

CA Workload Automation ESP Edition

Resource Counter Propagation Enhancement

Release 11.4



This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and may not be disclosed by you or used for any purpose other than as may be permitted in (i) a separate agreement between you and CA governing your use of the CA software to which the Documentation relates; or (ii) a separate confidentiality agreement between you and CA.

Notwithstanding the foregoing, if you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2014 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

CA Technologies Product References

This topic lists the CA Technologies products that are referenced in this document.

Note: The names in parenthesis are the shorter product names that are used in the documentation.

- CA Workload Automation ESP Edition (CA WA ESP Edition)
- CA Workload Automation Service Governor ESP Edition (CA WA Service Governor ESP Edition)

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At <http://ca.com/support>, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at <http://ca.com/docs>.

Contents

Chapter 1: Overview	7
About This Enhancement	7
Chapter 2: Installation Guide Updates	9
Propagate Resource Counter Updates.....	9
COMMCHAN: Configure Resource Counter Sharing Between Master Subsystems.....	10
MANAGER: Specify Scheduler Subsystem for Communications	11
RESDEF: Define Resources.....	13
Chapter 3: Command Reference Guide Updates	21
MANAGER Command: Specify Scheduler Subsystem for Communications.....	21
RESDEF Command: Define Resources	22

Chapter 1: Overview

About This Enhancement

This enhancement enables specified CA WA ESP Edition master subsystems to propagate updates made to threshold enterprise resource counters. If a resource counter is updated on one master, resource counters with the same name on other specified masters are also updated.

The following are the CA WA ESP Edition documentation changes for this enhancement:

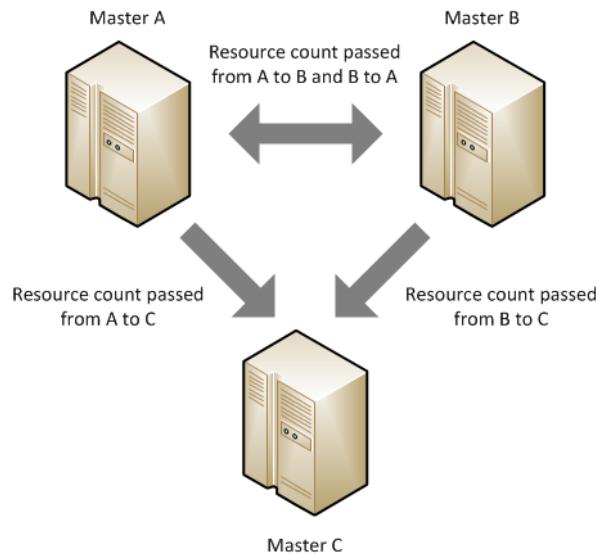
- [Propagate Resource Counter Updates](#) (see page 9)—New
- [COMMCHAN: Configure Resource Counter Sharing Between Master Subsystems](#) (see page 10)—New
- [MANAGER Initialization Parameter](#) (see page 11), [MANAGER Command](#) (see page 21)
 - Changed title and introduction
 - Added important notes to the NAME operand description.
 - Miscellaneous Changes:
 - Removed obsolete operand APPC
 - Combined and rewrote description for TCPIP and BIND operands.
- [RESDEF Initialization Parameter](#) (see page 13), [RESDEF Command](#) (see page 22)—Added RECEIVERS and SENDERS operands and an example at the end of the topic.

Chapter 2: Installation Guide Updates

Propagate Resource Counter Updates

You can configure CA WA ESP Edition master subsystems to propagate updates made to threshold enterprise resource counters. If a resource counter is updated on one master, resource counters with the same name on other specified masters are also updated.

You can propagate updates in both directions between masters or in one direction only. Here's an example:



Follow these steps:

1. Ensure that a MANAGER initialization parameter is coded on all the masters that must share threshold enterprise resource counter updates.
2. On the current master, code RECEIVERS, SENDERS, or both in the RESDEF initialization parameters for resources whose counters must be shared.
See an example in the RESDEF command documentation.
3. On the current master, code a COMMCHAN initialization parameter for each master specified in RECEIVERS and SENDERS in Step 2.
4. Repeat Steps 2 and 3 on all the other masters that must share resource counter updates.

COMMCHAN: Configure Resource Counter Sharing Between Master Subsystems

COMMCHAN specifies a master subsystem that can potentially share threshold enterprise resource counter updates with the current master. RESDEF initialization parameters or commands determine how the updates are shared.

On the current master, code COMMCHAN for each of the other masters that can share threshold enterprise resource counter updates. In the same way as the current master, code COMMCHAN initialization parameters on the other masters that share updates.

Where defined: Master, AGENTDEF data set

Syntax

```
COMMCHAN master_name
        {ADDRESS(scheduler_ip_addr)}
        {PORT(scheduler_port)}
```

master_name

Specifies the master subsystem name from the MANAGER initialization parameter on that master. The value must be unique for each COMMCHAN created.

Limits: Maximum of 16 characters

ip_addr

The TCP/IP address where the master runs. The address can be in dotted decimal notation (for an IPV4 address), comma-delimited hexadecimal notation (for an IPV6 address), or host name format.

port

The TCP/IP port number of the master.

Example: COMMCHAN Initialization Parameter

This example shows COMMCHAN initialization parameters in an AGENTDEF data set on master PROD1.

```
ENCRYPT KEYNAME(RESKEY) ALL
MANAGER NAME(PROD1) TCPIP
COMMCHAN PROD2 ADDRESS(32.1.1.1) PORT(7507)
COMMCHAN PROD3 ADDRESS(33.4.1.2) PORT(7507)
```

ENCRYPT specifies the key name for communication between the masters.

