U.S. DEPARTMENT OF ENERGY

PNNL-22602

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

The stringTricks Utility Library

DES-0042

Revision 1

Charlie Hubbard

June 2012

Pacific Northwest
NATIONAL LABORATORY

Proudly Operated by **Battelle** *Since 1965*

The stringTricks Utility Library

Charlie Hubbard

DES-0042
Revision 1
June 2012

DISCLAIMER

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under
Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from the Office of Scientific and Technical Information, P.O. Box 62, Oak Ridge, TN 37831-0062
ph: (865) 576-8401
fax: (865) 576-5728
email: reports@adonis.osti.gov

Available to the public from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Rd., Springfield, VA 22161
ph: (800) 553-6847
fax: (703) 605-6900
email: orders@ntis.fedworld.gov
online ordering: http://www.ntis.gov/ordering.htm

Contents

1 Introduction

In many, many places throughout the code base there is a need to perform operations on STL `std::string` text strings that are not supported by the `std::string` class. Over the years, we have developed a library of useful string manipulation functions. That library, defined by the file *stringTricks.h* and implemented by *stringTricks.cpp* is the focus of this document.

The primary documentation for the *stringTricks* library is the Doxygen-generated HTML documentation associated with the above source files. That documentation set is automatically built based on the source code itself. It provides the most detailed, most up-to-date descriptions of the code. The document you are reading now is supplemental. If contradictions between this document and the Doxygen-generated documentation are found, the Doxygen-generated documentation should be considered correct.

2 Library Functions

The *stringTricks* library contains the following string manipulation functions shown below (these definitions come from the file *stringTricks.h*). For a detailed description of the various function parameters, refer to the corresponding Doxygen documentation.

```
string  FormatFPNumber(const double number,
                       const int precision,
                       const double minThresh = 0.1,
                       const double maxThresh = 100000.0);
string& Replace(string &s, const string &oldSub, const string &newSub);
string  StringToLower(const string &original);
string  StringToUpper(const string &original);
string  ToHexString(unsigned long x, int width = 4);
string  Trim(const string &original);
```

2.1 `FormatFPNumber()`

This function is used to convert a floating point number to a string suitable for use on terminal displays, GUIs, data files, printed forms, etc. The caller specifies the number of decimal places of precision he wants for the output string, as well as the minimum and maximum magnitude threshold values. If the value to be converted is outside the specified range, the resulting string will be formatted using scientific notation; otherwise simple decimal notation is used.

2.2 `Replace()`

This function implements a global sub-string replacement ability. The caller specifies the string to be operated on, the sub-string within that string that should be replaced, and the text to replace it. The caller should be aware that the subsitutions take place in the *original* string (as opposed to a new string being returned, leaving the original untouched).

2.3 `StringToLower()`

This function replaces all uppercase letters in the supplied string with their lowercase equivalents. Unlike the `Replace()` function, this function does leave the original string untouched.

2.4 `StringToUpper()`

This function replaces all lowercase letters in the supplied string with their uppercase equivalents. Unlike the `Replace()` function, this function does leave the original string untouched.

2.5 `ToHexString()`

This function converts an integer value to a string in which the integer is represented in hexadecimal format. The caller can optionally set the total number of digits to be presented in the output string (padded to the left with zeros). By default, the returned string contains four digits, not including the "0x" prefix, which is present on all returned strings.

2.6 `Trim()`

This function trims leading and trailing whitespace off the caller-supplied string. The original string is left unmodified.

3 Linking Considerations

Users of the library must link their code against the *stringTricks.o* object file. However, some functions in the library make use of our regular expression library in their implementations. For this reason, users must also link against the *re.o* object file. For more information on our regular expression library, see design document ***DES-0008*, *The RE_c Utility Class***.

Pacific Northwest
NATIONAL LABORATORY

Proudly Operated by **Battelle** *Since 1965*

U.S. DEPARTMENT OF
ENERGY

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99352
1-888-375-PNNL (7665)
www.pnnl.gov