

# Host Javascript API

3.2.1.35

Generated by Doxygen 1.7.4

Fri Mar 16 2012 16:35:42



# Contents

<b>1</b>	<b>Host Javascript API</b>	<b>1</b>
1.1	Overview . . . . .	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class List . . . . .	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List . . . . .	5
<b>4</b>	<b>Class Documentation</b>	<b>7</b>
4.1	App Class Reference . . . . .	7
4.1.1	Detailed Description . . . . .	8
4.1.2	Member Function Documentation . . . . .	8
4.1.2.1	attachBundle . . . . .	8
4.1.2.2	authorizeUrl . . . . .	8
4.1.2.3	convertFromEntitizedString . . . . .	9
4.1.2.4	convertToEntitizedString . . . . .	9
4.1.2.5	error . . . . .	9
4.1.2.6	expandString . . . . .	9
4.1.2.7	getCommandLine . . . . .	9
4.1.2.8	getConfig . . . . .	10
4.1.2.9	getCredential . . . . .	10
4.1.2.10	getElevated . . . . .	10
4.1.2.11	getImplode . . . . .	10
4.1.2.12	getRemoveLocalStorage . . . . .	11
4.1.2.13	getRestart . . . . .	11

4.1.2.14	<a href="#">getRestartElevated</a>	11
4.1.2.15	<a href="#">getRestartPatchDirectory</a>	11
4.1.2.16	<a href="#">getRestartPatchFile</a>	11
4.1.2.17	<a href="#">hasBundle</a>	11
4.1.2.18	<a href="#">info</a>	12
4.1.2.19	<a href="#">loadInterop</a>	12
4.1.2.20	<a href="#">release</a>	12
4.1.2.21	<a href="#">setCommandLine</a>	12
4.1.2.22	<a href="#">setConfig</a>	12
4.1.2.23	<a href="#">setCredential</a>	12
4.1.2.24	<a href="#">setImplode</a>	13
4.1.2.25	<a href="#">setRemoveLocalStorage</a>	13
4.1.2.26	<a href="#">setRestart</a>	13
4.1.2.27	<a href="#">setRestartElevated</a>	13
4.1.2.28	<a href="#">setRestartPatchDirectory</a>	13
4.1.2.29	<a href="#">setRestartPatchFile</a>	14
4.1.2.30	<a href="#">warn</a>	14
4.1.3	<a href="#">Member Data Documentation</a>	14
4.1.3.1	<a href="#">instanceId</a>	14
4.1.3.2	<a href="#">onBundleAttached</a>	14
4.1.3.3	<a href="#">onInteropLoaded</a>	14
4.1.3.4	<a href="#">onWillAuthenticateWebGetTarget</a>	14
4.1.3.5	<a href="#">onWillConstructWebGetTarget</a>	15
4.1.3.6	<a href="#">onWillRestart</a>	15
4.2	<a href="#">Array Class Reference</a>	15
4.2.1	<a href="#">Detailed Description</a>	15
4.2.2	<a href="#">Member Function Documentation</a>	16
4.2.2.1	<a href="#">forEach</a>	16
4.2.2.2	<a href="#">forEachReverse</a>	16
4.2.2.3	<a href="#">indexOf</a>	16
4.2.2.4	<a href="#">removeElement</a>	16
4.2.2.5	<a href="#">removeIndex</a>	16
4.3	<a href="#">BundleLoader Class Reference</a>	17
4.3.1	<a href="#">Detailed Description</a>	17

4.3.2	Member Function Documentation	17
4.3.2.1	start	17
4.3.3	Member Data Documentation	17
4.3.3.1	name	17
4.3.3.2	onComplete	17
4.3.3.3	onProgress	18
4.3.3.4	url	18
4.4	Host Class Reference	18
4.4.1	Detailed Description	19
4.4.2	Member Function Documentation	19
4.4.2.1	addBundleAttachedEventHandler	19
4.4.2.2	addInteropLoadedEventHandler	19
4.4.2.3	decodeHex	19
4.4.2.4	downloadContent	19
4.4.2.5	elevate	20
4.4.2.6	getJSON	20
4.4.2.7	idFromString	20
4.4.2.8	init	20
4.4.2.9	loadBundles	20
4.4.2.10	loadCSS	21
4.4.2.11	loadJS	21
4.4.2.12	loadJSONArray	21
4.4.2.13	loadLanguage	21
4.4.2.14	release	22
4.4.2.15	removeBundleAttachedEventHandler	22
4.4.2.16	removeInteropLoadedEventHandler	22
4.4.2.17	urlToFileName	22
4.4.3	Member Data Documentation	22
4.4.3.1	app	22
4.4.3.2	hexTable	22
4.4.3.3	skinWindow	23
4.5	HttpDownload Class Reference	23
4.5.1	Detailed Description	23
4.5.2	Member Function Documentation	24

4.5.2.1	<a href="#">getConnectTimeout</a>	24
4.5.2.2	<a href="#">getExitCode</a>	24
4.5.2.3	<a href="#">getFileName</a>	24
4.5.2.4	<a href="#">getProxy</a>	24
4.5.2.5	<a href="#">getProxyRealm</a>	24
4.5.2.6	<a href="#">getRetryCount</a>	24
4.5.2.7	<a href="#">getUrl</a>	25
4.5.2.8	<a href="#">getUseCache</a>	25
4.5.2.9	<a href="#">getUseWriteCallback</a>	25
4.5.2.10	<a href="#">release</a>	25
4.5.2.11	<a href="#">setConnectTimeout</a>	25
4.5.2.12	<a href="#">setFileName</a>	25
4.5.2.13	<a href="#">setRetryCount</a>	26
4.5.2.14	<a href="#">setUrl</a>	26
4.5.2.15	<a href="#">setUseCache</a>	26
4.5.2.16	<a href="#">setUseWriteCallback</a>	26
4.5.2.17	<a href="#">start</a>	26
4.5.3	<a href="#">Member Data Documentation</a>	26
4.5.3.1	<a href="#">instanceId</a>	26
4.5.3.2	<a href="#">onComplete</a>	26
4.5.3.3	<a href="#">onDownloadRequired</a>	27
4.5.3.4	<a href="#">onProgress</a>	27
4.5.3.5	<a href="#">onStart</a>	27
4.5.3.6	<a href="#">onWrite</a>	27
4.6	<a href="#">OSError Class Reference</a>	27
4.6.1	<a href="#">Detailed Description</a>	28
4.6.2	<a href="#">Member Function Documentation</a>	28
4.6.2.1	<a href="#">isUnknown</a>	28
4.6.2.2	<a href="#">nameFromId</a>	28
4.7	<a href="#">SkinWindow Class Reference</a>	28
4.7.1	<a href="#">Detailed Description</a>	29
4.7.2	<a href="#">Member Function Documentation</a>	29
4.7.2.1	<a href="#">beginMove</a>	29
4.7.2.2	<a href="#">beginSize</a>	29

