

## 解説

木構造図用 CASE ツール間の  
データ交換言語: DXL†

長野 宏 宣<sup>1)</sup> 忠 海 均<sup>2)</sup> 岸 知 二<sup>3)</sup>  
 篠 木 裕 二<sup>4)</sup> 志 村 武<sup>5)</sup> 松 村 一 夫<sup>6)</sup>  
 藪 田 和 夫<sup>7)</sup>

## 1. はじめに

現在、ソフトウェアの仕様化や設計においては多様な図が用いられており、近年はそれらに基づいた多くの CASE ツールが開発されている。しかし、図の表現法や記法の十分な標準化がなされておらず、これが CASE ツールの普及の妨げにもなっている。そこで最近では、図を用いた CASE ツールを相互に接続し、CASE ツールで作成した図を交換することにより、多様な図やその記法を知らなくても、設計情報の再利用や流通ができるようにしようという動きが活発化してきた<sup>1)</sup>。CASE ツール間で図を交換するためのデータの交換形式としては、EIA の CDIF<sup>2)</sup>、IEEE の P1175<sup>3)</sup>などが提案されている。これらは、CASE ツールで取り扱う図のデータをプログラミング言語のような言語形式で記述し、これを CASE ツール間で共通な標準データ交換形式とするものであるが、この標準データ交換形式は、現在、ISO/IEC JTC 1/SC 7/WG 11 などの場で標準化が進められている。

本稿では、これらの標準化動向の一貫として木構造図を取り巻く状況、CASE ツール間でデータを交換するための標準形式の必要性、DXL仕様の

概要を解説する。また、国内の木構造図向けのツールを開発している主要な組織に参加を要請して行ったデータ交換実験結果についても報告する。

## 2. 木構造図を取り巻く標準化動向

国内ではプログラム論理表現様式として普及したさまざまな木構造図が、広くソフトウェア開発に使われている。しかし、プログラム構成要素は ISO 8631<sup>4)</sup> (JIS X 0128<sup>5)</sup>) で規格化されているものの、木構造図自体の標準化はなされていない<sup>6)</sup>。そこで、1988 年～1991 年に行われた日本規格協会の INSTAC における標準化調査研究の一環として、これらの木構造図の標準化が課題として取り上げられた。この調査研究により、すでにこれらの木構造図の作成や編集を支援する CASE ツール (以下、木構造図ツール) が普及し、さらに作成されたデータが大量に蓄積されているため、唯一の図とその記法を木構造図の規格とすることが非現実的であることが明らかになった。さらに、データの流通や再利用を促進するためには、木構造図用のデータ交換形式を定め、CASE ツール間でこれらのデータを交換できるようにすることが有効であり、今後はこのデータ交換形式を標準化すべきであることが明らかとなった<sup>7)</sup>。

これを受け、SC 7/WG 1\* 国内小委員会では、当時、米国で標準化が進められていた IEEE の P 1175 の活動に参加するとともに、P 1175 で開発されていた標準データ交換形式を木構造図に適用することを検討した。しかし、P 1175 の標準データ交換形式が木構造図への適用を考慮して

† Data Exchange Language for Tree Structured Charts by Hironobu NAGANO, Hitoshi TADAUMI, Tomoji KISHI (NEC Corp.), Yuji SHINOKI (Hitachi Ltd.), Takeshi SHIMURA (Oki Electronic Industry Co., Ltd.), Kazuo MATSUMURA (Toshiba Corp.) and Kazuo YABUTA (Fujiitsu Ltd.).

1) 日本電信電話(株)ソフトウェア研究所  
 2) 日本電信電話(株)ソフトウェア研究所  
 3) 日本電気(株) C&C ソフトウェア開発グループ  
 4) (株)日立製作所ソフトウェア開発本部  
 5) 沖電気工業(株)通信ネットワーク事業本部  
 6) (株)東芝 システム・ソフトウェア生産技術研究所  
 7) 富士通(株)システム本部

\* 1993 年 6 月の ISO/IEC JTC 1/SC 7 東京会議において、WG 1 は WG 11 (ソフトウェア技術用のデータ定義と表現) に統合され、現在は、WG 11 となっている。

おらず、これを木構造図に適用するときわめて冗長な構文になるため、木構造図ツール間でのデータ交換を目的に、P 1175 の標準データ交換形式との親和性にも配慮した DXL (Diagram eXchange Language for tree structured charts) を設計した。そして、DXL の仕様案を SC 7/WG 1 における国際標準化項目の一環として、1992 年の国際会議に文書提出し、各国と議論を開始している。

