

5.7 ■ 情報処理技術 — 過去十年そして今後の十年 —

コンピュータが消える日： インターネット時代のソフトウェア

青山 幹雄

新潟工科大学情報電子工学科

■消えゆくコンピュータとインターネット

■コンピュータの3世代進化

ソフトウェアという言葉は、情報処理学会が設立された1960年頃に生まれた⁵⁾。ENIACに続き、1949年に最初のプログラム内蔵式コンピュータEDSACが動いてから10年を経ていた。その後の情報処理技術の進化は、図-1に示す、3世代に分けられる。

第1世代は、大量データ処理の自動化のために汎用コンピュータが誕生し、技術の進歩と処理要求の増大に対応して高性能化の道を辿った。ユーザは政府や一部の企業などに限られており、それらの個別のニーズに対応するカスタムソフトウェアが中心であった。

第2世代は大衆化である。1981年に発売されたIBM PCが事実上の標準となった結果、複数のベンダがパソコンを安価に大量生産できるようになった(図-2)。その結果、企業や家庭で日常必要なワープロなどのソフトウェアがパッケージとしてハー

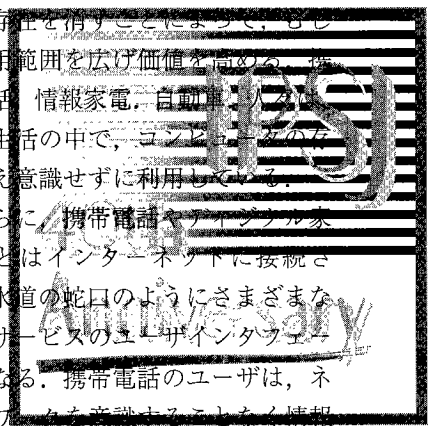
ドウェアとは独立して開発、流通されるようになり、ソフトウェアが独立したビジネスとして確立された。

第3世代は、インターネットとサービス化によるネットワーク遍在化の時代である。

■社会に浸透するコンピュータとインターネット

遍在化とは、携帯電話やデジタル家電など日常生活を取り巻く環境全体にコンピュータとソフトウェアが一層広く組み込まれ、浸透することを意味する。ユビキタス(Ubiquitous)あるいはパーベイシブ(Pervasive)と呼ばれる^{7), 9)}。水や空気のように、コンピュータはその存在を消すことなく、むしろ利用範囲を広げ価値を高める。携帯電話、情報家電、自動車、人々は日常生活の中で、コンピュータが在さず意識せずに利用している。

さらに、携帯電話やデジタル家電などはインターネットに接続され、水道の蛇口のようにさまざまな情報サービスのユーザインタフェースとなる。携帯電話のユーザは、ネットワークを意識することなく情報



