

解説



分散開発環境

1. 分散開発環境：新しい開発環境像を求めて†

青山 幹 雄†

1. はじめに

ソフトウェア開発環境の構造が大きく変貌しようとしている。従来の大型ホストを中心とする集中開発環境からワークステーションとネットワークを基盤とする分散開発環境への移行である。ソフトウェア実行環境のオープン化、ダウンサイジング化が、それ自身が巨大で複雑なソフトウェアシステムである開発環境にも大きな変化をもたらしている³⁹⁾。しかし、このような変化の背景には、情報処理技術にとどまらず、ソフトウェア開発のあり方をも変える組織的、社会的要因が指摘されている。

本稿では、まず、分散開発環境の出現の背景にある要因を分析する。次に、このような要因に対し、分散開発環境がもたらす効果と問題点を論ずる。さらに、この問題点を解決するためのアプローチとその技術について、具体的事例を交えて説明する。最後に、今後の課題についてふれる。

2. 分散開発環境とは

分散開発環境は、通常、図-1 に示すような二つの意味で用いられている。ひとつは、従来の集中ホストマシンに代わり、ワークステーション(WS)とネットワークからなる分散処理による開発環境(以下、分散処理環境と呼ぶ)である。一方、開発組織そのものが、地理的に離れた複数の開発拠点に分散する開発環境がある。これを、分散組織環境と呼ぶ。両者に共通の場合に、分散開発環境と呼ぶ。実際には、両方の分散形態が相互に推進力となり、分散処理環境上で分散組織による開発へ移行している。

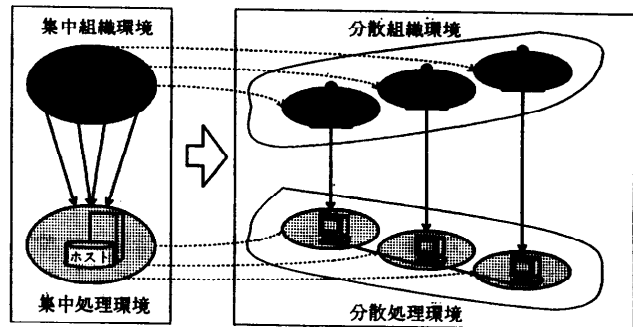


図-1 分散開発環境

3. 分散化の背景

ソフトウェア開発環境が分散する背景には、次のような要因が指摘されている。

3.1 技術的背景

(1) 情報処理技術の変化

コストパフォーマンスの優れた WS と高速ネットワーク技術の発展を核として、今や、一人一人のソフトウェア技術者が一台の WS を占有する一人一台の WS 環境の時代になっている^{49), 60)}。さらに、オフィス用、家庭用、携帯用と、一人で三台を使い分けることも可能である。このような分散開発環境の上に、CASE (Computer-Aided Software Engineering), GUI (Graphical User Interface), グループウェアなどの優れたソフトウェア資産が蓄積されている。このようなソフトウェア資産を異なるハードウェアや OS 上でも利用するために、基盤技術を標準化する重要性が認識されている。ソースレベルでのインタフェースである API (Application Program Interface) や各種プロトコルの国際標準や de facto standard が形成され、オープンアーキテクチャと呼ぶ、ハードウェアに依存しないソフトウェア環境が構築されるに至った。

† Distributed Development Environment : A New Paradigm of Software Development Environments by Mikio AOYAMA (Fujitsu Limited). Internet: mikio@miki.nakahara.fujitsu.co.jp

† 富士通(株)複合交換機事業部

(2) 開発環境に対する要求の変化

従来の TSS (Time Sharing System) による集中処理環境におけるレスポンス時間がプログラム生産性に大きな影響を及ぼすことが実験的に指摘され、生産性向上のためにレスポンス時間の短縮が要求された¹¹⁾。分散処理環境では、この問題を経済的に解消できる。さらに、WS の豊富な処理能力を活用し、開発者の能力を最高に発揮できる環境が望まれている。たとえば、マルチウィンドウシステムは、個人レベルでの並行作業を可能にする。また、高解像度ディスプレイによる GUI の開発は、図形を活用した、いわゆる視覚的開発などの新しい開発方法の実現を促した。COCOMO モデルと比較した結果、WS の利用により 30% の生産性向上を得たとの報告がある¹²⁾。

3.2 社会的背景

(1) ソフトウェア開発を取り巻く環境の変化

開発するソフトウェア規模の絶えざる増大¹³⁾にともない、ソフトウェア開発要員が依然不足している。多数の開発要員を集中して確保することが経済的にも、社会的にも困難となっているため、地域的に分散して開発する、分散組織環境へ移行している。筆者らが従事している通信ソフトウェアの開発では、開発要員の 50% 以上が遠隔地に分散している。

(2) ワークスタイルの変化

Drucker は、21 世紀のワークスタイルは従来の階層的組織における指揮命令型からチームやグループを中心とするネットワーク型組織へ移行すると指摘している¹³⁾。さらに、テレコミュニティング

年	60年代	70年代	80年代	90年代
利用環境	バッチ (コンピュータ室)	TSS (オフィス)	分散開発環境 (机上・どこでも)	分散開発環境/一人一台WS (机上・どこでも)
コンセプト	● 知能洋大ワークショップ	● Dynabook 構想発表	● CASE ウェア	● グループウェア
ワークステーション	● Xerox Alto ワークステーション	● Apple II	● IBM SUN-1 PC	● SUN Next
ネットワーク/LAN	● ARPANET 稼働開始	● Xerox Ethernet	● CSNET	● SUN SUN-4
基盤ソフトウェア	● AT&T UNIX OS	● Xerox Alto-OS	● Microsoft MS-DOS	● SUN NEWS
開発環境事例		● Xerox Smalltalk-80	● Xerox Cedar 環境	● Apollo Domain
開発環境プロジェクト例		● CMU Andrew Project	● Apollo DSEE 環境	● SUN SUN-3
その他		● MIT Athena Project	● Xerox Mesa 環境	● SUN SPARCstation 1
		● 第1回CASEワークショップ	● Xerox Cedar 環境	● SUN SPARCstation 1
		● 第1回CSCWワークショップ	● Xerox Cedar 環境	● SUN SPARCstation 1

図-3 分散開発環境の歴史

(Telecommuting) やテレワーク (Telework) と呼ばれる^{12), 55)}。在宅勤務, サテライトオフィス, リゾートオフィスなどの新しいオフィス形態が導入されている⁶¹⁾。また, 企業活動の国際化にともない, ソフトウェアの国際分散開発も始まっている。分散組織環境は, ソフトウェア開発に, このような新しいワークスタイルをもたらすと期待されている。