### 3. 木構造図用 CASE ツール間のデータ交換の必要性

現在国内では、HCP<sup>8)</sup>、PAD<sup>9)</sup>、SPD<sup>10)</sup>、TFF<sup>11)</sup>、YACII<sup>12)</sup> などの複数の木構造図が使われている。しかし、木構造図そのもの、及びそれらの作成を支援する木構造図ツールの標準化がなされていないため、複数社による大規模ソフトウェアの開発プロジェクトなどでは、各社が異なる木構造図を使用していることにより、以下の弊害が発生している。

- 発注側が納入するドキュメント記法の統一を要求した場合、ある一社の木構造図を開発に用いることになる。これにより、他の社の開発者はその木構造図を新たに学ばなければならず、また、自社の木構造図ツールも使えない。

- 発注側がドキュメント記法の統一を要求しなかった場合、各社ごとに別々の記法で記述することになる。これにより、開発途中での相互レビューや開発完了後の維持管理時に、開発者／維持管理者が複数の木構造図を学ぶことが必要になる。

- 最近の木構造図は木構造図ツールを用いて作成されることが多いが、現状では異なる記法の木構造図ルール間のデータ交換は困難であり、記法が違えば再利用／流通ができない。

しかし、特定の木構造図を標準としたり、新たな標準木構造図を制定することは、新たな技術者教育、木構造図ツールの再構築、木構造図ツールで作成した設計図の資産の再構築などの多大な費用を必要とする。そこで、現実解は、既存の木構造図を相互変換し、記法は異なっても木構造図の再利用／流通を可能とすることとなる。木構造図ツール間でデータを交換するにあたっては、ツールごとに独自のデータ交換形式を決めることは非効率であるから、標準的なデータ交換形式が必要となる。

### 4. DXL の概要

DXL の特徴は以下のとおりである。

1. ファイルや通信パスを介して容易に転送可能なように、テキストベースの言語となっている。
2. 各木構造図ツールで作成したデータからの変換を容易にするため、DXL の構文はツールによる解釈が容易となっており、また、人間が読み取り書いたりすることも可能である。
3. 木構造図との対応が容易なように、手続き的な記述をする構文となっている。
4. 将来の木構造図の改善や木構造図ツール特有の個別情報の付加などが可能なように、拡張性のある構文となっている。
5. 他の関連する国際的な標準 (IEEE P 1175) との整合性が取れている。

適用対象としている木構造図は、ISO 8631 (JIS X 0128) に準拠した図であり、DSD, HCP, PAD, PSD, R-chart, SPD, TFF, 及び YACII である。ISO 5807<sup>13)</sup> (JIS X 0121)<sup>14)</sup> のプログラム流れ図も、構造化された記述をしてあれば適用可能と考える。

表現できる記号は、ISO 8631 (JIS X 0128) のプログラム構成要素に対応する記号と、上記の木構造図が共通に用いている記号である。DXL の

表-1 DXL の主な構文要素

主な構文要素	意味
M_Packet	転送単位
Identification	モジュール識別情報
Module	木構造図の本体
Process	基本処理文
call	モジュール呼び出し文
break	抜け出し文
abstract	ブロック化文
if ... then else_if ... then ... else ... end_if	選択文
while ... loop ... end_while	前判定繰返し文
repeat loop ... until ... end_repeat	後判定繰返し文
continuous ... loop ... end_continuous	継続繰返し文



図-1 HCP チャートと対応する DXL の例

主な構文要素を表-1 に、HCP チャート記法に対応させて記述した DXL の記法例を図-1 に示す。表-1 と図-1 により、DXL では、一つの木構造図が “Module...End\_Module” の間に記述され、木構造図の記号と DXL の構文要素の対応が取れるように記述されることが分かる。しかし、DXL の構文要素は、各木構造図の記号と 1 対 1 には対応していない。これは、木構造図間で記号の意味する範囲が異なるためである。DXL の構文要素と各木構造図の記号との対応を付録に示す。

DXL が取り扱う情報は、木構造図が表現するモジュール内のアルゴリズムに関する情報であり、プログラムで扱うデータに関する設計情報、木構造図のグラフィック表現に関する情報、図形

