

# 4. 分散オブジェクト環境における部品組立型ソフトウェア開発の実証実験

佐藤和夫 日本IBM (株)  
 青山幹雄 新潟工科大学  
 長谷川邦夫 日本ユニシス (株)  
 高橋典幸 (株) 日立製作所  
 安竹由起夫 日本デジタルイタイクップメント (株)

## ■新職種企業の誕生

コンポーネントウェア<sup>5)</sup>の進展によってソフトウェア産業は、部品を提供するコンポーネントベンダと、それを組み立てて、アプリケーションを開発するコンポーネントインテグレータに専門化すべきであると考えている。

従来型ソフトウェア開発では、発注社と発注社から仕事を受注し、すべてを自社でまかなうSI(システムインテグレータ)社が存在していた。今後は新たに、部品を販売する部品ベンダが多数登場するであろう。ここで部品ベンダとは、部品カタログの提供と部品販売の他に、顧客の求めに応じて部品稼働環境の提供を行い、いわば部品のレンタル屋としての役割も担うと定義する。多数の部品ベンダの提供する部品は、多様でさまざまな品質がありうる。このような部品の中から、要求仕様や価格、品質などの条件を満たす、最適な部品を探索することは困難である。そこで部品ベンダの出現とともに、部品を評価して最適な部品をSI社にお勧めするコンポーネントブローカが登場してくる。大部分のSI社は、コンポーネントインテグレータ(以下インテグレータと呼

ぶ)へ転換し、部品組立に専念する。しかし、専門知識、コスト構造の違いにより、コンポーネントインテグレータは専門化し、開発ごとに協力しあう水平分業が進むであろう。水平分業は国内に限定されず、地球規模で最適なパートナーを探しあうことになる。この想定のもと、インターネットを利用した複数のソフトウェアインテグレータ間の水平分業は実用的であるかどうか、実用的であるとすれば今後解決すべき問題点は何かを明確にするために、実証実験を行った<sup>7)</sup>。

## 開発プロセスの変化

開発プロセスはどのように変化するのであろうか。従来、多くの場合は、要求定義→仕様→設計→コーディング→テストの局面を踏む。実証実験では、図-1に示すように、要求定義→部品検索→部品組立→システムインテグレーションの形になると想定した。

図-1はさらに、想定する各職種企業が、開発プロセスの中で担う役割を検討した結果も示す。図でコンポーネントブローカは、便宜上SI社と同じ行に入れてある。

## グループの体制、アプローチの概要

図-1はさらに仮想企業内での技術要素とその実証を示す。仮想企業としてはSI社、コンポーネントブローカ、インテグレータA、B、C、部品ベンダの計6社である。括弧内は、実験環境構築/実証実験を担当した参加メンバを示している。上記6社に対し9社1大学が参加し、仮想企業を構成した。

実証環境構築には、分業体制を担う各仮想企業間に各種規約(セキュリティ構築・運用手順書、CORBA環境インタフェース実装規約、CORBA環境実装手順書)を設定することと、それに対応したツール類などの開発が必要であった。グループ参加メンバがその得意技術を持ちより技術開発に行い、

