RED HAT ENTERPRISE MRG: MESSAGING, REALTIME, AND GRID

BRYAN CHE

HIGH-PERFORMANCE DISTRIBUTED COMPUTING

Red Hat® Enterprise MRG is a new platform that dramatically increases the value of an enterprise's Information Technology (IT) infrastructure by reducing complexity and cost while increasing performance, reliability, interoperability, and flexibility. By integrating Messaging, Realtime, and Grid functionality, Red Hat Enterprise MRG provides a revolutionary foundation for high-performance distributed computing for everything from service-oriented architecture (SOA) to virtualization to cloud computing to bare-metal mission-critical applications.

Red Hat Enterprise MRG is a next-generation IT infrastructure that solves significant challenges for today's IT decision-makers. Here are a few examples:

Most—if not all—workloads and environments can benefit from virtualization. But how do you leverage virtualization if:

- You have thousands or tens of thousands of heterogeneous servers running different applications?
- You have high variability or seasonality in your workloads?
- You want to utilize additional resources that aren't in your datacenter?
- You want to treat your datacenter as a computing utility or "internal cloud"?

Distributed applications and architectures require messaging software to distribute data and connect services across a wide variety of computers. But how do you leverage messaging if:

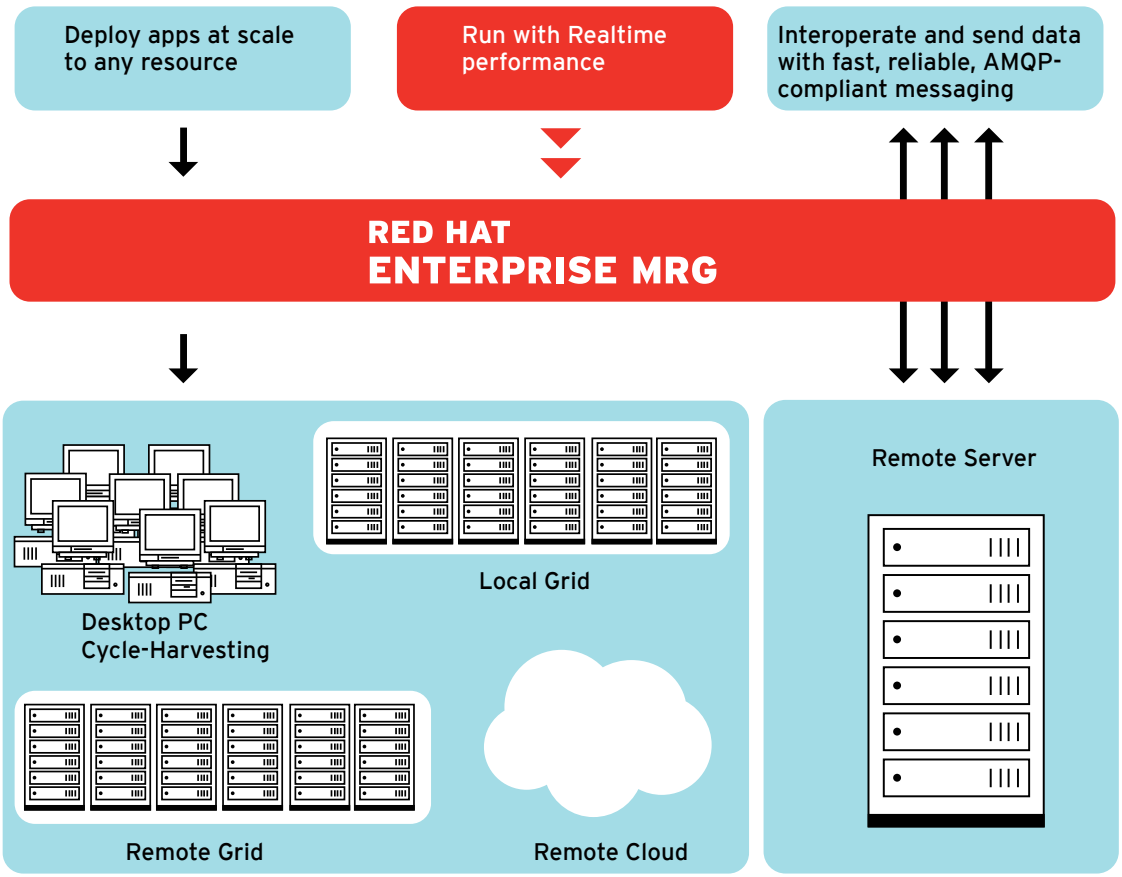
- Messaging software is segmented, complex, and incompatible?
- Messaging software is expensive and proprietary?
- Messaging software does not meet your functional or performance requirements?

