Guidance for using WSC z/OS Workload Manager Starter Policy

THIS SERVICE DEFINITION IS FOR SAMPLE PURPOSES ONLY. IT IS INTENDED TO PROVIDE EXAMPLES OF HOW TO SPECIFY WLM CONSTRUCTS.
INSTALLATIONS ARE EXPECTED TO CHANGE THE SERVICE DEFINITION FOR THEIR SPECIFIC SITUATION AS APPROPRIATE BEFORE ACTIVATING ANY POLICY. WE HAVE DOCUMENTED SOME ITEMS WE EXPECT CUSTOMERS TO CHANGE AND THOSE THINGS WE RECOMMEND THEY DO.

This service definition has a functionality level of LEVEL038 and was created on a z/OS Version 3 Release 1 system. You need to be aware of the levels supported in your environment and make sure you remove or disable any functions not supported by your installation. Functionality levels are described in detail in the z/OS MVS Planning Workload Management Manual, in the Chapter on migration.

HIGHLY RECOMMENDED ACTION ITEMS

A) Read the documentation.

- z/OS MVS Planning: Workload Manager V3.1
- z/OS MVS Planning: Workload Manager V2.5
- z/OS WLM homepage

B) Keep the number of active service class periods between 25 and 35.

This service definition contains 19 service class periods. The number of service class periods we generally recommend customers to have ACTIVE on an image at any one time is approximately 30. We have defined 19 in this starter policy as examples of how to classify various work. You will need to add, remove, and modify service classes in this policy to classify your workloads appropriately.

C) Set I/O Priority Management to NO.

If an analysis of your DASD subsystem shows high disconnect times and little or no IOS queue time, then we recommend you set this option to NO. If you have high IOS queue times and also have high disconnect times, then you will have to evaluate the benefits of using I/O Priority Management in your installation. I/O Priority management is turned off in this policy, which is the recommended value as of 2024 for most customer environments.

When I/O Priority Management is set to YES, disconnect times are included in the velocity goal calculation. High disconnect times will cause the system to calculate velocity goals that are higher than one would expect to receive if you did not have high disconnect times. Since the PI (and subsequent WLM decisions) are based

on the goals set versus goals achieved, you will need to set or adjust your velocity goals appropriately taking this calculation into consideration if I/O Priority Management is changed to NO.

With APAR OA66145, the velocity migration field in RMF is now bi-directional and reports what the execution velocity would be if the I/O Priority Management value is changed. It can be used to determine what the new velocity goal should be for each service class when disabling or enabling I/O Priority Management.

D) **Set Service Coefficients to the recommended values** if running z/OS 2.4 or below**.** The following recommended values are predefined in z/OS 2.5 and above and can no longer be modified. When changing values, it may be necessary to change the durations for any multi-period service class appropriately.

CPU 1.0
IOC 0.0
MSO 0.0
SRB 1.0

E) **Provide a default service class and a unique report class for all subsystems**. Monitor your RMF reports for tasks that fall into this service class and reclassify them appropriately.

Use the NEWWORK service class for all unclassified work in the absence of any other service class of your own. You should use this service class where you have no classification rules defined and provide a unique report class so you can easily identify which subsystem the work is coming from. For report classes, this policy uses the naming convention RC for Report Class, U for unclassified work, and the subsystem type. For example, the default report class for unclassified DDF work is RCUDDF.

F) **Use the SPM system rules to classify SYSTEM and SYSSTC work appropriately.** See the WLM planning manual section on classifying system tasks.

NOTE: If a program has been assigned the SYST attribute in either the program properties table(PPT) in IEFSDPPT in SYS1.LINKLIB or in the SCHEDxx member of SYS1.PARMLIB, then the address spaces executing these programs will be classified to SYSSTC unless you explicitly specify a classification rule for these tasks before the SPM rules.

G) Clarification on JES classification rules using Subsystem Collection.

WLM in z/OS allows work to be classified by system name. System name, however, cannot be used for JES. This is because the WLM classification call will be made on the system where the job ends conversion, which may or may not be the system where the job ends up being executed. Another classification qualifier, Subsystem Collection Name (SSC), is provided which allows JES work to be segregated by MAS. For JES shared spool complexes, the XCF group name, (in WLM it's called Subsystem Collection Name), should be used to provide classification integrity across the JES-PLEX members. The use of the system name qualifier is prohibited for JES rules. For JES2 you can display the appropriate name to use with the $DMASDEF command. The "XCFGRPNM" field will be displayed with the name of the qualifier you need to specify with the SSC type parameter of the JES classification rules. Using the SSC technique for JES rules will allow those with multiple JES MAS's to classify similar work differently among those JES MAS's. For example, batch job "compile" can have a job class of C and be assigned service class "BATFAST" in one JES MAS and service class "BATSLOW" in a different JES MAS as shown below. Each JES MAS of course may have multiple members (images).

