

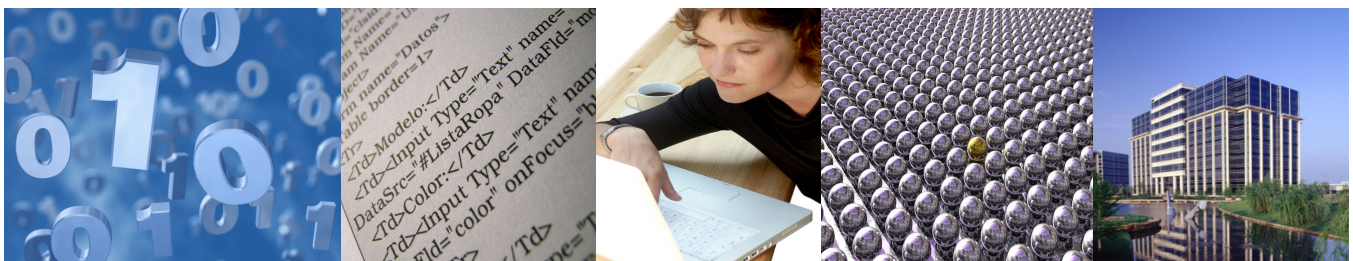
# Fault-Tolerant Simulation Environments Using Intelligent Mobile Agent Technology

By Bob DeAnna

**Recursion Software, Inc.**

March 6, 2006

Version 1.0



## TABLE OF CONTENTS

Abstract.....	1
Overview.....	1
Technical Overview .....	2
Figure 1. Agent-Based Linking of Distributed Systems Architecture .....	2
The Adapter Agent.....	2
Agent Manager .....	3
External Actors .....	3
Infrastructure Requirements and Protocol Interoperability.....	3
Primary Goals of VOYAGER <i>Edge</i> ® .....	3
About the Author.....	4



# Fault-Tolerant Simulation Environments Using Intelligent Mobile Agent Technology

By Bob DeAnna

---

## Abstract

DOD, NASA, FAA, DOE, industry and academia use simulators to conduct studies otherwise infeasible or impossible to perform and measure. In the past, advancements in high-performance computing have focused on producing larger and more accurate simulations, while the problem of how to transform the output -- possibly terabytes of data -- into knowledge has been neglected. Recursion Software is working with its clients to offer a solution to the current problems faced in simulation environments including large amounts of data to be processed across less-than-reliable mobile networks.

VOYAGER *Edge*®, a new distributed development platform from Recursion Software, utilizes an agent-based approach to link and manage a heterogeneous system composed of datasets, sensors, and other simulations (such as weather simulations). Simulation Training Evaluation Systems can leverage this to provide pertinent, timely information to participants using PDA's, laptops, and personal workstations. This approach includes mobile agents for reducing bandwidth usage and accessing, filtering and analyzing data at its source. VOYAGER *Edge* is a leader in this technology.

---

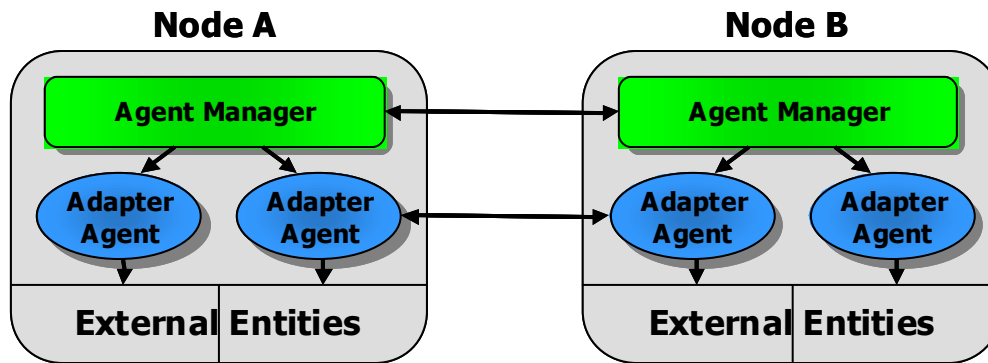
## Overview

- Mobile agents, deployed to edge devices by VOYAGER *Edge*, process and combine raw data from large, distributed, heterogeneous, multi-dimensional data sets with complex spatial and/or temporal dynamics to detect patterns and irregularities.
- Patterns and irregularities are detected using learning algorithms and data mining to examine correlations and perform classification, clustering, change and deviation detection, summarization, and dependency modeling.
- Mobile agents are used to pre-process, filter, polish, and extract features from raw data on edge devices ensuring that knowledge is transmitted rather than large amounts of raw data.
- Mobile agents select relevant subsets of data, remove noise and outliers, and decide on a strategy for parsing the data in the most efficient manner to a variety of devices across wireless networks.

## Technical Overview

Using Adapter Agents and Agent Managers, VOYAGER *Edge* provides the infrastructure for mobility in a distributed system including remote communications, security, code mobility (Mobile Agents), and proxies.