の配置に関する情報、図を紙に出力するための情報、各種の開発管理情報などは取り扱っていない。

### 5. データ交換実験の概要

DXL の仕様の検証と DXL を用いたデータ交換の実用性を検証するため、記法の異なる木構造図を DXL を介して交換する実験を実施した。実験への参加企業は、国内 SC7/WG1 のメンバーで実際に木構造図ツールをもつ 6 社である。なお、現在、DXL の仕様は改善中であるが、実験で使用した DXL 構文は、参考文献7) で定めた仕様に従っている。

### 5.1 実験方法

実験データには、各木構造図ツールを用いて書かれたサンプル図を用いた。DXLの実用性の検証をするために、サンプル図は、実際に業務で作成された図から各木構造図のもつ記号をできるだけ網羅しているものを選択した。サンプル図は、50個程度の処理記号に対応する図を各社が5とおりに用意した。これらを各社で交換し合い（自社対自社を含む）、実験パターンは、360とおりとなった。

実験の手順を図-2に示す。各社はそれぞれ、自社の木構造図ツールで作成したデータをDXLに交換する正交換ツールとその逆交換ツールを用意した。実験では、これらの交換ツールを使用して各社で用意した木構造図データをDXLに変換してデータ交換を行い（A1図からA1-DXLへの交換に対応）、さらにそれを自社の木構造図ツール

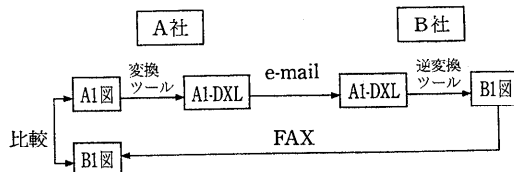


図-2 実験手順

へのデータに変換し（A1-DXLからB1図への交換に対応）、これを元の図（A1図に対応）と比較することにより、データ交換性を検証した。

実験結果の評価にあたっては、交換前の木構造図（A1図に対応）がもつ記号数をカウントし、DXLを介してどのくらい正確に他社木構造図に変換されたか（A1図とB1図との比較に対応）を交換率として算出した。交換率は、以下のように定義した。

$$\text{交換率} = \frac{(\text{元の図に含まれる記号の総数}) - (\text{他図へ変換したときに変換誤りが発生した記号数})}{(\text{元の図に含まれる記号の総数})}$$

ここで、「変換誤りが発生した記号数」は、交換後の図で変換誤りが発生した記号が、元の図のいくつの記号に対応するかから求めた。

### 5.2 実験結果

各図間の変換結果を、表-2に示す。交換率は全サンプルに対する交換率の平均値である。表では縦の列が原図の記法で、横の行が変換後の図を表す。たとえば、HCP(a)からPADへの交換では83%、PADからSPDへの交換では92%の記号が正しく変換された。

交換率には若干のばらつきはあり、完全な交換はできていないものの、利用法を特定すれば十分に実用になる高い交換率を達成したと考える。

その後、この実験結果を生かすために、サンプル図ごとの変換過程などの実験の過程を分析したところ、以下の課題が明らかになった。

表-2 全サンプルに対する記法間の変換率：原図→変換後の図 (%)

	HCP(a)	HCP(b)	PAD	SPD	TFF	YAC II	平均
HCP(a)	100	100	83	79	99	99	92
HCP(b)	88	95	74	81	85	94	86
PAD	94	87	100	92	87	100	92
SPD	100	96	98	100	95	100	98
TFF	97	100	100	97	100	100	99
YAC II	97	94	89	81	94	100	93

#### 1. 交換率のツール依存性

交換率が交換ツールの仕様に依存している。たとえば、

- 表-2からは、同じHCP記法でも、HCP(a)→HCP(b)変換とHCP(b)→HCP(a)変換で交換率が異なるが、これは2社の交換ツールの構文解釈の違いによる。

- 各変換において、数は少ないものの以下に示すように誤って変換されるものがある。

- 前判定情報、無条件分岐、ラベル、出口の消滅

- 前判定繰返しの後判定繰返しに、無条件分岐が基本処理文に、呼び出しが基本処理文に変換

- 余分な無処理文、余分な分岐条件の発生がある。変換誤りの原因には、原図と変換後の図

の記法間で記号の意味が1対1に対応つかないことと、変換ツールの構文解釈の違いの両方があるが、特定の記法と記号の組合せで発生するため、全体の交換率にはあまり影響を与えていない。

#### 2. 木構造図の変換上の課題

DXL要素と木構造図の記号要素が1対1に対応しないことにより、

変換誤りが発生したり、変換方法が一意に定まらないなどの問題が発生した。変換ツールで情報を保持するなどの対策で、ある程度は解決できるが、円滑なデータ交換のためには、各記号要素の意味の統一が重要であることが分かった。

