

航空管制用 RDPS

Radar Data Processing System for Air Traffic Control

田口 実男 山田 達朗 赤石 高敏

■ TAGUCHI Jitsuo ■ YAMADA Tatsuro ■ AKAISHI Takatoshi

最近の情報処理システムは、機能要求の高度化やコンピュータアーキテクチャの多様化など、システムを取り巻く環境の変化に柔軟に対応することが求められている。東芝はこれらの要求に応えるため、航空管制用レーダ情報処理装置 (RDPS : Radar Data Processing System) 用ソフトウェアの開発を行った。これまで、大型コンピュータを中心とした集中処理方式で確立されていたソフトウェアを、計算機やオペレーティングシステム (OS) の変化に柔軟に対応できるブロックアンドビルド基調の分散処理型ソフトウェアとして構築した。RDPS を取り巻く変化のなかで、今回開発したソフトウェアは、システムの構築を柔軟かつ迅速に行い、タイムリーに製品を供給するうえで極めて有効な手段である。

Recent data processing systems (DPS) are expected to flexibly meet the demands for sophistication of functional requirements and diversification of computer architectures. In response to this trend, Toshiba has revamped its radar DPS (RDPS) for air traffic control in accordance with today's advanced computer and software design practices. While conventional RDPS used to incorporate centralized software processing on a large central computer complex, we have developed a new model of RDPS by applying our own methodology of software construction, implementing the block-and-build approach to software engineering on a distributed processing system so as to flexibly respond to changes in computer hardware and/or operating systems.

These technology bases are expected to be applicable to the development of similar DPS in the ever-evolving software engineering environment.

1 まえがき

航空管制用レーダ情報処理装置 (RDPS : Radar Data Processing System) は、レーダで捕らえた航空機の情報 (航空機の距離や方位など) と飛行計画と呼ばれる運航情報 (飛行経路や予定時刻など) の相関を取り、航空機の便名、対地速度、予測進路などの情報を航空管制官に提供する情報処理システムである。

RDPSでは、近年の航空交通量の増大に伴う処理能力の向上と、計算機やOSの急速な変化に対する柔軟な対応が求められ始めている。しかし、大型コンピュータを中枢に据え、集中処理のコンセプトで構築された従来のRDPSでは、その対応が非常に困難な状況にある。

東芝は、このような背景の下、システムパフォーマンスの増強と、計算機やOSの多様化に柔軟に対応できるブロックアンドビルド基調の分散処理型RDPSソフトウェアを開発した。ここでは、その概要を述べる。

2 従来 RDPS の問題点と開発課題

従来のRDPSは、大型コンピュータの処理能力を最大限に引き出せるように、集中処理のコンセプトの下に構築されている。集中処理を基調としたシステムは、情報資源を複数の機能間

で共有し、大きな情報を一括で処理することに適しているが、情報処理能力の向上や計算機及びOSの変化への柔軟な対応は不得意としている。数世代前の大型コンピュータでシステムを実現した際は、RDPSも集中処理のコンセプトで構築すべきものであったが、現在のパソコン (PC) やワークステーションといった計算機の飛躍的な処理能力向上を考えると、

