

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”

Computer Aided Software Engineering System for Realtime Software Systems

近年, 情報制御システムの開発ソフトウェア量は, 制御範囲の拡大と複雑化に伴ってますます増大の一途をたどり, これに伴いソフトウェアを信頼性高く, 効率よく開発するための近代的ソフトウェア開発支援システムが強く要求されている。

日立製作所では, この要求にこたえるために, 従来ソフトウェア工学に基づいた各種ソフトウェア開発支援ツールを開発, 適用して効果を上げてきたが, さらに一歩進めて, 従来のプログラミング言語を中心としたソフトウェア開発作業から脱却し, ソフトウェアの設計から保守までを一貫自動化支援する高効率ソフトウェア開発, 保守支援システムResolve I (Realtime Software Engineering Platform with Integrated and Visual Interface I)を開発した。これにより, 大規模なソフトウェアも, より信頼性高く, かつ効率よく開発, 保守できるようにした。

高橋勇喜* Yûki Takahashi
大脇隆志* Takashi Ôwaki
林 利弘* Toshihiro Hayashi
大槻 繁** Shigeru Ôtsuki

1 はじめに

近年, 情報制御システムの開発ソフトウェア量は, 制御範囲の拡大と複雑化に伴ってますます増大の一途にある。これに伴いソフトウェアを信頼性高く, 効率よく開発するためのソフトウェア開発支援システムの整備が急務となっている。

日立製作所では, 従来, 情報制御システムのソフトウェア開発支援システムとして, 「HIDIC V90/5シリーズソフトウェア開発, 保守支援システム」¹⁾(ソフトCAD/CAM/CAT一貫支援ツール)を開発し, ソフトウェアの信頼性および生産性向上に効果を上げてきた。最近では, これに加え「大規模ソフトウェアの多人数・分散開発」, 「支援ツールのレベルアップによるソフトウェア品質個人差の解消と信頼性の確保」といった新たなニーズも強い。今回, これらを解決するため従来の「プログラミング言語」をベースとしたソフトウェアの開発作業から脱却し, 「ソフトウェアの設計から保守までを一貫自動化支援」するシステムResolve I (Realtime Software Engineering Platform with Integrated and Visual Interface I)として体系化した。

Resolve Iは, 長年の情報制御システム開発でのソフトウェアエンジニアリング適用で培った実績と経験に裏打ちされた,

実戦的なCASE(Computer Aided Software Engineering)ツール体系である。

2 開発思想とシステムの体系

2.1 Resolve Iの開発思想

Resolve Iは, 情報制御システムのソフトウェアを取り巻く環境の変化に対応して,

- (1) 設計から保守までを, 統合データベース上で, かつ高レベルインタフェースで支援すること。
 - (2) 複雑化するソフトウェアの構造, 動態をビジュアル化し, わかりやすくすること。
 - (3) 分散開発環境下での多人数による開発を, 高度な対話環境と統一された操作インタフェースによって支援すること。
- を基本思想として開発した(図1)。

(1) 設計仕様レベル一貫支援インタフェース

開発すべきソフトウェアの大規模化, 複雑化と開発期間の短縮化に伴い, ソフトウェア開発要員には, より多くの専門知識と各種ソフトウェアツールの利用技術が要求されている。一方, ソフトウェア開発の需要増大に対して, 開発要員の絶

* 日立製作所 大みか工場 ** 日立製作所 システム開発研究所

対的不足が叫ばれている。

このような状況にあっては、作業者をソフトウェア開発にかかわる雑作業から解放し、より本質的な設計作業だけに専念できるようにすることが必須(す)である。また、個々のプログラミング言語に依存した文法知識はできるだけツールによって支援し、新入開発要員でも早期に高信頼度のソフトウ

ェアを開発できるようにする必要がある。

これに対処するため、設計から保守まで、操作対象を設計仕様情報のレベル(脱プログラミング言語化インタフェース)で一貫して扱えるようにした。これにより、各開発工程間のつなぎのための手作業を排除し、各種仕様情報(設計仕様、プログラム ソース コードなど)間の不一致も防止することができる。脱プログラミング言語化インタフェースでの作業形態と効果を図2に示す。

(2) ソフトウェア開発支援環境のビジュアル化

システムの高機能化、大規模化は、またソフトウェアを複雑かつ膨大にしており、開発対象ソフトウェアの全体像の把握、動的なふるまいの把握をますます困難なものにしている。さらに、ソフトウェアの開発作業そのものの状況把握も、対象ソフトウェアの信頼性確保のために重要である。

これを解決するため、ソフトウェア開発支援ツールに、「ソフトウェアの可視化」の考えを取り入れ、ソフトウェアのふるまいと開発作業の状況を目で確かめられるようにした。これにより、ソフトウェアの動的、静的な状態把握が容易になり、システムの信頼性、保守性の向上を図っている。