4. 分散開発環境の発展

分散開発環境は, 技術面で, 分散処理の発展に大きく依存してきた。ハードウェア面では WS とネットワーク技術であり, ソフトウェア面では分散 OS などの基盤ソフトウェアの発展である。

図-2 に, 分散開発環境を構成する技術の発展をまとめた。図-3 は, 従来の TSS による集中開発環境から分散開発環境への移行を示す。

4.1 ワークステーションの発展

前川らは, WS を『個人計算環境と分散ネットワーク環境を前提とする高機能計算機』と定義している³⁵⁾。このような WS のアイデアは 1960 年代に Engelbert らによって研究された¹⁹⁾。その後, Xerox Palo Alto 研究所で 1973 年に開発された Alto により, 現在の WS の原形が確立された。1982 年には Appollo 社 (現 Hewlett-Packard 社) や Sun Microsystems 社で, 商用の WS が開発された。

一方, 1981 年に IBM 社が発表した IBM/PC は, いわゆるパーソナルコンピュータ (PC) が実務に広く利用される道を拓いた。初めはスタンドアロンで利用されていた PC も, 近年, ネット

ワークを介してプリンタやファイルなどを共有するようになった。高機能 PC と WS とでは, 性能の差がなくなりつつある。

4.2 ネットワークの発展

1969 年から運用を開始した ARPANET は, パケット交換による WAN (Wide Area Network) の構築技術を確認し, 分散開発環境における資源共有の道を拓いた。その後, 多数の WAN が構築されている⁵⁹⁾。パケット交換の考えは, Alto を接続するための LAN (Local Area Network) として開発された Ethernet に結実し, 各種の LAN にも応用されている⁴⁰⁾。我が国でも, 1980 年代後半から大規模の LAN や WAN が構築されている^{36), 45)}。また, 近年のいわゆるパソコン通信の普及により, 電子メールが, 郵便, 電話や FAX に次ぐ通信メディアとして定着しつつある。

4.3 分散開発環境の発展

WS とネットワークによる分散環境技術の可能性を探る目的で, いくつかの大学や研究機関において, 大規模な分散環境を構築する試みが 1980 年代初めに始まった。たとえば, CMU の Andrew プロジェクト⁴⁴⁾, MIT における Athena (アテナ) プロジェクト⁹⁾, Cambridge 大学における Cambridge Distributed Computing Environment の開発などがある^{11), 47)}。1980 年代後半には, このような実験的分散開発環境の成果がソフトウェア開発現場へ導入された。初期の事例として, Xerox 社の Cedar⁶⁷⁾ と Mesa Programming Environment⁶⁶⁾ や Appollo 社の DSEE (Domain Software Engineering Environment)⁹²⁾ などが知られている。

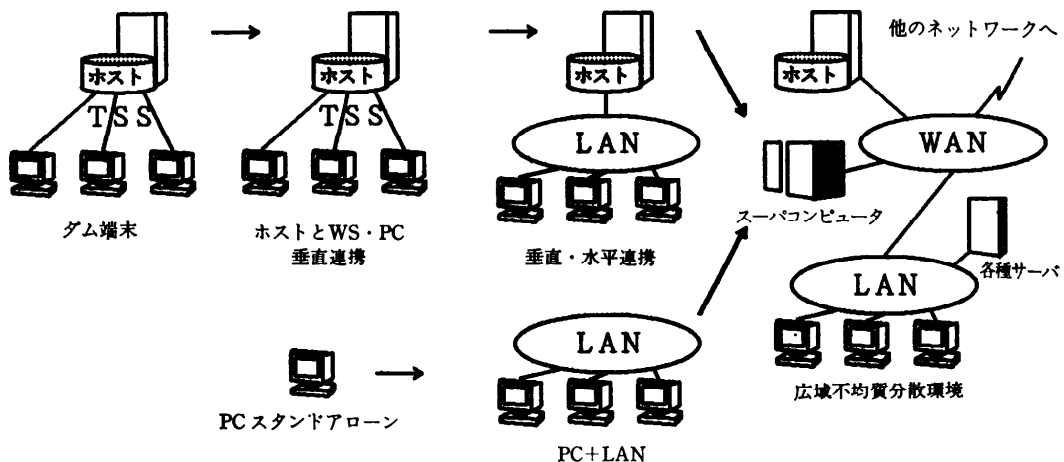


図-3 分散開発環境への推移

5. 分散開発環境の利点と問題点

分散開発環境は、さまざまな利点をもたらすと考えられている。一方、開発者や開発資源が分散することによる問題点も指摘されている。分散開発環境を活用し生産性を向上するためには、分散による利点を伸ばしつつ、その問題点をいかに改善するか懸かっている。

5.1 分散開発環境の利点

分散開発環境は、一人一人の開発者のレベルから開発環境全体まで、次のような改善効果をもたらすと期待されている。

(1) 人に優しい環境

WSの強力な処理能力を活用し、一人一人の技術者の作業環境を改善する。レスポンス時間の改善、GUIによる操作性と情報表現力の向上、ネットワークによる情報の収集と交換の改善など。また、組織分散やサテライトオフィスなどは、通勤条件やオフィススペースなどの生活環境の改善も狙いとしている。

(2) オープンな環境

オープンな環境は、開発環境構築の導入コストや運用コストが低減でき開発規模の増大に応じて拡張や分散が可能となるなど、利用者にとって利点が多い。このため、次のような特性を満たすことが指摘されている^{23), 24), 56)}。

- 互換性 (Compatibility)

国際標準などに準拠し、異なるインプリメンテーション (ベンダ) 間や版数間で OS インタフェースが保証され、利用者の選択の幅が広がるとともに既存ソフトウェア資産やデータが活用できる。

- 可搬性 (Portability)

異なるシステムや OS 間でアプリケーションの移植性が保証され、既存アプリケーション資産が活用できる。たとえば、異なるアーキテクチャのシステム間での API や同一アーキテクチャのシステム間での ABI (Application Binary Interface)。

- スケーラビリティ (Scalability)

同一のアプリケーションが、WS からスーパーコンピュータまで異なる規模 (Scale) のシステム上で動作可能となる。

- 相互運用性 (Interoperability)