### 3. DXL 仕様の改善点

意味不明瞭／曖昧な構文があった。たとえば、抜け出し文、選択文、継続繰返し文の構文である。これは、変換ツールの構文解釈の仕様の差が出た原因ともなっている。

## 6. データ交換の実用性

CASE ツール間のデータ交換の実用性は、その利用局面に依存する。以下に、おのおのの利用局面における実用性を示す。

1. 異なる木構造図を採用している複数社で共同開発する場合

(a) 木構造図のレビュー時 (DXL で木構造図を配布) には、完全に交換されなくては実用にならない。

(b) 木構造図の納入時 (DXL で納入) には、変換後の木構造図を手修正する度に依存する。変換率が低いほど実用性が低くなるが、現状では、何%までが許容範囲かは不明である。

2. 既存プログラムの流用／再利用時 (DXL に変換して流用)

(a) 木構造図をそのまま流用したい場合には、完全な交換がなされなくては実用にならない。

(b) 既存プログラムを修正することを前提に流用／再利用したい場合、既存プログラムを解析するときには、現状の変換率でも有効である。

3. 分散開発環境などでの木構造図ツール間の結合

同じ記法の木構造図ツールを使用しているも、会社や組織が異なっていると、ネットワーク接続条件などにより、バイナリデータのファイル転送などが困難な場合がある。このようなとき、同一仕様 (記法が同じ、記法の解釈が同じ) の木構造図ツール間であれば、現状の DXL の仕様で、十分、実用になる。

このほかに、DXL 自体を構造化プログラミングの教育用言語として採用するという利用法も想定される。

付録に示すように、木構造図の記法間で記号の

意味する範囲が異なる要素もある。これらの要素については、自動的に意味変換をすることはできず、100% の変換率は期待できない。しかし、DXL では、送られてきた DXL の文を変換ツールが解釈するための情報も記述できるような仕様にしてあるので、ある二つの記法間で 100% の変換率が必要な場合は、データ交換する両者のツールでローカルな規約を設けることにより、完全なデータ交換を達成する方法もある。

実用性からは、利用法を規定して交換できる情報の抽象度レベルを設定し、その範囲までの記号を使用している図の完全な交換を保証し、それ以外に 100% の交換を必要とする場合は、特定記法間でローカルな規約を設定することが現実的である。現在、SC7/WG11 国内小委員会では、DXL 仕様に、交換できる情報の抽象度に対する保証レベルを設け、ISO 8631 レベルでは完全交換が保たれるように仕様改善を実施中である。

## 7. 今後の予定

データ交換実験により、以下の課題が抽出された。

1. DXL で交換できる情報の抽象度に対して交換レベルを設定して、それを保証する必要がある。また、そのための変換ツール作成のためのガイドラインが必要である。

2. 変換ツールを作るにあたっては構文解析ルーチンを使用したがるが、DXL の文法に対する解釈の違いを解消し、交換ツールの作成を容易にするために、構文解析ルーチンを共有することが望ましい。

### 3. DXL の仕様の改善

意味不明瞭／曖昧な構文の明確化が必要である。

今後は、これらの課題を解決するために、SC7/WG11 を通じて、DXL の仕様の改善と標準化、DXL の実用化／普及のための各種施策の検討などを実施していく予定である。また、現在実用になっている木構造図では、プログラムで用いるデータ構造を記述するものも多く、データ構造を扱えるような拡張仕様の検討も実施していく予定である。さらに、DXL によって各記法の違いが明確になり、今後、各記法間で等価な交換ができるように、自然に記号の意味が統一に向かうこと

が期待される。

## 8. おわりに

DXL は、ISO/IEC JTC 1/SC 7/WG 11 などの場で標準化が進められている CASE ツール間のデータ交換形式の一つである。今回、この実験を通じ、実用的な CASE ツール間のデータ交換を実現するためには、以下が必要であることが分かった。

### 1. 図や記号の意味を統一すること

図で表現する情報の抽象度に対して交換レベルを設定して、記法間での記号の意味する範囲、及びその抽象度の違いを統一することが有効である。また、これを実現するために、変換ツール作成のためのガイドラインが必要である。

