MQSeries Workflow - Performance estimates for solution and capacity assessments

This SupportPac is intended to help IBM technical field experts to approximately determine the processing costs that are associated with the execution of a process instance, find a suitable hardware configuration for a customer offering, and judge the feasibility of an MQSeries Workflow solution proposal.
In this documentation, it is assumed that you are familiar with basic mathematics, MQSeries Workflow, and process modeling.

Version 2, October 2002, Richard Metzger

Contents

Acknowledgments

The following persons contributed in various ways to the content of this paper (listed in no particular order):
Hermann-Josef Weiten
Brigitte Schewe
Axel Dilger
Leonardo Alcantara
Ekkehard Voesch
Holger Kolata
Yoriko Ohnishi
Hohta Nojiri
and special thanks to Dominik Goetz for various hints to improve various descriptions.

Introduction

This document outlines the basic methodology for estimating the MQSeries Workflow performance and server throughput for any given workflow process model for several operating system platforms. It is intended to help you understand the performance impacts of various workflow constructs. In addition, it explains how to calculate the approximate load that is caused by running specific workflow processes on a system. It can also be helpful if you want to determine the approximate machine capacity that is needed to run a specific process mix.

NOTE:
The information in this publication is supplemental to the existing product documentation. It is not intended as a replacement for any product documentation. The information in this document is based on the hardware and software levels that are specified in this document.

NOTE to version 2:
The explanations to the various numbers you have to provide have been extended and (in case you don't have exact numbers yet) some useful defaults are given.
This second version has not only been updated to reflect the actual measurements and newer hardware models. It is also accompanied by a (simple) spreadsheet for performing the capacity planning calculations.

The Basic Workflow Unit (BWU)

When looking into workflow performance and throughput, it is not the most important thing to measure how many gigabytes of data you can put on a specific system or how many memory or processors a system has. These questions become interesting at a different stage. For the workflow performance and throughput, it primarily is interesting to find out 'how much workflow' can be handled on a certain system.
'Workflow' is, among other aspects, about running processes and activities. Therefore, a natural measurement unit could be to use the amount of activities that can be run in a specific timeframe. But activities can be simple or complex, so it is necessary to have a standardized activity. In addition, the main interest is to determine the cost of the workflow processing itself so that the processing costs for business-related programs that are started for a process activity are not considered. To determine how much workflow a system can handle it is necessary to have a workflow-related measurement unit, such as the Basic Workflow Unit (BWU).

How the Basic Workflow Unit (BWU) is defined:

A BWU describes the processing cost that is associated with the execution of a workflow activity having the following characteristics:

- No user-defined container members (only the default container)
- Staff resolution to one person (for example, Done_by Process_Starter), which includes the generation of one workitem
- No pushing of work item status updates to the clients
- Execution of a "no operation" activity program

Measuring the BWU capacity of a system

The following system setup has been used to measure the BWU rate that can be reached by this specific system setup:

- 4 dedicated physical disk drives for MQSeries log files, MQSeries queues, DB2 log files, and DB2 database tablespaces, respectively.
- The "BaseProc" process model, which consists of a synchronization activity as the very first activity followed by a sequence of 100 activities, all having the following properties:
  - Using a default data container
  - Done_by Process_Starter
  - Running a dummy program that has no function
  - Starting and exiting automatically
- No work item status updates are pushed to the clients
- 64 physical clients on system-time synchronized machines, each running a Program Execution Agent (PEA) and a workflow program that logs on to MQSeries Workflow, and creates and starts a process instance of the "BaseProc" process model.

Scenario

At first, each client starts its Program Execution Agent (PEA) and runs the program that creates and starts a process instance of the 'BaseProc' process model. Then, the MQSeries Workflow Server navigates each of the process instances to the first activity to be started and starts this activity. The corresponding synchronization programs are started on the PEAs. These programs are waiting for an exernal event, which, for example, could be a certain file appearing on a common LAN drive.
As soon as the synchronization programs trigger and recognize the external event, they wait until the completion of the next full minute before they return control to the PEA. Since all the clients are system-time synchronized, the completion messages of the first activity of the process instances arrive almost simultaneously at the MQSeries Workflow Server.
From then on, the MQSeries Workflow Server is kept busy with navigating to the next activities, sending StartProgram messages to the PEAs and receiving ProgramFinished messages from the PEAs until all the 100 activities of all the process instances have been executed.
In the next step, the MQSeries Workflow audit trail is evaluated to determine how many activities the MQSeries Workflow Server could complete per second. This result is then taken as the workflow capacity of the MQSeries Workflow Server, which is measured in BWUs per second.

Variations of the 'BaseProc' process model

You can determine what the relative costs of various additional modeling constructs are by running variations of the BaseProc process model. Modeling constructs can be blocks, additional container elements or additional work items, for example. You can determine the relative costs of other workflow functions, such as creating a process instance, or transferring a work item, by running variations of the scenario described above.

Determining the costs for processing a single process instance

To determine the costs for processing a single process instance, you must collect as many characteristics of the corresponding process model as possible. To be as precise as possible you need:

- The average number of activities that are executed in this process
- The average number of activities that are executed with container reuse
- The average number of work items that are generated per activity
- The average number of notifications that are generated per process instance
- The average number of additional container elements that are used in the process
- The average number of enterings and leavings of a block
- The average number of enterings and leavings of a subprocess
- The average number of work item transfers that occur for one process instance
- The average number of work item check-outs that occur for one process instance
- The average number of work item check-ins that occur for one process instance

When you have collected this information, combine the numbers with the relative cost factors that are described in the following section.

Step 1: Calculating the activity cost

By definition the cost of running one program activity is 1 BWU. Depending on the amount of data that is passed from or to the activity, and depending on the number of additional work items, the resulting costs can increase. Those increases are covered with other factors explained below. If you want to calculate the processing costs for a single process instance, take the average number of activities that are executed in a process instance.
Any process activities (activities where a subprocess is being started instead of a program) are treated the same way as program activities, because the basic cost of the transaction and the navigation to and from such an activity is more or less the same as for a program activity.

If you have no idea yet, how many activities you might have, because e.g. no detailled modelling and/or investigations have been done you might assume figures like 5 (or less) activities for simple processes, 10-15 activities for "midsize" processes and 30 (or more) activities for complex processes.

And you need to consider only the amount of activities that are typically (in average) executed within the process. If a process has 20 activities, but under normal circumstances only 5 of them are actually executed then you use 5 activities for your calculations.

Step 2: Calculating the bonus for activities with container reuse

Container reuse is something that takes place, when the output container of a preceding activity can be reused as input container for the current activity. This saves quite some processing effort (avoiding duplication of data) and takes place for an activity automatically whenever the following conditions are met (all of them):

- exactly one preceding activity (i.e. no join in control connectors takes place)
- data mapping only from the preceding activity
- no explicit container element to element mapping (only structure to structure)

An additional loop connector does not prevent container reuse.
Obviously processes with no branches and lot's of sequential activities can benefit most from this feature. The maximum degree of container reuse takes place with purely sequential processes. The maximum number of container reuses for a process with n sequential activities is n-1, e.g. 5 activities with same container structure will have 4 activities with container reuse.