異なるアーキテクチャのシステム間でもネット

ワークで相互接続して、コミュニケーションや協調処理などが可能となる。

- (3) リライアブルな環境

組織や各種サービスの機能分散と負荷分散により、特定のサーバの障害などにより開発環境全体が停止するリスクが分散される。この結果、開発作業全体の中断を防止できる。

5.2 分散開発環境の問題点

分散開発環境では、開発作業の主体である人と WS、さらには成果物であるドキュメントや種々の情報が分散する。このため、開発者やグループ間での協調と情報の収集、交換などの作業オーバーヘッドと作業品質の低下が問題となる。Adamsらはこれを次の5つの問題に要約している²⁾。

(1) 複数の人間が関与することによる問題 (Multiple people problems)

(2) 複数のオブジェクト (プログラムやデータ) を扱う問題 (Multiple object problems)

(3) 複数の版数を並行して開発保守する問題 (Multiple release problems)

(4) 複数のマシンが関与することによる問題 (Multiple machine problems)

(5) 複数のツールを利用することによる問題 (Multiple tool problems)

このような問題は、分散処理システムにおけるプロセスの同期や資源の管理問題と多くのアナログがある。たとえば、分散処理システムにおけるプロセッサへの機能分散と同様に、相互に密接な関係のあるプログラム群を異なる拠点で分散開発すると、拠点間のコミュニケーショントラフィックが増大する。また、分散組織環境においては、開発における情報の共有が困難となる。ドキュメント化されていない情報や、紙のドキュメントでは、相互に参照したり、リアルタイムに更新できない。このような、分散開発環境の問題点を実際の開発の中で調査した事例が報告されている。

6. 分散開発環境へのアプローチ

6.1 分散開発環境のモダリ化

分散開発環境の一般的構造モデルはまだ確立されていないが、本稿では、図-4 に示すような3階層モデルで考える。

(1) ハードウェア: WS とネットワークを回輪とする分散処理ハードウェア。

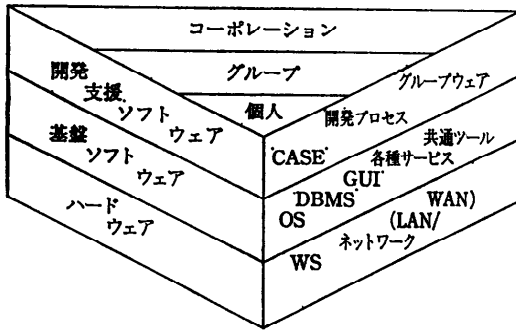


図-4 分散開発環境のモデル

(2) 基盤ソフトウェア：ハードウェア技術を活用するための、OS、ネットワークソフトウェア、ファイルシステムなど、分散開発環境で共通に使用するソフトウェア。

(3) 開発支援ソフトウェア：分散組織による開発活動をソフトウェア工学的アプローチにより支援するソフトウェア。

本稿では、ソフトウェア技術に注目し、基盤ソフトウェアと開発支援ソフトウェア階層を次の三

つの観点から整理する。

(1) 構造モデル

環境をシステムの静的構造からみたモデル。

(2) 処理モデル

環境をシステムの動的挙動からみたモデル。

(3) 開発モデル

環境の利用者、すなわち開発者からみたモデル。

6.1.1 構造モデル

分散開発環境の基盤である分散処理環境の構造モデルが幾つか提案されている。たとえば、IEEE Computer Society の P 1003 委員会で検討されている **POSIX** (Portable Operating System Interface for Computer Environments) の **OSE (Open Systems Environment)**²⁴⁾、OSF (Open Software Foundation) の **DCE (Distributed Computing Environment)**^{60), 57)}、UI (UNIX International) の **ATLAS**⁷³⁾ などである。いずれも、分散開発環境で必要となる各種サービスを統合した階層構造の環境である。例として、図-5 に ATLAS のアーキテクチャを示す。

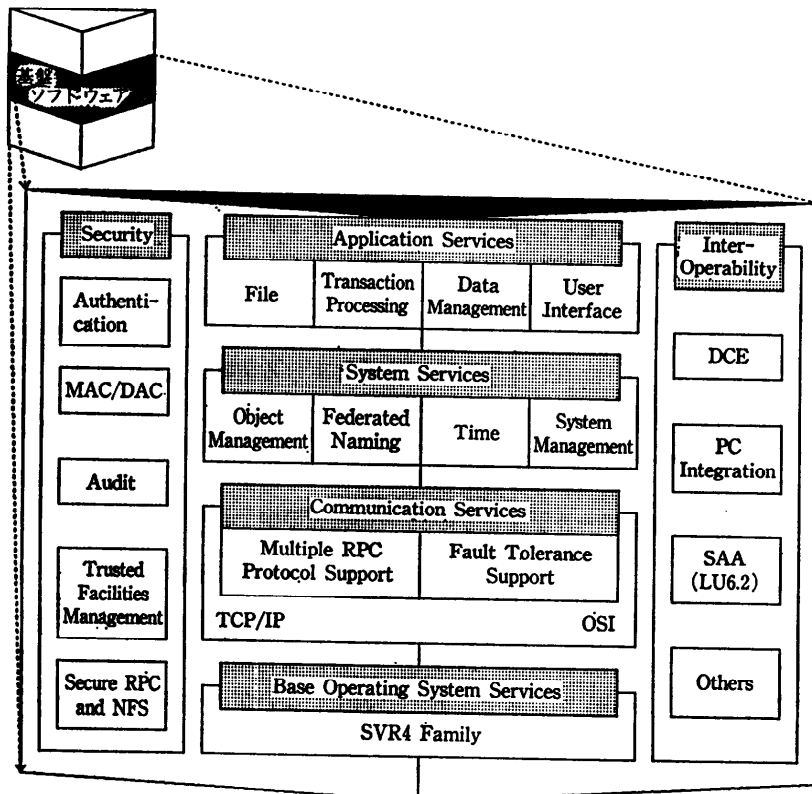


図-5 構造モデルの事例：ATLAS アーキテクチャ

6.1.2 処理モデル

分散開発環境で協調して処理を行うモデルとして、図-6 に示すクライアント/サーバモデル (Client/Server Model) が知られている¹⁴⁾。サービスを受けるクライアントの WS とサービスを提供するサーバあるいは WS との間で、ネットワークを介してメッセージを授受しながらサービスを実行する。このモデルは、分散ファイルシステム³⁴⁾やウィンドウシステム⁶³⁾など、分散処理環境を構成する多くの基盤ソフトウェアで実現されている。