Computing is increasingly critical and often provides competitive advantage. How do you guarantee your software will perform reliably and meet Service Level Agreements (SLAs)?

Red Hat Enterprise MRG solves these issues and others by helping enterprises:

- Move to a new model of efficient and scalable utility computing. Red Hat Enterprise MRG can schedule any workload—from sub-second calculations to server applications to high-performance computing workloads—to any computing resource. Red Hat Enterprise MRG allows your business to span local grids and remote grids, virtual machines, cycle-harvest desktop workstations, and dynamically provision and schedule cloud-based infrastructure like Amazon EC2. This moves enterprises toward a utility model of computing where they can schedule all their applications as jobs across a heterogeneous set of resources. The MRG grid scheduler can then automatically optimize utilization, priorities, load, and other key factors.

HIGH-PERFORMANCE DISTRIBUTED COMPUTING



- Implement large-scale virtualization with support for virtualization integrated into MRG. Includes sophisticated capabilities for managing virtualization deployments across heterogeneous servers with changing workloads, priorities, and seasonality.
- Distribute data via AMQP-compliant messaging bus with performance up to two orders of magnitude better than other solutions and deterministic low latency.
- Provide deterministic performance—especially for messaging and networking workloads in distributed environments—and meet stringent SLAs with a realtime kernel.

- Submit, manage, and monitor various distributed workloads with a single, unified interface.
- Simplify software architecture and deployment stacks by relying on one integrated platform rather than numerous, specialized, incompatible point products.

These capabilities offer tremendous core value for the enterprise. MRG, for example, allows industries the following specific gains:

- Financial institutions gain competitive advantage by building a more deterministic, lower-latency, reliable trading platform with consistent response times. Other resources, ranging from dedicated grids to dynamically-provisioned cloud-based resources for running Monte Carlo analysis of market conditions, provide similar gains.
- Government and the defense industry can create more reliable simulations or control systems and process and disseminate critical information faster.
- Telecommunications companies can provide increased reliability, performance, and interoperability for their networks and infrastructure.
- Animation studios can render movies more efficiently, completing them sooner with improved workflows.
- Oil and gas companies can save precious drilling time by performing seismic analysis with increased speed and efficiency.
- Internet companies can provide better performance and interoperability.

In short, Red Hat Enterprise MRG can provide any enterprise that has distributed data or computational resources with a competitive advantage. Enterprise MRG gives businesses the flexibility and performance to address peak computational demands through the three fundamental technologies of Messaging, Realtime, and Grid.

RED HAT ENTERPRISE MRG COMPONENTS

Red Hat Enterprise MRG integrates three technologies: Messaging, Realtime, and Grid. Even though the three technologies provide unique value in combination, they also offer significant individual advantages in their respective domains.

MRG MESSAGING

Messaging is the backbone of enterprise and high-performance computing, SOA deployments, and platform services. However, messaging offerings usually have restricted capabilities, forcing users to choose between high-speed messaging systems for deployments like market-data applications, reliable messaging systems for scenarios where guaranteed delivery is crucial, and other solutions for messaging of large, multi-gigabyte payloads. Furthermore, these different messaging systems are fundamentally incompatible with each other—meaning that enterprises that have broad messaging requirements often must purchase multiple, incompatible messaging products and deploy them in silos—increasing complexity and cost.

Red Hat Enterprise MRG Messaging brings tremendous value to enterprise messaging by providing the scalability, reliability, and performance to handle all messaging use cases. Red Hat Enterprise MRG provides Messaging functionality that is up to 100 times faster than other solutions—with reliability, durability, and support for multi-gigabyte messages. Furthermore, Red Hat Enterprise MRG implements Advanced Message Queuing Protocol (AMQP), the industry's first open messaging standard. AMQP offers unprecedented interoperability: cross-language, cross-platform, multi-vendor, spanning hardware and software, extending to the wire level. Red Hat is a founding member of the AMQP working group that is developing the standard.