4.7.2.3	<a href="#">browseForFolder</a>	30
4.7.2.4	<a href="#">close</a>	30
4.7.2.5	<a href="#">getHeight</a>	30
4.7.2.6	<a href="#">getWidth</a>	30
4.7.2.7	<a href="#">getX</a>	30
4.7.2.8	<a href="#">getY</a>	31
4.7.2.9	<a href="#">hide</a>	31
4.7.2.10	<a href="#">load</a>	31
4.7.2.11	<a href="#">minimize</a>	31
4.7.2.12	<a href="#">prompt</a>	31
4.7.2.13	<a href="#">release</a>	32
4.7.2.14	<a href="#">restore</a>	32
4.7.2.15	<a href="#">setOrigin</a>	32
4.7.2.16	<a href="#">setSize</a>	32
4.7.2.17	<a href="#">setTitle</a>	32
4.7.2.18	<a href="#">show</a>	33
4.7.3	<a href="#">Member Data Documentation</a>	33
4.7.3.1	<a href="#">instanceId</a>	33
4.7.3.2	<a href="#">onDidClose</a>	33
4.7.3.3	<a href="#">onWillClose</a>	33
4.8	<a href="#">SkinWindowPromptType Class Reference</a>	33
4.8.1	<a href="#">Detailed Description</a>	33
4.9	<a href="#">String Class Reference</a>	34
4.9.1	<a href="#">Detailed Description</a>	34
4.9.2	<a href="#">Member Function Documentation</a>	34
4.9.2.1	<a href="#">replaceAll</a>	34
4.9.2.2	<a href="#">trimChars</a>	34
4.10	<a href="#">TextFileReader Class Reference</a>	34
4.10.1	<a href="#">Detailed Description</a>	35
4.10.2	<a href="#">Member Function Documentation</a>	35
4.10.2.1	<a href="#">getBufferSize</a>	35
4.10.2.2	<a href="#">getFileName</a>	35
4.10.2.3	<a href="#">release</a>	35
4.10.2.4	<a href="#">setBufferSize</a>	35

4.10.2.5	setFileName	36
4.10.2.6	start	36
4.10.3	Member Data Documentation	36
4.10.3.1	instanceId	36
4.10.3.2	onComplete	36
4.10.3.3	onProgress	36
4.10.3.4	onRead	36
4.10.3.5	onStart	37
4.11	TextFileWriter Class Reference	37
4.11.1	Detailed Description	37
4.11.2	Member Function Documentation	37
4.11.2.1	close	37
4.11.2.2	open	37
4.11.2.3	release	38
4.11.2.4	write	38
4.11.3	Member Data Documentation	38
4.11.3.1	instanceId	38
<b>5</b>	<b>File Documentation</b>	<b>39</b>
5.1	app.js File Reference	39
5.1.1	Detailed Description	39
5.1.2	Function Documentation	39
5.1.2.1	createApp	39
5.2	bundleloader.js File Reference	39
5.2.1	Detailed Description	40
5.2.2	Function Documentation	40
5.2.2.1	createBundleLoader	40
5.3	host.js File Reference	40
5.3.1	Detailed Description	40
5.3.2	Variable Documentation	40
5.3.2.1	host	40
5.4	httpdownload.js File Reference	41
5.4.1	Detailed Description	41
5.4.2	Function Documentation	41



5.4.2.1	<a href="#">createHttpDownload</a>	41
5.5	<a href="#">oserror.js File Reference</a>	41
5.5.1	<a href="#">Detailed Description</a>	41
5.5.2	<a href="#">Variable Documentation</a>	42
5.5.2.1	<a href="#">osError</a>	42
5.6	<a href="#">skinwindow.js File Reference</a>	42
5.6.1	<a href="#">Detailed Description</a>	42
5.6.2	<a href="#">Function Documentation</a>	42
5.6.2.1	<a href="#">createSkinWindow</a>	42
5.7	<a href="#">skinwindowprompt.js File Reference</a>	42
5.7.1	<a href="#">Detailed Description</a>	42
5.7.2	<a href="#">Variable Documentation</a>	43
5.7.2.1	<a href="#">skinWindowPrompt</a>	43
5.8	<a href="#">textfilereader.js File Reference</a>	43
5.8.1	<a href="#">Detailed Description</a>	43
5.8.2	<a href="#">Function Documentation</a>	43
5.8.2.1	<a href="#">createTextFileReader</a>	43
5.9	<a href="#">textfilewriter.js File Reference</a>	43
5.9.1	<a href="#">Detailed Description</a>	44
5.9.2	<a href="#">Function Documentation</a>	44
5.9.2.1	<a href="#">createTextFileWriter</a>	44



## Chapter 1

# Host Javascript API

### 1.1 Overview

The host javascript API allows users to integrate with the Solid State Networks host platform.

The usage of this API is restricted to entities which have signed a license agreement with Solid State Networks, Inc.

Licenses are valid for only one PRODUCT usage. Please see the license agreement for further details.



## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">App</a> (Solid State's Application Interface ) . . . . .	7
<a href="#">Array</a> ( <a href="#">Array</a> extension ) . . . . .	15
<a href="#">BundleLoader</a> (Helper class to load bundles from JSON file ) . . . . .	17
<a href="#">Host</a> ( <a href="#">Host</a> Top level container for the host ) . . . . .	18
<a href="#">HttpDownload</a> (HTTP downloader ) . . . . .	23
<a href="#">OSError</a> (Operating system helper functions ) . . . . .	27
<a href="#">SkinWindow</a> (Solid State's HTML Skinned Window ) . . . . .	28
<a href="#">SkinWindowPromptType</a> (Skin window prompt question and return type flags ) . . . . .	33
<a href="#">String</a> ( <a href="#">String</a> extension ) . . . . .	34
<a href="#">TextFileReader</a> (Text file reader ) . . . . .	34
<a href="#">TextFileWriter</a> (Text file writer ) . . . . .	37



## Chapter 3

# File Index

### 3.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">app.js</a> (File containing <a href="#">App</a> class and creation function ) . . . . .	39
<a href="#">bundleloader.js</a> (File containing <a href="#">BundleLoader</a> class and creation function ) . .	39
<a href="#">host.js</a> (File containing <a href="#">Host</a> class ) . . . . .	40
<a href="#">httpdownload.js</a> (File containing <a href="#">HttpDownload</a> class and creation function ) . .	41
<a href="#">oserror.js</a> (File containing operating system helper functions ) . . . . .	41
<a href="#">skinwindow.js</a> (File containing <a href="#">SkinWindow</a> class and creation function ) . . . .	42
<a href="#">skinwindowprompt.js</a> (File containing skin window prompt flags ) . . . . .	42
<a href="#">textfilereader.js</a> (File containing <a href="#">TextFileReader</a> class and creation function ) .	43
<a href="#">textfilewriter.js</a> (File containing <a href="#">TextFileWriter</a> class and creation function ) . . .	43





## Chapter 4

# Class Documentation

### 4.1 App Class Reference

Solid State's Application Interface.

#### Public Member Functions

- void [attachBundle](#) (string bundleName, string path)
- bool [authorizeUrl](#) (string url)
- string [convertFromEntitizedString](#) (string entitizedString)
- string [convertToEntitizedString](#) (string utf8string)
- void [error](#) (string message)
- string [expandString](#) (string expand)
- string [getCommandLine](#) ()
- string [getConfig](#) (string key, string defaultValue)
- string [getCredential](#) (int type, string [host](#), string realm)
- bool [getElevated](#) ()
- bool [getImplode](#) ()
- bool [getRemoveLocalStorage](#) ()
- bool [getRestart](#) ()
- bool [getRestartElevated](#) ()
- string [getRestartPatchDirectory](#) ()
- string [getRestartPatchFile](#) ()
- bool [hasBundle](#) (string bundleName)
- void [info](#) (string message)
- void [loadInterop](#) (string fileName)
- void [release](#) ()
- void [setCommandLine](#) (string value)
- void [setConfig](#) (string key, string value)
- void [setCredential](#) (int type, string [host](#), string realm, string username, string password)

- void [setImplode](#) (bool value)
- void [setRemoveLocalStorage](#) (bool value)
- void [setRestart](#) (bool value)
- void [setRestartElevated](#) (bool value)
- void [setRestartPatchDirectory](#) (string value)
- void [setRestartPatchFile](#) (string value)
- void [warn](#) (string message)

### Public Attributes

- string [instanceId](#)
- event [onBundleAttached](#)
- event [onInteropLoaded](#)
- event [onWillAuthenticateWebGetTarget](#)
- event [onWillConstructWebGetTarget](#)
- event [onWillRestart](#)

#### 4.1.1 Detailed Description

Solid State's Application Interface.