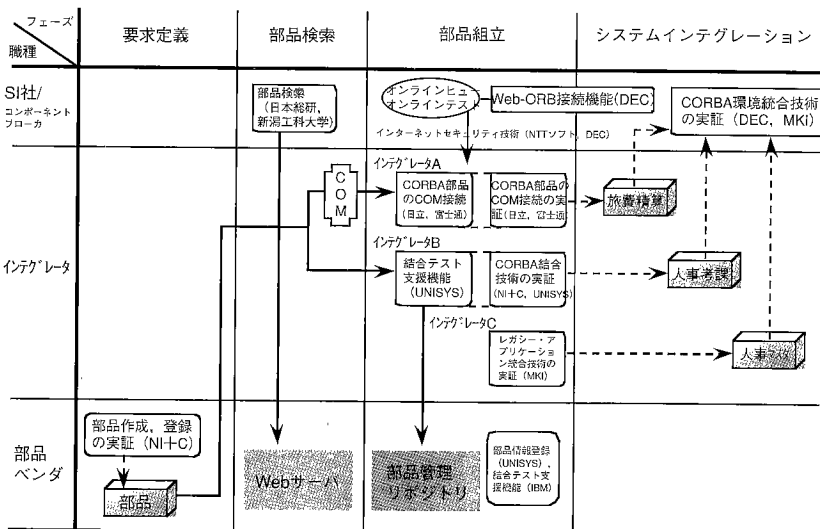


図-1 想定する開発プロセスと各職種企業の対応

他社との接続の実現性について確認した。これらの準備の後に実証実験を開始した。

実証実験では、人事情報システムを共通のプロトタイプアプリケーションとした。人事マスタがレガシーシステムとしてすでに存在する場合を想定し、それに対し、38.1Kステップの旅費精算と人事考課機能を付け加え、新規システムとして開発する作業を実験対象とした。開発の際に、部品組立型開発プロセスを用い、CORBA部品による組立てを、各仮想企業が分業しながら、インターネットを通じて分散環境で作業する実験を行った。

以下、新規開発規約／技術を中心に、実証実験の概要も含め、ソフトウェアCALSならではの考えられることに重点をおきながら紹介する。

### ■各実証実験のアプローチと成果

実証実験は4つのグループに別れて実施した。以下(I)～(IV)のグループごとに、開発した規約および技術を説明するが、それぞれの位置づけを、図-1を用いて説明する。

インテグレータが部品組立環境の企業間共用を行いながら部品組立型ソフトウェア開発を行う実証実験をグループ (I) (II) が実施した。

(I) は異ORB接続技術を用いた場合である。図-1で説明すると、フェーズでは部品組立、職種ではインテグレータと部品ベンダが、結合テスト支援機能と部品管理リポジトリを使用した部分である。(II) はCORBA部品のCOM接続技術を用いた場合である。図-1では、(I) と同じフェーズ、職種で、CORBA部品のCOM接続を使用した部分である。

部品とレガシーシステムとの接続の実現と部品の移植性の確認／統合を (III) のグループが実施した。図-1のフェーズでは部品組立およびシステムインテグレーション、職種ではSI社が、Web-ORB接続機能とCORBA環境統合技術を使用した部分である。

開発に必要な部品を評価するコンポーネントブローカを (IV) のグループが実施した。図-1のフェーズでは部品検索、職種ではコンポーネントブローカと部品ベンダが、部品検索を使用した部分である。

(I), (III) ～ (IV) では新規規約を制定した。また (I) ～ (IV) では、新規技術を開発した。そしてそれらを既存技術と組み合わせて実証実験を実施した。

(I) コンポーネント組立環境の企業間共用の実現  
異ORB接続技術を用いた場合  
OMG (Object Management Group) の制定するCORBA (Common Object Request Broker Architecture) 準拠の互いに異なる2つのORB (Object Request Broker) 環境の一方が部品提供側であり、もう一方が部品利用側という接続形態における部品組立型開発を実証テーマとした。

CORBAは分散システム構築のための基盤環境の

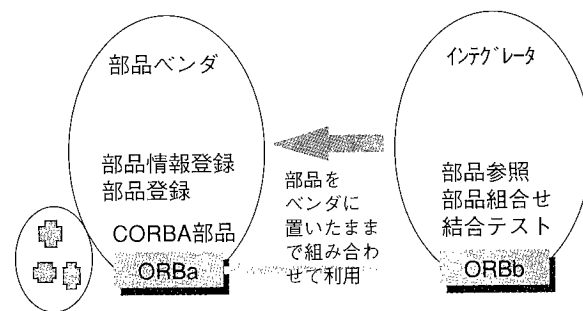


図-2 異種ORB接続による部品結合

1つとなりつつある。CORBAによりクライアント／サーバ型アプリケーションを、分散オブジェクト環境上で構築することが可能となり、分散したアプリケーションやデータの統合が容易になる。特に、CORBA2.0で規定された相互運用性に準拠すれば、異なるORB環境が連携し、利用者にとっては位置透過な1つのORB環境として利用できるため、プラットフォームに依存しない分散システムの実現が可能になる。