特集 Special Features

汎用コンピュータとカスタムソフトウェア			パーソナルコンピュータとパッケージ			インターネットとサービス		
年代	1960	1970	1980		1990	2000		
利用形態	業務プログラム (メインフレーム)	部門・業務ソフトウェア (ミニコン/オフコン)	個人 (PC)	グループ (CSCW)	企業パッケージ (ERP)	インターネット 電子商取引 ASP (98) ディジタル家電・携帯・ウェアラブル		
アーキテクチャ	バッチ処理	オンライン処理/TSS	第5世代プロジェクト (82~)	クライアント/サーバ	インターネット エージェント			
開発プロセス	モデルと標準	ウォータフォール (70)	ソフトウェア工場 COCOMO (81)	プロトタイプ ソフトウェアプロセス GMM (87)	バイラル (88) インクリメンタル/イテラティブ/コン (87) ISO-9000	ISO-12207[SLCP] (95) ISO-10006 (97)		
	方法	構造化定理 (66) 構造化分析・設計 複合設計 (73) 構造化プログラミング (68) 構造化設計 (78) 情報隠蔽 (71) JSD (75)	オブジェクト指向プログラミング	オブジェクト指向分析/設計	UML 1.1[OMG標準] (97)	コンポーネント指向 アーキテクチャ指向設計		
	言語	FORTAN (57) BASIC (64) Pascal (71) COBOL (60) SIMULA (67) Prolog (72)	C (72) 構造化言語	Ada (80) Smalltalk-80 PostScript (82)	C++ (84) オブジェクト指向言語 Perl (87) SGML (86)	ビジュアル言語 Visual Basic (91) HTML (91) VRML (94) XML (98)	スクリプティング言語 Java (95)	
開発環境	コンパイラ	プログラミング環境	統合開発[Ada]	CASE (86)	プロセス支援環境 分散開発環境	Webベース開発環境		
開発素材	OS	OS/360 (65) Multics (65) MVS (74) UNIX (70)	MS-DOS (80)	4.2BSD (84)	Win3.0 (90) Web (93) DOS/V (90) Win95 Linux (94)	Win2000		
	DB	RDB (70) E-Rモデル (76)		SQL[標準] (87) ODB	マルチDB データマイニング			
	NW	ARPAネット (69)	TCP/IP (83)	WWW (89) Web (93)				
	ミドルウェア			分散オブジェクト環境	CORBA, COM] (91) Jini (98) JavaRMI[JDK1.1] (97)			
	UI	C(Character)UI	GUI[Star (81) Mac (85) X-Window]	N(Network) UI[Web (93)]				
	再利用部品	数値計算ライブラリ [FORTRAN]	クラスライブラリ[Smalltalk-80]	フレームワーク[GUI]	デザインパターン (94) JavaBeans (97) EJB (97)			
アプリケーション/パッケージ	航空機座席予約システムSABRE (64) CAD (64) JR座席予約システムMARS (65) VisiCalc (79) Lotus1-2-3 (88)	WordStar (78) MS-Word (83) 一太郎 (85) PageMaker (85)	dBase (80) MacPaint (84)	ERP[SAP/R3] (92) Lotus Notes (89) MS-Office	SCM/SCP CRM KM			
ハードウェア	IBM/360 (64) PDP/8 (65) Alto (73) Altair PC (75) I-8008 (72) Intel 4004 (71) Apple (76)	Word Processor (72) Cray-1 (76) SUN WS IBM-PC (81) Macintosh (84) I-8086 (78) I-286 (82) I-386 (85) 日本語ワープロJW-10 (78)	組み込み用マイクロプロセッサ	PDA[Newton] (93) Pentium (93) Pentium-II (95)				
ソフトウェア工学関係国際会議	NATOガルミッシュ会議 (68) 第1回ソフトウェア工学国際会議 [ソフトウェア工学の提唱]	第6回ソフトウェア工学国際会議[東京] (82)		第20回ソフトウェア工学国際会議[京都] (98) 第1回APSEC[東京] (94) 第6回[高松] (99)				

図-1 ソフトウェア技術の進化

サービスを利用している。ソフトウェアのサービス化である。ソフトウェアやコンテンツを、必要に応じて、いつでも、どこでも、誰でも利用できる。特に、誰でも情報サービスを利用できることが情報化社会の実現に問われる。

今、現実社会はネットワーク上に爆発的に膨張し、進化しつつある仮想社会と対峙し、相互に影響を及ぼしながら情報を中心とする知識社会⁶⁾へと共進化 (Co-Evolution) しているといえる。

一方、ソフトウェアの開発は、今なお開発者個人の経験や技量に頼り、かつ、人海戦術に頼る状況が続いている。西暦2000年問題によって明らかとなったように、現代社会全体がソフトウェアのもたらすリス

クに晒されている。

本稿では、ソフトウェアとその開発技術の進化を概観し、ネットワーク時代のソフトウェアのあり方、ならびにその研究開発のビジョンを示す。

■ソフトウェアの自己組織化と進化の諸相

生命は35億年かけて単細胞から高度に器官分化した人類へ進化した。ソフトウェアも、単細胞に等しい初期の小規模なプログラムから、多様なコンポーネントからなる高度な組織体へと自己組織化を遂げている。ソフトウェアは工学としてそれまでに類型のない知的成果物である

