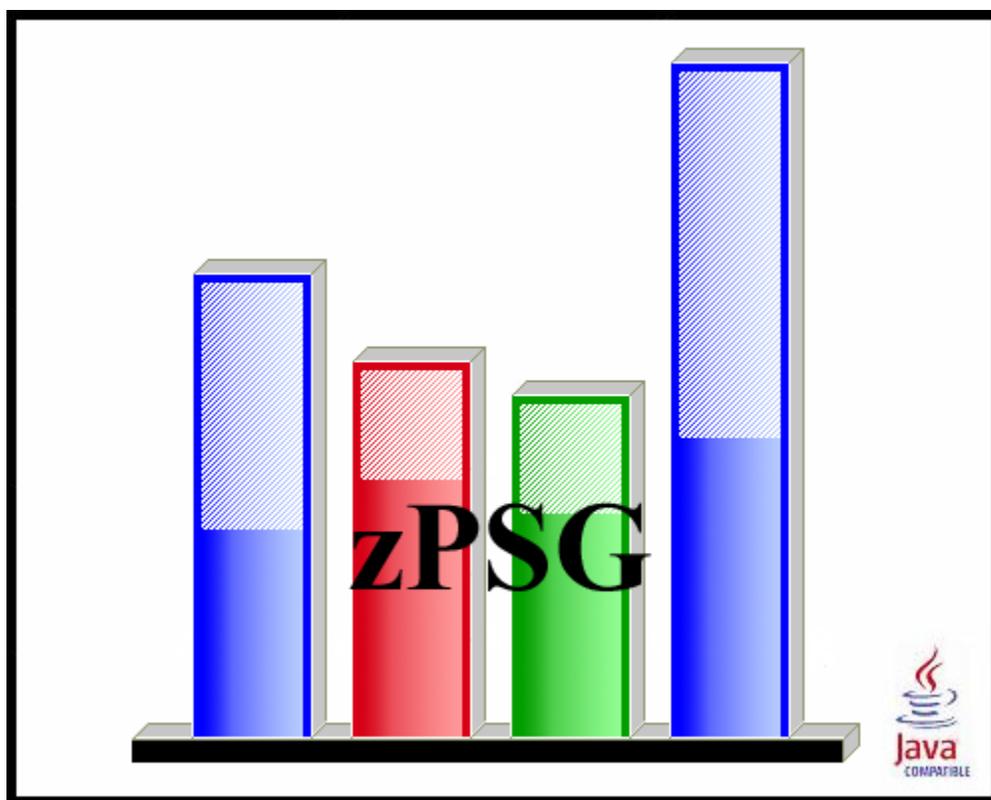


Processor Selection Guide  
for  
IBM System z

**zPSG**  
**User's Guide**  
for  
**WebSphere MQ**



© IBM Corporation – 2010, 2013

Version 5.3  
zPSG WMQ UG V53 2013c01.doc  
March 27, 2013

**The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.**

Multiprise*	Parallel Sysplex*
CICS*	RMF
DB2*	S/390*
e-business logo*	WebSphere*
Enterprise Storage Server	VSE/ESA
IBM*	VM/ESA*
IBM ^	z/VSE
IBM logo*	z/OS*
IBM System z9	z/VM*
IMS	zSeries
LSPR	IBM System z*
IBM System z10	IBM zEnterprise System
IBM zEnterprise 196 (z196)	IBM zEnterprise 114 (z114)
IBM zEnterprise EC12 (zEC12)	

\* Registered trademarks of IBM Corporation

**The following are trademarks or registered trademarks of other companies.**

Linux is a registered trademark of Linus Torvalds.

Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries

Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.

InstallShield Professional is a trademark of InstallShield Software Corporation

All other products may be trademarks or registered trademarks of their respective companies.

## WebSphere MQ

This tool provides estimates of System z processor capacity for applications using WebSphere MQ (WMQ) V7.1 on z/OS and Linux for System z.

See the [WMQ Glossary of Terms](#) for a definition of terms used in describing WMQ applications.

WebSphere MQ is a transport layer between cooperating applications. These cooperating applications can be on homogeneous or heterogeneous platforms including z/OS, z/Linux, xSeries/Linux, pSeries/Linux, AIX, Windows, iSeries, HP/UX, and Solaris. There are two principle Qualities of Service that use Persistent and non-Persistent messages. Persistent messages will survive planned and unplanned system outages while non-persistent messages do not survive Queue Manager restarts.

WebSphere MQ protocols provide rich support for applications:

- The Message Queue Interface (MQI) is supported in several programming languages.
- The point-to-point (including request/reply and client/server) and publish/subscribe application communication models are supported.
- The Java Message Service (JMS) is supported as an extension to the Queue manager and can interoperate with MQI applications.

Messages can be processed inside or outside syncpoint. Syncpoints are Logical Units of work that either all happen or none happens. In this sizing exercise, all Persistent messages are processed inside syncpoint so a single message is produced by an MQI program by using MQPUT, MQCMIT. This provides superior performance especially when many applications are processing messages from a common queue because of the recovery logic built into the MQ logger. Non-Persistent messages are processed outside syncpoint since these messages are transmitted as fast as possible.

## How To Do a WMQ Sizing

When you select WMQ sizing support from the **Product Selection** window to begin a new sizing, the **WMQ Application Definition** input window is presented.

