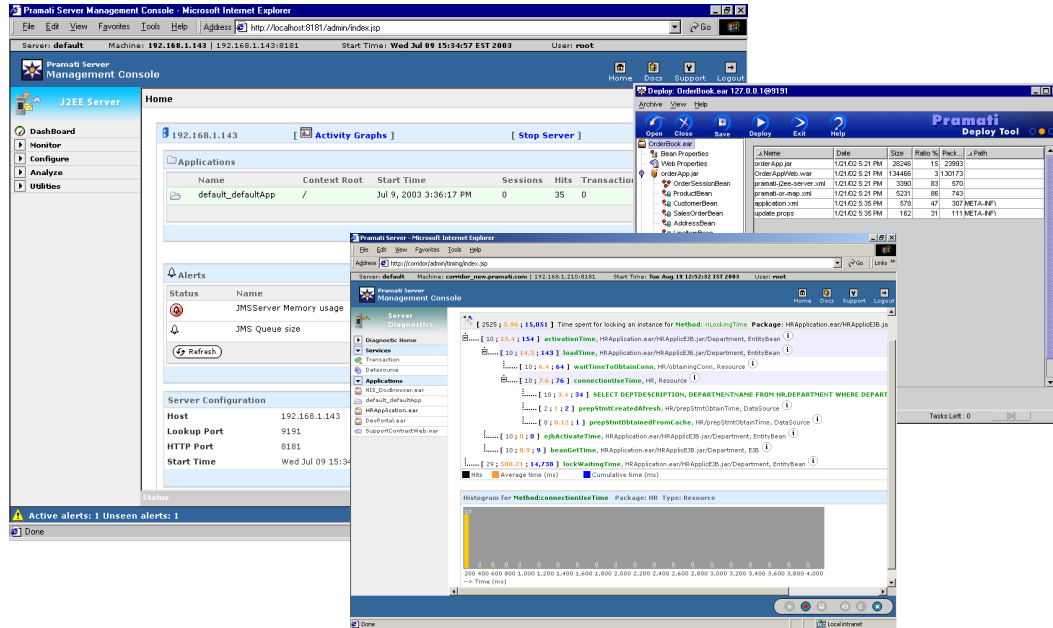




Pramati Server 4.1



Pramati Server 4.1 is a standards-based application server for deploying Enterprise Java applications. The platform is built to Java™ 2 Enterprise Edition (J2EE™) 1.3 specifications, and can directly run applications written for servers such as Tomcat and WebLogic™.



Key Features

- Dynamic Content Caching in Web Server with declarative controls and Web-based caching rules manager.
- Point-n-Run Server Framework for running Tomcat and other web applications, requiring no change in application code.
- Self-organizing and transparent Pramati Clustering Solution. Convert any Pramati Server node into a Pramati Cluster node.
- Web Load Balancer with Request Dispatcher that can redirect requests to any server (such as HTTP and PHP) based on session, URI, workload, node health and other parameters.
- Tunable concurrency controls for EJB deployments. Choose optimistic or pessimistic controls per entity bean type.
- In-built JMS module configurable for embedded J2EE use, pure JMS standalone, and JMS Cluster scenarios.
- GUI-based Deploy Tool for resolving and deploying JSP and EJB applications on local and remote Servers. Automatic generation and validation of all deployment descriptor XMLs.
- Enhanced command line operations with facility to write custom command sets for easily managing an application.
- Customizable server footprint for ISVs who can switch on or off platform services directly in configuration XMLs, depending on the application deployment scenario. For example, switching off the EJB service in a pure Web application scenario.
- Application Validator ensures adherence to standards.
- Web-based Diagnostics Tool collects fine-grained performance statistics from runtime Server environment. Perform execution path analysis for optimizing application performance.
- ANT handlers for application deployment and administration.
- Prime real-time visualization of Server activity available on Server Dashboard. Keep critical statistics and trends on the Dashboard by configuring display XML.
- JMX-based Server Management Console with published mbeans.
- Configurable Alerts Framework, and extensive Server and web activity logs for troubleshooting and monitoring.
- **Standards Compliance:** EJB™ 2.0, JSP™ 1.2, Servlets 2.3, JMS™ 1.0.2, JTA™ 1.0, JDBC 2.0 SE, JNDI™ 1.2, JavaMail™ 1.1, JAXP 1.1, JAAS™ 1.0, JMX 1.0, JCA™ 1.0

System Requirements

Any platform that supports JDK 1.3.1_01. Min 128 MB RAM (256 MB recommended). Min 50 MB hard disk space (application size not included).