(3) インタオペラブルな分散ソフト開発環境

大規模ソフトウェアを多人数で開発していくためには、分散した開発環境で作業を分担しながら、協調した開発を行っていくことが必要となる。これを支援するため、分散した個

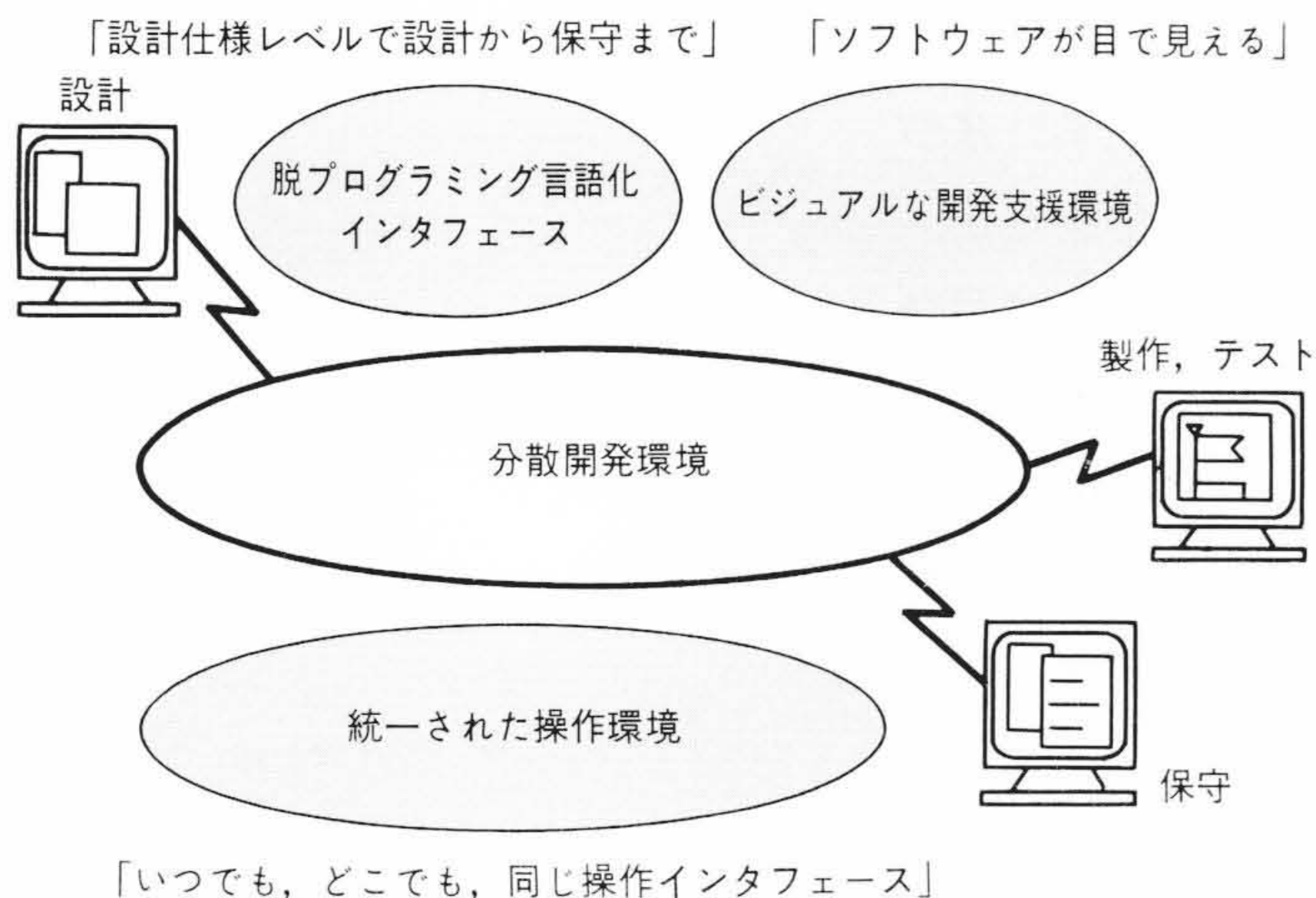
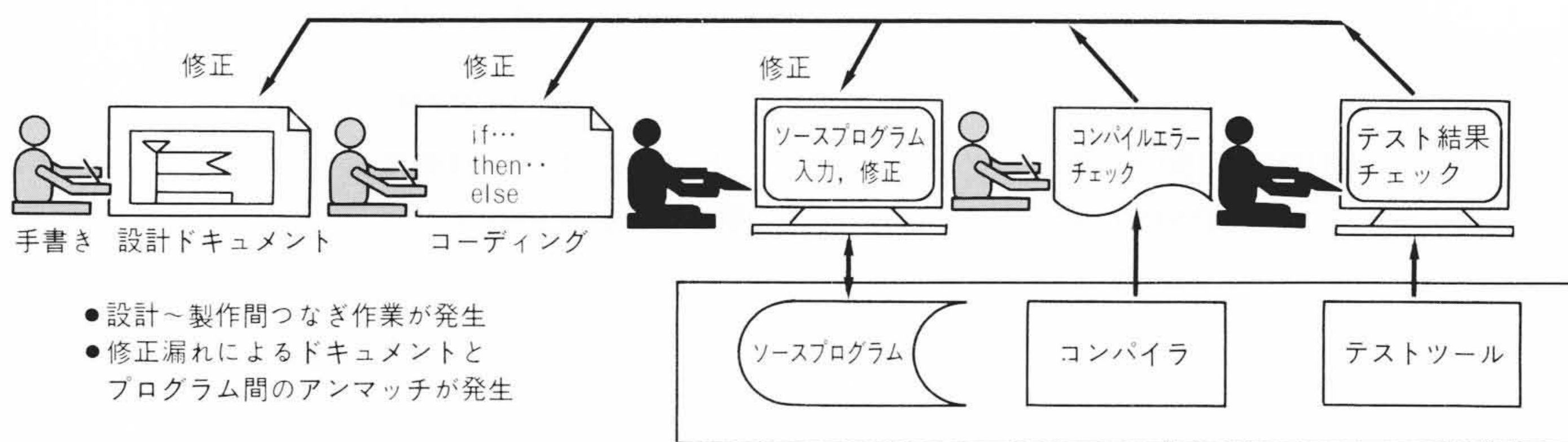


図1 Resolve Iの開発思想 Resolve Iは、脱プログラミング言語化インタフェース、ビジュアルな開発支援環境および統一された操作環境の三つの基本思想に基づき開発した。

