

解説



ネット指向パラダイムを求めて

1. 情報システム構築の新しいパラダイムを求めて†

—自律カプセル指向, ネット原理のすすめ—

翁 長 健 治†

1. ま え が き

1) ソフトウェアは、従来の財すなわち物財やエネルギー財と比較し、その性格においてまったく異質な新発明の財である。二値論理に従う知的判断と操作を通じて、ソフトウェアは長大な論理仕様を記述し、それを正確に実行する機械（コンピュータ）によって外部化し、知的価値を創成するという特異性をもっている。

2) VLSI 技術の激的革新により、コンピュータは普遍化し、ソフトウェアはわれわれの社会生活に多大な影響を及ぼすようになってきたが、その

*人間社会特性

*人間生理特性

*システム特性

*ソフトウェア製作技術

は十分に解明されておらず、在来工学のはざまにあって、多くの歪みが発生している。

3) 本文は、歪みの症状のいくつかをやや誇張しながらソフトウェアの悲劇として列挙し、次いでその発生原因を分析し、解決策の糸口を模索することを目的としている。自律カプセル指向とネット原理を糸口模索の重要なキーワードとして提示したい。すなわち

*自律カプセル化は、ソフトウェアの複雑性を制御し、成長と運用の柔弱性を確保するためのボトムアップ技術

と位置づけさらに

*ネット原理は、要求仕様の整合性分析、システム実現に向けての自律カプセル分割、自律

カプセル間の相互通信などの、トップダウン設計の数理

の役割を担わせることを提案したい。

これら二つのキーワードのおおのにおおに新規性はない。しかしソフトウェア研究へのネット理論の活用は限定され、有用性に対する一般的認識は低いのが実情である。したがって“ネット指向”ソフトウェアをすすめることの新規性は高いと私は考えている。

2. ソフトウェアの悲劇

1) ソフトウェアは過重な仕事量と、元来要求されてはならない要求や、期待されてはならない期待の重荷に苦しめられている。

社会の期待や産業界の要求に反し、若い学生は職業として、職責としてのソフトウェアに魅力を感じてはいない。小さなプログラム作りやゲームソフト作りは楽しくて興味はあるが、プロとして第3次オンライン・システムのような巨大なソフトウェア生産に参加するのは御免こうむりたいと考えている。職場としてのソフトウェア部門は、しばしば出荷製品の納期遅れの元凶とみなされ、一部の大企業を除けば苛酷な残業を強制されるという風聞が絶えない。

2) 新入社員の大半は心ならずも(?)ソフトウェア開発部門へ配属され、ソフトウェアとの長い付き合いを始める。中堅企業に就職した Dr. の卒業生が来訪してきたとき、「大学時代にはよく考えた後プログラム作りをしたものですが、今では手が先に動いて考えることは後手。時には考えることをやめてしまうことが多々です。新しい仕事・間断なき変更・納期逼迫など、忙し過ぎて頭が空っぽになっています」と不満をぶちまけたのは、たった2年前である。最近の電話によると残業なし、休暇取得奨励で休日が格段に増え余暇

† Toward a New Paradigm for Information System Design by
Kenji ONAGA (Dept. of Electronics and Computer Science,
Univ. of the Ryukyus).

† 琉球大学工学部電子情報工学科

の活用に苦心しているとのことである。

日本のソフトウェア業界は極端な不況に陥っているようである¹⁾。1991年3月期に過去最高益を出しながら一年後には赤字に転落し、人余りを言い出す企業が続出している。労働省の雇用調整助成金支給申請の6割はソフトウェア会社だという。これに対し米国のソフトウェア産業は高度成長を維持し1992年は前年比11%増、1200億ドルの規模、その反面ハードウェア出荷は1991年は前年比13%減、400億ドル余の実績であった。日本のソフトウェア産業は、どうやら構造的欠陥を抱え、package soft, downsizing, networkingの技術をもつ有力な外国企業の市場進出にさらされ、企業淘汰の機運を迎えているようである¹⁾。

3) ソフトウェア工学では論文が書けない、学位がとれないという風説がある。情報処理学会の論文誌を1989, 1990, 1991年度の3年間にわたり調べ、分野別に分類したところ、表-1を得た。分類細目は論文誌の目次を拾い集め適当にグループ分けした。ソフトウェア作成技法に直接関与するもの

