

東京証券取引所における株式約定接続支援システム

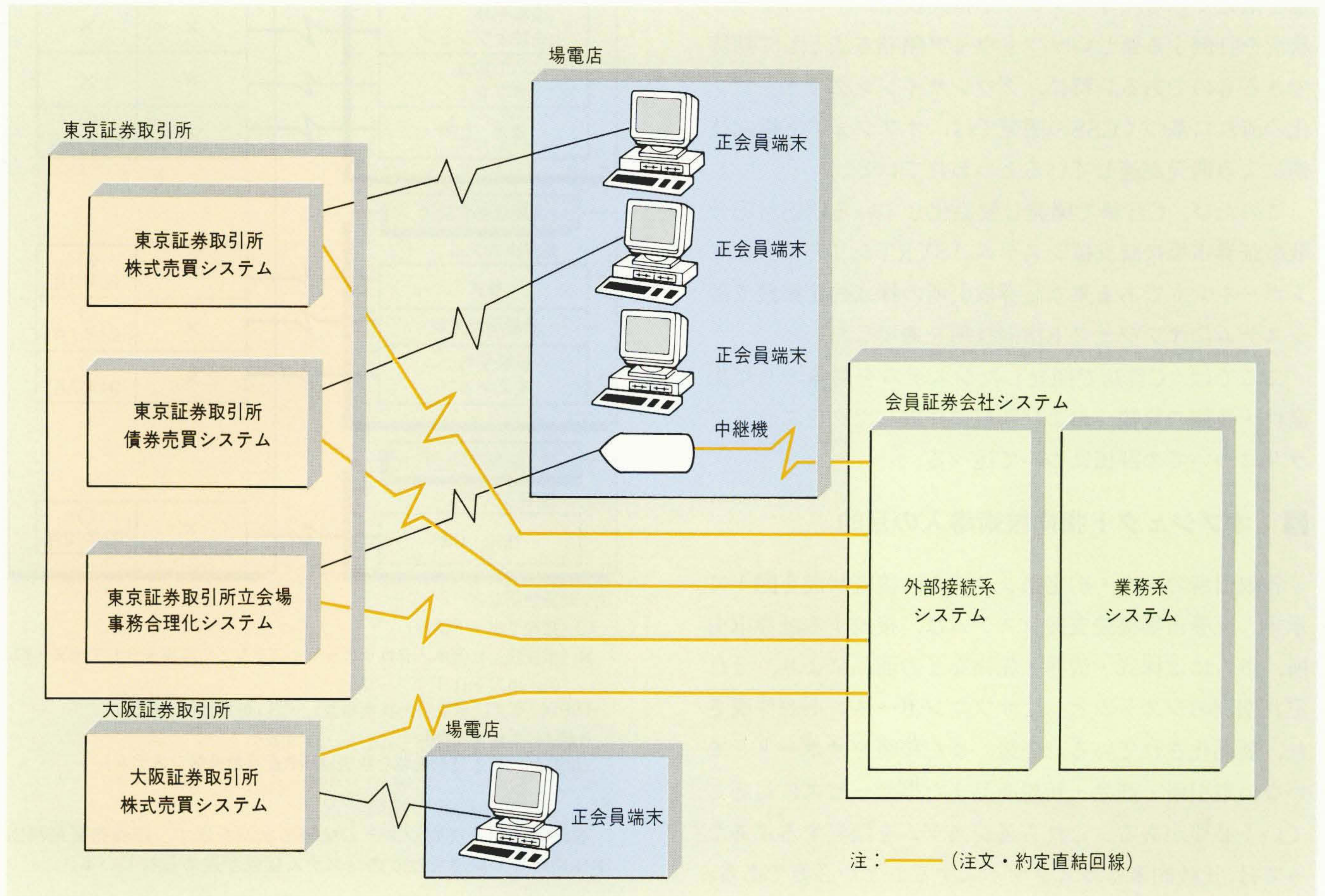
Securities Business System by Object-Oriented Technology

