Metastorm BPM: Scalability

A Metastorm White Paper

This white paper is provided courtesy of

Executive Summary

Metastorm BPM has been designed specifically to achieve high throughput for large number of simultaneous users in human intensive processes. It recognizes that there are significant differences between Enterprise Application Integration applications and human intensive business processes and takes advantage of the enterprise scalability features of the Microsoft Windows Server platform provide support for these human intensive processes on low cost commodity hardware platforms.

A single Metastorm BPM server will easily support several thousand active, on line process users including support for 500 simultaneous users with a Metastorm BPM action throughput of 30,000 process actions per hour. A Metastorm BPM service will scale by using faster processors or additional servers connected to the Metastorm BPM repository database to support hundreds of thousands of users with a throughput of hundreds of thousands of actions per hour.

Unlike many sizing exercises, the tests run by Metastorm set out to simulate a typical human intensive process workload with users using web browsers to process actions rather than windows clients that have been optimized to minimize load and maximize throughput. Metastorm BPM provides a cost effective platform for both development and production environments to provide organizations with process automation.

Introduction

Metastorm BPM is the leading Business Process Management software solution for automating, managing and controlling unique, human-intensive processes. Organizations around the world have deployed Metastorm BPM to meet critical process needs in the areas of compliance and risk management, customer service, supply chain operations, and internal operations. As a result, these organizations are citing significant returns on their investment. The deployment and use of Metastorm BPM delivers benefits that are immediate, have quick and measurable impact, and continue to increase in value over time.

When deploying Metastorm BPM, organizations have to consider the environment that they will require to support their applications. Consideration must be given to the number of users that the system will have to support and the number of folders (or work items) that the Metastorm BPM environment will have to process and support.

Metastorm BPM has been specifically designed to achieve high throughput for large user communities on low-cost commodity hardware platforms. This scalability is achieved through a combination of proven, stateless application design patterns and Metastorm BPM's use of the excellent scalability enterprise features of the Microsoft Windows Server System, Microsoft SQL Server, and the Microsoft .NET Framework.

There are significant differences between planning for a human intensive Business Process Management (BPM) systems and an Enterprise Application Integration (EAI) application. An EAI system focuses on moving data between servers and the primary consideration is on data transformation. The transaction rates are typically very high as the only time that people become involved in the process is to handle exceptions. Contrast a human intensive BPM system where the primary consideration is presenting data to people so that they are able to make decisions quickly. Even with thousands of users, transaction rates are, relative to an EAI system low; as a human has to consider the information they are presented with before making a decision and hence works much more slowly than a computer.

Metastorm has validated the scalability of Metastorm BPM by running a serious or performance tests over recent years. Whilst the methodology of the testing has varied little, each series of tests are run on updated hardware with more recent versions of Operating Systems and Metastorm BPM Software. This paper focuses on the most recent series of tests running Version 7 of Metastorm BPM on Windows Server 2003. The conclusions draw on information that was learned running earlier tests in the series.

These tests focus on the most meaningful metric in human-centric business process management. That is the capacity to handle both a large number of simultaneous users in addition to the ability to easily handle the peak number of workflow actions that an organization can expect. With Metastorm BPM's unique stateless design the two common scalability challenges faced by many BPM vendors – concurrent users and number of active workflow instances – are irrelevant. Our research and experience with simultaneous usage shows that a Metastorm BPM platform configured to support 1,000 simultaneous users can easily be compared to the 5,000 or more concurrent users. This ratio of 5:1 expands as the total number of users increased up to 33:1.

What does this all really mean?

Metastorm BPM scales to support more than 1,000,000 users on a fraction of the hardware of traditional mainframe or UNIX based transactional systems. A typical Metastorm BPM business process used regularly by over 1,000,000 users would be supported on 51 dual 2.8 GHz hyperthreaded CPU servers. This configuration would handle over 30,000 on line (simultaneous) users and in excess of 3,000,000 actions per hour.