There is **no** container reuse in cases like:

- the process forks (i.e. parallel process activities)
- the process makes a join (i.e. two/more parallel branches merge
- data container mapping is done on a per container member level (i.e. member 1 mapped to member 2 in the following container)
- data container joins and forks (data connectors merge/split)

To calculate the bonus add the activity cost (from above) and the total container cost (from below), then multiply by the average amount of activities with container reuse, divide by the average amount of activities (from above), and take 50% of the resulting figure.
Now subtract that from the activity related cost from step 1.

Scared? Too complicated? Either read it again until you understand it - or just use the spreadsheet.

If you don't know enough about the processes you try to do planning for, then you might use the following rules of thumb as an initial guess:

- for processes with lots of decisions: use 0% - 30% of the number provided for step 1
- for fully sequential processes: use the number provided for step 1 minus one (that's the best you can get)

Step 3: Calculating the work item cost

Take the average number of **additional** work items (short: WI) that are generated per activity and multiply this number with the cost factor for one work item. The cost factor for one work item is 0.02. Then multiply the result with the average number of activities that are executed in a process instance (from Step 1). Then, add the result of this step to your overall total so far.

As the definition of 1 BWU per activity already includes the creation of one WI, only additionally created WIs need to be considered

If you have no idea, how many work items you might have, then use the following:

- for fully automated processes with no human interaction: use 0 (Then you still include the one WI that is considered as part of the activity cost anyways. MQ Workflow is doing internal optimization under certain conditions (Activity Start Automatic, Exit Automatic, and Program Unattended) and does not create a WI if these conditions are met. In this case you might even use a "-1" for your calculations - or leave it at "0" which gives you a little additional buffer)
- for primarily human oriented workflow: use 10 work items as an initial guess.

Step 4: Calculating the notification cost

Take the average number of notifications that are generated per process instance and multiply this number with the cost factor for one notification. The cost factor for one notification is 0.02. Then, add the result of this step to your overall total so far.

Again, if you don't know how many notifications you will have, you might use "1" as an initial guess, if notifications will be used at all, or "0" otherwise.

Step 5: Calculating the container element cost

Take the average number of additional container elements that are used in the process and multiply this number with the cost factor for one container element (which is 0.01). That gives you the cost for the entire container.
Now - each activity has an Inputcontainer and an Output container. So you have to multiply by 2, which yields the total container related cost for one activity.
As a last piece you have to multiply it by the average number of activities (from Step 1) to get the overall container related cost for this process instance.
Add the result of this step to your overall total so far.

If you don't know exactly, how many container elements you might have, you might want to take an initial value of 10 or 20 for your calculations.

Step 6: Calculating the block cost

Take the average number of executions of a block in this process. For example, iterating 5 times through the same block counts as 5 block executions. Multiply this number with 0.8, which is the block cost factor.
Add the result of this step to your overall total so far.

Step 7: Calculating the work item transfer cost

Take the average number of times a work item is transferred in one process instance and multiply this number with the work item transfer cost factor of 0.3.
Add the result of this step to your overall total so far.

Step 8: Calculating the work item check-out cost

Take the average number of times a work item is checked out in one process instance, and multiply this number with the work item transfer cost factor of 0.7.
Then, add the result of this step to the result of the previous steps.

Note that a work item check-out and check-in together use more processing resources than the execution (start) of a work item.

Check-in and Check-out is usually used when you work with the MQWF Webclient.

Step 9: Calculating the work item check-in cost

Take the average number of times a work item is checked in in one process instance, and multiply this number with the work item transfer cost factor of 0.8.
Then, add the result of this step to the result of the previous steps.

Also - in most cases you have the same amount of check-ins as check-outs.

Step 10: Calculating the Process Monitor invocation cost

Take the average number of times the process monitor will be invoked for a single process instance, and multiply this number with the process monitor cost factor of 3.
Then, add the result of this step to the result of the previous steps.

Step 11: Calculating the process instance costs

In the previous section, it is described how to estimate the processing costs for executing all pieces that belong to one process instance. In addtion, it is necessary to consider the costs for creating, starting, and terminating a process instance itself. The costs for these actions are rated with 1.3 BWUs each. This means that you have at least 3.9 BWUs per process instance. Therefore, if you want to calculate the overall processing costs that are necessary for running one process instance, you must add 3.9 BWUs to the results that you obtained in the previous steps.

Determining the MQSeries Workflow base load

Base load is the load that is caused by workflow actions other than the workflow actions listed in the previous sections. The base load is mainly caused by queries, such as work item queries, worklist refreshs, and process instance queries, and by logon and logoff requests. In this paper, only the work item queries and logon/logoff requests are considered.

Number of worklist refreshes per user per hour

This information is only relevant when doing human oriented workflow, e.g. when there are users accessing their worklists and processing tasks assigned to them. Estimating the true costs of work item queries (respectively worklist refreshes) highly depends on the workflow scenario. For example, if the MQSeries Workflow Server wants to send a reply message to a client, the server has to collect the appropriate information from the database first, and then construct a reply message that is sent back to the client. The costs for the database query itself does not so much depend on the size of the result set (which is the amount of data that is returned by the database manager), whereas the costs for constructing the reply message highly depends on the size of the result set.

Based on the current experience, the average cost for one work item query is 1 BWU. When having complex filters and excessive sortings in every query you should use a higher cost factor (e.g. 2 BWU).

The estimation of the base load is based on considering the 'typical refresh rate' of a human MQSeries Workflow user (not of an automated MQSeries Workflow client deamon). The typical refresh rate contains the following considerations:
How much time does a user spend between workitem queries? During this time, the user usually selects a work item from a list, starts the work item, and interacts with the program that has been started as a result of starting the work item. In this scenario, the realistic worst case has also been

considered, in which the user issues a new work item query each time after finishing one work item. If, for example, the user needs an average of 3 minutes to complete a work item, then she/he would do 20 worklist refreshes per hour (average).
If you want to calculate the base load, multiply the average work list refresh rate with the average amount of currently logged-on users.

Number of logons/logoffs per user per hour

Each logon and logoff will contribute to the overall system load. As with worklist refreshes, this is only relevant when using human oriented workflow. Logon intensive time is normally during first hours of a working day. The same applies for logoff. Usually there is a substantial difference between the peak rates and the daily average rate. The average logon rate throughout the day might not represent the requirement of the system during peak logon hours. Therefore it would be a better planning practice to use a rate closer to the peak rate for the calculation.

A logon is associated with a cost of 3 BWU (several messages are flowing back and forth as part of the logon handshaking protocol).
A logoff costs 1 BWU.

Determining the overall load

The overall load that is caused by workflow processing consists of the overall process instance load and the base load. If you want to calculate the overall process instance load, multiply the process instantiation rate with the overall process instance costs.
Example: A process with an overall process cost of 31 BWUs should be instantiated every 5 seconds. In this case, the process instance load is 31 BWUs divided by 5 seconds = 6.2 BWU/sec. To get the overall load, you must add the base load with the overall process instance load.

