

## トヨタ自動車における分散システムの運用管理事例

尾崎 貴司      西野 正行

トヨタ自動車(株) システム企画部

トヨタ自動車では近年の分散システムの急速な普及に伴ない、社内に設置された分散システムの安定運用と障害の早期発見・対応が取れる分散運用自動化システムを構築した。本稿では分散システムの運用事例として、今回構築した分散運用自動化システムの紹介をする。

## The case of Management System for some Distributed Systems

Masayuki Nishino      Takashi Osaki

System Planning Div., TOYOTA MOTOR CORPORATION

TOYOTA MOTOR CORPORATION introduce an Operation-less Management System for some Distributed Systems. This paper discuss those specifications of the Operation-less Management System, and the management policy from the users point of view.

## 1. はじめに

筆者らは従来の基幹系ホスト/メインフレームの運用管理を行っていたが近年の分散システムの普及従来ホスト業務のクライアント/サーバー型システムへのダウンサイジングの進展に伴い新たに分散システムの運用管理を行うようになった。これに伴い、分散システムの運用管理にあたって必要な機能の整理・洗出しを行い、分散運用自動化システムを構築した本稿ではこのシステムの機能を整理し、筆者らのシステム構築にあたっての考え方を紹介する。

## 2. 分散運用自動化システム

クライアント/サーバー型で導入された分散システムの運用管理について原則はユーザの主管部署にて運用管理を行うが全社共有の分散システムについては従来のホスト/メインフレームでの運用管理経験を活かすべく筆者らの部署にて運用管理することとした。

そこで全社共有の分散サーバーを集中管理する場合でもあくまでもユーザの利便性を損なわないように、次の点を重視した。

- (1) 人手のかかるオペレーションの廃止  
(ツールで計画設定、作業の自動化推進)
- (2) システムの稼働設定管理はユーザ実施
- (3) システム維持管理面からできるだけ市販ツールを活用する

この考え方にに基づき市販の各種運用管理ツールを紙上比較のほか、実物での機能・性能評価を行い分散運用自動化システムにて利用するか選定した。各ツールの評価の一例を表2-1に示す。

表2-1 運用管理ツールの評価例

評価期間：95年8月～96年7月

	ツール U	ツール C	ツール S	ツール E
・HW機能仕様	△	○	○	○
・システム構成(管理方法)	ピア・ツー・ピア	クライアント・サーバ	クライアント・サーバ	クライアント・サーバ
・対応サーバの種類	UNIX,NT個別に	UNIX,NT一括	UNIX,NT一括	UNIX,NT一括
・社内環境適合度	HW投資が大	投資の抑制可	投資の抑制可	投資の抑制可
・SW基本機能仕様	△	○	△	○
	(相当な工数必要)		(ジョブ管理時間)	
・操作性(GUI)	Host経験者向き	初心者向き	初心者向き	初心者向き
・他のツールとの連携	可能(難)	可能(実績有り)	可能(やや難)	可能(やや難)
・自動スケジュール機能	○	○	△	△
	大規模/汎用に変	大規模/汎用向き		
・安定性	実績あり	実績あり		
・拡張性	不安有り	階層管理可		

これらの評価結果を基に分散運用自動化システムとして図2-1に示すようなH/W構成で複数の分散システムのサーバーを自動オペレーションで運用管理できるように構築した。

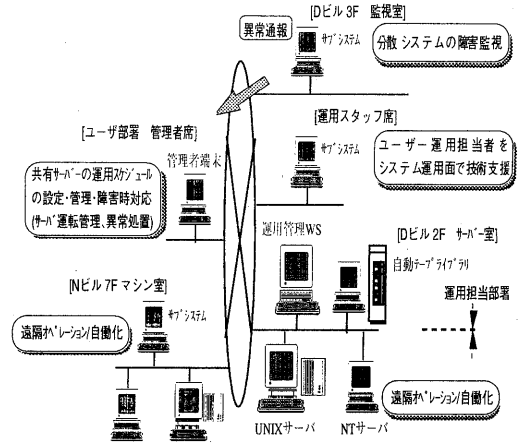


図2-1 分散運用自動化システム

今回構築したシステムは単一の市販されている運用管理ツールでは筆者らの期待している機能・性能が得られなかったため、複数の運用管理ツールを組み合わせられた形で構築している特に現在普及しているUNIX系サーバとWindows NT系サーバのそれぞれを1台のWorkstationにて管理できるようにシステムを構築できるように構成を考えてある。

