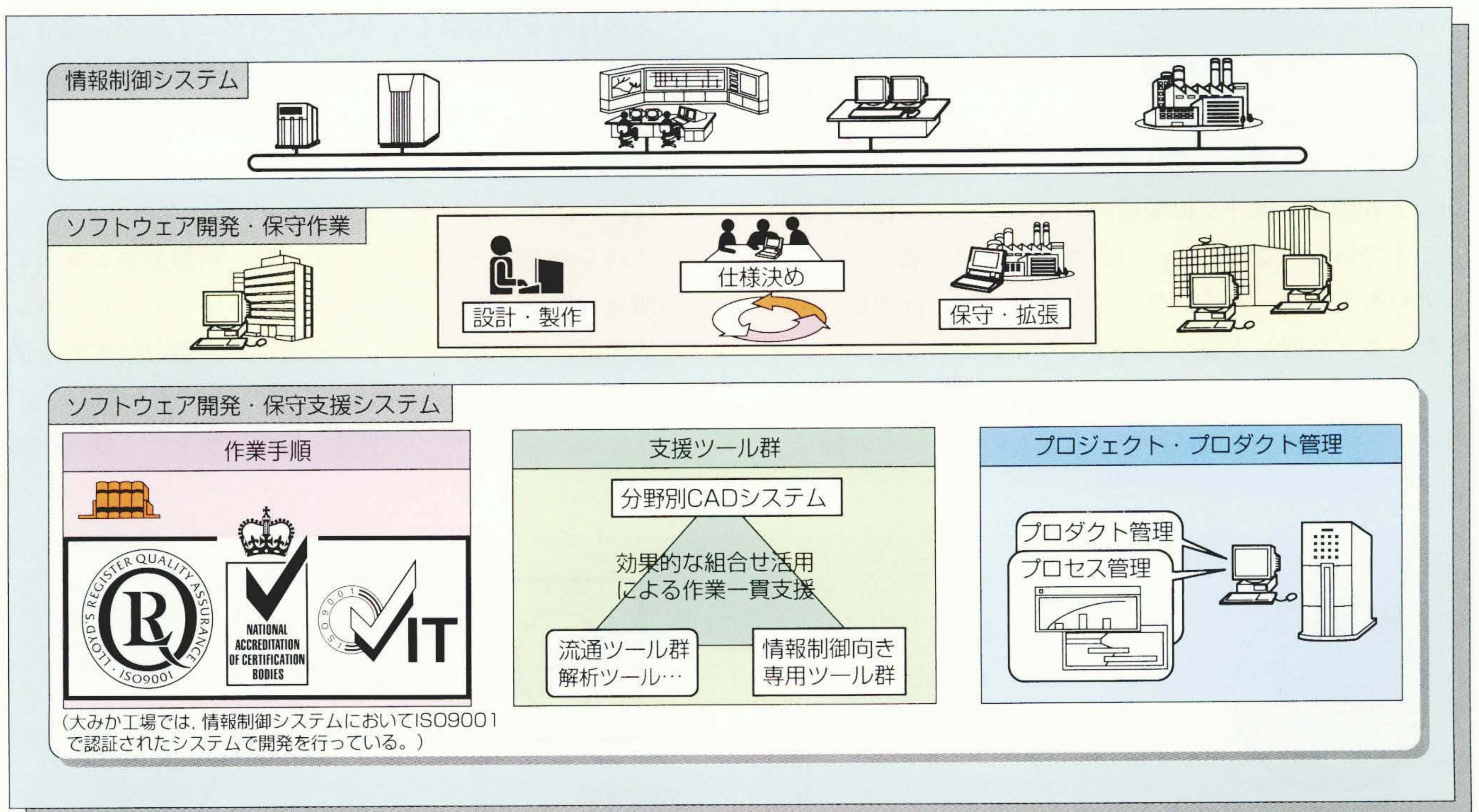


分散・オープン化時代における ソフトウェア開発・保守支援システム

—情報制御システムでのソフトウェアライフサイクル支援—

Software Development and Maintenance Support System in Opened
and Distributed Computer Era

中野利彦* Toshihiko Nakano 廣田敦彦* Atsuhiko Hirota
東原敏昭* Toshiaki Higashihara 高橋勇喜* Yūki Takahashi



ソフトウェア開発・保守支援システム

情報制御システムのソフトウェアの開発・保守では、高信頼性、高効率化が求められている。そのため、「仕様決め」から「保守・拡張」までのライフサイクル全般を、「作業手順」、「支援ツール群」、「プロジェクト・プロダクト管理」の三つの柱で支援するシステムを開発し、ソフトウェア開発・保守作業の環境変化に対する的確な把握と品質の高いソフトウェアの迅速な構築を可能とした。