MANAGER specifies the current master name, PROD1. You specify PROD1 in a COMMCHAN on the other masters that can potentially share threshold enterprise resource counter updates with PROD1.

A COMMCHAN is coded for masters PROD2 and PROD3. Those masters can potentially share threshold enterprise resource counter updates with master PROD1.

Here is a sample AGENTDEF data set on master PROD2:

```
ENCRYPT KEYNAME(RESKEY) ALL  
MANAGER NAME(PROD2) TCPIP  
COMMCHAN PROD1 ADDRESS(31.1.1.1) PORT(7507)  
COMMCHAN PROD3 ADDRESS(33.4.1.2) PORT(7507)
```

Here is a sample AGENTDEF data set on master PROD3:

```
ENCRYPT KEYNAME(RESKEY) ALL  
MANAGER NAME(PROD3) TCPIP  
COMMCHAN PROD1 ADDRESS(31.1.1.1) PORT(7507)  
COMMCHAN PROD2 ADDRESS(32.4.1.2) PORT(7507)
```

MANAGER: Specify Scheduler Subsystem for Communications

The MANAGER initialization parameter specifies the current master subsystem for communications.

Note: Code MASTER before the declarations of agents or schedulers that the current master communicates with.

Where defined: Master, AGENTDEF data set

Syntax

```
MANAGER [NAME(mgrname | CENTRAL_MANAGER) ]  
[TCPIP[(ipaddress)] [BIND]]  
[ENCRYPT|ENCRYPT(x'nnnnnnnnnnnnnnnn')]  
|ENCRYPT KEYNAME(keyname)|NOENCRYPT]
```

NAME(*mgrname* | CENTRAL_MANAGER)

The name of the CA WA ESP Edition subsystem. The *mgrname* operand has a 16-character maximum.

Default: CENTRAL_MANAGER

Important! For agents, specify the same name in the communication.managerid_n parameter in the agentparm.txt file on your agent system.

Important! For sharing resource counter updates, specify the same name in a COMMCHAN initialization parameter on each master sharing with the current master.

TCPIP[(*ipaddress*)] [BIND]

Specifies the DNS host name or TCP/IP address of the CA WA ESP Edition master subsystem broadcast by the MGRADDR command.

BIND causes CA WA ESP Edition to bind to the IP address when it connects to an agent. BIND guarantees that all connections from the master to the agents have the same TCP/IP address.

Limits: *ipaddress* is in IPV4 or IPV6 format.

Default: If you omit *ipaddress* or the entire operand, CA WA ESP Edition issues the GETHOSTID command to get the address. BIIND works with the default address or an address that you specify.

Note: If *ipaddress* is invalid on the master, CA WA ESP Edition does not issue the GETHOSTID command and cannot communicate with the agent.

ENCRYPT

Enables encrypted communication with other schedulers. The key is specified in the ENCRYPT initialization parameter.

ENCRYPT KEYNAME(*keyname*)

Enables encrypted communication with other schedulers. The key is defined with the CRYPTKEY command.

ENCRYPT(x'nnnnnnnnnnnnnnnn')

Enables encrypted communication with other schedulers. The key must be an even number of hexadecimal characters up to 16 characters.

NOENCRYPT

Disables encrypted communication with other schedulers

Example: MANAGER Initialization Parameter

The following is an example of the MANAGER initialization parameter for the AGENTDEF data set:

MANAGER NAME (CM CENTRAL)

Example: Set Manager Address Depending on Host

The following example sets the manager name CM_ESP with two different TCP/IP addresses, depending on the host subsystem.

```
IF SYSNAME='SYSA' THEN DO  
MANAGER NAME(CM_ESP) TCPIP(10.1.15.1)  
ENDDO  
ELSE IF SYSNAME='SYSC' THEN DO  
MANAGER NAME(CM_ESP) TCPIP(10.1.15.3)  
ENDDO
```

Example: MANAGER in AGENTDEF

The following is a sample AGENTDEF with the MANAGER initialization parameter:

```
MANAGER NAME(CM_CENTRAL)
ENCRYPT KEYNAME(AGTKEY1) ALL
MAPUSER JDOE TO(PROD) AGENT(TORSUN1)
AGENT TORSUN1 ADDRESS(xxx.xx.xx.xx) PORT(9999) NOENCRYPT
AGENT TORNT1 ADDRESS(xxx.xx.xx.xx) PORT(9998)
AGENT TORNT2 ADDRESS(xxx.xx.xx.xx) PORT(9997) +
    ENCRYPT KEYNAME(TORNT2_KEY)
```

Example: Bind CA WA ESP Edition to an IP Address

In this example, the MANAGER initialization parameter specifies a DNS host name called SYSA and requests that CA WA ESP Edition bind to the TCP/IP address corresponding to SYSA when connecting to an agent:

```
MANAGER NAME(CM_CENTRAL) TCPIP(SYSA) BIND
```

RESDEF: Define Resources

Note: You can also issue the RESDEF initialization parameter as a command. For details about command syntax and operands, see the *Command Reference Guide*

The RESDEF initialization parameter allows you to define resource definitions.