#### 4.1.2 Member Function Documentation

##### 4.1.2.1 void App.attachBundle ( string *bundleName*, string *path* )

This method attaches a bundle

##### Parameters

<i>bundleName</i>	Name of bundle to be checked
<i>path</i>	Local location of the bundle

##### 4.1.2.2 bool App.authorizeUrl ( string *url* )

This method checks if a url is in the app config whitelist

##### Parameters

<i>url</i>	Url to be checked
------------	-------------------

##### Returns

true if authorized, false otherwise

#### 4.1.2.3 string App.convertFromEntitizedString ( string *entitizedString* )

This method converts an HTML entitized string into an UTF-8 string

##### Parameters

<i>entitized-String</i>	Entitized string to be converted to UTF-8
-------------------------	---

##### Returns

utf8 string

#### 4.1.2.4 string App.convertToEntitizedString ( string *utf8string* )

This method converts a UTF-8 string into an HTML entitized string

##### Parameters

<i>utf8string</i>	UTF-8 string to be entitized
-------------------	------------------------------

##### Returns

entitized string

#### 4.1.2.5 void App.error ( string *message* )

Adds an error line to the host log file

##### Parameters

<i>message</i>	string to add to the log file
----------------	-------------------------------

#### 4.1.2.6 string App.expandString ( string *expand* )

This method expands the string with the current macros

##### Parameters

<i>expand</i>	Name of string to be expanded
---------------	-------------------------------

##### Returns

expanded string

#### 4.1.2.7 string App.getCommandLine ( )

gets the application command line

**Returns**

application command line

**4.1.2.8 string App.getConfig ( string key, string defaultValue )**

This method retrieves a config setting

**Parameters**

<i>key</i>	Name of config setting to be retrieved
<i>defaultValue</i>	Default value of key if nothing is found

**Returns**

value of key's setting

**4.1.2.9 string App.getCredential ( int type, string host, string realm )**

This method retrieves a credential

**Parameters**

<i>type</i>	Type of credential
<i>host</i>	<a href="#">Host</a> for the credential
<i>realm</i>	Realm for the credential

**Returns**

json object containing username and password

**4.1.2.10 bool App.getElevated ( )**

gets the host's current elevation status

**Returns**

host's current elevation status

**4.1.2.11 bool App.getImplode ( )**

gets the self delete on exit

**Returns**

self delete on exit

**4.1.2.12** `bool App.getRemoveLocalStorage ( )`

gets the flag value to remove the local storage directory on shutdown

**Returns**

flag value to remove the local storage directory on shutdown

**4.1.2.13** `bool App.getRestart ( )`

gets the flag value to restart the host

**Returns**

flag value to restart the host

**4.1.2.14** `bool App.getRestartElevated ( )`

gets the flag value to elevate the restart

**Returns**

flag value to elevate the restart

**4.1.2.15** `string App.getRestartPatchDirectory ( )`

gets the path to output directory for self update

**Returns**

path to output directory for self update

**4.1.2.16** `string App.getRestartPatchFile ( )`

gets the path to file for self update

**Returns**

path to file for self update

**4.1.2.17** `bool App.hasBundle ( string bundleName )`

This method checks if a bundle name has already been attached

**Parameters**

<i>bundleName</i>	Name of bundle to be checked
-------------------	------------------------------

**Returns**

true if attached, false otherwise

**4.1.2.18 void App.info ( string *message* )**

Adds an information line to the host log file

**Parameters**

<i>message</i>	string to add to the log file
----------------	-------------------------------

**4.1.2.19 void App.loadInterop ( string *fileName* )**

This method loads an interop library

**Parameters**

<i>fileName</i>	Name of interop library to be loaded
-----------------	--------------------------------------

**4.1.2.20 void App.release ( )**

releases the object

**4.1.2.21 void App.setCommandLine ( string *value* )**

sets the application command line

**Parameters**

<i>value</i>	application command line
--------------	--------------------------

**4.1.2.22 void App.setConfig ( string *key*, string *value* )**

This method sets a config setting

**Parameters**

<i>key</i>	Name of config setting to be retrieved
<i>value</i>	Value store in key parameter

**4.1.2.23 void App.setCredential ( int *type*, string *host*, string *realm*, string *username*, string *password* )**

This method sets a credential

**Parameters**

<i>type</i>	Type of credential
<i>host</i>	<a href="#">Host</a> for the credential
<i>realm</i>	Realm for the credential
<i>username</i>	Username for the credential
<i>password</i>	Password for the credential

**4.1.2.24 void App.setImplode ( bool *value* )**

sets the self delete on exit

**Parameters**

<i>value</i>	self delete on exit
--------------	---------------------

**4.1.2.25 void App.setRemoveLocalStorage ( bool *value* )**

sets the flag value to remove the local storage directory on shutdown

**Parameters**

<i>value</i>	flag value to remove the local storage directory on shutdown
--------------	--

**4.1.2.26 void App.setRestart ( bool *value* )**

sets the flag value to restart the host

**Parameters**

<i>value</i>	flag value to restart the host
--------------	--------------------------------

**4.1.2.27 void App.setRestartElevated ( bool *value* )**

sets the flag value to elevate the restart

**Parameters**

<i>value</i>	flag value to elevate the restart
--------------	-----------------------------------

**4.1.2.28 void App.setRestartPatchDirectory ( string *value* )**

sets the path to output directory for self update

**Parameters**

<i>value</i>	path to output directory for self update
--------------	--

#### 4.1.2.29 void App.setRestartPatchFile ( string *value* )

sets the path to file for self update

##### Parameters

<i>value</i>	path to file for self update
--------------	------------------------------

#### 4.1.2.30 void App.warn ( string *message* )

Adds a warning line to the host log file

##### Parameters

<i>message</i>	string to add to the log file
----------------	-------------------------------

### 4.1.3 Member Data Documentation

#### 4.1.3.1 string App.instanceId

Instance id used to bind proxy object to native object

#### 4.1.3.2 event App.onBundleAttached

Triggered when a bundle is attached

##### Parameters

<i>bundleName</i>	name of the bundle to attach
<i>path</i>	path of the bundle to attach
<i>successful</i>	true if the attach was successful, false otherwise

#### 4.1.3.3 event App.onInteropLoaded

Triggered when an interop library is loaded

##### Parameters

<i>fileName</i>	file name of the interop library
<i>successful</i>	true if the load was successful, false otherwise

#### 4.1.3.4 event App.onWillAuthenticateWebGetTarget

This event is triggered whenever a http authentication is required



**Parameters**

<i>type</i>	type of authentication
<i>host</i>	host for authentication
<i>realm</i>	realm for authentication
<i>username</i>	last username attempted
<i>password</i>	last password attempted

**Returns**

string: json object containing the username, password and cancel keys

**4.1.3.5 event `App.onWillConstructWebGetTarget`**

This event is triggered whenever a http get is attempted

**Parameters**

<i>url</i>	url for the http get
------------	----------------------

**Returns**

string: new url for the http get

**4.1.3.6 event `App.onWillRestart`**

Triggered when an restart is required

The documentation for this class was generated from the following file:

- [app.js](#)

## 4.2 Array Class Reference

[Array](#) extension.

