

Register

Command Reference

Table of contents

1 Description.....	2
2 Usage.....	2

1. Description

Register the deployment as an object to the repository.

The Register command registers an object in the project model maintained in the server. The registration information will include the values specified by the depot, type, name as well as -basedir, -installroot and -node. The -node argument specifies what node this object should be a child dependency to.

If the -install flag is set, the [Install](#) command is also run.

Bootstrapping non-basetype objects

In cases where the object is not a base type, a second syntax style can be used.

```
ctl -p project -m Deployment -c Register --\
    -name name -type YourType \
    -basedir /path/to/basedir -installroot /path/to/installroot \
    -install
```

... or ...

```
ctl -p project -m Deployment -c Register --\
    -xml /path/to/object.xml -install
```

This alternate usage does not assume a module for your type has been installed yet and relies on the Deployment basetype module instead.

static: This command can be run outside of an object context.

2. Usage

```
ctl -m Deployment -c Register [-basedir <>] [-depot <>]
[-description <>] [-install] [-installrank <>] [-installroot
<>] [-name <>] [-node <${framework.node.name}>] [-type <>]
[-xml <>]
```

2.1. Options

Option	Description	Type	Default
basedir	<i>base directory</i>	string	
depot	<i>project depot name</i>	string	<code>\${context.depot}</code>
description	<i>object description</i>	string	
install	<i>after registration, run Install command</i>	boolean	

Register

installrank	<i>install rank</i>	string	
installroot	<i>install directory</i>	string	
name	<i>object name</i>	string	<code>\${context.name}</code>
node	<i>after registration, bind this object to the specified node</i>	string	<code>\${framework.node.name}</code>
type	<i>type name</i>	string	<code>\${context.type}</code>
xml	<i>metadata xml file</i>	string	