Where defined: Master only

This initialization parameter has the following format:

```
RESDEF name ADD
    [ENTERPRISE|NODAL|GLOBAL |LOCAL]
    [THRESHOLD|RENEWABLE|DEPLETABLE]
    [NODE(xxx)][CPU(xxx)]
    [MAXIMUM(n)][AVAIL(0)]
    [DEVICE(xxxxx)|WLM(duration)|
        MONITOR(CPU) POLL(xx HOURS|xx MINUTES|xx SECONDS)]
    [GRAVITY|NOGRAVITY]
    [COMMENT(comment)]
    [RECEIVERS(rec1,rec2,...)]
    [SENDERS(sen1, sen2, ...)]
```

name

Indicates the resource name. Specify up to eight alphanumeric characters for default resources and up to 44 alphanumeric characters for other resources. The first character must be alphabetic or national. The resource name must not end with a period or have two consecutive periods. When using the SET operand, you can use wildcard characters.

ADD

Defines a new resource

ENTERPRISE

Indicates one resource counter controls the resource's availability across all nodes and CPUs

NODAL

Indicates one resource counter is maintained for each node

GLOBAL

Alias of NODAL

LOCAL

Indicates one resource counter is maintained for each CPU

THRESHOLD

Defines a threshold resource type

RENEWABLE

Defines a renewable resource type

DEPLETABLE

Defines a depletable resource type

NODE(*xxx*)

Refers to a node defined in the system topology. NODE limits LIST or SET to a specific node.

CPU(*xxx*)

Refers to a CPU defined in the resource topology. CPU limits LIST or SET to a specific CPU.

MAXIMUM(*n*)

Sets the maximum count of the resource you are defining. The default value is 0.

AVAIL(*n*)

Sets the available count of the resource you are defining. This value should not be manipulated, but left for CA WA ESP Edition to calculate.

DEVICE (xxxxx)

Indicates the IBM generic or esoteric device name that CA WA ESP Edition is to monitor

WLM(*duration*)

Specifies that CA WA ESP Edition obtains the number of unused service units over the specified duration from IBM WLM. CA WA ESP Edition obtains the number of unused service units before submitting the job. CA WA ESP Edition supports the following three durations:

- SUM60 — The unused service units are summed over the last 60 seconds.
- SUM180 — The unused service units are summed over the last 180 seconds.
- SUM600 — The unused service units are summed over the last 600 seconds.

Note:

- The WLM operand is only relevant if CA WA Service Governor ESP Edition is installed. If you code the WLM operand without CA WA Service Governor ESP Edition installed, CA WA ESP Edition ignores the WLM operand.
- If you code the WLM operand, you must code the SYSPLEX operand in the NODE command or initialization parameter when you define your system nodes.

MONITOR(CPU)

Specifies that CA WA ESP Edition asks all Agents to provide the percentage of CPU availability on the platforms they control at the frequency specified by the POLL operand

Note: The MONITOR operand is only relevant if the CA WA Service Governor ESP Edition is installed. If you use the MONITOR operand without CA WA Service Governor ESP Edition installed, CA WA ESP Edition ignores the MONITOR operand.

POLL(xx HOURS|xx MINUTES|xx SECONDS)

Specifies how often CA WA ESP Edition asks all Agents to provide the percentage of CPU availability on the platforms they control. xx is a number with a maximum corresponding to 24 hours whether in hours, minutes or seconds. If POLL specifies 0, CA WA ESP Edition does not ask Agents to provide the percentage of CPU availability on the platforms they control.

Note: The POLL operand is only relevant if the CA WA Service Governor ESP Edition is installed. If you use the POLL operand without CA WA Service Governor ESP Edition installed, the POLL operand is ignored.

GRAVITY

Indicates jobs are to be routed to other nodes the resource is available on

NOGRAVITY

Indicates jobs are not to be routed to other nodes a resource is available on

COMMENT (comment)

Used to add information about a particular resource

RECEIVERS(*rec1,rec2,...*)

Specifies the masters that can receive threshold enterprise resource count updates from the current master. The masters specified update their local resource count to match the count that was sent. Each master name in this operand must be specified in a COMMCHAN initialization parameter on the current master. You can use the hyphen and asterisk wildcard characters in master names.

SENDERS(*sen1,sen2,...*)

Specifies the masters that can send threshold enterprise resource count updates to the current master. The current master updates the local resource count to match the count that was sent. Each master name in this operand must be specified in a COMMCHAN initialization parameter on the current master. You can use the hyphen and asterisk wildcard characters in master names.

Define Real Resources

A real resource represents a hardware device such as tape drives. Since the CA WA ESP Edition master is only aware of the status of devices on the image it runs on, real resources only work in a single z/OS image. Define real resources as LOCAL and RENEWABLE and, using the MAX operand, identify the number of devices available to CA WA ESP Edition-submitted jobs:

```
RESDEF T3480 ADD LOCAL RENEWABLE MAX(5) DEVICE(3480)
```

The following example shows what appears when the above resource is listed:

```
RESDEF T3480 LIST
Resource T3480      Local Renewable
TORONTO  MVSA      Max=5 Avail=5 Real=5
```

CA WA ESP Edition ensures there are sufficient online, unallocated devices available to satisfy the current job's resource requirements.

For example, if job PAYJOB1 requires three units of a real resource called T3480, and CA WA ESP Edition sees three devices available, but only two devices are online and unallocated, CA WA ESP Edition waits for real resources and PAYJOB1 goes into a RESWAIT state.

Define Resources with WLM Operand (CA WA Service Governor ESP Edition only)

You must define an IBM WLM resource as LOCAL RENEWABLE.

To determine the MAX count, see the "CPU capacity table" in the *MVS planning: Workload Management* IBM manual. From this table, choose the number of service units per second for your CPU model and multiply it by either 60, 180 or 600 for the duration SUM60, SUM180 or SUM600 respectively. You can use the result or any higher number. If your installation has CPUs of different models, use the RESDEF SET command to adjust the counts on individual CPUs.