それぞれの機能としてどのように活用しているか、以下に機能ごとにまとめて説明する。

## 3. バッチ処理スケジュール管理機能

分散システムも従来のホスト/メインフレームでの処理形態の継承や基幹系システムのダウンサイジングの場合どうしても今までのバッチ系処理が残ってしまう特にシステム間で処理の前後関係を重視する連携を取っている場合それぞれの分散システムの処理をトータル的に管理する必要があるそこで筆者らは今回のシステムで図3-1のようなシステム間のスケジュール管理機能を構築した。

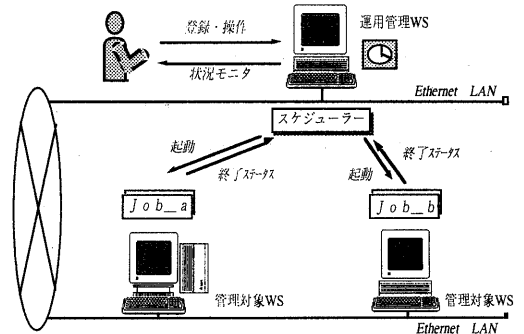


図3-1 バッチ処理スケジュール管理の概要

また、分散システムの中では従来のホスト/メインフレームや他の分散システムのサーバと D/B の共有やファイルの送受信による情報共有を行っている場合、連携をとる相互間のシステムに対して同期を取らなくてはならない。

そこで、筆者らはファイルの到着・生成をトリガにして後続の処理を起動できるようにシステム構築を行った。

ファイル受信に対するトリガの検知については図 3-2 に示すような方式を取った。

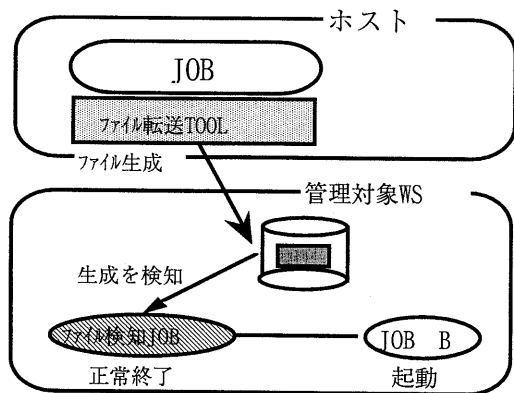


図 3-2 ファイル待ちトリガ検知方式

#### 4. 稼働管理機能

さまざまな全社・部門共有サーバーを一括管理する場合、それらの分散システムに障害が発生した場合（自動運用のオペレーションに失敗した場合）いかに的確に処置できるかが大きな課題となってくる。従って、システム開発者やシステム運用管理者を直ちにコールする必要がある障害については担当者コールにすぐにつうちできるようにしている。

ただし、障害の中には処置を急がない物件もあるため、この分散運用自動化システムでは図 4-1 や図 4-2 に示すように 2 種類の障害通報ができるようにしてある。