6.1.3 開発モデル

ソフトウェア開発活動を構成する主要素は、開発プロセスとプロダクトである。この観点から、分散開発環境は次の二つのモデルで考えられる。

- プロセスモデル
- プロダクトモデル

(1) プロセスモデル

ソフトウェアの開発過程を形式的に表現する、プロセスプログラミングが提案されている⁵⁸⁾。一方、グループによるソフトウェア開発を支援する枠組みとして CSCW (Computer-Supported Cooperative Work)¹⁷⁾ やグループウェア^{26), 28), 30), 69)}が提案されている。この二つのアプローチは、図-7 に示すように、分散開発環境を構築する上で、プロセス面から考える共通の視点を提供している。

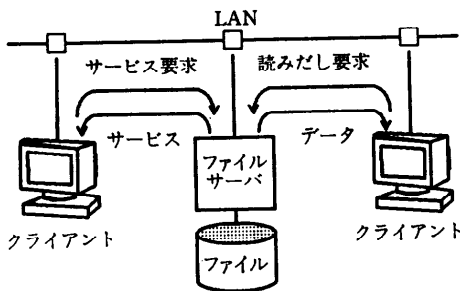


図-6 処理モデル：クライアント/サーバモデル

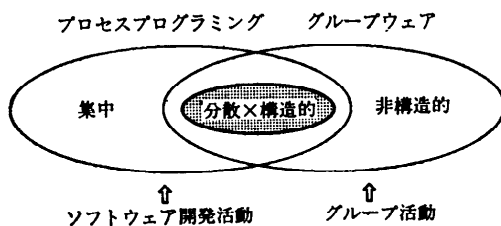


図-7 プロセスプログラミングとグループウェア

まず、開発プロセスと開発環境に相似性があることから⁵²⁾、開発環境を構築するには、その開発活動を分析し、開発プロセスの構造を把握する必要がある。Winograd は、『デザインとはツールそのものを設計することではなく、そのツールの使用を通して引き起こされるワークスタイルの変化を設計することである』と指摘している^{27), 75)}。開発プロセスと開発環境とが適合していなければ、所期の効果を達成できないだけでなく、生産性が低下する恐れさえある。

特に、分散組織環境では、各開発拠点が自律的になるため、開発プロセスが異なる傾向にある。また、各拠点の開発活動は相互に関係するため、その動的相互作用はきわめて複雑となる。このような問題を把握する上で、プロセスモデルは有効なアプローチを提供すると期待される。

一方、グループウェアは、ソフトウェア開発のみならず、グループによる広汎な活動を対象としている。グループウェアを構築するには、グループの構造や活動をモデル化する必要がある。このモデルは、構造的モデルと非構造的モデルに分類される²⁷⁾。ソフトウェア開発の観点から解釈すると、それぞれ、定型的な開発作業と非定型的な開発作業に対応する。したがって、ソフトウェア開発におけるグループウェアの構造的モデルは、開発プロセスそのものにほかならない。

このようなアプローチに基づき、実際の開発プロセスをモデル化する研究が行われている^{4), 25), 42)}。その結果、分散組織環境における開発プロセスの動的挙動が明確になるなどの有効性が報告されている。しかし、オフィス作業そのものがオープンであることから、モデルの適用範囲や限界も指摘されている²²⁾。

(2) プロダクトモデル

プロダクトモデルは、開発プロセスの入出力である成果物に着目する。分散開発環境では、プロダクトの生成、蓄積、参照、交換が分散する。さらに、プロダクトは相互に関係があるため、その管理や変更にともなう影響の波及はきわめて複雑となる。

プロダクトには、ドキュメントやソースコードなどの恒久的なものと会議の議事録など一時的なものがある。分散開発環境におけるプロダクトの構造とその流れを分析することがプロダクトモデ

ル構築の第一歩となろう。実際に、ソフトウェア開発における情報伝達を分析した事例^{45), 48)}や分散開発の管理支援システムの構築に応用した事例⁴⁹⁾の報告がある。さらに、オブジェクト指向DBにより、プロセスとプロダクトを統一的に管理する環境も提案されている³⁷⁾。

また、日本の企業風土では、フォーマルな情報のみならず、インフォーマルな情報の共有も重要であることが指摘されている³⁹⁾。

6.2 分散開発環境の分類

分散開発環境の形態は次のように分類できる。

6.2.1 ネットワーク形態による分類

(1) インハウス分散：集中組織で WS と LAN による分散処理環境を利用する形態。

(2) 都市（地域）内分散：同一都市内あるいはキャンパス内で複数の建物に分散する開発環境。このような環境では中距離の高速ネットワーク MAN (Metropolitan Area Network) により開発拠点が結合される。

(3) 広域分散：複数の都市あるいは国際的に分散した開発拠点を WAN で結んだ環境。

6.2.2 機能と負荷による分散

(1) 機能分散：作業の内容による分散。たとえば、各サーバは、計算、ファイル管理、印刷など異なるサービス（機能）を提供する。

(2) 負荷分散：作業の負荷による分散。たとえば、同一サービスを提供する複数のサーバの分散処理により実行時間を短縮する。

6.2.3 均質分散環境と不均質分散環境

(1) 均質分散：同一アーキテクチャのシステムで構成される分散開発環境。

(2) 不均質分散環境：異なるアーキテクチャのシステムで構成される分散開発環境⁵⁰⁾。

実際には、機能分散と負荷分散を組み合わせた、不均質な広域分散開発環境となる傾向にある。

6.2.4 開発モデルによる分類

組織分散における開発分担の観点から、次の二つに分類される。

(1) 組織機能分散

各拠点で、異なるプロセスやプロダクトを開発する。これは、さらに、次のように分類される。

●垂直（プロセス）分散：開発プロセスによる分散。たとえば、上流工程と下流工程を異なる拠点で開発する。

●水平（プロダクト）分散：開発プロダクトやシステムによる分散。たとえば、アプリケーションやサブシステムごとに異なる拠点で開発する。

(2) 組織負荷分散

各拠点で、同一工程や同一アプリケーションを分担して開発する。

筆者らが従事している通信ソフトウェアの開発においては、まず、下流工程を中心として水平分散を始めた。現在は、上流工程から分散組織開発を行っている。

7. 分散開発環境の基盤技術

分散開発環境を構築する要素技術を、図-4 の各階層に分けて概観する。

7.1 ハードウェア

7.1.1 ワークステーション (WS) とサーバ

WS は 80 年代初めに登場して以来、その処理能力が 10 年間で約 100 倍に向上した。特に、80 年代半ばに RISC (Reduced Instruction Set Computer) を採用した WS の登場により処理能力が飛躍的に向上した。一方、WS の形態も多様化する傾向にある。WS とサーバの分化、ディスクレス WS や X ウィンドウをサポートする X 端末など低価格の追求、ラップトップ WS による可搬性の追求、マルチプロセッサによる高処理能力の追求などがある。また、WS 上でのソフトウェア規模の増大に対応して、主メモリ容量やディスク容量も増大の一途にある。