従来のプログラミング言語をベースとしたソフト開発



脱プログラミング言語化ソフト開発

- プログラミング言語の文法知識がなくてもソフト作成可
- 設計仕様とプログラムの不一致を防止

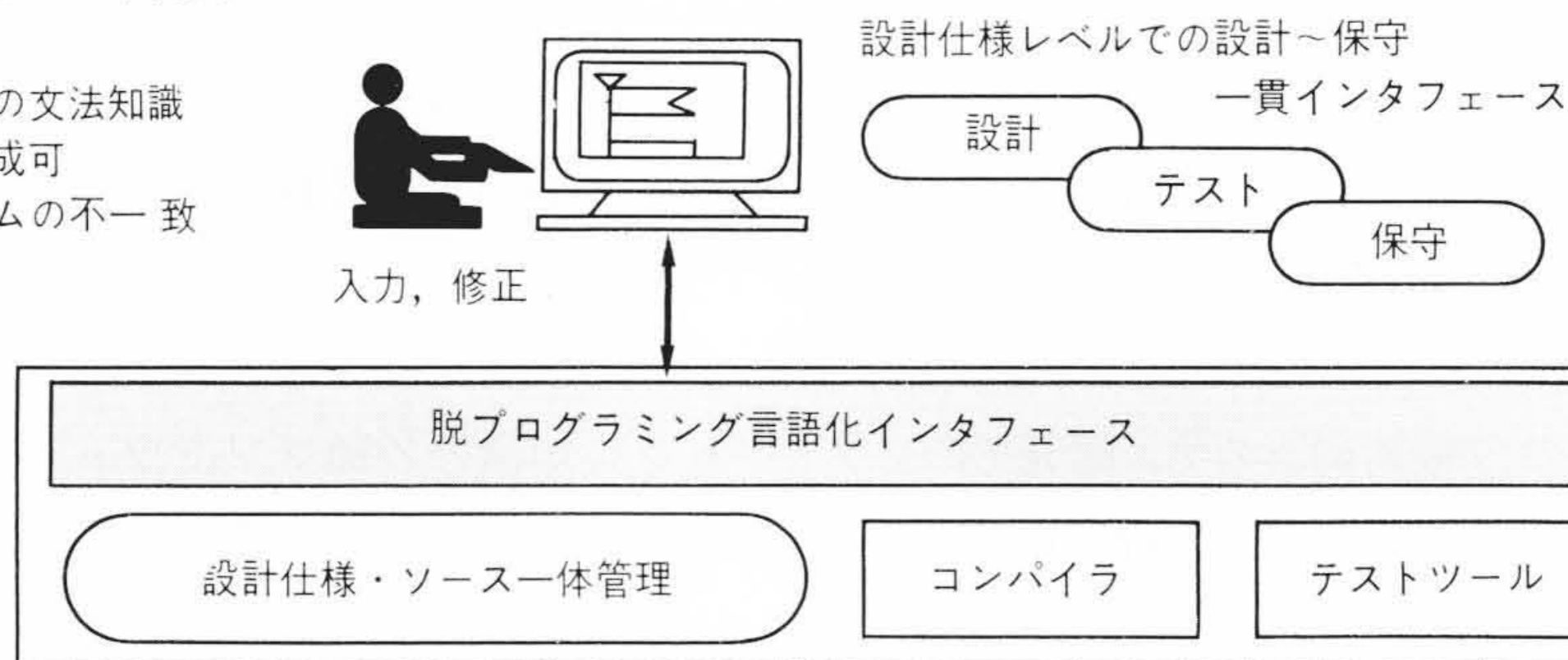
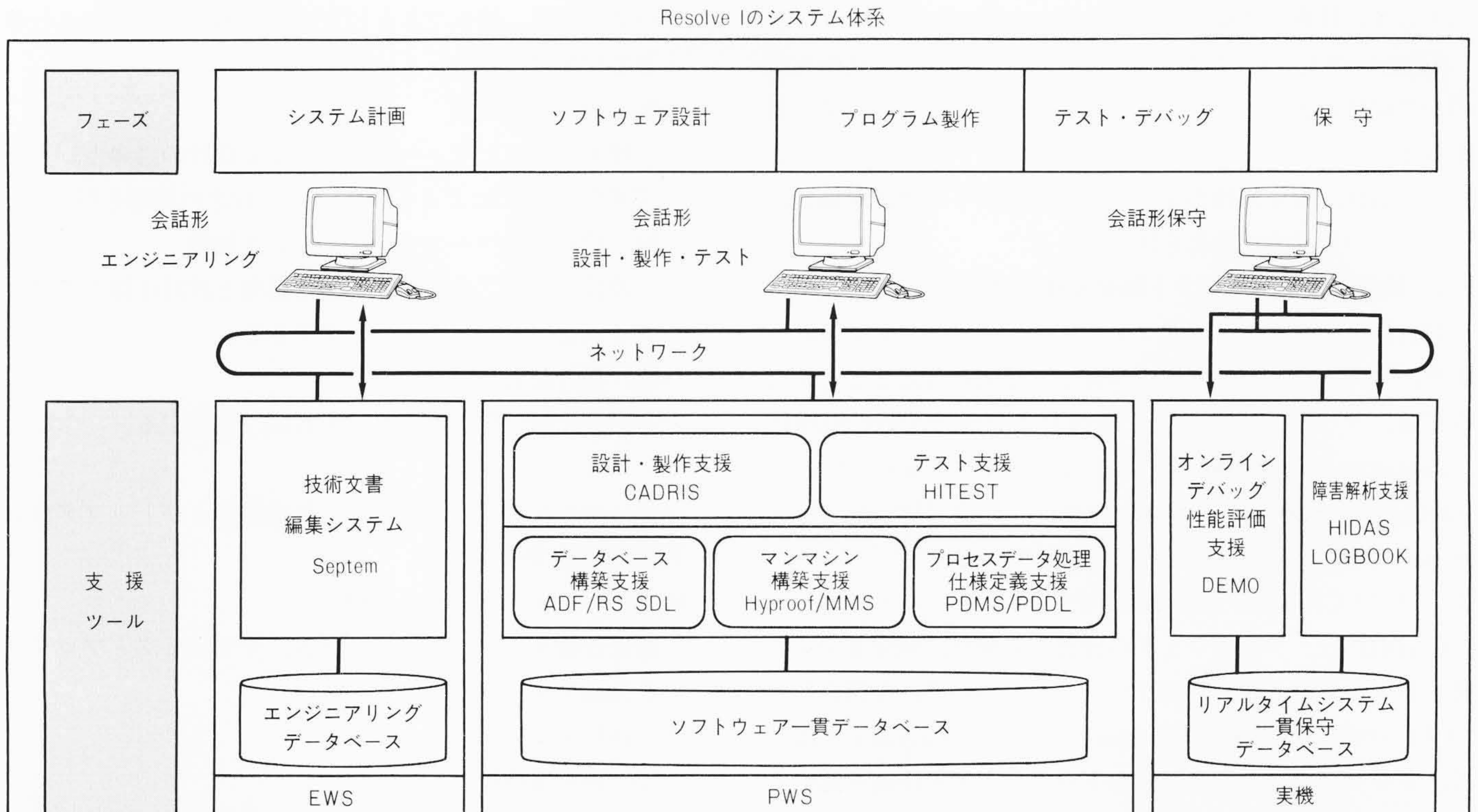


