AllJoyn™ Lighting Service Framework Controller Service 14.06 Interface Definition

January 5, 2015

Contents

[1 Introduction 5](#_Toc408386960)

[1.1 Purpose 5](#_Toc408386961)

[1.2 Scope 5](#_Toc408386962)

[1.3 References 5](#_Toc408386963)

[1.4 Acronyms and terms 5](#_Toc408386964)

[2 Definition Overview 6](#_Toc408386965)

[2.1 Overview 6](#_Toc408386966)

[3 Error Handling 7](#_Toc408386967)

[4 Controller Service Interface 8](#_Toc408386968)

[4.1 Interface name 8](#_Toc408386969)

[4.2 Properties 8](#_Toc408386970)

[4.3 Methods 8](#_Toc408386971)

[4.3.1 LightingResetControllerService 8](#_Toc408386972)

[4.3.2 GetControllerServiceVersion 8](#_Toc408386973)

[4.4 Signals 9](#_Toc408386974)

[5 Lamp Interface 10](#_Toc408386975)

[5.1 Interface name 10](#_Toc408386976)

[5.2 Properties 10](#_Toc408386977)

[5.3 Methods 10](#_Toc408386978)

[5.3.1 GetAllLampIDs 10](#_Toc408386979)

[5.3.2 GetLampSupportedLanguages 10](#_Toc408386980)

[5.3.3 GetLampManufacturer 11](#_Toc408386981)

[5.3.4 GetLampName 11](#_Toc408386982)

[5.3.5 SetLampName 12](#_Toc408386983)

[5.3.6 GetLampDetails 12](#_Toc408386984)

[5.3.7 GetLampParameters 12](#_Toc408386985)

[5.3.8 GetLampParametersField 13](#_Toc408386986)

[5.3.9 GetLampState 13](#_Toc408386987)

[5.3.10 GetLampStateField 14](#_Toc408386988)

[5.3.11 TransitionLampState 14](#_Toc408386989)

[5.3.12 PulseLampWithState 14](#_Toc408386990)

[5.3.13 PulseLampWithPreset 15](#_Toc408386991)

[5.3.14 TransitionLampStateToPreset 16](#_Toc408386992)

[5.3.15 TransitionLampStateField 16](#_Toc408386993)

[5.3.16 ResetLampState 17](#_Toc408386994)

[5.3.17 ResetLampStateField 17](#_Toc408386995)

[5.3.18 GetLampFaults 17](#_Toc408386996)

[5.3.19 ClearLampFaults 18](#_Toc408386997)

[5.3.20 GetLampServiceVersion 18](#_Toc408386998)

[5.4 Signals 18](#_Toc408386999)

[6 Lamp Group Interface 20](#_Toc408387000)

[6.1 Interface name 20](#_Toc408387001)

[6.2 Properties 20](#_Toc408387002)

[6.3 Methods 20](#_Toc408387003)

[6.3.1 GetAllLampGroupIDs 20](#_Toc408387004)

[6.3.2 GetLampGroupName 20](#_Toc408387005)

[6.3.3 SetLampGroupName 21](#_Toc408387006)

[6.3.4 CreateLampGroup 21](#_Toc408387007)

[6.3.5 UpdateLampGroup 22](#_Toc408387008)

[6.3.6 DeleteLampGroup 22](#_Toc408387009)

[6.3.7 GetLampGroup 23](#_Toc408387010)

[6.3.8 TransitionLampGroupState 23](#_Toc408387011)

[6.3.9 PulseLampGroupWithState 23](#_Toc408387012)

[6.3.10 PulseLampGroupWithPreset 24](#_Toc408387013)

[6.3.11 TransitionLampGroupStateToPreset 25](#_Toc408387014)

[6.3.12 TransitionLampGroupStateField 25](#_Toc408387015)

[6.3.13 ResetLampGroupState 26](#_Toc408387016)

[6.3.14 ResetLampGroupStateField 26](#_Toc408387017)

[6.4 Signals 26](#_Toc408387018)

[7 Preset Interface 28](#_Toc408387019)

[7.1 Interface name 28](#_Toc408387020)

[7.2 Properties 28](#_Toc408387021)

[7.3 Methods 28](#_Toc408387022)

[7.3.1 GetDefaultLampState 28](#_Toc408387023)

[7.3.2 GetDefaultLampState 28](#_Toc408387024)

[7.3.3 GetAllPresetIDs 29](#_Toc408387025)

[7.3.4 GetPresetName 29](#_Toc408387026)