CA WA ESP Edition routes a job to one of the available systems based on the service units assigned to the job and other resources the job requests.

IBM WLM feedback is a measurement of recent but past activity; it is not a precise prediction of a CPU load by the time CA WA ESP Edition submits a job. The real value of a WLM resource is the number of unused service units. The number of unused service units can differ from the number of service units available for a job if the job service class is higher than the service class of the currently executing workload.

The workload is balanced across the CPUs and nodes within the same sysplex, as long as they are all part of the same sysplex group. The node is defined with the NODE command or initialization parameter (it must include a SYSPLEX operand).

Note: Specify the same duration for all CPUs and nodes. CA WA ESP Edition issues a warning message if you specify a different duration for a specific CPU or node.

Define Resources with MONITOR(CPU) Operand (CA WA Service Governor ESP Edition only)

You must define a MONITOR(CPU) resource as LOCAL RENEWABLE.

CA WA ESP Edition asks all Agents to provide the percentage of CPU availability on the platforms they control. The Agents provide this percentage at the frequency specified by the POLL operand for CPUs where the corresponding CPU initialization parameter includes an AGENT operand. You can limit the scope of the resource you define by using the NODE and CPU operands.

CA WA ESP Edition routes a job to one of the available systems based on the MONITOR(CPU) resources and other resources a job requests.

The workload is balanced among the CPUs that belong to the same node. The node is defined with the NODE command or initialization parameter.

Related Information

For information on how to use IBM WLM or Agent workload balancing, see the *User Guide*.

Example: Defining a local renewable resource

In the following example, IMS is a local renewable resource with a count of three. IMS resides on the TOR1 z/OS image and is used to represent access to an IMS region. The maximum number of jobs requiring one unit of the IMS resource that can run concurrently is three.

```
RESDEF IMS ADD LOCAL RENEWABLE MAX(3) CPU(TOR1)
```

Example: Defining a global threshold resource

In the following example, NITESHFT is a global threshold resource with a count of one. NITESHFT represents a time period when specific jobs can run. The NITESHFT resource can be manipulated using the SET operand at the appropriate times to ensure night shift jobs run, for example, between 1am and 8am.

```
RESDEF NITESHFT ADD NODAL THRESHOLD AVAIL(1)
```

Example: Defining a local threshold resource

In the following example, PAYJOB1 is a local threshold resource with a count of zero. PAYJOB1 is used to represent a job's completion. To allow successor jobs to be submitted, you can manipulate the PAYJOB1 resource using the SET operand after PAYJOB1 successfully completes.

```
RESDEF PAYJOB1 ADD LOCAL THRESHOLD AVAIL(0)
```

Example: Defining a real resource

In the following example, T3480 is a real resource representing 3480 cartridge drives. CA WA ESP Edition compares the number of tape drives currently online and not allocated with CA WA ESP Edition internal counters to determine when jobs that require the T3480 resource are eligible for submission.

```
RESDEF T3480 ADD LOCAL RENEWABLE DEVICE(3480)
```

Example: Defining a local renewable resource

In the following example, CICS is a local renewable resource with a count of one. CICS resides on the TOR2 z/OS image and is used to represent access to a CICS region. The GRAVITY attribute indicates that the CICS resource is available to jobs on other nodes. If a job on another node requires the CICS resource, CA WA ESP Edition inserts the appropriate routing information into the requesting job JCL according to the system topology statements.

```
RESDEF CICS ADD LOCAL RENEWABLE MAX(1) CPU(TOR2) GRAVITY
```

Example: Defining a default resource

In the following example

- The RESDFLT initialization parameter or command indicates CA WA ESP Edition should make default resource assignments for scratch tapes:
`RESDFLT SCRATCHCART(SCRTAPES)`
- The RESDEF command defines a global depletable resource with a count of 500. CA WA ESP Edition uses historical information to assign scratch tape resource requirements to each job in an application:
`RESDEF SCRTAPES ADD NODAL DEPLETABLE AVAIL(500)`

Note: When default resources are turned on, every application CA WA ESP Edition generates uses default resources. To drop a default resource, code the following:

```
RESOURCE (0,SCRTAPES)
```

Example: Defining resources for Agent workload balancing

The following example creates the RESWIN resource and specifies that CA WA ESP Edition will receive the status of every CPU in the WIN node every five minutes:

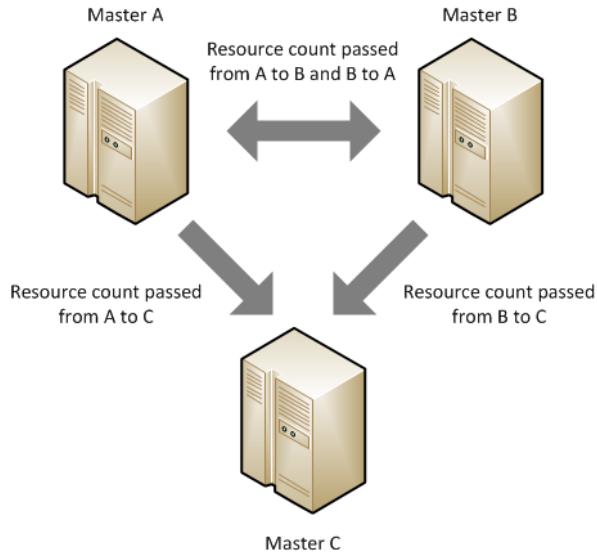
```
RESDEF RESWIN ADD NODE(WIN) LOCAL RENEWABLE MAX(100)+  
MONITOR(CPU) POLL(5 MINUTES)
```

CA WA ESP Edition can balance the workload among CPUs belonging to one node.