図2 脱プログラミング言語化ソフト開発インタフェース 設計仕様レベルの設計～保守一貫インタフェースにより、特定のプログラミング言語に依存したソフト開発から脱却した、高レベルのソフト開発環境を実現する。



注：略語説明 EWS (Engineering Work Station), PWS (Programming Work Station), CADRIS (Computer Aided Design and Reuse Environment with Intelligent Support), HITEST (Hitachi Integrated Test System), ADF/RS (Advanced Data Management Facilities for Realtime Application System), SDL (Structured Database Language), Hyproof (Hyper Programmig System by Object Oriented Formula), MMS (Man-Machine System Support), PDMS (Process Data Management System), PDDL (Process Data Description Language), DEMO (Determinate Evaluation, Monitoring and Output System), HIDAS (HIDIC Diagnosis Assistant System)

図3 ソフトウェア開発、保守支援システムResolve Iの体系 Resolve Iはワークステーション上で、システム計画、設計から保守まで一貫してツールで支援する体系となっている。

個の開発マシンをネットワークで結合し、各種共通仕様情報の共用化、統一した操作インタフェースの実現を図った。これにより、開発作業者は、おのこの開発場所で担当のソフトウェア開発を行いながら、共通仕様情報の参照や担当者間の仕様の連絡を統一したインタフェースで行えるようにした。

2.2 Resolve Iのシステム体系

本システムの構成を図3に示す。Resolve Iでは、システムの計画、設計から保守に至る各開発作業を一貫して、ツールで支援する。また、開発マシンが分散していても、統一した操作環境下で開発情報を共有しながら、協調開発を行えるようにしている。

(1) ソフトウェア開発一貫支援ツール体系

システム計画から保守までをツールによって一貫して支援する。各作業フェーズ間の中間生産物を含め、ソフトウェア情報を一元管理する「ソフトウェア一貫データベース」を具備する。これにより、ツール間の仕様情報の共有化、ツール間のつなぎ作業の自動化を図り、開発作業の質的向上、量的効率向上を実現する。

(2) ワークステーション応用分散開発環境

ワークステーション²⁾のGUI(Graphical User Interface)を利用し、使い勝手の良い、かつビジュアルな開発インタフェースを実現する。

また、物理的に分散配置された複数台のワークステーションをネットワークを介して結合し、ソフトウェア開発用のオペレーティングシステムとして評価の高いUNIX^{※)}、分散ファイル管理システムNFS(Network File System)などの業界標準のプラットフォームを採用している。これらにより、作業者は分散開発をしていても、ソフトウェア情報を統一的に扱うことができるようにしている。

3 ソフトウェア開発、保守支援システムの機能と特長

3.1 技術文書編集システム Septem

計画設計書や機能仕様書などのドキュメント作成、再利用

※) UNIX：UNIXオペレーティングシステムは、UNIXシステムラボラトリーズ社が開発し、ライセンスしている。

を、EWS(Engineering Work Station)上で支援するシステムとして、技術文書編集システムSeptemがある。

Septemでは、文字情報だけでなく図や表も一つの技術文書の中で同時に扱え、ドキュメントの作成、再利用を容易に行えるようにしている。また、計画ドキュメントから、ソフトウェア設計、製作情報のデータを自動生成する機能を備え、下工程との接続の自動化を図っている。

3.2 設計仕様レベルソフト開発支援システム

設計仕様レベルのインタフェースで、ソフトウェア設計、製作、テストおよび保守まで一貫して作業が行えることを支援する。ソフトウェア設計、製作および保守ではCADRIS (Computer Aided Design and Reuse Environment with Intelligent Support), テストではHITEST(Hitachi Integrated Test System)によって支援している。

(1) ソフトCADシステム CADRIS

CADRISは、ソフトウェアの設計から製作、保守までを支援するツールで、設計仕様をベースに定型化された図表形式のフォーマットやPAD³⁾(Problem Analysis Diagram)を用いて、トップダウンかつ高品質なプログラム開発作業を進めることができる。