分散処理環境を構成するサーバとして、コンピュータリングサーバ、ファイルサーバ、プリンタサーバ、コミュニケーションサーバなどが接続される。各開発者は、WS や X 端末などからネットワークを介してこれらのサーバを利用する。

7.1.2 ネットワーク

インハウス分散環境では、Ethernet などの LAN が、通信プロトコルの標準化とあいまって定着している⁴⁶⁾。大規模インハウス分散環境や都市内分散環境では、複数の LAN を高速の基幹 LAN により接続する、階層構造ネットワークを採る。基幹 LAN として、光ケーブルによる FDDI (Fiber Distributed Data Interface) など 100 Mbps 以上の LAN が普及しつつある。一方、広域分散環境では、拠点間を結ぶ WAN が要となる。高速 WAN にはパケット交換ネットワークが^{40), 49)}、パソコン

通信など比較的小量データの通信ではモデルによる通信が利用されている。最近では、ISDN や高速デジタル回線とルータやブリッジを組み合わせた LAN 間接続など、多様な形態の WAN が構築されている⁴⁹⁾。このような大規模ネットワークの性能予測や評価は困難であり、種々の研究が進められている²⁰⁾。

今後は、急速に向上する WS やサーバの処理能力とマッチするネットワークの構築が重要な課題となろう。また、音声とデータを統合する、IVD (Integrated Voice and Data)-LAN⁴⁵⁾ や無線 LAN, 数百 Mbps での交換を行う広帯域 ISDN, あるいは大規模ネットワークの管理システムなどの開発も期待されている。

7.2 基盤ソフトウェア

基盤ソフトウェアは、機能面から、次の三つのシステムに分類される。

- (1) オペレーティングシステム (OS)
- (2) ネットワークシステム
- (3) ファイル/データベースシステム

7.2.1 オペレーティングシステム

WS の OS として UNIX が広く普及している。UNIX をベースとする分散 OS も開発されている^{11), 11)}。一方、PC をネットワーク環境で利用するための、いわゆるネットワーク OS として、NetWare や LAN Manager などが利用されている。今後は、異機種が混在するオープンな分散処理環境が普及すると思われる。このような、分散処理環境のオペレーションを支援するためには、環境の透過性 (transparency) を保証する必要がある。

クライアント/サーバモデルを実現するための機構として、遠隔手続き呼び出し (RPC: Remote Procedure Call)⁷⁾ が広く利用されている。RPC は、従来の単一コンピュータ内での手続き呼び出しを複数のコンピュータに拡張したものである。初期の分散処理環境で RPC を実現した例として、Grapevine¹¹⁾ や Cedar⁶⁷⁾ が知られている。また、異機種分散処理環境では、RPC のプロトコルが異なる場合がある。種々の実現方法があるが⁵¹⁾、HRPC (Heterogeneous RPC), DCE や ATLAS などでは、IDL (Interface Description Language) により RPC 間の整合をとる方法が提供されている。

7.2.2 ネットワークシステム

ネットワークシステムは、コミュニケーションを中心として、次のような多様なサービスを提供している。

(1) コミュニケーションサービス⁷⁰⁾。

- 電子メール (Electronic mail) サービス
- 電子会議 (Electronic conference) サービス
- 電子ニュース (Electronic news) と電子掲示板 (BBS: Bulletin Board System) サービス

(2) ネットワークのオペレーションサービス
ネットワークに接続された他の WS あるいはサーバでコマンドの実行など基本的な処理を行うために、次のようなサービスが提供されている。

- ファイル転送 (File transfer)
- 遠隔ログイン (Remote login)
- 遠隔コマンド実行 (Remote command execution)

(3) ネットワークのリソース管理サービス

分散処理環境を運用するためには、分散した WS やサーバ、あるいはそのユーザなどの情報を管理する必要がある。このような情報を格納する分散データベースとその管理システムはディレクトリサービスあるいは名前サービスと呼ばれる。

(4) タイムサービス

分散した WS やサーバ間の同期をとるソフトウェアベースのサービス

(5) セキュリティサービス

次のようなセキュリティ確保のためのサービスが開発、あるいは提供されている。

- 認証 (Authentication)
- 許可 (Authorization)
- データの完全性 (Data integrity) の保証
- データの機密保護 (Data confidentiality)
- 署名確認 (Digital signature)

7.2.3 ファイル/データベースシステム

ソフトウェア開発における開発情報を蓄積し共有するために、従来のデータベースシステムでは提供されていなかった要件が指摘されている³⁰⁾。分散開発環境では、このような要件に加え、分散した開発情報の共有を支援する必要がある。このため、分散ファイルシステム (DFS: Distributed File System)^{18), 34)} が開発されている。

DFS はネットワークで接続された WS 相互、あるいは、WS とサーバ間でファイルを共有する

ための機構である。実用されている DFS としては、Sun Microsystems 社の NFS (Network File System), AT & T 社の RFS (Remote File System), CMU で開発され DCE に採用されている AFS (Andrew File System) などがある。NFS と RFS は、ユーザがファイルの物理的位置を意識する必要がない位置透過性を実現している。しかし、各 WS が共有するファイルシステムは異なる場合がある。一方、AFS ではネットワーク全体が単一のファイルシステムとなる。このファイルシステムは、WS 間で共有する Shared Name Space と型 WS ごとに所有する Local Name Space に分割して利用する。University of Newcastle で開発された UNIX United や UC Berkeley で開発された Sprite も単一のファイルシステムを実現している。

7.3 開発支援ソフトウェア

開発支援ソフトウェアは、特定の開発プロセスを支援するソフトウェアと全開発プロセス共通に支援するソフトウェアに分類できる。前者の例として、CASE 環境を取り上げる。後者の例として、開発プロセス支援環境とグループウェアを概観する。さらに、試験支援環境と各種支援ソフトウェアを分散開発環境に統合するための統合支援について紹介する。