- ソフトウェア工学
- 仕様技法
- プログラミング言語：8%
- データ工学
- 開発支援環境

の比重は8%、少し範囲を広げて

- 人工知能
- 知識システム：17.7%
- 自然言語処理

を加えると計25%に達する。風説が裏付けられたとみるべきか、分類の仕方が悪くて正当なカウントが行われていないので、判定不可とみるべきか、意見の分かれるところである。

1990年5月に情報処理学会創立30周年記念の論文賞が発表された。応募論文数114のうち、ソ

表-1 情報処理学会論文誌発表論文の分布 (1989, 1990, 1991)

分野	分類細目	細目別論文数	分野別中計(%)
基礎	数値解析 情報数学 オートマトン 暗号 アルゴリズム	73 (14.5%)	90 (17.9%)
	ヒューマンインタフェース ヒューマン・ファクタ	17 (3.4%)	
ソフトウェア	ソフト ソフトウェア工学 仕様技法 プログラミング データ工学 開発環境	40 (8.0%)	205 (40.8%)
	人工知能 知能システム 自然語処理 学習 CAI	89 (17.7%)	
	言語処理系 OS 並列処理	60 (12.0%)	
	データベース 情報検索	16 (3.2%)	
ハードウェア	ハードウェア アーキテクチャ ニューラル	51 (10.1%)	64 (12.7%)
	VLSI 回路解析 CAD	13 (2.6%)	
応用	画像処理 グラフィクス	72 (14.3%)	144 (28.6%)
	性能評価 ネットワーク 情報システム テキスト処理 パターン認識	72 (14.3%)	
1989, 1990, 1991年累計			503 (100%)

フトウェア、ソフトウェア工学、システム関係が41編あった(表-2参照)。最終的に選考された受賞論文11編中ソフトウェア、システム関係はたった3編であった。選考委員長の野口正一東北大教授は、多数の会員の関心のあり方が反映されていないのではないかと、日頃の一般論文の評価基準にも考えるべき問題を含んでいると指摘している²⁾。

4) 産業システムのソフトウェアは物価のインフレ同様に大規模なものに成長する傾向がある。あるレベルの情報化を達成すると、欲が出てきて、さらに新しい要求が自己増殖的に噴出してくるからである。情報システムにおけるソフトウェ

表-2 情報処理学会 30 周年記念論文選定経過要約

分野	応募論文数	1次候補	2次候補	受賞論文数
情報科学一般	1	—	—	—
基礎理論, 基礎技術	14	9	7	3
人工知能, 認知科学	24	14	6	2
データ処理	8	4	1	1
ソフトウェア	18	8	4	1
ソフトウェア工学	16	2	1	1
ハードウェア	12	3	2	2
ネットワーク	6	1	—	—
システム	7	1	1	1
信頼性と安全性	1	—	—	—
応用	6	1	—	—
その他	1	—	—	—
計	114	43	22	11

ア・コストはハード・コストを凌駕し、その差はさらに開きつつあると言われている。第3次オンラインは10メガライン、開発期間5年、工数4.5万人月の規模をもち、瀬戸大橋（9年、5万人月）、東京ドーム（3年、2万人月）の巨大土木工事に匹敵する巨大システムである³⁾。ソフトウェア工学は、産業界の苦悩に目を向け、その要求に答え得る技術レベルに達しているのであろうか？ 工学とは学習可能で、再現性のある平均品質を保証する技術体系のことである。情報システム以外のほか産業技術は大方工学レベルに達していると考えられる。

5) Carnegie-Mellon 大学ソフトウェア工学研究所の W. S. Humphrey 教授は、ソフトウェア生産組織の進化を5段階

レベル 1: 費用, スケジュール, 品質がまったく予想できない混沌とした状態

レベル 2: 費用と品質の変動が大, 場当たりので, 直観にたよる管理状態

レベル 3: 費用とスケジュールは信頼できるが品質保証が困難な定性的管理の状態

レベル 4: 品質が定量的に管理可能な状態

レベル 5: 品質改善の自動化と継続性が組み込まれた最適な管理状態

に分け、診断法と上位レベルに向上するために行うべき改善策を系統的に、分析的に提示している⁴⁾。同教授は80の日本ソフトウェア組織における200以上のプロジェクトを分析調査した結果、恵まれた大企業の一部を除き、そのほとんどが驚くべきことに最低位のレベル1にあることが判明したと言っている（参考文献4）pp. xii）。

