

eG WebLogic Monitor

Multi-Tier WebLogic Environments

To achieve scalability and performance, most Internet application deployments have evolved into multi-tier infrastructures where the web server tier serves as the web front-end, the business logic is executed on middleware application servers, and the backend storage and access is provided via database servers. While multi-tier infrastructures offer a variety of scalability and extensibility benefits, they are also more difficult to operate and manage. When a problem occurs (e.g., a slowdown), an administrator often has difficulty in figuring out which application(s) in the multi-tier infrastructure could be the cause of the problem - i.e., is it the network? or the database? or the middleware? or the web server? Comprehensive, routine monitoring of every infrastructure application and network device is essential to be able to troubleshoot effectively when problems occur.

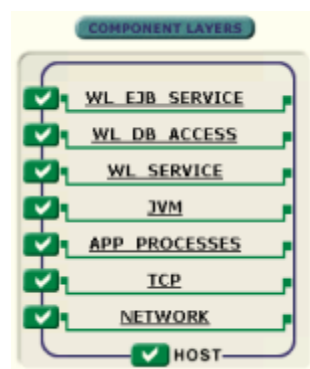
The application server middleware that hosts and supports the business logic components is often the most complex of the multi-tier infrastructure. To offer peak performance, an application server provides a host of complex functions and features including database connection pooling, thread pooling, database result caching, session management, bean caching and management etc. To ensure that the application server is functioning effectively at all times, all of these functions have to be monitored and tracked proactively and constantly.

The eG WebLogic Monitor

The eG WebLogic Monitor offers comprehensive monitoring for the BEA WebLogic application server. Over a hundred critical metrics relating to a WebLogic server instance can be monitored in real-time and alerts can be generated based on user-defined thresholds or auto-computed baselines. Through these metrics, administrators can track the functioning of the WebLogic server's database connection pools, thread pools, session management, bean container, cluster manager, etc. eG's customized layer model for WebLogic allows correlation of the WebLogic application server performance with other network and operating system metrics.

Benefits of the eG WebLogic Monitor

- In-depth real-time analysis of a WebLogic server's performance including analysis of various sub-components such as the Virtual Machine (JVM), Database Access (JDBC), Java Beans (EJB), Connectors, Clusters, etc
- Automatic baselining of WebLogic performance and intelligent time-based alerting when any unusual high activity is detected
- Automatic Correlation of WebLogic's performance with web server and database performance to pin-point the root-cause of problems
- A 100% web-based monitoring solution that allows anytime, anywhere monitoring and control of a WebLogic infrastructure



eG's WebLogic model



Topology of a multi-tier WebLogic architecture

In conjunction with eG's web server and database server models and its built-in single-click root-cause diagnosis capability, the eG WebLogic monitor allows administrators to quickly pin-point which of the infrastructure applications could be the cause of a bottleneck in a multi-tier infrastructure.



Versions Supported

Support for WebLogic Version 5.1 onwards is available as part of the eG suite with SNMP and the Java Management Extensions (JMX), being used as mechanisms through which the eG agent interfaces with a WebLogic server instance. A single eG agent is capable of monitoring all of the WebLogic application server instances executed on a system. The eG Reporter includes pre-canned reports highlighting the availability, performance and usage of a WebLogic server instance.

What the eG WebLogic Monitor Reveals

Server monitoring	<ul style="list-style-type: none"> Is the WebLogic process running? Is the memory usage of the server increasing over time? Is the server's request processing rate unusually high?
JVM monitoring	<ul style="list-style-type: none"> Is the JVM heap size adequate? Is the garbage collection tuned well or is the JVM spending too much time in garbage collection?
Thread monitoring	<ul style="list-style-type: none"> Are the server's execute queues adequately sized? Are there too many threads waiting to be serviced, thereby causing slow response time?
Security monitoring	<ul style="list-style-type: none"> How many invalid login attempts have been made? Are these attempts recurring?
JMS monitoring	<ul style="list-style-type: none"> Are there many pending messages in the messaging server? Is the message traffic unusually high?
Connector monitoring	<ul style="list-style-type: none"> What is the usage pattern of connections in a connector pool?
Cluster monitoring	<ul style="list-style-type: none"> Are all the servers in the cluster currently available? Is the load being balanced across the cluster?
Transaction monitoring	<ul style="list-style-type: none"> How many user transactions are happening? Are there too many rollbacks occurring?
Servlet monitoring	<ul style="list-style-type: none"> Which servlet(s) are being extensively accessed? What is the average invocation time for each servlet?
EJB Pool monitoring	<ul style="list-style-type: none"> Are there adequate numbers of beans in a bean pool? How many beans are in use? Are there any clients waiting for a bean?
EJB Cache monitoring	<ul style="list-style-type: none"> Is the cache adequately sized or are there too many cache misses? What is the rate of EJB activations and passivations?
EJB Lock monitoring	<ul style="list-style-type: none"> Is there contention for locks? How many beans are locked? How many attempts have been made to acquire a lock for each bean?
JDBC Connection monitoring	<ul style="list-style-type: none"> Are all the JDBC pools available? Is each pool adequately sized? What are the peak usage times and values? How many connection leaks have occurred?
JDBC call monitoring	<ul style="list-style-type: none"> How many JDBC calls have been made? What was average response time of those calls? What are the queries that take a long time to execute?

For more information
info@eginnovations.com
www.eginnovations.com