CADRISの機能を以下に述べる(図4)。

(a) 設計・製作支援機能

(i) 設計仕様定義機能

ソフトウェア設計仕様を図表形式のワークシートベースで入力、修正できる(テーブル仕様、モジュール仕様および処理手続き)。

(ii) 設計誘導機能

標準化されたフォーマットによる設計の誘導と、トップダウンPADによるトップダウン設計の誘導を行う。

(iii) 設計仕様ベースプログラミング機能

設計仕様、プログラムの同時編集と設計仕様上でプログラムをコンパイルチェックできる。

(iv) 設計品質検証機能

設計仕様の整合性、品質を評価、検証する。

(b) ドキュメンテーション機能

テーブル仕様書、モジュール仕様書およびトップダウンPAD図を自動生成する。

(c) オブジェクト自動生成機能

設計仕様からソースプログラム、オブジェクトプログラムを自動生成する。

(d) 再利用支援機能

(i) 再利用機能

会話形で既存ソフトウェアを検索できる。また、設計仕様とプログラムを一体管理した単位で、既存ソフトウェアを再利用できる。

(ii) リエンジニアリング支援機能

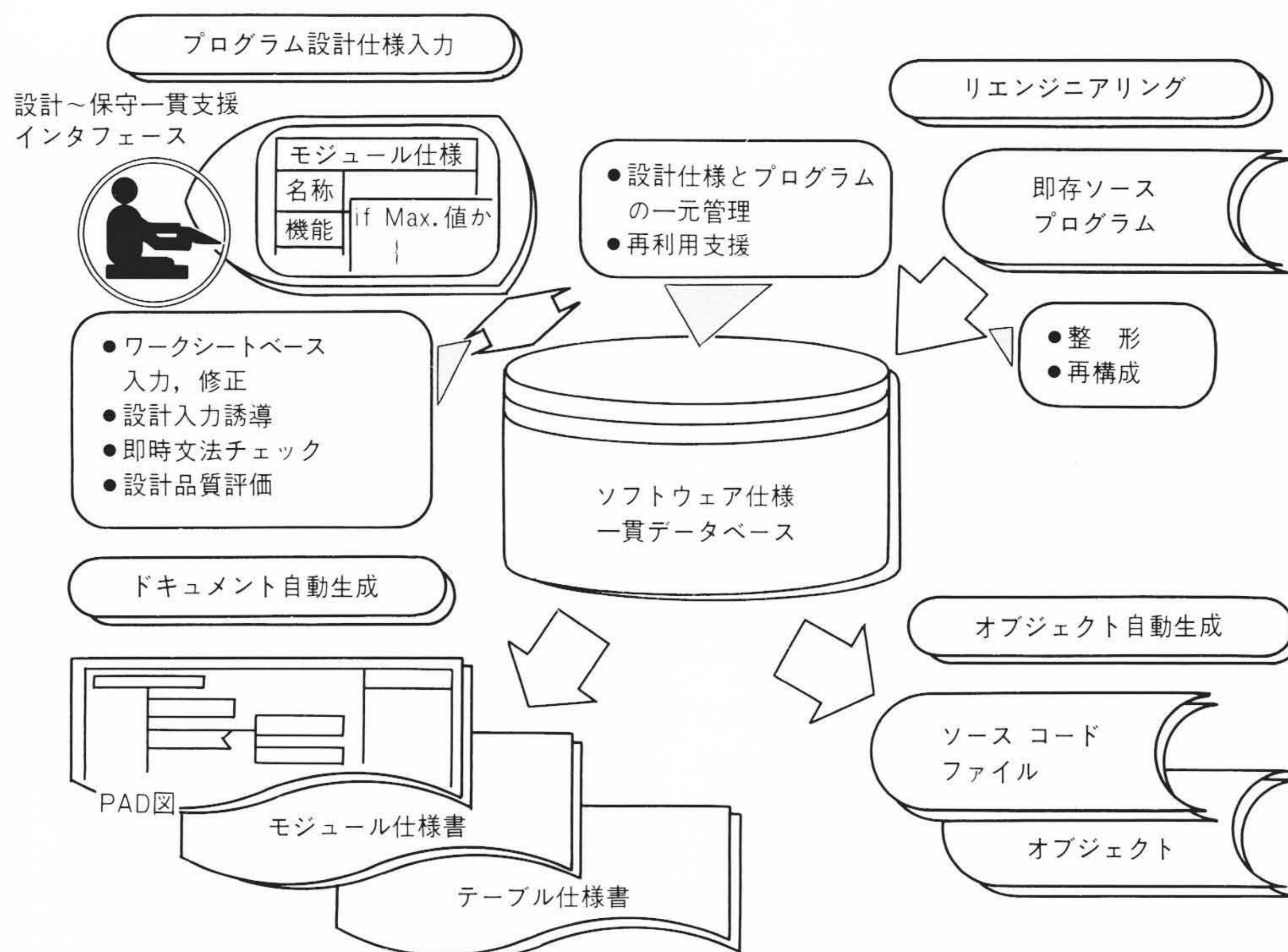


図4 ソフトCADシステムCADRISの機能 ソフトCADシステムCADRISは、ソフトウェアの設計、製作から保守までを一貫支援し、プログラムの設計品質の向上と保守性向上を実現する。

既存ソースプログラムから設計仕様情報を抽出して、再利用が可能なソフトウェアへの再構成を支援する。

CADRISによる設計支援の画面例を図5に示す。

これらの機能により、ドキュメンテーション作業および開発工程間のつなぎのための手作業を大幅に軽減できる。さらに、設計仕様、プログラムのビジュアル化と事前品質検証により、ソフトウェアの作り込み品質を高め、後戻り作業を極少化することができる。

(2) 機能一貫テスト支援システム HITEST⁴⁾

HITESTは、プログラム単体テスト、組み合わせテストおよびシステム疑似オンラインテストを一貫して支援する総合的なテストシステムで、テスト・デバッグ支援機能、テスト十分性評価、検証機能を持っている。

テスト支援機能では、設計仕様のレベルでテストデータを設定できる。また、予想結果とテスト結果を自動照合し、見やすい形でドキュメント出力することができる。テスト十分性評価機能では、被テストプログラムの実行網羅率を自動計測し、テストの十分性、進行状況を一目で判定できるようにしている。

HITESTによるテスト支援の画面例を図6に示す。

3.3 ソフトウェア設計支援専用ツール

一般的な手続きプログラムの設計、製作支援ツールとは別に、データベース処理、マンマシン処理、プロセスデータ処理といった定型化された処理に対する専用支援ツールを用意し、ソフトウェア開発をより容易にしている。各専用支援ツールではソフトウェアの仕様を個々の対象処理ごとに、より要求仕様に近い形(WHAT)で直接記述することによってプログラムの自動生成を行い、ソフトウェア開発、保守を簡単に