[7.3.5 SetPresetName 29](#_Toc408387027)

[7.3.6 CreatePreset 30](#_Toc408387028)

[7.3.7 UpdatePreset 30](#_Toc408387029)

[7.3.8 DeletePreset 31](#_Toc408387030)

[7.3.9 GetPreset 31](#_Toc408387031)

[7.4 Signals 31](#_Toc408387032)

[8 Scene Interface 33](#_Toc408387033)

[8.1 Interface name 33](#_Toc408387034)

[8.2 Properties 33](#_Toc408387035)

[8.3 Methods 33](#_Toc408387036)

[8.3.1 GetAllSceneIDs 33](#_Toc408387037)

[8.3.2 GetSceneName 33](#_Toc408387038)

[8.3.3 SetSceneName 34](#_Toc408387039)

[8.3.4 CreateScene 34](#_Toc408387040)

[8.3.5 UpdateScene 35](#_Toc408387041)

[8.3.6 DeleteScene 35](#_Toc408387042)

[8.3.7 GetScene 36](#_Toc408387043)

[8.3.8 ApplyScene 36](#_Toc408387044)

[8.4 Signals 37](#_Toc408387045)

[9 MasterScene Interface 38](#_Toc408387046)

[9.1 Interface name 38](#_Toc408387047)

[9.2 Properties 38](#_Toc408387048)

[9.3 Methods 38](#_Toc408387049)

[9.3.1 GetAllMasterSceneIDs 38](#_Toc408387050)

[9.3.2 GetSceneName 38](#_Toc408387051)

[9.3.3 SetMasterSceneName 39](#_Toc408387052)

[9.3.4 CreateMasterScene 39](#_Toc408387053)

[9.3.5 UpdateMasterScene 40](#_Toc408387054)

[9.3.6 DeleteMasterScene 40](#_Toc408387055)

[9.3.7 GetMasterScene 41](#_Toc408387056)

[9.3.8 ApplyMasterScene 41](#_Toc408387057)

[9.4 Signals 41](#_Toc408387058)

[10 Leader Election and State Sync Interface 43](#_Toc408387059)

[10.1 Interface name 43](#_Toc408387060)

[10.2 Properties 43](#_Toc408387061)

[10.3 Methods 43](#_Toc408387062)

[10.3.1 GetChecksumAndModificationTimestamp 43](#_Toc408387063)

[10.3.2 GetBlob 43](#_Toc408387064)

[10.3.3 Overthrow 44](#_Toc408387065)

[10.4 Signals 44](#_Toc408387066)

# Introduction

## Purpose

This document describes the specification of the AllJoyn™ Lighting Service Framework Controller Service interface. An application uses this interface to control multiple lamps and interact with the Lighting SDK.

## Scope

This document is intended for software engineers for AllJoyn lighting applications. It assumes familiarity with the AllJoyn SDK.

## References

The following reference documents are found on the AllSeen Alliance web site’s Doc and Downloads section.

* AllJoyn ™ Framework Tutorial
* AllJoyn™ Data Type Signature
* AllJoyn™ Lighting Service Framework Lamp Service Interface Definition

## Acronyms and terms

|  |  |
| --- | --- |
| Term | Definition |
| AllJoyn™ device | A device that supports the AllJoyn framework and can connect to a personal network. |
| Controller | An AllJoyn™ device that implements the collection of Controller Service interfaces. It can reside in a router, gateway, smartphone, tablet, etc. |
| Lamp service | Client interface implemented in an AllJoyn™ enabled lamp. |
| Lighting service framework | AllJoyn™ framework for building a complete lighting solution consisting of Controller Service and Lamp Service interfaces. |

# Definition Overview

## Overview

The Controller Service interface is implemented on a target device, more specifically it is a logical unit that can reside in a router, gateway, smartphone/tablet, or home automation controller. The interface allows a device to listen and connect to other AllJoyn™ lighting devices on the network, such as Lamp service. Once connected the controller can perform various operations on the Lamp service. For example, toggle lamp power state, change lamp color and brightness, and retrieve lamp specific details and attributes.

The controller service acts as a bridge between lamps and developer built AllJoyn™ applications. Such applications have the ability to connect directly to the controller service and interact with all the lighting devices connected to the controller itself.

# Error Handling

The method calls in the controller service interface return response codes in order to indicate whether the call resulted in an error or was successful.

