

# Geant 4

## Toward Geant4 version 10

Makoto Asai (SLAC PPA/SCA)  
For the Geant4 Collaboration  
Geant4 Technical Forum  
December 6<sup>th</sup>, 2012



NATIONAL  
ACCELERATOR  
LABORATORY

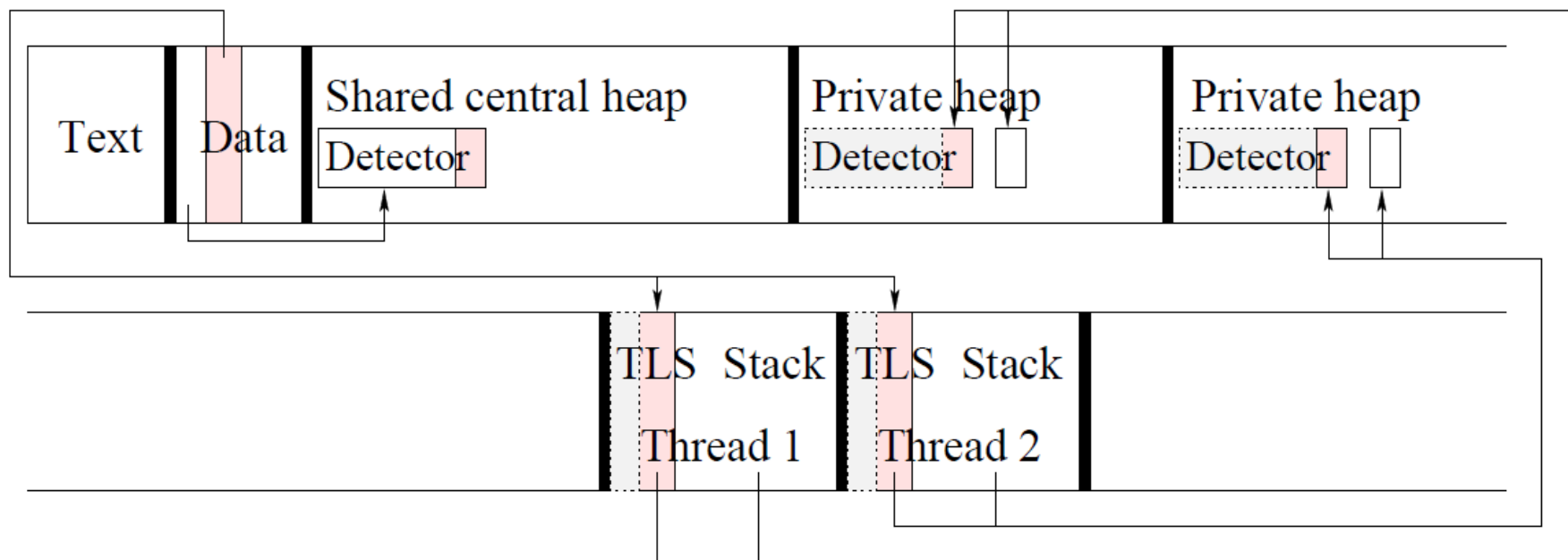


U.S. DEPARTMENT OF  
**ENERGY**

Office of Science

- The release in 2013 will be a major release.
  - Geant4 version 10
- The highlight is its **multi-threading capability**.
  - Some interfaces need to be changed due to multi-threading
- It offers **two build options**.
  - Multi-threaded mode (including single thread)
  - Sequential mode
    - In case a user depends on thread-unsafe external libraries, he may install Geant4 in sequential mode.
- This is the first major release since 2008.
  - This is a rare opportunity for us to clean up obsolete code and make interface improvements.
  - GNUmake will be dropped.

- Geant4 version 10 offers so-called event-level parallelism.
  - Each thread is tasked for an event or a bunch of events.
- Every data that is updated at event-level frequency or shorter has to be thread local.
  - To avoid the race problem.
- Status of current prototype (G4MT-9.5.p01)
  - Being tested on a Intel® Xeon Phi coprocessor (MIC)
  - Initial tests show good scalability up to tens of concurrent threads



# Preliminary studies on TBB

---



- Intel Threading Building Block is a library for task-based multi-threading code. Some LHC experiments show their interest in the use of TBB in their frameworks.
- We have verified that the current G4MT prototype can be used in a TBB-based application where TBB-tasks are responsible for simulating events.
  - We didn't need to modify any G4MT class to adapt to TBB.
- A simple example has been prepared that uses TBB and G4MT.
- We will provide an example or two at the beta release of version 10 to demonstrate the way of integrating TBB and G4MT.
  - We will keep communicating with our users to polish our top-level interfaces.

- Geant4 9.6 released on Nov.30
  - Final release of version 9 series
- Dec 2012 / Jan 2013
  - Conversion of v9.6 to G4MT
    - Final G4MT prototype release in early Feb. 2013
  - Move G4MT v9.6 to main development trunk
    - All development toward version 10 should be made to this development trunk.
    - We maintain native v9.6 in SVN brunch for potential patch release.
- Feb-May 2013
  - Migration of examples and tests
  - Massive tests for both computing performance and physics performance
- June 2013
  - Beta release : all the major changes related to multi-threading should be included
  - If necessary, more than one beta-releases may be made.
- Dec 2013
  - Public major release of Geant4 version 10

- Obsolete classes / methods
  - All classes / methods to be removed have **warning message** in v9.6.  
“This class becomes obsolete and will be removed at the next release.”
  - We are currently removing them from the SVN main development trunk **immediately**, so that they won't be converted into G4MT.
- Changes caused by / related with multi-threading
  - Finalize major changes by February 2013 before migration of examples / tests begins.
  - Reference tags should be made available to testers.
- Changes independent to multi-threading
  - Given it's a major release, we may have some other interface changes. Some come with the beta release, some come after. We make sure they run in multi-threading mode.

Note: In addition to the massive tests in multi-threaded mode, Beta release should also have reasonable number of already-migrated examples to demonstrate the ideas of multi-threading.

- After the beta release (or even after the first reference tag), we invite feedbacks from our customers.
  - Interfaces visible to users would be iterated.
  - Hoping that iteration is just adding interfaces rather than changing them.
  - If necessary, more than one beta-releases may be made.
- All examples we release with version 10 will migrate to all interface changes including multi-threading.
- Documents also need to be updated accordingly.
- Staging???
  - As usual, new features / classes may be added at any minor release as long as they won't cause user's migration. Thus any functionalities, which we currently have but we cannot catch up necessary interface changes or assuring thread safety, may be staged as long as we release base interfaces with version 10.