Handling Subprocesses

Subprocesses require a separate calculation as they are considered as independent processes. From a parent process' view a subprocess is only seen as a single activity and therefore only adds 1 BWU (cost of 1 activity) to the overall process cost. An important factor when dealing with subprocesses is their conditional instantiation rate from a parent process flow. Based on process conditions a parent process might call a subprocess several times during execution. Therefore each running instance of this parent process will have several trailing subprocesses adding to overall costs. When calculating this, it is recommended to calculate the load of just the parent process first, then calculate additional load for the expected number of subprocesses instantiated by parent processes and simply add up both results.

Some remarks before you start using the spreadsheet

	A	B	C	D	E	F
1	**MQ Series Workflow Capacity Estimation Spreadsheet**					
2	(User Inputs in BLUE)					
3						
4	**Process Instance parameters: (calculate once per process)**	**Quantity**	**Processing cost**		**Totals**	
5	**Average estimates**					
6	Activities	5	5	BWU		
7	Activities w/ container reuse (bonus)	0	0	BWU		
8	Workitems	0	0	BWU		
9	Notifications	0	0	BWU		
10	Container elements	10	1	BWU		
11	Blocks	0	0	BWU		
12	Workitem Transfers	0	0	BWU		
13	Workitem Check-Outs	0	0	BWU		
14	Workitem Check-Ins	0	0	BWU		
15	Process Monitor invocations	0	0	BWU		
16	Create/Start/Stop Process Instance	-----	3.9	BWU		
17						
18	Total Process Instance Cost:		9.9	BWU		
19						
20	Number of process instantiations per hour:	25000		**Load:**	**68.75**	**BWU/sec**
21						
22						
23	**System Base Load: (calculate once per system)**					
24						
25	**Worklist Query Load**					
26	Worklist refreshes per user per hour:	20				
27	Concurrently logged on users:	700	**Worklist Query Load:**		**3.89**	**BWU/s**
28						
29	**Logon/off load**					
30	Logons per hour	5				
31	Logoffs per hour	5	**Logon/off Load:**		**0.01**	**BWU/s**
32						
33				**Total Load:**	**72.64**	**BWU/s**
34						

- If you have multiple processes, just duplicate the "Process Instance Parameters" block (i.e. rows 4 to 20 in the above picture) and estimate each process separately. With this approach you always have the total load summarized in column E.
- If you have several major "variations" of the same process (as when you have three distinct paths, and the numbers vary greatly for each) then you might want to treat them as different processes to get better (since more detailled) planning results.
- Remember that performance and capacity is not an "average" thing. You will need to have some understanding of what the peak period demands will be. The peak is the busiest hour of the busiest day of the busiest month of the year. If your system survives that hour, it survives all other hours. Understand the business and what (you think) the end users will do. Performance and capacity planning are an on-going, incrementally activity; it is never really done. If you really have no idea about peak characteristics, you should assume that it will be at least double the average.

RS/6000 and pSeries throughput projections

The pSeries rPerf numbers have been used to compare the relative performance of various RS/6000 and pSeries systems and to project the expected workflow throughput. The rPerf numbers are published on the following Web site:
http://www-1.ibm.com/servers/eserver/pseries/hardware/system_perf.html.

The actual measurements have been taken on the following machine configurations:

- RS/6000 model F50 with 4 processors at 332MHz and 1 GB RAM (23 BWU/sec)
- RS/6000 model H80 with 6 processors at 500MHz and 4 GB RAM (57.5 BWU/sec)
- MQSeries Workflow Version 3.3.2
- 4 dedicated external SSA disk drives
- 64 workflow clients on Windows NT systems

The two base measurements have been used to construct a relation from rPerf to BWU/sec that approximates both measurements as close as possible (linear least squares fit).

System type / Processor	# of CPUs	Clock speed (MHz)	Relative Performance index (rPerf)	rPerf projected to BWU/s	Contin-gency factor for pro-jection	Projected throughput in BWUs per second	Projected throughput in BWUs per hour	Projected throughput in thousand BWUs per 24 hours
IBM RS/6000 Model R50								
7015	2	200	0.38	5.2	1	5.2	18,632	447
7015	4	200	0.69	9.5	1	9.5	34,059	817
7015	6	200	0.96	13.1	1	13.1	47,282	1,135
7015	8	200	1.24	17.0	1	17.0	61,307	1,471
IBM RS/6000 Model F50								
7025	1	166	0.33	4.6	1	4.6	16,429	394
7025	2	166	0.61	8.3	1	8.3	29,852	716
7025	3	166	0.85	11.7	1	11.7	42,073	1,010
7025	4	166	1.10	15.1	1	15.1	54,294	1,303
IBM RS/6000 Model F50H50								
7026	1	332	0.41	5.6	1	5.6	20,035	481
7026	2	332	0.73	10.0	1	10.0	35,862	861
7026	3	332	1.02	14.0	1	14.0	50,488	1,212
7026	4	332	1.33	18.3	1	18.3	65,714	1,577
IBM RS/6000 Model H70								
7026	1	340	0.68	9.3	1	9.3	33,458	803
7026	2	340	1.30	17.8	1	17.8	63,911	1,534

System type / Processor	# of CPUs	Clock speed (MHz)	Relative Performance index (rPerf)	rPerf projected to BWU/s	Contin-gency factor for pro-jection	Projected throughput in BWUs per second	Projected throughput in BWUs per hour	Projected throughput in thousand BWUs per 24 hours
7026	3	340	1.81	24.8	1	24.8	89,155	2,140
7026	4	340	2.32	31.8	1	31.8	114,399	2,746
IBM RS/6000 Model F80/H80								
7017	1	450	0.93	12.8	1	12.8	46,080	1,106
7017	2	450	2.03	27.8	1	27.8	100,174	2,404
7017	4	450	3.57	48.8	1	48.8	175,706	4,217
7017	6	500	4.55	62.3	1	62.3	224,190	5,381
IBM RS/6000 Model S70 Advanced								
7017	4	262	1.87	25.6	1	25.6	92,160	2,212
7017	8	262	3.36	46.0	1	46.0	165,688	3,977
7017	12	262	4.63	63.3	1	63.3	227,997	5,472
IBM RS/6000 Model M80								
RS64 III	2	500	2.49	34.1	1	34.1	122,721	2,945
RS64 III	4	500	4.42	60.5	1	60.5	217,843	5,228
RS64 III	6	500	6.49	88.9	1	88.9	319,864	7,677
RS64 III	8	500	8.53	116.8	1	116.8	420,407	10,090
RS64 IV	2	750	3.71	50.8	1	50.8	182,850	4,388
RS64 IV	4	750	6.68	91.5	1	91.5	329,229	7,901
RS64 IV	6	750	10.14	138.8	0.8	111.1	399,806	9,595
RS64 IV	8	750	13.28	181.8	0.8	145.4	523,611	12,567
IBM RS/6000 Model S80								
7017	6	450	5.01	68.6	1	68.6	247,030	5,929
7017	12	450	9.48	129.8	1	129.8	467,413	11,218
7017	18	450	13.28	181.8	0.8	145.5	523,631	12,567
7017	24	450	16.26	222.6	0.8	178.1	641,115	15,387
IBM p-Series p640-B80 L2-Cache 4MB(=L4), 8MB(=L8)								
P3-II L4	1	375	1.00	13.7	1	13.7	49,286	1,183
P3-II L4	2	375	1.92	26.3	1	26.3	94,629	2,271
P3-II L4	3	375	2.55	34.9	1	34.9	125,679	3,016
P3-II L4	4	375	3.47	47.5	1	47.5	171,021	4,105