**Figure 1. Agent-Based Linking of Distributed Systems Architecture**



**Key Architectural Components:** VOYAGER *Edge* Agent Managers and Adapter Agents

**External Entities:** Various Data Sources, Data Streams and Application APIs

**Nodes:** Workstations/Servers, Sensors, PDAs and Smart Phones

## The Adapter Agent

- Provides a connection between a local client and the larger system. Each Adapter implements the Adapter Agent interface.
- Communicates with the local client and provides a consistent interface to the other system components. The local client can be as diverse as a simulation, an external sensor, or user client workstation.
- Maintains a local table with available services provided by the client and known services that the client uses.
- Is an Agent Manager client. It relies upon the Manager for suggesting candidate services for fulfilling a client request for service.
- Evaluates available services and chooses which to use by evaluating service descriptions.

Each existing system has an Adapter that publishes its interface to the Manager, which offers functionality and data to distributed consumers. The Adapter Agent knows how to access, filter and analyze its local simulation data. Mobile agents are key to scalability for reducing network bandwidth usage. A Mobile Agent can be dispatched by an Adapter or Agent Manager to a remote platform, interact with the system hosted there, which might involve receiving large data volumes from the system, and after filtering and analyzing the data, send the results back to the client from which the mobile agent had been dispatched. This results in reduced network traffic and vulnerability to network outages.

Peer-to-peer interaction between Adapters via Agents makes the architecture more fault-tolerant and not dependent on a centralized routing system.

## Agent Manager

The Manager brokers links between Adapters. The Managers are themselves implemented as scalable agents. Adapters publish their services with the Manager, which maintain a local service database. Adapters make queries for services that match their needs. Once a service has been identified, the Adapter can communicate with it directly, without going through the Manager.

Mobile Agents are created and dispatched to remote platforms by Managers to interact with another Adapter or External Actors. Mobile Agents can have hard-code behavior in Java, and will also draw from intelligent systems technology, using an engine and a knowledge base. Mobile Agents are also used for load-balancing purposes to transfer computationally intensive work to underutilized platforms within the system confined.

## External Actors

External actors can be discrete event simulations in a Simulation Training and Exercise system, external sensors, or Simulation Training workstations. Each external actor must implement the standards interface to Adapter. An external actor requests services through their Adapter, or vice-versa, using the standard interface and receives replies with data.

## Infrastructure Requirements and Protocol Interoperability

The infrastructure needed is at least intermittent network connectivity, computing platforms that can run the Java Virtual Machine (JVM), and have VOYAGER *Edge* installed. PDA's and smart phones will have a lightweight VOYAGER *Edge* client installed for hosting and/or accessing Mobile Agents, dependent on the device specifications.

The VOYAGER *Edge* system uses the CORBA (IIOP), Web Services (SOAP), J2EE (RMI-over-IIOP) and XML-RPC industry standards to implement the remote communications and mobile agent capability. It readily integrates on diverse platform configurations due to the ubiquity of Java and the JVM.

## Primary Goals of VOYAGER *Edge*

Recursion Software is working with its clients to develop real-time decision systems for commercial, civilian, and military distributed computing platforms. Over the long term, VOYAGER *Edge* will help to:

- Drastically reduce the amount of raw data sent across wireless networks by enabling intelligent, onboard analysis utilizing mobile agents on edge devices to perform more of the data gathering, filtering and analysis required by distributed computing platforms.
- Reduce operating costs and make more efficient use of simulation environments.

## About the Author

Bob Deanna is the Chief Technology Officer at Recursion Software, Inc. Bob has 22 years of experience in software architecture, development and mentoring. Bob's expertise is in distributed application frameworks such as J2EE, CORBA and ATMI. He has architected and developed applications and middleware in Java, C/C++, COBOL, PLI, and Assembler on operating systems ranging from MVS to Unix, Linux and Windows. Bob has worked previously in numerous roles ranging from VP of architecture, senior consultant, lead engineer and trainer/presenter. In these capacities, Bob has worked either as an employee or long-term consultant for such companies as Verizon, ATT, Swiss Re-Insurance, State Farm Insurance, Bloomberg LP, 3M Health Information Systems, Merant Software and Zeosoft.

Bob received a BS in Mechanical Engineering from Rutgers University and a continuing education degree in C/C++ and Unix Programming from New York University.

He may be contacted by email at [bdeanna@recursionsw.com](mailto:bdeanna@recursionsw.com).



Copyright © 2006 Recursion Software, Inc. All rights reserved. Recursion Software, its logo, CINERGI, VOYAGER, VOYAGER Edge, C++ Toolkits, .NET Toolkit, and JGL Toolkit are trademarks or registered trademarks of Recursion Software, Inc. in the United States and other countries. All other names and trademarks are the property of their respective owners.

Recursion Software, Inc.  
2591 North Dallas Parkway  
Suite 200  
Frisco, Texas 75034  
1.800.727.8674 or 972.731.8800  
[www.recursionsw.com](http://www.recursionsw.com)