末光 哲* Satoru Suemitsu

井上文博* Fumihiko Inoue

曾根原 勝* Masaru Sonehara

伊野谷祐二* Yûji Inoya



証券市場接続支援システムの概要

証券市場接続支援システムは、証券市場システムと証券会社システムを接続するゲートウェイシステムである。証券会社システムのダウンサイジング・CSS(Client Server System)化を支援する。

証券業務の効率化・省力化が要望されており、証券取引所の売買システムと取引所会員(証券会社)の売買執行管理システムとの接続支援システムが必要である。このためのCSS対応分散型証券市場接続支援システム「SVRT(Communication Server Building Support System)シリーズ」を業界の動向に合わせて順次開発を進めている。

このたび、証券市場接続支援システムのコンポーネントである東京証券取引所の株式約定接続支援システム「SVRT/BA」にオブジェクト指向技術を適

用した証券市場接続支援システム向けクラスライブラリを開発した。

開発手順はオブジェクト指向開発に適しているといわれているスパイラル型開発手順を適用し、段階的に開発を行った。

証券市場接続支援システムは高い性能が要求されるシステムであるため、性能を重視した開発・評価を行った。その結果、C言語と比較しても遜(そん)色のない性能値を得ることができた。

* 日立製作所 情報システム事業部

1 はじめに

近年、ソフトウェアは大規模化、複雑化してきているため、生産性・保守性・拡張性などの面でさまざまな問題を抱えている。オブジェクト指向技術は、このような状況を打開する新しいソフトウェア開発方法として期待できるものである。特に、ダウンサイジングやオープン化の流れに基づくCSSの開発では、オブジェクト指向技術による開発が適しているといわれている。

このたび、C言語で開発し製品化しているCSS対応分散型証券市場接続支援システム「SVRTシリーズ」のコンポーネントである東京証券取引所の株式約定接続支援システムにオブジェクト指向技術を適用した。

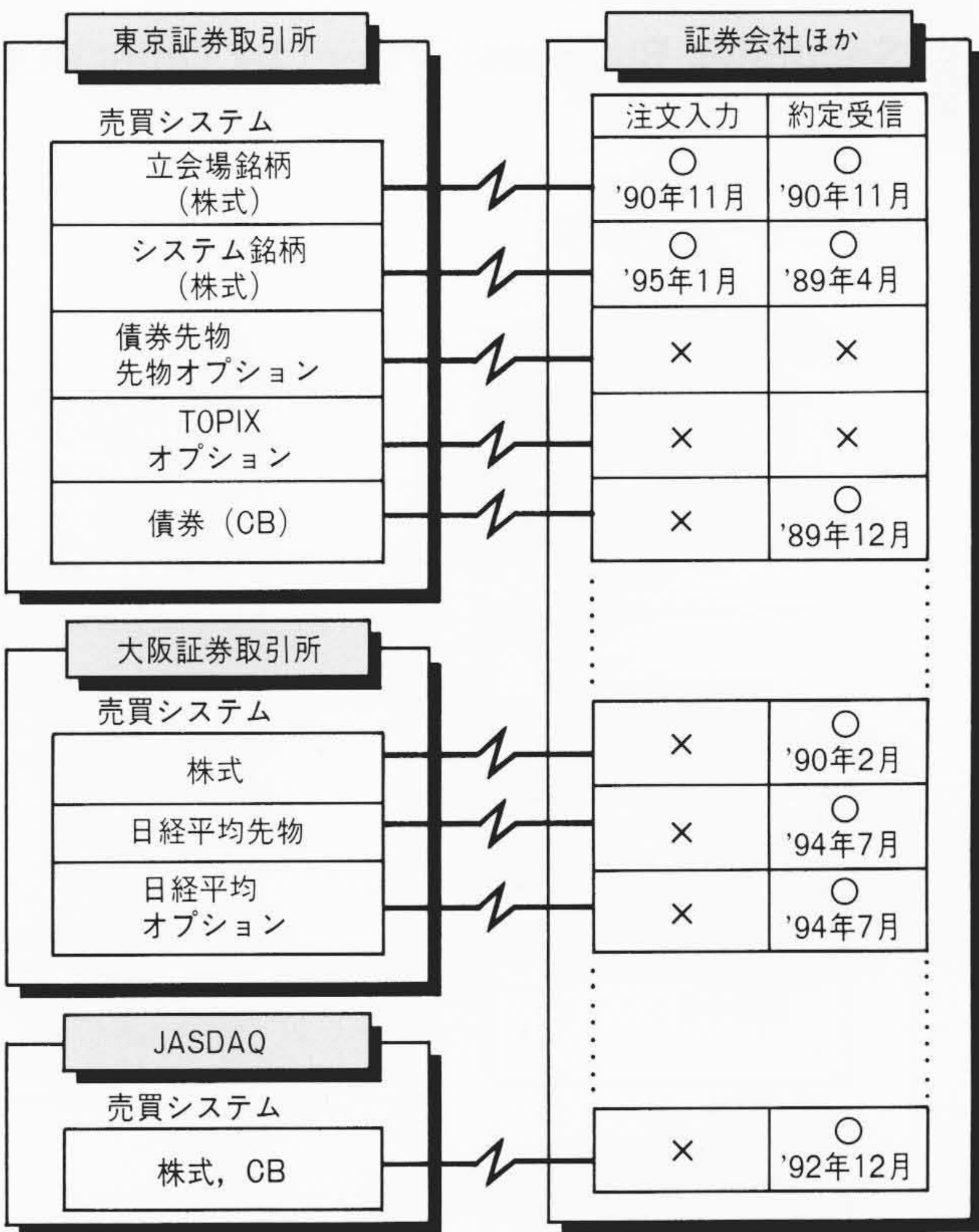
ここでは、C言語で開発したシステムを対象とした生産性・性能の比較、および新規に作成したクラスライブラリについての評価について述べる。

