ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Event Manager for Distributed Computing

Mr. Venkatesh Deshapande
M. Tech Final Year,
Department of Electronics and Communication Engineering,
Global Institute of Engineering and Technology,
Hyderabad.

Mr. G. Ahmed Zeeshan
Associate Professor,
Department of Electronics and Communication Engineering,
Global Institute of Engineering and Technology,
Hyderabad.

Abstract:

The event manager is a very sophisticated, high performance and precise solution for the challenges of distributed computing environment. Event Management Service (EMSv) is a publish/subscribe multipoint-to-multipoint communication mechanism based on the concept of event channels, by which a publisher communicates asynchronously through events with one or more subscribers over an event channel. Events consist of a standard header and zero or more bytes of publisher event data. Multiple publishers and multiple subscribers can communicate over the same event channel. Individual publishers and individual subscribers can communicate over multiple event channels. Publishers can also be subscribers on the same event channel. Subscribers are anonymous, so they can join and leave an event channel at any time without involving publisher(s).

Introduction:

The existing system and platform supports only Linux system based messages queue service. To implement the complex event processing with the message queues is very cumbersome and hard to achieve in the high performance distributed computing application systems. The existing message queue system approach can only solve the purpose of standalone computing environment. Few other disadvantages are like:

a) Event communication is only limited to standalone application computing. There is no support for the internodes communication.

b) Doesn't support message persistency in the case of system failures, thus renders applications to lose critical data.

The proposed Event Management Service (EMSv) is a publish/subscribe multipoint-to-multipoint communication mechanism based on the concept of event channels. Individual publishers and individual subscribers can communicate over multiple event channels. Event Manager comprises two subparts distributed among many processes across the distributed computers such as Event Management Server and Event Management Agent. Event Manager comprises two subparts distributed among many processes across the distributed computers:

- Event Management Server
- Event Management Agent

Event Management Server:

Event Management Server (EMS) implements the core functionality of EMSv. It maintains publisher and subscriber information and events published with retention time. When subscribing on an event channel, a process must specify which filters to apply on published events. EMS then matches the "patterns" accompanying the published events with filters installed on event channels by the subscribers before delivering the events to them. EMS runs on System Manager Nodes. One active and one standby instance of EMS exist in a distributed computing cluster.

Event Management Agent:

Event Management Agent (EMA) is the library that any process must link with to access EMSv functionality. It supports all the APIs of Event Management Service implementation. It provides APIs for various EMSv functionalities including life-cycle APIs and others to open/close event channels, subscribe/unsubscribe events, and publish events.

Events published by invoking the publish API in the Event Management Service are forwarded to EMS by EMA. The EMA also carries out preliminary event processing, channel access verification, and other event channel-related operations.

Description:

The Event Management Server (EMS) implements the core functionality of EMSv. When subscribing on an event channel, a process must specify which filters to apply on published events. EMS runs on System Manager Nodes. One active and one standby instance of EMS exist in a distributed computing cluster. The Event Management Agent (EMA) is the library that any process must link with to access EMSv functionality. Events published by invoking the publish API in the Event Management Service are forwarded to EMS by EMA. The proposed Event Management System provides the event persistency even after the system crash. This implementation preserves the events on another computer in the distributed system across the system failures.

The advantages of Event Management System:

•Event manager matches the "patterns" accompanying the published events with filters stalled on event channels by the subscribers before delivering the events to them. This feature makes sure that the asynchronous event communication possible.

•Publishers can also be subscribers on the same event channel. Subscribers are anonymous, so they can join and leave an event channel at any time without involving publisher.

•The Event Manager enables highly data intensive distributed computing applications such as telecommunications, enterprise systems and database.

Users of the Event Service library run on cluster nodes as:

Producer:

A Event producer generates Events (using the Producer API of the Event Service).

Consumer:

A consumer consumes Events that were generated by producers. If a consumer is not interested in all Events, it can specify filter criteria. A consumer can also be a producer. A consumer can be one of the types described in the following subsections or both:

Subscriber:

A subscriber for Events gets Events forwarded as they occur (push interface).

Reader:

A reader retrieves historical Event entries from the persistent Event log (pull interface).

Event Service:

Similar to the AIS Services, the Event Service mainly consists of a client library and a server. No assumptions are made as to how server instances are distributed across the nodes of an SA Forum cluster. In an implementation, the server could even be part of the library.

Event Service Library:

The Event Service library provides the following public C APIs:

- Producer API
- Subscriber API
- Reader API

Event Server:

The Event server applies the filtering criteria on Events for delivery to subscribed consumers and performs the logging into persistent storage. EMS runs on System Manager Nodes. One active and one standby instance of EMS exist in a distributed computing cluster.

ISSN No: 2348-4845

International Journal & Magazine of Engineering, Technology, Management and Research

A Peer Reviewed Open Access International Journal

Block Diagram:

Applications
Producer and consumer APIs
Event Service Library Drivers
Transport Service
Event Management Server
Operating System

Application Area:

- High performance system clusters and distributed computing systems.

- The distributed systems where application events loss cannot be tolerated.

- Distributed transaction systems

- Distributed Enterprise application management systems

Conclusion:

The Event Management system would be part of system middle ware and provides comprehensive event communication functionality of distributed computing environment. This is most suited form of event communications where system downtime is not acceptable thus provides redundancy mechanism to achieve the event delivery and production.

References:

[1] Distributed Systems Concepts and Design by George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair

[2] Wiley Series inCommunications Networking & Distributed Systems

[3] http://en.wikipedia.org/