Boost C++ Libraries

Goals and Scope

- Give a global overview of functionality in Boost C++ libraries
- Installing Boost, run code, finding resources and documentation
- Selecting libraries that we use in computational finance applications
- Improve programmer productivity by using industrial-strength C++ template-based C++ libraries

2

What is Boost?

- Suite of approximately 100 C++ libraries that implement functionality in certain domain
- Header-file source code (www.boost.org)
- You can freely distribute the code in commercial and non-commercial applications
- Peer-reviewed and high-quality code

3

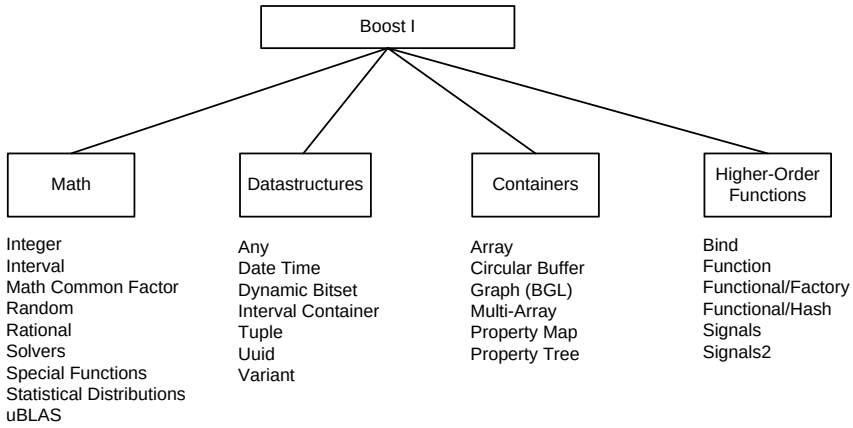
Boost Library Categories

- Math (Statistics, RNG, special functions, matrix and vectors)
- Datastructures (date-time, variant(union), tuple)
- Containers (array, multiarray, graph)
- Higher-order functions (functions, bind, signals)

- Others (concurrent programming, text, I/O, memory)

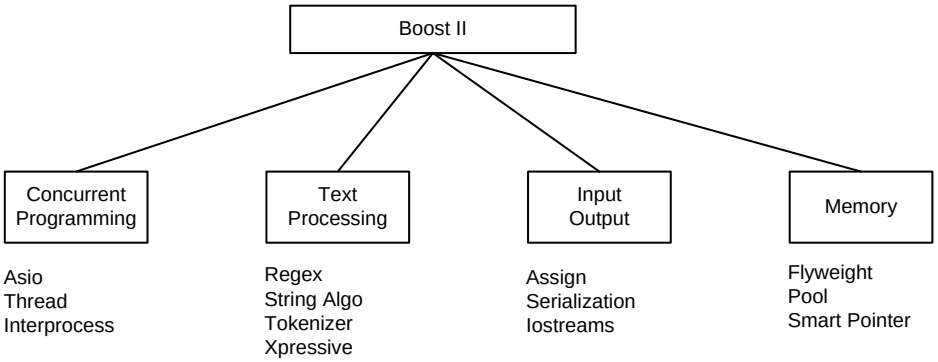
4

Specific Libraries (1/2)



5

Specific Libraries (2/2)



6

Resources

- www.boost.org
- www.datasimfinancial.com
- “Introduction to Boost C++ Libraries” (Demming/Duffy), see www.datasim-press.com
- Automagic Boost installer on www.boostpro.com (downloads)
- (VS-specific versions of the libraries)

7

Math Toolkit

- Extensive library for univariate (30) statistical distributions, special functions, nonlinear solvers
- Discrete and continuous statistical distributions + related free functions (e.g. pdf, cdf, kurtosis)
- Functions (Gamma, erf, Bessel, orthogonal functions, factorials)
- Solvers (Brent, Newton-Raphson, bisection, Halley, polynomials and rational functions)

8

uBLAS

- Matrix and vector library
- Datastructures (dense, sparse, banded,..)
- BLAS operations (*, + and combinations)
- Can create ‘views’ of matrices
- No support for numerical linear algebra

9

Array and MultiArray

- Higher-dimensional matrix/tensor class
- boost::array is a compile-time array class
- Useful for n-factor applications (e.g. PDE/FDM methods)

10

Date Time

- Extensive functionality for manipulating dates and time
- Support for many different kinds of applications
- Can use in fixed income applications (day count conventions, business dates)
- Avoids our having to create home-grown date classes

11

Random Number Generation

- Supports a number of congruential generators to generate pseudo-random numbers
- Generators (Mersenne Twister, lagged Fibonacci, rand48,...)
- Distributions (discrete and continuous cases)
- Useful for Monte Carlo applications

12

Higher-Order Functions

- Generalisation of function pointers and function objects
- Results in flexible code
- Some support for functional programming model (delayed arguments, unnamed (lambda) functions)
- The basis for next-generation design patterns

13

Smart Pointers

- Takes away some of the pain of 'raw' memory management (new/delete)
- Reference-counting mechanism
- Of particular applicability with STL containers
- Shared pointers are part of new C++ standard

14

Student Prerequisites and Approach

- Good knowledge of templates and template specialisation
- Knowing how to define the ‘link’ to Boost include files from your compiler
- Step-by-step approach in order not to get swamped by compiler errors
- Discovering which functionality in which libraries is most useful

15

Some Scenarios

- It is better to use Boost instead of creating your own home-grown code
- Develop new applications by ‘embedding’ Boost code in your classes
- Two main areas: data and functions
- We can use Boost libraries using OOP, GP and modular programming models
- Adapters, wrappers and Composition/Delegation principle important

16

What do we use?

- uBLAS and MultiArray structures for FDM
- uBLAS for LU decomposition and Cholesky decomposition
- Random library in combination with MC option pricers
- Math Toolkit to help with analytic option pricers (for example, non central chisquared distributions)
- (General utility modules)
- We shall see code examples, later

17

Summary

- We gave a focused overview of the Boost C++ Libraries
- The libraries are easy to use once you know 'trick'
- Libraries for data structures and mathematics particularly useful in the current context
- Excellent libraries!

18