

4F-3

# 電子メールをユーザインタフェースとした 分散トランザクションシステムの構築

市橋 幸治† 大久保 英嗣†† 溝井 仁††† 寺本 大輔†††

†立命館大学大学院理工学研究科 ††立命館大学理工学部情報学科

†††ディー・エス・ソフトウェア

## 1. はじめに

近年、情報システムのネットワーク化が進む中で、分散環境に対応したトランザクションシステムの必要性が増している。そこで、我々は、分散環境におけるアプリケーション開発を支援するためのプラットフォームである分散トランザクションシステムIXIを開発してきた。現在、その後継として、トランザクションを分散環境上に存在する資源にアクセスする際の論理的な単位として捉え、インターネットに適應した分散トランザクションシステム Mobile-IXI の開発を行っている。Mobile-IXI は、インターネット環境を考慮して、電子メールを基本にユーザインタフェースを構築している。これにより、Mobile-IXI は、ユーザ環境を幅広く許容し、モバイルコンピューティングに対しても適用することが可能となっている。

## 2. Mobile-IXI の概要

### 2.1 Mobile-IXI の特徴

Mobile-IXI は、次のような特徴を持っている。

1. 電子メールを基本にユーザインタフェースを構築している。

電子メールの性質上、送受信時のみネットワークに接続していればよいため、接続を長時間にわたって維持する必要がない。したがって、接続に伴うコストが削減される。また、電子メールの送受信に関する知識があれば、Mobile-IXI を利用することが可能となる。さらに、電子メールの送受信の機能のみを有すればよい点から、ユーザ側の環境を選ばない利点もある。

2. システム記述言語としてオブジェクト指向言語である Java を採用している。

これにより、オブジェクト指向や Java 言語の有する優れた点を継承できる。すなわち、Java の実行環境さえあればプラットフォームに依存することなく Mobile-IXI を稼働することができる。さらに、Java のクラスライブラリを用いることで、アプレットやアプリケーションの形で、Web 上の GUI の実装を容易に行うこともできる。

3. 非同期かつ非リアルタイムなシステムである。

メディア変換などの時間を十分にとれるため、マルチメディアにも対応しやすい。さらに、他の多くのトランザクションシステムが抱える、リアルタイム処理に関わる解決困難な問題から解放される利点もある。また、トランザクションを、ファイルなどの形で蓄積しておくことも可能である。

4. トランザクションの処理単位を電子メールとしている。

これにより、スケジューリングにおいて電子メールを単位として扱うことができるため、ステートメント単位で管理するよりも並行処理制御などの機能を簡略化でき、トランザクションの管理が容易になる。

### 2.2 システム構成と処理の流れ

図1に Mobile-IXI のシステム構成を示す。

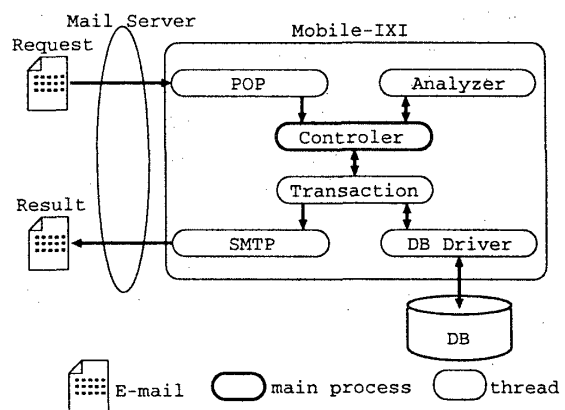


図 1: Mobile-IXI のシステム構成

An Implementation of Distributed Transaction System with E-Mail Based User Interface

Koji Ichihashi †Eiji Okubo ††Hitoshi Mizoi †††Daisuke Teramoto †††

†Graduate School of Science and Engineering, Ritsumeikan University

††Department of Computer Science, Faculty of Science and Engineering, Ritsumeikan University

†††D. S. Softworks CO., LTD.

