

# class FB JSAPI

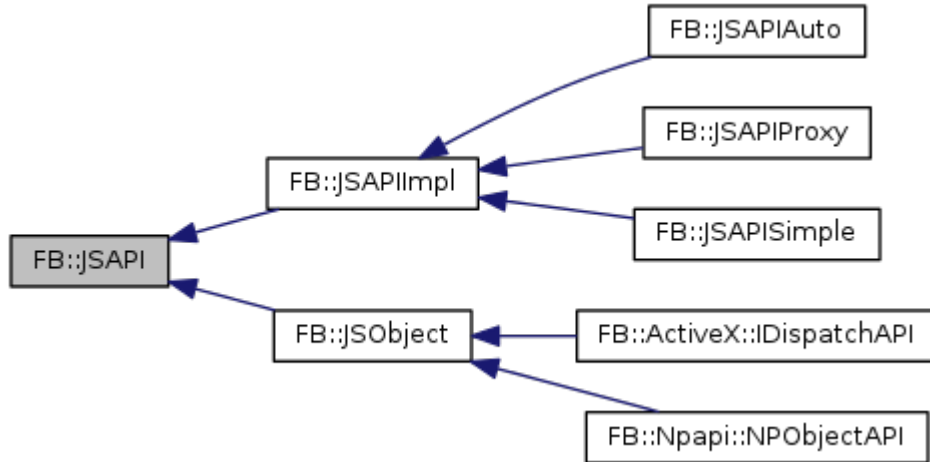
FB::JSAPI Class Referenceabstract

Public Member Functions | List of all members

JavaScript API class – provides a javascript interface that can be exposed to the browser. [More...](#)

```
#include "JSAPI.h"
```

Inheritance diagram for FB::JSAPI:



## Public Member Functions

**JSAPI** (void)  
Default constructor. [More...](#)

virtual **~JSAPI** (void)  
Finaliser. [More...](#)

virtual void **invalidate** ()=0  
Invalidates this object. [More...](#)

virtual void **shutdown** ()  
Called to notify the object that the plugin is shutting down. [More...](#)

virtual void **pushZone** (const **SecurityZone** &securityLevel)  
Pushes a new security level and locks a mutex (for every Push there *must* be a Pop!) [More...](#)

virtual void **popZone** ()  
Pops off a security level and unlocks the mutex (for every Push there *must* be a Pop!) [More...](#)

virtual void **setDefaultZone** (const **SecurityZone** &securityLevel)  
Sets the default zone (the zone the class operates on before a push) [More...](#)

virtual **SecurityZone** **getDefaultZone** () const  
Gets the default zone (the zone the class operates on before a push) [More...](#)

virtual **SecurityZone** **getZone** () const  
Gets the currently active zone. [More...](#)

virtual void **registerEventMethod** (const std::string &name, **JSObjectPtr** &event)  
Called by the browser to register an event handler method. [More...](#)

virtual void **registerEventMethod** (const std::wstring &name, **JSObjectPtr** &event)

virtual void **unregisterEventMethod** (const std::string &name, **JSObjectPtr** &event)  
Called by the browser to unregister an event handler method. [More...](#)

virtual void **unregisterEventMethod** (const std::wstring &name, **JSObjectPtr** &event)