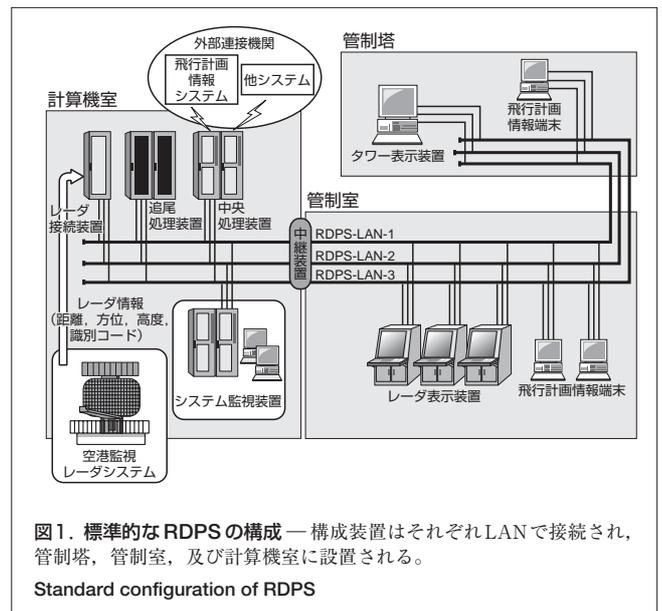


図1. 標準的なRDPSの構成 — 構成装置はそれぞれLANで接続され、管制塔、管制室、及び計算機室に設置される。

Standard configuration of RDPS

必ずしも最適な構造とはいえない。

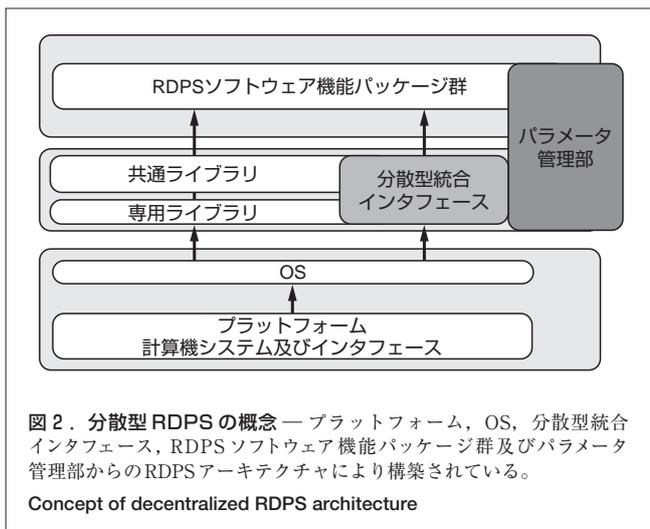
一方、分散処理を基調としたシステムは、小・中規模な情報を高速で処理することと、機能要求の高度化やコンピュータアーキテクチャの多様化などへの対応を得意としている。今日のRDPSのように、継続的に進歩していくシステムには最適なコンセプトといえる。このような情報処理システムを取り巻く環境の変化を受け、次のような課題に取り組み、分散処理型RDPSソフトウェアの開発を開始した。

- (1) 計算機やOSの変化に柔軟に対応する仕組み
- (2) システム拡張への柔軟な対応

今回開発を行った標準的なRDPSの構成例を図1に示すとともに、以下に、これらについて述べる。

3 計算機やOSの変化に柔軟に対応する仕組み

今回開発したRDPSソフトウェアは、図2に示すように、機能ごとにパッケージ化されたソフトウェア(以下、RDPSソフトウェア機能パッケージと呼ぶ)の集合で構成されている。



この構造は、計算機やOSの変化を分散型統合インタフェースで吸収することを特長としており、これによりマルチプラットフォームへの対応と上位互換を実現している。

分散型統合インタフェースは、専用ライブラリと共通ライブラリとから構成されている。専用ライブラリは、計算機やOSに依存して動作するソフトウェアから構成されており、計算機の変化に対しては、この専用ライブラリだけを修正することによって上位互換性を確保する。

共通ライブラリは、複数のアプリケーション(RDPSソフトウェア機能パッケージ群)で共通に使用されるソフトウェアで構成されており、計算機やOSのバージョンアップに対して、この共通ライブラリを修正することによって従来のRDPS

ソフトウェア機能パッケージをそのまま使用することが可能となる。

4 システム拡張への柔軟な対応

昨今の航空交通量の飛躍的な増大に伴い、システムに要求される機能や処理能力は、年々高度化されていく傾向にある。またその要求も、システムを所管する管制組織とその運用カテゴリーに応じて大きく変化することから、様々な運用要求に柔軟に対応できる仕組みを取り込まなければならない。これを実現するために、機能モジュール(以下、パッケージと呼ぶ)の分散・階層化と、それらのセミカタログ化を導入した。

まず、パッケージとして用意すべき“RDPSソフトウェアの機能”を抽出し、これらのパッケージが必要とする情報を定義することで、パッケージ相互の関係を明確化し機能の階層化を行った。