近年、情報制御システムの分野では、制御範囲の拡大と複雑化への対応、さらにシステムの早期立ち上げの要求に対して、より信頼性の高いソフトウェアを効率よく開発、保守することが求められている。また、ソフトウェア開発・保守を行うための計算機、ネットワークインフラストラクチャーは急速に進展しており、情報制御システム開発・保守環境の分散・オープン化が進展してきている。

これらの情報制御システムのソフトウェア開発・保守を取り巻く背景を的確にとらえ、信頼性の高い情報制御

システムの開発・保守支援を進めていく必要がある。

情報制御システムの開発・保守支援では、「仕様決め」から「保守・拡張」までのライフサイクル全般にわたって顧客のニーズを的確にとらえ、かつ品質の高いソフトウェアを迅速に提供するための種々の活動と、これを支援する開発・保守支援ツールの開発を進めていくことが、普遍の目標である。

今回、これら分散・オープン化の進展を踏まえ、より信頼性の高い情報制御システムのソフトウェアを提供するためのソフトウェア開発・保守支援システムを開発した。

*日立製作所 大みか工場

1. はじめに

近年、情報制御システムの分野では、制御範囲の拡大と複雑化への対応、さらにシステムの早期立ち上げの要求に対して、より信頼性の高いソフトウェアを効率よく開発、保守することが求められている。

このため日立製作所は、情報制御システムでオープンコンポーネントを取り込みながら、段階的システム拡張が可能な「オープン自律分散技術」⁴⁾を開発し提供している。

また、ソフトウェア開発・保守支援環境では、ソフトウェア開発・保守を行うための計算機、ネットワークインフラストラクチャーが急速に進展しており、情報制御システム開発・保守支援環境の分散・オープン化が進展してきている。このため、パソコン上での流通ソフトウェアやネットワークインフラストラクチャー技術を活用した、より高度な支援が可能となってきた。

ここでは、このような背景の中で、情報制御システムのライフサイクル全般にわたって顧客のニーズを的確に

とらえ、かつ品質の高いソフトウェアを継続的に提供している日立製作所の活動、およびソフトウェア開発・保守支援システムについて述べる。

2. ソフトウェア開発・保守支援システムの考え方

ソフトウェア開発・保守支援システムの目的は、情報制御システム、およびソフトウェア開発・保守作業の環境変化を的確に把握し、顧客が満足するシステムを高品質、かつ迅速に実現することである。

この目的を実現するためにソフトウェア開発・保守支援では、情報制御システムの仕様決めから保守・拡張までのソフトウェアライフサイクル全般を、(1)作業手順、(2)支援ツール群、(3)プロジェクト・プロダクト管理の三つの柱で総合的に支援している。

これらの支援を実現するためには、情報制御システムの開発・保守環境でのワークステーションや、パソコンの活用が進展している中で、流通ツール群の効果的な活用が不可欠である。