virtual void	<b>registerEventInterface</b> (const <b>JSObjectPtr</b> &event) Called by the browser to register a <b>JSObject</b> interface that handles events. This is primarily used by IE. Objects provided to this method are called when events are fired by calling a method of the event name on the event interface. <a href="#">More...</a>
virtual void	<b>unregisterEventInterface</b> (const <b>JSObjectPtr</b> &event) Called by the browser to unregister a <b>JSObject</b> interface that handles events. <a href="#">More...</a>
virtual void	<b>getMemberNames</b> (std::vector< std::string > &nameVector) const =0 Called by the browser to enumerate the members of this <b>JSAPI</b> object. <a href="#">More...</a>
virtual size_t	<b>getMemberCount</b> () const =0 Gets the member count. <a href="#">More...</a>
virtual bool	<b>HasMethod</b> (const std::wstring &methodName) const
virtual bool	<b>HasMethod</b> (const std::string &methodName) const =0 Query if the <b>JSAPI</b> object has the 'methodName' method. <a href="#">More...</a>
virtual bool	<b>HasMethodObject</b> (const std::wstring &methodObjName) const
virtual bool	<b>HasMethodObject</b> (const std::string &methodObjName) const Query if 'methodObjName' is a valid methodObj. <a href="#">More...</a>
virtual bool	<b>HasProperty</b> (const std::wstring &propertyName) const
virtual bool	<b>HasProperty</b> (const std::string &propertyName) const =0 Query if 'propertyName' is a valid property. <a href="#">More...</a>
virtual bool	<b>HasProperty</b> (int idx) const =0 Query if the property at "idx" exists. <a href="#">More...</a>
virtual <b>JSAPIPtr</b>	<b>GetMethodObject</b> (const std::wstring &methodObjName)
virtual <b>JSAPIPtr</b>	<b>GetMethodObject</b> (const std::string &methodObjName) Gets a method object ( <b>JSAPI</b> object that has a default method) <a href="#">More...</a>
virtual <b>variant</b>	<b>GetProperty</b> (const std::wstring &propertyName)
virtual <b>variant</b>	<b>GetProperty</b> (const std::string &propertyName)=0 Gets a property value. <a href="#">More...</a>
virtual void	<b>SetProperty</b> (const std::wstring &propertyName, const <b>variant</b> &value)
virtual void	<b>SetProperty</b> (const std::string &propertyName, const <b>variant</b> &value)=0 Sets the value of a property. <a href="#">More...</a>
virtual <b>variant</b>	<b>GetProperty</b> (int idx)=0 Gets the value of an indexed property. <a href="#">More...</a>
virtual void	<b>SetProperty</b> (int idx, const <b>variant</b> &value)=0 Sets the value of an indexed property. <a href="#">More...</a>
virtual void	<b>RemoveProperty</b> (const std::wstring &propertyName)
virtual void	<b>RemoveProperty</b> (const std::string &propertyName)=0 Removes a property. <a href="#">More...</a>
virtual void	<b>RemoveProperty</b> (int idx)=0 Removes an indexed property. <a href="#">More...</a>
virtual <b>variant</b>	<b>Invoke</b> (const std::wstring &methodName, const std::vector< <b>variant</b> > &args)
virtual <b>variant</b>	<b>Invoke</b> (const std::string &methodName, const std::vector< <b>variant</b> > &args)=0 Called by the browser to invoke a method on the <b>JSAPI</b> object. <a href="#">More...</a>
virtual <b>variant</b>	<b>Construct</b> (const std::vector< <b>variant</b> > &args)=0

Called by the browser to construct the **JSAPI** object. [More...](#)

## Detailed Description

JavaScript API class – provides a javascript interface that can be exposed to the browser.

**JSAPI** is the core class for all interaction with javascript. All PluginCore-derived Plugin objects must provide a **JSAPI** object to provide the javascript interface for their <object> tag, and methods or properties of that object can return other **JSAPI** objects.

Important things to know about **JSAPI** objects:

- Unless you have unusual needs, you will most likely want to extend **FB::JSAPIAuto** instead of extending **JSAPI** directly.
- Any time you work with a **JSAPI** object you should use it with a `boost::shared_ptr`. **FB::JSAPIPtr** is a typedef for a `boost::shared_ptr<JSAPI>` which may be useful.
  - From inside the object you can use the `shared_from_this()` method to get a `shared_ptr` for "this"
- Objects passed in from javascript, including functions, will be passed in as **FB::JSObject** objects which extend **JSAPI**.

Author

Richard Bateman

See Also

**FB::JSAPIAuto**

Definition at line 56 of file **JSAPI.h**.

---

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

- **JSAPI.h**
-