3. 新しい区分: Humanware と Software とそして Hardware

1) 今日の計算機の使われ方は、発明の初期では思いもよらない方向に向かっている。計算を速くかつ人手を使わずに行わせたいとする願望から発明されたコンピュータは、今日ではむしろ

*プラントの自動制御

*多様な通信サービス

*企業活動の省力化・迅速化・統合化

などの実現に役立つことが要求されている。コンピュータが単なる計算に有用なばかりでなく

「生活の便利性を高める道具」

すなわち知的機械であることが判明した時点で、ソフトウェアは過重な期待と技術レベルと不釣り合いな重責を背負うことになった。

2) 太古の昔から人間は生活の安全と豊かさを求めて、生活の秩序作り、物事の手順、約束事の仕組みなど

「生活の技術 Humanware」

を発展させ、それらを現実のものとする道具、機械、組織、軍隊などの

「実現手段 Hardware」

を作り上げてきた。これらの Hardware は自然素材を素直に加工・組立・結合したもので

「ハードウェアのアナログ世界」

と呼ばれるべきものであろう。コンピュータの出現は異変をもたらした、すなわちハードウェアに

「デジタル新世界」

が侵入してきたのである。この侵入は現在も進行中で、電話・テレビ・制御装置などは次々とデジタル化されつつある。自然界にはデジタル的に動作するものは存在しない。すべてはアナログ量すなわち連続量であり、デジタルはこれを理想化したり、近似するときのみ創成される。

3) デジタル的な物の見方は思想・哲学の世界では2元論としてギリシャの昔から存在している。キリスト教・イスラム教などの一神宗教はその最たるものでわれわれの生活規範に

「正」か「邪」か

「ある」か「ない」か

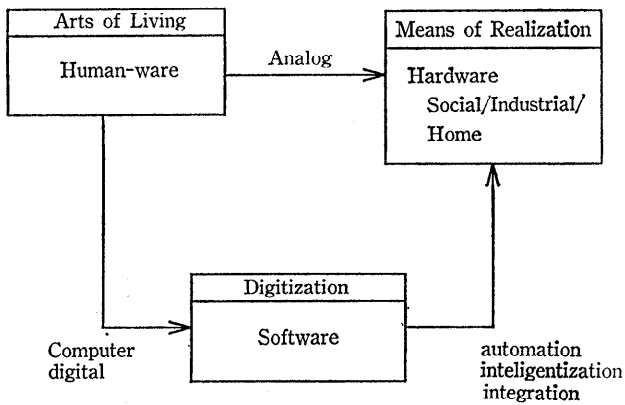


図-1 Software の位置付け

「する」か「しない」かの二者択一を日常的に迫ってくる。

図-1 は Humanware, Software, Hardware, Analog, Digital を図式的に位置付けたものである。Software は

「生活を豊かなものにする機械」としての Computer のために

「Analog 的な Humanware」部分を Digital に翻訳する役割を、知らず知らずのうちに背負ってしまったのである。

Computer 応用の初期、単純で、小規模で、ナイーブな応用に対しては当然とされてきたこの役割も、企業の OA や統合情報システム、社会的インフラストラクチャとしての交通網やバンキングシステムへと発展し、われわれの日常生活に直接影響を及ぼすようになった今日

「ソフトウェアの役割」を根底から検討し返すことが必要になってきた。

- 4) 従来の software life-cycle 論によれば
 - * ユーザ要求分析
 - * 要求仕様作成
 - * 機能設計
 - * 詳細設計
 - * 製造管理
 - * 検査・検証管理
 - * 運用保持管理
 - * ドキュメント管理

がソフトウェア工学の主要な任務とされている。注目しなければならないのは、上流設計部と呼ばれる、要求分析、要求仕様作成の部分がソフトウェア技術者の責務とする考え方である。上流設計

はまさに

「Humanware の Analog 的物物の
2 元論的列挙
2 元論的選択
2 元論的明確化」

を要求されているにほかならない。

5) このような要求仕様作りの行為は必ず

- * 日常生活慣習に変更
- * 価値判断基準の導入
- * 人間特性との整合性の変化

を引き起こす。この役割りを小人数グループのソフトウェア技術者が担うには、

あまりに重大すぎて、危険である。

要求分析はユーザ主導であるから、責任はあちら側にあると主張することもできる。しかし、災いはやがて我が身に降りかかってくる。ユーザは金を出すという強い立場を利用し、とめどのない、次第にエスカレートする要求仕様の変更をおしつけてくる。納期の遅れ、低い製品品質、性能の悪さなどをすべてソフトウェア開発者の側にあるとする無責任さを許容する最悪の事態におちいる。