System type / Processor	# of CPUs	Clock speed (MHz)	Relative Performance index (rPerf)	rPerf projected to BWU/s	Contin-gency factor for pro-jection	Projected throughput in BWUs per second	Projected throughput in BWUs per hour	Projected throughput in thousand BWUs per 24 hours
P3-II L8	2	375	1.99	27.2	1	27.2	98,079	2,354
P3-II L8	4	375	3.59	49.1	1	49.1	176,936	4,246
P3-II L8	2	450	2.27	31.1	1	31.1	111,879	2,685
P3-II L8	4	450	4.01	54.9	1	54.9	197,636	4,743
IBM p-Series p610-6E1 / 6C1 L2-Cache 4MB(=L4), 8MB(=L8)								
P3-II L4	1	375	1.00	13.7	1	13.7	49,286	1,183
P3-II L4	2	375	1.92	26.3	1	26.3	94,629	2,271
P3-II L8	1	450	1.19	16.3	1	16.3	58,650	1,408
P3-II L8	2	450	2.27	31.1	1	31.1	111,879	2,685
IBM p-Series p620-6F0 / 6F1 / p660-6H0 / 6H1 L2-Cache 2MB(=L2), 4MB(=L4), 8MB(=L8)								
RS64 III L2	1	450	0.93	12.7	1	12.7	45,836	1,100
RS64 III L4	2	450	2.02	27.7	1	27.7	99,557	2,389
RS64 III L4	4	450	3.55	48.6	1	48.6	174,964	4,199
RS64 IV L2	1	600	1.26	17.3	1	17.3	62,100	1,490
RS64 IV L4	2	600	2.69	36.8	1	36.8	132,579	3,182
RS64 IV L4	4	600	4.57	62.6	1	62.6	225,236	5,406
RS64 IV L8	6	668	7.46	102.1	1	102.1	367,671	8,824
RS64 IV L8	1	750	1.91	26.1	1	26.1	94,136	2,259
RS64 IV L8	2	750	3.49	47.8	1	47.8	172,007	4,128
RS64 IV L8	4	750	5.85	80.1	1	80.1	288,321	6,920
RS64 IV L8	6	750	8.23	112.7	1	112.7	405,621	9,735
IBM p-Series p630-6C4 / 6E4								
Power 4	1	1000	1.69	23.1	1	23.1	83,293	1,999
Power 4	2	1000	3.29	45.0	1	45.0	162,150	3,892
Power 4	4	1000	5.89	80.6	1	80.6	290,293	6,967
IBM p-Series p660-6M1 L2-Cache 4MB(=L4), 8MB(=L8)								
RS64 III L4	2	500	2.49	34.1	1	34.1	122,721	2,945
RS64 III L4	4	500	4.42	60.5	1	60.5	217,843	5,228

System type / Processor	# of CPUs	Clock speed (MHz)	Relative Performance index (rPerf)	rPerf projected to BWU/s	Contin-gency factor for pro-jection	Projected throughput in BWUs per second	Projected throughput in BWUs per hour	Projected throughput in thousand BWUs per 24 hours
RS64 IV L8	2	750	3.75	51.3	1	51.3	184,821	4,436
RS64 IV L8	4	750	6.68	91.5	1	91.5	329,229	7,901
RS64 IV L8	6	750	10.14	138.8	0.8	111.1	399,806	9,595
RS64 IV L8	8	750	13.28	181.8	0.8	145.4	523,611	12,567
IBM p-Series p670								
Power 4	4	1100	6.93	94.9	1	94.9	341,550	8,197
Power 4	8	1100	12.72	174.1	0.8	139.3	501,531	12,037
Power 4	16	1100	24.46	334.9	0.6	200.9	723,317	17,360
IBM p-Series p680-S85 L2-Cache 8MB(=L8), 16MB(=L16)								
RS64 III L8	6	450	6.14	84.1	1	84.1	302,614	7,263
RS64 III L8	12	450	11.66	159.6	0.8	127.7	459,737	11,034
RS64 III L8	18	450	16.29	223.0	0.7	156.1	562,005	13,488
RS64 III L8	24	450	20.27	277.5	0.6	166.5	599,413	14,386
RS64 IV L16	4	600	5.6	76.7	1	76.7	276,000	6,624
RS64 IV L16	6	600	8.23	112.7	1	112.7	405,621	9,735
RS64 IV L16	12	600	15.63	214.0	0.7	149.8	539,235	12,942
RS64 IV L16	18	600	21.91	300.0	0.6	180.0	647,910	15,550
RS64 IV L16	24	600	27.65	378.5	0.6	227.1	817,650	19,624
IBM p-Series p690								
P4 L5.6	8	1100	12.72	174.1	0.8	139.3	501,531	12,037
P4 L11.2	16	1100	24.46	334.9	0.6	200.9	723,317	17,360
P4 L16.8	24	1100	33.94	464.7	0.6	278.8	1,003,654	24,088
P4 L22.4	32	1100	42.80	586.0	0.6	351.6	1,265,657	30,376
IBM p-Series p690 Turbo								
P4 L11.2	16	1300	28.96	396.5	0.6	237.9	856,389	20,553
P4 L16.8	24	1300	39.95	546.9	0.6	328.2	1,181,379	28,353
P4 L22.4	32	1300	50.56	692.2	0.6	415.3	1,495,131	35,883
IBM p-Series p690 HPC								
P4 L11.2	8	1300	15	205.4	0.8	164.3	591,429	14,194
P4 L16.8	16	1300	28.96	396.5	0.6	237.9	856,389	20,553

Explanations:

Column title	Explanation
System type	Indicates the IBM pSeries / RS/6000 system and processor type
# of CPUs	Indicates the amount of installed physical processors.
Clock speed (MHz)	Indicates the clock rate with which the processors are running.
Relative Performance index	Indicates the rPerf numbers as published at http://www-1.ibm.com/servers/eserver/pseries/hardware/system_perf.html.
rPerf projected to BWU/sec	The result of the projection of the rPerf number to BWU/sec using the linear least squares approximation based on the measurements.
Contingency factor for projection	For larger machines, 20% and 40% contingency have been included.
Projected throughput in BWU per second.	Throughput in BWU/sec including contingency
Projected throughput in BWU per hour	Indicates the figure from the previous colum scaled to 1 hour.
Projected throughput in 1000 BWU per 24 hours	Indicates the figure from the previous column times 24 divided by 1000

S/390 and zSeries throughput projections

The ITR figures have been used to compare the relative peformance of various S/390 and zSeries systems and to project the expected workflow throughput. The ITR figures are published at the following Web site: http://www-1.ibm.com/servers/eserver/zseries/lspr/zSeries.html. The actual measurement has been taken on the following machine configurations:

- IBM S/390 model 9672-R64 (G3 Enterprise Server) with 6 processors (33 BWU/sec)
- IBM S/390 model 9672-R46 (G5 Enterprise Server) with 4 processors (54 BWU/sec)
- MQSeries Workflow 3.3.2
- 64 MQSeries Workflow clients on Windows NT systems

The two base measurements have been used to construct a relation from ITR to BWU/sec that approximates both measurements as close as possible (linear least squares fit).