ここでは、図-2に示すように、部品ベンダおよびインテグレータは、それぞれ任意のORB環境を利用できる。部品を提供する環境と、部品を利用してアプリケーションを組み立てる環境が、それぞれCORBAによる分散オブジェクト環境である。部品はCORBAオブジェクトである。CORBA2.0のIIOP (Internet Inter-ORB Protocol) <sup>4)</sup> を利用して、利用する側 (インテグレータ) は部品を提供側 (部品ベンダ) の環境においたままで利用 (実行) できる。これを部品の共用と呼ぶ。部品ベンダは部品の情報を自社の部品管理リポジトリを通して公開する。あるいはより簡便かつ広範な利用を可能にする方法としてWebに公開することもできよう。なおここでは、部品ベンダとインテグレータの間の部品利用契約を前提として上記の方法をとったが、(IV) 「エージェントによる部品情報の収集」の節ではWebによる部品情報開示の方法を紹介する。

こうしてインテグレータは、複数の部品ベンダの部品情報を参照して必要な部品を探索し、その稼働を部品ベンダの環境下で確認してアプリケーションに組み込むことになる。

(I) の実証実験のポイントは、異なるORBの接続下において以下の機能確認を通してCORBA接続の技術を確認し、部品の共用を実現することである。

- 部品ベンダが部品管理リポジトリに部品情報を登録する。
- インテグレータから部品ベンダの部品情報を検索する。
- インテグレータが部品を組み合わせたアプリケーションを想定し、部品ベンダ環境上の部品を実行する。部品情報として以下のものを想定した。
- 部品基本情報

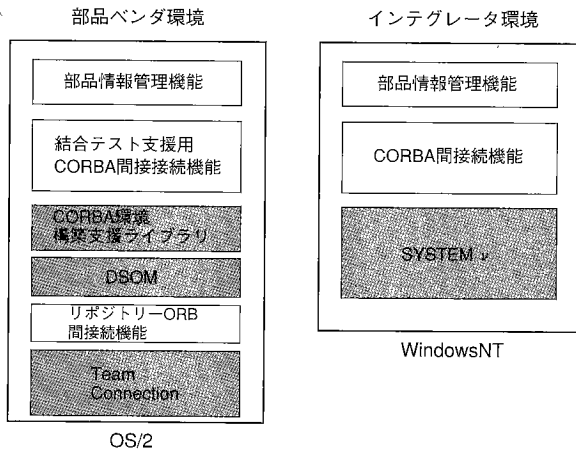


図-3 異種ORB接続のための実証環境

- 部品テストケース情報
- 部品変更情報

この実証実験の前提として異種ORBの接続に関し以下の規約を定めた。

- 結合テスト支援用CORBA間接続規約

異種のORBを持つ部品ベンダおよびインテグレータのそれぞれの環境で部品情報管理機能を利用し、ORBを通してリポジトリへアクセスできるようにするために、クライアントとORBのインタフェースを定めた。

- リポジトリORB間接続規約—市販の各種リポジトリを利用する想定で、ORBとリポジトリ間のインタフェースを定めた。

また、実際の実証作業のために以下のソフトウェアを開発した(図-3参照)。

- 部品情報管理機能
- 結合テスト支援用CORBA間接続機能
- リポジトリORB間接続機能

実証作業では以下の既存ソフトウェアを利用した。

- OS: WindowsNTおよびOS/2 Warp
- ORB: SYSTEM v [nju:] 3.0 (日本ユニシス) およびDSOM3.0 (IBM)
- リポジトリ: TeamConnection (IBM)

異種ORBの接続は実証実験以前にいくつかのORB間で確認されている。しかし実証実験によってCORBAオブジェクトによるソフトウェア部品化の実現可能性と、一般の業務アプリケーションにおけるCORBA連携の実用の可能性を示すことができたと考えられる。

