

発表概要

分散プログラミング言語 X10 上の耐故障性を備えたアプリケーション

河内谷 清久仁^{1,a)}

2014年3月17日発表

X10 は言語仕様として分散処理を記述できるプログラミング言語である。X10 アプリケーションは複数の「プレース」(計算ノード)を用いて実行可能で、「at」文によりプレースの移動を行うことで分散処理を行うことができる。しかし従来の X10 では、プレースを構成する計算ノードが1つでも故障すると、X10 アプリケーションの処理全体が終了してしまっていた。この問題を回避するため我々は、計算ノードが故障した場合にアプリケーションにそれを例外 (DeadPlaceException) として通知し、残りのノードで実行を継続することが可能な「Resilient X10」という言語拡張を行っている。本発表ではこの Resilient X10 の機能を用いて耐故障性のある分散アプリケーションをどのように構成するかについて述べる。故障が起きた場合のアプリケーションの対応として、(a) 無視して正常ノードの結果だけを使う、(b) 残りのノードに作業を再分配する、(c) 定期スナップショットから実行を再開する、の3手法を紹介する。また、ライブラリとして、X10 がもともと備えている分散配列 (DistArray) への耐故障性とスナップショット機構の追加についても述べる。耐故障性を追加したアプリケーションやライブラリの、ベースとなるコードからの変更は非常にわずかである。またどれも、従来の X10 上でも (ノードが故障しない限りは) 実行可能である。耐故障性追加の実行性能への影響についても評価を示す。

Fault-tolerant Applications on a Distributed Programming Language X10

KIYOKUNI KAWACHIYA^{1,a)}

Presented: March 17, 2014

X10 is a programming language which supports distributed computing in itself. X10 applications can run over multiple “places” (computation nodes), and perform distributed computing by changing the execution place using “at” statements. However, in a conventional X10 environment, when a node which consists of the place(s) fails, the entire processing of X10 application is aborted. To solve this problem, we have been extending X10 as a “Resilient X10” where the node failure is reported as a DeadPlaceException and the execution can continue using remaining nodes. In this presentation, we show how to construct fault-tolerant distributed applications over the Resilient X10 functions. Three typical methods are introduced to handle the node failure — (a) just ignore it and use results from remaining nodes, (b) re-assign the work to remaining nodes, or (c) restart the computation from a periodic snapshot. We also show a fault-tolerant extension of an existing distributed X10 library DistArray. These modifications to add fault tolerance were very small, and the modified codes can still run on normal X10 as far as node failure does not happen. Impacts to execution performance by the modifications are also shown.

¹ 日本アイ・ビー・エム (株) 東京基礎研究所
IBM Research - Tokyo, Koto, Tokyo 135-8511, Japan

^{a)} kawatiya@jp.ibm.com