さらに、ソフトウェア開発・保守作業を行う個人への

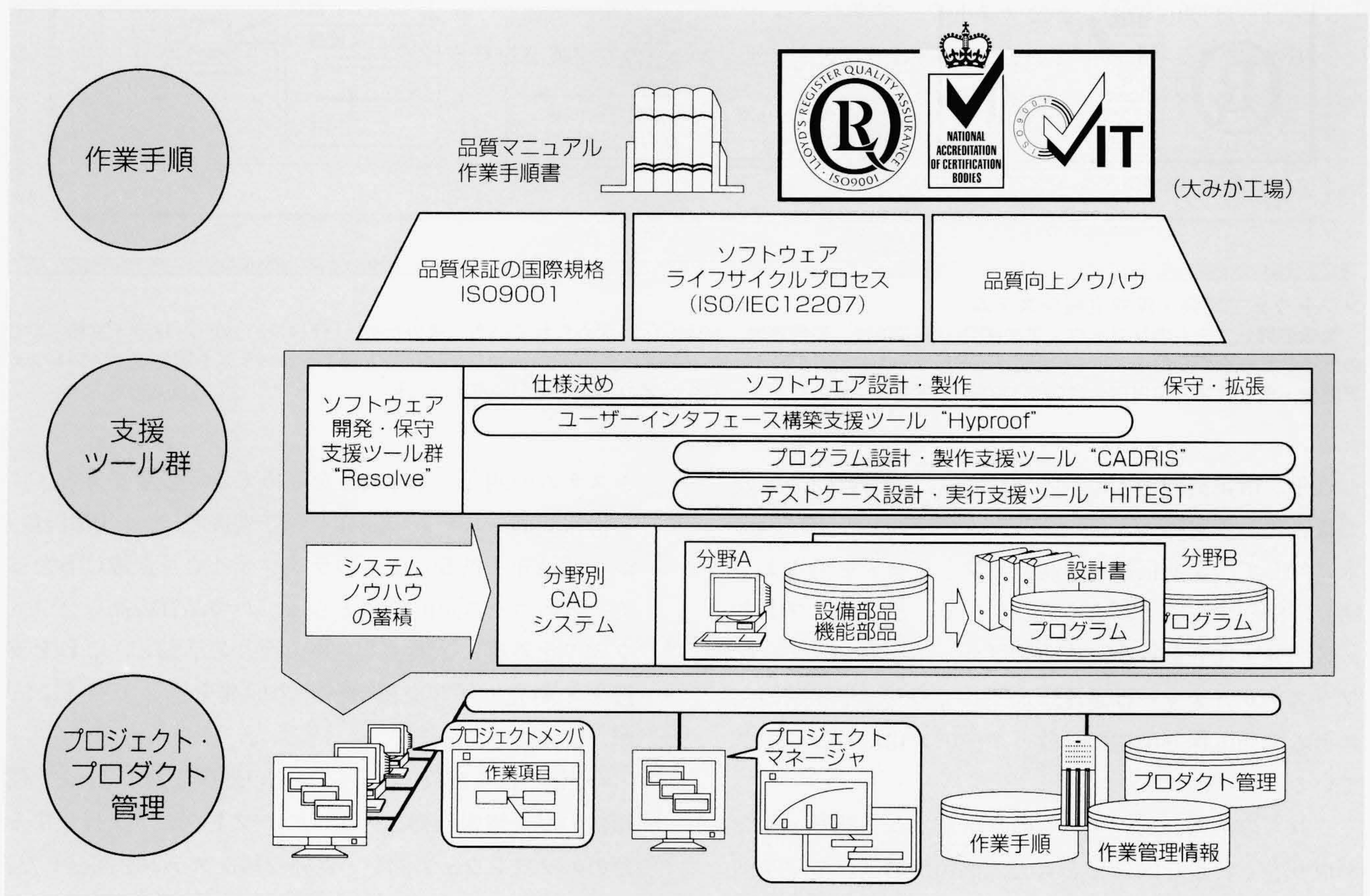


図1 ソフトウェア開発・保守支援システム

ソフトウェアライフサイクル全般を「作業手順」、「支援ツール群」、「プロジェクト・プロダクト管理」の三つの柱で総合的に支援する。

パソコンの普及や、情報流通を円滑に行うための新たなネットワークインフラストラクチャーが進展してきており、これらを活用した効率的な分散開発・保守作業環境の提供が求められている。

3. ソフトウェア開発・保守支援システム

ソフトウェアの開発・保守支援は、「仕様決め」から「保守・拡張」までのライフサイクル全般を、「作業手順」、「支援ツール群」、「プロジェクト・プロダクト管理」の三つの柱で支援している(図1参照)。

3.1 作業手順

ソフトウェアの開発は、複数の人々によって分散して進められる。また、将来の保守・拡張では、前回の開発時とは異なる人々によって開発される。このような状況でシステムを高品質でかつ効率よく進めるためには、どのような作業をいかなる手順や方法で行うべきかというソフトウェア設計・生産・保守過程(プロセス)が重要である。

このソフトウェアを生み出す過程での品質確保について種々の国際的な活動がある。品質保証に関する国際規格としてISO9000規格が、またソフトウェア開発・保守などの作業プロセスについては、ソフトウェア ライフサイクル プロセス(ISO/IEC 12207)がある。

そのため、これらソフトウェアの品質確保のための国際的な規準に適合させながら、長年培われてきた品質確保のノウハウをベースに、開発・保守の作業を体系化し

ている。この体系は、共通な品質マニュアルの整備、および情報制御システムの各分野ごとの作業手順として整備し、運用している。各プロジェクトでは、この作業手順に基づいてシステムの開発・保守を行うことにより、高水準で均質な作業の実現を図っている。なお、この品質確保のための活動の維持についてはISO9001 with Tick-IT^{※1)}の認証を受けている。

3.2 支援ツール群

作業手順をより効果のあるものとし、高品質なシステムを完成していくために、機能仕様設計から製作・テストまでを一貫して支援する種々の開発・保守支援ツール群をResolve(Real-time Software Engineering Platform with Integrated and Visual Interface)として提供している。

また、各アプリケーション分野ごとに、システムのノウハウを最大限に活用するための分野別CADシステムを提供している。

3.2.1 ユーザーインタフェース仕様決め・

構築支援ツール “Hyproof”

システム仕様を早期に確定することは、システムを早期に稼働させるために重要である。特にユーザーインタフェースはシステムと人との間の仕様であり、プラントの安定操業を行ううえで使いやすさが重要である。