Platform	JDK version
WinNT 4.0, SP5	JDK 1.3.1_01, 1.3.1_02 and 1.4
WinNT 2000	JDK 1.3.1_01, 1.3.1_02
Linux RedHat 6.2	JDK 1.3.1_01, 1.3.1_02
Solaris 7, 8	JDK 1.3.1_01, 1.3.1_02

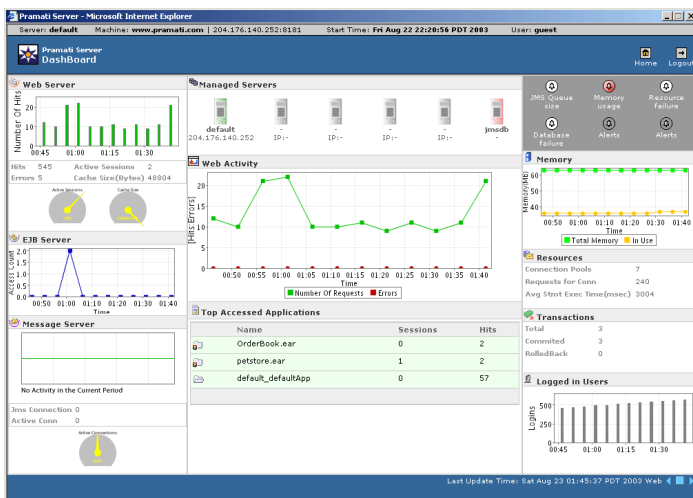
JDBC drivers for other databases may be provided by respective vendors.

Database	JDBC Driver
Oracle 8i	Thin driver V 8.1.5.0.0, MERANT DataDirect Connect JDBC V2.2
Oracle 9i	Oracle thin driver 9.0.1.1.0
Informix 7.3.0 TC3	Driver V 2.10 JCIN361
Cloudscape 4.0	Cloudscape 4.0 RMI JDBC Driver
MS SQL Server 7	MERANT DataDirect Connect JDBC V2.2, JTurbo Driver 2.3

Server Configuration and Management

Web-based Server Management Console

- Local and remote management of a network of application servers through web browser.
- Consolidated view of all the servers and deployed applications on the network.
- Deploy, upgrade and undeploy applications over the web.
- Built using Java Management Extension (JMX) standards.
- Customizable and extensible to suit specific management requirements through published mbeans. All manageable components exposed as mbeans and manageable from Console.
- Plug-in additional custom mbean components through API.



Server Management Dashboard

- Web-based utility that gives administrator a real-time snapshot of all server and application activity, local and on the network.
- Keep critical statistics and trend plots on Dashboard for timely monitoring and action.
- XML-based display control.
- Choose statistics and trends to view by modifying XML.

Alert mechanism

Define events that can trigger an alert and necessary action. For example, set up an alert when an exceptionally high load or DB connection failure is detected.

Write custom platform "services"

- Framework for incrementally adding custom services.
- Consolidated command shells with extensible command set.
- API for creating new server footprints.

XML-based Server configuration

- Server and Cluster nodes fully configurable from XMLs. On-demand sync-up of Cluster re-configuration using XMLs.
- Separate configuration XMLs for EJB, Web, JMS and other platform services, including clustering.
- Configure a lighter platform footprint by switching off individual services, depending on application requirement.

Command line operations

- J2EE Server Shell, Resource Shell and JMS Shells, with extensible command sets.
- Extended commands for application management.
- New flexible framework for Shells and Commands, customizable by ISVs configuring the application server for an application.
- Write custom command sets for application-specific handling.

Server maintenance

- Hot manageability, with no Server restarts required.
- Extended management of Server instances.
- Add, configure new instances (virtual hosting).
- Global setting for page encoding for multi-lingual page displays.
- Configurable ports for Pramati Server Administration Service for easy configuration of Server nodes on an existing set-up.
- Shutdown Immediate facility for forcing Server shutdowns.
- Server Launcher and Application Startup Hooks to program application-specific processing during Server start-up.

Server monitoring

- Improved statistics on EJB and Web containers, virtual hosts, platform services and deployed applications.
- Low-overhead activity history collection framework.
- History plots for monitoring memory usage and traffic patterns.
- Improved Diagnostics Tool for execution path analysis and debottlenecking application performance.

Server logging

- Reporting Framework for messages, events, debug traces.
- Unique per-incident message tracing (even under heavy stress load conditions on the Server).
- Execution trace and message trace (stack).
- Augmented error messages in the Server log.