できるようにしている。

(1) データベース構築支援ツール ADF/RS SDL⁵⁾

リアルタイム用データベース管理システムADF/RS (Advanced Data Management Facilities for Realtime Application System)のデータベース設計支援機能は、データベース設計情報を一元管理するデータディクショナリを中心とし、データベース設計言語SDL (Structured DataBase Language)により、最適なデータベース設計を誘導、支援する。SDLでは、データベースの論理設計(処理仕様設計)と、物理設計(性能設計)を独立に行えるため、データベース処理プログラムの作成が簡単で、かつデータベースの拡張、変更も容易になる。

(2) マンマシン構築支援ツール Hyproof/MMS⁶⁾

Hyproof/MMS (Hyper Programming System by Object Oriented Formula/Man-Machine System Support)は、マンマシンシステムを「何を実現したいか」というWHATレベルで定義すれば、要求されるマンマシン処理プログラムを自動生成する、オブジェクト指向ベースの制御用第四世代言語システムである。これを使えば、ユーザーは「どうプログラミングするか」というHOWを意識することなくマンマシンシステムを容易に構築できる。

また、GUIを採用した会話形の作画システムにより、CRT画面を見たイメージそのままで作成していくことができ、計算機の専門家でないエンドユーザーによる画面の追加、メンテナンスも可能にしている。

(3) プロセスデータ処理仕様定義ツール PDMS/PDDL

PDDL (Process Data Description Language)は、大量のプロセス入出力点仕様を簡単に効率よく定義する言語である。

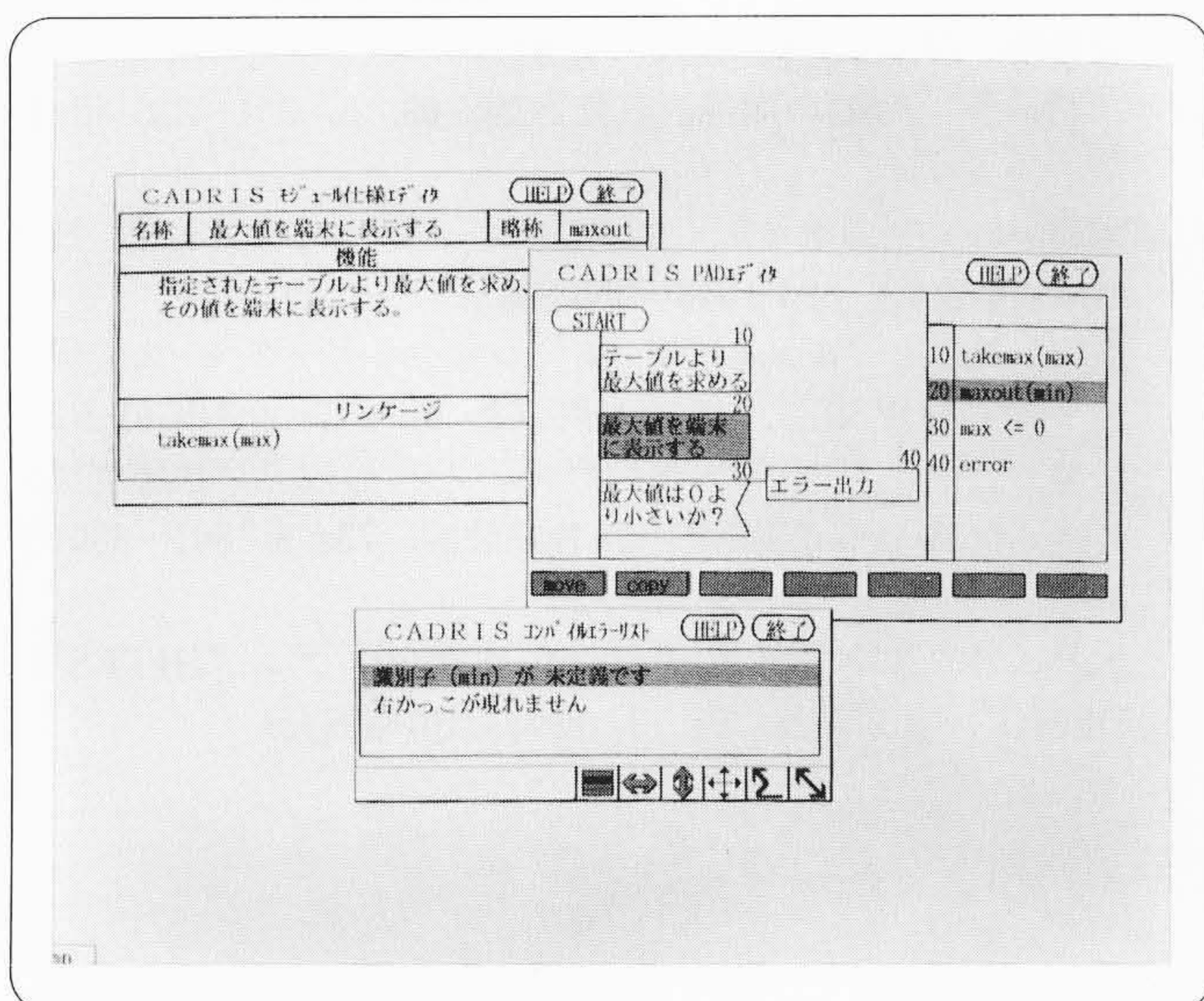


図5 ソフトCADシステムCADRISの画面例 図表形式による入力の誘導と設計仕様上でのコンパイル、およびコンパイルエラーの対応箇所表示により、設計と製作が誤りなく行える。

