3P-1

仮想マシンを用いた分散システムの耐故障性評価環境の検討

神林　亮†　　佐藤　三久‡

†筑波大学第三学群情報学類　　‡筑波大学大学院システム情報工学研究科

1　はじめに

今日では, 多くのシステムが処理能力向上を目的に分散システムとして構築されているが, それら全てに充分な耐故障性があるわけではない. なぜなら, その向上を行う上で以下の２つの問題が存在するからである.

- 分散環境での故障の再現
  分散システムがどの程度の耐故障性を持っているかを検証する場合, 故障を擬似的に発生させた上でシステムが正しく動作するかどうかテストしなければならないが, 実環境に近いような故障発生シナリオでのテストは多大な労力を要するため行われない.
- 耐故障性の評価方法
  システム特性は, 定量的な指標で比較されることが望ましいが, 耐故障性を測る定量的な指標は現状では存在しない. そのため, 耐故障性の向上が行われても, その優劣を効率的かつ適切に論ずることはできない.

これらの問題に対して様々な研究が行われている [1][2] が, 我々はより簡便に耐故障性を検証, また定量的な評価を行うことのできるシステムについて検討し, 試作を行う.

2　仮想マシンを用いたフォルトインジェクタ

耐故障性の検証と評価を行うために, 擬似的に故障を発生させ, それに対するシステムの振る舞いを観察するという手法を採用する.

我々はそれを実現する方法として, 仮想マシン技術を用いたフォルトインジェクタ Fault VM を提案する. フォルトインジェクタとは, 評価対象のシステムに対して擬似的な故障を注入する機構である.

仮想マシン上で耐故障性を評価したいシステムを動作させ, そこにソフトウェア的に擬似的な故障を注入するという方法をとることで, 以下が可能となる.

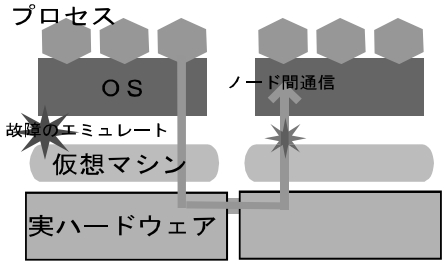

図 1: 概念図

- 評価環境が故障をエミュレートすることで, システムのコードに手を加えずに故障を注入
- 評価対象のシステムからは実際にハードウェアが故障したように見える特性により, 実環境に限りなく近い環境を実現
- 障害の起きた仮想マシン以外は動作し続ける特性による, 故障後のシステムの観察

概念図を図 1 に示す.

3　システムの概要と実装

仮想マシン Xen をベースとし, 以下の機能を提供する.

- シナリオベースでの故障の注入
- メモリ, ネットワークなどの各種 IO, レジスタ, 割り込み, 電源の異常などの多様な故障の注入
- 複数ノードにまたがるような故障発生パターン
- 種別ごとにベンチマークを行い, 単一の数値により耐故障性の程度を出力

現時点ではプロトタイプの実装が完了しており, メモリ故障の注入が可能である.

以下でその実装について述べる.

Xen 内で, 各々の仮想的なマシンは Domain と呼ばれ, 様々な特権を持つ Domain0 と, その他の DomainU が存在する. Fault VM では, 評価作業用のシステムが動作する Domain0 から, 評価対象のシステムが動作する DomainU に故障を注入する.

メモリ故障の注入では, Domain0 から DomainU の仮想物理メモリをマップし, その領域に, 特定のルールで書き込みを行うことで擬似的なメモリ故障としている.

メモリのマッピングには Xen が持つドメイン管理用のユーザレベルライブラリ libxc を利用している*.

Design and Implementation of Fault Tolerance Evaluation Environment for Distributed System using Virtual Machine

†Ryo Kanbayashi　‡Mitsuhisa Sato

†College of Information Sciences, Third Cluster of Colleges, University of Tsukuba

‡Graduate School of Systems and Information Engineering, University of Tsukuba

*正確にはサードパティが提供している libxc のメモリ操作 API のラッパライブラリ Xenaccess を改変し使用している

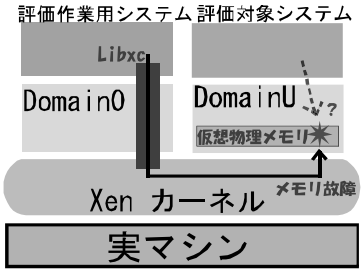

図 2: メモリ故障の注入

Xen では, OS を Xen 上で効率的に動作させるための修正を加えた上で実行する準仮想化と, ハードウェアによる仮想化支援*により, 修正を行わずに動作させる完全仮想化の 2 つの仮想化方法を選択できる. しかし, 準仮想化では OS のページテーブル管理やハードウェアアクセスの方法が変更されるため, 故障時の挙動が通常と異なったものになると予想された. このため, Fault VM では当初, 評価対象のシステムを準仮想化で動作させていたが, 完全仮想化で動作させるよう変更した. これにより, Fault VM を動作させるマシンにハードウェア仮想化支援が必須となる制約が生まれたが, Windows などの準仮想化に対応しない OS の評価が可能となった.