また、作業工数的にもユーザーインタフェース関連に

※1) ISO9001 with TickIT：ソフトウェア製品も含むISO 9001認証

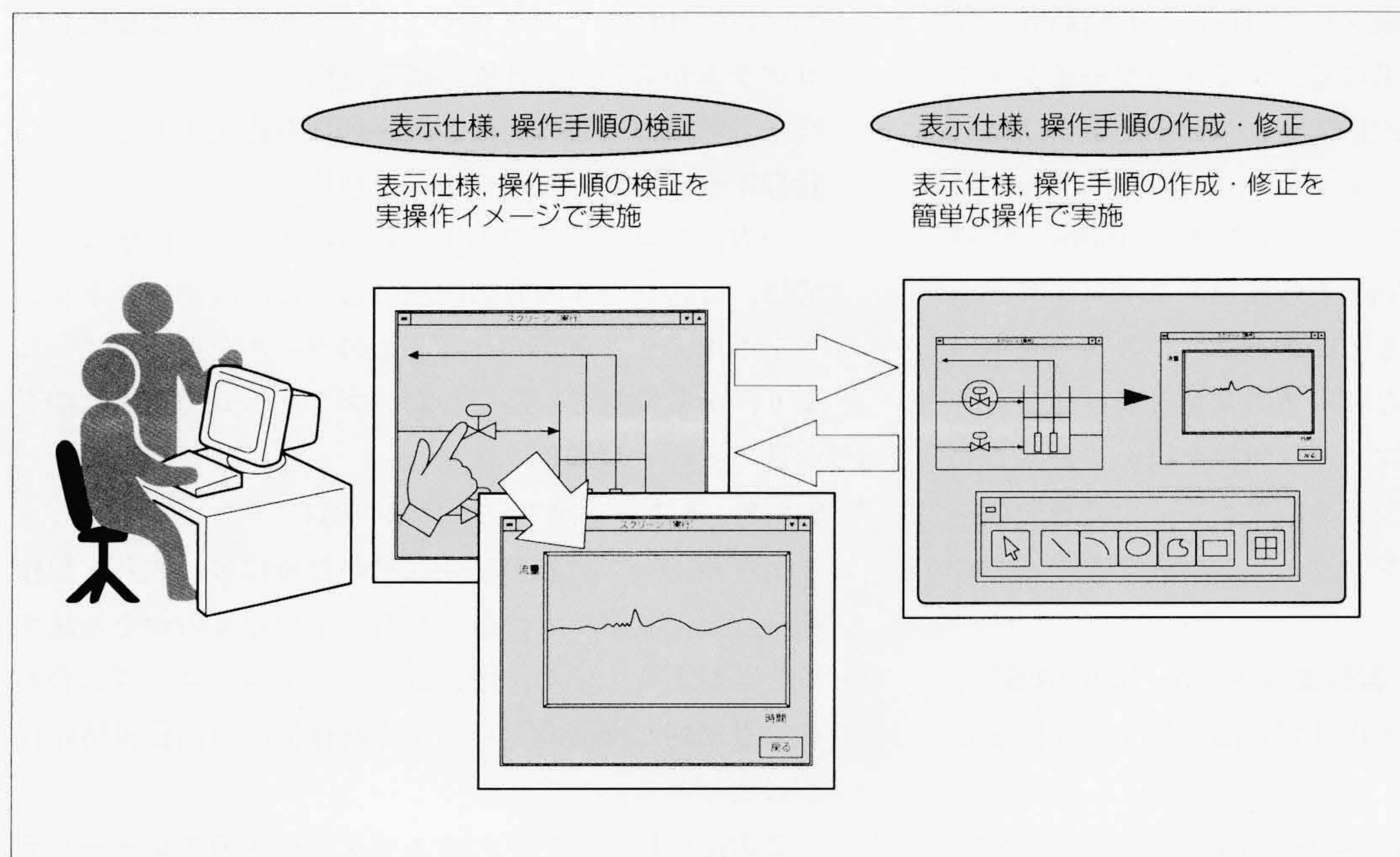


図2 ユーザーインタフェース仕様打ち合わせ支援ツール“Hyproof/Speculer”

