

libAHstl Reference Manual

0.1

Generated by Doxygen 1.4.4

Fri Apr 7 01:18:07 2006

Contents

1	libAHstl Directory Hierarchy	1
1.1	libAHstl Directories	1
2	libAHstl Namespace Index	3
2.1	libAHstl Namespace List	3
3	libAHstl File Index	5
3.1	libAHstl File List	5
4	libAHstl Directory Documentation	7
4.1	AH/ Directory Reference	7
4.2	AH/STL/ Directory Reference	8
5	libAHstl Namespace Documentation	9
5.1	AH Namespace Reference	9
5.2	std Namespace Reference	11
6	libAHstl File Documentation	13
6.1	AH/STL/String.h File Reference	13

Chapter 1

libAHstl Directory Hierarchy

1.1 libAHstl Directories

This directory hierarchy is sorted roughly, but not completely, alphabetically:

AH	7
STL	8

Chapter 2

libAHstl Namespace Index

2.1 libAHstl Namespace List

Here is a list of all namespaces with brief descriptions:

AH	9
std	11

Chapter 3

libAHstl File Index

3.1 libAHstl File List

Here is a list of all files with brief descriptions:

AH/STL/ String.h	13
--	----

Chapter 4

libAHstl Directory Documentation

4.1 AH/ Directory Reference

Directories

- directory [STL](#)

4.2 AH/STL/ Directory Reference

Files

- file [String.h](#)

Chapter 5

libAHstl Namespace Documentation

5.1 AH Namespace Reference

Functions

- void [STLStringToken](#) (const string &line, vector< string > &result, const char *separator=AH_WHITE_SPACE)
extract all tokens from a STL string in one call
- void [STLStringDelLead](#) (string &line, const char *remove=AH_WHITE_SPACE)
removes leading characters from { line}
- void [STLStringDelFollow](#) (string &line, const char *remove=AH_WHITE_SPACE)
removes following characters from { line}
- void [STLStringDelSurround](#) (string &line, const char *remove=AH_WHITE_SPACE)
removes leading and following characters from { line}
- string [STLStringPrintf](#) (unsigned int maxSize, const char *format,...)

5.1.1 Function Documentation

5.1.1.1 void AH::STLStringDelFollow (string & line, const char * remove = AH_WHITE_SPACE)

removes following characters from { line}

{ [STLStringDelFollow\(\)](#) } removes all following characters in { line}. The set of characters to be removed can be specified. Without explicit specification white space is removed.

Parameters:

line String where the characters are removed

remove which characters to remove (default is white space)

5.1.1.2 void AH::STLStringDelLead (string & *line*, const char * *remove* = AH_WHITE_SPACE)

removes leading characters from { *line* }

{ [STLStringDelLead\(\)](#) } removes all leading { characters } in { *line* }. The set of characters to be removed can be specified. Without explicit specification white space is removed.

Parameters:

line String where the characters are removed

remove which characters to remove (default is white space)

5.1.1.3 void AH::STLStringDelSurround (string & *line*, const char * *remove* = AH_WHITE_SPACE) [inline]

removes leading and following characters from { *line* }

{ [STLStringDelSurround\(\)](#) } removes all leading and following characters in { *line* }. The set of characters to be removed can be specified. Without explicit specification white space is removed.

Parameters:

line String where the characters are removed

remove which characters to remove (default is white space)

5.1.1.4 string AH::STLStringPrintf (unsigned int *maxSize*, const char * *format*, ...)

convert printf(3S) compatible format string into STL string.

{ [STLStringPrintf\(\)](#) } converts a printf(3S) compatible format string and all arguments into a STL string. The maximum size is 1024 characters including the terminating 0. If this size is too short then use the other version of { [STLStringPrintf\(\)](#) } which accepts the desired size as first argument.

If the size of the buffer is exceeded the returned string is either empty or truncated (depends on the C99 compatibility of the C++ compiler).

Parameters:

maxSize maximum size of message (number of characters)

format printf(3S) compatible format string

... remaining arguments according to format string

5.1.1.5 void AH::STLStringToken (const string & *line*, vector< string > & *result*, const char * *separator* = AH_WHITE_SPACE)

extract all tokens from a STL string in one call

{ [STLStringToken\(\)](#) } works on a { string }(STL) like { strtok(3S) } on C type character arrays. Other than { strtok() } all the work is done in one call, storing the tokens in the string vector { *result* }.

{ UtilStringToken() } does not clear the string vector, so this must be done before the function call.

Parameters:

line string to extract tokens from

result sequential string container

separator token separator characters (default is white space)

5.2 std Namespace Reference

Chapter 6

libAHstl File Documentation

6.1 AH/STL/String.h File Reference

```
#include <AH/Config/Platform.h>
#include <string>
#include <vector>
```

Namespaces

- namespace [AH](#)
- namespace [std](#)

Defines

- #define [AH_WHITE_SPACE](#) " \t\n\r"

Functions

- void [AH::STLStringToken](#) (const string &line, vector< string > &result, const char *separator=AH_WHITE_SPACE)
extract all tokens from a STL string in one call
- void [AH::STLStringDelLead](#) (string &line, const char *remove=AH_WHITE_SPACE)
removes leading characters from { line}
- void [AH::STLStringDelFollow](#) (string &line, const char *remove=AH_WHITE_SPACE)
removes following characters from { line}
- void [AH::STLStringDelSurround](#) (string &line, const char *remove=AH_WHITE_SPACE)
removes leading and following characters from { line}
- string [AH::STLStringPrintf](#) (unsigned int maxSize, const char *format,...)

6.1.1 Define Documentation

6.1.1.1 `#define AH_WHITE_SPACE " \t\n\r"`

Index

AH, [9](#)
 STLStringDelFollow, [9](#)
 STLStringDelLead, [9](#)
 STLStringDelSurround, [10](#)
 STLStringPrintf, [10](#)
 STLStringToken, [10](#)
AH/ Directory Reference, [7](#)
AH/STL/ Directory Reference, [8](#)
AH/STL/String.h, [13](#)
AH_WHITE_SPACE
 String.h, [14](#)

std, [11](#)
STLStringDelFollow
 AH, [9](#)
STLStringDelLead
 AH, [9](#)
STLStringDelSurround
 AH, [10](#)
STLStringPrintf
 AH, [10](#)
STLStringToken
 AH, [10](#)
String.h
 AH_WHITE_SPACE, [14](#)