2 オブジェクト指向技術導入の目的

各取引所の注文・約定系システムの接続状況を図1に示す。証券市場接続支援システムは、接続する証券取引所、さらには株式・債券・先物などの商品により、それぞれ個別のシステムとしてサブコンポーネントが作成され、製品化されている。今後、まだ接続をサポートしていない取引所・商品・新規商品またはサービスに対応していく必要がある。これら各システムを開発するにあたっては、比較的多くカスタマイズすることが必要である。そこで、オブジェクト指向技術を導入し、証券市場接続支援システム向けクラスライブラリを作成した。そのクラスライブラリを再利用し、今後のシステム開発でのカスタマイズ時の効率向上を図ることを目的とした。

3 システム概要

証券市場接続支援システムのシステム構成を図2に示す。UNIX[®]上のOLTP(Online Transaction Processing)であるOpenTP1を前提とし、分散トランザクション処理ベースプログラム(SVRT/BASE)・共通制御機能(SVRT/CC)を基本プログラムとしている。分散トランザクション処理ベースプログラムにより、トランザクション制御などのシステム全体の基本的な制御機能が提供



注：略語説明など
○ (接続サポート済み)
× (未接続。ビデオ、フロッピーディスクなどの媒体を介して注文・約定の入出力を行う。)
TOPIX (東京証券取引所株価指数), CB (転換社債, 社債)
日経 (日本経済新聞)
JASDAQ (日本証券業協会店頭銘柄気配自動通報システム)

図1 注文・約定の接続状況
各証券取引所売買システム接続の状況を示す。証券売買業務効率化のため、証券会社とのシステム接続が進められている。

されている。
証券市場接続支援システムは、証券取引所の売買システムと取引所会員の売買執行管理システムを接続するAPP(Application Program Product)である。ステータス管理や通番管理を行うことにより、制御電文の順序性を監視したり、業務データの欠落を防止している。
今回のオブジェクト指向技術を適用した東京証券取引所の株式約定接続支援システムは、同取引所と取引所会員を接続し、会員から送信した株式注文に対する約定結果を東京証券取引所から受信し、電文の正当性を確認してから取引所会員へ引き渡すものである。