To put this in dollars, the server assets required to support over 1,000,000 highly active process users would be less than US $175,000 using enterprise quality dual CPU servers. This configuration provides not just high scalability but when combined with enterprise class load balancing hardware, software, and database clustering technologies would offer truly world class reliability and maintenance flexibility.

Metastorm BPM: Designed for Scalability from the Ground Up

Stateless Design

Scalability and high-performance are inherent features of Metastorm BPM because of its stateless and transactional design. Client processes communicate with the Metastorm BPM Transaction engine using semantically-rich but compact XML messaging. All interactions with the Metastorm Process Engine are treated as transactions with no state being held between transactions. Resources are allocated for an individual transaction and freed upon completion. As a result, a higher order of requests can be processed with the absolute minimum or resource requirements. On the other hand pooling and resource management techniques are utilized so that overhead of resource instantiation does not counterweight the benefits of Metastorm BPM's stateless design.

Scalabile, High Performance Transaction engine

The Metastorm Process Engine implements and benefits from the stateless design. Implemented using (native) C# and C++, and as a robust high-performance multi-threaded set of components, it treats all Metastorm BPM transactions in isolated fashion using advanced thread management and resource allocation. This prevents critical resource deadlocks and enables system administrators to easily correlate user load and resource usage for better capacity planning.

To further enhance scalability, the Metastorm Process Engine offers built in asynchronous processing for tasks such as notifications and external events (Raise Flag). Asynchronous processing allows the Metastorm Process Engine to perform event-driven tasks without affecting transaction processing.

In addition, Metastorm BPM leverages the scalability features offered by Window's Enterprise Services and .NET. Its implementation as a set of stateless COM+ and .NET components benefits from Just-in-time (JIT) activation, a key scalability feature in multi-tier applications. JIT activation impacts the binding between a client and an engine server component. It offers Metastorm BPM clients the capability of holding a durable reference to a server component while optimizing system resources used on behalf of the server. Server component instances are deactivated as soon as transactions are completed freeing memory resources for other connections thus resulting into higher scalability.

Database Connection Pooling

As JIT activation optimizes client connectivity to the Metastorm Process Engine, similarly, the built-in Windows Enterprise Services and .NET resource pooling benefits Metastorm Process Engine connections to database servers. A configurable pool of database connections is maintained and reused for Metastorm BPM transactions. All Metastorm BPM database transactions are performed through this connection pool.

Given that establishing connections to a database server is expensive in performance and

resource terms, this greatly enhances Metastorm BPM's performance and scalability. This extends to access to external databases, since all database access is performed using Microsoft's Database Access Components (MSDAC).

Dynamic Load Balancing
Metastorm BPM provides an easily configurable load balancing through the Metastorm Process Engine's service list. Using Dynamic Load Balancing (DLB) transaction processing can be spread across multiple engines transparently. This pool of engines offers transparent failover. If an engine is not available, the request will be issued to the next available engine, and so on until an operational engine is found.

This feature operates across both DCOM and HTTP and enables system administrators to deploy sophisticated load balancing architectures.
Since Metastorm BPM DLB is interface-based, third-parties can add their own implementations that may integrate with advanced server allocation heuristics offered by specialized load balancing products.

Configuration Options

It is possible to configure Metastorm BPM in a number of different ways to provide a scalable and resilient Metastorm BPM service. From a standard set up of a single Metastorm Process Engine running with a single web server, the configuration may be enhanced in a number of different ways to scale both up and out.

Multiple web servers

It is a standard Metastorm BPM feature to be able to connect multiple web servers to one Metastorm Process Engine. In this basic configuration, a user can connect to any web server that is available.

It is possible to run the Metastorm BPM web extensions on a Microsoft Server cluster, running IIS within the clustered environment. In this environment, should one web server fail, an automatic fail over capability would be provided to an alternative web server in the cluster.

Multiple engines

A Metastorm BPM service may also be configured with multiple Metastorm Process Engines each running on its own server. In this configuration, it is recommended that Dynamic Load balancing be used to provide automatic failover.
Multiple engine / web server combinations
Both a Metastorm Process Engine and web server may be run on the same server. Multiple servers running the engine / web server combination may be deployed against a single database. This is a frequent combination for many organizations who want the benefits of scalability and resilience.

