{! Link.base = "../"; /\* all generated links will be relative to this \*/ !} {+include("static/header.html")+}

{+publish.classesIndex+}

# {! var classType = ""; if (data.isBuiltin()) { classType += "Built-In "; } if (data.isNamespace) { if (data.is('FUNCTION')) { classType += "Function "; } classType += "Namespace "; } else { classType += "Class "; } !} {+classType+}{+data.alias+}

Version {+ data.version +}.

Extends {+ data.augments .sort() .map( function($) { return new Link().toSymbol($); } ) .join(", ") +}.

{+resolveLinks(data.classDesc)+} {# isn't defined in any file #}

*Defined in:*  {+new Link().toSrc(data.srcFile)+}.

{+classType+}Summary

| Constructor Attributes | Constructor Name and Description |
| --- | --- |
| {! if (data.isPrivate) output += "<private> "; if (data.isInner) output += "<inner> "; !} | **{+ new Link().toSymbol(data.alias).inner('constructor')+}**{+ makeSignature(data.params) +}  {+resolveLinks(summarize(data.desc))+} |

{! var ownProperties = data.properties.filter(function($){return $.memberOf == data.alias && !$.isNamespace}).sort(makeSortby("name")); !} Field Summary

| Field Attributes | Field Name and Description |
| --- | --- |
| {! if (member.isPrivate) output += "<private> "; if (member.isInner) output += "<inner> "; if (member.isStatic) output += "<static> "; if (member.isConstant) output += "<constant> "; !} | {+member.memberOf+}.**{+new Link().toSymbol(member.alias).withText(member.name)+}**  {+resolveLinks(summarize(member.desc))+} |

{! var borrowedMembers = data.properties.filter(function($) {return $.memberOf != data.alias}); var contributers = []; borrowedMembers.map(function($) {if (contributers.indexOf($.memberOf) < 0) contributers.push($.memberOf)}); for (var i = 0, l = contributers.length; i < l; i++) { output += "Fields borrowed from class "+new Link().toSymbol(contributers[i])+": " + "" + borrowedMembers .filter( function($) { return $.memberOf == contributers[i] } ) .sort(makeSortby("name")) .map( function($) { return new Link().toSymbol($.alias).withText($.name) } ) .join(", ") + ""; } !} {! var ownMethods = data.methods.filter(function($){return $.memberOf == data.alias && !$.isNamespace}).sort(makeSortby("name")); !} Method Summary

| Method Attributes | Method Name and Description |
| --- | --- |
| {! if (member.isPrivate) output += "<private> "; if (member.isInner) output += "<inner> "; if (member.isStatic) output += "<static> "; !} | {+member.memberOf+}.**{+new Link().toSymbol(member.alias).withText(member.name.replace(/\^\d+$/, ''))+}**{+makeSignature(member.params)+}  {+resolveLinks(summarize(member.desc))+} |

{! var borrowedMembers = data.methods.filter(function($) {return $.memberOf != data.alias}); var contributers = []; borrowedMembers.map(function($) {if (contributers.indexOf($.memberOf) < 0) contributers.push($.memberOf)}); for (var i = 0, l = contributers.length; i < l; i++) { output += "Methods borrowed from class "+new Link().toSymbol(contributers[i])+": " + "" + borrowedMembers .filter( function($) { return $.memberOf == contributers[i] } ) .sort(makeSortby("name")) .map( function($) { return new Link().toSymbol($.alias).withText($.name) } ) .join(", ") + ""; } !} {! var ownEvents = data.events.filter(function($){return $.memberOf == data.alias && !$.isNamespace}).sort(makeSortby("name")); !} Event Summary

| Event Attributes | Event Name and Description |
| --- | --- |
| {! if (member.isPrivate) output += "<private> "; if (member.isInner) output += "<inner> "; if (member.isStatic) output += "<static> "; !} | {+member.memberOf+}.**{+new Link().toSymbol(member.alias).withText(member.name)+}**{+makeSignature(member.params)+}  {+resolveLinks(summarize(member.desc))+} |

{! var borrowedMembers = data.events.filter(function($) {return $.memberOf != data.alias}); var contributers = []; borrowedMembers.map(function($) {if (contributers.indexOf($.memberOf) < 0) contributers.push($.memberOf)}); for (var i = 0, l = contributers.length; i < l; i++) { output += "Events borrowed from class "+new Link().toSymbol(contributers[i])+": " + "" + borrowedMembers .filter( function($) { return $.memberOf == contributers[i] } ) .sort(makeSortby("name")) .map( function($) { return new Link().toSymbol($.alias).withText($.name) } ) .join(", ") + ""; } !}

{+classType+}Detail

{! if (data.isPrivate) output += "<private> "; if (data.isInner) output += "<inner> "; !} **{+ data.alias +}**{+ makeSignature(data.params) +}

{+resolveLinks(data.desc)+}

*Author:* {+data.author+}.

