

Verifying Mobile Agent Design Patterns with RPOO

Elthon Alex da Silva Oliveira

Federal University of Alagoas - Campus Arapiraca, Brazil

Emerson Ferreira de Araújo Lima

Federal University of Campina Grande, Brazil

Jorge César Abrantes de Figueiredo

Federal University of Campina Grande, Brazil

INTRODUCTION

The act of modeling concurrent distributed systems is not a trivial task. Besides, when mobility is added to the scenario, things get worse because of new problems like variations in the communication conditions and remote execution. An important thing to be considered is how to analyze these mobile agent-based systems in order to validate and improve them.

An interesting tool to verify such systems is named Petri net. Petri net, or simply PN, is a powerful formal, graphical, and executable tool commonly used for verifying communication protocols and concurrent systems. A variant of Petri net, named RPOO (Guerrero, 2002)—Object Oriented Petri net, brings all the advantages of Petri nets and object-oriented semantics and puts them together. This union produces a new tool, with the formal semantics of Petri net models and the object-oriented semantics of some programming languages, like C++ and Java. This makes the semantics of formal behavior models closer to the semantics of those programming languages. With this, formal models of programs are built easier than with classical Petri nets. Besides, a closer model, speaking about semantics, is used to verify and analyze mobile agent design patterns which gives a more trusted verification.

This article presents the formalization and analysis of three migration design patterns—iterinary, star-shaped, and branching—done by using of RPOO. A brief comparison between RPOO models and classical Colored Petri net (Jensen, 1992, 1997) models is also briefly presented.

BACKGROUND

Mobile Agent Migration Design Patterns

In this article we consider three migration design patterns proposed in Tahara, Ohsuga, and Honiden (1999): iterinary,

star-shaped, and branching patterns. In the sequence we detail each pattern. We used a message sequence diagram to show an overall picture of the design patterns.

Iterinary

This pattern provides a way to execute the migration of an agent, which will be responsible for executing a given job in remote hosts. The agent receives an iterinary on the source agency, indicating the sequence of agencies it should visit. Once in an agency, the agent executes its job locally and then continues on its iterinary. After visiting the last agency, the agent returns to its source agency. This pattern is a good solution to agents that need to execute sequential jobs. In Guedes, Machado, and Medeiros (2003) and Medcraft (2003), case studies that apply this pattern are shown.

In Figure 1, we present a possible execution sequence for this pattern. We use a notation that is equivalent to the one presented in Klein, Rausch, Sihling, and Wen (2001). In this notation, an object is used to represent an entity that controls agents' execution in a given agency (creation, destruction, migration) and indicates their location. Migrations are represented by message passing from one agency entity to the other. The message is labeled as MIGRATING AGENT. Before migration, agent execution is interrupted (arrow labeled as destroy()). Execution is continued in the target agency (arrow labeled as initialize()).

As we can see, there are three agencies: a SourceAgency and two search agencies (DestinationAgency1 and DestinationAgency2). Following the diagram, we see that there is an agent (IterinaryAgent) that sets its iterinary, moves to the first search agency where it executes its job, then it moves to the second one, executes the job, and returns to the source agency.

Star-Shaped

On the star-shaped pattern, the agent receives a list of agencies that it has to migrate to. Initially, the agent migrates to

