

第1回 ミッションクリティカル システム技術概論

にしむら とおる よこざき だいごろう おくむら やすひろ ゆ ら しゅんすけ
西村 徹 / 横関 大子郎 / 奥村 恭弘 / 由良 俊介
NTT情報流通プラットフォーム研究所

ミッションクリティカルシステムに求められるソフトウェア技術を、4回に分けて分かりやすく解説します。第1回では技術要件の解説を行い、第2回と第3回では、それぞれ構成技術であるトランザクション技術と高可用性技術を概説し、最終回では現状のオープンソース製品の適用性の展望を述べます。

ミッションクリティカルシステムの基本要件

航空機や原子炉の制御システム、電話システム、銀行システムなど、その停止や機能低下、誤作動が、人命の喪失や利用者、企業の損失に直結する、言い換えると、継続的な正常稼動が最重要視されるシステムをミッションクリティカルシステムと呼びます。

ミッションクリティカルシステムは、次に示す品質指標と性能指標を、高いレベルで両立させることが求められます。

■ 高品質要件

システムに対する品質指標としては、RAS (Reliability, Availability, Serviceability) が広く利用されており、次のように定義されます。

Reliability (信頼性) : システムが故障せずに稼動し続けられ

る時間に関する性質で、システム起動後、故障が発生するまでの平均時間 (MTTF: Mean Time To Failure) ^{*1} という指標で表現します。

Serviceability (保守性) : 故障発生時の修復の迅速さに関する性質で、平均修復時間 (MTTR: Mean Time To Repair) という指標で表現します。

Availability (可用性) : システムが稼動している割合に関する性質で、稼動率という指標で表現され、 $MTTF / (MTTF + MTTR)$ と計算します。

ミッションクリティカルシステムは、「24時間365日」の連続稼動が理想的です。しかし、実際に1年間無停止で稼動することは非常に難しく、「稼動率99.999%を満たすこと^{*2}」というように、可用性の尺度で品質を求めることが一般的です。高レベルの可用性は、信頼性および保守性の向上によって達成されます。

■ 高性能要件

性能に関しては、応答時間 (利用者がアクションを起してからその反応が振り返るまでの時間) と処理能力 (単位時間に実行できる処理総数) に関する指標があります。例えば、パイ

ロットが舵を操作してから1分後に羽根が動く応答時間では乗客は安心してその飛行機には乗っていられないでしょうし、振込処理が1日当たり100件をこなす能力しかない銀行システムでは日々発生する要求を処理できません。

ミッションクリティカルシステムの構成要素

私たちが普段利用している銀行で稼動しているシステムを例に、ミッションクリティカルシステムの構成について解説します。もっとも身近なのはATM (Automatic Teller Machine) でしょう。ATMのように、システムと利用者の境界に位置する装置を業務端末と呼びます。ATMから入力された利用者の要求は、ネットワーク転送され、業務サーバ上のソフトウェアによって処理されます。またシステムは、サブシステムによって階層的に構成されます。システムは、業務サーバ、業務端末、ネットワークによって構成され、業務サーバは、CPU (Central

*1 システムは永続的に動作するという前提のもと、平均故障間隔 (MTBF: Mean Time Between Failure) という指標もあり、 $MTBF = MTTF + MTTR$ と定義されます。近年は、MTBFを本文のMTTFの意味で用いる場合もあります。