ことから、その進化は、図-1に示すように、技術革新の連続であった^{5), 8)}。さらに、増大し続ける要求に対して、ソフトウェアは図-3に示すように巨大化の道を辿ってきた。

一方、ソフトウェアの進化が新たなビジネスモデルや社会的変化を生み出した。それは、ソフトウェアへの新たなニーズを喚起し、ソフトウェアの進化を促した。ソフトウェアと実社会もまた共進化を遂げてきたといえる。

■ソフトウェア工学の進化

増大するソフトウェアへのニーズを満たすためにソフトウェア工学が誕生し、進化した。

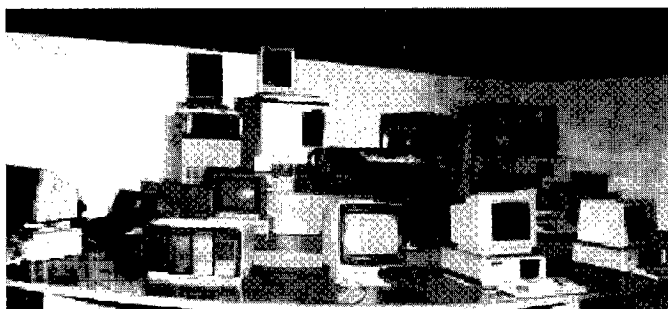


図-2 初期のパーソナルコンピュータ群
(BostonのComputer Museumにて著者撮影)

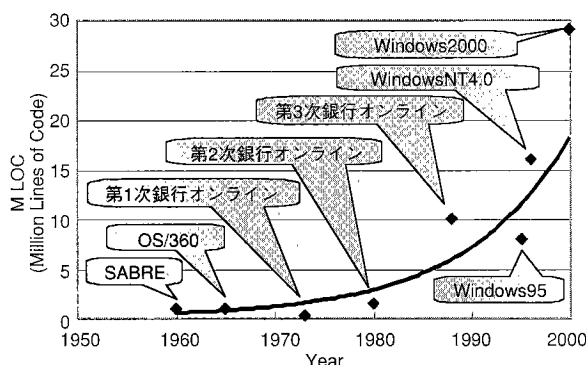


図-3 巨大化するソフトウェアシステム

■総合化とドメイン指向へ向かう現代ソフトウェア工学

初期のソフトウェア開発はプログラミングが中心であった。やがて、1960年代のソフトウェア規模の増大による「ソフトウェア危機」の高まりを契機にソフトウェア工学 (Software Engineering) が1968年に提唱された^{4), 5)}。以来、さまざまな技術やアイデアが提唱され、実践の場で試みられた。これらの技術の積み重ねによって、現代のソフトウェア工学は、総合工学と特定のアプリケーションドメインへ特化したドメイン指向ソフトウェア工学の2つの側面を持つようになった^{1), 2)}。

(1) 総合工学としての現代ソフトウェア工学

現代の大規模で複雑なソフトウェア開発では、多数の開発者が高度な要求を分析し、設計、実装、試験してシステムへ統合するために方法論

に則り、アーキテクチャやコンポーネントを開発、利用する。現代ソフトウェア工学はプログラミングだけではなく、これらの技術を統合した総合工学といえる。

(2) ドメイン指向ソフトウェア工学

ソフトウェアは、あらゆる産業や社会の分野へ適用される。この適用分野をアプリケーションドメイン、あるいは単にドメインと呼ぶ。ドメインは固有の要求や問題の構造を持つ。ソフトウェア開発も、ドメインの固有性を反映したドメイン指向ソフトウェアアーキテクチャやドメイン指向開発方法論が重視されるようになった。たとえば、組み込みソフトウェアはリアクティブソフトウェアの一種であるので、外部との相互作用が要求の本質となる。そのため、状態遷移モデルに基づく固有の開発方法論が開発され、広く実践されている。