The general approach to sizing WMQ applications on System z is to determine which of the pre-defined scenario profiles best represent the customer's application and provide an average scenario execution rate per second during a peak interval for each of these. Refer to [Scenario Descriptions](#) for a description of the scenario profiles that can be included in a sizing. The **Profile** button for each scenario can be used to display a detailed description of the processing included in that scenario. In addition to the scenario execution rate, also provide the message persistence, request message size, reply message size, and the application bindings (Linux only). Default values are provided for each of these input fields.

For Linux, the Local Queue Manager scenario can use Shared, Trusted, or Isolated binding for the Requester and Responder when they connect to the Queue Manager. The Distributed Queuing scenario applications also use these options for the Requester and Responder but have an additional option of the channel binding. This can be 'Normal' if the channels use 'Shared' binding or 'Fastpath' if MQIBINDTYPE = FASTPATH is specified in the Windows Registry or unix QM.INI file. Client Channel scenario applications are specified as Normal or FASTPATH channel followed by the binding of the Responder application (Shared, trusted, isolated).

Normal applications use SHARED (default) or ISOLATED bindings with the application and Queue Manager in separate processes. SHARED bindings share some resources between applications and the Queue Manager while ISOLATED bindings do not share any resources.

Trusted and Fastpath bindings mean the application or channel runs as an extension of the Queue Manager. This can double the throughput of local Queue manager scenarios but can cause stability problems if applications misbehave. MQIBINDTYPE=Fastpath could cause problems if user exits misbehave.

The usual span of time for a peak interval is 15 minutes, and you want to specify the average transactions per second for that interval. Note that if you have statistics for the average transaction rate for prime shift or for a day or week, you might want to apply a peak-to-average multiplier factor to averages for long periods of time to arrive at an average rate for a 15 minute interval.

To see results, click on the **Summary Report**, **CPU Utilization** or **Transaction Rate** buttons in the **Reports and Capacity Projections** section of the window.

**Note:** The application window images shown in this user guide have been provided as a representation of the windows the user will see when using **zPSG** but there may be minor differences from the current version of **zPSG**, such as version numbers and dates.

## **Scenario Descriptions**

This section provides a list of the pre-defined scenarios that can be included in a sizing. A scenario is included in a sizing by providing an execution rate greater than zero. The message persistence, request message size, reply message size, and application bindings (Linux only) should also be specified.

There are One-Way, Request-Reply, and Publish/Subscribe messaging scenarios for System z. The pre-defined scenarios for z/OS support all three types of messaging. The pre-defined scenarios for Linux on System z support Request-Reply and Publish/Subscribe messaging only.

### **One-Way Messaging (z/OS only)**

#### **Local Queue Manager (One-Way)**

- Sender application puts msg to common input queue on local queue mgr.
- Target application gets msgs from common input queue

#### **Client Channels (One-Way)**

- Sender application puts msg to common input queue over client channel
- Target application gets msgs from common input queue over client channel

#### **Shared Queues (One-Way)**

- Sender application puts msg to common shared queue
- Target application gets msgs from a shared queue

### **Request-Reply Messaging**

#### **Local Queue Manager**

- Requester application puts msg to common input queue on local queue mgr, holds on to msg identifier returned in the msg descriptor. Requester application then waits for a reply to arrive on common reply queue.
- Responder application gets msgs from common input queue & places a reply to common reply queue. Queue manager copies over msg identifier from request msg to correlation identifier of reply msg.
- Requester application gets a reply msg from common reply queue using msg identifier held from when request msg was put to the common input queue, as the correlation identifier in msg descriptor.

#### **Distributed Queuing**

- Requester application puts msg to local definition of remote queue located on system image of responder & holds on to msg identifier returned in msg descriptor. Requester application then waits for a reply to arrive on local queue.

- Msg channel agent takes msgs off the channel & places them on common input queue of MQ on responder system image.
- Responder application gets msgs from common input queue, & places a reply to queue name extracted from msg descriptor (name of local definition of remote queue located on requester system image). Queue mgr copies over msg identifier from request msg to correlation identifier of reply msg.
- Msg channel agent takes msgs off transmission queue & sends them over channel to MQ of requester system image.
- Requester application gets a reply from local queue. Requester application uses msg identifier held from when request msg was put to local definition of remote queue, as correlation identifier in msg descriptor.

### **Client Channels**

- Requester application puts request msg (over client channel) to common input queue, holds on to msg identifier returned in the msg descriptor. Requester application then waits for a reply to arrive on common reply queue.
- Responder application gets msgs from common input queue & places a reply to common reply queue. Queue mgr copies over the msg identifier from request msg to correlation identifier of reply msg as correlation identifier in msg descriptor.
- Requester application gets reply msg (over client channel) from common reply queue. Requester application uses msg identifier held from when request msg was put to the common input queue as correlation identifier in msg descriptor.

### **Shared Queues (z/OS only)**

- Requester application puts msg to shared input queue, holds on to msg identifier returned in the msg descriptor, & waits.
- Responder application gets msgs from shared input queue & places reply to shared reply queue. Application copies over msg identifier from request msg to correlation identifier of reply msg.
- Requester application gets reply msg from shared reply queue. Requester application uses msg identifier held from when request msg was put to the common input queue as correlation identifier in msg descriptor.

## **Publish/Subscribe Messaging**

### **Publish/Subscribe**

- Single Publisher, Many Subscribers (1:N) Scenario using Topic-based routing
- All subscribers used unique subscriber queues. Persistent subscribers received five messages in each transaction.
  1. A publisher publishes a message to the single topic.
  2. Each subscriber then receives the message.
- This test case provides asynchronous messaging since there is no connection between the number of messages in the system and the number of publishers or subscribers. The publisher publishes the next message without any 'think' time. Message count is the number of published messages plus those consumed by the subscribers.