Example: Passing Resource Counter Updates Using RECEIVER and SENDER

This example shows how to use the RECEIVER and SENDER operands to pass threshold enterprise resource counter updates to other master subsystems.

Assume that you have three masters, A, B, and C. Each master requires enterprise threshold resource "myres" with a maximum value of 3. Assume that your requirements for passing resource counter updates are as shown in the following diagram:



Here is the RESDEF initialization parameter for master A:

```
resdef myres add threshold enterprise max(3) receivers(B,C) senders(B)
```

Here is the RESDEF initialization parameter for master B:

```
resdef myres add threshold enterprise max(3) receivers(A,C) senders(A)
```

Here is the RESDEF initialization parameter for master C:

```
resdef myres add threshold enterprise max(3) senders(A,B)
```

Chapter 3: Command Reference Guide Updates

MANAGER Command: Specify Scheduler Subsystem for Communications

The MANAGER command specifies the current master subsystem for communications.

Type: Authorized command

```
MANAGER [NAME(mgrname | CENTRAL_MANAGER)  
[TCPIP[(ipaddress)] [BIND]]
```

NAME(*mgrname* | CENTRAL_MANAGER)

Specifies the name of the CA WA ESP Edition subsystem.

Limits: Maximum 16 characters

Default: CENTRAL_MANAGER

Important! For agents, specify the same name in the communication.managerid_n parameter in the agentparm.txt file on your agent system.

Important! For sharing resource counter updates, specify the same name in a COMMCHAN initialization parameter on each master sharing with the current master.

TCPIP[(*ipaddress*)] [BIND]

Specifies the DNS host name or TCP/IP address of the CA WA ESP Edition master subsystem broadcast by the MGRADDR command.

BIND causes CA WA ESP Edition to bind to the IP address when it connects to an agent. BIND guarantees that all connections from the master to the agents have the same TCP/IP address.

Limits: *ipaddress* is in IPV4 or IPV6 format.

Default: If you omit *ipaddress* or the entire operand, CA WA ESP Edition issues the GETHOSTID command to get the address. BIND works with the default address or an address that you specify.

Note: If *ipaddress* is invalid on the master, CA WA ESP Edition does not issue the GETHOSTID command and cannot communicate with the agent.

Examples

The following is an example of the MANAGER command:

```
MANAGER NAME(CM_CENTRAL)
```

The following example sets the manager name CM_ESP with two different TCP/IP addresses, depending on the host subsystem:

```
IF SYSNAME='SYSA' THEN DO  
MANAGER NAME(CM_ESP) TCPIP(10.1.15.1)  
ENDDO  
ELSE IF SYSNAME='SYSC' THEN DO  
MANAGER NAME(CM_ESP) TCPIP(10.1.15.3)  
ENDDO
```

In this example, the MANAGER command specifies a DNS host name called SYSA and requests that CA WA ESP Edition bind to the TCP/IP address corresponding to SYSA when connecting to an agent:

```
MANAGER NAME(CM_CENTRAL) TCPIP(SYSA) BIND
```

RESDEF Command: Define Resources

The RESDEF command defines, displays, deletes and updates resources.

Type: Authorized command

Authority: You need access to the RESOURCE.*resname* security profile.

```
RESDEF name ADD|SET  
      [ENTERPRISE|NODAL|GLOBAL|LOCAL]  
      [THRESHOLD|RENEWABLE|DEPLETABLE]  
      [OWNERS|WAITERS]  
      [NODE(xxx)][CPU(xxx)]  
      [MAXIMUM(n) | INCREMENT(n) | DECREMENT(n)] [AVAIL(0)]  
      [DEVICE(xxxxx) | WLM(duration)]  
          MONITOR(CPU) POLL(xx HOURS|xx MINUTES|xx SECONDS)]  
      [GRAVITY|NOGRAVITY]  
      [COMMENT(comment)]  
      [FORCE]  
      [RECEIVERS(rec1,rec2,...)]  
      [SENDERS(sen1, sen2, ...)]  
  
RESDEF name LIST|DELETE [LASTUSED(criteria)]
```

name

Indicates the resource name.

Limits:

- Up to eight alphanumeric characters for default resources.
- Up to 44 alphanumeric characters for other resources.
- The first character must be alphabetic or national.
- No period at the end or two consecutive periods.
- Wildcard characters are allowed when you use the SET, DELETE, LIST, or LASTUSED operands.

ADD

Defines a new resource.

SET

Sets resource counts of a defined resource.

ENTERPRISE

Indicates one resource counter is maintained across all nodes and CPUs

NODAL

Indicates one resource counter is maintained for each node

GLOBAL

Alias of NODAL.

LOCAL

Indicates one resource counter is maintained for each CPU.

THRESHOLD

Defines a threshold resource type.

RENEWABLE

Defines a renewable resource type.

DEPLETABLE

Defines a depletable resource type.

OWNERS

Displays jobs that are currently executing and holding a resource. Use the OWNERS operand with LIST.

WAITERS

Displays jobs waiting for a resource. Use the WAITERS operand with LIST.

NODE(*xxx*)

Refers to a node defined in the system topology. NODE limits LIST or SET to a specific node.

CPU(*xxx*)

Refers to a CPU defined in the resource topology. CPU limits LIST or SET to a specific CPU.

MAXIMUM(*n*)

Sets the maximum count of the resource you are defining. The default value is 0.

INCREMENT(*n*)

Specifies the value by which to increment the maximum count of the resource. You can only specify INCREMENT along with the SET operand.