## (II) コンポーネント組立環境の企業間共有の実現

### CORBA 部品のCOM接続技術を用いた場合

部品としてはCORBAオブジェクト以外にも以下のものが挙げられる。

- COM/DCOM/ActiveX
- JavaBeans

これらの技術は、併存・競合しながら、コンポーネントウェアを構成する技術として利用されると考えられる。

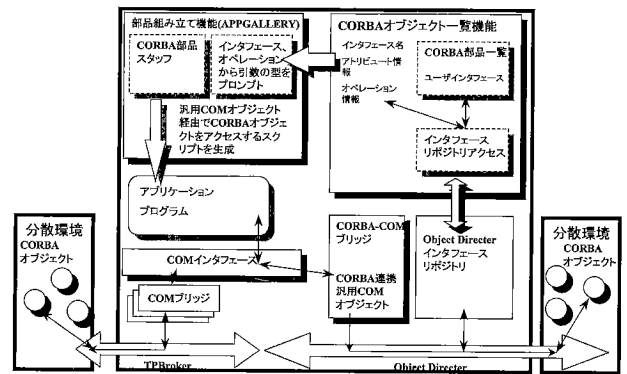


図-4 CORBA-COM接続環境

利用者の環境はすでにマルチプラットフォーム化が進んでおり、上記のような異なる分散オブジェクト環境上のオブジェクトを共用するアプリケーションを開発することも必要になってくると考えられる。ここでは、CORBAとCOMを例として、両者のオブジェクトを接続し実際にアプリケーションを開発することにより、異なる分散オブジェクト環境間のオブジェクトを接続する上で生じる問題点を洗い出すとともに、その解決法を実証することを目的としている。想定している環境を図-4に示す。

この実証実験は、異なる分散オブジェクト環境間の部品を組み合わせて、ソフトウェアを開発することが目的である。そのため、部品組立による開発環境として、APPGALLERY (日立製作所) を使用した。同製品はCOMオブジェクトを部品として、部品組立によりアプリケーションを作成できるビジュアル開発環境である。

また、ORBの実装による差異がCOMとの整合性に及ぼす問題を明確にするため、複数のORBを使用するものとしObjectDirector (富士通) とTPBroker (日立製作所) の2つを利用した。なお、CORBAとCOMの接続には、双方のORBに付属するCORBA-COMブリッジ機能を使用した。

CORBAとCOMを接続しアプリケーションを開発するための環境として、実証実験に先立ち以下の機能を開発した。各機能の関係を図-5に示す。

### 1) CORBA-COM部品組立機能

APPGALLERY上でCORBAオブジェクトの一覧表示し、選択されたオブジェクトを部品として使用するなどの機能を実装した。

### 2) テスト支援機能

1) の環境と連携して、作成したアプリケーションのテストを支援する。以下の機能を実装した。

- テスト画面自動生成機能
- 問題点表示機能
- 性能モニタ機能

### 3) COMオブジェクトからCORBA-IDL (Interface Definition Language) 生成機能

異なる分散オブジェクト環境をまたがってオブジェ

クトを移植する場合を想定し、その支援機能として指定されたCOMオブジェクトと同等のインタフェースを持つCORBAオブジェクトを定義するCORBAIDLを自動生成する機能を実装した。

これらの機能を開発する段階で、異なる分散オブジェクト環境間のマッピングで何点かの問題があった。代表的な例として以下の2つを示す。

- 1) COMインタフェースの型とCORBAインタフェースの型の違い  
 実証実験では双方のAndを取った形で実現した。
- 2) 例外の扱い

CORBAでは、オペレーションの例外はIDLのexception, raisin文などにより明確に意識し扱われるのに対し、COMではこのような仕組みがない。CORBAのオペレーションの引数に加えて、例外情報のオブジェクトをインスタンスとして扱えるオブジェクト型変数を追加したCOMオブジェクトにマッピングされる。