System type	# of CPs	Mixed ITR	ITR projected to BWU/s	Contin-gency factor for projection	Projected throughput in BWUs per second	Projected throughput in BWUs per 1 hour	Projected throughput in thousand BWUs within 24 hours
IBM ES/9000 (W3) (zSeries 900 2064-1C1 = 1.00)*							
9021-711	1	0.25	8.2	1	8.2	29,447	707
9021-821	2	0.47	15.1	1	15.1	54,521	1,309

System type	# of CPs	Mixed ITR	ITR projected to BWU/s	Contin-gency factor for projection	Projected throughput in BWUs per second	Projected throughput in BWUs per 1 hour	Projected throughput in thousand BWUs within 24 hours
9021-822	2	0.48	15.5	1	15.5	55,687	1,336
9021-831	3	0.68	21.9	1	21.9	78,720	1,889
9021-832	3	0.69	22.3	1	22.3	80,178	1,924
9021-941	4	0.87	28.1	1	28.1	101,170	2,428
9021-942	4	0.9	28.9	1	28.9	104,086	2,498
p9021-9X2	5	1.05	33.8	1	33.8	121,579	2,918
9021-952	5	1.1	35.2	1	35.2	126,827	3,044
9021-962	6	1.27	40.9	1	40.9	147,236	3,534
9021-972	7	1.46	47.1	1	47.1	169,394	4,065
9021-982	8	1.64	52.6	1	52.6	189,512	4,548
9021-9X2	10	1.96	63.2	1	63.2	227,414	5,458
IBM S/390 G3 Enterprise (zSeries 900 2064-1C1 = 1.00)*							
9672-RA4	1	0.14	4.4	1	4.4	15,744	378
9672-RB4	2	0.25	8.2	1	8.2	29,447	707
9672-RC4	3	0.48	15.5	1	15.5	55,687	1,336
9672-R14	1	0.19	6.2	1	6.2	22,158	532
9672-R24	2	0.36	11.7	1	11.7	41,984	1,008
9672-R34	3	0.53	17.0	1	17.0	61,227	1,469
9672-R44	4	0.69	22.0	1	22.0	79,303	1,903
9672-R54	5	0.83	26.8	1	26.8	96,505	2,316
9672-R64	6	0.97	31.3	1	31.3	112,832	2,708
9672-R74	7	1.11	35.6	1	35.6	127,993	3,072
9672-R84	8	1.22	39.4	1	39.4	141,696	3,401
9672-R94	9	1.33	42.8	1	42.8	154,233	3,702
9672-RX4	10	1.42	45.8	1	45.8	164,729	3,954
9672-RY4	10	1.56	50.1	1	50.1	180,182	4,324
IBM S/390 G4 Enterprise Server (zSeries 900 2064-1C1 = 1.00)*							
9672-RA5	1	0.2	6.3	1	6.3	22,741	546
9672-RB5	2	0.35	11.3	1	11.3	40,526	973
9672-RC5	3	0.58	18.7	1	18.7	67,349	1,616
9672-R15	1	0.25	8.1	1	8.1	29,156	700
9672-R25	2	0.46	14.9	1	14.9	53,646	1,288
9672-R35	3	0.66	21.1	1	21.1	76,096	1,826
9672-R45	4	0.84	26.9	1	26.9	96,797	2,323

System type	# of CPs	Mixed ITR	ITR projected to BWU/s	Contin-gency factor for projection	Projected throughput in BWUs per second	Projected throughput in BWUs per 1 hour	Projected throughput in thousand BWUs within 24 hours
9672-R55	5	1.07	34.3	1	34.3	123,328	2,960
9672-R65	6	1.22	39.3	1	39.3	141,405	3,394
9672-R75	7	1.36	43.7	1	43.7	157,440	3,779
9672-R85	8	1.48	47.6	1	47.6	171,435	4,114
9672-R95	9	1.59	51.0	1	51.0	183,680	4,408
9672-RX5	10	1.68	53.9	1	53.9	193,885	4,653
9672-RY5	10	1.89	60.8	1	60.8	218,959	5,255
IBM S/390 G5 Enterprise Server (zSeries 900 2064-1C1 = 1.00)							
9672-RA6	1	0.34	10.9	1	10.9	39,354	945
9672-R16	1	0.45	14.5	1	14.5	52,087	1,250
9672-RB6	2	0.64	20.6	1	20.6	74,079	1,778
9672-R26	2	0.86	27.7	1	27.7	99,543	2,389
9672-RC6	3	1.24	39.9	1	39.9	143,527	3,445
9672-RD6	4	1.62	52.1	1	52.1	187,511	4,500
9672-T16	1	0.49	15.8	1	15.8	56,716	1,361
9672-T26	2	0.94	30.2	1	30.2	108,803	2,611
9672-R36	3	1.36	43.7	1	43.7	157,417	3,778
9672-R46	4	1.77	56.9	1	56.9	204,874	4,917
9672-R56	5	2.16	69.4	1	69.4	250,015	6,000
9672-R66	6	2.53	81.3	1	81.3	292,842	7,028
9672-R76	7	2.89	92.9	1	92.9	334,511	8,028
9672-R86	8	3.22	103.5	1	103.5	372,708	8,945
9672-R96	9	3.54	113.8	1	113.8	409,747	9,834
9672-RX6	10	3.84	123.5	1	123.5	444,472	10,667
IBM S/390 G5 Turbo Enterprise Server (zSeries 900 2064-1C1 = 1.00)							
9672-Y16	1	0.6	19.3	1	19.3	69,449	1,667
9672-Y26	2	1.14	36.7	1	36.7	131,953	3,167
9672-Y36	3	1.66	53.4	1	53.4	192,141	4,611
9672-Y46	4	2.15	69.1	1	69.1	248,858	5,973
9672-Y56	5	2.63	84.6	1	84.6	304,417	7,306
9672-Y66	6	3.07	98.7	1	98.7	355,346	8,528
9672-Y76	7	3.49	112.2	1	112.2	403,960	9,695
9672-Y86	8	3.89	125.1	1	125.1	450,259	10,806
9672-Y96	9	4.25	136.6	0.9	123.0	442,735	10,626