次に、ベンチマーク先、モデル・型名、納入地などの運用カテゴリーごとに必要とされる機能を一覧表にし、機能と運用形態ごとの必要性を明確化することで、それらを標準機能とオプション機能に区分した(表1)。ここで標準機能とは、ある製品モデルを構築するために必要とする最小限のパッケージ機能を示し、オプション機能とは、運用カテゴリーの違いによって標準機能に追加・選択できるパッケージ機能を示している。

表1. 標準機能とオプション機能
Standard functions and optional functions

項目	機能処理名称	運用カテゴリー				備考
		A	B	C	D	
標準機能	AAA 機能	○	○	○	○	
	BBB 機能	○	○	○	○	
	CCC 機能	○	○	○	○	
	DDD 機能	○	○	○	○	
	EEE 機能	○	○	○		
	FFF 機能	○	○	○		
オプション機能	XXX 機能	○	○		○	
	YYY 機能	○				
	ZZZ 機能	○				

最後に、階層化された標準・オプション機能と、計算機への機能配分を分析することで、機能レベルと実装レベルでの分散化を行った。

今後、新たなRDPSソフトウェアへの要求を実現する場合には、分散・階層化されたパッケージの組合せや、新たなパッケージの開発により柔軟に実現される。

4.1 分散・階層化

計算機システムの大型化と、それに伴うシステム要求の変化への対応の困難性といった従来の集中型システムの課題を解決し、更に機能拡張への柔軟な対応と従来以上の高信頼性と抗堪性を確保するために、機能レベルで分散・階層化したシステムとして構築することとした。RDPSソフトウェアの分散化及び階層化構造の概念を図3に示す。

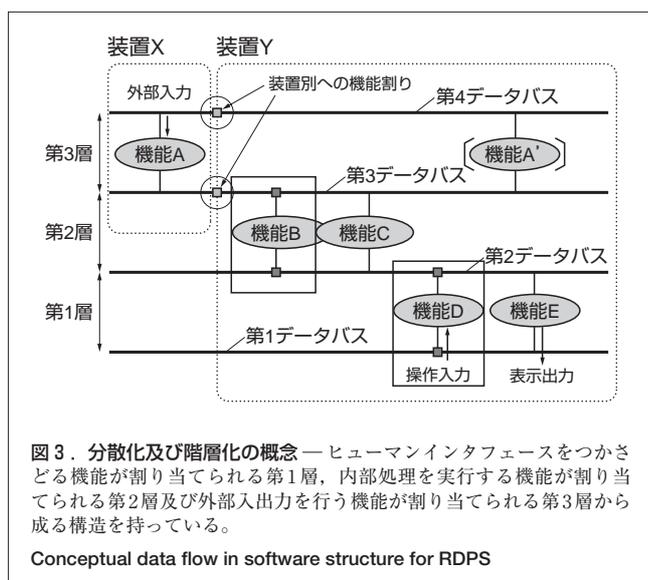


図3. 分散化及び階層化の概念 — ヒューマンインタフェースをつかさどる機能が割り当てられる第1層、内部処理を実行する機能が割り当てられる第2層及び外部入出力を行う機能が割り当てられる第3層から成る構造を持っている。

Conceptual data flow in software structure for RDPS

この構造の特長は、第1～第4データベースに流れる情報を、その入出力先で分類した点にある。例えば、第4データベースは外部システムから入出力されたデータが流れるバスであり、中間層にある各種機能で処理された後に、表示出力とのインタフェース情報の流れる第1データベースを介して出力される。このように、同一階層のバスに流れる情報が相手先に応じて分類されているため、あたかもハード基板を装置に挿抜するかのごとく柔軟に機能の追加・削除が行える構造となっている。

また実装レベルにおいては、計算機上で機能を配分しシステムを構築する必要があるため、追加・削除の可能性がある機能（オプション機能）については、データベースとの接続を論理的なコネクタによって実現している。これにより、システム導入後の機能拡張やシステムパフォーマンスの向上に対して柔軟な対応が可能となる。加えて、同一のソフトウェア構造で構築されているプラットフォームに対しては、必要な機能を柔軟に追加・削除することが可能となる。