実証実験は、これらの機能を用いてアプリケーション開発作業を行い、その中でCORBA-COM接続による部品組立機能を使用した。

実験結果からCORBA部品のCOM接続技術は、異なる分散オブジェクト環境間の部品を組み立ててアプリケーションを作成する手法において、非常に有効であることが確認できた。またテスト支援機能により、CORBA部品単体でのテストや実行時の問題点表示・性能測定が実現できたことは、開発環境としての有効性も示せたと考える。さらにはインテグレータが保持しているCOM部品からIDL生成をすることにより、COM部品をCORBA部品として利用できる可能性を示すことができた。

実験により、複数の分散オブジェクト環境の部品を使用したアプリケーション開発が十分実用性のあることが実証できた。

### (III) レガシーシステムとの接続の実現とコンポーネントの移植性の確認/統合

#### 1) レガシーシステムとの接続の実現

部品組立型開発が、次世代のソフトウェア開発の主流になるためには、既存アプリケーションへの対応が重要なキーポイントとなる<sup>1), 2)</sup>。実証実験ではこの点を考慮し、人事マスタをレガシーシステムとして、CORBA環境へ移行する実験を行った。レガシーシステムをCORBA環境へ移行する際の共通的な移行手順を洗い出し、それを規約とし、一般的なデータベースシステムである人事マスタシステムをCORBAオブジェクトとしてラッピング<sup>3)</sup>する作業を行った。規約では、移行対象となるシステムの分析、移行時に実装するラッ

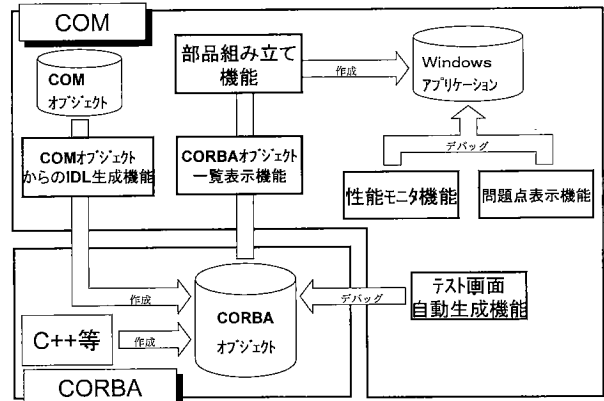


図-5 実装した機能

パー、アダプタの構造を挙げており、実証実験は、この規約が有効であるかを検証し、接続を確認した。ただし規約では、CORBA環境側からのレガシーシステムの管理方法が規定されておらず、統一的なオブジェクト管理の枠組みが必要であると考えている。

#### 2) コンポーネントの移植性の確認

部品組立型開発の最終段階において、SI社は図-1で示すように、複数のインテグレータの組み立てた部品を統合する役割も持つ。部品の統合では、各部品の仕様、動作の整合性が重要であるため、組み立てた部品の検査が必要となる。

実証実験では、部品の統合に向けて各部品の仕様、動作を検査するためのツールとして、WebからCORBA部品をテストするツールを開発し、部品統合が円滑に行われるかどうかを検証した。WebとCORBAを接続するテストツールの概要を開発プロセスとあわせて図-6に示した。

