

Node

Table of contents

1 Description.....	2
2 Node.....	2
3 EntityReference.....	2
4 Examples.....	2

1. Description

The Node data type represents an object of a Node base type.

2. Node

Attribute	Description	Required
name	The object name.	Yes
type	The object type name.	Yes
maprefuri	The maprefUri	Yes
description	The object description.	Yes
includeExistingResources	Boolean flag specifying whether to merge included entities with existing child resources.	No, defaults to true.
osfamily	The OS family name. e.g, Unix, Windows	Yes
osname	The OS name. e.g, Linux, MacOS	Yes
osversion	The OS version. e.g, 7.1	Yes
osarch	The OS architecture. e.g, i386, PowerPC	Yes
hostname	The system FQDN. eg., host1.widgets.com	Yes

3. EntityReference

Uses one or more [EntityReference](#) elements to represent child resources.

4. Examples

Shows Node data type used in the [object-update](#) Ant task. The properties - os.family, os.name, os.version, os.arch - are all properties normally defined by Ant.

```
<osfamily property="os.family"/>
```

```
<property name="maprefId"
  value="Node_${framework.node}"/>
<property name="node.mapref-uri"
  value="http://www.opendepo.org/DepOnt#${maprefId}"/>

<object-update depot="${context.depot}"
  resultproperty="result"
  comment="Updated node object from Node.Register">
  <node maprefUri="${node.mapref-uri}"
    name="${framework.node}"
    type="${entity.classname}"
    description="${entity.description}"
    osfamily="${os.family}"
    osname="${os.name}"
    osversion="${os.version}"
    osarch="${os.arch}"/>
</object-update>
<condition property="wasUpdated">
  <equals arg1="0" arg2="${result}"/>
</condition>
<fail unless="wasUpdated">Failed saving object.</fail>
```