**Public Member Functions**

- void [forEach](#) (function block, object context)
- void [forEachReverse](#) (function block, object context)
- int [indexOf](#) (object elt)
- void [removeElement](#) (object element)
- void [removeIndex](#) (int index)

**4.2.1 Detailed Description**

[Array](#) extension.

## 4.2.2 Member Function Documentation

### 4.2.2.1 void Array.forEach ( function *block*, object *context* )

array enumerator

#### Parameters

<i>block</i>	function to call during each object
<i>context</i>	block context

### 4.2.2.2 void Array.forEachReverse ( function *block*, object *context* )

array enumerator

#### Parameters

<i>block</i>	function to call during each object
<i>context</i>	block context

### 4.2.2.3 int Array.indexOf ( object *elt* )

gets the index of an object

#### Parameters

<i>elt</i>	object to get the array index of
<i>from</i>	optional starting point

#### Returns

index of object in array

### 4.2.2.4 void Array.removeElement ( object *element* )

removes an object

#### Parameters

<i>element</i>	object to remove
----------------	------------------

### 4.2.2.5 void Array.removeIndex ( int *index* )

removes an object at the given index

**Parameters**

<i>index</i>	index of object
--------------	-----------------

The documentation for this class was generated from the following file:

- jshelper.js

## 4.3 BundleLoader Class Reference

Helper class to load bundles from JSON file.

**Public Member Functions**

- void [start](#) ()

**Public Attributes**

- string [name](#)
- event [onComplete](#)
- event [onProgress](#)
- string [url](#)

### 4.3.1 Detailed Description

Helper class to load bundles from JSON file.

### 4.3.2 Member Function Documentation

#### 4.3.2.1 void BundleLoader.start ( )

starts the bundle loader

### 4.3.3 Member Data Documentation

#### 4.3.3.1 string BundleLoader.name

Name of the bundle in the {LocalStorage} directory

#### 4.3.3.2 event BundleLoader.onComplete

Triggered when the bundle loader is complete.

**Parameters**

<i>successful</i>	true if the download was successful, false otherwise.
-------------------	---

**4.3.3.3 event BundleLoader.onProgress**

Triggered when the bundle loader has made progress.

**Parameters**

<i>percent</i>	(-1.0: Still calculating) (0.0 to 1.0: Percent completed)
----------------	---

**4.3.3.4 string BundleLoader.url**

Url location of the bundle

The documentation for this class was generated from the following file:

- [bundleloader.js](#)

**4.4 Host Class Reference**

[Host](#) Top level container for the host.

**Public Member Functions**

- void [addBundleAttachedEventHandler](#) (event handler)
- void [addInteropLoadedEventHandler](#) (event handler)
- void [decodeHex](#) (string str)
- void [downloadContent](#) (string url, event progressCallback, event completeCallback)
- void [elevate](#) ()
- void [getJSON](#) (string url, event completeCallback)
- string [idFromString](#) (string str)
- void [init](#) (void [skinWindow](#))
- void [loadBundles](#) (string url, event statusCallback, event progressCallback, event completeCallback)
- void [loadCSS](#) (string url, event completeCallback)
- void [loadJS](#) (string url, event completeCallback)
- void [loadJSONArray](#) (array urls, event completeCallback)
- void [loadLanguage](#) (string url, event completeCallback)
- void [release](#) ()
- void [removeBundleAttachedEventHandler](#) (event handler)
- void [removeInteropLoadedEventHandler](#) (event handler)
- string [urlToFileName](#) (string url, string directory, string extension)

## Public Attributes

- [app](#) `app`
- [array](#) `hexTable`
- [SkinWindow](#) `skinWindow`

### 4.4.1 Detailed Description

[Host](#) Top level container for the host.

### 4.4.2 Member Function Documentation

#### 4.4.2.1 void Host.addBundleAttachedEventHandler ( event *handler* )

subscribe to the BundleAttached event

##### Parameters

<i>handler</i>	callback function for BundleAttached event
----------------	--

#### 4.4.2.2 void Host.addInteropLoadedEventHandler ( event *handler* )

subscribe to the InteropLoaded event

##### Parameters

<i>handler</i>	callback function for InteropLoaded event
----------------	---

#### 4.4.2.3 void Host.decodeHex ( string *str* )

decode a hex string

##### Parameters

<i>str</i>	string to decode
------------	------------------

#### 4.4.2.4 void Host.downloadContent ( string *url*, event *progressCallback*, event *completeCallback* )

dynamically retrieve content using httpDownload

##### Parameters

<i>url</i>	url of the content to retrieve
<i>progress-Callback</i>	called while downloading with percent downloading

<i>complete-Callback</i>	called when function is complete with successful parameter
--------------------------	--

#### 4.4.2.5 void Host.elevate ( )

elevates the host

#### 4.4.2.6 void Host.getJSON ( string url, event completeCallback )

gets a JSON object from a url

##### Parameters

<i>url</i>	url of json object
<i>complete-Callback</i>	called when function is complete with json object as parameter

#### 4.4.2.7 string Host.idFromString ( string str )

creates an id from a string

##### Parameters

<i>str</i>	string to create id from
------------	--------------------------

##### Returns

id created from string

#### 4.4.2.8 void Host.init ( void skinWindow )

initializes the host

#### 4.4.2.9 void Host.loadBundles ( string url, event statusCallback, event progressCallback, event completeCallback )

load bundles from url containing json file

##### Parameters

<i>url</i>	url of json object containing bundle information
<i>statusCallback</i>	called when function changes status with name and state parameters
<i>progress-Callback</i>	called when function changes progress with percent parameter

<i>complete-Callback</i>	called when function is complete with successful parameter
--------------------------	--

#### 4.4.2.10 void Host.loadCSS ( string url, event completeCallback )

dynamically load CSS files from url

##### Parameters

<i>url</i>	url of CSS
<i>complete-Callback</i>	called when function is complete with successful parameter

#### 4.4.2.11 void Host.loadJS ( string url, event completeCallback )

dynamically load javascript files from url

##### Parameters

<i>url</i>	url of javascript
<i>complete-Callback</i>	called when function is complete with successful parameter

#### 4.4.2.12 void Host.loadJSONArray ( array urls, event completeCallback )

dynamically load javascript files from url

##### Parameters

<i>urls</i>	array of urls for javascript files
<i>complete-Callback</i>	called when function is complete with successful parameter

#### 4.4.2.13 void Host.loadLanguage ( string url, event completeCallback )

load language JS

##### Parameters

<i>url</i>	app content url to language files
<i>complete-Callback</i>	called when function is complete with successful parameter

#### 4.4.2.14 void Host.release ( )

releases the host

#### 4.4.2.15 void Host.removeBundleAttachedEventHandler ( event *handler* )

unsubscribe from the BundleAttached event

##### Parameters

<i>handler</i>	callback function for BundleAttached event
----------------	--

#### 4.4.2.16 void Host.removeInteropLoadedEventHandler ( event *handler* )

unsubscribe from the InteropLoaded event

##### Parameters

<i>handler</i>	callback function for InteropLoaded event
----------------	---

#### 4.4.2.17 string Host.urlToFileName ( string *url*, string *directory*, string *extension* )

converts a url to a filename in the specified directory

##### Parameters

<i>url</i>	url to convert
<i>directory</i>	output directory
<i>extension</i>	extension of filename

##### Returns

path to file

### 4.4.3 Member Data Documentation

#### 4.4.3.1 app Host.app

[App](#) attached to host

#### 4.4.3.2 array Host.hexTable

Hex table for encoding/decoding hex



#### 4.4.3.3 SkinWindow Host.skinWindow

Window attached to host

The documentation for this class was generated from the following file:

- [host.js](#)

## 4.5 HttpDownload Class Reference

HTTP downloader.