### 2. 変換ツールの仕様の保証

変換ツールを作るにあたって、データ交換言語の解釈を統一する必要がある、そのためには構文解析ルーチンの共用や検証システムなどが必要である。

### 3. CASE ツールの仕様の統一

その CASE ツールの仕様により、データ交換言語との完全な対応が取れないもの（たとえば、データ交換言語の構成要素とツールで扱う記号要素との対応がつかない場合など）については、CASE ツール自体の仕様変更が必要となる。

現在、ISO/IEC JTC 1/SC 7/WG 11 では、CDIF や P 1175 などの成果が提案され、国際規格化に向けての審議が開始されている。この活動の中で、データフロー図や状態遷移図などを対象としたデータ交換言語の検討/標準化がなされていく予定であり、CASE ツールの普及にとともに、さまざまな図の交換形式の標準化ニーズが高まってくると予想される。しかし、このような標準化にあたり、今回のような実験は、まだほとんどなされていないのが実情である。今回の実験から、机上で開発して机上での記述実験で検証した DXL の仕様上の問題点や実用上の問題点が明らかとなった。今後、データ交換言語の標準化にあたり、このような実験が不可欠であることを、SC 7/WG 11 での標準化活動に反映していく予定である。

**謝辞** 今回の実験は、標準化活動の一環として各社のボランティアで実施しました。実際に実験を実施していただいた(株)日立製作所 西尾

高典氏、(株)東芝 松尾 尚典氏、日本電気(株) 山下 智慎氏、NTT ソフトウェア(株)宮崎 裕司氏、及び各社の関係各位に感謝致します。また、直接/間接にご指導/ご協力していただいた慶応大学の山本 喜一助教授、日本 IBM の大場 充氏をはじめとする国内 SC 7/WG 11 の委員の方々のご厚意に感謝致します。

## 参考文献

- 1) 松本: CASE 環境統合のためのインタフェースの標準化現状, 情報処理, Vol. 31, No. 8, pp. 1086-1094 (Aug. 1990).
- 2) EIA: EIS/IS-81, CDIF—Framework for Modeling and Extensibility (July 1991).
- 3) IEEE: IEEE Std 1175, Reference Model for Computing System Tool Interconnections (Trial-Use Standard) (Aug. 1992).
- 4) ISO/IEC 8631 Information Technology—Program constructs and conventions for their representation (1989).
- 5) JIS X 0128—1988 プログラム設計要素及びその表記法.
- 6) 長野, 東: 小特集: ソフトウェア工学における標準化動向: プログラム構成要素及びその表記法, 情報処理, Vol. 28, No. 9, pp. 1166-1172 (Sep. 1987).
- 7) 平成3年度ソフトウェア開発・システムの文書化標準化調査研究報告書, (財)日本規格協会 情報技術標準化研究センター, pp. 11-49 (平成4年3月).
- 8) 長野, 浅見, 忠海: 階層化プログラム設計法—HCP チャート—, 電気通信協会 (1993).
- 9) 二村: プログラム技法—PAD による構造化プログラミング—, オーム社 (1984).
- 10) Azuma, M., Tabata, T., Oki, Y. and Kamiya, S.: SPD: A Humanized Documentation Technology, IEEE Trans. Software Eng., Vol. SE-11, No. 9, pp. 945-953 (1985).
- 11) 松村他: ソフトウェア設計記述法, 東芝レビュー, Vol. 41, No. 8, pp. 672-676 (1986).
- 12) Murakami, N.: On the Presentation and Automation of Design Specifications in YAC II, Information Processing Soc. Japan, IPSJ, Tokyo (1983).
- 13) ISO 5807 Information processing—Documentation symbols and conventions for data, program and system flowcharts, program network charts and system resources charts.
- 14) JIS X 0121—1986 情報処理用流れ図・プログラム網図・システム資源図号.

(平成5年9月9日受付)

付 録

































プログラム 構成要素 (IS)	プログラム 構成要素 (その他)	DXL 要素	HCP	PAD	SPD	TFF	YAC II
基本		基本 処理文	 手続き部	 手続き部	 手続き部	 手続き部	 手続き部
順次		基本 処理文 の並び	 手続き部  手続き部	 手続き部  手続き部	 手続き部  手続き部	 手続き部  手続き部	 手続き部  手続き部
前判定 繰返し	前判定 繰返し1		 手続き部  手続き部 (条件)	 手続き部	 (FOR:条件) 手続き部	 手続き部 条件 D	 終了条件 手続き部
	前判定 繰返し2		 手続き部	 手続き部	 (WHILE:条件) 手続き部	 手続き部	 終了条件 手続き部
後判定 繰返し	後判定 繰返し1			 手続き部	 (DO:条件) 手続き部		 反復条件 手続き部
	後判定 繰返し2						 終了条件 手続き部
継続 繰返し		継続 繰返し	 手続き部	なし	なし	 手続き部	なし