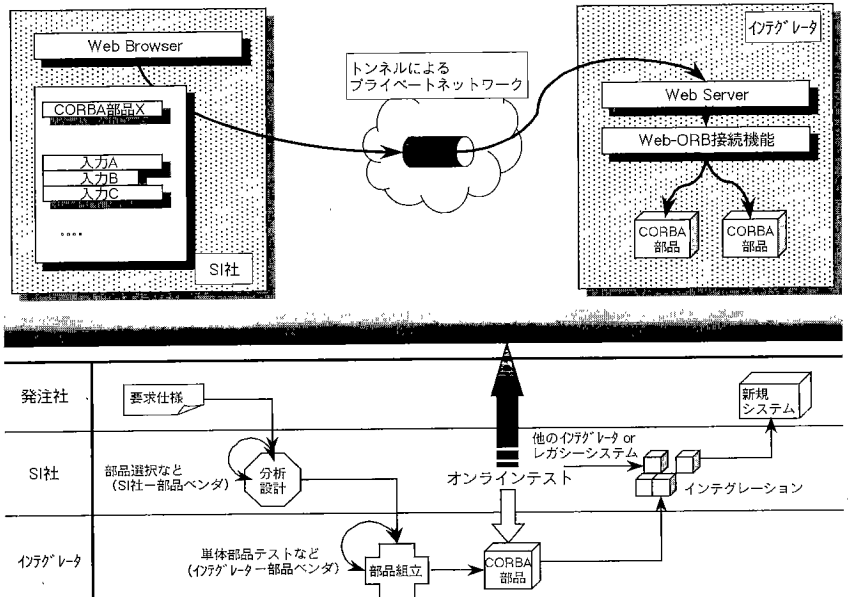


図-6 Webを介したCORBA部品のテスト

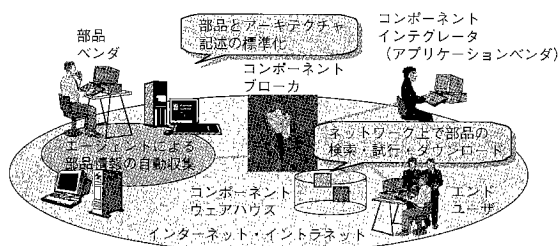


図-7 ベンダブローカインテグレータ・モデル

テストツールは、インテグレータ内で開発中のCORBA部品をSI社内からWebサーバを介して起動することを実現している。

実験では、実際にインターネットを介し、このツールを用いて複数企業間でCORBA部品のオンラインテストを行った。結果として、Webを部品検証の基盤として利用することにより、テスト環境の削減、検証作業の削減を実現でき、部品統合を円滑に行うことができたと考えている。

#### (IV) コンポーネントブローカ

##### 1) コンポーネントブローカとは

部品ベンダとインテグレータとの分化は、その間で部品の販売、流通などの取引（ブローカ）が行われる新たな職種を必要とする。これをコンポーネントブローカと呼ぶ。この結果、次世代ソフトウェア開発の産業構造として図-7に示す、ベンダブローカインテグレータ・モデルを想定している。

ソフトウェアはハードウェアと異なり、ネットワーク上で電子的に流通できる。インターネットの発展に伴い、ソフトウェアや部品をネットワーク上で販売、流通、保守することが可能となってきた。実際、いくつかのWebサイトでソフトウェアや部品の販売が始まっている。コンポーネントブローカはネットワーク上でソフトウェアやコンポーネントの電子商取引を行うサービスであると位置付けられる。

##### 2) コンポーネントブローカを実現する技術

###### a) ソフトウェアコマース言語SCL

ソフトウェアや部品を販売するためには、その仕様に加え商取引の情報を統一した形式で提供する必要がある。このため、ソフトウェアコマース言語SCL (Software specification and Commerce Language) を開発した<sup>6)</sup>。従来から、オブジェクトの標準インタフェース定義としてOMGのCORBA-IDLなどが提案されている。しかし、CORBA-IDLなどでは商取引情報を規定していない。

SCLはCORBA-IDLの上位互換としてインタフェース定義に加えて商取引情報も統合して提供する点に特長がある。さらに、電子商取引のために、Webのホームページとして定義できるように、HTMLの拡張として定義した。デザインパターンの1つであるFontDialogDirectorを例として、図-8にテキスト形式での記述例を、図-9にWeb上の記述例を示す。

##### b) エージェントによる部品情報の収集

インターネット上に分散するコンポーネントブローカのWebを探索し、部品情報を自動的に収集するソフトウェアエージェントシステムを開発した。

##### c) コンポーネントプレイヤ

コンポーネントが実際に利用できるかどうか試してみるコンポーネントプレイヤを開発した。プレイヤとのインタフェースは、SCLの1部として定義されているので、Web上で遠隔試行が可能となった。

