

# class FB JSAPIAuto Invoke

Construct  
FireJSEvent  
get\_valid  
getAttribute  
getMemberCount  
getMemberNames  
GetMethodObject  
GetProperty  
GetProperty  
HasMethod  
HasMethodObject  
HasProperty  
HasProperty  
Invoke  
isReserved  
JSAPIAuto  
registerAttribute  
registerMethod  
registerProperty  
RemoveProperty  
RemoveProperty  
setAttribute  
SetProperty  
SetProperty  
setReserved  
ToString  
unregisterMethod  
unregisterProperty

```
FB::variant FB::JSAPIAuto::Invoke ( const std::string &          methodName,  
                                   const std::vector< variant > & args          virtual  
                                   )
```

Called by the browser to invoke a method on the **JSAPI** object.

#### Parameters

**methodName** Name of the method.  
**args** The arguments.

#### Returns

result of method call

Implements **FB::JSAPI**.

Definition at line 329 of file **JSAPIAuto.cpp**.

```
330 {  
331     boost::recursive_mutex::scoped_lock lock(m_zoneMutex);  
332     if(!m_valid)  
333         throw object_invalidated();  
334  
335     if (memberAccessible(m_zoneMap.find(methodName))) {  
336         try {  
337             MethodFunctorMap::iterator it = m_methodFunctorMap.find(methodName);  
338             if(it == m_methodFunctorMap.end())  
339                 throw invalid_member(methodName);  
340  
341             return it->second.call(args);  
342         } catch (const FB::bad_variant_cast& ex) {  
343             std::string errorMsg("Could not convert from ");  
344             errorMsg += ex.from;  
345             errorMsg += " to ";  
346             errorMsg += ex.to;  
347             throw FB::invalid_arguments(errorMsg);  
348         }  
349     } else {  
350         throw invalid_member(methodName);  
351     }  
352 }
```

**FB::invalid\_arguments**

Thrown by a JSAPI object when the argument(s) provided to a SetProperty or Invoke call are found to b...

**Definition:** [JSEExceptions.h:47](#)

**FB::bad\_variant\_cast**

Thrown when variant::cast<type> or variant::convert\_cast<type> fails because the type of the value st...

**Definition:** [variant.h:133](#)