System type	# of CPs	Mixed ITR	ITR projected to BWU/s	Contin-gency factor for projection	Projected throughput in BWUs per second	Projected throughput in BWUs per 1 hour	Projected throughput in thousand BWUs within 24 hours
9672-YX6	10	4.59	147.6	0.9	132.8	478,154	11,476
IBM S/390 G6 Enterprise Server (zSeries 900 2064-1C1 = 1.00)							
9672-X17	1	0.68	21.9	1	21.9	78,709	1,889
9672-X27	2	1.28	41.2	1	41.2	148,157	3,556
9672-X37	3	1.86	59.8	1	59.8	215,291	5,167
9672-X47	4	2.41	77.5	1	77.5	278,952	6,695
9672-X57	5	2.94	94.5	1	94.5	340,299	8,167
9672-X67	6	3.45	110.9	1	110.9	399,330	9,584
9672-X77	7	3.95	127.0	1	127.0	457,204	10,973
9672-X87	8	4.42	142.1	0.9	127.9	460,445	11,051
9672-X97	9	4.86	156.3	0.9	140.6	506,281	12,151
9672-XX7	10	5.28	169.8	0.9	152.8	550,034	13,201
9672-XY7	11	5.67	182.3	0.9	164.1	590,661	14,176
9672-XZ7	12	6.03	193.9	0.8	155.1	558,368	13,401
IBM S/390 G6 Turbo Enterprise Server (zSeries 900 2064-1C1 = 1.00)							
9672-Z17	1	0.79	25.4	1	25.4	91,441	2,195
9672-Z27	2	1.48	47.6	1	47.6	171,307	4,111
9672-Z37	3	2.14	68.8	1	68.8	247,700	5,945
9672-Z47	4	2.78	89.4	1	89.4	321,779	7,723
9672-Z57	5	3.4	109.3	1	109.3	393,543	9,445
9672-Z67	6	3.99	128.3	1	128.3	461,834	11,084
9672-Z77	7	4.55	146.3	0.9	131.7	473,987	11,376
9672-Z87	8	5.08	163.3	0.9	147.0	529,199	12,701
9672-Z97	9	5.58	179.4	0.9	161.5	581,286	13,951
9672-ZX7	10	6.04	194.2	0.8	155.4	559,293	13,423
9672-ZY7	11	6.48	208.3	0.8	166.7	600,037	14,401
9672-ZZ7	12	6.87	220.9	0.8	176.7	636,150	15,268
zSeries 800 (zSeries 900 2064-1C1 = 1.00)							
2066-001	1	0.77	24.8	1	24.8	89,126	2,139
2066-002	2	1.4	45.0	1	45.0	162,047	3,889
2066-003	3	2	64.3	1	64.3	231,496	5,556
2066-004	4	2.54	81.7	1	81.7	293,999	7,056

System type	# of CPs	Mixed ITR	ITR projected to BWU/s	Contin-gency factor for projection	Projected throughput in BWUs per second	Projected throughput in BWUs per 1 hour	Projected throughput in thousand BWUs within 24 hours
zSeries 900 1xx (zSeries 900 2064-1C1 = 1.00)							
2064-101	1	0.95	30.5	1	30.5	109,960	2,639
2064-102	2	1.8	57.9	1	57.9	208,346	5,000
2064-103	3	2.58	83.0	1	83.0	298,629	7,167
2064-104	4	3.32	106.7	1	106.7	384,283	9,223
2064-105	5	4.01	128.9	0.9	116.0	417,734	10,026
2064-106	6	4.65	149.5	0.9	134.6	484,405	11,626
2064-107	7	5.25	168.8	0.9	151.9	546,908	13,126
2064-108	8	5.79	186.2	0.9	167.5	603,162	14,476
2064-109	9	6.25	201.0	0.8	160.8	578,739	13,890
2064-110	10	7.73	248.5	0.8	198.8	715,785	17,179
2064-111	11	8.32	267.5	0.8	214.0	770,418	18,490
2064-112	12	8.87	285.2	0.8	228.2	821,347	19,712
2064-113	13	9.39	301.9	0.75	226.4	815,154	19,564
2064-114	14	9.88	317.7	0.75	238.2	857,691	20,585
2064-115	15	10.34	332.5	0.75	249.3	897,624	21,543
2064-116	16	10.78	346.6	0.75	260.0	935,821	22,460
2064-1C1	1	1	32.2	1	32.2	115,748	2,778
2064-1C2	2	1.9	61.1	1	61.1	219,921	5,278
2064-1C3	3	2.75	88.4	1	88.4	318,307	7,639
2064-1C4	4	3.56	114.5	1	114.5	412,062	9,889
2064-1C5	5	4.34	139.5	0.9	125.6	452,111	10,851
2064-1C6	6	5.08	163.3	0.9	147.0	529,199	12,701
2064-1C7	7	5.8	186.5	0.9	167.8	604,204	14,501
2064-1C8	8	6.48	208.3	0.8	166.7	600,037	14,401
2064-1C9	9	7.13	229.2	0.8	183.4	660,226	15,845
zSeries 900 2xx (zSeries 900 2064-1C1 = 1.00)							
2064-2C1	1	1.21	38.9	1	38.9	140,055	3,361
2064-2C2	2	2.3	74.0	1	74.0	266,220	6,389
2064-2C3	3	3.32	106.7	1	106.7	384,283	9,223
2064-2C4	4	4.29	137.9	0.9	124.1	446,902	10,726
2064-2C5	5	5.22	167.8	0.9	151.1	543,783	13,051
2064-2C6	6	6.11	196.4	0.8	157.2	565,775	13,579
2064-2C7	7	6.95	223.5	0.8	178.8	643,558	15,445

System type	# of CPs	Mixed ITR	ITR projected to BWU/s	Contin-gency factor for projection	Projected throughput in BWUs per second	Projected throughput in BWUs per 1 hour	Projected throughput in thousand BWUs within 24 hours
2064-2C8	8	7.76	249.5	0.8	199.6	718,563	17,246
2064-2C9	9	8.52	273.9	0.8	219.1	788,937	18,934
2064-210	10	9.23	296.8	0.75	222.6	801,264	19,230
2064-211	11	9.91	318.6	0.75	239.0	860,296	20,647
2064-212	12	10.55	339.2	0.75	254.4	915,855	21,981
2064-213	13	11.16	358.8	0.75	269.1	968,809	23,251
2064-214	14	11.73	377.1	0.75	282.9	1,018,292	24,439
2064-215	15	12.27	394.5	0.75	295.9	1,065,169	25,564
2064-216	16	12.77	410.6	0.75	307.9	1,108,575	26,606

Contingency factor rules:
up to ITR=4: 100%, up to ITR=6: 90%, up to ITR=9: 80%, above ITR=9: 75%

SUN throughput projections

For the SUN platform, there is no projection base available that can be compared to the RS/6000 and S/390 platforms. Therefore, only a subset of measurements and projections is offered here. All projections are made by using the TPC-C benchmark results, which are published at the following Web site: http://www.tpc.org/tpcc/default.asp .
One set of measurements has been taken on a SUN Enterprise 450 (Ultra 4) machine with the following machine configuration:

- 4 CPUs (at 296MHz)
- 2GB memory
- 4 dedicated external SCSI drives

This machine reached a maximum throughput of 17 BWUs per second (using MQSeries Workflow Version 3.2.2.)