Robust resource connection management

- High availability datasource, with auto failover from primary to secondary source.
- Connection shrinking time interval.
- Multiple pools.
- Resources defined against multiple JDBC driver versions.
- Enhanced connection pool validation with default and custom capacity increment.
- Connection refresh time (enabling auto refresh of connections).

Interoperability over IIOP

Interoperate applications deployed on Pramati Server with other J2EE and CORBA applications over RMI/IIOP. Naming, security and transaction propagated across deployed applications.

EJB Applications Platform

- Deploy EJB 1.1 beans with EJB 2.0 beans in EJB 2.0 or 1.1 JARs.
- All enterprise bean types supported.
- Deploy Message Driven Beans.
- Deploy entity beans with container managed persistence (2.0).

- Define container managed relationships among entity beans.
- EJB QL support for vendor neutral queries.
- Local, home, local home and remote interface support.

Container managed relationships

- Uni and bi-directional relationships.
- All cardinalities between relationships.
- Auto-resolution of relationships during deployment.

Configurable optimistic concurrency

Concurrency control over a bean instance can be set to optimistic or pessimistic mode during deployment. Given the nature of an application, deployers can fine tune performance by choosing concurrency type for each entity bean in the application.

Message Driven Beans

- Deploy MDBs on EJB 2.0 Container and run with any JMS server.
- Initiate and complete transactions using MDBs.
- Modify databases triggered by these beans.

JCA 1.0 Connectors

- RARs deployable in EJB container like standard J2EE component.
- Performance optimized through pooling of connector resources.

Web Applications Platform

- Enhanced Web Container throughput using extensive object pooling and Dynamic Content Caching.
- Enhanced HTTP Server performance through static-file caching.
- Multiple deployment options, with Point-n-Run Framework to run web applications off third-party Server directories.
- Hot deployment of updated web components.
- Optimizable session maintenance with an advanced session storage and retrieval mechanism.
- Session tracking using URL rewriting and cookies.
- Enterprise level security and SSL support with client authentication capability.
- Pluggable JSSE providers.
- DNS-based Virtual Hosting architecture.
- Direct pick-up from under doc root for unit deployment of a JSP, Servlet or HTML page on Web server.

Web load balancing

- Intelligent HTTP-aware load balancing and traffic management of multiple network protocols.
- Fault tolerance and automatic failover of front-end and back-end machines.
- Increased system capacity, throughput and reduced latency.
- Separately installable and configurable Server component.
- Rule-based request dispatching capabilities. For example, all dynamic content served from one node.
- Web-based and command line management of LB.
- Statistics for dynamically tuning LB performance.
- Failsafety through a configured "warm" Load Balancer node. Automatic re-takeover when original node comes back..

Dynamic Content Caching

- Cache frequently used JSP-driven content.
- Implement caching strategies as caching rules that can be based on nature of application usage.
- Event and time-based invalidation or expiry of cache.
- Set up second URL as trigger to invalidate and refresh cache.

Virtual hosting

- Enhanced support for virtual hosts.
- Hierarchical configuration.
- Host-specific logging and caching.
- Agent-based processing.

Protection against denial-of-service (DoS) attacks

Pramati Web Server provides the following shields against Denial-of-Service attacks:

- Request accept rates.
- Header size.
- Request body size.
- Request or IP block.

Third-party web server support

- Plugins available for configuring Apache HTTP Server, Microsoft® IIS and Oracle® Forms Server to work with Pramati Web Container.
- Support for Apache HTTP Server versions 1.3.2 and 2.0, and Microsoft IIS (version 5.0). No manual configuration required.
- High performance connection pooling and Dynamic Content Caching extended to plug-in.

In-built Pramati Message Server

- Distributed transactional messaging through XA implementation Guaranteed once and only once message delivery.
- Persistent messaging ensures message retrieval using durable subscribers in case of server breakdown.
- Web-based management of administered objects.
- Architecture minimizes server load and optimizes performance.
- Distributed messaging through firewalls.

Application Deployment

- GUI-based Deploy Tool manages complex O-R mappings, and auto-resolves all possible container managed relationships.
- Seamless deployment of applications on a cluster of J2EE Servers.
- One-time code generation and storage in Prepared Archive (PAR) files, for easy distribution of deploy-ready applications.
- A quick and simple mechanism for deployment on remote Servers by mapping the application to the target environment.
- Support for PHP, Perl, CGI based applications.

Drag-n-Drop deployment handlers for other servers

- Drag-n-drop applications into a hot deployment directory.
- Autodeploying and staging facility.
- No redeployment, migration or change in directory structure. No change in application files.
- API available for writing custom deployment handlers.