##### d) 部品のネットワーク流通

コンポーネントインテグレータはネットワーク上でコンポーネントブローカからコンポーネントをダウンロードして組み合わせることができる。さらに、インターネット上のプッシュ技術の発展により、バグ修正や機能変更などに伴うコンポーネントの交換がネットワークを経由して行えるようになってきた。このように、コンポーネントブローカは部品の販売からアフターケアまでネットワーク上で一貫して実行することが可能となってきた。

##### 3) コンポーネントブローカのプロトタイプ

コンポーネントブローカのプロトタイプをWebサーバとして開発した（日本総研，新潟工科大学）。このプロトタイプを用いて、約16,000件にのぼる実際の部品情報の収集や検索などの実証実験を行った。

### ■実証実験に対する評価

今後のソフトウェア産業ではコンポーネントウェアを利用したビジネスモデルへ移行すると考えている。その想定のもとに、各種の職種に特化した企業群が、インターネットを利用してCORBA環境を基盤として水平協業するために必要な規約、ツール、環境を整備した。さらに、人事情報システムを実際に開発することにより、その有効性を6社の仮想企業からなる実験から実証した。

コンポーネントウェアを、今後複数職種企業間でビジネスモデルとして採用するには、次の技術的要件の考慮が必要である。

- A) 部品仕様やその購買情報の入手が容易
- B) 多様な部品の利用が容易
- C) 複数のサブシステムを、インターネットを通じ、1つの巨大システムとして開発可能
- D) レガシーシステムとの共存が容易

開発プロセスごとに、上の課題へどれだけ対応できるかを評価してみる。

##### (I) 部品組立、システムインテグレーション

多くのレガシーシステムが社会的に不可欠になっているが、ソフトウェア産業の近代化のためには、分散オブジェクト環境を利用したアーキテクチャへの移行が望ましい。レガシーシステムと共存しつつ徐々に移行するシナリオとして、またシステムの巨大化へ対応するため、CORBA環境を採用した。(I) (II) 「コン

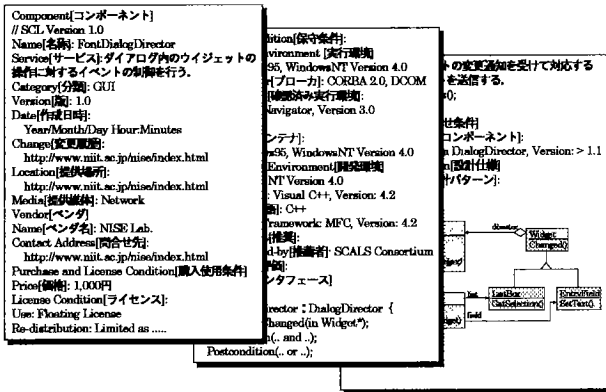


図-8 SCLの記述例

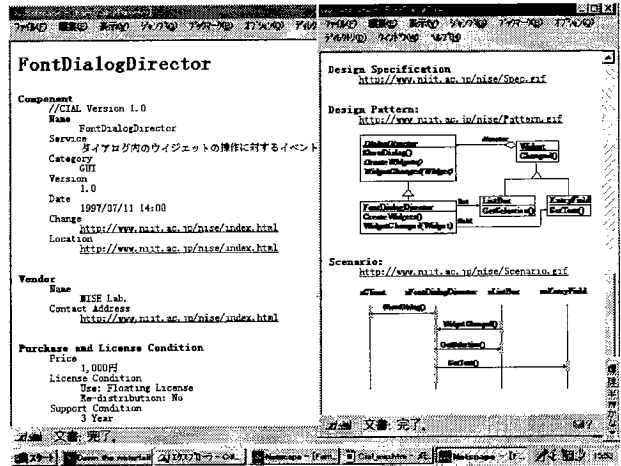


図-9 SCLのホームページ表現

「コンポーネント組立環境の企業間共用の実現」において要求C)が容易に実現できるための異なるORB間接続、ORB-リポジトリ間接続規約と環境を開発した。また、要求B)実現のためにCORBA部品をCOM環境でも利用可能なことを確認し、テストツールを開発した。また、(III)「レガシーシステムとの接続の実現とコンポーネントの移植性の確認/統合」において、要求D)の実現のためにレガシーシステムのCORBA環境への移行規約を開発した。製造された部品の検取作業に便利な、WebからのCORBA部品のテストを可能にする、Web-ORB間接続規約とツールを開発した。