System type	# of CPUs	Clock speed (MHz)	Tpc-C>	Throughput relative to 450-4-296	Contingency factor	Resulting throughput relative to 4500-8	Projected throughput in BWUs per second/FONT>	Projected throughput in BWUs per hour	Projected throughput in 1000 BWUs in 24 hours
SUN Ultra Enterprise 6000 (c/s)									
	8	250	15694	1.05	1.00	1.05	17.92	64499	1548
SUN Enterprise 450									
tpc projected !!	4	296	14892	1.00	1.00	1	17.00	61200	1469
	4	400	20124	1.35	1.00	1.35	22.97	82703	1985
SUN Enterprise 3500 Server (c/s)									

System type	# of CPUs	Clock speed (MHz)	Tpc-C>	Throughput relative to 450-4-296	Contingency factor	Resulting throughput relative to 4500-8	Projected throughput in BWUs per second/FONT>	Projected throughput in BWUs per hour	Projected throughput in 1000 BWUs in 24 hours
	8	336	21872	1.47	1.00	1.47	24.97	89887	2157
SUN Enterprise 4500									
tpc projected !!	8	400	28725	1.93	1.00	1.93	32.79	118050	2833
	14	400	50268	3.38	1.00	3.38	57.39	206588	4958
SUN Enterprise 6500 Server									
	24	336	53050	3.56	1.00	3.56	60.56	218020	5232
SUN Starfire Enterprise 10000									
	64	400	115396	7.75	0.75	5.81	98.80	355683	8536
newer tpc result	64	400	156873	10.53	0.75	7.9	134.31	483528	11605

Another set of measurements has been taken using a customer-specific process model. In this case, the finished process rate has been measured. The machine on which the measurements have been made is a SUN 4500 Server that has been configured as follows:

- 8 processors
- 5 GB RAM
- Solid-state disk drives dedicated for MQ queues, MQ log, DB2 tablespaces, DB2 log
- XML message generator
- Multithreaded, multiple UPESes (w/dedicated queues) on the same system

The costs for processing one process instance (using the calculation method described above) is 17.1 BWUs. For MQSeries Workflow Version 3.2.2, you can derive an approximate workflow processing capacity of 80 BWUs per second from the measured process instance completion rate (4.7 process instances per second).

Note:

The two configurations cannot be compared directly, due to the following differences:

- The usage of the MQ Server interface (vs. MQ Clients)
- The High speed I/O subsystem
- The multiple MQ queues for the program execution

All results favorite the configuration using the MQ Server interface, a high-speed I/O subsystem, and multiple MQ queues, because the projected throughput figures that are achieved are higher. On an IBM RS/6000 machine, the same magnitude of improvements can be expected.

System type	# of CPUs	Clock speed (MHz)	Tpc-C result	throughput relative to 4500-8	Contin-gency factor	Resulting throughput relative to 4500-8	Projected throughput in BWUs per second	Projected throughput in BWUs per hour	Projected throughput in thousand BWUs in 24 hours
Ultra Enterprise 6000 (c/s)									
	8	250	15694	0.55	1.00	0.55	43.71	157355	3776.53
Enterprise 450									
tpc result projected	4	296	14892	0.52	1.00	0.52	41.47	149306	3583.35
	4	400	20124	0.70	1.00	0.7	56.05	201765	4842.36
Enterprise 3500 Server (c/s)									

System type	# of CPUs	Clock speed (MHz)	Tpc-C result	throughput relative to 4500-8	Contin-gency factor	Resulting throughput relative to 4500-8	Projected throughput in BWUs per second	Projected throughput in BWUs per hour	Projected throughput in thousand BWUs in 24 hours
	8	336	21872	0.76	1.00	0.76	60.91	219293	5263.03
Enterprise 4500									
tpc result projected	8	400	28725	1.00	1.00	1	80	288000	6912
	14	400	50268	1.75	1.00	1.75	140	504000	12096
Enterprise 6500 Server									
	24	336	53050	1.85	1.00	1.85	147.75	531892	12765.41
Starfire Enterprise 10000									
	64	400	115396	4.02	0.75	3.01	241.04	867739	20825.75
newer tpc result	64	400	156873	5.46	0.75	4.1	327.68	1179635	28311.25

Windows throughput projections

Our throughput measurement was done on a Dual Pentium III 550MHz (XEON) system with 512MB RAM and 3 disk drives, running Windows NT 4.0 and MQWF 3.3.0. The result was 16 BWU/sec. To get a rough estimate for different processor speeds, you can scale linearly. For different amount of processors (1 or 4) just divide by 1.7 (for one processor - average scaling factor determined from comparing various sources) or multiply by 1.5 (for 4 processors - average scaling factor determined from various sources). For Pentium IV based systems an increase of 10% could be assumed.

With these figures, one gets the following table:

System type / Processor	Clock speed (MHz)	Performance relative to Dual P-III 550Mhz	Relative performance projected to BWU/s	Contin-gency factor for pro-jection	Projected throughput in BWUs per second	Projected throughput in BWUs per hour	Projected throughput in thousand BWUs per 24 hours
Single Pentium III							
	300	0.32	5.1	1	5.1	18,481	444
	500	0.53	8.6	1	8.6	30,802	739
	700	0.75	12.0	1	12.0	43,123	1,035
	800	0.86	13.7	1	13.7	49,283	1,183
Dual Pentium III							
(measured)	550	1.00	16.0	1	16.0	57,600	1,382
	600	1.09	17.5	1	17.5	62,836	1,508
	700	1.27	20.4	1	20.4	73,309	1,759
	1000	1.82	29.1	1	29.1	104,727	2,513

System type / Processor	Clock speed (MHz)	Performance relative to Dual P-III 550Mhz	Relative performance projected to BWU/s	Contin-gency factor for pro-jection	Projected throughput in BWUs per second	Projected throughput in BWUs per hour	Projected throughput in thousand BWUs per 24 hours
Quattro Pentium III							
	550	1.50	24.0	1	24.0	86,400	2,074
	600	1.64	26.2	1	26.2	94,255	2,262
	700	1.91	30.5	1	30.5	109,964	2,639
	1000	2.73	43.6	1	43.6	157,091	3,770
Single Pentium IV							
	1000	1.18	18.8	1	18.8	67,765	1,626
	1500	1.76	28.2	1	28.2	101,647	2,440
	2000	2.35	37.6	1	37.6	135,529	3,253
	2400	2.82	45.2	1	45.2	162,635	3,903
Dual Pentium IV							
	1000	2.00	32.0	1	32.0	115,200	2,765
	1500	3.00	48.0	0.9	43.2	155,520	3,732
	2000	4.00	64.0	0.9	57.6	207,360	4,977
	2400	4.80	76.8	0.9	69.1	248,832	5,972
Quattro Pentium IV							
	1000	3.00	48.0	0.9	43.2	155,520	3,732
	1500	4.50	72.0	0.9	64.8	233,280	5,599
	2000	6.00	96.0	0.8	76.8	276,480	6,636
	2400	7.20	115.2	0.8	92.2	331,776	7,963

Contingency rules: 3 times above reference: 10%, 6 times above reference: 20%.

Multi-tier and multi-system configurations

If a system becomes CPU-bound and you need to extend your system, you can add a second hardware system that can host the DB2 database. This configuration is also known as 3-tier configuration. In a 3-tier configuration, the components are distributed as follows:

Tier 1		*Tier 2*		*Tier 3*
Workflow Clients MQSeries Client	TCP/IP	MQSeries Queue Manager Workflow Servers	TCP/IP	DB2 database manager with MQSeries Workflow database

In a 2-tier setup, the communication between the MQSeries Workflow servers (which are database clients) and the DB2 database server is realized via the inter-process-communication facilities of the operating system, whereas in a 3-tier setup, the communication is realized via the network.

This additional communication overhead influences the achievable overall throughput and response times. The limiting factors are line speed, networking load and capacity, and communication software efficiency.