DECREMENT(*n*)

Specifies the value by which to decrement the maximum count of the resource. You cannot decrement the maximum resource count below zero. You can only specify DECREMENT along with the SET operand.

AVAIL(*n*)

Sets the available count of the resource you are defining. This value should not be manipulated, but left for CA WA ESP Edition to calculate.

DEVICE (*xxxxx*)

Indicates the IBM generic or esoteric device name that you want CA WA ESP Edition to monitor.

WLM(*duration*)

Specifies that CA WA ESP Edition obtains the number of unused service units over the specified duration from IBM WLM. CA WA ESP Edition obtains the number of unused service units before submitting the job. CA WA ESP Edition supports the following three durations:

- SUM60—The unused service units are summed over the last 60 seconds.
- SUM180—The unused service units are summed over the last 180 seconds.
- SUM600—The unused service units are summed over the last 600 seconds.

Note:

- The WLM operand is only relevant if CA WA Service Governor ESP Edition is installed. If you code the WLM operand without CA WA Service Governor ESP Edition installed, CA WA ESP Edition ignores the WLM operand.
- If you code the WLM operand, you must code the SYSPLEX operand in the NODE command or initialization parameter when you define your system nodes.

MONITOR(CPU)

Specifies that CA WA ESP Edition asks all Agents to provide the percentages of CPU availability on the platforms they control at the frequency specified by the POLL operand

Note: The MONITOR operand is only relevant if the CA WA Service Governor ESP Edition is installed. If you use the MONITOR operand without CA WA Service Governor ESP Edition installed, CA WA ESP Edition ignores the MONITOR operand.

POLL(xx HOURS|xx MINUTES|xx SECONDS)

Specifies how often CA WA ESP Edition asks all Agents to provide the percentage of CPU availability on the platforms they control. xx is a number with a maximum corresponding to 24 hours whether in hours, minutes or seconds. If POLL specifies 0, CA WA ESP Edition does not ask Agents to provide the percentage of CPU availability on the platforms they control.

Note: The POLL operand is only relevant only if the CA WA Service Governor ESP Edition is installed. If you use the POLL operand without CA WA Service Governor ESP Edition installed, CA WA ESP Edition ignores the POLL operand.

GRAVITY

Indicates jobs are to be routed to other nodes the resource is available on.

NOGRAVITY

Indicates jobs are not to be routed to other nodes a resource is available on.

COMMENT (*comment*)

Used to add information about a particular resource.

FORCE

Used with the DEL operand to delete a resource regardless of whether it is in use.

LIST [LASTUSED(*criteria*)]

Displays the definition for the specified resource. If you specify the LASTUSED operand, the resource is only listed if it was last used before the date or date and time specified in *criteria*.

Important! Resources last used before CA WA ESP Edition r11.3 was installed are always listed, even if LASTUSED is set before that point in time.

DELETE [LASTUSED(*criteria*)]

Deletes the definition for the specified resource. If you specify the LASTUSED operand, the resource is only deleted if it was last used before the date or date and time specified in *criteria*.

Important! Resources last used before CA WA ESP Edition r11.3 was installed are not deleted by the DELETE LASTUSED operand.

RECEIVERS(*rec1,rec2,...*)

Specifies the masters that can receive threshold enterprise resource count updates from the current master. The masters specified update their local resource count to match the count that was sent. Each master name in this operand must be specified in a COMMCHAN initialization parameter on the current master. You can use the hyphen and asterisk wildcard characters in master names.

SENDERS(*sen1, sen2,...*)

Specifies the masters that can send threshold enterprise resource count updates to the current master. The current master updates the local resource count to match the count that was sent. Each master name in this operand must be specified in a COMMCHAN initialization parameter on the current master. You can use the hyphen and asterisk wildcard characters in master names.

Defining a resource

Before you use the resources feature, a data set called the RESFILE needs to be defined and identified in the CA WA ESP Edition initialization parameters.

Resource definitions are stored in the RESFILE. If CA WA ESP Edition is started with the RESFORM option, information stored in the RESFILE is lost. A useful technique to ensure that system topology and resource definitions are not lost on a reformat (RESFORM) of the RESFILE is to code the following in your CA WA ESP Edition initialization parameters:

```
IF RESFORM THEN DO
  NODE TORONTO ADD ROUTEJCL('/*XEQ TORONTO')
  CPU T1 ADD NODE(TORONTO) ROUTEJCL('/*JOBPARM SYSAFF=T1') ORDER(1) CURRENT
  CPU T2 ADD NODE(TORONTO) ROUTEJCL('/*JOBPARM SYSAFF=T2') ORDER(2)
  CPU T3 ADD NODE(TORONTO) ROUTEJCL('/*JOBPARM SYSAFF=T3') ORDER(3)
  RESDEF T3480 ADD LOCAL RENEWABLE MAX(5) CPU(T1)
  RESDEF CICSUP ADD NODAL THRESHOLD AVAIL(0)
  RESDEF DB2TAB1 ADD NODAL RENEWABLE AVAIL(3)
ENDDO
```

If you are using the resource feature extensively and have a large number of resources defined, you may prefer not to code all your resource definitions in the CA WA ESP Edition initialization parameters. An alternative method is to store your resource definitions in a data set and issue the LOAD command to process the RESDEF commands, in the event the RESFILE is reformatted.

Note: The LOAD command cannot be used in the CA WA ESP Edition initialization parameters.