### Public Member Functions

- int [getConnectTimeout](#) ()
- int [getExitCode](#) ()
- string [getFileName](#) ()
- string [getProxy](#) ()
- string [getProxyRealm](#) ()
- int [getRetryCount](#) ()
- string [getUrl](#) ()
- bool [getUseCache](#) ()
- bool [getUseWriteCallback](#) ()
- void [release](#) ()
- void [setConnectTimeout](#) (int value)
- void [setFileName](#) (string value)
- void [setRetryCount](#) (int value)
- void [setUrl](#) (string value)
- void [setUseCache](#) (bool value)
- void [setUseWriteCallback](#) (bool value)
- void [start](#) ()

### Public Attributes

- string [instanceId](#)
- event [onComplete](#)
- event [onDownloadRequired](#)
- event [onProgress](#)
- event [onStart](#)
- event [onWrite](#)

#### 4.5.1 Detailed Description

HTTP downloader.

## 4.5.2 Member Function Documentation

### 4.5.2.1 `int HttpDownload.getConnectTimeout ( )`

gets the seconds to wait for connection before timeout

#### **Returns**

seconds to wait for connection before timeout

### 4.5.2.2 `int HttpDownload.getExitCode ( )`

gets the exit code for download

#### **Returns**

exit code for download

### 4.5.2.3 `string HttpDownload.getFileName ( )`

gets the file name where url is saved

#### **Returns**

file name where url is saved

### 4.5.2.4 `string HttpDownload.getProxy ( )`

gets the proxy host and port

#### **Returns**

proxy host and port

### 4.5.2.5 `string HttpDownload.getProxyRealm ( )`

gets the proxy realm

#### **Returns**

proxy realm

### 4.5.2.6 `int HttpDownload.getRetryCount ( )`

gets the number of times to retry

#### **Returns**

number of times to retry

**4.5.2.7** string HttpDownload.getUrl ( )

gets the url to download

**Returns**

url to download

**4.5.2.8** bool HttpDownload.getUseCache ( )

gets the caching use in the http request

**Returns**

caching use in the http request

**4.5.2.9** bool HttpDownload.getUseWriteCallback ( )

gets the write callback use in the http request

**Returns**

write callback use in the http request

**4.5.2.10** void HttpDownload.release ( )

releases the object

**4.5.2.11** void HttpDownload.setConnectTimeout ( int *value* )

sets the seconds to wait for connection before timeout

**Parameters**

<i>value</i>	seconds to wait for connection before timeout
--------------	---

**4.5.2.12** void HttpDownload.setFileName ( string *value* )

sets the file name where url is saved

**Parameters**

<i>value</i>	file name where url is saved
--------------	------------------------------

**4.5.2.13 void HttpDownload.setRetryCount ( int *value* )**

sets the number of times to retry

**Parameters**

<i>value</i>	number of times to retry
--------------	--------------------------

**4.5.2.14 void HttpDownload.setUrl ( string *value* )**

sets the url to download

**Parameters**

<i>value</i>	url to download
--------------	-----------------

**4.5.2.15 void HttpDownload.setUseCache ( bool *value* )**

sets the caching use in the http request

**Parameters**

<i>value</i>	caching use in the http request
--------------	---------------------------------

**4.5.2.16 void HttpDownload.setUseWriteCallback ( bool *value* )**

sets the write callback use in the http request

**Parameters**

<i>value</i>	write callback use in the http request
--------------	--

**4.5.2.17 void HttpDownload.start ( )**

Start downloading

**4.5.3 Member Data Documentation****4.5.3.1 string HttpDownload.instanceId**

Instance id used to bind proxy object to native object

**4.5.3.2 event HttpDownload.onComplete**

Triggered when download is complete

**Parameters**

<i>successful</i>	true if the download was successful, false otherwise
-------------------	--

**4.5.3.3 event `HttpDownload.onDownloadRequired`**

Triggered when an actual download is required

**4.5.3.4 event `HttpDownload.onProgress`**

Triggered when download has made progress

**Parameters**

<i>percent</i>	(-1.0: Still calculating) (0.0 to 1.0: Percent completed)
----------------	---

**4.5.3.5 event `HttpDownload.onStart`**

Triggered when download is started

**4.5.3.6 event `HttpDownload.onWrite`**

Triggered when a write is required

**Parameters**

<i>buffer</i>	Hex converted buffer for the download
---------------	---------------------------------------

**Returns**

bool: true to continue download, false to cancel

The documentation for this class was generated from the following file:

- [httpdownload.js](#)

## 4.6 OSErrors Class Reference

Operating system helper functions.

**Public Member Functions**

- bool [isUnknown](#) (int id)
- string [nameFromId](#) (int id)

### 4.6.1 Detailed Description

Operating system helper functions.

### 4.6.2 Member Function Documentation

#### 4.6.2.1 `bool OSErr.isUnknown ( int id )`

checks if the error code is handled

##### Parameters

<i>id</i>	error code
-----------	------------

##### Returns

true if handled, false otherwise

#### 4.6.2.2 `string OSErr.nameFromId ( int id )`

converts an error code to a string

##### Parameters

<i>id</i>	error code
-----------	------------

##### Returns

stringified name of error code

The documentation for this class was generated from the following file:

- [oserror.js](#)

## 4.7 SkinWindow Class Reference

Solid State's HTML Skinned Window.

### Public Member Functions

- bool [beginMove](#) ()
- bool [beginSize](#) ()
- string [browseForFolder](#) (string text, string startPath)
- bool [close](#) ()
- int [getHeight](#) ()
- int [getWidth](#) ()

- int [getX](#) ()
- int [getY](#) ()
- bool [hide](#) ()
- bool [load](#) (string windowConfigUrl)
- bool [minimize](#) ()
- int [prompt](#) (string text, string caption, int type)
- void [release](#) ()
- bool [restore](#) ()
- bool [setOrigin](#) (int x, int y)
- bool [setSize](#) (int width, int height)
- void [setTitle](#) (string title)
- bool [show](#) ()

### Public Attributes

- string [instanceId](#)
- event [onDidClose](#)
- event [onWillClose](#)

#### 4.7.1 Detailed Description

Solid State's HTML Skinned Window.

#### 4.7.2 Member Function Documentation

##### 4.7.2.1 bool SkinWindow.beginMove ( )

This method informs the native window of any movement so that the native/browser windows can be synchronized

#### Returns

true upon success, false upon failure

##### 4.7.2.2 bool SkinWindow.beginSize ( )

This method informs the native window of any movement so that the native/browser windows can be synchronized

#### Returns

true upon success, false upon failure

#### 4.7.2.3 string SkinWindow.browseForFolder ( string *text*, string *startPath* )

This method allows the user to select a folder

##### Parameters

<i>text</i>	Message text to display
<i>startPath</i>	Folder to start in

##### Returns

path selected string

#### 4.7.2.4 bool SkinWindow.close ( )

This method will exit the launcher application

##### Returns

true upon success, false upon failure

#### 4.7.2.5 int SkinWindow.getHeight ( )

gets the height of the window (-2 if minimized, -1 if maximized)

##### Returns

height of the window (-2 if minimized, -1 if maximized)

#### 4.7.2.6 int SkinWindow.getWidth ( )

gets the width of the window (-2 if minimized, -1 if maximized)

##### Returns

width of the window (-2 if minimized, -1 if maximized)