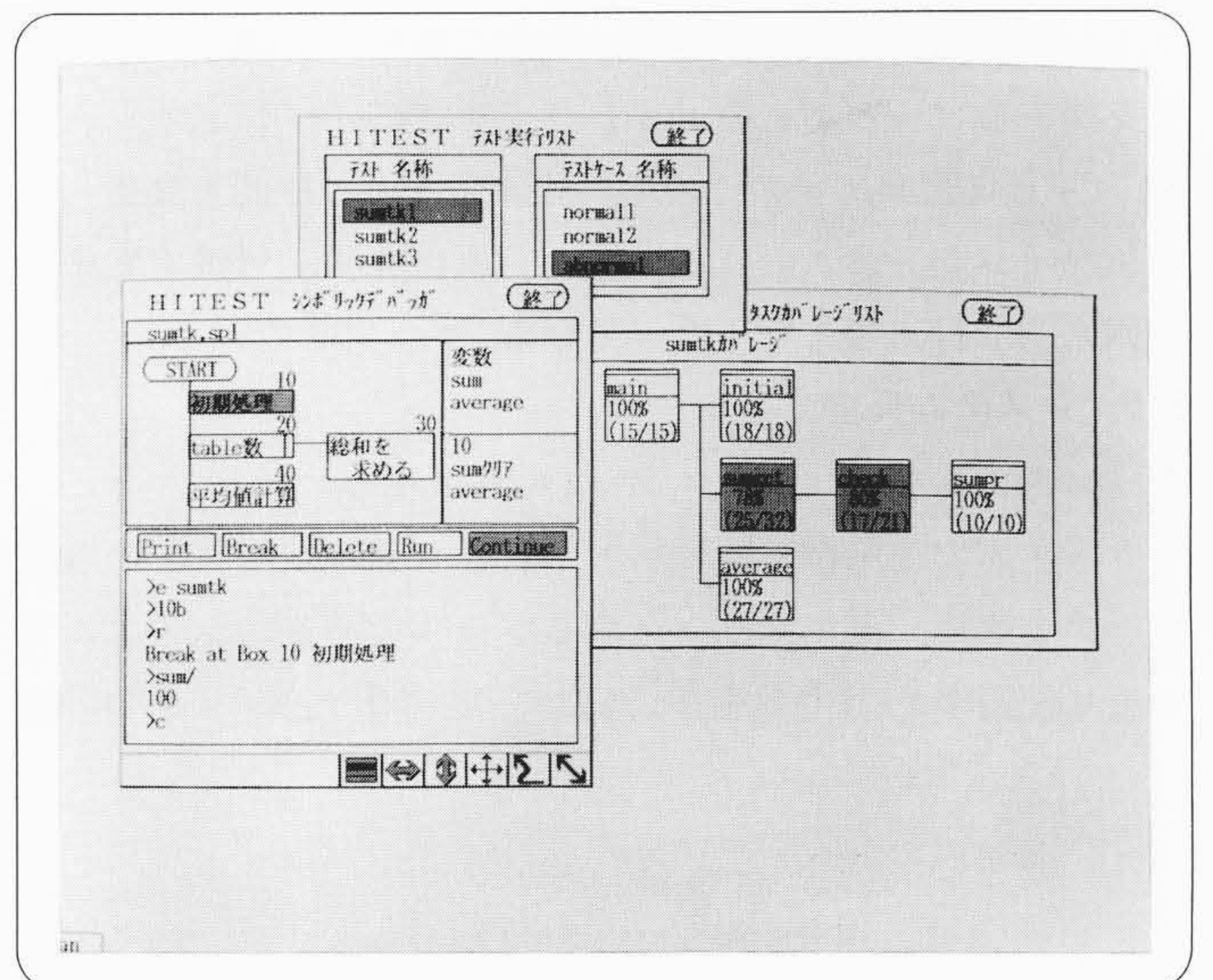


図6 機能一貫テスト支援システムHITESTの画面例 テスト結果の照合が容易に行え、また、テストカバレッジ率測定機能により、ソフトウェアテストの十分性を検証できる。

