Agent-Based House-Buying System Demonstration

Hidehito Gomi and Daishi Kato, NEC Corp
Kyohei Kawazoe, INTEC Web and Genome Corp
Rada Chikorva, Jung Ung Min, Keesoo Kim, and Waqar Mohsin, Stanford University
Overview

- Objective of demo
- Agent-based house-buying scenario
- System architecture
  - MobidgetLite
  - Infomaster
  - JATLite
- Agent communications among MobidgetLite, Infomaster, and JATLite
- Message translations
- Distributed agents on wired and wireless network
- Screen shots
Objective of Demo

- Mission of FX-Agents Research
  - To construct an extensible open e-marketplace and populate it with working autonomous software agents for the financial domain.

- Objective of Demo
  - To construct an example of extensible open e-marketplace and populated it with limited numbers of working autonomous software agents for the real estate market, as an instance of the financial domain.
Agent-Based House-Buying Scenario

Listing agent
“Mitsui_Yellow_Pages”

Buyer
“Taro”

Titlecompany
“Fidelity_National”

Insurers
“State_Farm” and
“Allstate”

Lender
“Wells_Fargo_Bank”

Seller
“Akira”
System Architecture

**JATLite**
Developed by CDR, Stanford University

**MobidgetLite**
Developed by NEC

**Infomaster**
Developed by Logic Group, Stanford University

**Facilitator**

**Listing agent**

**MobidgetLite-Infomaster Gateway agent**

**MobidgetLite-JATLite Gateway agent**

**JATLite Router**

**Lender agent**

**Insurer agent**

**Titlecompany agent**

**Buyer agent**

**Java Objects**

**KQML-Classic Strings**

**KQML-Classic Strings**

**KQML-Classic Strings**

**Java Objects**

**Java Objects**

**Java Objects**
MobidgetLite: An Agent Server for E-Commerce

- HTTP Server (Tomcat + Servlet)
- WWW Gateway (special agent)
- Parse XML messages
- Java Objects
- XML strings
- MobidgetLite Agent Server
- Asynchronous Message passing
- Original format/protocols In MobidgetLite
- Languages used in agent communication depends on implementation of agents
Infomaster: An Information Integration System

User Interfaces

User Agents

Knowledge Base

Facilitator

Wrapper Agents

Sources

SQL

TDT

KIF
JATLite: A Java Agent Infrastructure with Message Routing
Agent Communication between MobidgetLite and JATLite

MobidgetLite-JATLite Gateway Agent

JATLite Agent

JATLite Router

MobidgetLite Agent Server

Java VM

Java VM

Java VM

JATLite system

Demo system agent

Java Objects

(KQML)

application agents

Java Objects

(KQML)

Java Objects

(KQML)

Parse KQML messages

KQML strings

KQML strings

KQML-Classic strings

Java Objects (contains KQML instance)
Agent Communication between MobidgetLite and Infomaster

- MobidgetLite-Infomaster Gateway Agent
  - Java Objects (contains KQML instance)
- Infomaster system
  - Infomaster Facilitator
  - Infomaster Listing Agent
- KQML-Classic strings
- Java VM
- Java VM
- Demosystem agent
- Java Objects (KQML)
- application agents

Infomaster system

MobidgetLite system

Java VM

Java VM

Java VM
Agent Communication between JATLite and Infomaster

JATLite Agent <-> JATLite Router <-> JATLite-Infomaster Gateway Agent <-> Infomaster Listing Agent <-> Infomaster Facilitator

JATLite system <-> Infomaster system
Message Translation

KQML:
(performative
 :sender <name>
 :receiver <name>
 :content <expression>)

KQML Classic:
(PACKAGE
 :sender <name>
 :receiver <name>
 :content (performative <expression>))

Java Object: ?

Direct Mapping

JATLite
Infomaster
Mobidget Lite
Distributed Agents on Wired and Wireless Network
JATLite-Infomaster
Gateway agent

JATLite Router
@logic

JATLite Insurer
“Allstate”

JATLite Titlecompany
“Fidelity_National”