



**TTM**

Total Transaction Management

**AMon™**

## Enterprise-Class Monitoring System for Apache ActiveMQ

### BENEFITS

**Monitoring:** Early detection and resolution of problems increases overall system efficiency and reduces downtime.

**Customizable Solution:** Quickly develop monitoring agents (Monlets) for the ActiveMQ message broker. The user can develop customized alerts without changing the core ActiveMQ product. Users are not limited by the default monitoring tools currently provided in ActiveMQ.

**Ease of Development and Deployment:** Users can easily develop and deploy their Monlets to meet specific requirements.

**Metrics and Performance Data:** Monlets can be customized to provide performance data that can be used by operators for observation of system performance.

**Audit:** Monlets can provide historical data of system performance, which is useful for planning deployment of additional hardware and software.

**Logging:** AMon provides the ability to easily transform selected broker log messages into SNMP notifications.

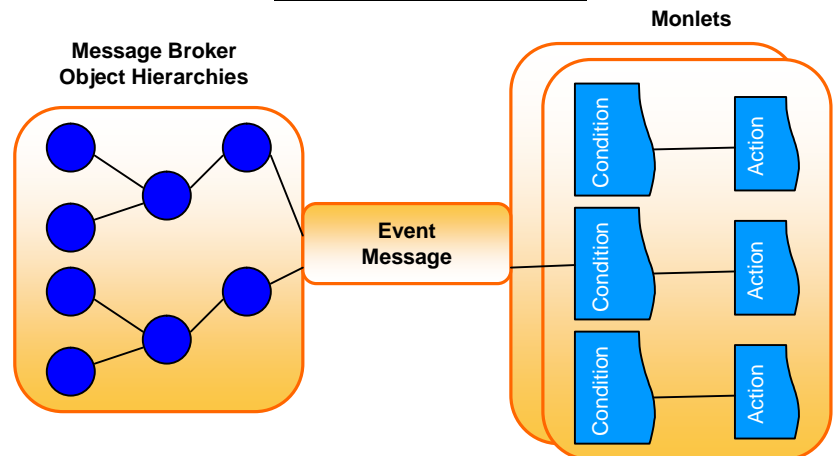
**Trigger Mechanism:** Monlets can be designed to automatically initiate scripts and send alerts to resolve performance or resource issues. This feature helps reduce downtime and increases efficiency of the ActiveMQ system.

**SNMP Support:** You can monitor ActiveMQ via the SNMP protocol.

[Apache ActiveMQ](#) is the most popular and capable Open Source message-oriented middleware (MOM) system. All enterprise-class MOM systems, including ActiveMQ, need enterprise-class monitoring capabilities to ensure a high level of efficiency, reliability, availability and performance for the system as a whole.

AMon™ provides a highly configurable and extensible enterprise-class monitoring system for Apache ActiveMQ with the **Monlet Container, Monlets and SNMP Agent**.

### AMon Framework



### About TTM

Total Transaction Management (TTM) specializes in providing quality software and professional services for some of the largest corporations in the world since 1995.

Since its entry into Open Source software by releasing the first ActiveMQ User Guide, TTM has applied its knowledge and expertise in mission-critical messaging middleware to Apache ActiveMQ including:

- AMon Monitoring System for Apache ActiveMQ
- LDAP and JDBC Security Plugins for Apache ActiveMQ
- ActiveMQ Architecture and Implementation Services including prototyping, troubleshooting, proof-of-concept, custom development, and testing.
- ActiveMQ Reference Guide
- Foundation for ActiveMQ

[www.ttmsolutions.com](http://www.ttmsolutions.com)

