

Title	分散環境におけるフォールトレラントソフトウェアの構成法に関する研究
Author(s)	伊関, 浩
Citation	
Issue Date	1997-03
Type	Thesis or Dissertation
Text version	author
URL	<a href="http://hdl.handle.net/10119/1014">http://hdl.handle.net/10119/1014</a>
Rights	
Description	Supervisor:片山 卓也, 情報科学研究科, 修士

# 分散環境におけるフォールトトレラント ソフトウェアの構成法に関する研究

伊関 浩

北陸先端科学技術大学院大学 情報科学研究科

1997年2月14日

キーワード: フォールトトレランス、ソフトウェアフォールトトレランス, FTAG, CML.

コンピュータ利用の増加にともない、高い信頼性を持つコンピュータの必要性が増加している。システムの一部に誤りが発生した場合、その誤りがシステム全体に波及しないような機構が必要とされる。このような、システム内に誤りが存在しても、システムのサービスを続けるための機構をフォールトトレランスと呼ぶ。フォールトトレランスを実現する手法として、ハードウェア的なものとソフトウェア的なものの二つの手法が存在する。本研究では、ソフトウェアによってフォールトトレランスを実現する、フォールトトレラントソフトウェアを研究対象とする。

これまでは、フォールトトレラントソフトウェアを構成するための手法は、手続き型による命令的な計算モデルに基づいていた。しかし、関数的な手法によってフォールトトレラントソフトウェアを構成した場合、手続き的なモデルと比較して、過去の計算状態の保存に複雑な処理が必要なく、障害の原因究明や障害からの回復等の処理が用意に行えるという利点がある。

FTAG(Fault-Tolerant Attribute Grammars) モデルは属性文法に基礎をおく階層的関数型計算モデル HFP にフォールトトレランスの機構を拡張したモデルである。FTAG では、ハードウェア障害に対処するチェックポイントや、ソフトウェア障害に対処する N-バージョンプログラミングやリカバリブロックといった技法が容易に実現できる。しかし、FTAG モデルは理論のみであるため、これを実験的に確かめる必要がある。

本研究の目的は、共有メモリを持たない疎結合のマルチプロセッサシステム上に FTAG を実装することにより、関数型モデルのフォールトトレランスにおける利点を実証すると共に、分散環境におけるフォールトトレラントソフトウェアの構成方法を与えることである。

FTAG モデルでは、全ての計算はモジュールと呼ばれる数学的関数の集合によって記述される。このモジュールの計算順序は、属性値の依存関係のみによって決定され、依存関係を持たないモジュールは並列に実行させることが可能である。よって、実装に用いる言語は、並行計算の仕組みを持っていることが望ましい。ここではこの条件を満足する言語として CML(Concurrent ML) を選んだ。

CML は関数型言語である SML(Standard ML) 上で並列プログラミングをするためのシステムである。CML のプログラムは並行して動くスレッドの集合として表現され、スレッド同士はチャンネルを通じてメッセージの通信を行う。スレッドやチャンネルは動的に作成可能であり、この性質は FTAG モデルによるモジュールと属性値の流れを表現するために役に立つ。

FTAG のプログラムリストは以下の手順で CML の環境上で動作が可能となる。

- (1).FTAG のプログラムをフォールトトレランスに関する処理を含む CML のプログラムに変換する。このプログラムが実際にアプリケーションの計算をする部分である。
- (2). プログラムが実行されたときの計算木の管理をする ノードマネージャと、再実行や複製の際に必要なとされる属性値を保存するオブジェクトベースの管理を行うワークスペースマネージャを作成する。
- (3). この二つのマネージャと、先に変換した CML のプログラムを合成することによって、CML インタプリタ上で動作するプログラムを生成する。

上で述べた ノードマネージャ、ワークスペースマネージャ、そしてアプリケーションの計算部の三つをコンポーネントと呼ぶ。ノードマネージャは、アプリケーション実行部が作る計算木の管理を行う。計算木の形を記憶して、誤りを検出して再実行が行われたときにシステムに矛盾がないように計算木を整形し、計算を再開させる。ワークスペースマネージャは、再実行や複製が起こったときにシステムの状態の回復に必要な属性値を保存管理する。アプリケーション実行部は、FTAG のプログラムを CML のプログラムに変換したもので、実際に計算を行うコンポーネントである。

モジュールをスレッドに分割し、チャンネルを利用することによって各ノードが並列に動作する。この為このモデルは分散環境に実現することが可能である。上に述べたアーキテクチャに基づいて、FTAG モデルの分散計算機上への実装の為の方法論を提唱した。そしてネームサーバシステムについての実験を行い、FTAG モデルの有用性、分散環境への適正について実験によって確かめることができた。