4 証券市場接続支援システムへのオブジェクト指向技術の適用

4.1 証券市場接続支援システムのクラス構成
証券市場接続支援システムを、メッセージ中継変換シ

※) UNIXは、X/Open Company Limitedがライセンスしている米国ならびに他の国における登録商標である。

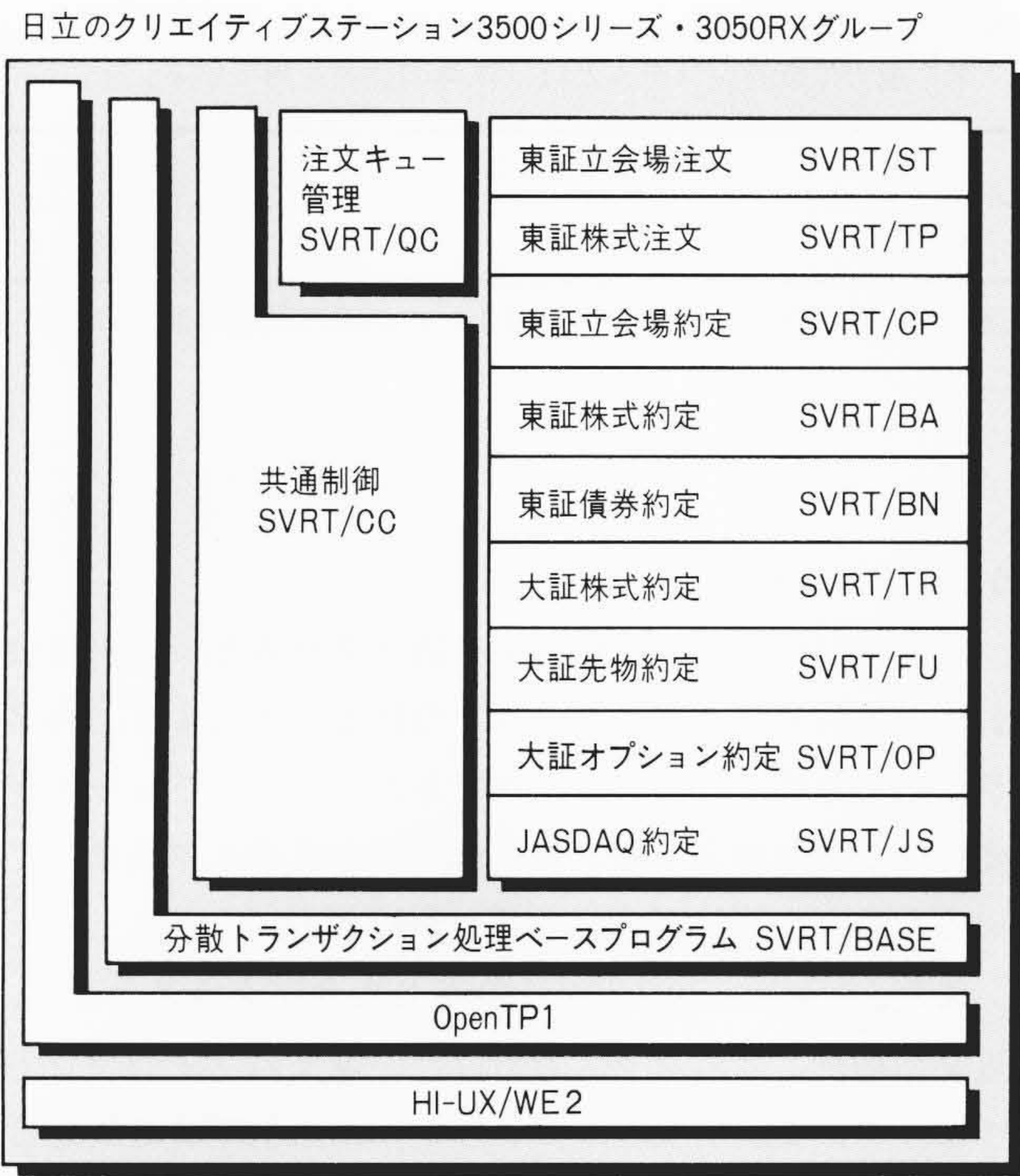


図2 システム構成
取引所別、注文・約定別に機能がメニュー化されているので、必要に応じてシステムを組み合わせることができる。

システムとして抽象化して設計を実施した。オブジェクト図を図3に示す。各システムの個別に作成するクラスはこれらを継承する。

4.2 SVRT/BASE下システムへのオブジェクト指向技術の適用

証券市場接続支援システムは、性能面などへの配慮から、証券取引所への回線数に対応してプロセスを複数起動し、各取引所との電文を処理している。これらのプロセスは動作上の情報を共有する。これらの共有情報は、プロセス間で共有するオブジェクトとして実装することになる。実装方式として幾つか考えられるが、共有する情報は信頼性が要求されるため、性能と障害時の回復処理を考慮し、図4に示すようにSVRT/BASEがインタフェースを提供するOpenTP1の高信頼ファイルを使用した。ここに共有情報を属性として保持・更新できるように設計した。