## Linux under z/VM

**Define z/VM Environment**

Define z/VM environment or user defined z/VM cost factor.  
If both are entered, only the user defined z/VM cost factor will be used.

**z/VM Environment**

Number of Linux guests being sized: 1

Average number of virtual CPs per guest: 1.0

Default

**User Defined z/VM Cost Factor**

Cost Factor: 0.0 %

Default

Return Cancel Default All

Define z/VM environment.

This window is displayed when the **Linux deployed as a guest under z/VM** button is clicked on the primary definition window for the application being sized.

When Linux is deployed as guest under z/VM the CPU capacity requirement for the workload increases based on a number of factors including the workload characteristics, number of Linux guests being sized, and the average number of virtual CPs assigned to each guest.

The user can define the z/VM environment and let the tool estimate the z/VM cost factor or they can specify the user defined z/VM cost factor. When both are entered, the user defined z/VM cost factor will be used to calculate the capacity requirement with z/VM.

## **z/VM Environment**

### **Number of Linux guests being sized**

This field is automatically assigned based on the number of guests being sized. When sizing a single application the value will always be 1. This number varies in an aggregation sizing based on the number of applications assigned to the concurrency set being sized. This field is display only and can't be changed by the user.

### **Average number of virtual CPs per guest**

Average number of virtual CPs assigned to guest(s) being sized. Default value is 1.0

## **User Defined z/VM Cost Factor**

### **Cost Factor**

User defined z/VM cost factor specified as a percentage. Default value is 0.0%

Note: When Linux is deployed as a single guest under z/VM and the number of virtual CPs is equal to the number of real CPs, you should increase the CPU capacity requirement by 9% for WMQ.

## **Push Buttons**

Click the **Return** button to return to the calling window.

Click the **Default All** button to set all the values on this window to the default setting.

Click the **Cancel** button to return to the calling window without saving any changes.

When Linux is deployed under z/VM with multiple guests involved, a detailed z/VM sizing should be done using the **zVM-Planner** tool, described on the next page. The capacity requirement for each Linux guest, determined from **zPSG**, will be needed as an input metric. For sizing Linux guests with workload environments that are not yet supported in **zPSG**, contact Techline for sizing assistance.

**zPSG** application sizings are generally done for peak period activity. When many Linux guests are active under a single z/VM image, it is likely that the individual guest peaks do not occur at the same time. Therefore an opportunity exists for complementary peaks, thus lessening the overall z/VM capacity requirement. The **zVM-Planner** tool can help with this assessment.

You can also request sizing assistance for z/VM from Techline, using the appropriate URL:

For IBMers: <http://w3-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS2881>

For BPs: <http://www.ibm.com/partnerworld/techline>

## **z/VM Planner for Linux Guests on IBM System z Processors**

**zVM-Planner** is a PC-based productivity tool under Windows XP or Vista, designed to provide capacity planning insight for IBM System z processors running various Linux applications as guests under z/VM. Capacity results are based on analysis of a variety of benchmarks, both Linux native and Linux under z/VM. The tool is generic concerning software release levels, generally applying to z/VM v5.1 and later.

**zVM-Planner** input consists primarily of VM guest definitions and capacity requirements for each intended Linux application (a variety of Linux applications are supported). The expected capacity requirement for each Linux guest (a required input) can usually be obtained using the companion **zPSG** sizing tool or Techline assistance. The combined guest capacity requirement is then determined for optimally complementary peaks and for totally concurrent peaks. The degree of peak concurrency is user-selectable between these values. The resulting capacity requirement is combined with that of VM to support the entire complement of guests. Another companion tool, **zPCR**, can then be used to identify a processor and partition that can accommodate the VM image. All capacity values must be relative to a Reference-CPU setting that is common for all tools involved.

For scenarios where new Linux guests are to be added to an existing VM image, a **zVM-Planner** model of the existing VM guest configuration should first be built. The new Linux guests can then be added to determine the overall VM capacity requirement.

For scenarios where a VM image is to be added to an existing host processor, a **zPCR** model of the existing LPAR configuration should first be built. A partition that can deliver the capacity required by the VM image can then be added. **zPCR** can help assess any processor upgrade that may be necessary to accommodate the VM image.

Several guest metrics are available to help balance how the overall capacity will be distributed, including the number of VCPs (virtual CPUs), and Share and Capping assignments.

Results are presented in tables and graphs that can be captured for documentation purposes. **zVM-Planner** studies can be saved for future reference. Both a User's Guide and integrated context sensitive help are included.

**IBM employees** can obtain **zVM-Planner** and other CPS tools via the Intranet

[w3.ibm.com/support/america/wsc/cpsproducts.html](http://w3.ibm.com/support/america/wsc/cpsproducts.html)

**IBM Business-Partners** can obtain **zVM-Planner** and other CPS tools via the Internet

[partners.boulder.ibm.com/src/atmastr.nsf/WebIndex/PRS1762](http://partners.boulder.ibm.com/src/atmastr.nsf/WebIndex/PRS1762)

For questions concerning CPS tools, contact Capacity Planning Support:

- Lotus Notes: [CPS\\_PC\\_Tools/Gaithersburg/IBM@IBMUS](mailto:CPS_PC_Tools/Gaithersburg/IBM@IBMUS)
- E-mail: [cpstools@us.ibm.com](mailto:cpstools@us.ibm.com)

## WMQ Application Definition for Linux

zPSG - WMQ Application Definition [untitled]