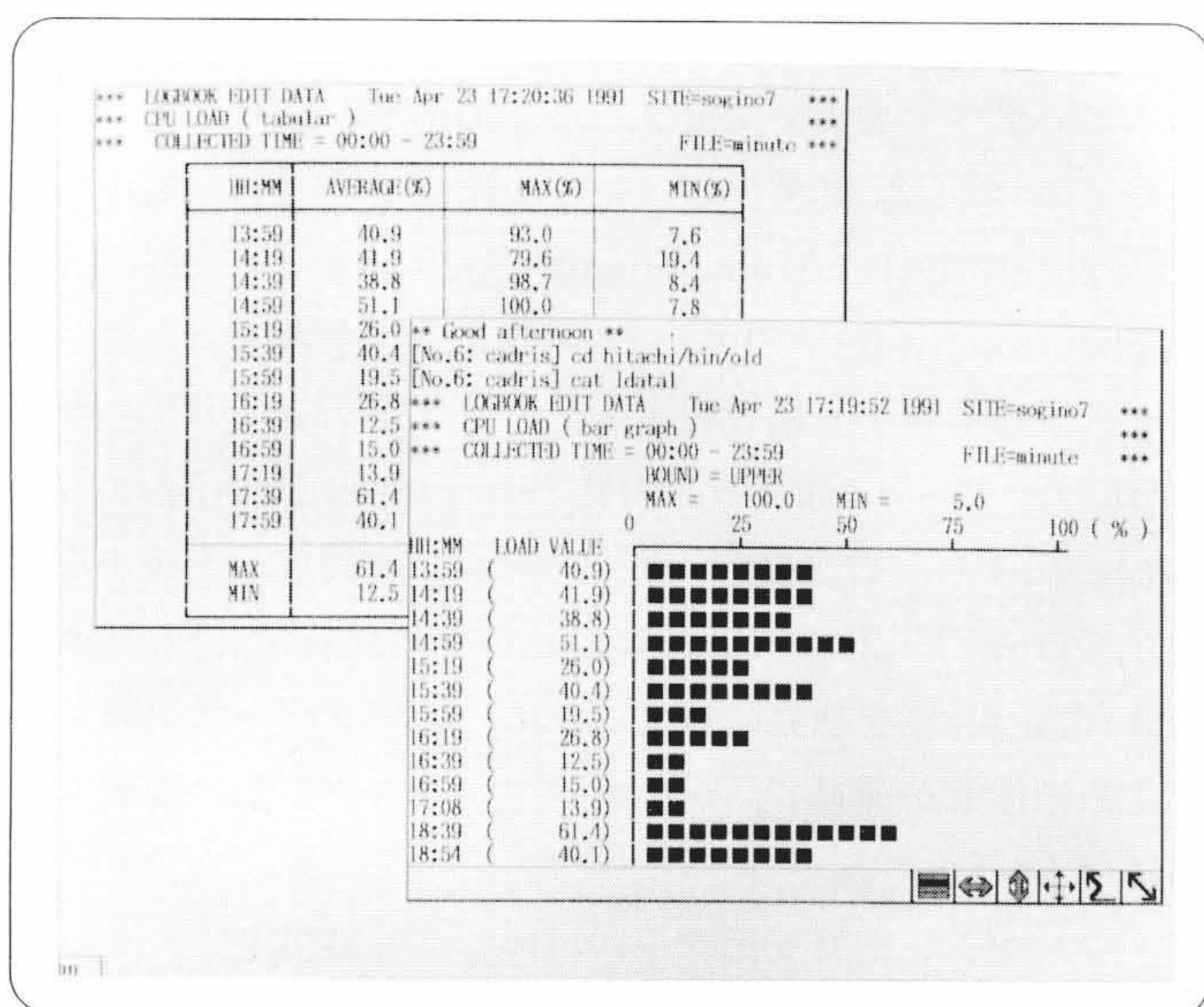


図7 システム稼動状況モニタLOGBOOKの画面例 システムの稼動状況をジャーナル履歴、統計情報などの見やすい形で編集表示できるため、システムの障害解析・予防保全を容易に行える。

PI/O(Process Input/Output)への入出力を論理レベルでサポートするために、PI/Oの物理仕様の定義を行う。また、プロセスデータをアプリケーションプログラムが扱いやすいレベルに変換するための、データ加工処理仕様などの定義も行える。

3.4 オンライン性能評価、デバッグ支援システム DEMO⁷⁾

システムの大規模化、複雑化に伴い、組み合わせテスト以降、総合テスト、モニタランに入ってから発生した機能的、性能的なトラブルは、システムの複数の構成要素が複雑に絡み合い、その原因究明、対策には、多くの時間と高度な知識や経験が必要とされていた。DEMO(Determinate Evaluation, Monitoring and Output system)は、こういったオンライン計算機システムの機能、性能の解析、評価、デバッグを支援するツールである。オンラインでのイベントトレースや、CPU負荷率、タスクの応答性測定などの機能を具備し、オンラインシステムの状況把握から原因究明、対策立案までを体系的に行えるようになっている。

3.5 システム保守支援ツール

(1) システム稼動状況モニタ LOGBOOK

システムの運用後の稼動状況、各種資源の使用状況の履歴データを収集、モニタリングするツールである。システムのメモリ使用率やCPU負荷率のほか、アプリケーションの各種

イベント情報の履歴もジャーナルデータとして蓄積し、見やすい形で編集表示できる。これにより、システム障害を引き起こす可能性のある要因や兆候を事前に検出し、システムの予防保全を図ることができる。LOGBOOKによるジャーナル履歴、統計情報の出力画面例を図7に示す。

(2) システム障害解析支援ツール HIDAS

システム障害は、いったん発生すると、ソフトウェア、ハードウェアの要因が複雑に絡み合っていて、その原因究明が困難であることが多い。HIDAS(HIDIC Diagnosis Assistant System)は、システム障害発生時、原因究明に必要なデータを即時に収集し、用途に見合った形で編集し出力する。これにより、障害原因の的確な把握と迅速な対応を可能にしている。

4 おわりに

ソフトウェア開発、保守支援システムResolve Iでは、従来から進めてきた「ソフトウェア設計、製作、テストのCAD/CAM/CAT化」の適用経験を踏まえ、さらにこれを一歩進めて、ユーザーフレンドリーな「脱プログラミング言語化」を目指し、かつソフトウェア設計から保守までの一貫支援を図った。これにより、保守性の高いソフトウェアをより信頼性高く、かつ効率よく開発することが可能となった。

今後も、さらに適用経験をフィードバックしながら、より高信頼度のソフトウェアを、より短期間で開発するための支援体系を充実、整備していく考えである。

参考文献

- 1) 森, 外: HIDIC V90/5シリーズソフトウェア開発・保守支援システム, 日立評論, **70**, 5, 535~539(昭63-5)
- 2) 林, 外: 情報制御システムの核「HIDIC V90/5シリーズ」, 日立評論, **73**, 8, 787~794(平3-8)
- 3) 二村, 外: PADの開発, 日立評論, **68**, 5, 351~355(昭61-5)
- 4) 大島, 外: 制御用ソフトウェア機能一貫テストシステム, 日立評論, **62**, 12, 893~898(昭55-12)
- 5) 廣田, 外: リアルタイム用リレーショナルデータベース管理システム“ADF/RS”, 日立評論, **68**, 5, 403~408(昭61-5)
- 6) 大脇, 外: オブジェクト指向形マンマシンシステム構築支援システム—Hyproof/MMS—, 日立評論, **73**, 8, 801~806(平3-8)
- 7) 大島, 外: オンラインデバッグ支援システム“HITEST-DEMO”, 日立評論, **66**, 3, 207~210(昭59-3)