KARLSRUHE INSTITUTE OF TECHNOLOGY

SOFTWARE ENGINEERING PRACTICE

WINTER TERM 2015/2016

rootJS

Node.js bindings for ROOT 6

Jonas Schwabe
Theo Beffart
Sachin Rajgopal
Christoph Wolff
Christoph Haas
Maximilian Früh

Contents

1. NodeApplication

describe class NodeApplication here

1.1. ctorCallback

Name	`NodeApplication::ctorCallback(args:  FunctionCallbackInfo<Value>)`
Visibility	Public
Parameters	*args: FunctionCallbackInfo<Value>*
Return value	**none**
behavior	describe beahviour

1.2. staticCtorCallback

Name	NodeApplication::staticCtorCallback(args: FunctionCallbackInfo<Value>)
Visibility	Public
Parameters	*args: FunctionCallbackInfo<Value>*
Return value	**none**
behavior	describe beahviour

1.3. memberGetterCallback

Name	`NodeApplication::memberGetterCallback(property: Local<String>, info: PropertyCallbackInfo<Value>)`
Visibility	Public
Parameters	*property: Local<String>, info: PropertyCallbackInfo<Value>*
Return value	**none**
behavior	describe beahviour

1.4. memberSetterCallback

Name	`NodeApplication::memberSetterCallback(property: Local<String>, value: Local<Value>, info: PropertyCallbackInfo<Value>)`
Visibility	Public
Parameters	*property: Local<String>, value: Local<Value>, info: PropertyCallbackInfo<Value>*
Return value	**none**
behavior	describe beahviour

1.5. memberFunctionCallback

Name	NodeApplication::memberFunctionCallback(args: FunctionCallbackInfo<Value>)
Visibility	Public
Parameters	*args: FunctionCallbackInfo<Value>*
Return value	**none**
behavior	describe beahviour

1.6. staticGetterCallback

Name	`NodeApplication::staticGetterCallback(property:  Local<String>, info:  PropertyCallbackInfo<Value>)`
Visibility	Public
Parameters	*property: Local<String>, info: PropertyCallbackInfo<Value>*
Return value	**none**
behavior	describe beahviour

1.7. staticSetterCallback

Name	`NodeApplication::staticSetterCallback(property: Local<String>, value: Local<Value>, info: PropertyCallbackInfo<Value>)`
Visibility	Public
Parameters	*property: Local<String>, value: Local<Value>, info: PropertyCallback-Info<Value>*
Return value	**none**
behavior	describe beahviour

1.8. staticFunctionCallback

Name	NodeApplication::staticFunctionCallback(args: FunctionCallbackInfo<Value>)
Visibility	Public
Parameters	*args: FunctionCallbackInfo<Value>*
Return value	**none**
behavior	describe beahviour

1.9. Initialize

Name	`NodeApplication::Initialize(exports:  Local<Object>, module:` `Local<Object>)`
Visibility	Public
Parameters	*exports: Local<Object>, module: Local<Object>*
Return value	**none**
behavior	describe beahviour

1.10. Instance

Name	NodeApplication::Instance()
Visibility	Public
Parameters	*none*
Return value	**NodeApplication** describe return value
behavior	describe beahviour

1.11. getIsolate

Name	`NodeApplication::getIsolate()`
Visibility	Public
Parameters	*none*
Return value	**Isolate*** describe return value
behavior	describe beahviour

1.12. getExports

Name	`NodeApplication::getExports()`
Visibility	Public
Parameters	*none*
Return value	**Local**<**Object**> describe return value
behavior	describe beahviour

1.13. getTemplateFactory

Name	NodeApplication::getTemplateFactory()
Visibility	Public
Parameters	none
Return value	**TemplateFactory** describe return value
behavior	describe beahviour

1.14. getFunctionFactory

Name	`NodeApplication::getFunctionFactory()`
Visibility	Public
Parameters	*none*
Return value	**ProxyFunctionFactory** describe return value
behavior	describe beahviour

1.15. getObjectFactory

Name	NodeApplication::getObjectFactory()
Visibility	Public
Parameters	*none*
Return value	**ProxyObjectFactory** describe return value
behavior	describe beahviour