File Help

**IBM System z Processor Sizing**  
**WebSphere MQ 7.1 on Linux**

Customer:  Application name:

Scenario Descriptions

Request-Reply Messaging	Message Pairs per Second	Message Persistence	Request Msg Size (K)	Reply Msg Size (K)	Application Bindings	
<b>A: Local Queue Manager</b>	<input type="text" value="0.00"/>	Non-Persistent <input type="button" value="v"/>	<input type="text" value="4"/>	<input type="text" value="4"/>	Shared / Shared <input type="button" value="v"/>	<input type="button" value="Profile"/>
<b>B: Distributed Queuing</b>	<input type="text" value="0.00"/>	Non-Persistent <input type="button" value="v"/>	<input type="text" value="4"/>	<input type="text" value="4"/>	Shared / Normal / Shared <input type="button" value="v"/>	<input type="button" value="Profile"/>
<b>C: Client Channels</b>	<input type="text" value="0.00"/>	Non-Persistent <input type="button" value="v"/>	<input type="text" value="4"/>	<input type="text" value="4"/>	Normal / Shared <input type="button" value="v"/>	<input type="button" value="Profile"/>

Publish/Subscribe Messaging	Message Pairs per Second	Message Persistence	Interface	Location	Number of Subscribers	
<b>D: Publish/Subscribe</b>	<input type="text" value="0.00"/>	Non-Persistent <input type="button" value="v"/>	JMS <input type="button" value="v"/>	Local <input type="button" value="v"/>	<input type="text" value="10"/>	<input type="button" value="Profile"/>

Reports and Capacity Projections

Linux deployed as guest under z/VM

characterizing the average transaction

to support peak transaction rate

support within SDP of  %

[Enter transaction information.](#)

This window is displayed when the **PSG** button is clicked on the **Product Selection** window when **WebSphere MQ** has been selected for Linux.

**Note: A transaction rate greater than zero must be specified for at least one scenario profile to get a sizing estimate.**

## WMQ Application Definition for z/OS

zPSG - WMQ Application Definition [untitled]

File Help

**IBM System z Processor Sizing**  
**WebSphere MQ 7.1 on z/OS**

Customer :  Application name :

Scenario Descriptions

One-Way Messaging			Messages per Second	Message Persistence	Message Size (K)	Msgs Per Commit	Number of Queue Mqs
A: Local Queue Manager	<input type="text" value="0.00"/>	Non-Persistent	<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="button" value="Profile"/>
B: Client Channels	<input type="text" value="0.00"/>	Non-Persistent	<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="button" value="Profile"/>	
C: Shared Queues	<input type="text" value="0.00"/>	Non-Persistent	<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="button" value="Profile"/>	

Request-Reply Messaging			Messages per Second	Message Persistence	Message Size (K)	Reply Msg Size (K)	Msgs Per Commit	Number of Queue Mqs
D: Local Queue Manager	<input type="text" value="0.00"/>	Non-Persistent	<input type="text" value="2"/>	<input type="text" value="2"/>	<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="button" value="Profile"/>
E: Distributed Queuing	<input type="text" value="0.00"/>	Non-Persistent	<input type="text" value="2"/>	<input type="text" value="2"/>	<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="button" value="Profile"/>
F: Client Channels	<input type="text" value="0.00"/>	Non-Persistent	<input type="text" value="2"/>	<input type="text" value="2"/>	<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="button" value="Profile"/>
G: Shared Queues	<input type="text" value="0.00"/>	Non-Persistent	<input type="text" value="2"/>	<input type="text" value="2"/>	<input type="text" value="2"/>	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="button" value="Profile"/>

Publish/Subscribe Messaging			Messages per Second	Message Persistence	Message Size (K)	Number of Subscribers
H: Publish/Subscribe	<input type="text" value="0.00"/>	Non-Persistent	<input type="text" value="2"/>	<input type="text" value="10"/>	<input type="text" value="10"/>	<input type="button" value="Profile"/>

Reports and Capacity Projections

characterizing the average transaction

to support peak transaction rate

support within SDP of  %

[Enter transaction information.](#)

This window is displayed when the **PSG** button is clicked on the **Product Selection** window when **WebSphere MQ** has been selected for z/OS

**Note:** A transaction rate greater than zero must be specified for at least one scenario profile to get a sizing estimate.

## Description of Input Fields

### Menu bar

#### File

<b>New</b>	Start a new study. Sets all fields to initialization values.
<b>Load</b>	Load a previously saved study
<b>Save</b>	Save the current study
<b>Save as</b>	Save the current study as a new file
<b>Exit</b> (Ctrl-E)	Exit window and return to the <b>Product Selection</b> window
<b>Exit zPSG</b>	Terminate <b>zPSG</b> execution (Ctrl-Q). Exit zPSG can also be invoked from the Exit zPSG button on the tool bar.

#### Help

<b>Context Help (F1)</b>	Help for this window
<b>About zPSG</b>	Product information

### Toolbar

- ? button** Click this button to go to Help for this window.
- Exit zPSG button** Click this button to terminate **zPSG** execution.

### Customer :

Input field, for documentation purposes, not required. If you want to save a copy of the sizing estimate, you can use this field to document which sizing it is.

### Application name :

Input field, for documentation purposes, not required. If you want to save a copy of the sizing estimate, you can use this field to document which sizing it is.

## **Input Fields and Buttons**

### **Messages per Second (One-Way Messaging)**

Specify a message per second execution rate for each of the scenarios that should be included in the sizing. Specify a value of zero if the scenario should not be included in the sizing. The default value for each of the scenarios is zero.