Table 1. Controller service response codes

|  |  |
| --- | --- |
| Response Code | Description |
| LSF\_OK | Successful status |
| LSF\_ERR\_NULL | Unexpected NULL pointer |
| LSF\_ERR\_UNEXPECTED | An operation was unexpected at this time |
| LSF\_ERR\_INVALID | A value was invalid |
| LSF\_ERR\_UNKNOWN | An unknown value |
| LSF\_ERR\_FAILURE | A failure has occurs |
| LSF\_ERR\_BUSY | An operation failed and should be retired later |
| LSF\_ERR\_REJECTED | The request was rejected |
| LSF\_ERR\_RANGE | Value provided was out of range |
| LSF\_ERR\_INVALID\_FIELD | Invalid param/state field |
| LSF\_ERR\_MESSAGE | Invalid message |
| LSF\_ERR\_INVALID\_ARGS | The arguments were invalid |
| LSF\_ERR\_EMPTY\_NAME | The name is empty |
| LSF\_ERR\_RESOURCES | Not enough resources |
| LSF\_ERR\_REPLY\_WITH\_INVALID\_ARGS | The reply received for a message has invalid arguments |
| LSF\_ERR\_PARTIAL | The requested operation was only partially successful |
| LSF\_ERR\_NOT\_FOUND | The entity of interest was not found |
| LSF\_ERR\_NO\_SLOT | There is no slot for new entry |
| LSF\_ERR\_DEPENDENCY | There is a dependency of the entity for which a deleted request was received. |
| LSF\_ERR\_CODE\_LAST | The last LSF response code |

# Controller Service Interface

## Interface name

|  |  |  |  |
| --- | --- | --- | --- |
| Interface name | Version | Secured | Object path |
| org.allseen.LSF.ControllerService | 1 | no | /org/allseen/LSF/ControllerService |

## Properties

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property name | Signature | List of Values | Writable | Description |
| Version | u | Positive integers | No | Interface version number. |

## Methods

### LightingResetControllerService

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |

**Description**

Reset the state of the Lighting Controller Service. Clears any defined presets, lamp groups, and scenes.

### GetControllerServiceVersion

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| version | Yes | u | Controller service version number. |

**Description**

Gets the version of the Controller Service.

## Signals

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signal name | Parameter name | | | Sessionless | Description |
|  | Name | Signature | Mandatory |  |  |
| ControllerServiceLightingReset |  |  |  | Yes | Notify any listener that the controller service has been reset. |

# Lamp Interface

## Interface name

|  |  |  |  |
| --- | --- | --- | --- |
| Interface name | Version | Secured | Object path |
| org.allseen.LSF.ControllerService.Lamp | 1 | no | /org/allseen/LSF/ControllerService |

## Properties

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property name | Signature | List of Values | Writable | Description |
| Version | u | Positive integers | No | Interface version number. |

## Methods

### GetAllLampIDs

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampIDs | Yes | as | Array of lamp IDs where each ID represents a lamp connected to the controller. |

**Description**

Get the Lamp IDs of all lamps connected to the Controller Service.

### GetLampSupportedLanguages

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| supportedLanguages | Yes | as | Array of languages supported by the lamp. |

**Description**

Gets the supported languages for the specified lamp.

### GetLampManufacturer

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| language | Yes | s | The language of the returned result. Must be a language supported by the lamp. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| language | Yes | s | The language of the returned manufacturer result. |
| manufacturer | Yes | s | Name of the manufacturer. |

**Description**

Gets the manufacturer name for the specified lamp.

### GetLampName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| language | Yes | s | The language of the returned result. Must be a language supported by the lamp. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| language | Yes | s | The language of the returned manufacturer result. |
| lampName | Yes | s | Name of the lamp. |

**Description**

Gets the name for the specified lamp.

### SetLampName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| lampName | Yes | s | The new name of the lamp. |
| language | Yes | s | The language in which the lamp name is inputted. Must be a language supported by the lamp. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| language | Yes | s | The language of the returned manufacturer result. |

**Description**

Sets the name for the specified lamp.

### GetLampDetails

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| lampDetails | Yes | a{sv} | Map containing details of the lamp. |

**Description**

Gets the details for the specified lamp.

### GetLampParameters

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| lampParameters | Yes | a{sv} | Map containing the parameters values of the lamp. |

**Description**

Gets all the parameter of the lamp.

### GetLampParametersField

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| lampParameterFieldName | Yes | s | The field name of the lamp parameter to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp |
| lampParameterFieldName | Yes | s | The field name of the returned lamp parameter. |
| lampParameterFieldValue | Yes | s | The value of the queried parameter field. |

**Description**

Gets the value of a specific parameter for the lamp.