また分散化と階層化により、ソフトウェア相互の独立性が高まり、ある機能に障害が発生した場合においても、その影響が他の機能に対し連鎖的な影響を及ぼす頻度が顕著に下がることにもつながった。加えてハードウェアの故障に対しても、類似装置に同等の機能を持たせ代替機材として

運用に供することが可能となり、抗堪性の高いシステムを構築することが可能となった。

4.2 分散型統合インタフェース

ここまで述べてきた分散・階層化されたソフトウェアを実現する内部の仕組みとして、機能間及び装置間のインタフェース、すなわち分散型統合インタフェースが開発の鍵となった。

従来の集中型のシステムでは、このようなインタフェースは不要であったが、分散処理でシステムを構築するには避けては通れない課題である。

パッケージがこのインタフェースを意識せずに動作できるように、インタフェース機能を一つのパッケージとしてライブラリ化し、現在使用されているすべてのインタフェースプロトコルに対応するパッケージを開発した。それらを選択することで、多様なインタフェースへの対応を可能とした。

選択といっても、このインタフェースを使って情報をやり取りするには、相手に応じた通信条件（プロトコルやデータフォーマットなど）を設定する必要がある。この条件をパラメータファイルとしてパッケージの外におくことで、各パッケージは、今回開発した分散統合型インタフェースによってマルチプラットフォーム上で動作することが可能となり、接続相手や物理的なインタフェースを意識しないで、データの受け渡しが行えるようになった。

4.3 パラメータ管理

先にも述べたとおり、多様なパッケージ構成やインタフェースに対応するため、その構造を定義し各種パラメータを受け渡すための仕組みにも工夫を凝らした（以下、パラメータ管理という）。パラメータ管理部の具体的な例を図4に示す。

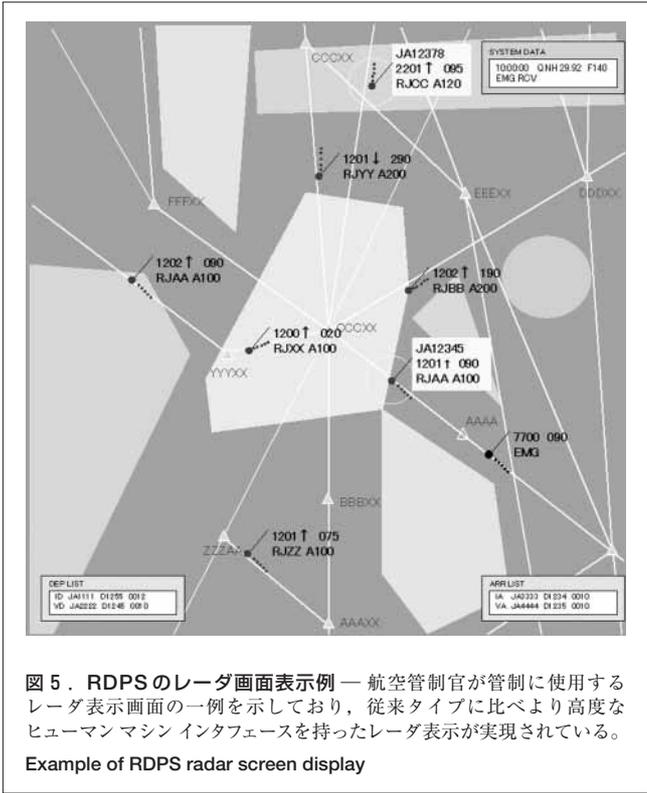
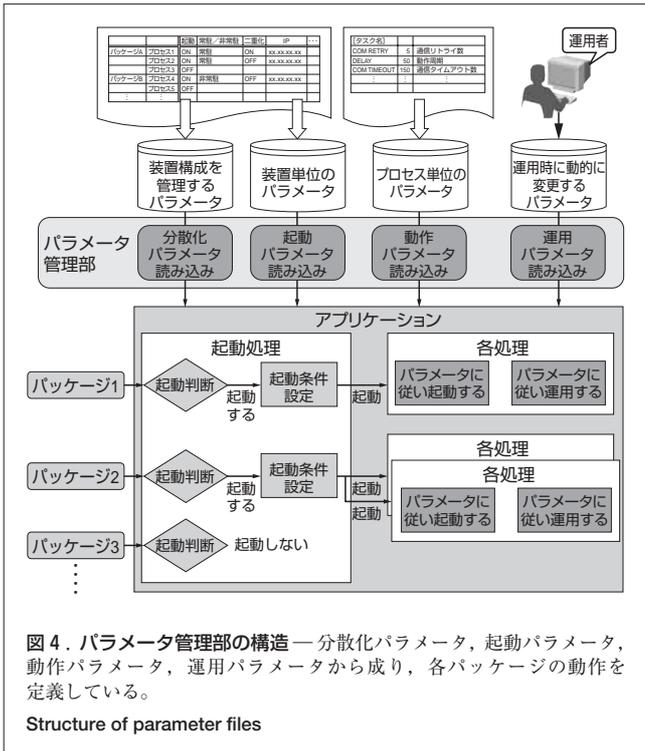
パラメータ管理部は、次に示す四つのパラメータファイルにより、全体のタスクの動作を定義している。