4. 新しい責任分担を求めて

1) ソフトウェア工学の守備範囲が広すぎる。ほかの工学の分化の歴史をみれば、ソフトウェア工学にも分化の時期が到来したと考えるべきである。すなわち次の三分野

Humanware 工学：人間が古代から蓄積してきた Arts of Living を整理し、分類し、記述し相互作用を工学的に解明する。社会工学と共通項をもつが人間特性を強調する点で一線を画する。すなわち、従来庶務、人事、労務部門が所轄していた事物への工学的アプローチを強化して、一つの学問分野へ発展させることを課題とする；

情報システム工学：Humanware, Software, Hardware の特性、相互依存関係を明らかにし、要求分析技術、機能設計技術の確立を基本課題とする。性能評価、システム統合、評価尺度（メトリックス）、ヒューマン・インタフェース、通信ネットワーク、マルチメディア、オフィス・システム、データベースなどが研究トピックスである；

ソフトウェア生産工学：安定した責任ある要求仕様、機能設計を基に高品質なソフトウェアをコスト制御・スケジュール制御の下で生産する技術開発を基本課題とする。プログラム設計法、再利用技術、プログラム合成管理、工程管理、開発支援環境、検査検証技術、ドキュメンテーション管理などが主要な研究トピックスとなる；

に分化させ、その責任分担を明確にすべきである。ここで注意すべきことは、列記した研究トピックスは何も目新しいものではないが、Humanware を切り離して工学に格上げし、要求分析と機能設計を情報システム工学の責任とした。大規模ソフトウェアの制御され管理された生産だけをいわゆるソフトウェア工学の責任分野とし、実質を名で表すべく、ソフトウェア生産工学と改称したことにある。

2) 今やプログラミング言語やプログラム技巧はソフトウェア生産技術の主要関心事ではない。すなわちこれらを前提にした上での大規模ソフトウェアの生産に特化した Operations Research(OR) 的技術が主な関心事である。機械工学技術者は必ずしも旋盤や鋳物の熟練工ではないのと同様に、ソフトウェア生産技術者は必ずしもプログラミング技巧者である必要はない。無論、現在の技巧レベルを知らずして生産を語ることはできず、技巧レベルの向上への努力なくして生産性向上をはかることができないのは当然のことである。

3) Humanware 工学は、従来心理学、社会学、経済学の分野と思われてきた領域への工学アプローチの侵入である。たとえば、外国人新入社員に

- *企業の社風を説明し、その根拠を明らかにする；
- *仕事の手順をマニュアル化する；
- *報告・連絡・つき合いなどのルールを説明し、根拠を明らかにする；

などのような仕事の困難さが理解できれば、Humanware 工学の意義と研究テーマなどもおのずと明白になってくる。日本人が欧米で借家契約するとき、家具の状態、庭・植

木・家屋の状態、入居・退去、災害、家賃決定などの明確さに腰を抜かすほど驚く経験をするはずである。どうしてあのようなリスト・アップを考えつくのだろうか？ これは2元論的列挙・2元論的表現のもたらす結果の一例であろう。

契約に関する西洋・東洋の差異は Digital 表現と Analog 表現の差異にほかならない。Humanware 工学は情報システム化を陽に考慮する必要はないが、デジタル的思考とそれがもたらす得失への深い理解は必須不可欠である。またルールやプロトコルは設計されるべきもので、人間特性との整合性によって品質が尺度付けられる、という方法論が基本となる。

4) 情報システム工学においては、ソフトウェア実現技術を陽に考慮する必要はないかわり人間システムを直接的に分析することが求められる。物を対象にした従来のシステム工学の数理アプローチだけでは対応ができず、社会規範・法律、労務・人事、社会組織、経営・経済にまつわる人間特性を陽に摂取したアプローチが必要とされる。情報システム技術者は、ユーザとソフトウェア生産者の中間に立って、人間特性に適合した要求仕様と、安定的成長可能性をもつ機能設計を提供できる技術を確立することが責務となる。