### GetLampState

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| lampState | Yes | a{sv} | Map containing the current state values of the lamp. |

**Description**

Gets all the state values of the lamp.

### GetLampStateField

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| lampStateFieldName | Yes | s | The field name of the lamp state to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| lampStateFieldName | Yes | s | The field name of the returned lamp state key. |
| lampStateFieldValue | Yes | s | The value of the queried state field. |

**Description**

Gets the value of a specific state field for the lamp.

### TransitionLampState

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| lampState | Yes | a{sv} | The lamp state to transition to. |
| transitionPeriod | Yes | u | Time in ms for the transition to occur over. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |

**Description**

Change the specified lamp’s state via transition.

### PulseLampWithState

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| fromLampState | Yes | a{sv} | The “fromState” or the first lamp state of the pulse effect. |
| toLampState | Yes | a{sv} | The “toState” or the second lamp state of the pulse effect. |
| period | Yes | u | The period (in ms) of a single pulse. That is the elapsed time of one fromLampState and one toLampState. |
| duration | Yes | u | The time (in ms) spent in the each lamp state for a single pulse. |
| numPulses | Yes | u | The total number of pulses for the effect. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |

**Description**

Perform a pulse effect on the specified lamp using lamp states.

### PulseLampWithPreset

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| fromPresetID | Yes | u | The preset in which the pulse effect starts. |
| toPresetID | Yes | u | The preset in which the pulse effect ends. |
| period | Yes | u | The period (in ms) of a single pulse. That is the elapsed time of one fromLampState and one toLampState. |
| duration | Yes | u | The time (in ms) spent in the each lamp state for a single pulse. |
| numPulses | Yes | u | The total number of pulses for the effect. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |

**Description**

Perform a pulse effect on the lamp using lamp presets.

### TransitionLampStateToPreset

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| presetID | Yes | u | The preset ID for the lamp to transition to. |
| transitionPeriod | Yes | u | Time in ms for the transition to occur over. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |

**Description**

Change the specified lamp’s state to a preset via transition.

### TransitionLampStateField

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| lampStateFieldName | Yes | s | The field name key of the lamp state that will be transitioned. |
| lampStateFieldValue | Yes | s | The new value to be set for the state field key. |
| transitionPeriod | Yes | u | Time in ms for the transition to occur over. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| lampStateFieldName | Yes | s | The name of transitioned lamp state field. |

**Description**

Change a specific state field of a lamp via transition.

### ResetLampState

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |

**Description**

Reset the entire lamp state of the specified lamp.

### ResetLampStateField

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| lampStateFieldName | Yes | s | The field name of the lamp state to reset. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| lampStateFieldName | Yes | s | The name of the state field that was reset. |

**Description**

Reset a particular state field of the specified lamp.

### GetLampFaults

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| lampFaults | Yes | au | Array of fault codes for the lamp. |

**Description**

Gets the fault codes for the specified lamp.

### ClearLampFaults

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |
| lampFault | Yes | u | The lamp fault to clear. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| lampFault | Yes | u | The fault code that was cleared. |

**Description**

Clears the fault code for the specified lamp.

### GetLampServiceVersion

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampID | Yes | s | The ID of the lamp to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp. |
| lampServiceVersion | Yes | u | The version number of the lamp service interface. |

**Description**

Gets the version number of the Lamp Service running of the lamp.

## Signals

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signal name | Parameter name | | | Sessionless | Description |
|  | Name | Signature | Mandatory |  |  |
| LampNameChanged | lampID | s | Yes | Yes | Notify listener that a lamp’s name has changed. |
| lampName | s | Yes |
| LampStateChanged | lampID | s | Yes | Yes | Notify listener that a lamp’s state has changed. |
| lampName | s | Yes |
| LampsFound | lampID | s | Yes | Yes | Notify listener that a new lamp was found. |
| LampsLost | lampIDs | as | Yes | Yes | Notify listener that lamps lost connection from the controller. |

# Lamp Group Interface

## Interface name

|  |  |  |  |
| --- | --- | --- | --- |
| Interface name | Version | Secured | Object path |
| org.allseen.LSF.ControllerService.LampGroup | 1 | no | /org/allseen/LSF/ControllerService |

## Properties

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property name | Signature | List of Values | Writable | Description |
| Version | u | Positive integers | No | Interface version number. |

## Methods

### GetAllLampGroupIDs

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupIDs | Yes | as | Array of lamp group IDs where each ID represents a lamp connected to the controller. |

**Description**

Get the lamp group IDs of all lamps connected to the Controller Service.