Figure 1. Message sequence chart for Itinerary

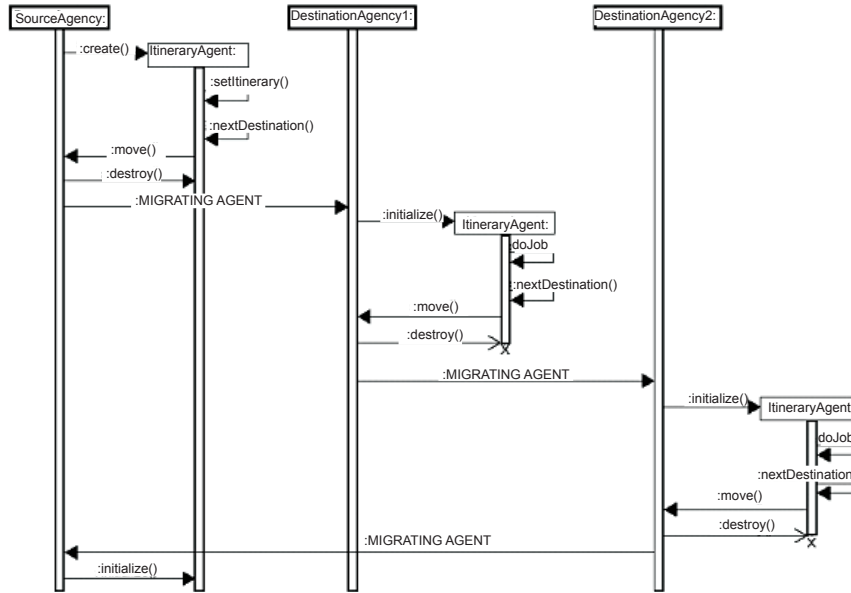
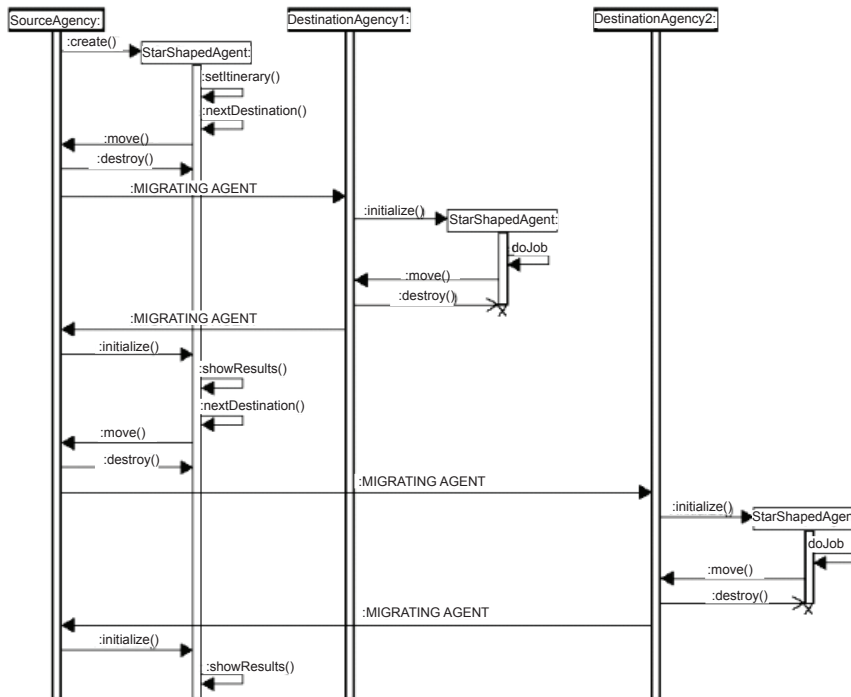


Figure 2. Message sequence chart for Star-Shaped



the first destination agency in the list. After migration is completed, it executes the relevant job and resumes migration going back to the source agency. The agent repeats this cycle until the last agency on its list is visited. The advantage of this pattern is that the agent stores the results of its job in the source agency and does not need to migrate to the others' agency with them. Depending on the application, the results can be shown to the user as soon as the agent stores them in the source agency. In this way, the user can

already know the partial results before the agent finishes its migration through all search agencies.

In Figure 2, we can see an execution sequence for the Star-Shaped pattern. In this diagram, we have the same configuration of the sequence diagram shown for the itinerary pattern: three agencies and one agent. Following the diagram, we observe that the agent sets its itinerary and then travels to the first search agency. After executing its job, the agent returns to the source agency, where it stores the job's result.

7 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/verifying-mobile-agent-design-patterns/17207

Related Content

Assessing Human Mobile Computing Performance by Fitts' Law

Thomas Alexander, Christopher Schlick, Alexander Sievert and Dieter Leyk (2009). *Mobile Computing: Concepts, Methodologies, Tools, and Applications* (pp. 206-224).

www.irma-international.org/chapter/assessing-human-mobile-computing-performance/26501

"Checking Into" Outdoor Lifestyle?: Mobile Location-Based Games as a Site of Productive Play in Marketing Campaigns

Elaine Jing Zhao (2014). *Interdisciplinary Mobile Media and Communications: Social, Political, and Economic Implications* (pp. 245-258).

www.irma-international.org/chapter/checking-into-outdoor-lifestyle/111727

Investigating the Associated Factors of Trust on Online Transactions

Andriew Lim (2018). *Mobile Commerce: Concepts, Methodologies, Tools, and Applications* (pp. 1295-1302).

www.irma-international.org/chapter/investigating-the-associated-factors-of-trust-on-online-transactions/183341

The Use of Embedded Mobile, RFID, and Augmented Reality in Mobile Devices

Greg Gogolin (2012). *Mobile Technology Consumption: Opportunities and Challenges* (pp. 126-137).

www.irma-international.org/chapter/use-embedded-mobile-rfid-augmented/60216

Performance Evaluation of Multimedia Traffic Transmission Under Binomial and Poissonian Primary Traffics in Cognitive Radio Networks

Abdelaali Chaouband Elhassane Ibn-Elhaj (2012). *International Journal of Mobile Computing and Multimedia Communications* (pp. 87-105).

www.irma-international.org/article/performance-evaluation-multimedia-traffic-transmission/69535