### **Message Pairs per Second (Request-Reply Messaging)**

Specify a message pairs per second execution rate for each of the scenarios that should be included in the sizing. Specify a value of zero if the scenario should not be included in the sizing. The default value for each of the scenarios is zero.

### **Message Persistence**

Select from Non-Persistent or Persistent for the scenarios that should be included in the sizing that support message persistence. The default value for each of the scenarios that support persistence is Non-Persistent.

### **Message Size (K) (One-Way Messaging)**

Specify the request message size in KB for each of the scenarios that should be included in the sizing. The default value for each of the scenarios is 4K

### **Request Message Size (K)**

Specify the average request message size in KB for each of the scenarios that should be included in the sizing. The default value for each of the scenarios is 4K

### **Reply Message Size (K)**

Specify the average reply message size in KB for each of the scenarios that should be included in the sizing. The default value for each of the scenarios is 4K

### **Application Bindings (Linux Only)**

Select the appropriate application bindings for the scenarios that should be included in the sizing. Local Queue Manager scenario applications can use Shared, Trusted, or Isolated binding for the Requester and Responder when they connect to the Queue Manager. The Distributed Queuing scenario applications also use these options for the Requester and Responder but have an additional option of the channel binding. This can be 'Normal' if the channels use 'Shared' binding or 'Fastpath' if MQIBINDTYPE = FASTPATH is specified in the Windows Registry or unix QM.INI file. Client Channel scenario applications are specified as Normal or FASTPATH channel followed by the binding of the Responder application (Shared, trusted, isolated).

Normal applications use SHARED (default) or ISOLATED bindings with the application and Queue Manager in separate processes. SHARED bindings share some resources between applications and the Queue Manager while ISOLATED bindings do not share any resources.

### **Number of Subscribers**

Specify the number of subscribers between 10 and 100. This input field is only used for the Publish/Subscribe scenario. The default value is 10.

### **Interface (Publish/Subscribe for Linux Only)**

Click on the drop-down icon to choose one of the following options:

1. JMS (default)
2. CMQI

### **Location (Publish/Subscribe for Linux Only)**

Click on the drop-down icon to choose one of the following options:

1. Local (default)
2. Remote

### **Profile button**

Click this button to view a detailed description of the processing for the scenario associated with this button. Each scenario has a Profile button.

Note: A short description for each scenario is displayed in a tool tip when the mouse pointer is over the name of the scenario.

## **Reports and Capacity Projections**

This section provides buttons to view output windows with summary reports and capacity projections.

### **Summary Report button**

Click this button to view a summary of the input assumptions for the sizing and a breakdown of the CPU/transaction among the transaction profiles included in the sizing.

### **CPU Utilization button**

Click this button to see an output window with estimates of processor utilization for all System z processors supported in zPSG.

### **Transaction Rate button**

Click this button to see an output window with estimates of transaction rates that can be supported on all System z processors supported in zPSG. You can also see the transaction rates that can be supported within a Saturation Design Point (SDP) specified for the processors.

### **SDP %**

#### **Description**

Input field, numeric, valid range is 1 to 100.

SDP stands for Saturation Design Point. This is a classic capacity planning concept which allows you to examine the amount of workload than can be supported in less than the full capacity of the processor model. It applies to the Transaction Rate output window and enables you to determine how much work can fit into a processor that is already being used for other applications.

#### **Default**

The default is 90%.

### **Return button**

Click this button to return to the ***Product Selection*** window.

### **Reference-CPU button**

Click this button to go to a window to change the System z processor used as a basis for capacity ratings. See the Reference-CPU section in the zPSG User's Guide for information about this setting.

### **Linux deployed as a guest under z/VM button**

**For Linux only**

Click this button for important sizing considerations when Linux is deployed as a guest under z/VM. See [Linux under z/VM](#) for details about these sizing considerations.

## WMQ Application Activity Summary

Scenario	Single Capacity	Scenario Rate / sec	Total Capacity	CPU Distribution
<b>Request-Reply Messaging</b>				
1. A: Local Queue Manager	0.0587	100.00	5.875	12.5%
2. B: Distributed Queuing	0.0995	50.00	4.976	10.6%
3. C: Client Channels	0.1715	7.00	1.201	2.6%
<b>Publish/Subscribe Messaging</b>				
4. D: Publish/Subscribe	0.3480	100.00	34.800	74.3%
<b>Load overall Total:</b>	<b>0.1823</b>	<b>257.00</b>	<b>46.852</b>	<b>100%</b>

This window is displayed when the **Summary Report** button is clicked on the primary **WMQ Application Definition** window. It shows a breakdown of the CPU per transaction for the various pre-defined transactions included in the sizing.

### Menu bar

#### File

##### Output

Write contents to a flat (PRN) file.

##### Copy

Write contents to Window's clipboard

#### Graph

Generates a pie chart showing the distribution of application activity

#### Help

**Context Help (F1)** Help for this window

**About zPSG** Product information

## **Toolbar**

### **1st button**

Click this button to send sizing information to a PRN file for processing outside of zPSG.

### **2nd button**

Click this button to send sizing information to the clipboard, so that you can copy it into a note or other document.

### **? button**

Click this button to go to Help for this window.

## **Scenario column**

Lists the pre-defined scenarios available on the primary **WMQ Application Definition** window.

## **Single Scenario Capacity column**

Reflects the amount of CPU (as represented by the Capacity Rating) for each scenario, and at the bottom for the average transaction.