### GetLampGroupName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to query. |
| language | Yes | s | The language of the returned result. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the queried lamp group. |
| language | Yes | s | The language of the returned result. |
| lampGroupName | Yes | s | Name of the lamp group. |

**Description**

Gets the name for the specified lamp group.

### SetLampGroupName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to query. |
| lampName | Yes | s | The new name of the lamp group. |
| language | Yes | s | The language in which the lamp group name is inputted. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampID | Yes | s | The ID of the queried lamp group. |
| language | Yes | s | The language of the returned result. |

**Description**

Sets the name for the specified lamp group.

### CreateLampGroup

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampIDs | Yes | as | Array of lamp IDs to be included in the new group. |
| lampGroupIDs | Yes | as | Array of lamp group IDs to be included in the new group. |
| lampGroupName | Yes | s | The new name of the lamp group. |
| language | Yes | s | The language in which the lamp group name is inputted. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the newly created lamp group. |

**Description**

Creates a new lamp group on the controller.

### UpdateLampGroup

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to update. |
| lampIDs | Yes | as | Array of lamp IDs to be included in the updated group. |
| lampGroupIDs | Yes | as | Array of lamp group IDs to be included in the updated group. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the updated lamp group. |

**Description**

Updates a lamp group already existing on the controller.

### DeleteLampGroup

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to delete. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the deleted lamp group. |

**Description**

Deletes a lamp group that exists on the controller.

### GetLampGroup

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the updated lamp group. |
| lampID | Yes | as | The IDs of the lamps contained in the group. |
| lampGroupIDs | Yes | as | The IDs of the lamp groups contained within the group. |

**Description**

Get a lamp group which resides on the controller.

### TransitionLampGroupState

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to query. |
| lampState | Yes | a{sv} | The lamp state to transition to. |
| transitionPeriod | Yes | u | Time in ms for the transition to occur over. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the queried lamp group. |

**Description**

Change the specified lamp group’s state via transition.

### PulseLampGroupWithState

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to query. |
| fromLampState | Yes | a{sv} | The “fromState” or the first lamp state of the pulse effect. |
| toLampState | Yes | a{sv} | The “toState” or the second lamp state of the pulse effect. |
| period | Yes | u | The period (in ms) of a single pulse. That is the elapsed time of one fromLampState and one toLampState. |
| duration | Yes | u | The time (in ms) spent in the each lamp state for a single pulse. |
| numPulses | Yes | u | The total number of pulses for the effect. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampIGroupID | Yes | s | The ID of the queried lamp group. |

**Description**

Perform a pulse effect on the specified lamp group using lamp states.

### PulseLampGroupWithPreset

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to query. |
| fromPresetID | Yes | u | The preset in which the pulse effect starts. |
| toPresetID | Yes | u | The preset in which the pulse effect ends. |
| period | Yes | u | The period (in ms) of a single pulse. That is the elapsed time of one fromLampState and one toLampState. |
| duration | Yes | u | The time (in ms) spent in the each lamp state for a single pulse. |
| numPulses | Yes | u | The total number of pulses for the effect. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the queried lamp group. |

**Description**

Perform a pulse effect on the lamp group using lamp presets.

### TransitionLampGroupStateToPreset

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to query. |
| presetID | Yes | u | The preset ID for the lamp to transition to. |
| transitionPeriod | Yes | u | Time in ms for the transition to occur over. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the queried lamp group. |

**Description**

Change the specified lamp group’s state to a preset via transition.

### TransitionLampGroupStateField

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to query. |
| lampGroupStateFieldName | Yes | s | The field name key of the lamp groupstate that will be transitioned. |
| lampGroupStateFieldValue | Yes | s | The new value to be set for the state field key. |
| transitionPeriod | Yes | u | Time in ms for the transition to occur over. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the queried lamp group. |
| lampGroupStateFieldName | Yes | s | The name of transitioned lamp group state field. |

**Description**

Change a specific state field of a lamp group via transition.

### ResetLampGroupState

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The lamp group ID of the queried lamp. |

**Description**

Reset the entire lamp group state of the specified lamp.

### ResetLampGroupStateField

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampGroupID | Yes | s | The ID of the lamp group to query. |
| lampGroupStateFieldName | Yes | s | The field name of the lamp group state to reset. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampGroupID | Yes | s | The ID of the queried lamp group. |
| lampGroupStateFieldName | Yes | s | The name of the state field that was reset. |

**Description**

Reset a particular state field of the specified lamp group.