■現代ソフトウェア工学の進化

現代ソフトウェア工学は、プロセス、プロダクト、ピープルの3つの視点で捉えられる。プロセスとは開発過程である。プロダクトは開発対象の構造やそのドキュメントなどの成果物である。ピープルとは、複数の開発者の人間的要因や協調して開発を進めるためのマネジメントや組織を意味する。過去10年間、ソフトウェア工学も大きな変化を遂げてきた。

■ソフトウェアプロセスの進化

プロセスの概念は、1970年にRoyceが提案したウォーターフォール型プロセスから始まった。1980年代のGUI技術の発展は、人間の感性や使い勝手の設計問題を提起した。そのため、プロトタイピングという新たな開発プロセスを生んだ。さらに、1987年のプロセスプログラミングの提案からプロセスそのものを設計する方法論が生まれた。これは、CMM (Capability Maturity Model) など、プロセスの組織的改善技術へ発展し、プロセスの概念が1990年代を通して現場に普及した。開発方法論の面では、オブジェクト指向が上流の分析・設計からライフサイクル全体に展開され、再利用技術とあいまって、ソフトウェア開発の基盤技術として成熟した。

一方、インターネットのもたらした時間をベースとする競争モデルは、ソフトウェアの提供とライフサイクルの概念も変えた。要求に応じて段階的に開発し、進化させるインクリメンタルプロセスや時間ベースとするプロセスモデルの開発、実践を促した。また、ソフトウェアが使い続けられるためには、ハードウェアやユーザ要求の変化に応じてソフトウェアが進化する必要がある。インクリメンタルな開発プロセスはソフトウェア進化に適した開発プロセスでもある。

■ソフトウェアプロダクトの進化

プロダクトの進化は、図-4に示すように、単細胞構造から機能分化したコンポーネントへの自己組織化といえる。1960年代半ばのOS/360に始まるOSの出現によりソフトウェアの分化が始まった。次に、1980年代のオープン化とダウンサイジングによって、アプリケーションが2層さらには3層のクライアント/サーバに分化した。1990年代には、3つの大きな変革があった。ソフトウェアアーキテクチャやコンポーネントなどの高水準な構造化と構成技術の出現、インターネットを指向したソフトウェアアーキテクチャの出現、ソフトウェアのサービス化とコンテンツの出現である。

オブジェクト指向は、カプセル化に基づく独立性の高いモジュール化の仕組みと上流工程からの分析・設計方法論によるモデル化技術を提供したが、大規模で複雑なソフトウェアを設計するためには、より高水準な構造化技術が必要となった。このため、ソフトウェア全体の構造や振舞いを設計するソフトウェアアーキテクチャ技術が提案された。さらに、ソフトウェアアーキテクチャを実行可能ソフトウェアとして実現したフレームワークの開発や局所的な設計のパターン化も提案された。これらの技術に基づいて、オブジェクト群をコンポーネントとしてパッケージ化し、組み合わせるコンポーネント技術が開発され、実践されつつある。

インターネットはソフトウェアアーキテクチャにも変革をもたらした。特定クライアントとサーバに閉じたクライアント/サーバアーキテクチャから不特定多数のクライアントとサーバに開放されたネットワークソフトウェアアーキテクチャへの進化である。そこでは、電子商取引などの多様な新しいソフトウェアと