Database

A Metastorm BPM service can only use one database as its repository. The resilience and scalability options for the Metastorm BPM repository database will depend upon the database selected and the platform it is running on. It is recommended that you contact your database vendor for advice in this area.

Metastorm BPM Scalability Testing and Results

Metastorm are often asked what configuration is required to support a particular number of Metastorm BPM users. Metastorm undertook the tests described in this section of the paper to assist organizations planning to use Metastorm BPM to estimate their hardware requirements for the environment where the Metastorm BPM service will run. It does so by answering the question

"Given a certain usage profile, what number of simultaneous users will a configuration support and with that number of users how many actions per hour would the configuration support?"

A Metastorm BPM service is defined as all Metastorm BPM components that are attached to one Metastorm BPM repository database.

A Metastorm BPM server is defined, for the basis of this paper, as the server running the Metastorm Process Engine, the Metastorm BPM web server extensions and a Microsoft IIS web server.

This paper uses the term "simultaneous user". Simultaneous users are the number of users that simultaneously cause, through a web browser, the Metastorm Process Engine to respond to a request. This could be caused by a user opening a folder, starting an action, submitting an action, causing a refresh on a form, or displaying a Metastorm BPM list. It is this number that matters, rather than the number of Metastorm BPM users that are concurrently signed on to the system (which will typically be significantly greater than the number of simultaneous users).

To provide a method of sizing that would be useful for organizations, a process has been built that reflects different factors that impact throughput. During the test the number of Metastorm BPM actions processed is measured. It is easier for an organization to estimate the number of Metastorm BPM actions they expect to be processed than to have to estimate values for the many different factors (a few of which are mentioned above) that cause the Metastorm Process Engine and the web server to respond to a request and impact the throughput of the Metastorm BPM service. If all of these factors were taken into account it would be impossible for an organization to forecast them accurately.

The Benchmark Process

The tests were run using load testing facilities within Microsoft Visual Studio TeamSuite. This offers the ability to test web applications and XML-based applications in a flexible and relatively easy way. TeamSuite allows you to "record" a set of actions and then replay them with variable

numbers of multiple simultaneous users. You are also able to program time intervals and traffic profiles to simulate other conditions

The process used in these tests was designed to reflect a particular profile of usage, and undertake a number of different factors that impact the throughput of a Metastorm BPM service. This includes opening, completing and submitting a blank form; taking user actions; and executing system actions. The form itself includes field dependencies and contains an image. The test also includes requesting and displaying the To Do list including its filter.

Each test was run for 5 minutes, simulating a designated number of simultaneous users, and the number of actions completed in the period was measured and recorded.
For the single server test, the tests were run initially with 100 simultaneous users. The tests were repeated increasing the number of simultaneous users by 100 each time until there were 600 simultaneous users.

For the test using two servers each with a Metastorm Process Engine and a webserver, the tests were run initially with 200 simultaneous users. The tests were repeated increasing the number of simultaneous users by 200 each time until there were 1,200 simultaneous users. Each test for a particular user load was repeated and the mean value of the results was taken.
The objective of the test was to identify the peak user load for a particular configuration. Each test was stopped after the results indicated that the peak user load had been reached. At no time did the system “fail”. Testing could had continued with additional simultaneous users but response time would have increased and the total action throughput decreased.
The tests simulate users accessing Metastorm BPM through a web browser.

Environment

The initial tests were run on a dual Xeon CPU server with 2 GByte of RAM and 2.8 GHz CPUs with hyperthreading turned on. The server was running Windows Server 2003 SP1. A Metastorm Process Engine, the Metastorm BPM web server extensions and Microsoft IIS Version 6 web server were running on the server. The .NET settings on the server had been changed from the default settings in accordance with Microsoft’s recommendations.