## Signals

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signal name | Parameter name | | | Sessionless | Description |
|  | Name | Signature | Mandatory |  |  |
| LampGroupsNameChanged | lampGroupsIDs | as | Yes | Yes | Notify listener that a lamp group’s name has changed. |
| LampGroupsCreated | lampGroupsIDs | as | Yes | Yes | Notify listener that a new lamp group is created. |
| LampGroupsUpdated | lampGroupsIDs | as | Yes | Yes | Notify listener that a lamp group was updated. |
| LampGroupsDeleted | lampGroupsIDs | as | Yes | Yes | Notify listener that a lamp group was deleted. |

# Preset Interface

## Interface name

|  |  |  |  |
| --- | --- | --- | --- |
| Interface name | Version | Secured | Object path |
| org.allseen.LSF.ControllerService.Preset | 1 | no | /org/allseen/LSF/ControllerService |

## Properties

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property name | Signature | List of Values | Writable | Description |
| Version | u | Positive integers | No | Interface version number |

## Methods

### GetDefaultLampState

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampState | Yes | a{sv} | Map containing the default lamp state. |

**Description**

Gets the default lamp state.

### GetDefaultLampState

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampState | Yes | a{sv} | Map containing the new default lamp state. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| lampState | Yes | a{sv} | Map containing the default lamp state. |

**Description**

Sets the default lamp state.

### GetAllPresetIDs

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| presetIDs | Yes | as | Array of preset IDs. |

**Description**

Get all IDs of presets which reside on the controller.

### GetPresetName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| presetID | Yes | s | The ID of the preset to query. |
| language | Yes | s | The language of the returned result. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| presetID | Yes | s | The ID of the queried preset. |
| language | Yes | s | The language of the returned result. |
| presetName | Yes | s | Name of the preset. |

**Description**

Gets the name for the specified preset.

### SetPresetName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| presetID | Yes | s | The ID of the preset to query. |
| presetName | Yes | s | The new name of the preset. |
| language | Yes | s | The language in which the preset name is inputted. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| presetID | Yes | s | The ID of the queried preset. |
| language | Yes | s | The language of the returned result. |

**Description**

Sets the name for the specified preset.

### CreatePreset

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| lampState | Yes | a{sv} | The lamp state to represent the preset. |
| presetName | Yes | s | The name of the new preset. |
| language | Yes | s | The language in which the preset name is inputted. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| presetID | Yes | s | The ID of the newly created preset. |

**Description**

Creates a new preset on the controller.

### UpdatePreset

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| presetID | Yes | s | The ID of the preset to update. |
| lampState | Yes | a{sv} | The lamp state to represent the preset. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| presetID | Yes | s | The ID of the updated preset. |

**Description**

Updates an existing preset on the controller.

### DeletePreset

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| presetID | Yes | s | The ID of the preset to delete. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| presetID | Yes | s | The ID of the deleted preset. |

**Description**

Deletes an existing preset on the controller.

### GetPreset

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| presetID | Yes | as | The ID of the preset to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| presetID | Yes | s | The ID of the queried preset. |
| lampState | Yes | a{sv} | The lamp state representing the preset. |

**Description**

Fetches a preset that exists on the controller.

## Signals

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signal name | Parameter name | | | Sessionless | Description |
|  | Name | Signature | Mandatory |  |  |
| DefaultLampStateChanged |  |  |  | Yes | Notify listener that the default lamp state has changed. |
| PresetsNameChanged | presetIDs | as | Yes | Yes | Notify listener that a presets name has changed. |
| PresetsCreated | presetIDs | as | Yes | Yes | Notify listener that a new preset has been created. |
| PresetsUpdated | presetIDs | as | Yes | Yes | Notify listener that a preset has been updated. |
| PresetsUpdated | presetIDs | as | Yes | Yes | Notify listener that a preset has been deleted. |

# Scene Interface

## Interface name

|  |  |  |  |
| --- | --- | --- | --- |
| Interface name | Version | Secured | Object path |
| org.allseen.LSF.ControllerService.Scene | 1 | no | /org/allseen/LSF/ControllerService |

## Properties

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property name | Signature | List of Values | Writable | Description |
| Version | u | Positive integers | No | Interface version number. |

## Methods

### GetAllSceneIDs

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| sceneIDs | Yes | as | Array of scene IDs. |

**Description**

Get all IDs of scenes which reside on the controller.