{+example+}

Parameters: {+((item.type)?""+("{"+(new Link().toSymbol(item.type)+"} ")) : "")+} **{+item.name+}** *Optional, Default: {+item.defaultValue+}* {+resolveLinks(item.desc)+} Deprecated: {+resolveLinks(data.deprecated)+} Since: {+ data.since +} Throws: {+((item.type)?"{"+(new Link().toSymbol(item.type))+"} " : "")+} **{+item.name+}** {+resolveLinks(item.desc)+} Returns: {+((item.type)?"{"+(new Link().toSymbol(item.type))+"} " : "")+}{+resolveLinks(item.desc)+} Requires: {+ new Link().toSymbol(item) +} See: {+ new Link().toSymbol(item) +}

Field Detail

{! if (member.isPrivate) output += "<private> "; if (member.isInner) output += "<inner> "; if (member.isStatic) output += "<static> "; if (member.isConstant) output += "<constant> "; !} {{+new Link().toSymbol(member.type)+}} {+member.memberOf+}.**{+member.name+}**

{+resolveLinks(member.desc)+}

*Defined in:*  {+new Link().toSrc(member.srcFile)+}.

*Author:* {+member.author+}.

{+example+}

Deprecated: {+ resolveLinks(member.deprecated) +} Since: {+ member.since +} See: {+ new Link().toSymbol(item) +} Default Value: {+resolveLinks(member.defaultValue)+}

Method Detail

{! if (member.isPrivate) output += "<private> "; if (member.isInner) output += "<inner> "; if (member.isStatic) output += "<static> "; !} {{+new Link().toSymbol(member.type)+}} {+member.memberOf+}.**{+member.name.replace(/\^\d+$/, '')+}**{+makeSignature(member.params)+}

{+resolveLinks(member.desc)+}

*Defined in:*  {+new Link().toSrc(member.srcFile)+}.

*Author:* {+member.author+}.

{+example+}

Parameters: {+((item.type)?"{"+(new Link().toSymbol(item.type))+"} " : "")+}**{+item.name+}** *Optional, Default: {+item.defaultValue+}* {+resolveLinks(item.desc)+} Deprecated: {+ resolveLinks(member.deprecated) +} Since: {+ member.since +} Throws: {+((item.type)?"{"+(new Link().toSymbol(item.type))+"} " : "")+} **{+item.name+}** {+resolveLinks(item.desc)+} Returns: {+((item.type)?"{"+(new Link().toSymbol(item.type))+"} " : "")+}{+resolveLinks(item.desc)+} Requires: {+ resolveLinks(item) +} See: {+ new Link().toSymbol(item) +}

Event Detail

{! if (member.isPrivate) output += "<private> "; if (member.isInner) output += "<inner> "; if (member.isStatic) output += "<static> "; !} {{+new Link().toSymbol(member.type)+}} {+member.memberOf+}.**{+member.name+}**{+makeSignature(member.params)+}

{+resolveLinks(member.desc)+}

*Defined in:*  {+new Link().toSrc(member.srcFile)+}.

*Author:* {+member.author+}.

{+example+}

Parameters: {+((item.type)?"{"+(new Link().toSymbol(item.type))+"} " : "")+}**{+item.name+}** *Optional, Default: {+item.defaultValue+}* {+ resolveLinks(item.desc) +} Deprecated: {+ resolveLinks(member.deprecated) +} Since: {+ member.since +} Throws: {+((item.type)?"{"+(new Link().toSymbol(item.type))+"} " : "")+} **{+item.name+}** {+ resolveLinks(item.desc) +} Returns: {+((item.type)?"{"+(new Link().toSymbol(item.type))+"} " : "")+}{+resolveLinks(item.desc)+} Requires: {+ resolveLinks(item) +} See: {+ new Link().toSymbol(item) +}

©{+JSDOC.opt.D.copyright+}

Documentation generated by [JsDoc Toolkit](http://code.google.com/p/jsdoc-toolkit/) {+JSDOC.VERSION+} on {+new Date()+}