2010 Seventh International Conference on Information Technology

Flexile Middleware: Responding to Operational Demands of Critical Network-Based Applications

Thomas A. DuBois
The Boeing Company
Ridley Park, PA

Richard Perry
Villanova University
Villanova, PA

Abstract

The best sources of critical network-based applications are operations that require ad hoc network formation, interoperability among a wide variety of node capabilities, real-time or near real-time responsiveness, high bandwidth data dissemination, quality of service, and execution of network applications that may have an impact on human life or other effect deemed critical by human society. This paper will discuss how these network-based operational applications are driving the need for autonomic (or self-managing) middleware. A new concept, called flexile middleware, will be introduced as an approach to address this emerging need.

Key Words: adaptive middleware, autonomic middleware, self-managing middleware, heterogeneous networks, interoperability, mobile ad hoc network.

1. Introduction

The integration of four areas of development is necessary to build a network for tactical systems [1]: (1) Communication System; (2) Network Processing; (3) Net-Ready Applications; and (4) Middleware. The first three areas will be discussed briefly with attention given to middleware and the introduction of flexile middleware.

2. Communications, Processing, and Net-Ready Applications

Multiple assets collaborating together to accomplish a mission will most probably have different communication systems operating on different frequencies with different waveforms and transmission range. This operational environment creates challenges for networked solutions: (a) Design a system that is open enough to allow the use of different radios without requiring changes to the software; (b) Ensure the system has the growth to support insertion of guard, anti-tamper, and other information assurance technologies as they become available; and (c) Establish a set of minimum performance criteria for interim communication systems until more fully compliant systems enter production.

The network processor hosts the middleware, net-ready applications, and the interface software. A network processor will typically contain one or more processors and graphics cards. Network processing also includes routers and switches. There can be quite a large range of network processing from as small as a handheld device to as large as a major computing installation. Available memory and processing speed has a direct impact on the number of applications and middleware services that can be installed. This type of network heterogeneity drives the need for adaptive and self-managing middleware.

The main purpose of net-ready applications is improved situational awareness. Applications result from user desires for capabilities to support operational requirements. Networked communications between assets collaborating to accomplish a mission can decisively improve performance.

Operational mission requirements determine the net-ready applications. Physical factors can scope the number and types of applications that can be installed. The net-ready applications determine which middleware services are needed. The middleware also has a direct impact on processing throughput and memory.

3. Middleware and the Introduction of Flexile Middleware

In the software architecture of a networked system, middleware is software that resides between the applications and operating system [2]. It is software that consists of a set of services that allow multiple processes running on one or more machines to interact across a network. It provides publish-subscribe capabilities, discovery, message translation, and other basic services needed by network-based applications.

Adaptive middleware has emerged as a topic of considerable interest motivated by the understanding that most ad hoc distributed systems have limited resources, intermittent connectivity, and are hybrid. When middleware can adapt to the context of the distributed system, efficiencies can be realized. Context-aware adaptive middleware can address quality of service requirements, improve the perception of quality, and enhance the ability to deliver service [3].

Middleware can be either static or dynamic. Static middleware can be customizable or configurable. Dynamic middleware can be tunable or mutable. Mutable

978-0-7695-3984-3/10 $26.00 © 2010 IEEE
DOI 10.1109/ITNG.2010.116

middleware allows for the most significant changes in behavior that can occur while it is executing [4].

Flexile describes something that is able to flex or bend easily. Flexile middleware has both static and dynamic features. It is static by containing a set of core services that must be present in all processors across the network. It is dynamic in multiple ways. Knowledge of the operating environment and mission objectives will be used to set parameters used by the middleware. The services themselves would not change, but they would have adaptive qualities resulting from domain knowledge embodied in parameters used by the middleware. A second level of adaptive behavior occurs when the services themselves need to change based on user intent and the ability to fit applications with associated services with the host's processing capabilities. It is this second type of adaptive behavior, which is the inspiration for the term, flexile middleware.

Middleware that is flexile has the following characteristics:

(a) Adaptive in both performance and size;
(b) Service content determined by applications;
(c) Domain-specific rules determine priorities;
(d) Performance monitor that assesses performance versus accomplishing mission objectives; and
(e) Ability to reconfigure middleware services based on usage statistics, learning rules, etc.

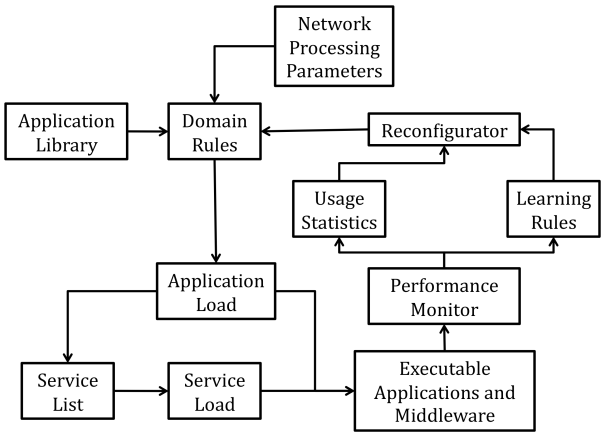

Figure 1. Flexile Middleware Block Diagram

Network Processing Parameters – These parameters are specific to the allocation of resources on host hardware to support network functions.

Application Library – This is a database containing the complete set of available net-ready applications.

Domain Rules – This is a knowledge base designed to determine which applications are the best ones to install on the host hardware.

Application Load – This represents the "best" subset of net-ready applications that should be loaded on the host processor.

Service List – This is a database containing the complete set of middleware services available to all of the net-ready applications.

Service Load – This represents the subset of services used by the net-ready applications residing in the Application Load.

Executable Applications and Middleware – This is an executable image of the net-ready applications and middleware services.

Performance Monitor – This module monitors system usage and occasionally calls the *Reconfigurator*.

Usage Statistics – This is a database created by the *Performance Monitor*. It is used to learn how the system is being used and to automatically ascertain user intent for network usage.

Learning Rules – This is a knowledge base governing the extent to which the *Domain Rules* can change, and specifically how to change those rules.

Reconfigurator – This module uses *Usage Statistics* and *Learning Rules* to propose new *Domain Rules*.

4. Summary

Net-ready applications designed for critical missions offer the potential to improve the effectiveness of those missions. Networks to support these missions will need to be mobile, ad hoc, and highly heterogeneous. These factors contribute to the growing need for middleware that is adaptive in terms of behavior and structure. Flexile middleware that "bends to fit" usage, operational, and physical constraints is proposed to address this need.

5. References

[1] T. DuBois, "Networking Challenges for Tactical Aviation Systems", *Presentation to Airborne Networks Conference*, Washington, DC, November 5-6, 2009.

[2] P. Naur and B. Randell ed., "Software Engineering", *NATO Software Engineering Conference*, Jan 1969, p. 14.

[3] J. Loyall, P. Pal, K. Rohloff, M. Gillen, "Issues in Context-Aware and Adaptive Middleware for Wireless, Mobile Networked Systems", BBN Technologies, Cambridge, MA.

[4] S. Sadjadi, "A Survey of Adaptive Middleware", Software Engineering and Network Systems Laboratory, http://users.cis.fiu.edu/~sadjadi/Publications/AdaptiveMiddlewareSurvey.ps, Department of Computer Science and Engineering, Michigan State University, MI.