Real resources

A real resource represents a hardware device such as tape drives. Since the CA WA ESP Edition master is only aware of the status of devices on the image it runs on, real resources only work in a single z/OS image. Define real resources as LOCAL and RENEWABLE and, using the MAX operand, identify the number of devices available to CA WA ESP Edition-submitted jobs:

```
RESDEF T3480 ADD LOCAL RENEWABLE MAX(5) DEVICE(3480)
```

The following example shows what appears when the above resource is listed:

```
RESDEF T3480 LIST
Resource T3480      Local Renewable
TORONTO  MVSA      Max=5 Avail=5 Real=5
```

CA WA ESP Edition ensures there are sufficient online, unallocated devices available to satisfy the current job's resource requirements.

For example, if job PAYJOB1 requires three units of a real resource called T3480, and CA WA ESP Edition sees three devices available, but only two devices are online and unallocated, CA WA ESP Edition waits for real resources and PAYJOB1 goes into a RESWAIT state.

Resources with IBM WLM operand (CA WA Service Governor ESP Edition only)

You must define an IBM WLM resource as LOCAL RENEWABLE.

To determine the MAX count, see the “CPU capacity table” in the *MVS planning: Workload Management* IBM manual. From this table, choose the number of service units per second for your CPU model and multiply it by either 60, 180 or 600 for the duration SUM60, SUM180 or SUM600 respectively. You can use the result or any higher number. If your installation has CPUs of different models, use the RESDEF SET command to adjust the counts on individual CPUs.

CA WA ESP Edition routes a job to one of the available systems based on the service units assigned to the job and other resources the job requests.

IBM WLM feedback is a measurement of recent but past activity; it is not a precise prediction of a CPU load by the time CA WA ESP Edition submits a job. The real value of a WLM resource is the number of unused service units. The number of unused service units can differ from the number of service units available for a job if the job service class is higher than the service class of the currently executing workload.

The workload is balanced across the CPUs and nodes within the same sysplex, as long as they are all part of the same sysplex group. The node is defined with the NODE command or initialization parameter (it must include a SYSPLEX operand).

Note: Specify the same duration for all CPUs and nodes. CA WA ESP Edition issues a warning message if you specify a different duration for a specific CPU or node.

[Resources with MONITOR\(CPU\) operand \(CA WA Service Governor ESP Edition only\)](#)

You must define a MONITOR(CPU) resource as LOCAL RENEWABLE.

CA WA ESP Edition asks all Agents to provide the percentage of CPU availability on the platforms they control. The Agents provide this percentage at the frequency specified by the POLL operand for CPUs where the corresponding CPU initialization parameter includes an AGENT operand. You can limit the scope of the resource you define by using the NODE and CPU operands.

CA WA ESP Edition routes a job to one of the available systems based on the MONITOR(CPU) resources and other resources a job requests.

The workload is balanced among the CPUs that belong to the same node. The node is defined with the NODE command or initialization parameter.

[Setting resource counts](#)

For depletable and threshold resources, there is only one count. You can change the count using the MAXIMUM, INCREMENT, DECREMENT, or AVAIL operand of the RESDEF command.

For renewable resources, there are two counts: maximum and available. Only the maximum count can be changed using the MAXIMUM, INCREMENT, or DECREMENT operand of the RESDEF command. If the maximum is changed, the available count changes by the same amount.

The following is an example of the display produced when a renewable resource is listed and what happens when the MAX count is manipulated:

```
RESDEF CICS LIST
Resource CICS  Local Renewable
TORONTO  MVSA  Max=4  Avail=2
```

```
RESDEF CICS SET MAX(6)
```

```
RESDEF CICS LIST
Resource CICS  Local Renewable
TORONTO  MVSA  Max=6  Avail=4
```

The following is an example of the display produced when a resource is listed and what happens when the AVAIL count is changed for a threshold resource:

```
RESDEF DAYSHIFT LIST
Resource DAYSHIFT Local Threshold
TORONTO    MVSA      Avail=1

RESDEF DAYSHIFT SET AVAIL(2)

RESDEF DAYSHIFT LIST
Resource DAYSHIFT Local Threshold
TORONTO    MVSA      Avail=2
```

[Related information](#)

For information on requesting resources, see “[RESOURCE Statement: Identify Resources](#)”.

For information on specifying default resources, see “[RESFLT Command: Identify Default Resources](#)”.

For more information on resources, see the *User Guide*.

For information on how to use IBM WLM or Agent workload balancing, refer to the *User Guide*.

For more information on the RESFILE, see the *Installation and Configuration Guide*.

[Example: Displaying resource definitions](#)

In the following example, all existing resource definitions are displayed:

```
RESDEF - LIST
```

[Example: Defining a local renewable resource](#)

In the following example, IMS is a local renewable resource with a count of three. IMS resides on the TOR1 z/OS image and is used to represent access to an IMS region. The maximum number of jobs requiring one unit of the IMS resource that can run concurrently is three.

```
RESDEF IMS ADD LOCAL RENEWABLE MAX(3) CPU(TOR1)
```

Example: Defining a global threshold resource

In the following example, NITESHFT is a global threshold resource with a count of one. NITESHFT represents a time period when specific jobs can run. The NITESHFT resource can be manipulated using the SET operand at the appropriate times to ensure night shift jobs run, for example, between 1am and 8am.

```
RESDEF NITESHFT ADD NODAL THRESHOLD AVAIL(1)
```

Example: Defining a local threshold resource

In the following example, PAYJOB1 is a local threshold resource with a count of zero. PAYJOB1 is used to represent a job's completion. To allow successor jobs to be submitted, you can manipulate the PAYJOB1 resource using the SET operand after PAYJOB1 successfully completes.

```
RESDEF PAYJOB1 ADD LOCAL THRESHOLD AVAIL(0)
```