(2) 部品検索

部品流通を広くインターネット上で実現するためには、その仕様や購買情報の記述方法の標準化が望ましい。そこで(IV)「コンポーネントブローカ」においてA)が実現できるための仕組みとして、仕様情報だけでなく商取引情報も取り入れた部品仕様表記要領を開発した。その要領に沿った情報を検索するツールも開発した。

このように、技術的な要件を満たすように規約が開発され、実験された。これらの規約が実用に耐えることが人事情報システムというレガシーシステムへ旅費精算、人事考課機能の追加というアプリケーションの開発体験により検証されたと評価する。

実証実験の範囲外であるが、このビジネスモデルの実用性の観点から、残る課題を2つ挙げておく。

まず、経済的な観点からの検討が必要である。実証実験担当者のコメントとして、コストの点では、従来よりも優れているという感触を得ている。

次に、部品が分散する場合には、通信量が従来よりも増大する可能性が高い。通信速度からくるレスポンスの遅れ、および通信コストの点から実用性の検討が必要である。

■実証実験のソフトウェア産業に与える影響

ソフトウェアCALSは、ソフトウェア開発において  
IPSJ Magazine Vol.39 No.9 Sep. 1998

各参加企業間の協調を一般化し定めることにより、ソフトウェア産業全体および各企業の生産性の向上をねらっている。本実証実験は、さらにもう一步踏み込み、コンポーネントウェアを採用した産業構造への転換を行い、ソフトウェア産業の近代化による劇的な生産性の向上を目指すものである。

技術的な観点からは、規約の開発およびその実証により、提案したビジネスモデルが有効であることが確認できた。経済性の観点での検討は今後を待つ必要があるが、実験担当者は、現時点でも実用的であるとの感触を得た。これは、現時点においても技術的な観点からはコンポーネントウェアを用いた企業間の水平協業ができることを意味している。

さらに制定した規約は、分散オブジェクト環境CORBAを基盤として用いて、Webブラウザからのテストや、COM部品の使用も可能としている。これは、過去の膨大なレガシーシステムの資産を活かしながら、コンポーネントウェアでの利用を可能にすることを意味する。さらに分散オブジェクト技術は、複数のシステムが分散したままで、1つの巨大システムとして連携できることをも意味する。

参考文献

- 1) OMG: Common Business Objects and Business Object Facility (1996).
- 2) Grothen, T. and Schwarb, R.: Implementing Business Objects: CORBA Interfaces for Legacy Systems. (1996).
- 3) Parodi, J.: Building "Wrappers" for Legacy Software Applications, 1996, [http://www.digital.com/info/objectbroker/product/obwp\\_wrap.htm](http://www.digital.com/info/objectbroker/product/obwp_wrap.htm)
- 4) OMG: The Common Object Request Broker: Architecture and Specification, Revision 2.0, <http://www.omg.org> ( July 1995).
- 5) 青山幹雄, 中所武司, 向山 博 (編): コンポーネントウェア, 共立出版 (1998).
- 6) 青山幹雄, 山下利夫, 小堀慎介: ソフトウェアコマースのためのカタログ記述言語SCL, 情報処理学会ソフトウェア工学研究会, No.115-5, pp.33-40 (July 1997).
- 7) 佐藤義男, 青山幹雄, 山下利夫, 村山一美, 高原 清, 安竹由起夫: 次世代ソフトウェアCALS基盤における部品組立型ソフトウェア開発, 情報処理学会ソフトウェア工学研究会, No.118-8, pp.55-62 (Mar. 1998).

(平成10年7月14日受付)