本物に近いユーザーインタフェースを実際の利用者に実感してもらい、その場での意見を逐次反映することができるツールであり、仕様の早期確定を支援する。

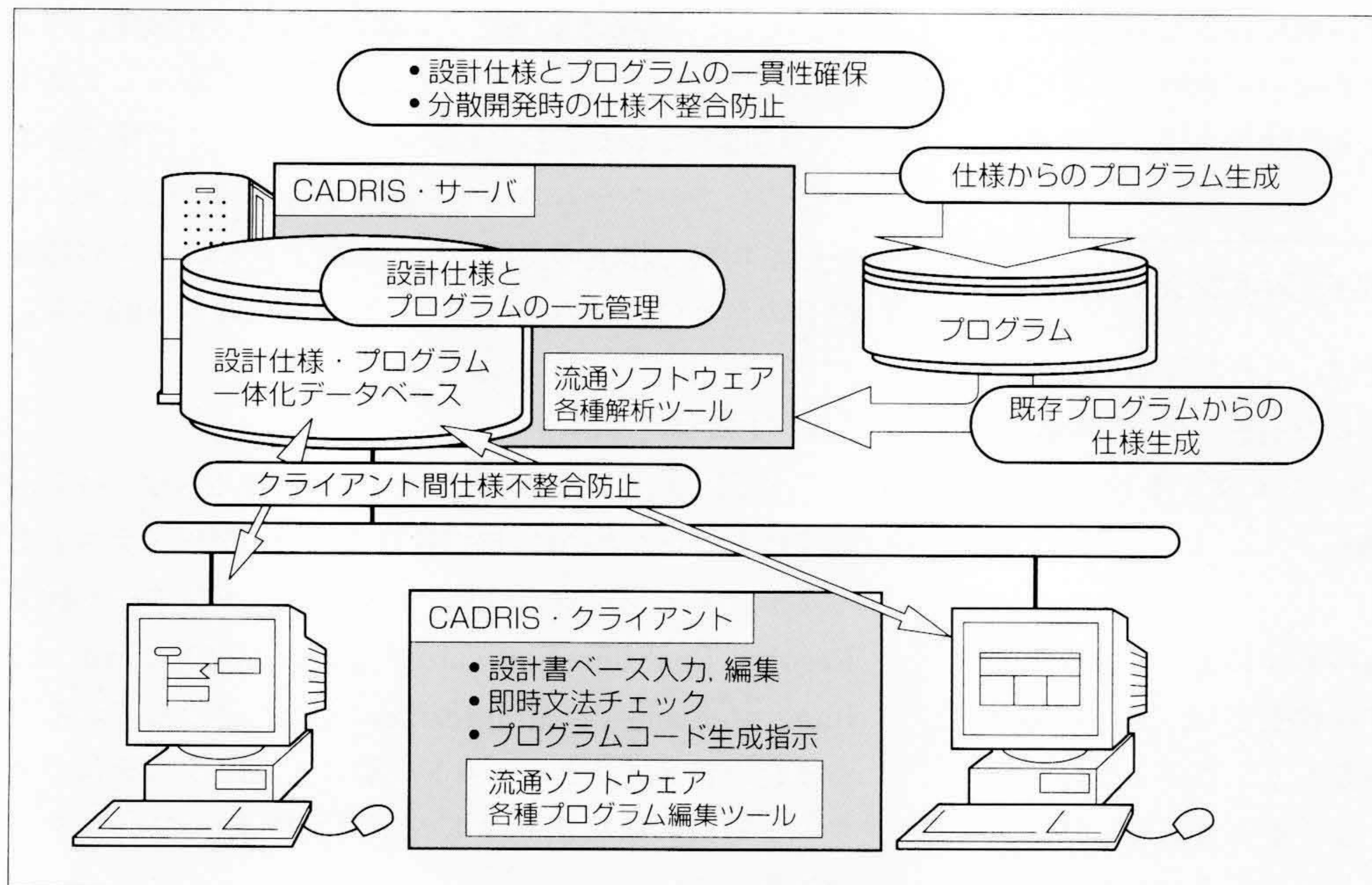


図3 プログラム設計・製作支援ツール「CADRIS」

プログラムおよびドキュメントを設計仕様ベースで作成する。プログラム設計仕様とプログラムの一元性、分散開発での仕様不整合を防止することにより、品質の高いプログラム開発を支援する。また、既存システムのプログラムから設計仕様を構築する機能を持ち、継続的な品質向上を実現している。

ますます多くの工数が必要となってきた。このため、システム稼動間近に、ユーザーインタフェースの不都合が見つかった場合、システム稼動時期に影響を及ぼしかねない。

この問題の解決には、仕様決めの段階で検討しているユーザーインタフェースを顧客に実際に使用してもらい、利用者にとって使いやすい仕様を早期に決めることが重要である。

このため、ユーザーインタフェースのプロトタイプが簡単に作成でき、画面の表示仕様や操作性が容易に確認できる仕様打ち合わせ支援ツール「Hyproof/Specular」、およびその打ち合わせ結果に基づいてユーザーインタフェースのソフトウェアを生成する「Hyproof/MMS」(Hyproof/Man-Machine System Support)を開発し、適用している(図2参照)。このツール群は、Windows^{※2)}パソコン上で動作し、Windows上の流通ソフトウェアを活用した画面の作成や変更も可能である。このツールを用いることにより、最終的な利用者の意見を聴きながら、その場でユーザーインタフェース仕様を決めていくことができる。同時に、打ち合わせ結果から、設計書やユーザーインタフェースのソフトウェアを生成することができる。