#### 4.7.2.7 int SkinWindow.getX ( )

gets the x origin of the window

##### Returns

x origin of the window



#### 4.7.2.8 int SkinWindow.getY ( )

gets the y origin of the window

##### Returns

y origin of the window

#### 4.7.2.9 bool SkinWindow.hide ( )

This method will hide the window

##### Returns

true upon success, false upon failure

#### 4.7.2.10 bool SkinWindow.load ( string *windowConfigUrl* )

This creates a loads a new window

##### Parameters

<i>windowConfigUrl</i>	URL to the window configuration file
------------------------	--------------------------------------

##### Returns

true upon success, false upon failure

#### 4.7.2.11 bool SkinWindow.minimize ( )

This method will minimize the launcher application

##### Returns

true upon success, false upon failure

#### 4.7.2.12 int SkinWindow.prompt ( string *text*, string *caption*, int *type* )

This method prompts the user for a response

##### Parameters

<i>text</i>	Message text to display
<i>caption</i>	Caption for prompt
<i>type</i>	Prompt constant question type

**Returns**

Prompt constant return value

**4.7.2.13 void SkinWindow.release ( )**

releases the object

**4.7.2.14 bool SkinWindow.restore ( )**

This method will restore a minimized launcher application

**Returns**

true upon success, false upon failure

**4.7.2.15 bool SkinWindow.setOrigin ( int x, int y )**

Set the origin of the launcher window

**Parameters**

<i>x</i>	The desired x value of the launcher window
<i>y</i>	The desired y value of the launcher window

**Returns**

true upon success, false upon failure

**4.7.2.16 bool SkinWindow.setSize ( int width, int height )**

Set the size of the launcher window

**Parameters**

<i>width</i>	The desired width of the launcher window
<i>height</i>	The desired height of the launcher window

**Returns**

true upon success, false upon failure

**4.7.2.17 void SkinWindow.setTitle ( string title )**

Set the title of the launcher window

**Parameters**

<i>title</i>	The title of the launcher window
--------------	----------------------------------

**4.7.2.18 bool SkinWindow.show ( )**

This method will show the window

**Returns**

true upon success, false upon failure

**4.7.3 Member Data Documentation****4.7.3.1 string SkinWindow.instanceId**

Instance id used to bind proxy object to native object

**4.7.3.2 event SkinWindow.onDidClose**

Triggered when the window is closed

**4.7.3.3 event SkinWindow.onWillClose**

Triggered before the window is closed

**Returns**

bool: true to allow the window to close, false otherwise

The documentation for this class was generated from the following file:

- [skinwindow.js](#)

**4.8 SkinWindowPromptType Class Reference**

Skin window prompt question and return type flags.

**4.8.1 Detailed Description**

Skin window prompt question and return type flags.

The documentation for this class was generated from the following file:

- [skinwindowprompt.js](#)

## 4.9 String Class Reference

[String](#) extension.

### Public Member Functions

- void [replaceAll](#) (string oldString, string newString)
- void [trimChars](#) (string chars)

### 4.9.1 Detailed Description

[String](#) extension.

### 4.9.2 Member Function Documentation

#### 4.9.2.1 void [String.replaceAll](#) ( string *oldString*, string *newString* )

replaces all instances of one string with another

##### Parameters

<i>oldString</i>	string to find
<i>newString</i>	string to replace

#### 4.9.2.2 void [String.trimChars](#) ( string *chars* )

trims specified chars off the ends

##### Parameters

<i>chars</i>	string of characters to trim
--------------	------------------------------

The documentation for this class was generated from the following file:

- [jshelper.js](#)

## 4.10 TextFileReader Class Reference

Text file reader.

### Public Member Functions

- int [getBufferSize](#) ()

- string [getFileName](#) ()
- void [release](#) ()
- void [setBufferSize](#) (int value)
- void [setFileName](#) (string value)
- void [start](#) ()

### Public Attributes

- string [instanceId](#)
- event [onComplete](#)
- event [onProgress](#)
- event [onRead](#)
- event [onStart](#)

#### 4.10.1 Detailed Description

Text file reader.

#### 4.10.2 Member Function Documentation

##### 4.10.2.1 int TextFileReader.getBufferSize ( )

gets the max size of the buffer being read into

#### Returns

max size of the buffer being read into

##### 4.10.2.2 string TextFileReader.getFileName ( )

gets the file name where url is saved

#### Returns

file name where url is saved

##### 4.10.2.3 void TextFileReader.release ( )

releases the object

##### 4.10.2.4 void TextFileReader.setBufferSize ( int *value* )

sets the max size of the buffer being read into

#### Parameters

<i>value</i>	max size of the buffer being read into
--------------	--

#### 4.10.2.5 void `TextFileReader.setFileName` ( string *value* )

sets the file name where url is saved

##### Parameters

<i>value</i>	file name where url is saved
--------------	------------------------------

#### 4.10.2.6 void `TextFileReader.start` ( )

Start file reading

### 4.10.3 Member Data Documentation

#### 4.10.3.1 string `TextFileReader.instanceId`

Instance id used to bind proxy object to native object

#### 4.10.3.2 event `TextFileReader.onComplete`

Triggered when file reader is complete

##### Parameters

<i>successful</i>	true if the file reader was successful, false otherwise
-------------------	---

#### 4.10.3.3 event `TextFileReader.onProgress`

Triggered when file reader has made progress

##### Parameters

<i>percent</i>	(-1.0: Still calculating) (0.0 to 1.0: Percent completed)
----------------	---

#### 4.10.3.4 event `TextFileReader.onRead`

Triggered when file reader reads new bytes

##### Parameters

<i>text</i>	Bytes read from file
-------------	----------------------

## 4.10.3.5 event TextFileReader.onStart

Triggered when file reader is started

The documentation for this class was generated from the following file:

- [textfilereader.js](#)

## 4.11 TextFileWriter Class Reference

Text file writer.

### Public Member Functions

- bool [close](#) ()
- bool [open](#) (string fileName)
- void [release](#) ()
- int [write](#) (string value)

### Public Attributes

- string [instanceId](#)

#### 4.11.1 Detailed Description

Text file writer.

#### 4.11.2 Member Function Documentation

##### 4.11.2.1 bool TextFileWriter.close ( )

Close file

### Returns

true if successful, false otherwise

##### 4.11.2.2 bool TextFileWriter.open ( string *fileName* )

Open file for writing

### Parameters

<i>fileName</i>	name of the file to close
-----------------	---------------------------

**Returns**

true if successful, false otherwise

**4.11.2.3 void TextFileWriter.release ( )**

releases the object

**4.11.2.4 int TextFileWriter.write ( string *value* )**

Write string to file

**Parameters**

<i>value</i>	string to write to file
--------------	-------------------------

**Returns**

number of bytes written to file

**4.11.3 Member Data Documentation****4.11.3.1 string TextFileWriter.instanceId**

Instance id used to bind proxy object to native object

The documentation for this class was generated from the following file:

- [textfilewriter.js](#)



## Chapter 5

# File Documentation

### 5.1 app.js File Reference

File containing [App](#) class and creation function.

#### Classes

- class [App](#)  
*Solid State's Application Interface.*

#### Functions

- void [createApp](#) ()

#### 5.1.1 Detailed Description

File containing [App](#) class and creation function.

#### 5.1.2 Function Documentation

##### 5.1.2.1 void [createApp](#) ( )

Create instance of app

### 5.2 bundleloader.js File Reference

File containing [BundleLoader](#) class and creation function.

## Classes

- class [BundleLoader](#)

*Helper class to load bundles from JSON file.*

## Functions

- void [createBundleLoader](#) ()

### 5.2.1 Detailed Description

File containing [BundleLoader](#) class and creation function.