**Get the message with TTM**

	FEATURES	BENEFIT
<b>MONLET CONTAINER</b>	The Monlet Container is embedded within an ActiveMQ message broker process and hosts one or more Monlets (monitoring agents).	The container provides the mechanisms to quickly develop and deploy programmable Monlets for ActiveMQ.
	Monlet Container is an event-driven mechanism, which Monlets subscribe to, that provides real-time access to the message broker's run-time object hierarchy (i.e., the broker's operational runtime state) and the messages that flow through the message broker.	Monitor 35 event objects representing a wide variety of broker processing and timer events that occur within the broker and Monlet container, respectively.
	A Monitoring Expression Language (MEL), which is based on the Unified Expression Language (UEL).	Develop conditional expressions or statements (predicate functions) that test for events and the message broker's runtime state using MEL.
	Apache Camel provides access to a myriad of integration endpoints that can be leveraged for the implementation of 'actions' (e.g., sending alerts, activating scripts, writing to a database, etc) that are invoked in response to a particular event or runtime condition.	Integration with Apache Camel and its Java Domain Specific Language (DSL), which together with the MEL, provides for the quick development of Monlets.
	Many aspects of the Monlet container and the Monlets can be dynamically configured.	This allows you to make changes to the operational characteristics of the container and Monlets without having to stop and restart the ActiveMQ message broker.
<b>MONLET</b>	A Monlet is primarily designed to monitor events and/or conditions that occur within the ActiveMQ message broker.	A Monlet is also designed to collect data, analyze data, and send notifications (alerts) whenever a particular event or condition occurs.
	For example, a Resource Monlet tests whether a particular object representing a resource (e.g., memory, queue, etc.), has breached a threshold and then invokes an action if the threshold is breached.	The action(s) can be invoking a script, firing off an email, or raising an SNMP notification. The AMon container provides access to those resources pertaining to the JVM (memory, threads, etc.), as well as resources (queues, topics, connections, messages,
	For example, a Logging Monlet captures log4j messages that are being written by the message broker to its system log file with a severity of WARN or higher.	The Monlet can then express the log message through a SNMP notification or trap.
<b>SNMP MIB</b>	AMon's SNMP v2 Agent and its corresponding Management Information Base (MIB).	Network management applications (e.g., IBM Tivoli, HP OpenView, Nagios, etc.) now have the ability to monitor the ActiveMQ message broker via the SNMP protocol.
	AMon's SNMP v2 Agent also includes a Camel SNMP integration endpoint.	Using Camel, Monlets have the ability to send SNMP notifications (a.k.a., traps) to the network management applications. The AMon SNMP Agent also generates its own traps whenever the message broker starts and stops.

**Evaluation and purchase info:** You can download AMon for evaluation or purchase annual subscription for your enterprise at [www.ttmsolutions.com/Transactional\\_Software\\_Solutions/Active\\_Monitor\\_AMon.php](http://www.ttmsolutions.com/Transactional_Software_Solutions/Active_Monitor_AMon.php)  
For AMon implementation and Monlet development services, please contact TTM at 760-591-0273 or email [info@ttmsolutions.com](mailto:info@ttmsolutions.com).

<b>SYSTEM REQUIREMENTS:</b>	
OPERATING SYSTEM:	Linux, Windows, Unix
MIDDLEWARE:	Apache ActiveMQ 5.2
JAVA:	Java 5 or 6

	Entry Level	Standard Level	Premium Level
Number of CPUs	Up to 2 CPUs	Up to 32 CPUs	Unlimited
Request by	Email only	Phone or email	Phone or email
Phone availability	N/A	Business hours (8 to 8 ET)	24 x 7
Number of incidents included	2	10	20
Number of customer contacts	1	5	10

Copyright © 2009 Total Transaction Management, LLC. All rights reserved.  
AMon™, Total Transaction Management™, and all related product and service names, design marks and slogans are trademarks or registered trademarks of Total Transaction Management, LLC. All other product and service marks contained herein are the trademarks of their respective owners.



**Total Transaction Management**

570 Rancheros Dr., Suite 140  
San Marcos, CA 92069  
Tel (760) 591-0273  
Fax (760) 591-9960

Email [info@ttmsolutions.com](mailto:info@ttmsolutions.com)  
Website [www.ttmsolutions.com](http://www.ttmsolutions.com)