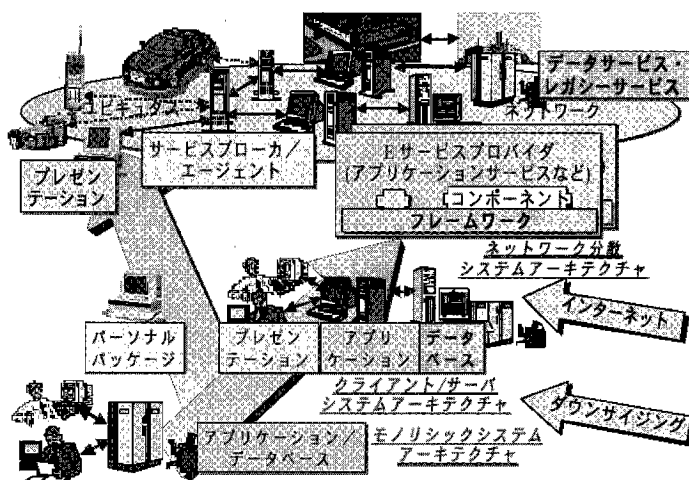


図-4 ソフトウェアアーキテクチャの進化

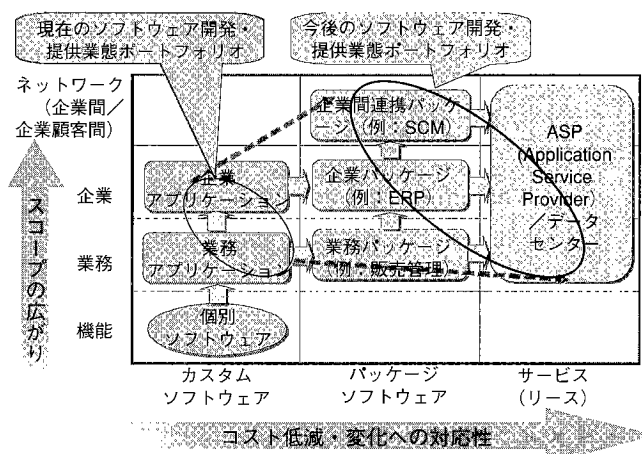


図-5 ソフトウェア開発・提供形態の進化

サービスが開発されている。

さらに、インターネット上で、ソフトウェアとコンテンツ、サービスが融合しつつある。たとえば、ソフトウェアをネットワーク上でリースするASP（Application Service Provider）などが可能となった。ユーザは必ずしもソフトウェアを所有する必要はない。ネットワーク上でソフトウェアが提供するサービスの価値をかうのである。

■ピープル：マネジメントと組織の進化

マネジメントの面では、もはや1人がすべて開発を担当するのではなく、技術の分化と深化に対応した専門家が必要となっている。ドメイ

ンの専門家であるドメインエンジニア、アーキテクチャの専門家であるアーキテクト、コンポーネント開発の専門家などである。また、企業全体の戦略的意思決定を行うCIO（Chief Information Officer）やCTO（Chief Technology Officer）の役割も重視されつつある。これは、企業の組織構造と人材育成の変革、さらには、産業構造の分化を促す。

一方、インターネットは、複数の組織による国際分散開発やオープンな組織によるオープンソース型開発などのグローバルな仮想開発を可能とした。

■インターネット時代のソフトウェアの挑戦

インターネットそのものが巨大なコンピュータであり、ソフトウェアであり、仮想企業や仮想社会を形成する。それは、新たなソフトウェア、サービスの開発の機会をもたらすとともに、課題も提起する。インターネット時代に対応したソフトウェア工学の新たな体系が求められている^{1)~3), 9)}。

■ネットワークと遍在化がもたらすソフトウェア工学への挑戦

(1) ネットワーク化がもたらす課題

ネットワーク化によりソフトウェアは点から面へ広がる。たとえば、電子商取引などの新しいソフトウェアの開発では、次のような要求がある。

- a) ネットワーク上で多様なソフトウェア、コンテンツ、サービスの統合
- b) 動的に進化するネットワークやビジネスモデルに対応した進化可能なソフトウェアの実現
- c) 24時間365日サービスを提供できる高信頼性、頑健性、安全性
- d) スケーラビリティの拡大と規模、複雑度増大への対応

このような要求を満たす新しいソフトウェアアーキテクチャの開発、開発プロセス、開発方法論が求められる。

さらに、インターネットは、ソフトウェアやサービスの開発、提供に国境を越えたグローバルな競争をもたらす。我が国のソフトウェア産業の国際競争力が試されるだろう。特に、我が国のソフトウェア産業は、図-5に示すように、従来のカスタムソフトウェア開発中心から、パッケージソフトウェア、サービスを含むソフトウェア開発の新たなビジネスモデルの確立が必要である。