3.2.2 プログラム設計・製作支援ツール「CADRIS」

プログラム設計・製作・保守での品質低下の要因とし

て、仕様変更が重なり、設計書とプログラムが乖(かい)離してしまう問題がある。

さらに、分散・オープン化した開発環境で品質の高いプログラムを迅速に開発するためには、分散配置された複数計算機間での仕様の不整合が発生する可能性がある。

プログラム設計・製作支援ツール「CADRIS」(Computer Aided Design and Reuse Environment with Intelligent Support)(図3参照)はこれに対応したツールである。プログラム、およびドキュメントを設計仕様ベースで作成することを特徴とし、さらに次の機構を持つ。

- (1) 設計仕様からのプログラム生成機構、および既存プログラムからの設計仕様再構築機構
- (2) 分散した複数の場所で同一仕様を修正するといった仕様不整合の要因を防止する管理機構

一方、プログラムの編集や各種のチェック機構については、このツール固有の機能のほかに、流通ソフトウェアを活用することができる。このツールを用いることにより、分散開発でも品質の高いプログラムの迅速な開発を進めることができる。

3.2.3 テスト支援ツール「HITEST」

情報制御システムの早期稼動のためには、設計・製作を行ったプログラムの最終品質が十分なものを検証する必要がある。また、情報制御システムのニーズに合わせた継続的な機能向上に対する既存機能の品質維持確保が必要である。

この品質検証をプログラムモジュールのモジュールテ

※2) Windowsは、米国Microsoft Corp.の登録商標である。

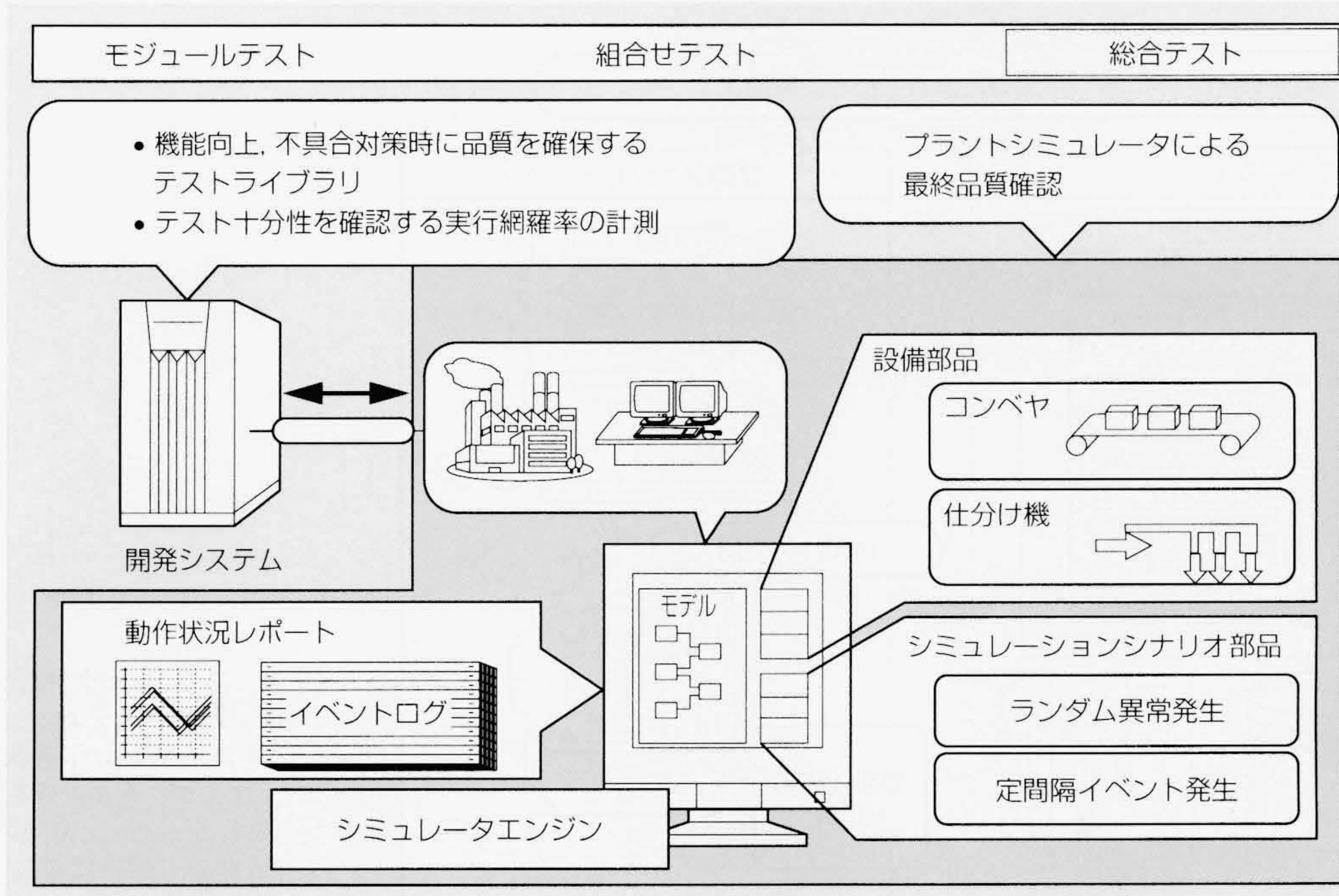


図4 テスト支援ツール「HITEST」による総合テスト支援

他システムやプラントとのインタフェースをシミュレーションでテストすることにより、システムの最終品質を向上させる。また、シミュレーションモデルの各種部品を用意することにより、シミュレーション品質の向上を図っている。