別の表現を借りて言えば、ソフトウェア工学は、要求定義を Humanware 工学に、さらに上流設計を「人間特性に特化したシステム工学」に引き渡し、白らは目標が明確な「ソフトウェアに特化した生産工学」に専念すべきである、ということになる。

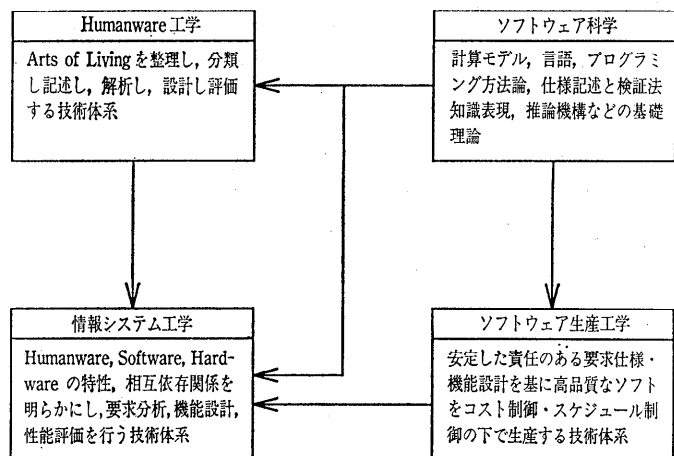


図-2 ソフトウェアをとりまく科学と諸工学の責任分担

プログラミング言語、仕様記述と検証、計算モデル、プログラミング方法論、知識表現と推論機構などは「ソフトウェア科学(原論)」の責務であろう。図-2 に相互関係を模図化してみた。

5. 新しい鍵：自律分散アプローチ

1) Main-frame Computer の存在はソフトウェア設計の発達を大きく歪めたのではなかろうか？ 巨大なソフトウェアが Main-frame に常駐し、「巨大な関係の網目」を監視し、データを交換し、蓄積し、タイミング情報を出力する。この動作の複雑さはコード数の指数関数的に増大するから 10M ラインの第 3 次オンラインシステムの複雑さはわれわれの想像をはるかに凌駕している。

元来このようなわれわれ人間の理解力を超える複雑さをもつソフトウェアを作ってはならないし、作れるはずもない。このことに気付いた人たちが、機能的にまとまった部分に分解する「Modular 設計」を提案した。しかし Module は依然として Main-frame に居住し、貪欲にデータを、関係の維持を要求し、かつわれわれはこれに忠実に答えねばならないもの、として規定されている。ソフトウェア Module のこの貪欲さは、アナログ Hardware にはまったくみられない。機械を連結し、組み合わせると高い機能を実現するには大きな制約があり、ほかの工学ではとっくの昔にあきらめている。機械的接触や化学的伝達には物理的、幾何学的、時間的に大きな制約があるのは、だれの目にも明白であるからである。

例として運送業をなりわいとする企業の情報システム化を考えてみよう。企業には職務に応じて部門があり、半ば自律的に働くが利潤を上げるという共通目標、を達成すべく協調して働くことは言をまたない。大きい企業では、部門は独自の生活慣習、勤働論理をもち、部門間の交流は、ある限られた幅のインタフェースを通じてしか行われない。これは人間特性の良し悪しを具現したものである。情報システム化の狙いは、これを非効率と断じ、インタフェースの幅を広げ、即時往来可能なものにしようとする点にある。

2) Main-frame 指向にもとづく企業の情報システム化のねらいは企業小宇宙(モデルという言葉もある)を Main-frame 内にソフトウェア的に実現しようと努める点である。この努力は、次の

三大要因

*企業人間システムという Analog 世界を Digital 的に仕様することに本質的な困難がつきまとう。したがって要求仕様は安定せず時とともに複雑、増大化する；

*Digital 的ソフトウェアの生産コストは仕様規模の指数関数で増大する。高位言語表現や自動化ツールを用いたとしても、指数的増加特性を克服することは原理的にできない；