7.3.1 分散 CASE 環境

CASE 環境⁵⁴⁾が、下流工程を支援するローヤ CASE を中心として、分散処理環境に対応しつつある⁴³⁾。メインフレームあるいは UNIX サーバをサーバとして、WS がクライアントとなるクライアント/サーバモデルの環境である。たとえば、通信ソフトウェア開発の上流工程から一貫支援する分散 CASE 環境が開発されている^{16), 21), 48), 68)}。

分散開発環境で特に問題となる特定の作業を支援する環境も開発されている。たとえば、DSEE のプログラムビルダでは、ネットワーク内の複数のサーバが並行してプログラムをリンクする³³⁾。また、NSE (Network Software Environment)²⁾ や CVS (Concurrent Versions System)⁶⁾ は、分散開発環境で複数の開発者が同一プログラムを並行して更新する作業を支援する。

7.3.2 開発プロセス支援とグループウェア

(1) 開発プロセス支援環境

開発プロセス支援環境とは、開発プロセスやプロダクトを各開発者の WS に表示し、各工程の作業に適したツールを自動的に起動したり、各生成物が求める条件を満足しているか判定したりすることにより、統一的な作業環境を提供する^{25), 70)}。たとえば、稲田らは関数型プロセス記述言語 PDL (Process Description Language) とそのインタプリタにより、JSD (Jackson System Development) の開発支援環境を開発した²⁵⁾。

実際の開発プロセスを記述する試みも行われている。望月らは、HFSP (Hierarchical and Functional Software Process) モデルにより、実時間システムの開発プロセスを記述している⁴²⁾。筆者らは、木構造チャートを拡張した言語により通信ソフトウェアの開発プロセスを記述した⁴⁾。従来、作業標準として自然言語で記述されていた開発プロセスをより形式的に表現できるため、ライフサイクルを通して作業過程や成果物が明確となった。

(2) グループウェア

分散開発環境にグループウェアを導入することは自然なアプローチである。分散処理環境を利用して、分散組織によるコミュニケーションやコーディネート支援する。

これまで、電子メールや電子会議などのコミュニケーションを中心とした非構造的グループウェアの事例が多数報告されている³⁸⁾。今後は、設計理由 (Design Rationale) の記録⁶⁹⁾やグループレビュー⁸⁾など、ソフトウェア開発における固有の問題を支援するグループウェアの研究開発が期待される。詳細は、文献69)を参照願いたい。

7.3.3 試験支援環境

組み込み型ソフトウェアの試験では、ターゲットシステム上の試験環境がメモリや処理能力などの制限を受け、十分でないことが多い。ターゲットシステムと WS が LAN を介して連携し、試験を効率的に実行する環境が実現されている⁴⁰⁾。

また、筆者らは、分散処理環境と分散処理システムとのアナログに着目し、分散処理システムのシミュレータを分散処理環境で実現した¹⁰⁾。

7.3.4 分散開発環境の統合

各種支援ソフトウェアを分散処理環境に統合するための支援環境が開発されている。たとえば、

ESPRIT プロジェクトの PCTE (Portable Common Tool Environment)⁷²⁾ や OMG (Object Management Group) の OMA (Object Management Architecture)⁶⁴⁾ などが知られている。

8. 今後の課題

分散開発環境への移行は、今後ますます進むと思われる。しかし、分散処理そのものが発展途上にあるので、分散開発環境も研究開発すべき課題が多い。以下、幾つかのトピックスをあげるが、詳細は文献 62) を参照願いたい。

(1) 分散処理の課題：分散処理環境は分散処理システムと多くのアナログがあるため、分散処理システムの研究開発成果を分散処理環境へ取り込むことが期待される。

(2) グループウェアや開発プロセスなどのソフトウェア工学的アプローチの課題：実際のソフトウェア開発プロセスはきわめて複雑であり、今後、理論と実践の両面からモデル化の研究が必要である。

(3) 分散並行開発：分散開発は、同時に、並行開発を促している^{4), 21)}。複数の分散したグループが並行して開発する形態である。開発ライフサイクル短縮のために、このような開発プロセスの研究を期待したい。

(4) 大規模（広域）分散開発環境の運用管理：分散開発環境自体が巨大で複雑となっているため、その運用管理のコストが増大している。開発環境の保守と運用の自動化、省力化の推進が必要である。

(5) セキュリティ：ネットワークにウィルスが侵入すると一挙に感染が広がる危険があることなど、セキュリティは分散開発環境の重要課題である。米国における 5 段階セキュリティ基準 [74] の策定など、今後、分散開発環境を構築する必要条件となろう。

(6) 標準化：不均質でオープンな分散処理環境への移行にともない、分散処理環境の各階層における標準化はますます重要となる。OS、通信プロトコル、DB、GUI などの標準化が進められている。

9. まとめ：窓から世界が見える

分散開発環境の現状を展望した。分散開発環境は、強力な処理能力と円滑なコミュニケーション、

分かりやすいインタフェース、使い勝手のよいツール群などによりソフトウェア開発のワークスタイルを変革することが期待されている。それは、コンピュータ資源の利用を最適化する環境から人間の能力を最高に発揮できる環境への転換である。開発者にとって心地良い環境であり、時間や空間に制約されない環境でありたい。あたかも高層ビルの窓から街を見おろすように、一人一人の WS のウィンドウから、ソフトウェア開発の全体像が見える環境。このような開発環境を実現するためには、まだまだ課題が多い。この分野における、今後の活発な研究開発を期待したい。

謝辞 本稿の執筆に際し、有益なご助言をいただいた三菱電機(株)水野忠則氏ならびに本誌編集委員会各位に感謝いたします。また、資料調査と内容討議にご協力いただいた関係各位に感謝いたします。