4.3 開発手順

開発手順としては、オブジェクト指向技術の開発に適しているスパイラル型開発手順を適用し、開発を二段階に分けて実施した。

第一段階では必要最小限の機能だけとして実現性を確認し、第二段階でその改善と機能追加を行ってシステム

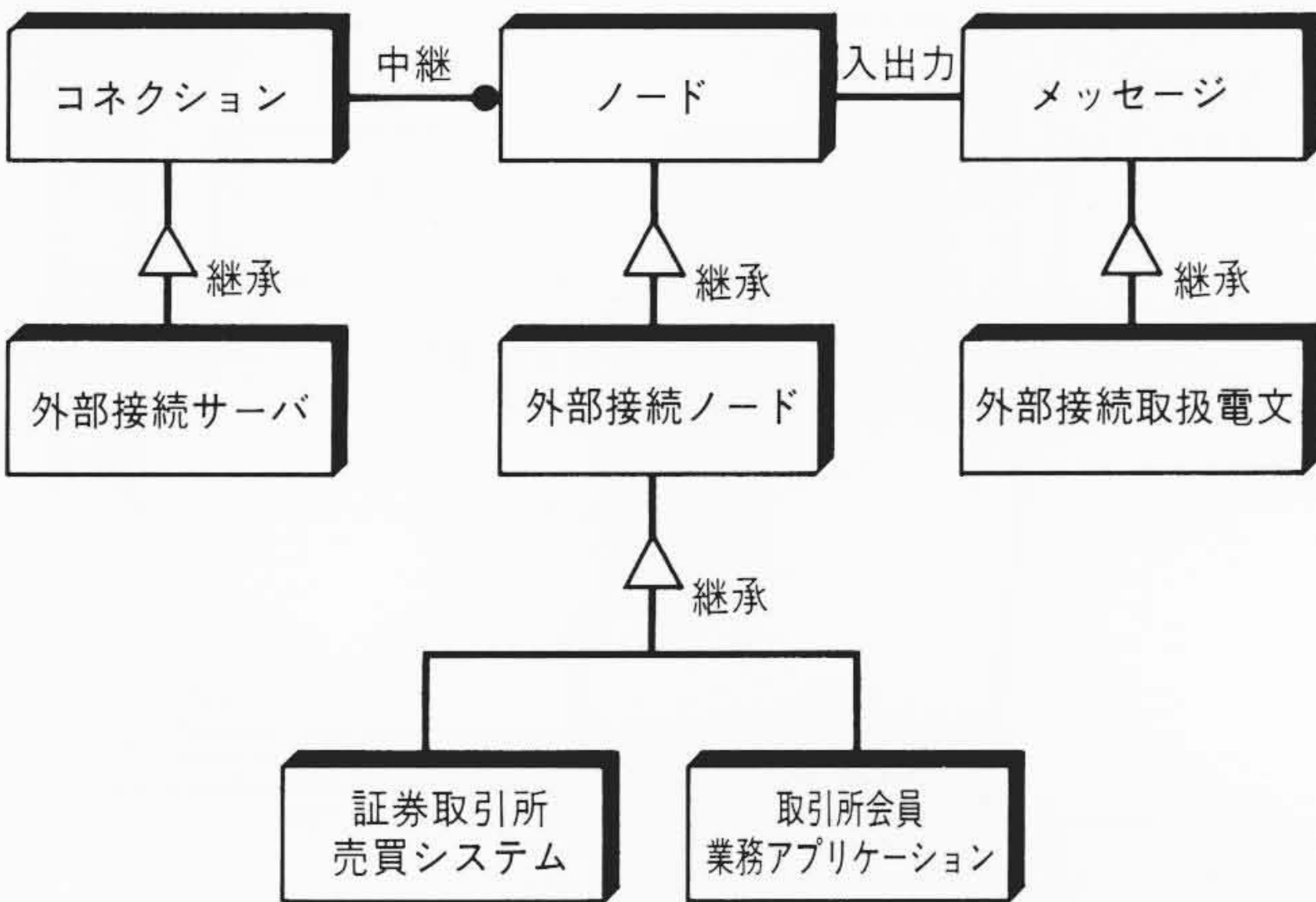


図3 証券市場接続支援システムにおけるオブジェクト図
これら8クラスは証券市場接続システム共通であり、各システム固有のクラスはこれらを継承する。

を構築した。

(1) 第一段階

第一段階では、最初にシステムとして動作する必要最小限の機能だけに絞り、実現性を確認した。実現性の確認では、特に性能に重点を置いて実施した。証券市場接続支援システムは高性能が要求されるため、当初のオブジェクトモデルや各クラスのメソッドの設計で性能が引き出せるか、ボトルネックはどこかなどを確認・検討するためである。

(2) 第二段階

第二段階では、第一段階での結果を踏まえ、設計の見直しを行い、性能向上施策と積み残した機能を盛り込んだシステムを構築した。第一段階ですでにシステムをダウンさせるような初歩的不良を除去できていたことや、不明確な点の洗い出しができていたことから、スムーズに作業を進めることができた。

5 適用効果

5.1 生産性

今回の適用は、再利用・流用クラスライブラリがなく、まったくの初期から行ったため、この章で述べる生産性のデータは、C言語による開発とC++による開発との比較により、一つのケースとしてとらえる必要がある。