*人間が管理できる複雑性には限界があり、Digital 的ソフトウェア品質はその限界に達すると急速に劣化する；

によって、将来性を閉ざされている。前記の困難を克服するには原理的転換を必要とする。すなわち

*省力、機能、データ透明性に関する要求仕様を局所にとどめ Digital 的ソフトウェアの規模上限に強い枠をはめる；

*Digital・ソフトウェアは必ず Hardware でカプセル化し、自律個体として認識し、個性を認める；

*自律個は、人間と同様、そのプライバシーが尊重され、守るべき義務と利益を享受する権利をもつ。なぜならばカプセル化した Hardware はある特定グループの人間に奉仕し、その人たちによって教育され運営維持されるから、情報システムの中で、その特定グループの人格を代表する；

*情報の共有、通信の透明性、協調行動などは自動的に行われるのではなく、人間の意志によって陽に統制される上級ネットワークを通じて行われる；

*応答が遅く、情報アクセスに制約があったとしても人間特性としてこれを容認する。しかしこれら遅延や制約をレベルの問題に帰着させるならば、テクノロジーによって応答特性は逐次改善可能である；

という設計思想に立脚しなければならないだろう。これを

『自律分散思考』

と称することにしよう。

6. 一歩改善：自律カプセル指向とネット原理

1) さて、本来あるべき姿をいくら論じてみても現実感にとぼしく、空論に終わる。あるべき姿

に向けて第一歩を提案しなければならない。情報システム構築の新しい原理として

i) 全体を自律個の協調ネットとしてとらえる。自律個はある定められた規範内で人格を保証され自主行動をとる。

ii) 自律個は局所的要求仕様を満足する Software を Hardware カプセル化したものであり、局所の人間によって維持運営され、学習させられ、成長するものである。

iii) Software をオブジェクトの協調ネットとしてとらえる。

iv) オブジェクトは、データとタスクをクラスでカプセル化したものであり、属性の継承関係によってファミリーを作り、メッセージで通信によってタスクを依頼し合うプログラム単位である。の4本柱を採用する。これを

『Object/Hardware カプセル指向に基づく協調ネット原理』と称することにしよう。

2) Carl Adam Petri は1962年に学位論文“Communication with automaton”によって今日ペトリネット理論と称される理論を創生した。1982年訪日の際、ペトリ博士との個人的対話において、「通信し合うオートマトン」の概念を学位公聴会において擁護するのに大変苦勞したと語っている。直積オートマトンを考えれば十分であるという批判に対し、オートマトンは自律個のモデルであり、自律個の社会は決して直積オートマトンではモデル化できない、と反論したそうである。ペトリネットは主として構造理論として発展している。

C. A. R. Hoare は1978年、論文“Communicating Sequential Processes”を発表し、並行動作するプロセスを記述し解析するための公理的代数を提案した。その後並列処理言語 Occam, そのハードウェア・チップ Transputer を設計・開発して世に送り出した。R. Milner は1980年、著書“A Calculus of Communicating Systems”を発表し、システムの形式的仕様記述言語の代数論を確立した。Petri と Hoare/Milner の理論は、それぞれ

『Object Net の幾何学的構造論と代数的言語論』を別個に発展させてきたと考えることができる。今後この二つの融合が進めばソフトウェア科学の統合的な数理となるであろう。

3) Smalltalk/IV に代表されるオブジェクト指向プログラミングは、人間的な目でみた世界を素直に、コンピュータの世界に移す技法を提供するものとして高い評価を得ている。D. Savic はその著書⁹⁾でオブジェクト思考の特徴を簡潔で分かりやすく要約をしているが、下記はさらに筆者が整理したものである。

i) 生産性を一桁向上させたとは評価されている構造化プログラミングはトップダウン手法であるのに反し、オブジェクト指向はボトムアップ手法である。

ii) トップダウン手法はまず問題が与えられていると考える。その問題を機能分解することに力点を置く。実行はできないが理論的に必要な部分に分解し、何段階にもわたる詳細化を経て、データ構造や処理手順は固まり、実行可能なコードができる。問題が変化すると、構造全体がくずれ、上層の小さな変更でもその影響が下層全体に及ぶ傾向がある。

iii) ボトムアップ手法では、問題分野が与えられればよい。まずデータ構造を認識し、それに対する基本操作を行う小さいプログラムを書く。次は、これらのプログラムを「組み合わせ」るだけで働くプログラムを考える。不足する場合にさらに新しいデータ構造を作り、追加していく。すなわち小さな「エキスパート」を多数作り、息を吹き込んで生命を与え、問題を解いてくれるように依頼するだけで問題解決できるシステムを目指す。

iv) これら小エキスパートは自律しているが軽く結合してネットを形成する。需要と供給を互いに通信し合い、大きな仕事を行うために協調する。

ここまで論旨を進めてくると、賢明な読者には「Net理論+オブジェクト指向プログラミング」が、ソフトウェア産業の今日の窮状を救う可能性と将来性をもっていることを理解していただけるだろう。