*2 1年に5分故障している計算になります。

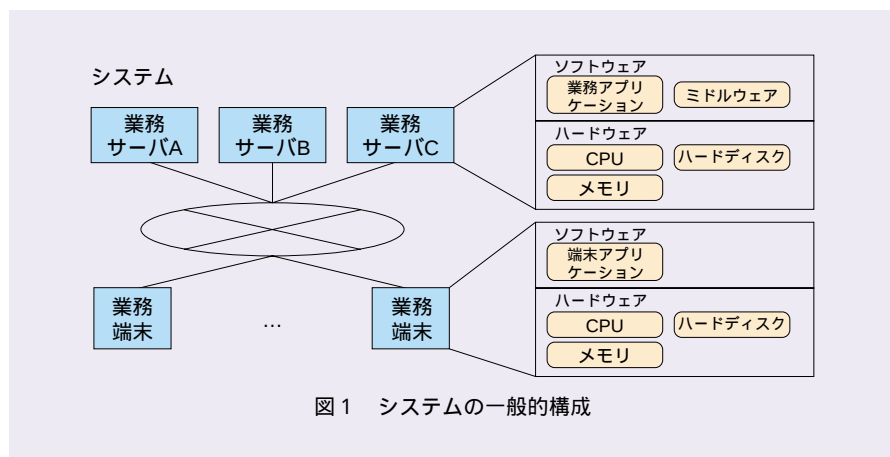


図1 システムの一般的構成

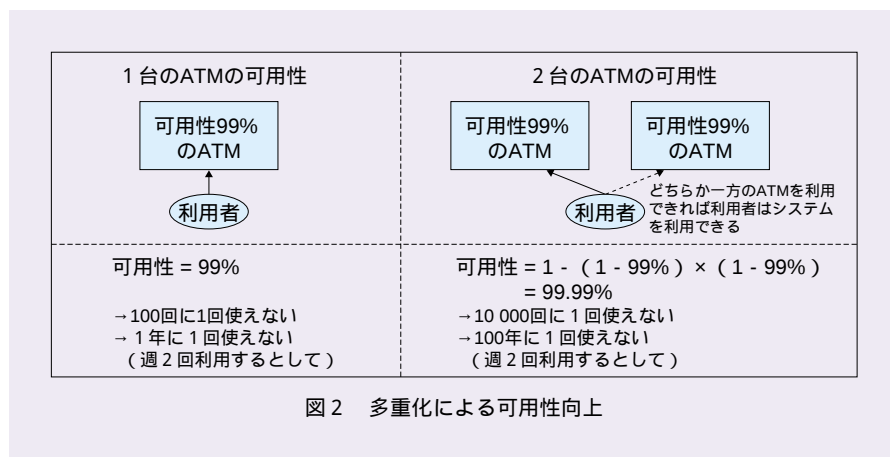


図2 多重化による可用性向上

Processing Unit)、ハードディスク、メモリなどのハードウェアや業務アプリケーション、ミドルウェアなどのソフトウェアによって構成されます(図1)。

可用性向上の方法論

システムの可用性を向上するためには、システム自体の信頼性・保守性を向上する方法のほかに、サブシステムを多重化する方法があります。

サブシステムの多重化によって、可用性を向上する方法を図2に示します。

多重化されたサブシステムは、すべて稼働している状態とし、利用者はそのうちの1つを選択的に利用します。万が一、選択したサブシステムが故障している場合は、もう1つのサブシステムを利用します。多重化されたシステム

の稼働率は、サブシステム1台の稼働率を99%と仮定した場合、99.99%となります。可用性を向上する目的で多重化されたシステムの例としては、DNS(Domain Name System)におけるルートDNSサーバなどがあります。サブシステムの多重化による可用性の向上技術は、クラスタリングと呼ばれます。

信頼性向上技術

システムの信頼性向上は、それ自身の信頼性を向上する方法(単体の信頼性向上)と、同等の機能を冗長に持たせる方法(冗長化による信頼性向上)があります。複数のサブシステムから構成されるシステムは、サブシステムの信頼性の向上により、それ自身の信頼性が向上します。したがって、以下では単体と冗長化の2つの観点

で、サブシステムの代表的な例であるハードウェアとソフトウェアにおける信頼性向上の方法を示します。

■ 単体の信頼性向上

ハードウェア単体の信頼性を向上する技術としては、ハードウェア製造段階での品質管理や、稼働条件を考慮したハードウェア設計などがあります。主電源の一時的停止に備えた無停電電源装置(UPS: Uninterruptible Power Supply)が組み込まれた電源システムや、誤り訂正機能を持つメモリはその一例です。

一方、ソフトウェア単体の信頼性を向上する技術としては、トランザクション処理があります。この技術は、例えばATMを利用してAさんの口座からBさんの口座に10万円の振込みを依頼したときに、データベース上の2人の口座台帳に確実に記録するなど、データの整合が取れた状態を維持する方法で、データベース管理システムやアプリケーションサーバに実装されます。これらの機能を利用した業務アプリケーションは、データの不整合がなくなるため、結果として信頼性が向上します。

■ 冗長化による信頼性向上

ハードウェアの冗長化による信頼性向上の例としては、専用回線のバックアップとしてダイヤルアップのISDN回線を用意する方法などがあります。

一方、ソフトウェアの冗長化については、Nバージョンプログラムがあります。この技術は、同一の目的を実現する複数のプログラムの多数決の結果をシステムの結果とるように構成することにより、ソフトウェアの不具合を隠蔽する技術で、真に高い信頼性を求められる分野で利用されます。

