

## EnableActionFilter

Enables a filter for the actions.

### Syntax

```
ActionMapManager.EnableActionFilter( name, enable )
```

Parameter	Description
name	Filter name.
enable	True to enable the filter, false otherwise.

## EnableActionMap

Enables an action map.

### Syntax

```
ActionMapManager.EnableActionMap( name, enable )
```

Parameter	Description
name	Action map name.
enable	True to enable the filter, false otherwise.

## EnableActionMapManager

Enables/Disables ActionMapManager.

### Syntax

```
ActionMapManager.EnableActionMapManager( enable, resetStateOnDisable )
```

Parameter	Description
enable	Enables/Disables ActionMapManager.
resetStateOnDisable	Resets the different Action states when ActionMapManager gets disabled.

## LoadFromXML

Loads information from an XML file.

### Syntax

```
ActionMapManager.LoadFromXML( name )
```

Parameter	Description
name	XML file name.

## InitActionMaps

Initializes the action maps and filters found in given file

#### Syntax

```
ActionMapManager.InitActionMaps( path )
```

Parameter	Description
path	XML file path.

## SetDefaultActionEntity

Sets the new default entity.

#### Syntax

```
ActionMapManager.SetDefaultActionEntity( id, updateAll )
```

Parameter	Description
id	EntityId of the new default action entity.
updateAll	Updates all existing action map assignments.

## GetDefaultActionEntity

Gets the currently set default action entity.

#### Syntax

```
ActionMapManager.GetDefaultActionEntity()
```

## LoadControllerLayoutFile

Loads the given controller layout into the action manager.

#### Syntax

```
ActionMapManager.LoadControllerLayoutFile( layoutName )
```

Parameter	Description
layoutName	layout name.

## IsFilterEnabled

Checks if a filter is currently enabled.

#### Syntax

```
ActionMapManager.IsFilterEnabled( filterName )
```

Parameter	Description
-----------	-------------

filterName	filter name.
------------	--------------