### GetSceneName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| sceneID | Yes | s | The ID of the scene to query. |
| language | Yes | s | The language of the returned result. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| sceneID | Yes | s | The ID of the queried scene. |
| language | Yes | s | The language of the returned result. |
| sceneName | Yes | s | Name of the scene. |

**Description**

Gets the name for the specified scene.

### SetSceneName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| sceneID | Yes | s | The ID of the scene to query. |
| sceneName | Yes | s | The new name of the scene. |
| language | Yes | s | The language in which the scene name is inputted. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| sceneID | Yes | s | The ID of the queried scene. |
| language | Yes | s | The language of the returned result. |

**Description**

Sets the name for the specified scene.

### CreateScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| transitionlampsLampGroupsToState | Yes | a(asasa{sv}u) | Collection of state transition effects to be included in the scene. |
| transitionlampsLampGroupsToPreset | Yes | a(asassu) | Collection of preset transition effects to be included in the scene. |
| pulselampsLampGroupsWithState | Yes | a(asasa{sv}a{sv}uuu) | Collection of state based pulse effects to be included in the scene. |
| pulselampsLampGroupsWithPreset | Yes | a(asasssuuu) | Collection of preset based pulse effects to be included in the scene. |
| sceneName | Yes | s | The name of the new scene. |
| language | Yes | s | The language in which the scene name is inputted. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| sceneID | Yes | s | The ID of the newly created scene. |

**Description**

Creates a new scene on the controller.

### UpdateScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| sceneID | Yes | s | The ID of the scene to update. |
| transitionlampsLampGroupsToState | Yes | a(asasa{sv}u) | Collection of state transition effects to be included in the scene. |
| transitionlampsLampGroupsToPreset | Yes | a(asassu) | Collection of preset transition effects to be included in the scene. |
| pulselampsLampGroupsWithState | Yes | a(asasa{sv}a{sv}uuu) | Collection of state based pulse effects to be included in the scene. |
| pulselampsLampGroupsWithPreset | Yes | a(asasssuuu) | Collection of preset based pulse effects to be included in the scene. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| sceneID | Yes | s | The ID of the updated scene. |

**Description**

Updates a scene existing on the controller.

### DeleteScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| sceneID | Yes | s | The ID of the scene to delete. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| sceneID | Yes | s | The ID of the deleted scene. |

**Description**

Deletes an existing scene on the controller.

### GetScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| sceneID | Yes | as | The ID of the scene to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| sceneID | Yes | s | The ID of the queried scene. |
| lampState | Yes | a{sv} | The lamp state representing the preset. |
| transitionlampsLampGroupsToState | Yes | a(asasa{sv}u) | Collection of state transition effects included in the scene. |
| transitionlampsLampGroupsToPreset | Yes | a(asassu) | Collection of preset transition effects included in the scene. |
| pulselampsLampGroupsWithState | Yes | a(asasa{sv}a{sv}uuu) | Collection of state based pulse effects included in the scene. |
| pulselampsLampGroupsWithPreset | Yes | a(asasssuuu) | Collection of preset based pulse effects included in the scene. |

**Description**

Fetches a scene that exists on the controller.

### ApplyScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| sceneID | Yes | s | The ID of the scene to apply. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| sceneID | Yes | s | The ID of the applied scene. |

**Description**

Applies a scene to the scene’s members.

## Signals

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signal name | Parameter name | | | Sessionless | Description |
|  | Name | Signature | Mandatory |  |  |
| ScenesNameChanged | sceneIDs | as | Yes | Yes | Notify listener that a scenes name has changed. |
| ScenesCreated | sceneIDs | as | Yes | Yes | Notify listener that a new scene has been created. |
| ScenesUpdated | sceneIDs | as | Yes | Yes | Notify listener that a scene has been updated. |
| ScenesDeleted | sceneIDs | as | Yes | Yes | Notify listener that a scene has been deleted. |
| ScenesApplied | sceneIDs | as | Yes | Yes | Notify listener that a scene has been applied. |

# MasterScene Interface

## Interface name

|  |  |  |  |
| --- | --- | --- | --- |
| Interface name | Version | Secured | Object path |
| org.allseen.LSF.ControllerService.MasterScene | 1 | no | /org/allseen/LSF/ControllerService |

## Properties

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property name | Signature | List of Values | Writable | Description |
| Version | u | Positive integers | No | Interface version number. |

## Methods

### GetAllMasterSceneIDs

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| masterSceneIDs | Yes | as | Array of master scene IDs. |

**Description**

Get all IDs of master scenes which reside on the controller.