Example: Defining a real resource

In the following example, T3480 is a real resource representing 3480 cartridge drives. CA WA ESP Edition compares the number of tape drives currently online and not allocated with CA WA ESP Edition's internal counters to determine when jobs that require the T3480 resource are eligible for submission.

```
RESDEF T3480 ADD LOCAL RENEWABLE DEVICE(3480)
```

Example: Defining a local renewable resource

In the following example, CICS is a local renewable resource with a count of one. CICS resides on the TOR2 z/OS image and is used to represent access to a CICS region. The GRAVITY attribute indicates that the CICS resource is available to jobs on other nodes. If a job on another node requires the CICS resource, CA WA ESP Edition inserts the appropriate routing information into the requesting job JCL according to the system topology statements.

```
RESDEF CICS ADD LOCAL RENEWABLE MAX(1) CPU(TOR2) GRAVITY
```

Example: Defining a default resource

In the following example

- The RESDFLT command indicates CA WA ESP Edition should make default resource assignments for scratch tapes:

```
OPER RESDFLT SCRATCHCART(SCRTAPES)
```

- The RESDEF command defines a global depletable resource with a maximum count of 500. CA WA ESP Edition uses historical information to assign scratch tape resource requirements to each job in an application:

```
RESDEF SCRTAPES ADD NODAL DEPLETABLE AVAIL(500)
```

Note: When default resources are turned on, every application CA WA ESP Edition generates uses default resources. To drop a default resource, code the following:

```
RESOURCE (0,SCRTAPES)
```

Example: Defining resources for Agent workload balancing

The following example creates the RESWIN resource and specifies that CA WA ESP Edition will receive the status of every CPU in the WIN node every five minutes:

```
RESDEF RESWIN ADD NODE(WIN) LOCAL RENEWABLE GRAVITY MAX(100)+  
MONITOR(CPU) POLL(5 MINUTES)
```

CA WA ESP Edition can balance the workload among CPUs belonging to one node.

Example: Setting the available count of a threshold resource

In the following example, the available count of the LOWPRIO resource is set to one:

```
RESDEF LOWPRIO SET AVAIL(1)
```

Example: Setting the available count of a depletable source

In the following example, the available count of the SCRTAPES resource is set to 550:

```
RESDEF SCRTAPES SET AVAIL(550)
```

Example: Setting the maximum count of a renewable resource

In the following example, the count of the CICSUP resource is set to two:

```
RESDEF CICSUP SET MAX(2)
```

Example: Increasing the maximum count of a renewable resource

In the following example, the count of the CICSUP resource is incremented by three:

```
RESDEF CICSUP SET INCREMENT(3)
```

Example: Deleting a resource

In the following example, the COCONUT resource is deleted:

```
RESDEF COCONUT DELETE
```

Example: Displaying a specific resource

In the following example, jobs using and jobs waiting for the DBASE resource are displayed:

```
RESDEF DBASE LIST OWNERS WAITERS
```

Example: List all resources not used since a specified date

In the following example, RESDEF is issued to list all the resources last used before January 08, 2008. The output shows that local, threshold resource NiteShift was last used by the application APPL2 on March 20, 2006 and local, renewable resource T3480 has never been referenced by any application since the LASTUSED feature was installed.

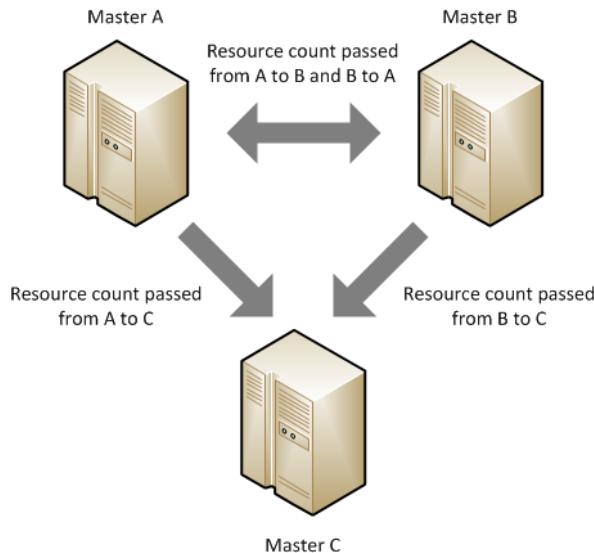
```
RESDEF - LIST LASTUSED(08/01/01)Resource NiteShift Local Threshold  
Lastused 14:30 MAR20 2006 by APPL2  
Resource T3480 Local Renewable Lastused: no information
```

Note: When you specify the date criteria in the RESDEF LASTUSED operand, all the resources used before the specified date are listed. For example, RESDEF - LASTUSED(2007jun1) specifies that the resource was last used before June 1, 2007.

Example: Passing Resource Counter Updates Using RECEIVER and SENDER

This example shows how to use the RECEIVER and SENDER operands to pass threshold enterprise resource counter updates to other master subsystems.

Assume that you have three masters, A, B, and C. Each master requires enterprise threshold resource "myres" with a maximum value of 3. Assume that your requirements for passing resource counter updates are as shown in the following diagram:



Here is the RESDEF command for master A:

```
resdef myres add threshold enterprise max(3) receivers(B,C) senders(B)
```

Here is the RESDEF command for master B:

```
resdef myres add threshold enterprise max(3) receivers(A,C) senders(A)
```

Here is the RESDEF command for master C:

```
resdef myres add threshold enterprise max(3) senders(A,B)
```