MRG Messaging uses Linux-specific optimizations to achieve peak performance on Red Hat Enterprise Linux® with MRG Realtime. It can also be deployed on non-Linux platforms like Microsoft® Windows® and Sun® Solaris,® though without realizing the full performance and quality of service benefits that Red Hat Enterprise Linux provides. MRG Messaging also supports most major development languages.

MRG Messaging includes the following features:

- **AMQP support.** MRG messaging implements AMQP, an open and interoperable messaging standard developed by the AMQP Working Group. Red Hat is one of the founding members of the AMQP Working Group and Red Hat Enterprise MRG is compliant with AMQP 0-10, the most current version of the standard.
- **Flexible messaging paradigms.** MRG Messaging includes comprehensive support for various messaging paradigms, including store-and-forward, transaction distribution, publish-subscribe, content-based routing, queued file transfer, point-to-point connections among peers, and market data distribution.
- **Multi-language client support.** MRG Messaging offers clients the ability to use a variety of languages across several platforms, including Java (JMS), C++, Python, .NET (WCF), and more. MRG Messaging scripting clients allow rapid application development with a light-weight tool, while still offering all the performance and transactional power of AMQP.

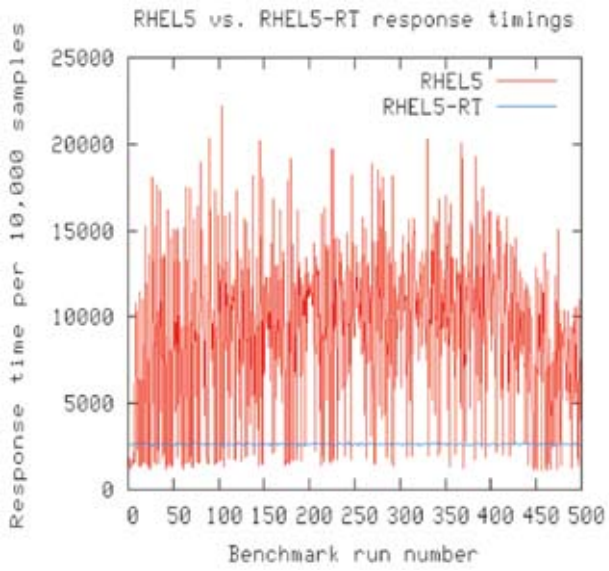
- **High performance.** MRG Messaging can achieve over 6 million OPRA messages per second throughput on an 8-core box using gigabit ethernet. For more information, see the whitepaper.
- **AIO durable messaging journal.** MRG Messaging can achieve sustained durable messaging throughput rates of 500,000 messages per second per LUN on typical hardware, a rate that is orders of magnitude better than previously possible. MRG Messaging achieves these rates through the inclusion of a new AIO journal specifically optimized for persistent messaging storage on Red Hat Enterprise Linux.
- **Native RDMA infiniband support.** MRG Messaging offers native RDMA infiniband support. This enables MRG messaging to achieve extremely low end-to-end latencies (in the microsecond range) for reliable messaging.
- **MRG Realtime optimizations.** MRG Messaging is optimized to run on the MRG Realtime kernel to provide optimal, deterministic latencies and outstanding throughput.
- **Transient and durable messaging.** Durable messaging ensures that messages and queues are restored in the unlikely event of a broker crash or an operating system crash. Transient messaging reduces the processing needed for messages and guarantees delivery as long as the broker continues to run.
- **Large message support.** MRG Messaging supports multi-gigabyte messages.
- **Clustering and failover.** MRG Messaging provides clustering and failover to ensure that your applications continue to function if a broker or operating system should crash.
- **Federation.** MRG Messaging provides broker federation to enable deployments across multiple geographies.
- **Transactions.** MRG Messaging provides support for messaging transactions, including distributed transaction (XA) support. With transactions, you can ensure that all messages in a group are delivered as a whole—if the delivery of one message fails, then the delivery of all messages in that group fails.
- **Security.** MRG Messaging includes SSL support, role-based access control, SASL authentication, and other security features.
- **Queue semantics.** MRG includes support for queue semantics like ring queues, last value queues, and replay queues.
- **XML.** For XML messages, MRG supports dynamic routing based on XQuery expressions in the messaging broker as well as in-flight message transformation.
- **Distributed management console.** MRG Messaging contains a console for instrumentation (including historical), configuration, and the ability to perform management operations through a web interface to control a network of deployed machines.

