CnC-Scala

CnC-Scala

The CnC-Scala system under development in the Habanero project at Rice University builds on past work on the Intel Concurrent Collections (CnC) and Habanero CnC projects.

A short introduction to the Concurrent Collections model can be found in the following article:

- The Concurrent Collections Programming Model. Michael G. Burke, Kathleen Knobe, Ryan Newton, Vivek Sarkar. Technical Report TR 10-12, Department of Computer Science, Rice University, December 2010. To appear as a book chapter in Encyclopedia of Parallel Computing, David Padua (Ed.), Springer Verlag, 2011.

CnC-Scala supports two execution modes:

1. Continuation based: if an item is not available, the Step is suspended and the continuation resumed when the item becomes available. This mode uses the Scala continuation plugin's support for delimited continuations. This allows the user to write only the step execution code and avoid defining extra methods that explicitly declare data dependences.
2. Data-Driven Future based: User step explicitly declares it data dependences in advance before the Step.compute() is called. As such the items are always available and there is no need for continuations. However, this limits the Step dependences to be data independent.

As of release 0.1.2 these execution modes have been combined into a single runtime and can be used together in computation steps.

To get started with CnC-Scala, see the Installation instructions. Once successfully installed, please refer to the examples page on how to write CnC-Scala programs. Here are some slides presenting CnC-Scala. A YouTube video demonstrating how to write CnC-Scala programs is also available. CnC-Scala was also presented at Scala Days 2012 [slides].

For questions or comments please contact Shams Imam (shams[AT]rice[DOT]edu)

Habanero Home

Research Projects

Publications

Target Applications

Multicore Platforms

People

Related Links

- CnC-Python
- Habanero-Scala