ここでは、オブジェクト指向技術を適用してC++で作成した東京証券取引所の株式約定接続支援システムと、C言語で開発した同既存システムを比較する。

今回開発したシステムでは、既存システムよりも工数を55%削減した。次の開発以降は、クラスの再利用・流用ができるので、いっそうの工数削減が見込まれる。

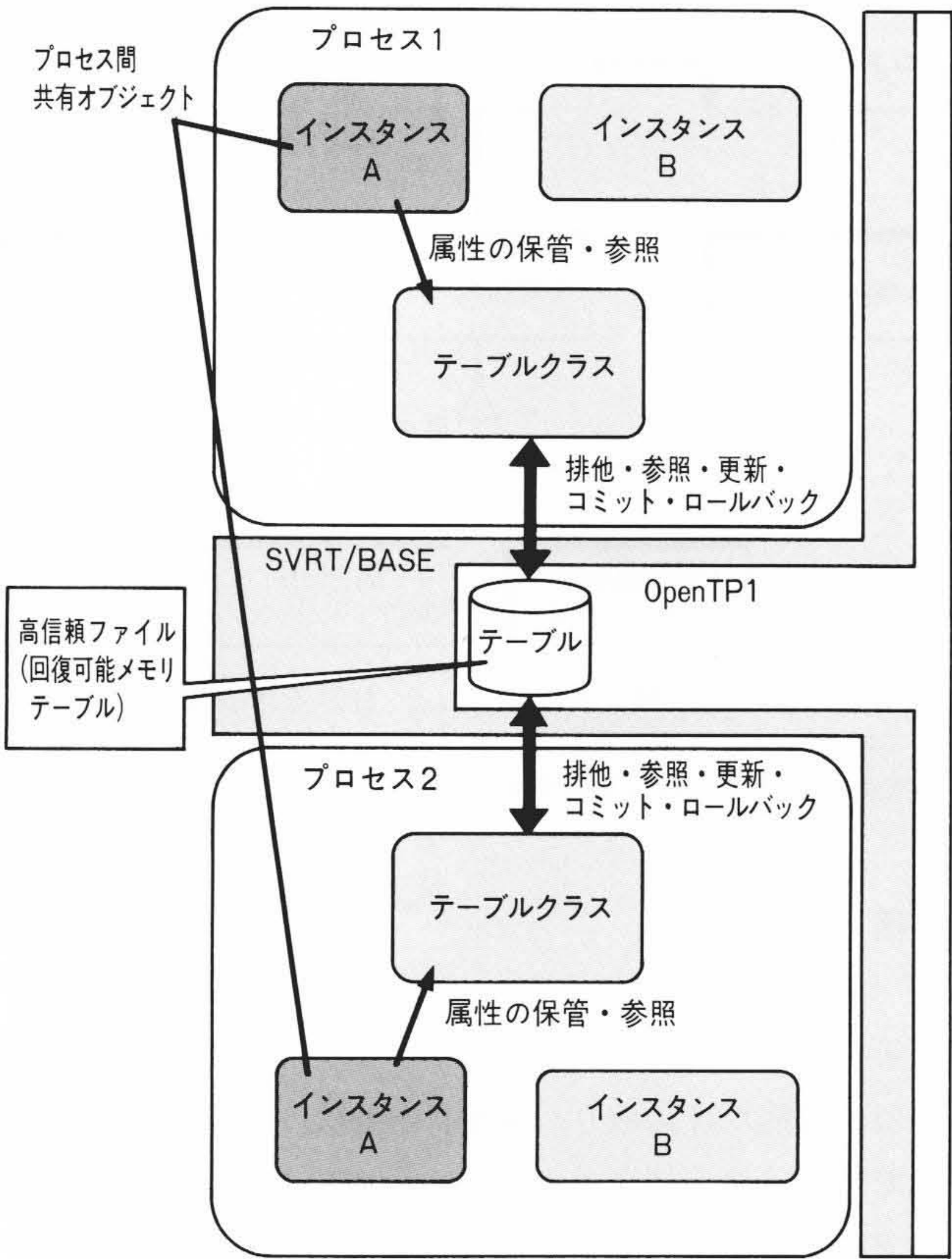


図4 プロセス間共有オブジェクトの実装
OpenTP1のテーブル上に属性を保管し、それを参照・更新することにより、プロセス間の共通オブジェクトとすることができる。

作成ステップ数では、既存システムの53%で開発することができた。ステップ数の減少は、不良の作りこみを減少させることにもつながるものである。さらに、プログラムの理解が容易になるため、将来の再利用・流用・保守作業に望ましい影響をもたらすことになる。

5.2 性能

性能については、既存システム、第一段階成果物、および第二段階成果物の三つで比較を行った。その結果を表1に示す。

今回の適用では、実現性の確認の結果をフィードバックした効果が現れ、既存システムとほぼ同等の性能を得ることができた。また第二段階での値は、システムとして要求される性能値の180%に達している。

表1 性能比較
第二版では既存システムとほぼ同等の性能を得ている。

	既存システム	第一版	第二版
性能値*	100	91.0	98.4