保守性向上技術

システムの保守性の向上は、それ自身の保守性を向上させる方法と、サブ

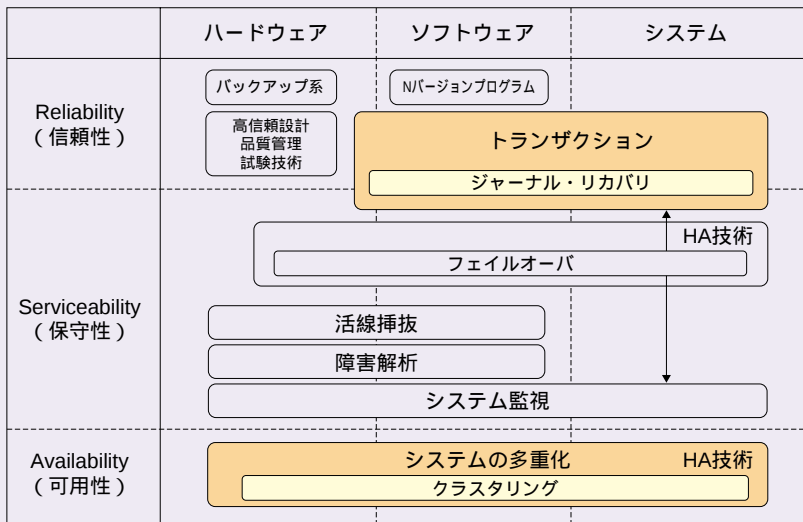


図3 ミッションクリティカルシステムの構成技術

システムの多重化による方法があります。前者は、故障の原因となる構成要素の修復（修繕や取替え）を迅速に行う方法で、ハードディスクやメモリなどのハードウェアの活線挿抜、データベースのリカバリ処理とそれに備えるジャーナル^{*3}取得処理が一例です。後者は、システムを構成するサーバを多重化し、稼動しているサーバ（現用系）の故障発生をシステム監視機能などで検出し、多重化したサーバ（待機系）に切替えて運用を継続する方法で、一般にフェイルオーバと呼ばれます。前述のクラスタリングと比較して、故障が発生して初めて多重化したサーバの利用を開始する点が異なりますが、2台以上のサーバでの同時の運用が難しいデータベース製品などで利用されます。

性能向上技術

可用性・保守性を向上させる方法として紹介した多重化は、性能向上

^{*3} ジャーナル：データベースの実更新に先立ち記録される変更データのこと。リカバリ処理において、この変更データを順次読み込み反映させることで、データベースを復元します。

の目的でも利用されます。同一の機能を提供するサーバをN台配置するクラスタリングは、クラスタ間のオーバヘッドがない場合にN倍の処理能力を発揮します。通常はオーバヘッドが0になることはまれで、Nの数に依存してオーバヘッドが増大するソフトウェアもありますので、クラスタリングを利用する際には、性能試験などの実施により、要求条件を満たしていることを確認することが重要です。

前述したミッションクリティカルシステムの構成技術を図3にまとめます。このうち、トランザクション技術と多重化されたサーバ群を取り扱うHA（High Availability）技術は、ミッションクリティカルシステムの必須要件である可用性向上のための構成技術として非常に重要です。第2回と第3回に分けて紹介します。

ミッションクリティカルシステムとオープンソースソフトウェア

オープンソースソフトウェア（OSS: Open Source Software）をミッションクリティカル分野で利用する流れが活発になってきています。海外では、ミュンヘン市が業務に利用するコン

ピュータ約14,000台を、OSSのLinuxベースに移行する計画を進めています。日本においては、情報化白書2004⁽¹⁾の「自治体のオープンソース採用」の節（4.3節）において、『2000年代に入り、地方自治体の間では国や特定ベンダーへの依存体質を改め、地域が独自に電子自治体プロジェクトを推進しようという動きが目立っている。...（中略）...2003年以降、この傾向はさらに強まり、自治体は積極的にオープンソースの採用や普及にまでも力を入れるようになった。』と紹介され、また電話システムにLinuxを適用するためのキャリアグレードLinux仕様が、OSDL（Open Source Development Lab）において定義されています。そこで最終回では、現状のオープンソフトウェア製品のミッションクリティカル分野への適用性について詳しく紹介します。

■参考文献

- (1) 日本情報処理開発協会編：“情報化白書2004,” コンピュータ・エージ社, 2004.



（左から）西村 徹/ 横関 大子郎/
奥村 恭弘/ 由良 俊介

インターネットの世界においても、止まらないシステムの構築は今後ますます重要視されるでしょう。今後3回の連載で、構成技術や市中のオープンソース製品について分かりやすく解説していきます。ご期待ください。

問い合わせ先

NTT情報流通プラットフォーム研究所
第二推進プロジェクト
TEL 0422-59-3524
FAX 0422-59-2699
E-mail nishimura.toru@lab.ntt.co.jp