参 考 文 献

- 1) Acetta, M. et al.: Mach: A New Kernel Foundation for UNIX Development, Proc. USENIX Summer Conf., pp. 93-112 (1986).
- 2) Adams, E. W., Honda, M. and Miller, T. C.: Object Management in a CASE Environment, Proc. 11th ICSE, pp. 154-163 (1989).
- 3) 青木, 小池, 中谷: 日本企業グローバル化の研究, PHP 研究所 (1989).
- 4) Aoyama, M.: Distributed Concurrent Development of Software Systems: An Object-Oriented Process Model, Proc. IEEE COMPSAC '90, pp. 330-337 (1990).
- 5) 青山, 小池: 交換ソフトウェアの分散並行開発支援環境, 電子情報通信学会 交換システム研究会, No. SSE 90-25, pp. 7-12 (1990).
- 6) Berliner, B.: CVS II: Parallelizing Software Development, Proc. USENIX Winter Conf., pp. 341-351 (1990).
- 7) Birrell, A. D. and Nelson, B. J.: Implementing Remote Procedure Calls, ACM Trans. Computer Systems, Vol. 2, No. 1, pp. 39-59 (1984).
- 8) Brothers, L., Sembugamoorthy, V. and Muller, M.: ICICLE: Groupware for Code Inspection, Proc. ACM CSCW '90, pp. 169-181 (1990).
- 9) Champine, G. A., Geer, Jr., D. E. and Ruh, W. N.: Project Athena as a Distributed Computer System, IEEE Computer, Vol. 24, No. 9, pp. 40-51 (1990).
- 10) Chang, C. K., Chang, Y. F. and Aoyama, M.: A Real-Time Distributed Simulator of PBX with Software Reuse, J. Simulation, Vol. 54, No. 2, pp. 71-79 (1990).
- 11) Coulouris, G. F. and Dollimore, J.: Distributed

- Systems, Addison-Wesley (1988) [水野忠則 (監訳): 分散処理システム, 電気書院 (1991)].
- 12) Cross, T. B. and Raizman, M. B.: Telecommuting: The Future Technology of Work, Richard D. Irwin, Inc. (1986) [松下 温 (訳): テレコミュニケーション, 近代科学社 (1988)].
 - 13) Drucker, P.: The Coming of the New Organization, Harvard Business Review, Jan.-Feb. 1988, pp. 45-53 (1988) [高度情報技術と情報ネットワークがもたらす未来型組織の構想, DIAMONDハーバードビジネス, Apr.-May 1988, pp. 16-30 (1988)].
 - 14) Francis, B.: Client/Server: The Model for the '90s, Datamation, 1990. 2. 15号 (1990) [クライアント・サーバ・システム, 日経コンピュータ, 1990, 4. 23号, pp. 169-174 (1990)].
 - 15) 福田, 井上, 津田: 遠隔地ソフトウェア開発の実験, 情報処理学会ソフトウェア工学研究会, No. 71-7 (1990).
 - 16) 福山, 米田, 浅見, 桂木: 分散形ソフトウェア開発環境の構成方式, NTT R & D, Vol. 39, No. 7, pp. 973-986 (1990).
 - 17) Greif, I.(ed.): Computer-Supported Cooperative Work, Morgan Kaufmann (1988).
 - 18) Gifford, D. K., Needham, R. M. and Schroeder, M. D.: The Cedar File System, Comm. ACM, Vol. 31, No. 3, pp. 288-298 (1988).
 - 19) Goldberg, A. (ed.): A History of Personal Workstations, ACM Press (1988) [村井 純 (監訳): ワークステーション原典, アスキー出版局 (1990)].
 - 20) Gusella, R.: A Measurement Study of Diskless Workstation Traffic on an Ethernet, IEEE Trans. Comm., Vol. 38, No. 9, pp. 1557-1568 (1990).
 - 21) Harrison, W. H., Osher, H. and Sweeney, P. F.: Coordinating Concurrent Development, Proc. ACM CSCW '90, pp. 157-168 (1990).
 - 22) Hewitt, C.: Offices are Open Systems, ACM Trans. Office Inf. Syst., Vol. 4, No. 3, pp. 271-287 (1986).
 - 23) Hubley, M.: Distributed Open Environments BYTE, Vol. 16, No. 12, pp. 229-238 (1991).
 - 24) IEEE: Draft Guide to the POSIX Open Systems Environment, P 1003.0, IEEE Computer Society (1991).
 - 25) 稲田, 萩原, 井上, 鳥居: ソフトウェア開発過程の形式化とその詳細化による支援システムの作成—JSD を例として—, 電子情報通信学会論文誌, Vol. J72-D-1, No. 12, pp. 874-882 (1989).
 - 26) 石井 裕: グループウェア技術の研究動向, 情報処理, Vol. 30, No. 12, pp. 1502-1508 (1989).
 - 27) 石井 裕: グループウェアのデザイン, bit, Vol. 23, No. 3, pp. 273-283 (1991).
 - 28) Johansen, R.: Groupware, The Free Press, (1988) [合津 泉 (訳): グループウェア, 日経 BP 社 (1990)].
 - 29) 亀田靖浩: 大規模ソフトウェア開発の現状と課題, 電子情報通信学会誌, Vol. 74, No. 5, pp. 457-460 (1991).
 - 30) 川越, 岸, 鶴岡: オブジェクト指向データベースと CASE, 情報処理, Vol. 32, No. 5, pp. 577-585 (1990).
 - 31) 川尻, 中村, 清兼, 青山: リアルタイム分散並行開発環境: ICAROS, 情報処理学会ソフトウェア工学研究会, No. 73-15 (1990).
 - 32) Leblang, D. B. and Chase, Jr., R. P.: Computer-Aided Software Engineering in a Distributed Workstation Environment, Proc. ACM Symp. on Practical Software Development Environments, pp. 104-112 (1984).
 - 33) Leblang, D. B. and Chase, Jr., R. P.: Parallel Software Configuration Management in a Network Environment, IEEE Software, Vol. 4, No. 6, pp. 28-35 (1987).
 - 34) Levy, E. and Silberschatz, A.: Distributed File Systems: Concepts and Examples, ACM Comput. Surv., Vol. 22, No. 4, pp. 321-374 (1990).
 - 35) 前川 守 (編): ワークステーション, 丸善 (1990).
 - 36) 松方 純: 大学における大規模 LAN の構築, 情報処理学会論文誌, Vol. 30, No. 1, pp. 25-35 (1989).
 - 37) Matsumoto, Y. and Ajisaka, T.: A Data Model in the Software Project Database KyotoDB, JSSST Adv. Software Sci. Tech., Vol. 2, pp. 103-121 (1990).
 - 38) 松下 温 (編著): グループウェア入門, オーム社 (1991).
 - 39) 松崎, 中村, 平田, 木村: ホストなき世界の到来, 日経コンピュータ, 1991.10.7号, pp. 48-106 (1991).
 - 40) Metcalfe, R. M. and Boggs, D. R.: Ethernet: Distributed Packet Switching for Local Computer Networks, Comm. ACM, Vol. 19, No. 7, pp. 395-403 (1976).
 - 41) Mital, R. M., Kim, M. M. and Berg, R. A.: A Case Study of Workstation Usage During the Early Stages of the Software Development Life Cycle, Proc. ACM Symp. on Practical Software Development Environments, pp. 70-76 (1986).
 - 42) 望月, 山内, 片山, 鈴木: ソフトウェアプロセス—実時間システムにおけるケース・スタディ, 情報処理学会ソフトウェア工学研究会, No. 71-18 (1990).
 - 43) 森 永輔: 分散開発環境の実現目指すローケース ツール, 日経コンピュータ, 1991.5.20号, pp. 93-102 (1991).
 - 44) Morris, J. H. et al.: Andrew: A Distributed Personal Computing Environment, Comm. ACM, Vol. 29, No. 3, pp. 184-201 (1986).
 - 45) 村井, 中村, 楠本, 加藤: 大規模分散環境 WIDE のアーキテクチャ, 情報処理学会マルチメディア通信と分散処理研究会, No. 51-4 (1991).
 - 46) 村上健一郎: TCP/IP によるコンピュータネットワークの構築技法, 情報処理, Vol. 31, No. 11, pp. 1586-1595 (1990).
 - 47) 村岡, 寛, 永田, 村井 (編): 知のキャンパス, bit, 1991年4月号別冊 (1991).