- (1) 装置構成を定義するための分散化パラメータ
- (2) 各装置に機能を割り付ける起動パラメータ
- (3) パッケージ単位の動作を定義する動作パラメータ
- (4) 運用者が動的に変更する運用パラメータ

4.3.1 分散化パラメータ パッケージがどの装置（計算機）に存在するかを定義する情報であり、各パッケージの常駐／非常駐の区別、二重化の有無といったデータが含まれる。

なお、この分散化パラメータは、システム全体の機能配分を一元管理できるよう、分散型のRDPSソフトウェアを構成する装置のいずれかがマスタとなって、スレーブの装置に配布する方式を採用した。

4.3.2 起動パラメータ 自装置（計算機）に存在するパッケージに関するデータから構成されており、自己を規定する情報が定義されたものである。



4.3.3 動作パラメータ 各パッケージを実行するために必要なデータが定義されており、通信リトライ数、動作周期、通信タイムアウト数などで構成されている。

4.3.4 運用パラメータ 各プロセスにおいて運用時に運用者が動的に変更することができるパラメータである。警報処理のしきい値や、画面の色、文字の大きさなどを変更するためのデータなどが含まれ、ユーザーに開放するパラメータである。

5 分散処理型 RDPS ソフトウェアの開発成果

以上に述べてきたパッケージでは、ユーザーの運用カテゴリに応じた様々なバリエーションへの対応が可能となった。分散型統合インタフェース、パラメータ管理、分散・階層化では分散処理型 RDPS ソフトウェアの開発の核となる技術開発ができた。これらにより、ブロック アンド ビルドのコンセプトに基づいた柔軟なシステム構築が可能となった。

ここで開発した RDPS ソフトウェアにレーダ情報を模擬する試験パッケージを組み合わせ、システム全体で十分なパフォーマンスを発揮し、レーダ表示装置にシナリオどおりの情報表示ができることを確認した。今回開発を行った RDPS の構成品の一つであるレーダ表示装置の画面表示例を図 5 に示す。

6 あとがき

様々なユーザーの運用カテゴリを分析し、RDPS ソフトウェア機能パッケージの更なる拡充と処理性能の向上を図るとともに、様々なシステム規模に応じた製品の供給と、高い抗堪性を持つ RDPS の構築を実現した。

引き続き、RDPS の能力向上を積極的に推進し、航空管制業務支援へ更に貢献していきたい。

- 

田口 実男 TAGUCHI Jitsuo
 社会ネットワークインフラ社 小向工場 指揮・情報システム技術部参事。情報処理システムの開発・設計に従事。
 Komukai Operations
- 

山田 達朗 YAMADA Tatsuro
 社会ネットワークインフラ社 官公電波システム事業部 電波システム技術部。航空管制システム機器の開発・設計に従事。
 Defense & Electronic Systems Div.
- 

赤石 高敏 AKAIISHI Takatoshi
 東芝電波システムエンジニアリング(株) 第1技術部主務。航空管制システム機器の開発・設計に従事。
 Toshiba Electronics Engineering Corp.