(2) 遍在化する組み込みソフトウェア開発の課題

組み込みソフトウェアは、現在でもハードウェアの一部と考えられ、人海戦術的な開発に頼っている。しかし、携帯電話のソフトウェアの規模が1M LOCを超えているように、組み込みソフトウェアも大規模で複雑となり、「組み込みソフトウェア工学」と呼ぶべき技術の体系化が必要である。遍在化する組み込みソフトウェアは、一般ユーザがコンピュータを意識せずに使用できる必要がある。従来のコンピュータ中心の設計から人間中心・ユーザ中心の設計が求められる。

■超ソフトウェア工学 (Extreme Software Engineering) とソフトウェア工学の基礎

超高層ビルの建築などは極限工学 (Extreme Engineering) と呼ばれる。超高層建築がますます高層化するようにグローバルな金融ネットワークや航空管制などの巨大ソフトウェアもますます大規模で複雑になっている。1980~90年代の米国航空管制ソフトウェア開発の破綻など、超巨大ソフトウェア開発は依然としてリスクが高く、開発技術、マネジメント技術の限界が試されることになる。

一方、ソフトウェア工学の基礎技術は、形式化、メトリクスと計測、試験など依然として技術的課題が少なくない。ソフトウェア工学の基礎技術の確立が求められる。

■ソフトウェアの社会性

ソフトウェアは、これまで、「目に見えない」という、いわば、ソフトウェア異質論があった。しかし、西暦2000年問題で明らかとなったように、ソフトウェアは社会の基盤となっている。学会をはじめ、大学や産業界は人々がソフトウェアを理解し、活用できるように積極的に働きかける必要があるのではないかと。他産業と同様、ソフトウェアの啓蒙と研究開発の戦略的推進を図る必要がある。

■ Software is Not Enough: ミレニアムの扉を開く者

パソコンは、個人が親しく使えるコンピュータとして名づけられ、広く利用されるようになった。しかし、それは、多くの人に、時として苦痛をもたらした。ネットワーク化と遍在化はコンピュータそのものが社会環境になることを意味する。ここでは、ソフトウェアは人を楽しませ、幸せな生活をもたらす環境を実現するためのサービスとして働くよう求められている。

今、時代はパソコンからインターネットへの転換期にある。それは、ハードウェアからソフトウェアとサービスへの転換でもある。ソフトウェアの新たな研究、開発、ビジネスの機会に満ちている。また、同時にかってないグローバルな競争も引き起こすだろう。夜明け前の闇の中でたじろいでなどいられない。自らのビジョンを信じ、ひたすら走り続ける挑戦者こそ新たなミレニアムへの扉を最初に開く幸運を得るだろう。

最後に、本稿をまとめるにあたりご協力いただいた関係各位に感謝する。

参考文献

- 1) 青山幹雄 他: 連載「ソフトウェア新時代」, 情報処理, Vol.39, No.4~ Vol.40, No.3 (Apr. 1998~Mar. 1999).
- 2) 青山幹雄, 佐伯元司, 本位田真一, 深澤良彰 (編): ソフトウェアテクノロジーシリーズ, 全12巻, 共立出版 (1999~).
- 3) Brereton, P. et al.: The Future of Software, CACM, Vol.42, No.21, pp.78-84 (Dec. 1999).
- 4) Brooks, Jr. F. P.: The Mythical Mon-Myth, Addison-Wesley (1996) [1987年出版(訳)10月の神話, アジソン・ウェズレイ (1990)].
- 5) Campbell-Kelly, M. and Aspray, W.: Computer: A History of the Information Machine, Harper Collins (1996) [1996年出版(訳): コンピュータの歴史 200年 文芸春秋 (1999)].
- 6) Drucker, P. E.: Management Challenges for the 21st Century, Harper Business (1999) [1999年出版(訳): 明日を支配するもの, ダイアモンド社 (1999)].
- 7) Norman, D. A.: The Invisible Computer, MIT Press (1998).
- 8) Yau, S. S.: The 50 Years of Computing, IEEE Computer, Vol.29, No.10, pp.24-111 (Oct. 1996).
- 9) The Next 50 Years, CACM, Vol.40, No.2, pp.29-142 (Feb. 1997).

(平成12年4月10日受付)