### 5.2.2 Function Documentation

#### 5.2.2.1 void [createBundleLoader](#) ( )

Create instance of bundle loader

## 5.3 [host.js](#) File Reference

File containing [Host](#) class.

## Classes

- class [Host](#)

*[Host](#) Top level container for the host.*

## Variables

- [Host](#) [host](#)

### 5.3.1 Detailed Description

File containing [Host](#) class.

### 5.3.2 Variable Documentation

#### 5.3.2.1 [Host](#) [host](#)

precreated global instance of the host

## 5.4 httpdownload.js File Reference

File containing [HttpDownload](#) class and creation function.

### Classes

- class [HttpDownload](#)  
*HTTP downloader.*

### Functions

- void [createHttpDownload](#) ()

#### 5.4.1 Detailed Description

File containing [HttpDownload](#) class and creation function.

#### 5.4.2 Function Documentation

##### 5.4.2.1 void [createHttpDownload](#) ( )

Create instance of httpDownload

## 5.5 osererror.js File Reference

File containing operating system helper functions.

### Classes

- class [OSError](#)  
*Operating system helper functions.*

### Variables

- [OSError](#) `osError`

#### 5.5.1 Detailed Description

File containing operating system helper functions.

## 5.5.2 Variable Documentation

### 5.5.2.1 `OSError` `osError`

precreated global instance of [OSError](#)

## 5.6 `skinwindow.js` File Reference

File containing [SkinWindow](#) class and creation function.

### Classes

- class [SkinWindow](#)  
*Solid State's HTML Skinned Window.*

### Functions

- void [createSkinWindow](#) ()

### 5.6.1 Detailed Description

File containing [SkinWindow](#) class and creation function.

### 5.6.2 Function Documentation

#### 5.6.2.1 void `createSkinWindow` ( )

Create instance of `skinWindow`

## 5.7 `skinwindowprompt.js` File Reference

File containing skin window prompt flags.

### Variables

- `SkinWindowPrompt` [skinWindowPrompt](#)

### 5.7.1 Detailed Description

File containing skin window prompt flags.

## 5.7.2 Variable Documentation

### 5.7.2.1 SkinWindowPrompt skinWindowPrompt

precreated global instance of SkinWindowPrompt

## 5.8 textfilereader.js File Reference

File containing [TextFileReader](#) class and creation function.

### Classes

- class [TextFileReader](#)  
*Text file reader.*

### Functions

- void [createTextFileReader](#) ()

### 5.8.1 Detailed Description

File containing [TextFileReader](#) class and creation function.

### 5.8.2 Function Documentation

#### 5.8.2.1 void createTextFileReader ( )

Create instance of textFileReader

## 5.9 textfilewriter.js File Reference

File containing [TextFileWriter](#) class and creation function.

### Classes

- class [TextFileWriter](#)  
*Text file writer.*

### Functions

- void [createTextFileWriter](#) ()

### 5.9.1 Detailed Description

File containing [TextFileWriter](#) class and creation function.

### 5.9.2 Function Documentation

#### 5.9.2.1 void createTextFileWriter ( )

Create instance of textFileWriter

# Index

addBundleAttachedEventHandler  
  Host, [19](#)  
addInteropLoadedEventHandler  
  Host, [19](#)  
App, [7](#)  
  attachBundle, [8](#)  
  authorizeUrl, [8](#)  
  convertFromEntitizedString, [8](#)  
  convertToEntitizedString, [9](#)  
  error, [9](#)  
  expandString, [9](#)  
  getCommandLine, [9](#)  
  getConfig, [10](#)  
  getCredential, [10](#)  
  getElevated, [10](#)  
  getImplode, [10](#)  
  getRemoveLocalStorage, [10](#)  
  getRestart, [11](#)  
  getRestartElevated, [11](#)  
  getRestartPatchDirectory, [11](#)  
  getRestartPatchFile, [11](#)  
  hasBundle, [11](#)  
  info, [12](#)  
  instanceId, [14](#)  
  loadInterop, [12](#)  
  onBundleAttached, [14](#)  
  onInteropLoaded, [14](#)  
  onWillAuthenticateWebGetTarget, [14](#)  
  onWillConstructWebGetTarget, [15](#)  
  onWillRestart, [15](#)  
  release, [12](#)  
  setCommandLine, [12](#)  
  setConfig, [12](#)  
  setCredential, [12](#)  
  setImplode, [13](#)  
  setRemoveLocalStorage, [13](#)  
  setRestart, [13](#)  
  setRestartElevated, [13](#)  
  setRestartPatchDirectory, [13](#)  
  setRestartPatchFile, [14](#)  
  warn, [14](#)  
app  
  Host, [22](#)  
app.js, [39](#)  
  createApp, [39](#)  
Array, [15](#)  
  forEach, [16](#)  
  forEachReverse, [16](#)  
  indexOf, [16](#)  
  removeElement, [16](#)  
  removeIndex, [16](#)  
attachBundle  
  App, [8](#)  
authorizeUrl  
  App, [8](#)  
beginMove  
  SkinWindow, [29](#)  
beginSize  
  SkinWindow, [29](#)  
browseForFolder  
  SkinWindow, [29](#)  
BundleLoader, [17](#)  
  name, [17](#)  
  onComplete, [17](#)  
  onProgress, [18](#)  
  start, [17](#)  
  url, [18](#)  
bundleloader.js, [39](#)  
  createBundleLoader, [40](#)  
close  
  SkinWindow, [30](#)  
  TextFileWriter, [37](#)  
convertFromEntitizedString  
  App, [8](#)  
convertToEntitizedString  
  App, [9](#)  
createApp  
  app.js, [39](#)  
createBundleLoader  
  bundleloader.js, [40](#)

- createHttpDownload
  - httpdownload.js, [41](#)
- createSkinWindow
  - skinwindow.js, [42](#)
- createTextFileReader
  - textfilereader.js, [43](#)
- createTextFileWriter
  - textfilewriter.js, [44](#)
- decodeHex
  - Host, [19](#)
- downloadContent
  - Host, [19](#)
- elevate
  - Host, [20](#)
- error
  - App, [9](#)
- expandString
  - App, [9](#)
- forEach
  - Array, [16](#)
- forEachReverse
  - Array, [16](#)
- getBufferSize
  - TextFileReader, [35](#)
- getCommandLine
  - App, [9](#)
- getConfig
  - App, [10](#)
- getConnectTimeout
  - HttpDownload, [24](#)
- getCredential
  - App, [10](#)
- getElevated
  - App, [10](#)
- getExitCode
  - HttpDownload, [24](#)
- getFileName
  - HttpDownload, [24](#)
  - TextFileReader, [35](#)
- getHeight
  - SkinWindow, [30](#)
- getImplode
  - App, [10](#)
- getJSON
  - Host, [20](#)
- getProxy
  - HttpDownload, [24](#)
- getProxyRealm
  - HttpDownload, [24](#)
- getRemoveLocalStorage
  - App, [10](#)
- getRestart
  - App, [11](#)
- getRestartElevated
  - App, [11](#)
- getRestartPatchDirectory
  - App, [11](#)
- getRestartPatchFile
  - App, [11](#)
- getRetryCount
  - HttpDownload, [24](#)
- getUrl
  - HttpDownload, [24](#)
- getUseCache
  - HttpDownload, [25](#)
- getUseWriteCallback
  - HttpDownload, [25](#)
- getWidth
  - SkinWindow, [30](#)
- getX
  - SkinWindow, [30](#)
- getY
  - SkinWindow, [30](#)
- hasBundle
  - App, [11](#)
- hexTable
  - Host, [22](#)
- hide
  - SkinWindow, [31](#)
- Host, [18](#)
  - addBundleAttachedEventHandler, [19](#)
  - addInteropLoadedEventHandler, [19](#)
  - app, [22](#)
  - decodeHex, [19](#)
  - downloadContent, [19](#)
  - elevate, [20](#)
  - getJSON, [20](#)
  - hexTable, [22](#)
  - idFromString, [20](#)
  - init, [20](#)
  - loadBundles, [20](#)
  - loadCSS, [21](#)
  - loadJS, [21](#)
  - loadJSONArray, [21](#)
  - loadLanguage, [21](#)