スト、および組合せテスト実行、さらにはシステム最終段階での総合テスト実行までを一貫して支援するツールがテスト支援ツール「HITEST」(Hitachi Integrated Test System)である。HITESTの特徴は次のとおりである。

- (1) 機能向上や、不具合対策時の品質低下を防止するためのテストライブラリ機能(テスト前提条件と予想結果をライブラリ化し、テスト結果を自動照合)
- (2) 被テストプログラムのテスト十分性を判定するための実行網羅率を自動計測する機能
- (3) 最終品質を確認する総合テストでは、プラント設備を使用しての情報制御システムの調整作業期間短縮や、既設システム停止の最小限化などのプラントの早期稼動を目的とした、プラントシミュレータを活用したテスト機能(図4参照)

特に総合テストの支援では、開発対象システムを取り巻く機器やほかのシステムの入出力動作インタフェースのシミュレーションを行い、最終的な動作確認を行う。シミュレーションによるテストを行う場合は、シミュレーションモデルについても構築を容易にし、かつ信頼性を高める必要がある。このため、このツールでは、プラント機器や情報伝達のモデル、さらにシミュレーションのシナリオを部品化している。この結果、これらの部品を組み合わせることにより、目的とするほかの計算機や、設備の動作を模擬するシミュレーションモデルを容易に構築することができるようになった。

3.2.4 分野別CADシステム

品質の高いシステムを構築するうえで、各アプリケーション分野での経験を効果的に活用することが重要である。分野別CADシステムは、各分野での仕様決めからソフトウェア設計・製作までのノウハウを整備しCAD化したものである。このシステムの特徴は次のとおりである。

- (1) 各分野での設備や、機能の部品を持ち、この部品の組合せ、および各部品のパラメータを定義することによるシステム機能の構築を可能とする。
- (2) 定義されたシステム機能に基づいてシステムのデータベース構造や、プログラムを生成する。

分野別CADシステムは、経験のある部分に対して大きな効果をもたらすが、個別の情報制御システムは、システムごとに、経験のある部分と、新たに開発を行う部分が混在することが多い。これに対しては、前記の支援ツール群と分野別CADシステムを効果的に組み合わせて活用していくことにより、品質の高い情報制御システムを提供している。

3.3 プロジェクト・プロダクト管理

分散開発で品質の高いソフトウェアを効率よく開発、保守するためには、作業手順に基づく作業状況、および設計書、プログラムといったプロダクトの完成状況をプロジェクトごとに的確に管理することが不可欠である。