注：* 既存システムの単位時間当たりの処理件数を100とした値

5.3 クラス抽出

証券取引所からの受信電文を扱うクラスと、取引所会員への送信電文を扱うクラスを分けることにより、両クラスの独立性を高めることができた。インタフェースが変更になった場合でも、変更個所が1クラスに集まっているため、容易に対応することができる。

また、メッセージクラスを継承したサブクラスとして、電文の種別ごとにクラスを設けた。電文のフォーマットチェック、業務アプリケーションへの送信電文編集をこれらのサブクラスのメソッドとした。電文のフォーマットをメッセージクラスにカプセル化することにより、他のクラスの電文フォーマットへの依存度を極小化することができた。

6 おわりに

ここでは、OpenTP1下で動作する証券市場接続システムにオブジェクト指向技術を適用した事例について述べた。

今回の適用では、性能に重点を置いた設計を行ったため、C言語と比較しても遜色のない性能値を得ることができた。また生産性では、クラスライブラリを再利用することにより、いっそうの効率向上を見込むことができる。

今後はさらに開発を進め、より再利用性の高い証券市場接続支援システム向けクラスライブラリを構築し、そのクラスライブラリを用いて品質、保守性および生産性に優れた製品を開発していく考えである。

参考文献

1) J. ランボー, 外: オブジェクト指向方法論OMT, トッパン (1993)
2) 水野: プロトコル言語, カットシステム (1994-7)