MRG REALTIME

For the most time-critical workloads, "close enough" isn't good enough. For instance, in the highly competitive financial services marketplace, milliseconds may make the difference between a trade and a lost customer. In these environments, consistently fast response times are a huge competitive advantage.

MRG Realtime provides the highest levels of predictability for consistent low-latency response times, giving an enterprise the ability to meet the needs of time-sensitive workloads. For most customers, a properly tuned standard Red Hat Enterprise Linux system provides acceptable levels of determinism. For applications with extremely demanding latency requirements, MRG Realtime provides new levels of determinism. It optimizes lengthy kernel codepaths to ensure that they do not become bottlenecks and allows for better prioritization of applications. This results in consistent, predictable response times for high priority applications.

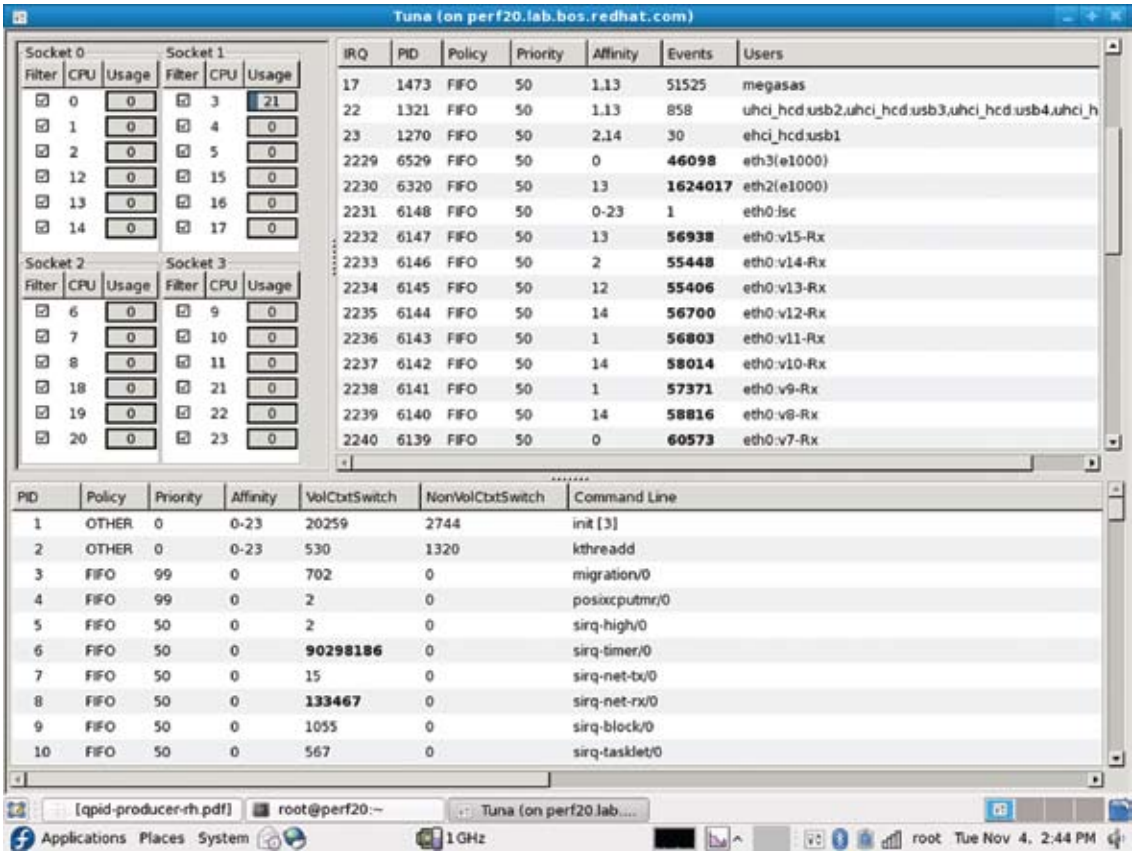
DETERMINISTIC PERFORMANCE



MRG Realtime is a set of Red Hat Package Manager (RPM) packages that are add-ons or replacements for standard Red Hat Enterprise Linux 5 components. MRG Realtime consists of:

- a realtime kernel, which replaces the standard Red Hat Enterprise Linux 5 kernel,
- a set of configuration utilities designed to allow realtime tuning,
- a set of performance monitoring tools that allows realtime performance to be monitored and provides information to support realtime tuning, and
- documentation, including information on how to tune a system for optimal latency.

TUNA TUNING TOOL IS INCLUDED WITH MRG REALTIME.



The realtime capabilities of MRG are contained entirely in its realtime kernel, which is currently based on the Linux 2.6.24 kernel. This means that MRG Realtime system runtime environments, system utilities, and glibc runtime libraries are completely unchanged, and all their applications will continue to run. There is no need to recompile applications to run them with MRG Realtime; applications that work on Red Hat Enterprise Linux will continue to work unchanged on Red Hat Enterprise Linux with MRG Realtime.

Red Hat is leading the development of MRG Realtime capabilities in the mainstream open source kernel community. To date, Red Hat engineers have written about 90 percent of the Linux realtime kernel code. Furthermore, by developing these realtime capabilities as part of the broader Linux community, Red Hat ensures that the MRG Realtime offering continues to benefit from the latest Linux kernel features like support for new hardware or bug fixes. Solutions that are not accepted by the mainstream Linux kernel community will eventually become dead-end forks of Linux.

Red Hat Enterprise MRG Realtime includes a plethora of features developed over many years that enhance determinism and performance. Some of the key enhancements in MRG Realtime include:

- **Low-latency interrupt handling.** The lengthy and non-interruptible codepaths inherent in device interrupt handlers are a primary source of system non-determinism. MRG Realtime addresses this by breaking these long-running kernel codepaths into separately scheduled portions. This ensures that lower priority event handling does not block priority realtime processing.
- **Enhanced priority handling.** MRG Realtime includes more fine-grained tuning control. For example, it is possible to schedule network processing at higher priority than disk and storage processing. Additionally, there are scheduler enhancements to further guarantee that higher priority processes get uninterrupted service.
- **Priority inversion avoidance.** MRG Realtime, when used with the standard Red Hat Enterprise Linux glibc runtime library, provides priority inheritance. This is a mechanism to ensure that lower priority processes do not block higher priority processes when contending for shared resources. A scalable priority inheritance algorithm is a requirement for realtime Java.
- **Timer precision.** All realtime applications inherently require accurate timing events. This makes it possible to have time-stamped messages and event timeouts. Conventional Linux has had its timing accuracy governed by a periodic system interrupt, which results in imprecise timing. In contrast, MRG Realtime uses hardware-based event timers, resulting in substantially more accurate and predictable timing.
- **Optimized gettimeofday().** Many applications frequently call gettimeofday() operations for things like time-stamping in log files. MRG Realtime includes an optimized gettimeofday() implementation that does not incur a context switch, resulting in significantly improved performance.
- **Latency tracer.** The latency tracer is a peak detector. It identifies the longest running, non-preemptable kernel codepaths, which is invaluable when determining whether non-deterministic response times are occurring in kernel or application space.

MRG GRID

Red Hat Enterprise MRG Grid lets enterprises move to a new model of utility computing as well as run traditional high-performance computing workloads. With MRG, enterprises can scale their IT infrastructure to meet business demands and at the same time, decrease costs by fully utilizing computing resources in a dynamic and flexible manner.

MRG Grid can dynamically and rapidly run jobs across disparate computing resources, whether they are dedicated grids in local datacenters, physically remote servers in other data centers, or cycles harvested from idle desktop workstations. These systems could be running any platform from Windows to Linux; they could be virtualized containers or bare metal or even dynamically provisioned grids in the cloud. Furthermore, MRG Grid can schedule workloads ranging from sub-second executions to long-running, massively parallel jobs. For everything from adding on-demand capacity for the Christmas rush, to simplifying capital infrastructure, to adding additional computing power for the most demanding workloads, Red Hat Enterprise MRG Grid offers tremendous value and revolutionary flexibility.