2. TemplateFactory

describe class TemplateFactory here

2.1. createTemplate

Name	`TemplateFactory::createTemplate(clazz:  TClassRef)`
Visibility	Public
Parameters	*clazz: TClassRef*
Return value	**Local<FunctionTemplate>** describe return value
behavior	describe beahviour

3. TemplateCache

describe class TemplateCache here

3.1. contains

Name	TemplateCache::contains(type: TClassRef)
Visibility	Public
Parameters	type: TClassRef
Return value	**bool** describe return value
behavior	describe beahviour

3.2. get

Name	TemplateCache::get(type: TClassRef)
Visibility	Public
Parameters	type: TClassRef
Return value	**Local**<**FunctionTemplate**> describe return value
behavior	describe beahviour

3.3. store

Name	TemplateCache::store(type: TClassRef, tpl: Local<FunctionTemplate>)
Visibility	Public
Parameters	*type: TClassRef, tpl: Local<FunctionTemplate>*
Return value	**none**
behavior	describe beahviour

4. ProxyFunctionFactory

describe class ProxyFunctionFactory here

4.1. createProxyFunction

Name	`ProxyFunctionFactory::createProxyFunction(info:   TMethod)`
Visibility	Public
Parameters	*info: TMethod*
Return value	**ProxyFunciton** describe return value
behavior	describe beahviour

4.2. fromArgs

Name	`ProxyFunctionFactory::fromArgs(name:  string, clazz:  TClassRef, args:  FunctionCallbackInfo)`
Visibility	Public
Parameters	*name: string, clazz: TClassRef, args: FunctionCallbackInfo*
Return value	**ProxyFunction** describe return value
behavior	describe beahviour

5. ProxyFunction

describe class ProxyFunction here

5.1. ProxyFunction

Name	ProxyFunction::ProxyFunction(address: void*, info: TFunction)
Visibility	Public
Parameters	*address: void*, info: TFunction*
Return value	**«constructor»** describe return value
behavior	describe beahviour

5.2. convertArgs

Name	`ProxyFunction::convertArgs(args:  FunctionCallbackInfo)`
Visibility	Public
Parameters	*args: FunctionCallbackInfo*
Return value	**ProxyObject[]** describe return value
behavior	describe beahviour

5.3. call

Name	`ProxyFunction::call(args: ProxyObject[])`
Visibility	Public
Parameters	*args: ProxyObject[]*
Return value	**ProxyObject** describe return value
behavior	describe beahviour

5.4. isTemplateFunction

Name	`ProxyFunction::isTemplateFunction()`
Visibility	Public
Parameters	none
Return value	**bool** describe return value
behavior	describe beahviour

6. ProxyObjectFactory

describe class ProxyObjectFactory here

6.1. createProxyObject

Name	`ProxyObjectFactory::createProxyObject(type: TDataMember, holder: ProxyObject)`
Visibility	Public
Parameters	*type: TDataMember, holder: ProxyObject*
Return value	**ProxyObject** describe return value
behavior	describe beahviour

7. ProxyObject

describe class ProxyObject here

7.1. ProxyObject

Name	ProxyObject::ProxyObject(address: void*, type: TDataMember)
Visibility	Public
Parameters	*address: void*, type: TDataMember*
Return value	**«constructor»** describe return value
behavior	describe beahviour

7.2. getAddress

Name	`ProxyObject::getAddress()`
Visibility	Public
Parameters	*none*
Return value	**void*** describe return value
behavior	describe beahviour

7.3. getType

Name	ProxyObject::getType()
Visibility	Public
Parameters	*none*
Return value	**TDataMember** describe return value
behavior	describe beahviour

7.4. set

Name	`ProxyObject::set(value:  ProxyObject)`
Visibility	Public
Parameters	*value: ProxyObject*
Return value	**none**
behavior	describe beahviour

7.5. get

Name	`ProxyObject::get()`
Visibility	Public
Parameters	*none*
Return value	**Local**<**Value**> describe return value
behavior	describe beahviour

7.6. isPrimitive

Name	`ProxyObject::isPrimitive()`
Visibility	Public
Parameters	*none*
Return value	**bool** describe return value
behavior	describe beahviour