

Exception

1.0.5

Generated by Doxygen 1.8.4

Tue Apr 29 2014 12:57:50

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	Class Documentation	5
3.1	Exception Class Reference	5
3.1.1	Detailed Description	7
3.1.2	Constructor & Destructor Documentation	7
3.1.2.1	Exception	7
3.1.2.2	Exception	7
3.1.3	Member Function Documentation	7
3.1.3.1	begin	7
3.1.3.2	end	7
3.1.3.3	get_message	7
3.1.3.4	print	7
3.1.3.5	set_message	7
3.1.4	Friends And Related Function Documentation	8
3.1.4.1	operator<<	8
3.1.4.2	operator<<	8
4	Example Documentation	9
4.1	test.cc	9

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

std::exception	
std::runtime_error	
Exception	5

Chapter 2

Class Index

2.1 Class List

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

Exception	5
-------------------------------------	---

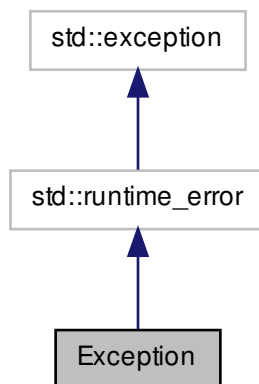
Chapter 3

Class Documentation

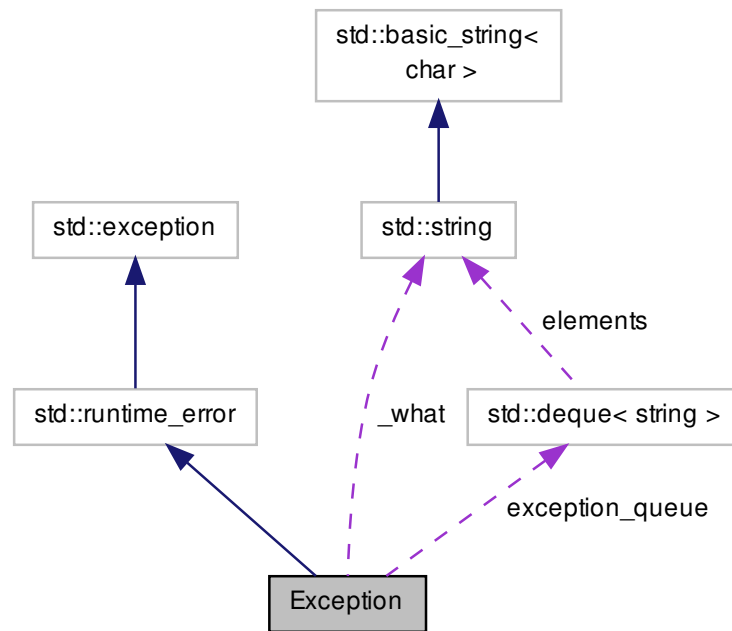
3.1 Exception Class Reference

```
#include <Exception.h>
```

Inheritance diagram for Exception:



Collaboration diagram for Exception:



Public Member Functions

- `Exception ()`
- `Exception (const Exception &e)`
- `Exception (const string &arg)`
- `Exception & operator= (const Exception &rhs)`
- `deque< string >::const_iterator begin () const`
- `deque< string >::const_iterator end () const`
- `string get_message (void) const`
- `void set_message (const string &msg)`
- `void set_rethrow_message (const string &file, const int linenum)`
- `const char * what () const throw ()`
- `void update_what ()`

Protected Member Functions

- `virtual void print (ostream &os=cerr) const`

Protected Attributes

- `deque< string > exception_queue`
- `string _what`

Friends

- ostream & [operator<<](#) (ostream &os, [Exception](#) &a)
- ostream & [operator<<](#) (ostream &os, [Exception](#) *a)

3.1.1 Detailed Description

The class [Exception](#), all the subclasses as defined by the user, are a form of conditions that a reasonable application might want to catch.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 [Exception::Exception \(\)](#) `[inline]`

Constructs an [Exception](#) with no specified detail message.

3.1.2.2 [Exception::Exception \(const string & arg \)](#) `[inline]`

Constructs an [Exception](#) with a specified detail message.

3.1.3 Member Function Documentation

3.1.3.1 [deque<string>::const_iterator Exception::begin \(\) const](#) `[inline]`

Get the iterator pointing to the beginning of the dequeue.

Referenced by `print()`.

3.1.3.2 [deque<string>::const_iterator Exception::end \(\) const](#) `[inline]`

Get the iterator pointing to the end of the dequeue.

Referenced by `print()`.

3.1.3.3 [string Exception::get_message \(void \) const](#) `[inline]`

Get the most current message

3.1.3.4 [void Exception::print \(ostream & os = cerr \) const](#) `[protected], [virtual]`

Prints this and its backtrace to the specified output stream.

References `begin()`, and `end()`.

3.1.3.5 [void Exception::set_message \(const string & msg \)](#)

Add a message.

3.1.4 Friends And Related Function Documentation

3.1.4.1 ostream& operator<< (ostream & os, Exception & a) [friend]

Prints this and its backtrace to the specified output stream.

3.1.4.2 ostream& operator<< (ostream & os, Exception * a) [friend]

Prints this and its backtrace to the specified output stream.

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

- Exception.h
- Exception.cc

Chapter 4

Example Documentation

4.1 test.cc

This is an example of how to use the Test class. More details about this example.

Index

- begin
 - Exception, [7](#)
- end
 - Exception, [7](#)
- Exception, [5](#)
 - begin, [7](#)
 - end, [7](#)
 - Exception, [7](#)
 - get_message, [7](#)
 - operator<<, [8](#)
 - print, [7](#)
 - set_message, [7](#)
- get_message
 - Exception, [7](#)
- operator<<
 - Exception, [8](#)
- print
 - Exception, [7](#)
- set_message
 - Exception, [7](#)