Mobile-IXIでは、ユーザからの電子メールを入力とし、ユーザへの結果通知も電子メールとしている。Mobile-IXIへの要求メールには、電子メール本来のヘッダ、Mobile-IXI固有の制御情報、実際のオペレーション記述(データベース操作言語など)が含まれる。処理の流れは、以下のようになる。

1. メールサーバから電子メールを取得する。
2. 電子メールを Mobile-IXI の処理形式に変換する。
3. 電子メール毎にトランザクションスレッドを起動し、電子メール内の記述に従って、データベースアクセスなどの処理を行う。
4. 処理結果を含む電子メールを生成し、ユーザに送信する。

さらに、Mobile-IXI 各部の機能は次のようになっている。

**Controler** : Mobile-IXI 全体の制御、主にスレッドの生成と管理を行う。

**POP** : Mobile-IXI 内で常に動作し、POP3 を用いてメールサーバから電子メールを取り出す。

**Analyzer** : 電子メールを Mobile-IXI の処理形式に変換する。

**Transaction** : トランザクション制御(並行処理制御および一貫性制御など)を行う。

**SMTP** : 処理結果を含む電子メールを生成し、SMTP でユーザに送信する。

**DB Driver** : データベースを操作するためのインタフェースを提供する。

### 3. Mobile-IXI 間の通信

Mobile-IXI は、単体でも機能するが、単体では操作可能な DB などのアプリケーションのシステム形態に限られる。そこで、現在、その拡張性や制御の容易な点を利用して、分散環境に適応した機能の実装を行っている。本実装では、Mobile-IXI 間のインタフェースも電子メールとし、電子メールを用いたサーバ間通信を行う。これにより、従来のコネクションベースのサーバ間通信よりもコストの低い制御が可能となる。

Mobile-IXI 間の通信は、ユーザとの通信と基本的には同じである。異なる点は、制御情報が拡張されることである。これは、Mobile-IXI 間の通信においてはシステムの処理結果と要求記述が併存するためである。さらに、Mobile-IXI では、多様なアプリケーションを実現するために複数の処理方式を採用し、それらの制御を行う必要がある。このために、Mobile-IXI では責任集中型と責任委譲型の2つの処理方式を採用している。以下でそれらについて述べる。

責任集中型では、図2で示すように、ユーザから最初に要求を受け取った Mobile-IXI が電子メールの分割を行い、各々の要求を処理可能な Mobile-IXI に転送する。さ

らに、各々の Mobile-IXI からの結果を集約し、ユーザに送信する。したがって、ユーザから要求を受け取って、結果を返すまでを1つの Mobile-IXI が保証する。

責任委譲型では、図3で示すように、要求を受け取った Mobile-IXI が自分で処理可能な部分のみを処理し、その結果をメールに添付して、電子メール内の他の要求を処理可能な Mobile-IXI に転送する。これを繰り返し、すべての処理結果が揃った時点でユーザに送信する。したがって、各々の Mobile-IXI は自分で処理可能な部分のみを保証する。

これらの処理方式は使用するアプリケーションによって選択される。また、多様なアプリケーションに対応するため、これらの処理方式の併存も可能にすることを考えている。

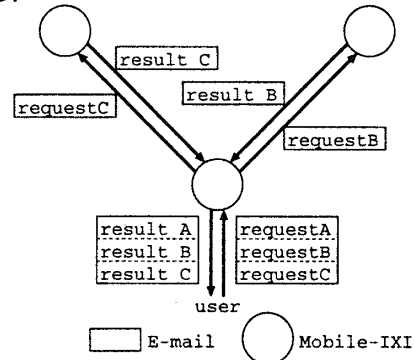


図 2: 責任集中型

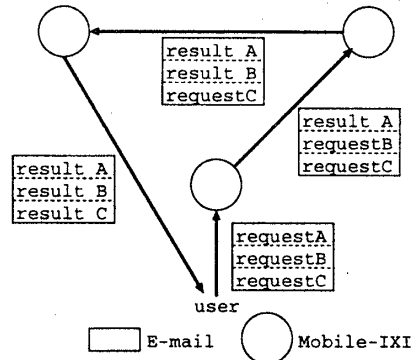


図 3: 責任委譲型

### 4. おわりに

本稿では、分散トランザクションシステム Mobile-IXI の概要と処理方式について述べた。現段階では、既存のアプリケーションであるデータベースと電子メールを操作対象とすることで、ユーザの要求を実現している。今後は、操作可能な既存アプリケーションを増やすとともに、独自のアプリケーションを実装することを考えている。また、アプリケーション開発者が、Mobile-IXI 上のアプリケーションを容易に構築するために、基本的な機能を実現するためのスレッドライブラリの構築を予定している。