- 48) 長野, 山口: 交換ソフトウェア分散開発環境の実用化, NTT R & D, Vol. 39, No. 10, pp. 1375-1384 (1990).
- 49) 長野宏宣: 分散開発環境の基盤技術, 情報処理, Vol. 33, No. 1, pp. 14-21 (1992).
- 50) Notkin, D. et al.: Heterogeneous Computing Environments, Comm. ACM, Vol. 30, No. 2, pp. 132-140 (1987).
- 51) Notkin, D. et al.: Interconnecting Heterogeneous Computer Systems, Comm. ACM, Vol. 31, No. 3, pp. 258-273 (1988).
- 52) Notkin, D.: The Relationship between Software Development Environment and the Software Process, Proc. ACM Symp. on Practical Software Development Environments, pp. 107-109 (1988).
- 53) 貫井, 栗原, 三原: グループウェア支援機能の実験的考察, 情報処理学会ソフトウェア工学研究会, No. 71-8 (1990).
- 54) 大筆, 川越(編): CASE 環境特集号, 情報処理, Vol. 31, No. 8 (1990).
- 55) 大沢 光: テレワーキング革命, 日本実業出版社 (1988).
- 56) OSF: Guide to OSF/1: A Technical Synopsis, O'Reilly & Associates (1991).
- 57) OSF: 分散コンピューティング環境 DCE の全貌, Computer Today, 1991年7月号, pp. 17-55 (1991).
- 58) Osterweil, L.: Software Processes are Software Too, Proc. 9th ICSE, pp. 2-13 (1987).
- 59) Quarterman, J.S. and Hoskins, J.S.: Notable Computer Networks, Comm. ACM, Vol. 29, No. 10, pp. 932-971 (1986), [村井 純(訳): 世界のネットワーク, bit, Vol. 19, No. 3-7 (1987)].
- 60) 斉藤信男(編): ワークステーションと分散処理, コンピュートロール, Vol. 33, コロナ社 (1991).
- 61) 坂本, 小澤口: 志木サテライトオフィス第2期実験を振り返って, 電子情報通信学会誌, Vol. 74, No. 3, pp. 240-244 (1991).
- 62) 坂下善彦: 分散開発環境の事例と今後の展望, 情報処理, Vol. 33, No. 1, pp. 32-39 (1992).
- 63) Scheifler, R. W. and Gettys, J.: The X Window System, Software-Practice and Experience, Vol. 20 (S2), pp. 5-34 (1990).
- 64) Soley, R. M. (ed.): Object Management Architecture Guide Revision 1.0, Object Management Group, Inc. (1990).
- 65) 鈴木, 森田: PBX 網と LAN を統合する IVD-LAN, 日経コミュニケーション, 1989.11.6号, pp. 90-97 (1989).
- 66) Sweet, R. E.: The Mesa Programming Environment, ACM SIGPLAN Notices, Vol. 20, No. 7, pp. 216-229 (1985).
- 67) Swinehart, D.C., Zellweger, P.T. and Haggmann, R.B.: The Structure of Cedar, ACM SIGPLAN Notices, Vol. 20, No. 7, pp. 230-244 (1985).
- 68) 高野, 水野, 坂下, 北島: 分散型開発システムで仕様書作成からプログラム生成までを支援する, 日経エレクトロニクス, 1987.7.13号, pp. 179-194 (1987).
- 69) 垂水浩幸: グループウェアのソフトウェア開発への応用, 情報処理, Vol. 33, No. 1, pp. 22-31 (1992).
- 70) Taylor, R.N. et al.: Foundations for Arcadia Environment Architecture, Proc. ACM Symp. on Practical Software Development Environments, pp. 1-13 (1988).
- 71) Thadhani, A.J.: Factors Affecting Programmer Productivity during Application Development, IBM Syst. J., Vol. 23, No. 1, pp. 19-35 (1984).
- 72) Thomas, I.: PCTE Interfaces: Supporting Tools in Software-Engineering Environments, IEEE Software, Vol. 6, No. 6, pp. 15-22 (1989).
- 73) UI-ATLAS Distributed Computing Architecture: A Technical Overview, UNIX International (1991).
- 74) U.S. Dep. of Defense National Computer Security Center: Trusted Computer System Evaluation Criteria, DOD 5200. 28-STD (1985).
- 75) Winograd, T. and Flores, F.: Understanding Computers and Cognitions, Addison-Wesley (1986) [平賀 譲(訳): コンピュータと認知を理解する, 産業図書 (1989)].
- 76) 山村, 中島, 後藤(編): 電子メールとグループ通信特集号, 情報処理, Vol. 28, No. 8, pp. 1014-1059 (1987).

(平成3年10月1日受付)



青山 幹雄 (正会員)

1980年岡山大学大学院工学研究科修士課程修了。同年富士通(株)入社。現在、同社複合交換機事業部にて大規模通信ソフトウェアの開発手法、開発支援環境などの開発と適用に従事。1986~88年米国イリノイ大学客員研究員。IEEE Software の Editor。IEEE COMPSAC などのプログラム委員。電子情報通信学会、ソフトウェア科学会、IEEE Computer Society, ACM 各会員。