- release, [21](#)
- removeBundleAttachedEventHandler, [22](#)
- removeInteropLoadedEventHandler, [22](#)
- skinWindow, [22](#)
- urlToFileName, [22](#)
- host
  - host.js, [40](#)
- host.js, [40](#)
  - host, [40](#)
- HttpDownload, [23](#)
  - getConnectTimeout, [24](#)
  - getExitCode, [24](#)
  - getFileName, [24](#)
  - getProxy, [24](#)
  - getProxyRealm, [24](#)
  - getRetryCount, [24](#)
  - getUrl, [24](#)
  - getUseCache, [25](#)
  - getUseWriteCallback, [25](#)
  - instanceId, [26](#)
  - onComplete, [26](#)
  - onDownloadRequired, [27](#)
  - onProgress, [27](#)
  - onStart, [27](#)
  - onWrite, [27](#)
  - release, [25](#)
  - setConnectTimeout, [25](#)
  - setFileName, [25](#)
  - setRetryCount, [25](#)
  - setUrl, [26](#)
  - setUseCache, [26](#)
  - setUseWriteCallback, [26](#)
  - start, [26](#)
- httpdownload.js, [41](#)
  - createHttpDownload, [41](#)
- idFromString
  - Host, [20](#)
- indexOf
  - Array, [16](#)
- info
  - App, [12](#)
- init
  - Host, [20](#)
- instanceId
  - App, [14](#)
  - HttpDownload, [26](#)
  - SkinWindow, [33](#)
  - TextFileReader, [36](#)
- TextFileWriter, [38](#)
- isUnknown
- OSError, [28](#)
- load
  - SkinWindow, [31](#)
- loadBundles
  - Host, [20](#)
- loadCSS
  - Host, [21](#)
- loadInterop
  - App, [12](#)
- loadJS
  - Host, [21](#)
- loadJSArray
  - Host, [21](#)
- loadLanguage
  - Host, [21](#)
- minimize
  - SkinWindow, [31](#)
- name
  - BundleLoader, [17](#)
- nameFromId
  - OSError, [28](#)
- onBundleAttached
  - App, [14](#)
- onComplete
  - BundleLoader, [17](#)
  - HttpDownload, [26](#)
  - TextFileReader, [36](#)
- onDidClose
  - SkinWindow, [33](#)
- onDownloadRequired
  - HttpDownload, [27](#)
- onInteropLoaded
  - App, [14](#)
- onProgress
  - BundleLoader, [18](#)
  - HttpDownload, [27](#)
  - TextFileReader, [36](#)
- onRead
  - TextFileReader, [36](#)
- onStart
  - HttpDownload, [27](#)
  - TextFileReader, [36](#)
- onWillAuthenticateWebGetTarget
  - App, [14](#)

- onWillClose
  - SkinWindow, [33](#)
- onWillConstructWebGetTarget
  - App, [15](#)
- onWillRestart
  - App, [15](#)
- onWrite
  - HttpDownload, [27](#)
- open
  - TextFileWriter, [37](#)
- OSError, [27](#)
  - isUnknown, [28](#)
  - nameFromId, [28](#)
- osError
  - oserror.js, [42](#)
- oserror.js, [41](#)
  - osError, [42](#)
- prompt
  - SkinWindow, [31](#)
- release
  - App, [12](#)
  - Host, [21](#)
  - HttpDownload, [25](#)
  - SkinWindow, [32](#)
  - TextFileReader, [35](#)
  - TextFileWriter, [38](#)
- removeBundleAttachedEventHandler
  - Host, [22](#)
- removeElement
  - Array, [16](#)
- removeIndex
  - Array, [16](#)
- removeInteropLoadedEventHandler
  - Host, [22](#)
- replaceAll
  - String, [34](#)
- restore
  - SkinWindow, [32](#)
- setBufferSize
  - TextFileReader, [35](#)
- setCommandLine
  - App, [12](#)
- setConfig
  - App, [12](#)
- setConnectTimeout
  - HttpDownload, [25](#)
- setCredential
  - App, [12](#)
- setFileName
  - HttpDownload, [25](#)
  - TextFileReader, [36](#)
- setImplode
  - App, [13](#)
- setOrigin
  - SkinWindow, [32](#)
- setRemoveLocalStorage
  - App, [13](#)
- setRestart
  - App, [13](#)
- setRestartElevated
  - App, [13](#)
- setRestartPatchDirectory
  - App, [13](#)
- setRestartPatchFile
  - App, [14](#)
- setRetryCount
  - HttpDownload, [25](#)
- setSize
  - SkinWindow, [32](#)
- setTitle
  - SkinWindow, [32](#)
- setUrl
  - HttpDownload, [26](#)
- setUseCache
  - HttpDownload, [26](#)
- setUseWriteCallback
  - HttpDownload, [26](#)
- show
  - SkinWindow, [33](#)
- SkinWindow, [28](#)
  - beginMove, [29](#)
  - beginSize, [29](#)
  - browseForFolder, [29](#)
  - close, [30](#)
  - getHeight, [30](#)
  - getWidth, [30](#)
  - getX, [30](#)
  - getY, [30](#)
  - hide, [31](#)
  - instanceId, [33](#)
  - load, [31](#)
  - minimize, [31](#)
  - onDidClose, [33](#)
  - onWillClose, [33](#)
  - prompt, [31](#)
  - release, [32](#)
  - restore, [32](#)

- setOrigin, [32](#)
- setSize, [32](#)
- setTitle, [32](#)
- show, [33](#)
- skinWindow
  - Host, [22](#)
- skinwindow.js, [42](#)
  - createSkinWindow, [42](#)
- skinWindowPrompt
  - skinwindowprompt.js, [43](#)
- skinwindowprompt.js, [42](#)
  - skinWindowPrompt, [43](#)
- SkinWindowPromptType, [33](#)
- start
  - BundleLoader, [17](#)
  - HttpDownload, [26](#)
  - TextFileReader, [36](#)
- String, [34](#)
  - replaceAll, [34](#)
  - trimChars, [34](#)
- TextFileReader, [34](#)
  - getBufferSize, [35](#)
  - getFileName, [35](#)
  - instanceId, [36](#)
  - onComplete, [36](#)
  - onProgress, [36](#)
  - onRead, [36](#)
  - onStart, [36](#)
  - release, [35](#)
  - setBufferSize, [35](#)
  - setFileName, [36](#)
  - start, [36](#)
- textfilereader.js, [43](#)
  - createTextFileReader, [43](#)
- TextFileWriter, [37](#)
  - close, [37](#)
  - instanceId, [38](#)
  - open, [37](#)
  - release, [38](#)
  - write, [38](#)
- textfilewriter.js, [43](#)
  - createTextFileWriter, [44](#)
- trimChars
  - String, [34](#)
- url
  - BundleLoader, [18](#)
- urlToFileName
  - Host, [22](#)
- warn
  - App, [14](#)
- write
  - TextFileWriter, [38](#)