Example:
The 3-tier setup in the test environment consisted of 2 RS/6000 F50 machines that were connected via a single 16 MBit Token Ring. With this setup, a degradation between 6% and 10% has been measured in throughput, compared to a 2-tier setup with 1 RS/6000 F50 machine.
Another potential growth scenario is to add another workflow system that can be set up as follows:

<table>
<tr><th>Tier 1</th><th></th><th>Tier 2</th><th></th><th>Tier 3</th></tr>
<tr><td rowspan="2">MQSeries
Workflow clients
MQSeries Client</td><td rowspan="2">TCP/IP</td><td>MQSeries Queue Manager
MQSeries Workflow
servers (System 1)</td><td rowspan="2">TCP/IP</td><td rowspan="2">DB2 database manager
with MQSeries Workflow
database</td></tr>
<tr><td>MQSeries Queue Manager
MQSeries Workflow
servers (System 2)</td></tr>
</table>

In this configuration, there are two MQSeries Workflow systems in the same system group; the MQSeries Workflow clients are connected (equally distributed) to the two Queue Managers, which are grouped to an MQ cluster. The MQSeries Workflow servers of the two MQSeries workflow systems are connected to the same DB2 MQSeries Workflow database.
The results that have been achieved in the test environment showed that there is roughly a 60% (on Tier 2) to 40% (Tier 3) distribution of the overall load.

Spreadsheet assistance

The MQ Series Workflow Capacity Estimation Spreadsheet provides you some help for the required processing capacity for the machines used in a 3-tiered multiple system setup. You can find the figures in rows 35 to 46.

35	**Multiple system setup configurations**			
36	3-tier (1 workflow system (WF), 1 database system(DB))	WF:	43.59	BWU/s
37		DB:	29.06	BWU/s
38				
39	3-tier (2 workflow systems(WF), 1 database system(DB))	WF:	21.79	BWU/s
40		WF:	21.79	BWU/s
41		DB:	29.06	BWU/s
42				
43	3-tier (3 workflow systems(WF), 1 database system(DB))	WF:	14.53	BWU/s
44		WF:	14.53	BWU/s
45		WF:	14.53	BWU/s
46		DB:	29.06	BWU/s

It just takes the calculated total load and splits it up into a workflow related piece (60% of the total load), which is evenly distributed among the number of workflow systems and a database related piece (40% of the total load).

Sample calculations

In this section, you are provided with two typical samples of performance estimates for solution and capacity assessments, together with the necessary interpretations and explanations.

Sample 1

The first sample shows the scenario of a typical customer input that is handled by the MQSeries Workflow system. It is explained which assumptions can be made to get a sizing estimate.
The customer scenario is as follows:

- 500,000 orders per year
- 2500 orders per day
- 5000 orders per day peak
- Orders are kept in the queue for 3 days
- 40 instances of work items in the workflow process
- 100 users using the workflow process
- 10 roles are assigned to the workflow process
- Every user can assume any and all roles

This information has to be converted into MQSeries Workflow-relevant information, so the revised and necessary information is following:

- There are 5000 process instances per day peak
- The process instance lifetime is 3 days
- There are 40 activities (on average)per process instance (some of these activities reside in sub-processes)
- There are 15 users per role (on average)
- The activities 'are performed by' up to 7 roles

When the tests were peformed, there was no more detailed business process model available so that 5 additional activities per process instance were assumed to meet the criterion that some of the activities reside in subprocesses. Further, it was assumed that the data container holds 15 data members (on average) because no details were available about the amount of workflow container elements.
Assuming that all role sets are disjunct, there are up to 7*15 (=105) work items per activity.
For planning purposes, the peak load that the workflow system has to handle is estimated with 5000 process instances per day. To break this time frame down to a smaller unit, it is assumed that these 5000 process instances are evenly distributed 4 hours. The result of this is an instantiation rate of about 0.35 process instances per second.
The processing costs for a single process instance can be estimated by using the rules and calculation factors described in the previous sections.

Process instantiation, start, and stop	3.9 BWUs
45 activities	45 BWUs
15 container members/activity * 2 (in-container and out-container) * 0.01 BWU/member * 45 activities	13.5 BWUs
105 work items/activity * 45 activities * 0.02 BWU/work item	94.5 BWUs
Total process instance costs	156.9 BWUs

Because there is no data available about the real user actions, such as the rate for work item queries, a maximum of 50% of the process instantiation costs are estimated as the base load. This leasds to processing costs of 235 BWUs (156.9 + 50%) per process instance. If you combine these processing

costs with the previously calculated process instantiation rate, you ge a processing capacity of 82,25 BWUs per second (235 BWUs per process instance * 0.35 process instances per second = 82.25 BWUs per second).
If you look up the processing capacity of 82,25 BWUs per second in the tables shown in the previous sections, a suitable system would be a RS/6000 M80 8-way (projected for 96 BWUs/sec), or a RS/6000 S80 12-way (projected for 106.7 BWUs/sec).

Sample 2

In the second sample, the emphasis is on the first steps of a process only so that the scenario for the sample is different. The customer scenario is as follows:
5000 employees of a company perform a certain planning task once a month for their working assignments. To perform the task, the employees connect to MQSeries Workflow to create and to start one process instance, and then run the first activity of a process with multiple activities. The whole processing should be performed within the first 10 minutes of a defined time frame, for example, on the first day of a month, at 9 am PST. The other activities of these process instances are less time-critical and can be processed during the rest of the day. The machines that are used for this workflow scenario are existing SUN systems (quoted as 2.5 times as powerful as an RS/6000 F50). The processing costs that are generated by each employee is estimated as follows:

- For the logon process: approximately 2 BWUs
- For the QueryProcessTemplates: approximately 1 BWU
- For the CreateStartProcessinstance: 2.6 BWUs
- For running 1 activity: 1 BWU
- For 10 container members (assumption): 0.2 BWUs (0.01*10*2)

As a result, each employee generates processing costs of 6.8 BWUs per person within the first 10 minutes. With this result, you can calculate the required processing rate as follows:
Having 5000 persons, each one processing 6.8 BWUs within 10 minutes (600 seconds), a processing capacity of 5000*6.8/600 = 56.7 BWUs per second is required.
Server load considerations:
A F50 (4-way, 332 MHz) can process 20 BWUs/sec. One of the SUN systems of the customer, which has been rated as being 2.5 times as powerful, is expected to process 2.5*20 = 50 BWUs/sec (first approximation). This means that a single SUN machine is not sufficient for processing the required 56.7 BWUs per second.
If you have a 3-tier setup with two SUN machines as workflow systems, and a third machine that is used as the database server, the available processing capacity is more than sufficient. A 3-tier setup like this (also see the section about the multi-tiered multi-system configuration) is expected to process 50+60% = 80 BWUs/sec. Further, the load that takes place once in a month and lasts only for a few hours, like the load in the described scenario, does not require an investment for dedicated hardware systems.
If the system setup is intended to utilize MQSeries Workflow API clients on the user systems and you want to connect all 5000 users of the scenario, the necessary number of MQ Client connections will highly increase. To reduce the number of MQ Client connections to the system on which the Queue Manager is running, it is recommended to set up more than one Queue Manager (each one on a separate system). With this setup, you can also distribute the load between the 2 MQSeries Workflow systems.
Note:
This does does not apply for users that connect to MQSeries Workflow via a browser using the MQSeries Workflow Web Client.

Notices

Trademarks and service marks