MRG Grid is API-less from a workload standpoint. This means that users do not need to code applications and jobs to a specific API in order to take advantage of MRG grid scheduling, simplifying development and preventing lock-in. MRG Grid does include a full web services-based API for the management, monitoring, and administration of jobs so that enterprises can include MRG Grid as part of their broader workflow. Additionally, MRG Grid provides powerful command-line tools, a web-based management console, and leverages MRG Messaging for an AMQP-compliant messaging interface for job submission.

MRG Grid is based on the Condor Project started and hosted by the University of Wisconsin-Madison. Red Hat and the University of Wisconsin formed a strategic partnership to release Condor under an Open Source Initiative (OSI) approved open source license that makes it possible for Condor to be included in open source distributions. Red Hat and the University continue to co-develop Condor together, bringing innovation from the research community to the enterprise. Condor has been in active development and used by a wide community since 1988. Based on their joint vision of advancing open source software, Red Hat and the University of Wisconsin are collaborating to add enhanced enterprise stability and functionality to Condor, add high-throughput computing (HTC) capabilities to Linux, and ultimately advance and strengthen the Condor project and community.

MRG Grid includes the following features:

- **Scalable grid scheduler.** The MRG grid scheduler is based on Condor, which powers many of the largest grids in the world and scales beyond tens of thousands of nodes with ease.
- **Virtualization.** The MRG Grid allows you to submit a virtual machine (VM) as a user job and supports migration of the VM.
- **Cloud scheduling.** MRG Grid allows you to use computing resources from cloud-based environments like Amazon EC2. Furthermore, MRG Grid can seamlessly blend and integrate cloud-based infrastructure with local infrastructure.