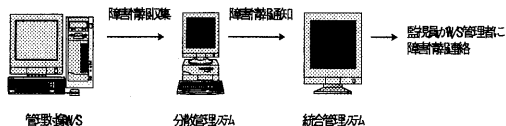


図 4-1 重要障害の通報方法

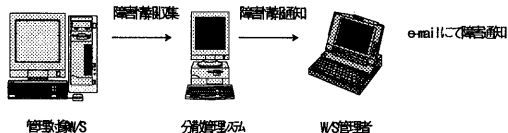


図 4-2 通常障害の通報方法

#### 5. 自動バックアップ取得機能

日常的な重要なオペレーションとして発生する D/B のバックアップやシステムバックアップについても、定常的な作業ながら人間によるヒューマンエラーを回避し、より確実なバックアップ/リカバリの保証をするために、分散運用自動化システムにてバックアップ取得ができるようにした。今回構築した自動バックアップシステムの構成は図 5-1 に示し、バックアップ取得の方法を図 5-2 に示す。

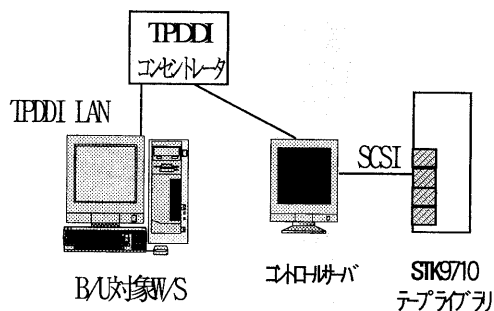


図 5-1 自動バックアップシステムの構成

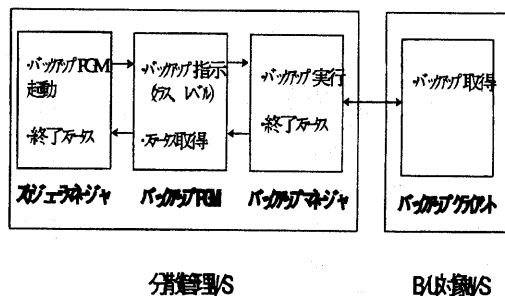


図 5-2 バックアップ取得の方法

#### 6. 自動電源 On/Off 機能

また、分散システムとして起動・停止が発生するが、該当する分散システムが広範囲で利用される場合、その分散システムを起動・停止できる時間帯が限られて、実際の運用管理担当者に対し多大な負荷をかけてしまう事になる。従って、今回構築した分散運用自動化システムでは特に重要なサーバーの電源の入り/切りを自動で行えるようにした。その例として、UNIX 系のワークステーションの停止の例を図 6-1 に示す。

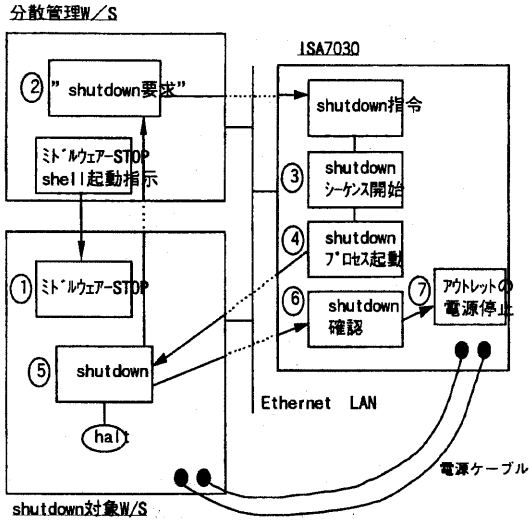


図6-1 UNIX サーバの停止方法の例

## 7. まとめ

今回、「本来分散システムを誰が、どのように運用管理すべきか」関係者で討議しながら、そのオペレーション作業を廃止し、ユーザの稼働要望に対して柔軟に対応できるシステム環境を整備するため、分散運用自動化システムを構築することができた。今後もさらにこのような分散システムの運用のあり方について検討を重ねながら、より信頼性が高く、より柔軟性の高いシステム構築を進めていきたい。

最後に今回のシステム構築にあたって、多彩な経験と優れた技術サポートをしていただいた株式会社CSK オープンシステム推進室の笹辺和義室長に感謝いたします。

### [参考文献]

西野 正行、尾崎 貴司、佐藤 敏郎；“企業における分散システムの運用管理における一考察”，情報処理学会 研究報告 96-DSM-5 (1997.5)