## **Scenario Rate / sec column**

Reflects the number of scenarios completed per second for each of the pre-defined scenarios as specified on the **WMQ Application Definition** window.

## **Total Tran Capacity column**

Reflects the amount of CPU (as represented by the Capacity Rating) for each scenario multiplied by the transaction rate specified in the **Scenario Rate / sec** column, and at the bottom for all transactions.

## **CPU Distribution column**

Shows the percentage of the CPU/transaction used by each scenario.

## **Percent of workload estimated to be eligible for zAAP Processing = 0% (z/OS only)**

No zAAP eligible processing expected for WebSphere MQ.

## **Push Buttons**

Click the **Return** button to return to the primary ***WMQ Application Definition*** input window.

Click the **Utilization Report** button to go to the ***WMQ Processor Capacity Projections - Processor Utilization*** output window.

Click the **Transaction Rate Report** button to go to the ***WMQ Processor Capacity Projections - Transaction Rate Supported*** output window.

Click the **Show Assumptions** button to see a list of the assumptions for the sizing in the ***WMQ Application Transaction Assumptions*** window.

## WMQ Transaction Assumptions

For this application ...  
the WMQ workload is characterized by the following assumptions:

<u>Request-Reply Messaging</u>		<u>Message Pairs per Second</u>	<u>Message Persistence</u>	<u>Request Msg Size (k)</u>	<u>Reply Msg Size (k)</u>	<u>Application Bindings</u>
1. A: Local Queue Manager		100.00	Non-Persistent	4	4	Shared / Shared
2. B: Distributed Queuing		50.00	Non-Persistent	4	4	Shared / Normal / Shared
3. C: Client Channels		7.00	Non-Persistent	4	4	Normal / Shared
<u>Publish/Subscribe Messaging</u>				<u>Interface</u>	<u>Location</u>	<u>Number of Subscribers</u>
4. D: Publish/Subscribe		100.00	Non-Persistent	JMS	Local	10

Return

This window is displayed when the **Show Assumptions** button is clicked on the **WMQ Application Activity Summary** window.

All assumptions as listed will be included when generating output for the **Summary** window.

Click the **Return** button to return to the **WMQ Application Activity Summary** window.

## WMQ Processor Utilization

### WMQ Processor Capacity Projections

IBM System z Processor Sizing  
**WebSphere MQ 7.1 on Linux**  
Transaction Rate Supported  
IFL CPs

Capacity based on z/OS-1.13 LSPR Data (08/28/2012) using "Average/L" workload  
Capacity basis: 2094-701 @ 593 MIPS for a shared single partition configuration

Processor	Feature	Flag	MSU	Capacity Rating	SDP = 90% ETR	SDP = 100% ITR
<b>zEnterprise EC12/700</b>						
2827-7xx I1	1W IFL	=		1,621	8,608	9,565
2827-7xx I2	2W IFL	=		3,063	16,269	18,077
2827-7xx I3	3W IFL	=		4,468	23,731	26,368
2827-7xx I4	4W IFL	=		5,836	30,999	34,444
2827-7xx I5	5W IFL	=		7,165	38,056	42,285
2827-7xx I6	6W IFL	=		8,455	44,909	49,899
2827-7xx I7	7W IFL	=		9,708	51,563	57,292
2827-7xx I8	8W IFL	=		10,924	58,024	64,471
2827-7xx I9	9W IFL	=		12,106	64,297	71,441
2827-7xx I10	10W IFL	=		13,252	70,389	78,210
2827-7xx I11	11W IFL	=		14,366	76,304	84,782
2827-7xx I12	12W IFL	=		15,447	82,047	91,163
2827-7xx I13	13W IFL	=		16,504	87,658	97,398
2827-7xx I14	14W IFL	=		17,536	93,141	103,490
2827-7xx I15	15W IFL	=		18,545	98,497	109,441
2827-7xx I16	16W IFL	=		19,530	103,730	115,255
2827-7xx I17	17W IFL	=		20,492	108,842	120,936
2827-7xx I18	18W IFL	=		21,433	113,837	126,486
2827-7xx I19	19W IFL	=		22,352	118,717	131,908
2827-7xx I20	20W IFL	=		23,249	123,485	137,206
2827-7xx I21	21W IFL	=		24,138	128,205	142,451
2827-7xx I22	22W IFL	=		25,018	132,878	147,643
2827-7xx I23	23W IFL	=		25,889	137,505	152,783

Reference-CPU set to the IBM 2094-701 rated at 593  
Processors in view = 233; In listbox = 233; Selected = 000

Return

To view FLAG information, place pointer on a processor flag indicator

This window is displayed when the **Utilization** button is clicked on the **WMQ Application Definition** window or the **Utilization Report** button is clicked on the **WMQ Application Activity Summary** window.

**Menu bar****File****Output** Write report contents to a flat (PRN) file.**Copy** Write report contents to Window's clipboard**Graph** (for processors currently selected in table)**Capacity** Generate bar graph depicting capacity values**Utilization** Generate bar graph showing utilization on selected processors**Help****Context Help (F1)** Help for this window**About zPSG** Product information**Toolbar****1st button**

Click this button to send sizing information to a PRN file, for processing outside of zPSG.

**2nd button**

Click this button to send sizing information to the clipboard, so that you can copy it into a note or other document.

**? button**

Click this button to go to Help for this window.