### GetSceneName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| masterSceneID | Yes | s | The ID of the master scene to query. |
| language | Yes | s | The language of the returned result. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| masterSceneID | Yes | s | The ID of the queried master scene. |
| language | Yes | s | The language of the returned result. |
| masteSceneName | Yes | s | Name of the master scene. |

**Description**

Gets the name for the specified master scene.

### SetMasterSceneName

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| masterSceneID | Yes | s | The ID of the master scene to query. |
| masterSceneName | Yes | s | The new name of the master scene. |
| language | Yes | s | The language in which the master scene name is inputted. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| masterSceneID | Yes | s | The ID of the queried master scene. |
| language | Yes | s | The language of the returned result. |

**Description**

Sets the name for the specified master scene.

### CreateMasterScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| scenes | Yes | as | Array of scenes to be contained in the master scene. |
| masterSceneName | Yes | s | The name of the new master scene. |
| language | Yes | s | The language in which the master scene name is inputted. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| masterSceneID | Yes | s | The ID of the newly created master scene. |

**Description**

Creates a new master scene on the controller.

### UpdateMasterScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| masterSceneID | Yes | s | The ID of the master scene to update. |
| scenes | Yes | as | The scenes to be included in the master scene. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| masterSceneID | Yes | s | The ID of the updated master scene. |

**Description**

Updates a master scene existing on the controller.

### DeleteMasterScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| masterSceneID | Yes | s | The ID of the masterScene to delete. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| masterSceneID | Yes | s | The ID of the master scene. |

**Description**

Deletes an existing master scene on the controller.

### GetMasterScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| masterSceneID | Yes | s | The ID of the master scene to query. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| masterSceneID | Yes | s | The ID of the queried master scene. |
| scenes | Yes | as | The scenes that are part of the master scene. |

**Description**

Fetches a master scene that exists on the controller.

### ApplyMasterScene

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| masterSceneID | Yes | s | The ID of the master scene to apply. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| responseCode | Yes | u | The result code of the operation. |
| masterSceneID | Yes | s | The ID of the applied master scene. |

**Description**

Applies a master scene to the scene’s members.

## Signals

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signal name | Parameter name | | | Sessionless | Description |
|  | Name | Signature | Mandatory |  |  |
| MasterScenesNameChanged | sceneIDs | as | Yes | Yes | Notify listener that a master scenes name has changed. |
| MasterScenesCreated | sceneIDs | as | Yes | Yes | Notify listener that a new master scene has been created. |
| MasterScenesUpdated | sceneIDs | as | Yes | Yes | Notify listener that a master scene has been updated. |
| MasterScenesDeleted | sceneIDs | as | Yes | Yes | Notify listener that a master scene has been deleted. |
| MasterScenesApplied | sceneIDs | as | Yes | Yes | Notify listener that a master scene has been applied. |

# Leader Election and State Sync Interface

## Interface name

|  |  |  |  |
| --- | --- | --- | --- |
| Interface name | Version | Secured | Object path |
| org.allseen.LeaderElectionAndStateSync | 1 | no | /org/allseen/LeaderElectionAndStateSync |

## Properties

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property name | Signature | List of Values | Writable | Description |
| Version | u | Positive integers | No | Interface version number |

## Methods

### GetChecksumAndModificationTimestamp

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| checksumAndTimestamp | Yes | a(uut) | Object containing the checksum and timestamp for every blob. |

**Description**

Get the checksum and timestamp for each blob on the controller.

### GetBlob

**Inputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Signature | Description |
| blobType | Yes | u | The type of blob to fetch. |

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| blobType | Yes | u | The type of blob fetched. |
| blob | Yes | s | The payload of the blob |
| checksum | Yes | u | The checksum of the blob. |
| timestamp | Yes | t | The last modified timestamp of the blob. |

**Description**

Get the blob given a specific blob type.

### Overthrow

**Inputs**

None.

**Outputs**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | Mandatory | Return signature | Description |
| success | Yes | b | Boolean indicating success or failure of the overthrow action. |

**Description**

Attempts to overthrow other controllers as the leader.

## Signals

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Signal name | Parameter name | | | Sessionless | Description |
|  | Name | Signature | Mandatory |  |  |
| BlobChanged | blobType | u | Yes | Yes | Notify listener that a blob has changed on the controller. |
| blob | s | Yes |
| checksum | u | Yes |
| timestamp | t | Yes |