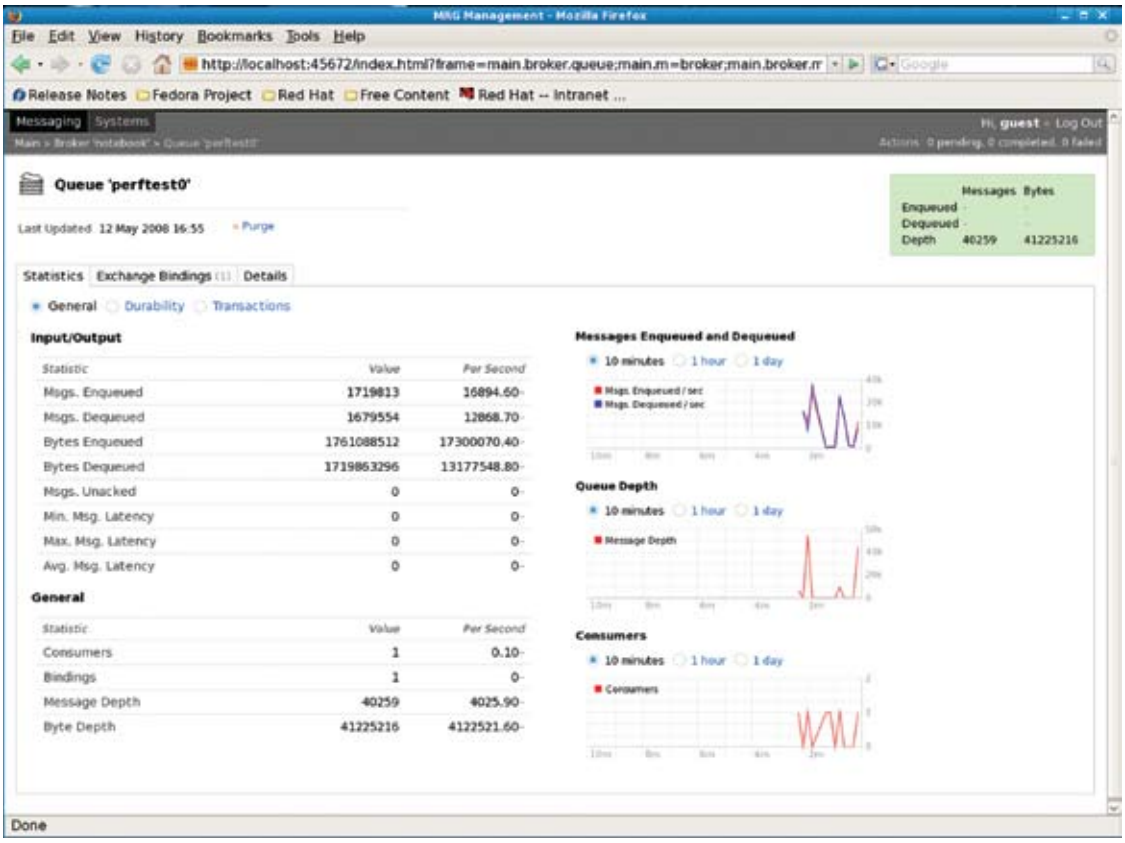
- **Desktop cycle-harvesting.** Desktop cycle-harvesting allows you to leverage the unused capacity of desktop systems to add processing power to your grid.
- **ClassAds.** ClassAds provides a flexible language for policy and meta-data description.
- **Policies.** MRG Grid allows flexible, customizable policies specified by jobs and resources via ClassAds.
- **Low-latency scheduling.** The integration of Messaging and Grid technology allows scheduling and response times in the millisecond range. MRG can also be integrated with Microsoft Excel® to run calculations on a grid.
- **Concurrency limits.** MRG includes the ability to set currency limits on jobs. These limits can restrict the instances of licensed software running at one time or control access to scarce resources.
- **Dynamic provisioning.** MRG Grid dynamically adjusts resource slots based on jobs.
- **Federated grids/clusters.** A mechanism known as flocking allows independent pools to use each others' resources, controllable by customizable policies.
- **Multiple standards-based APIs.** A web services interface provides job submission and management functionality. CLI provides a highly scriptable interface to all functionality, with consistent output.
- **Workflow management.** The MRG Grid provides sophisticated workflow management capabilities.
- **High availability.** The Negotiator and Collector, via HAD, and the Schedd, via Schedd fail-over, can have their state replicated to allow for graceful fail-over upon service disruption.
- **Disk space management.** NeST, a multi-protocol storage management system, exposes the ability to manage (allocate, free, reserve, etc.) disk space to a user's jobs.
- **Database support.** All data about jobs and resources can be stored in a database via Quill.
- **Compute On-Demand (COD).** MRG Grid allows a node or set of nodes to be claimed in such a way that others may use the claimed nodes until they are needed.
- **Dynamic pool creation.** Through technology known as Glide-ins, nodes can be dynamically added to a pool to service user jobs.
- **Priority based scheduling.** MRG Grid can schedule based on priority.
- **Accounting.** User and group resource utilization is tracked and the information is accessible to administrators.
- **Security.** MRG Grid includes comprehensive security.

- **Parallel universe.** MRG Grid provides an extensible framework for running parallel (including MPI) jobs. In this environment, MRG Grid automatically co-allocates compute nodes. Furthermore, MRG Grid provides framework implementation for MPICH1, MPICH2, and LAM.
- **Java universe.** MRG Grid provides explicit support of jobs written in Java.
- **Time scheduling for job execution (Cron).** MRG Grid allows a job (or multiple jobs) to be started at specific times, with customizable policy for failures such as missed deadlines.
- **Backfill.** The MRG Grid allows otherwise unused nodes to run jobs provided by BOINC.
- **File staging.** In the absence of a shared filesystem, MRG Grid supports automatic file staging (job input) and online file I/O (like file streaming from submit to execute nodes) via Chirp and remote syscalls.
- **Dedicated and undedicated node management.** MRG Grid enables dedicated resources (clusters) to be augmented with otherwise undedicated resources (desktops) using flexible policies.
- **Master-Worker (MW).** MRG Grid provides a C++ framework that enables a single master process to allocate and manage multiple worker processes. This data is processed based on master specified policies.
- **Condor-C.** Condor-C allows jobs in one queue to be moved to another queue.
- **Management tools.** MRG Grid includes powerful browser-based tools for managing daemons and machines, security, compute jobs, scalability settings, priorities, and more. MRG also provides sophisticated monitoring capabilities.