**Table*****Processor column***

A list of all processor models supported in zPSG

***Feature column*****For z/OS & Linux**Using the General Purpose CPs option under Table View, a designation of how many general purpose processing engines (CPs) for this entry. For example, 4W ("W" is short for "way") indicates 4 CPs or engines. Also see ***Flag*** column below.***Feature column*****For Linux only**Using the IFL CPs option under Table View, a designation of how many IFL engines for this entry. For example, 4W IFL ("W" is short for "way") indicates 4 IFL engines. Also see ***Flag*** column below.***Flag column***

If you place your cursor on a row in this column, an explanatory message about the System z model designation and the number of CP or IFL engines for the entry.

***MSU column***

Only for the General Purpose CPs Table View (does not apply to IFLs). Shows the MSU rating assigned to the number of CP engines for this entry.

**Capacity Rating column**

The capacity ratings reflect the relative capacity of each processor table entry to the reference-CPU and its capacity rating assigned on the Reference-CPU window. When **zPSG** is started the reference-CPU will be set to a 2094-701 (a z9 EC/700 processor with 1 general purpose CP) with a capacity rating of 602 MIPS.

**Projected Utilization column**

Shows the estimated CPU% for each processor entry in the table, based on the sizing inputs specified for the pre-defined transactions. This is the primary output for a sizing.

**# Servers Required column**

If the estimated CPU% is greater than 100% (and therefore cannot fit on the processor), this column reflects the number of these models that would be needed to accommodate the load.

**Table View Options Box**

Click a radio button in each section to customize the processor entries shown in the table:

- **General Purpose CPs** shows entries with some number of general CP engines
- **IFL CPs** shows entries with some number of IFL engines (for Linux only)
- **Family** shows all processor models for the family selected (Default)
- **All** shows all processor models supported in zPSG
- **Within SDP** shows all models that can accommodate the load within the Saturation Design Point
- **Selected** shows only selected models. Models are selected by clicking on the entry while holding down the Ctrl key on your keyboard.

**Return button**

Click this button to return to the primary **WMQ Application Definition** window.

## WMQ Transaction Rate Supported WMQ Processor Capacity Projections

zPSG - WMQ Processor Capacity Projections [untitled]

File Graph Help

IBM System z Processor Sizing  
**WebSphere MQ 7.1 on Linux**  
Transaction Rate Supported  
IFL CPs  
Capacity based on z/OS-1.13 LSPR Data (08/28/2012) using "Average/L" workload  
Capacity basis: 2094-701 @ 593 MIPS for a shared single partition configuration

Processor	Feature	Flag	MSU	Capacity Rating	SDP = 90% ETR	SDP = 100% ITR
<b>zEnterprise EC12/700</b>						
2827-7xx I1	1W IFL	=		1,621	8,608	9,565
2827-7xx I2	2W IFL	=		3,063	16,269	18,077
2827-7xx I3	3W IFL	=		4,468	23,731	26,368
2827-7xx I4	4W IFL	=		5,836	30,999	34,444
2827-7xx I5	5W IFL	=		7,165	38,056	42,285
2827-7xx I6	6W IFL	=		8,455	44,909	49,899
2827-7xx I7	7W IFL	=		9,708	51,563	57,292
2827-7xx I8	8W IFL	=		10,924	58,024	64,471
2827-7xx I9	9W IFL	=		12,106	64,297	71,441
2827-7xx I10	10W IFL	=		13,252	70,389	78,210
2827-7xx I11	11W IFL	=		14,366	76,304	84,782
2827-7xx I12	12W IFL	=		15,447	82,047	91,163
2827-7xx I13	13W IFL	=		16,504	87,658	97,398
2827-7xx I14	14W IFL	=		17,536	93,141	103,490
2827-7xx I15	15W IFL	=		18,545	98,497	109,441
2827-7xx I16	16W IFL	=		19,530	103,730	115,255
2827-7xx I17	17W IFL	=		20,492	108,842	120,936
2827-7xx I18	18W IFL	=		21,433	113,837	126,486
2827-7xx I19	19W IFL	=		22,352	118,717	131,908
2827-7xx I20	20W IFL	=		23,249	123,485	137,206
2827-7xx I21	21W IFL	=		24,138	128,205	142,451
2827-7xx I22	22W IFL	=		25,018	132,878	147,643
2827-7xx I23	23W IFL	=		25,889	137,505	152,783

Reference-CPU set to the IBM 2094-701 rated at 593  
Processors in view = 32; In listbox = 32; Selected = 001

Return

To view FLAG information, place pointer on a processor flag indicator

Table View

General Purpose CPs  IFL CPs

Family zEC12/700

All  Within SDP  Selected

This window is displayed when the **Transaction Rate** button is clicked on the **WMQ Application Definition** window or the **Transaction Rate Report** button is clicked on the **WMQ Application Activity Summary** window.

**Menu bar****File****Output** Write report contents to a flat (PRN) file.**Copy** Write report contents to Window's clipboard**Graph** (for processors currently selected in table)**Capacity** Generate a bar graph depicting capacity values**ETR** Generate bar graph showing transaction rate supported at SDP**ITR** Generate bar graph showing maximum transaction rate supported**Help****Context Help (F1)** Help for this window**About zPSG** Product information**Toolbar****1st button**

Click this button to send sizing information to a PRN file, for processing outside of zPSG.

**2nd button**

Click this button to send sizing information to the clipboard, so that you can copy it into a note or other document.

**? button**

Click this button to go to Help for this window.