これらの作業状況およびプロダクトの管理を実現するためのツールがプロジェクト管理情報支援システムであ

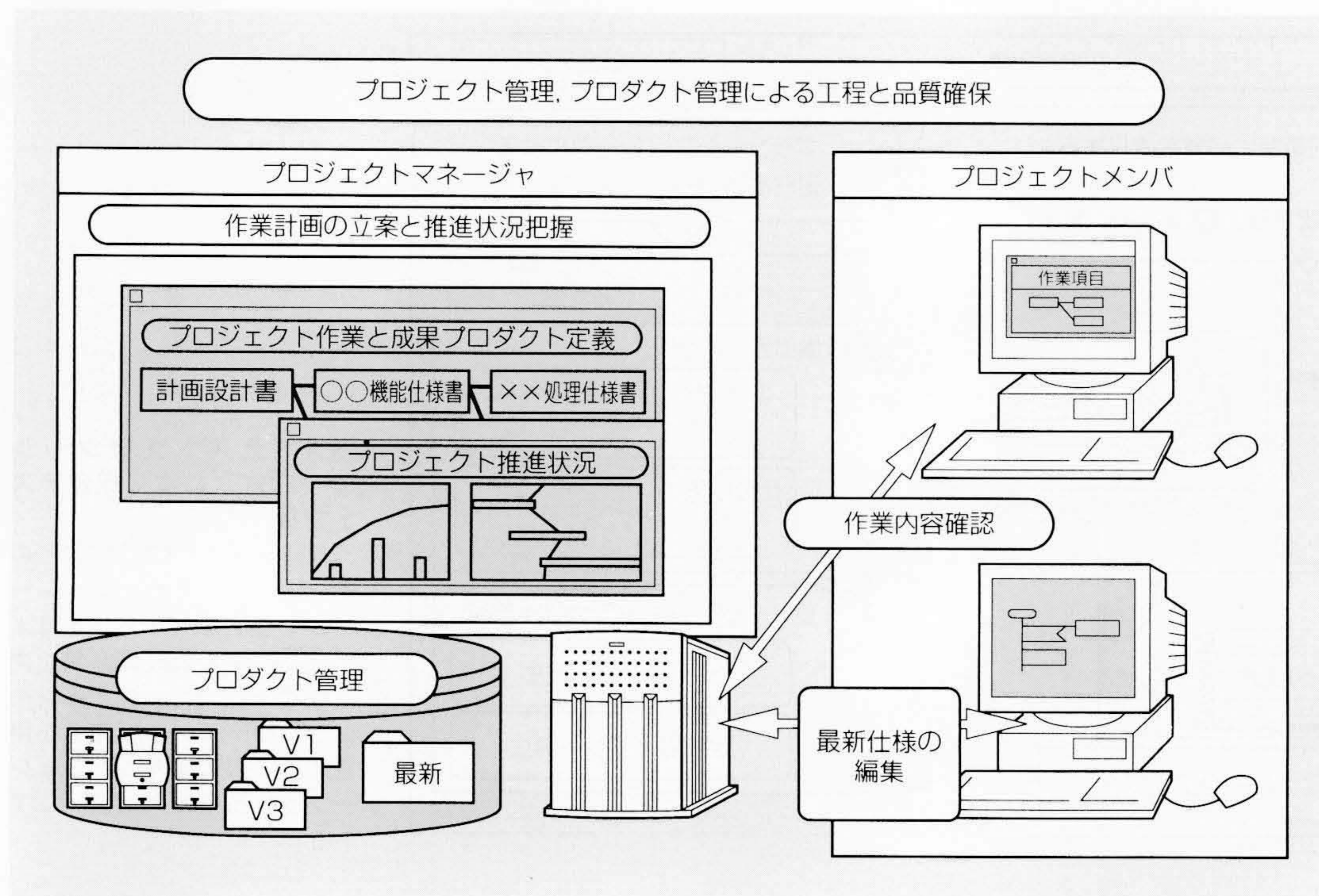


図5 プロジェクト管理
情報支援システム

プロジェクトの作業状況と、プロダクトの管理を的確に行うことにより、システム開発、保守の期限、および品質確保が支援できる。

る(図5参照)。このシステムの特徴は次のとおりである。

- (1) プロジェクト全体のメンバが、行うべき作業および成果とするプロダクトを正しく認識するためのプロジェクト可視化機能
- (2) プロジェクト推進の問題点を早期に把握し、的確な対策を可能とするため、開発・保守支援ツールから管理情報を自動収集する機能
- (3) プロジェクト内での情報不一致による品質低下を防止するため、最新プロダクトの共有、変更来歴の管理を行う成果物管理機能

これらにより、分散開発での情報の共有や、分散開発プロジェクトでのきめ細かい管理、および保守時の品質確保が実現できる。

さらに、システムはプラント運用を開始した後も、継続的な機能向上を進めていく必要がある。この継続的な機能向上でも品質の高いソフトウェアを継続的に提供していくためには、完成したプロダクトを的確に管理す

ることが必要である。

このため、開発を終了したプロジェクトのプロダクトもすべて日立製作所内に保管、管理し、保守時に利用するとともに、保守結果が反映できるようにしている。

4. おわりに

ここでは、高品質な情報制御システムを迅速に提供し続けるために、分散・オープン化の進展を踏まえて、「仕様決め」から「保守・拡張」までのライフサイクル全般にわたって支援するソフトウェア開発・保守支援システムについて述べた。

ここで述べた活動は、情報制御のさまざまな分野でのシステム開発に適用され、開発効率の向上、品質の向上に寄与している。

今後も、情報制御システムのいっそうの高品質化と生産性向上を目指し、ソフトウェア生産技術の開発を進めていく考えである。

参考文献

- 1) 高橋, 外: 情報制御システム向けソフトウェア開発, 保守支援システム“Resolve-I”, 日立評論, 73, 8, 795~800(平3-8)
- 2) 和田, 外: 分散化が進む情報制御システムのソフトウェア開発・保守支援システム“Resolve-II”, 日立評論, 77, 7, 475~480(平7-7)
- 3) 林, 外: 計算機制御システムのための高品質ソフトウェア開発方法, 電気学会論文誌C, 114巻, 6号, 645~653(平6-6)
- 4) 小林, 外: 分散・オープン化時代における情報制御プラットフォームの新展開一次期HIDIC AZシステムによる高信頼・高生産支援一, 日立評論, 78, 10, 721~726(平8-10)