MRG MANAGEMENT

Red Hat Enterprise MRG includes a variety of powerful tools for management and monitoring. First among these is a web-based management console that provides a unified management interface across all the components of MRG. With this fully AJAX-enabled management console, enterprises can do everything from managing messaging queue rates to monitoring the utilization of their grids. They can set up rules to grow their grids dynamically when they reach capacity, as well as tune and administer the underlying operating systems in their deployment environments. The Red Hat Enterprise MRG management capabilities are built on native messaging capabilities, so AMQP-compliant messaging is used for its management and monitoring events.

UNIFIED MANAGEMENT



In addition to the web-based console, Red Hat Enterprise MRG also includes scriptable command-line utilities for administration. Because MRG uses AMQP for its management, any messaging client—including scripting languages like Python—can fully participate in the management of a Red Hat Enterprise MRG deployment.

Finally, Red Hat Enterprise MRG includes a web services API for the full administration of MRG Grid jobs.

INTEGRATED VALUE

Each of the components of Red Hat Enterprise MRG—Messaging, Realtime, and Grid—provides distinctive value in and of themselves. However, the integration of these technologies into a single platform provides tremendous value and benefits, including:

- **Deterministic low-latency messaging.** The combination of MRG Realtime and Messaging allows for highly deterministic response times and reliably low latency.
- **Messaging-based reliable, low-latency job scheduling.** By leveraging MRG Messaging and Realtime components, MRG Grid can reliably schedule and execute jobs in less than a second. This vastly increases the range of jobs that MRG can meaningfully handle. If a job scheduler takes several seconds or even minutes to schedule a job, then there is no value in having that job scheduler run rapid jobs. By providing a message-based interface for sub-second job scheduling, MRG Grid can handle all workloads, from sub-second executions to long-running, massively parallel jobs. Furthermore, by leveraging AMQP, Red Hat Enterprise MRG provides a cross-platform and cross-language protocol for submitting jobs from a variety of environments.
- **Interoperability.** Red Hat Enterprise MRG has support for AMQP, bringing interoperability and a new software/hardware ecosystem to everything from basic messaging tasks to distributing data on a grid to management tasks.
- **Unified management.** Red Hat Enterprise MRG builds its management capabilities on top of an AMQP-compliant messaging system. This provides the ability to manage MRG via a messaging interface across a variety of languages and platforms. It also allows for the use of the powerful web-based management console.
- **Reduced complexity and breakthrough value.** Red Hat Enterprise MRG provides an integrated platform for high-performance distributed computing. Without MRG, enterprises would typically need to build a complicated architecture spanning a number of different products to get all the same capabilities. Fast messaging, reliable messaging, large-file messaging, deterministic latency, workload scheduling, and scalable virtulization are all available in this one platform: Red Hat Enterprise MRG. Furthermore, enterprises have to tie these products together to meet their use cases—usually through custom development. Compounding this difficulty is the fact that many of these products are fundamentally incompatible and require deployment in silos for specialized functionality. Finally, each different product has its own management tools and requirements. As an integrated platform, Red Hat Enterprise MRG vastly simplifies the deployment, management, and architecture of enterprise IT while significantly increasing functionality, performance, and value.

LINUX AUTOMATION

Linux Automation with Red Hat creates more flexible and efficient IT infrastructure, reduces IT operating and capital costs, and improves IT service levels by enabling any application certified for Red Hat Enterprise Linux to run optimally anywhere from bare-metal infrastructure to virtualized environments to cloud infrastructure.

Linux Automation has significant benefits at the application level, and they compound greatly when deployed on the grid with Red Hat Enterprise MRG. Through MRG, you can bring the benefits of Linux Automation to all your workloads across all your computing resources.

SUMMARY

Red Hat Enterprise MRG is a revolutionary platform for high-performance distributed computing. By integrating Messaging, Realtime, and Grid functionality, Red Hat provides its customers tremendous value, greatly reducing the costs and complexity in IT infrastructure while simultaneously increasing performance, utilization, reliability, flexibility, and interoperability. For more information on Red Hat Enterprise MRG, visit **www.redhat.com/mrg.**

RED HAT SALES AND INQUIRIES NORTH AMERICA

1-888-REDHAT1
www.redhat.com

www.redhat.com
#865344_1208