The Engine was connected to a SQL 2000 database running on a separate server running Windows Server 2000 SP4. This server connected via TCP/IP through a 100 Mbit Ethernet. This server had 1Gbyte of RAM and a single 900 MHz CPU. All servers in the tests were connected on the same LAN segment (I.E. not through hubs or routers).

In none of the tests was database performance a factor.

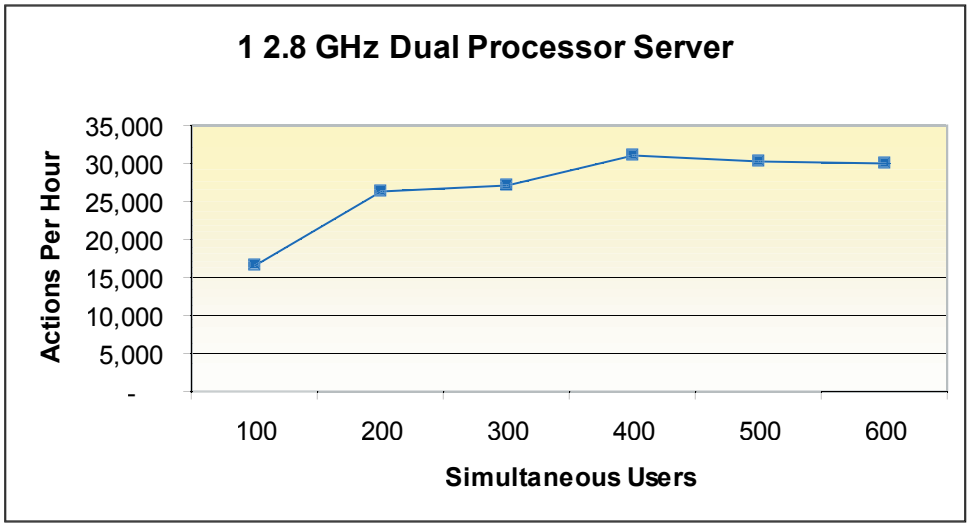
During these tests, different parts of the Environment were monitored and changed to assess the change in throughput of the Metastorm BPM service. This included:

- Monitoring the amount of memory in the server running the Metastorm Process Engine and IIS web server,

- Changing the processor speed in the CPU running the Metastorm Process Engine and IIS web server*,
- Adding additional servers - running both a Metastorm Process Engine and an IIS web server - to connect to the same SQL Server 2000 database.

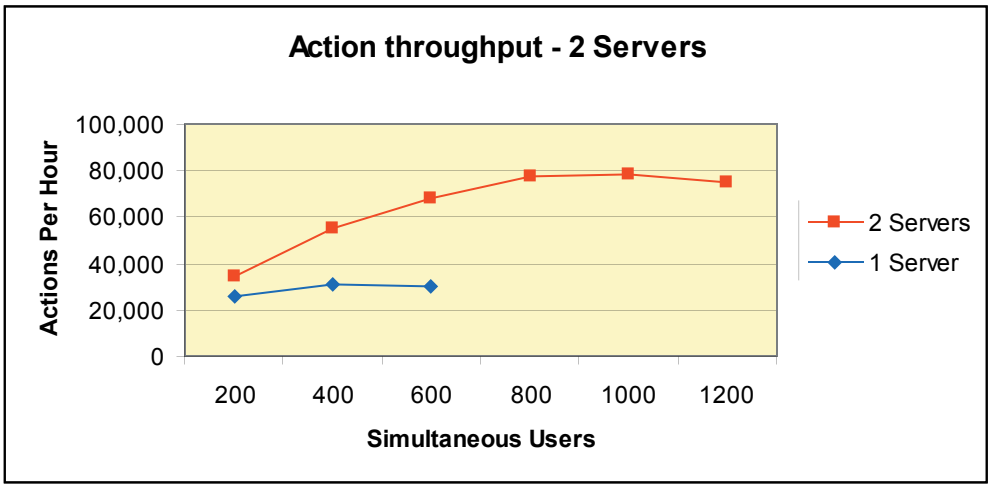
Summary of Results

Throughput in the test running on a dual CPUsingle server with hyperthreading enabled, 2.8 GHz processors and 2Gbyte of memory peaked with 400 simultaneous users at a rate in excess of 31,000 Metastorm BPM process actions per hour. Testing ceased with a successful run of 600 simultaneous users with a throughput in excess of 30,000 actions per hour.