**Table*****Processor column***

A list of all processor models supported in zPSG

***Feature column*****For z/OS & Linux**Using the General Purpose CPs option under Table View, a designation of how many general purpose processing engines (CPs) for this entry. For example, 4W ("W" is short for "way") indicates 4 CPs or engines. Also see **Flag** column below.***Feature column*****For Linux only**Using the IFL CPs option under Table View, a designation of how many IFL engines for this entry. For example, 4W IFL ("W" is short for "way") indicates 4 IFL engines. Also see **Flag** column below.***Flag column***

If you place your cursor on a row in this column, an explanatory message about the System z model designation and the number of CP or IFL engines for the entry.

**MSU column**

Only for the General Purpose CPs Table View (does not apply to IFLs). Shows the MSU rating assigned to the number of CP engines for this entry.

**Capacity Rating column**

The capacity ratings reflect the relative capacity of each processor table entry to the reference-CPU and its capacity rating assigned on the Reference-CPU window. When **zPSG** is started the reference-CPU will be set to a 2094-701 (a z9 EC/700 processor with 1 general purpose CP) with a capacity rating of 602 MIPS.

**SDP= xx % -- ETR column**

Shows the transaction rate for the average transaction that can be supported within the Saturation Design Point specified on the **WMQ Application Definition** window (the default SPD is 90%). ETR stands for External Throughput Rate, which is a standard System z term for transaction rate.

**SDP=100% -- ITR column**

Shows the transaction rate for the average transaction that can be supported at 100% CPU utilization. ITR stands for Internal Throughput Rate, which is a standard System z term indicating the throughput that can be achieved at 100% CPU. ITR is computed by dividing the ETR by the CPU% (expressed as a decimal). This is the way to correctly rate the processor capacity of each entry in the processor table for this workload (as opposed to MIPS ratings, which are generally erroneous).

**Table View Options Box**

Click a radio button in each section to customize the processor entries shown in the table:

- **General Purpose CPs** shows entries with some number of general CP engines
- **IFL CP's** shows entries with some number of IFL engines (for Linux only)
- **Family** shows all processor models for the family selected (Default)
- **All** shows all processor models supported in zPSG
- **Within SDP** shows all models that can accommodate the load within the Saturation Design Point
- **Selected** shows only selected models. Models are selected by clicking on the entry while holding down the Ctrl key on your keyboard.

**Return button**

Click this button to return to the primary **WMQ Application Definition** window.

## **WMQ Sizing Assistance**

Here are instructions for accessing the System z questionnaire from Techline. Note that on the Techline websites there are sizing questionnaires for distributed platforms in addition to System z. Be sure to use System z questionnaires for System z sizings. The questions and sizing methodologies are different from distributed platforms.

### **For IBMers:**

1. Obtain the latest copy of the WMQ sizing questionnaire for System z from the following website:
  - <http://w3-03.ibm.com/support/techline/global/swsz.html>
2. Submit a sizing request to Techline using the instructions found in the sizing questionnaire.

### **For Business Partners:**

1. Obtain the latest copy of the WMQ sizing questionnaire for System z via:
  - Phone: Call PartnerLine at 1-800-426-9990 (US and Canada)
  - Email: [pwcs@us.ibm.com](mailto:pwcs@us.ibm.com)
  - Online: <http://www.ibm.com/partnerworld/techline>
2. Submit a sizing request to Techline using the instructions found in the sizing questionnaire.

## **WMQ Glossary of Terms**

### **Bindings Mode Connection**

When a JMS connection is made in bindings mode, MQ JMS uses the Java Native Interface (JNI) to call the MQ Queue Manager directly rather than communicating over TCP/IP. This connection mode is much more efficient when the sender and receiver reside in the same image of z/OS. Connections that require TCP/IP are called TCP mode connections.

### **JMS**

Java Message Service. A peer to peer communication facility that can be used by software components or applications, usually in conjunction with MQ Series. This can double the CPU cost per message compared to MQI applications.

### **Local Mode**

In the context of JCA Connectors, local mode refers to a means of accessing CICS or IMS without using TCP/IP sockets. Local mode is generally more efficient since it is optimized to exploit the fact that the caller and callee are on the same system.

### **MQ Message**

A string of bytes that is meaningful to the applications that use it

### **MQ Queue**

A named data structure for holding messages until they are retrieved by an application. Multiple senders and receivers can be associated with a single queue.

### **MQ Queue Manager**

A named group of address spaces that run as a z/OS subsystem and manage the resources associated with WebSphere MQ. Applications connect to a Queue Manager using its name.

### **Non-Persistent Message**

A non-persistent message will be delivered as fast as possible provided there are no system failures. These messages are kept mainly in memory and will be purged from Queues when the Queue manager is restarted.

### **Persistent Message**

A persistent message is guaranteed to be delivered by MQ once and only once. It must be written to a file or a database to guarantee delivery.

### **Point-To-Point Messaging**

This messaging model enables the delivery of an MQ message to only one recipient, also called a consumer.

### **Publish/Subscribe Messaging**

This messaging model supports the delivery of an MQ message to multiple recipients called topic subscribers.

## **Queue Manager**

See [MQ Queue Manager](#).

## **TCP Mode (or Client Mode) Connection**

When a JMS connection is made in TCP mode, JMS uses TCP/IP to call the MQ Queue Manager rather than communicating over the Java Native Interface as it does in bindings mode. With TCP Mode Connections, the MQ Queue Manager does not have to be on the same server, or indeed the same platform.

## **Transacted Session**

This option is used to group a series of messages into an atomic unit of work. All messages in the work unit either succeed or fail. The application server commits the session. If the application server detects an error, it may roll back the transaction. The message is not actually sent until the transaction is committed. The next transaction begins after a call to either commit or rollback.