図-3 木構造図間の記号の意味の違いと DXL の意味要素との対応関係 (1)

プログラム構成要素 (JIS)	プログラム構成要素 (その他)	DXL 要素	HCP	PAD	SPD	TFF	YAC II
単岐選択		選択文 1					
双岐選択							
多岐選択	(1対1に対応しない)	選択文 2					
打切り		打切り文 (exit) return文 抜け出し文 (break)					なし
なし	goto文	goto文					ラベル名

XXX: 特定のテキストを記述

図-4 木構造図間の記号の意味の違いと DXL の意味要素との対応関係 (2)

プログラム構成要素 (JIS)	プログラム構成要素 (その他)	DXL 要素	HCP	PAD	SPD	TFF	YAC II
なし	開始	開始文					
なし	終了	終了文		なし	なし		
なし	無処理	無処理文		なし	なし	なし	なし
なし	モジュール呼出し	モジュール呼出し文			なし		
なし	ブロック化	ブロック文		なし			

図-5 木構造図間の記号の意味の違いと DXL の意味要素との対応関係 (3)





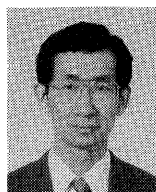
長野 宏宣 (正会員)

昭和 45 年九州工業大学電子工学科卒業。同年電々公社 (現, NTT) に勤務。一貫してソフトウェア生産技術の研究実用化に従事。現在, ソフトウェア研究所ソフトウェア開発技術研究部に所属。昭和 57 年本会論文賞受賞。



忠海 均 (正会員)

昭和 56 年大阪大学理学部物理学科卒業。昭和 58 年同大学院物理学科修士課程修了。同年電々公社 (現, NTT) に勤務。ソフトウェア生産技術及び開発支援, システムの研究実用化に従事。現在, ソフトウェア研究所ソフトウェア開発技術研究部にて, ソフトウェア開発方法論及び CASE ツールの研究実用化に従事。日本ソフトウェア科学会, IEEE CS 会員。



岸 知二 (正会員)

1956 年生。1979 年京都大学工学部情報工学科卒業。1981 年同大学院修士課程修了。同年日本電気(株)入社。1986 年より 1 年間カーネギーメロン大学でソフトウェア開発環境の研究に従事。現在マイコンソフトウェア開発研究所技術課長。ソフトウェア開発環境, 構成管理, 分析・設計支援に関心をもつ。IEEE Computer Society 会員。



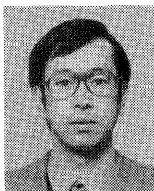
篠木 裕二 (正会員)

1952 年生。1974 年東海大学工学部電子工学科卒業。1979 年早稲田大学大学院修士課程 (物理学及び応用物理学専攻) 修了。同年(株)日立製作所入社。ソフトウェア開発本部主任技師。入社以来, 基本ソフトウェア向け開発技法及び開発支援環境の研究・開発・運用に従事し, 現在はソフトウェア工学に興味をもつとともに処理系ソフトウェアの開発に従事。SC 7/WG 11 委員, IEEE Computer Society 会員。



志村 武

1961 年生。1983 年工学院大学電子工学科情報工学コース卒業。1985 年同大学院修士課程修了。同年, 沖電気工業(株)入社。現在, 通信ソフトウェア開発のための支援システムの開発に従事。



松村 一夫 (正会員)

昭和 48 年東北大学通信工学科卒業。昭和 50 年同大学院工学研究科修士課程修了。同年(株)東芝総合研究所に入所。以来, 言語コンパイラ, ソフトウェア開発支援システムの研究開発に従事。現在, 同社研究開発センター, システム・ソフトウェア生産技術研究所に所属。電子情報通信学会, IEEE Computer Society 各会員。本会情報規格調査会 SC 7/WG 11 委員, 論文誌編集委員, ソフトウェア工学研究会幹事。



藪田 和夫 (正会員)

1976 年上智大学理工学部電気電子工学修士課程修了。同年(株)富士通入社。以来ソフトウェア生産技術の研究開発に従事。1993 年末よりシステム事業推進本部システム技術統括部 SDAS システム部に所属。