4 分散ソフトウエアの信頼性の評価

実際に, Fault VM 上でソフトウェアの耐故障性の評価を行い, 評価環境の有用性を確認する. この評価では現時点で利用可能なメモリ故障の注入のみを用いる. 耐故障性の評価を行うシステムとしては, 商業的な場面で多く運用されている, ウェブサーバの負荷分散と高可用性を目的とした HA(High Availability) サーバシステムを対象とする (図 3). 対象システムは Domain を複数用いて一台の物理マシン上に構築した†.

対象システムでは, 図中左部のロードバランサ‡が, バックエンドのウェブサーバへの HTTP リクエストをラウンドロビンでバランシングする. ロードバランサのうち lv0 が稼働系で通常の状態でのバランシングを行う. lv0 と lv1 はハートビート§により互いの生死の監視を行っており, lv0 が故障で動作不能になった時にのみ, 待機系の lv1 が代わりにバランシングを行う.

評価の方法は, 継続的なリクエストがクライアントから行われている状況で, ロードバランサが故障した場合の挙動を確認することである. 確認する対象は単位時間当たりに処理できたリクエスト数の変化と, 各サーバ上のログファイルである. 故障の内容は, 3 秒おきに 10

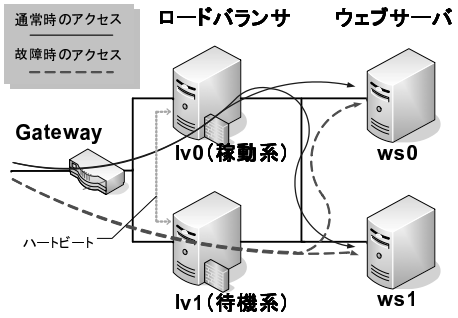

図 3: 構築した HA サーバシステムの構成

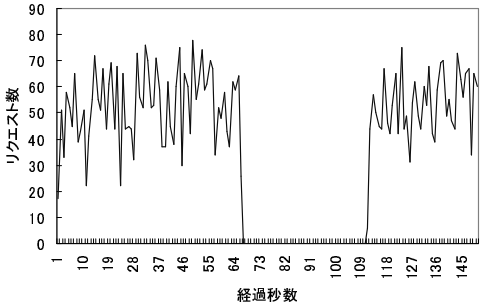

図 4: 処理できたリクエスト数の変化

ビットずつのメモリ故障¶を起こし, ロードバランサを動作不能にすることにした.

メモリ故障を継続的に注入した結果, 実験開始から 66 秒後, 220 ビットのメモリ故障が注入された段階でロードバランサが動作不能となり lv0 と lv1 の切り替えが発生した‖. これに対応し, リクエスト数の変化のグラフ (図 4) でも, 値が 0 となる区間が発生した. しかし, 45 秒後には復旧しており, 構築した HA サーバシステムは故障発生時でも正しく動作を継続することが可能であると確認できた.

5 おわりに

本稿では, 仮想マシン技術を用いた分散システム向けフォルトインジェクタ Fault VM のプロトタイプ実装について述べ, その有用性を示した. 今後は機能の拡張を進め, 定量的な評価の枠組みを検討する.

謝辞 本研究の一部は科学技術振興事業団戦略的基礎研究推進事業（CREST) の支援による.

参考文献

[1] Timothy K. Tsai. Ravishankar K. Iyer. Doug Jewitt. An Approach towards Benchmarking of Fault-Tolerant Commercial Systems, FTCS-26, October 1996.

[2] S. Potyra. et al. Evaluation Fault-Tolerant System Designs using FAUmachine, EETS'07, September 4, 2007.

*Intel の VT(Virtualization Technology) や AMD の AMD-V などがあり, プロセッサに実装される.

†プロトタイプでは複数台のマシンを用いての検証は未実装であったため

‡LVS(Linux Virtual Server) による

§HA クラスタ構築ソフトウェア Heartbeat による

¶ランダムなアドレスに起きるビット反転

‖lv1 上の Heartbeat のログによる