Connecting extra Engines to the Metastorm BPM database

A second dual CPU Metastorm BPM server (running both a Metastorm Process Engine and an IIS web server), with hyperthreading enabled, 2.8 GHz processors and 2Gbytes of RAM, was added to the original configuration. The Metastorm BPM service now consists of two servers of a similar configuration, connected to the same SQL Server 2000 database. Testing ranged from 200 to 1,200 simultaneous users.



With 200 users, a throughput in excess of 34,000 actions per hour was recorded. Throughput increased to a peak of 78,000 process action per hour with 1,000 simultaneous users. With 1,200 simultaneous users, a throughput of 75,000 actions per hour was recorded. Testing ceased at 1,200 simultaneous users although the Metastorm BPM Serviced had not "failed" but there was degradation in throughput.

Memory

Memory on the Metastorm BPM servers was monitored during the tests. At no point did memory utilization increase to where it became a performance bottleneck.

Summary Results One Metastorm BPM Server v. Two Metastorm BPM Servers

No. Metastorm BPM Servers	1	2
RAM	2GByte	2GByte
Processor Speed	2.8 GHz	2.8 GHz
No CPUs	2	2
Action Throughput 200 Simultaneous Users	26,262	34,674
Action Throughput 600 Simultaneous Users	30,054	68,010
Action Throughput 1,200 Simultaneous Users	N/A	75,216

Conclusion

The tests that have been run indicate that:

- A minimum processor speed of 2.5GHz should be recommended for the Metastorm BPM server with a minimum RAM of 1 GByte per processor.
- A dual CPU server with hyperthreading enabled, 2.8 GHz processors and 2Gybte of RAM would support up to 500 simultaneous users with a throughput in excess of 30,000 Metastorm BPM process actions per hour. This would equate to a significantly higher number of logged on (concurrent) users.
- Increasing the speed of the processor of the Metastorm Process Engine server improves throughput. Tests indicate that a 50% increase in CPU speed results in a 50% increase in action throughput*.
- Using additional Engine / Web server combinations connected to the Metastorm BPM database increases the throughput of the Metastorm BPM service. The addition of a second Metastorm BPM server doubled the action throughput*.
- Provided the server running the Metastorm Process Engine and Web server has 1 GByte of RAM installed per CPU, adding additional memory does not impact throughput.
- A slow network segment between the database server and the Engine server has a significant impact on Metastorm Process Engine throughput. Similarly, a very busy network segment will also adversely impact throughput*.
- All servers in the Metastorm BPM service should be on the same network segment*.
- The Metastorm BPM service does not “fail” as the number of simultaneous users exceeds the optimum but response time may increase and the total action throughput decreases.

In the tests run to date, RAM in the Metastorm BPM server does not appear to be a bottleneck. Increasing the amount of RAM above 1 Gbytes is not likely to have an impact on the throughput of the Metastorm BPM service.

Scalability Scenario

An organization has 20,000 users all of whom will submit blank forms twice a week.

3,000 of these users will be involved in approving and processing these forms. This will result in additional 140,000 actions per week.

This is a total of 180,000 actions per week. The key question in determining the configuration required to run this system is the peak load that the system will be subjected to.

Let us assume that all 40,000 submissions and 50,000 of the additional 140,000 actions take place between 9:00am and 12:00 noon on a Monday morning. The rest of the actions are spread evenly across the week. Hence the peak load occurs on a Monday morning at a rate of 30,000 actions per hour. Let us also assume that there is a maximum of 500 simultaneous users

at any one time during this period.

The test application running on Windows Server 2003 with 2GB of RAM and two 2.8 GHz processor connecting to an SQL Server 2000 database on its own server can expect optimal response times with loads up to 30,000 actions per hour with 500 simultaneous users. This server should be adequate with the peak load estimated above.

