

分散コンポーネントによるコミュニティウェア基盤の実装¹

5 J-4

久保 裕也², 金子郁容³

慶応義塾大学 政策・メディア研究科

1 はじめに

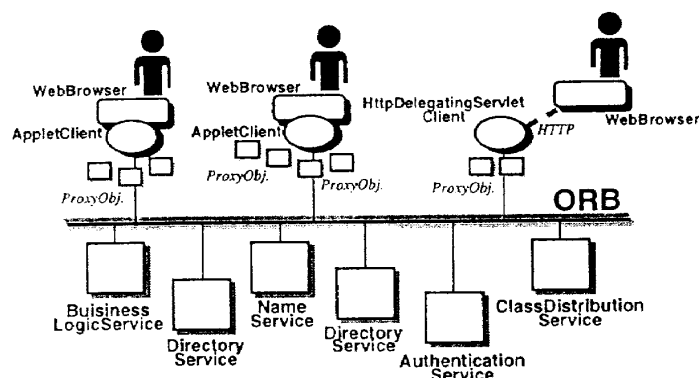
DNS, NIS+, X500などは、大規模複雑系を階層的権限委譲という情報ガバナンス手法によって一定の成功を収めてきた。しかし今日では、そうした階層性を再構成する際のコストの高さや、名前空間の衝突の調停に関するルールの不備なども露見してきており、純粋な階層的権限委譲だけを管理手法とする形態では不足であるというように考えられる。計算機環境が多ユーザ・分散環境として社会化するのに伴い、可伸的・適応的な資源管理・知識構造化・意思決定支援を行うことは、ますます難しい問題となってきている。

我々は、人々の自発的関与によってフラジイルな関係として形作られる「知縁的コミュニティ」の組織原理に着目している。階層的情報管理手法の次世代を担うパラダイムとして、コミュニティメンバーを自律的で責任能力のあるシステム運営主体とみなし、相互的・動的に権限を付与することでその力を活用するための方法論を体系化することが求められていると考える。本稿は以上のアイデアを実際にアプリケーションシステム的设计問題に展開し、その具体的な実装と運用について述べるものである。

2 Nomos—自己記述的システム

Nomos⁴システムは、汎用的なコミュニティウェアを形成する基盤としてデザインされる。アプリケーションは分散オブジェクト関連の各種技術をコンポーネント化したものとして、再構成可能な部品群として提供される。ユーザは自身にその部品の運用と、場

合によっては開発までを担うこととなる。システムはその運用・編集を通じて自己記述的に最適化をされてゆく。



3 コンポーネント環境の構成

動作単位

Nomosのシステムでは、それぞれのユーザが自発的に、自己の興味関心に基づいてアプリケーションやデータを準備し、それらを自分がアカウントを持つワークステーション上でHORB [3]の分散オブジェクトとして公開する。これがNomos動作単位となり、それぞれネットワークサービスとしての機能をボトムアップ的に構成する。個々の分散オブジェクトサービスは、相互に接続・協調することによって、一群のアプリケーションとして動作する。

クラス配布基盤

分散オブジェクトサービスであるNomosを利用するために使われるORB Proxyは、Nomos自身を基盤として提供される。すなわち、Nomosの一部であるディレクトリサービスとHttpServletが、ユーザのローカル環境にNomosシステムのクライアントを構成するのに必要なクラスを提供し、ダウンロードさせる。

¹Communityware Base by Distributed Components

²Hiroya KUBO

Graduate School of Media and Governance,
Keio University

5322 Endo, Fujisawa, Kanagawa 252, Japan

E-mail: hiroya@netcom.mag.keio.ac.jp

³Ikuyo KANEKO

Graduate School of Media and Governance,
Keio University

⁴ギリシア語で「人為」「言語」「理性」の意

認証・セキュリティ基盤

セキュリティ機構の実装基盤については、HORBに備わっている分散 Access Control List を拡張する形で実現している。基本的に認証システムは抽象クラスとして提供される。具体的な認証システム運用は、そのサブクラスの実装によって提供する。たとえばクラス配布サービスを実際に運用する上では、ディレクトリサーバへのクラスの登録の際のソース内容検証・署名発行や、クラスのダウンロードの可否を制御するなどといったように、慎重な設定を施す必要がある。コミュニティ内での役割モデルなど、動的な条件付け権限に応じてサービス利用内容を制御する。

永続オブジェクト基盤

多ユーザで共有され、仮想マシンプロセスのライフサイクルを越えて永続する必要があるオブジェクトについては、MVC 構造の設計手法を採用することにより、Modelドメインに当たるものとして、サーバ側に集約的に保持されることになる。

永続オブジェクト基盤には、ObjectDesign 社の ObjectStorePSE [4] を利用している。現在の実装では、この永続オブジェクトを複数のリモートクライアントオブジェクトから共有する際の、スレッド同期にボトルネックがある。この問題を改善するためには、本格的なオブジェクト指向データベースである ObjectStore(PSE と API 上位互換) へ下位の実装基盤をアップグレードすることが可能である。

Applet,HttpServlet,Servlet などによるサービス中継基盤

ORB 上の各種のサービスは、Applet クライアント、HTTP による Web クライアント、SMTP による Mail によるクライアントなど、プロトコルを交換する servlet を介在させることで、様々なクライアント環境上で利用することができる。このことは、サービスの適用範囲を広げ、可用性を高めることに繋がる。

リフレクションによるコンポーネント自動構成・ディレクトリサービス基盤

Java1.1 から採用された Reflection API [5] により、オブジェクトに定義された属性やメソッドを動的

に検知できる。この機能を利用することで、各種の分散オブジェクトサービスを協調的に動作させる際の相互運用上のインターフェイス整合を検査する。特に、一旦格納された永続オブジェクトや、動作中の分散オブジェクトの内部状態を検査して、その値を適宜修正するようなサービスを作り出すためにこの API を利用する。

オブジェクト指向データベースには、RDBMS における SQL のような標準的な表・列の操作手続きが存在していなかったが、このリフレクション機能を利用することで、オブジェクトの各階層の内容を一般化し、データベース・ディレクトリサービスのようにアクセスさせることができる。この機能を認証機構と併せることにより、データの分散管理・分散運用アプリケーションを実現できる。

4 まとめ

本稿では、Nomos システムの着想とその構成について、自己記述的コンポーネントによる創発的システムの、Java 分散オブジェクト関連技術の応用による実装についてを述べた。今後は、システムの運用実績の蓄積と、導入事例による評価・検証を行いつつ、コンポーネント群の充実化を進めていきたいと考えている。

参考文献

- [1] 金子, 松岡, 下河辺. 自発する編集的相互性. ボランティア経済の誕生 P.401-P403. 実業之日本社. 1998.
- [2] HIRANO Satoshi. HORB: Distributed Execution of Java Programs. WWCA97 "WorldWide Computing and Its Applications. 1997.
<http://www.horb.org/>
- [3] Object Design, Inc.
<http://www.odi.com/products/products.html>
- [4] Sun Microsystems, Inc.
<http://www.javasoft.com/products/jdk/1.2/docs/guide/reflection/>