* string

**startTime**
This property indicates the time when an asynchronous job was started.
*Type:* string/date

**type**
Returns the type of the MBean.
*Type:* string

*Constraints:* Read Only