Other Considerations

The estimate above is based on an average throughput for a typical usage profile for one particular application. User actions require more processing power than system (conditional and timed) actions, and large forms will soak up more power than small forms. It would be advisable to build in spare capacity, not only to cope with variations in the workload profile from the tests, but also for growth as the number of processes deployed increases.

A decision also has to be made as to whether a backup Metastorm BPM server is required. If a backup server is required, it would be advisable to add a second server to run a second Metastorm Process Engine and web server. This would also provide additional spare capacity doubling the original throughput.

Scalability Scenario – Financial Services

A large financial institution in the North West of the USA has stated that the organization will "continue to grow, but we are currently targeting at least 8,000 total users (5,000 concurrent), with 1,000,000+ active workflow objects requiring throughput of 40,000 workflow objects per day (1,000 exiting per peak hour). In this environment, fault tolerance and high availability are critical for our potential 24x7 operations and business continuity."

Based on 8,000 total users and 5,000 concurrent , our testing and experience has shown that 1,000 simultaneous users would be supported with 2 Metastorm Process Engines based on a 5:1 ratio of online/concurrent users to simultaneous users.

For the business process workload, 40,000 workflow objects per day could be considered 40,000 Metastorm BPM Actions, either Blank Form or Folder Actions. For planning purposes, while the organization states that these figures are for a 24x7 operation we would still assume an 8 hour day with a peak rate of 10,000 actions/hour. Our testing shows that with the dual server rate of 75,000 Metastorm BPM actions/hour for 1,200 simultaneous users, Metastorm BPM would require 2 Engines with enough additional capacity for additional workflow actions/hour.

In this scenario the peak of 10,000 workflow actions/hour would also be easily handled in the 2 Engine server configuration with significant spare capacity.

Now if the 40,000 workflow items are just submissions, and there are 5 actions/day per submitted item, then we have 200,000 actions/day (including the submission) or 25,000 actions/

hour. This would be a peak load of 50,000 actions/peak hour. Even at this load level 2 Metastorm Process Engines would provide processing of more than 50,000 workflow actions per hour leaving additional capacity for future growth.

Notes:
1. This sizing model uses 2.8 GHz servers.
2. In a complete performance planning exercise other variables would be included in performance testing including the size of forms and messages, operational requirements, integration, and process characteristics.

More Customer Examples

Consumer and Industrial Credit Provider use Metastorm BPM for customer support and end of lease account management. Worldwide process lease processing requires 10,000 Metastorm BPM actions/day on a dual server environment.

Automobile Manufacturer uses Metastorm BPM for invoice matching and uses a cluster of two Metastorm Process Engine servers running on Compaq Proliant 4 CPU servers with 2GB of RAM. The number of Metastorm BPM folders handled is over 4,000,000/year and the number of transactions executed is well over 26,400,000 / year. This equates to approximately 16,000 / transactions an hour.

Insurance Company validated Metastorm BPM system at 200 actions/minute for 4,000 users with sub second response time. The company was preparing to deploy this system in May and performed this independent analysis in preparation for deployment.

One of the world's largest subway systems uses Metastorm BPM for all incident reporting with over 3,000 active users.

** This conclusion was established while testing with an earlier version of Metastorm BPM. We do not anticipate the conclusion would be different if the test was repeated with the current version.*

This white paper is provided courtesy of Metastorm, Inc. As the first breakaway BPM vendor, Metastorm is a leader in business process management (BPM) software and best practice methodologies for modeling, automating, integrating, and improving both human and system-based processes. Metastorm BPM™ is a complete solution for roundtrip process improvement, designed specifically to address complex processes that are unique to organizations. Metastorm's 1200+ global client base in manufacturing, retail, financial services, business services, healthcare and government are achieving rapid ROI and Enterprise Process Advantage® in customer service, supply chain operations, risk management, and internal operations. More information is available at www.metastorm.com.

1-877-321-META (6382) +44 (0) 208-971-1500 www.metastorm.com