7. ネット指向ソフトウェア研究の現状

1) さて、本特集をネット指向パラダイムと銘打つ責任として、実質的に新しい物の見方、研究テーマ、研究技法を言及したり、提案したりすることができるのであろうか？ その成否は本特集

の各解説を読了した読者の判定に待たねばならないが、ここでは本特集の基礎となった二つのチュートリアル（1992年4月20日第5回軽井沢ワークショップ併設，1992年7月20日ソフトウェア工学研究会特集）と一つの特集⁶⁾のねらいと問題意識について述べるにとどめよう。

2) ネットを「キーワード付きダイアグラム」とみるならば、従来すでに多くの技法が開発され活用されている。プログラム理解のためにはフローチャート、PAD、データフローグラフなどがある。プログラム・テキストが一次元の列であるのに反し、ダイアグラムはキーワード付きの箱や円を枝で結合し2次元の広がりをもつことから、go-to文やif文、repeat文などの相互作用を簡潔に表現できる。さらにソフトウェア理解にはE-R図があり、機能の入出力・制御、データの相互依存などを2次元配置によって表現するため複雑なシステム構造を総括的にとらえることに役立っている。

しかし、ネットは従来ソフトウェアの視覚化ツールとしてソフトウェア工学の側に便宜的に使われてきたきらいがある。ネット理論の側にも、ソフトウェアを重要な応用分野とすべきだとする自覚に欠け、相互交流が不足していることは確かである。ネット理論は元来事象間の関係 (relation) に関する学問で、それ自身代数的、幾何学的な二つのアプローチを有しているが、情報システムをオブジェクト間の動的相互作用としてとらえるとき、統一的方法論の枠組みを提供することが期待される。しかし両アプローチ間の交流は残念ながら希薄である。

ネット理論

$$= \begin{cases} \text{幾何学的アプローチ} = \begin{cases} \text{グラフ理論} \\ \text{ベトリネット理論} \end{cases} \\ \text{代数的アプローチ} = \begin{cases} \text{時制理論} \\ \text{プロセス代数} \end{cases} \end{cases}$$

3) ネット指向とオブジェクト指向は、どこでどのように異なるのであろうか。不正確、誤解を恐れずにあえて列挙してみると、次のようになる。

*システムを『通信し合う自律個の結合体』としてみるという点で、両者は同一の世界観を有している；

*オブジェクト指向はプログラミング言語をもち、ボトムアップ的、構造的、実践的であ

るが、これといった理論をいまだ有していない；

*ネット指向はトップダウン的、分析的で、汎用的理論をもっている。しかしこれといった設計技法をいまだ確立していない。

*両者の良き理解者は少数である；

*ネット指向は分析、性能評価、検証などの上流設計にすぐれ、オブジェクト指向はデータ設計、コーディングなどにすぐれている；

情報システム構築は、トップダウンとボトムアップの反復修正プロセスであるから、ソフトウェアに対しては

(情報システム構築)

= (ネット指向デザイン)

+ (オブジェクト指向コーディング)

で立ち向かうべきであらうという物の見方にごく自然に導かれる⁷⁾。ネット指向について本特集の著者は必ずしも同一意見ではない。「使えるかどうか試してみたい」から「ぜひ使うべきである」まで広いスペクトルに散らばっている。現在コンセプトだけが先行し、実用に耐える技術の蓄積が不足していることは否めないが、将来性はあるものと確信している。

4) ソフトウェア構成にボトムアップ手法を取り入れるとシステム機能を逐次的に、進化論的に拡大成長させ得る可能性があることは理解できる。しかし実際に、どんな機能の小エキスパートたちを、どれだけ逐次的に準備していけば、おのおの水準の要求仕様を次々満足させるように調和設計できるのか、その方法論がただちに明らかになるわけではない。

ネットはオブジェクト世界の、通信と機能に立脚した離散モデルであるが、多層の単純化とグループ化を介在させる必要がある。ネット指向ソフトウェアにおいては、グループ化粒度とインタフェース通信の難易度との関連性を理論的に、実証的に解明することが重要な研究課題となるであらう。

8. 結 び

1) 計測自動制御学会の中に『離散事象システム研究会』がありネット理論とその応用に関する研究を、この数年続けている。最近の主要トピックスはネット指向設計であり、電子情報通信学会

英文論文誌 A の特集号にその成果を報告し⁹⁾, さらに次の特集「ネット指向ソフトウェア」を目下募集中である¹⁰⁾.

