
発表概要

Mobile JDK: モバイルエージェント環境における 計算機資源への移動透過な操作が可能なクラスライブラリ

星名大輔[†] 関口龍郎[†]
山内裕史[†] 米澤明憲[†]

モバイルエージェントはネットワークでつながれた計算機の間を移動し、自らの仕事を完遂するために、それぞれの計算機資源（例えばファイル・ソケット・GUI）を利用する。資源の中には、エージェントの移動先の計算機上に複製するのが困難なもの（ファイル・ソケット）と複製するのが容易なもの（GUI）がある。既存の Java ベースのモバイルエージェントシステムでは、エージェントが移動後もこれまで操作していた資源を操作したいとき、移動を起こす命令の前後に、各資源ごとにその資源への切断や再接続、もしくは、複製のためのコードを、プログラマが明示的に書かなくてはならない。移動は、実行中の様々な時点で起こり得るため、そのようなコードを明示的に書くのは困難であり、かつ、プログラムのモジュラリティの低下につながる。そこで、我々は次の3つの特徴を持つクラスライブラリを提供する。(1) 資源へ継続的に操作するための機構が資源を扱うクラス内部に実装されている。(2) 使用中の資源への操作を場所透明に行える。(3) 資源を扱うクラスがあるパッケージについて、標準のクラスライブラリと各クラスのインターフェイスやクラス間の階層構造が同じである。これらの特徴により、プログラマは資源への継続的操作のためのコードを書く必要がないため、プログラムのモジュラリティが増加する。また、モバイルエージェントは自身と使用中の資源がネットワークのどこに存在していても、その資源を操作できる。さらに、移動を考慮していない資源を扱う既存の Java プログラムが最小限度の変更でモバイル環境で使えるようになるという利点がある。我々の主な成果は、標準のクラスライブラリのインターフェイスを基に、上の3点の機能を持つようなクラスライブラリを自動で生成するための方針を考案したことである。

Mobile JDK: Java Class Library Supporting Migration-Transparent Resource Access

DAISUKE HOSHINA,[†] TATSUROU SEKIGUCHI,[†] HIROSHI YAMAUCHI[†]
and AKINORI YONEZAWA[†]

Mobile agents migrate over a network of computers and exploit computational electronic resources (e.g., files, ports and GUI components) to carry out their tasks. Some resources (e.g., files and sockets) can not migrate with agents and others can create their copies at the destination host. Mobile agents often need to access those resources after migration. In existing mobile agent systems, however, the programmer has to write explicitly the instructions that disconnect and reconnect to resources at migration. Since migration may happen at any time during execution, it is difficult to write such instructions. We develop a class library which has the following features: (1) Each class includes a mechanism that allows mobile agents to access resources continuously (2) It offers operations to access resources in a location-transparent manner. (3) The same interfaces and the same class hierarchy as those of the standard JDK. These features improve the modularity of a program because the code that realizes the effect of migration on resources is encapsulated in our library. A mobile agent can operate on resources continuously after migration with our class library. Furthermore, any existing Java program written without considering migration can also be used in a mobile environment with a small amount of modifications. Our main contribution is that we develop general translation rules that generate a set of classes satisfying the above three features from a set of non-mobile resource classes by examining their type signatures.

(平成 12 年 3 月 23 日発表)

[†] 東京大学理学系研究科情報科学専攻
Department of Information Science, Graduate School
of Science, University of Tokyo