Point-n-Run Tomcat applications with no code change

- Any application running on Tomcat Server as well as Apache HTTP Server can be run from a Pramati Server installation.
- Migrate server configurations and mimic source environment on Pramati Server.

Application versioning and replication

- Application versioning that allows rolling back redeployed application with offline client JAR preparation.
- Distribute and 'replicate' applications from one instance to another.

Deploy Tool

- In-built with Pramati Server for developers preferring a GUI-based deployment over hand-coding the various XML deployment descriptors.
- Prepares archives for deployment by resolving O-R mappings, security and application resources (JMS, Mail, XA, Connectors), and environment settings.
- Validates enterprise archives for conformance to J2EE 1.3.
- In-built Task Manager monitors all 'unfinished tasks', and guides deployer through sequence of unfinished tasks. This reduces iterations through testing and deployment.

Application Validator

- Parses XML and archives to ensure they are in compliance with the J2EE specification
- Points out if there are any inconsistencies. This ensures strict and efficient adherence to the standards.
- Uses JAXP standard.

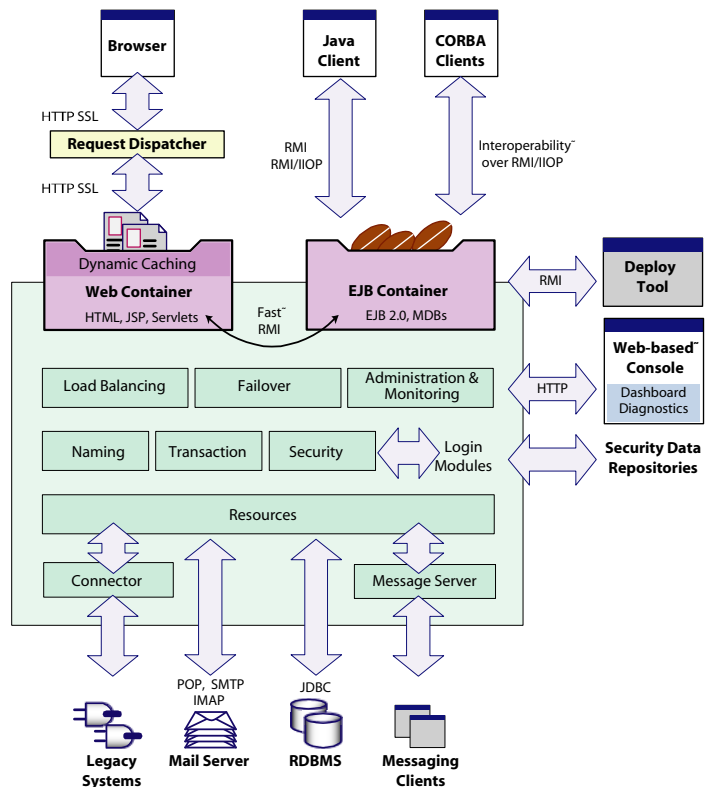
JAAS-based Security Framework

- Integrated with EJB and Web Containers and Message Server.
- Declarative access control lists for naming, resource, administration, security and deployment permissions.
- Security using run-as elements with role propagation.
- Digital certificate, user and user group administration.
- Certificate-based login modules, with out-of-the-box implementations for LDAP, DB, and XML.
- Implicit security context propagation from EJB to Web Container as well as among EJB nodes in Cluster.
- RMI/SSL and HTTPS.
- Optional time-out configuration of RMI/SSL sockets.
- Pluggable provider for JSSE implementation.
- Mutual authentication between client and server.

Pramati Clustering Solution

- Peer-to-peer model eliminates master-slave relationships
- No single point of failure.
- Transparent and does not require any change to application code before redeploying for clustering.
- Can convert existing a standalone Server into a Cluster node.
- Form Cluster from EJB, Web, Load Balancer and Full J2EE Server nodes.
- Web-based Cluster configuration and management. File-based configuration and state persistence, database not mandatory.

- Load balancing and failover for Web and EJB applications.
- Transparent failover extended to all bean types.
- Seamless replication of bean states.
- Dynamic allocation of system resources as scaling requirements vary with workload.
- Inter-node communications channel with very high reliability. Broadcast mechanism makes new nodes immediately visible to other nodes.
- Pluggable load balancing algorithms. Ships with default weighted and round-robin load balancers.



Customizable Pramati Services Framework

- Platform footprint composed from modular stack of dependent and independent platform services, driven by configuration XML.
- Reconfigurable binary even after platform installation.
- Non-programmatic way of defining services to start with Server start-up. Also sequence the services to start up.

Contact

For further information, visit <http://www.pramati.com>
See Server in Action at <http://demo.pramati.com>