2) 文部省科学研究補助金・重点領域「自律分散システム」が 1992 年度に終了した。参加者百数名, 予算 5~6 億円, 3 年計画で生物, 動物, 人工システムにおける自律性, 分散性, 協調性, 自己成長性の研究を展開し, 統一理論に向けての努力がなされてきた^{9), 9)}. ソフトウェアに対しても自律分散指向は有効であると考ええる. 人間の意を体する自律カプセルが社会的トランザクションを集団として実行する協調ネットは, (WS+ユーザ) を通信網で統合したパソコン通信, VAN の一歩先の前方にあり, 飛躍しすぎた考えでは決してない. 他方, 電化製品や, 計測制御機器のインテリジェント化はカプセル化そのものである.

3) VLSI と Workstation の発展によって促進されている down-sizing は, われわれが提唱する新しいパラダイムの実現の好機である. Workstation は自律カプセルのよい候補であり, ネットワーク機能を通じて協調ネットを実現する道がひらけてくる. このため自律的に協調を実現する分散プロトコルの設計論が研究課題として浮上してくる. 人間特性が陽に考慮されねばならない点で, 従来の分散アルゴリズム論をはるかに超える, 高い社会的視点が必須となってくる.

謝辞 本特集の企画と実施は多くの人々の努力のたまものである. 特に電総研大蔭和仁氏, 東京工業大学佐伯元司氏, 東芝システムソフトウェア研究所本位真一氏, 日立システム開発研究所村田智洋氏の献身的努力なしには本特集は陽の目をみなかったであろう. 感謝の念にたえない.

参考文献

- 1) 日本経済新聞: 開国迫られるソフト業界 上, 中, 下, 1992年12月16, 17, 18日朝刊.

- 2) 情報処理, Vol. 31, No. 5, pp. 573-574 (May 1990).
- 3) 亀田靖治: 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 (May 1991).
- 4) Humphrey, W. S. 藤野監訳: ソフトウェアプロセスの成熟の改善, 日科技連 (1991).
- 5) Savic, D. 小林訳: オブジェクト指向応用プログラミング, プレンティスホール/トッパン (Nov. 1991).
- 6) 電子情報通信学会英文論文誌, IEICE Trans. Fundamentals, special section, Vol. E 75-A, No. 10, pp. 1315-1428 (Oct. 1992).
- 7) 6) の電子情報通信学会英文論文誌中の S. Honiden, and N. Uchihira: "Net-Oriented Analysis and Design" pp. 1317-1325 と Y. Nagao, H. Ohta, H. Urabe and S. Kumagai: "Petrinet Based Programming System for FMS" pp. 1326-1334, Oct. 1992. を見よ.
- 8) 特集: 自律分散システム, 計測と制御, Vol. 29, No. 10 (Oct. 1990).
- 9) 計測自動制御学会, 自律分散システムシンポジウム予稿集, 1991年1月; 1992年1月; 1993年1月.
- 10) 電子情報通信学会英論文誌, IEICE Trans. Fundamentals, 1993年10月予定, 応募締切 1993年3月.

(平成5年1月13日受付)



翁長 健治 (正会員)

1934 年生. 1957 年愛媛大学電気工学科卒, 1959 年大阪大学大学院通信工学専攻修士課程修了, 1965 年イリノイ州立大学電気工学科博士課程修了, Ph. D., 1970 年工学博士 (大阪大学). 1965 年アラバマ大学助教授, 1966~1979 年大阪大学助手, 助教授, 1979~1992 年広島大学工学部教授. 現在, 琉球大学工学部教授, 組合せ OR, ネットワーク・フロー, グラフ理論, ベトリネット理論, 自律分散アルゴリズムに興味をもつ. 著書「グラフ理論」(コロナ社), 「ベトリネットその応用」(計測自動制御学会) など. 電子情報通信学会, OR 学会, IEEE, ACM などの各会員.