____	1	SSC		WSCSYS1	___	________
____	2		TC	C	___	BATFAST
____	1	SSC		WSCSYS2	___	________
____	2		TC	C	___	BATSLOW

ITEMS THAT WILL NEED YOUR ATTENTION

1. **<u>You must change the goals and importance levels to meet the needs of your workloads.</u>**
   In the absence of any data of your own, the service classes included in this policy can be good starting points. You will additionally need to modify the classification rules to suit your installation.

2. **Modify the names of any constructs** (workloads, service classes, and report classes) to suit your own naming conventions if necessary.

3. **Modify or delete any application or scheduling environments as appropriate for your installation.**

4. **For the CB, CICS, and DDF rules (as well as others) it is highly useful to specify unique report classes for transactions that use the same service class.**

This allows you to better isolate various transactions and their performance to greatly help with any problem determination tasks.

5. **Monitors for z/OS and applications need to be properly classified in order to give the monitor appropriate importance while not causing performance issues of its own.**

   For most monitors, such as RMF, it is acceptable to classify them into the SYSSTC service class. Any monitors which may have long running, CPU intensive tasks, should be classified to a service class with an importance of 1 and a very high velocity goal, such as 70 or 80, and CPU Critical set to YES.

6. **Tenant Resource Groups and Tenant Report Classes.**
   Tenant resource groups allow the metering and optional capping of workloads, along with the ability to map those workloads directly to Tailored Fit Pricing for IBM Z solutions. Comparable to Resource Groups, Tenant Resource Groups allow you to specify a maximum processor capacity or memory limit which work associated with the group may use.

   This policy defines an example Tenant Resource Group, RGZCX, associated with a Tenant Report Class, RCZCX. It is used to cap CPU for zCX workloads and enforce ILMT. Optionally, Tenant Resource Groups can cap based on an IBM provided 64-character Solution ID.

7. **SYSSTC considerations.**
   The SYSSTC service class is for high importance, short running tasks. It has the second highest dispatching priority (FE). Long running tasks which end up here have the potential to block any other work.

   For DB2 and IMS, only IRLM address spaces should be in SYSSTC as lock manager needs a high dispatching priority in order to let work flow properly through the system. All other DB2 address spaces (DB2XMSTR, DB2XDBM1, etc.) should be at importance 1 with a high velocity and CPU Critical set to YES.

8. **Importance Levels**
   The best practice is to use of all five importance levels to differentiate work. This ensures that during periods of resource contention, the business will know where tradeoffs are being made between different workloads, and which will see delay. If only one or two importance levels are used for the majority of active work, it makes it difficult to tell which service class(es) would see performance

degradation first or where tradeoffs would be made between different online workloads.

9. **Use of CPU and Storage Critical.**
   When CPU Critical is set to YES for a service class, it ensures the work in that service class will not get a lower dispatch priority than work at a lower importance. In general, limit use of CPU critical. It's intended use is when rapid workload shifts happen regularly and WLM will not be fast enough in adjusting priorities. Note that it only protects work from lower importance work, not from work at its same or higher importance. This policy specifies CPU Critical for the Db2 address spaces.

   Storage Critical is defined in a classification rule and is used to assign storage protection to those address spaces. Use of Storage Critical is limited to subsystem types ASCH, JES, OMVS, STC, and TSO.

10. **Use of CCImp parameter.**
    Starting with z/OS 3.1, there is a new IEAOPTxx parameter, CCImp, which tells Workload Manager which importance level(s) to implicitly assign CPU protection. The options are 0, 1, or 2, with 1 being the default. A value of 1 tells Workload Manager to set any work at importance one for the first period of any service class to be CPU Critical. We recommend accepting the default of CCImp=1.

11. **Setting realistic velocity goals.**
    To set reasonable velocity goals, check the velocities of SYSTEM and SYSSTC using an RMF Workload Activity Report to determine the highest achievable velocities and use them as a basis for the rest of the policy. Note that smaller n-way partitions will necessitate lower velocity goals.

12. **Use of percentile response time goals over average response time.**
    Only workloads with very predictable patterns and distributions should use average response time goals. A single long running task can change the measured average response time and change the PI calculations WLM is monitoring. Most workloads which can use a response time goal will need a percentile response time goal defined in order to remove the influence of outliers on response time calculations.

13. **zCX considerations.**
    This policy defines one service class in started tasks, ZCXHI, for zCX instances. If multiple zCX instances differ in the workload they're running or have different performance requirements, separate service classes should be created.

zCX service class(es) should be defined with a velocity goal under the Started Task Subsystem and classified using the *job_name* or *userid*.

Note that most of zCX processing is zIIP-eligible and will run on zIIPs if there is enough capacity. zCX workloads are also affected by changes to the IIPHONORPRIORITY OPT parameter, which can be set at the system level or service class level to specify whether zIIP-eligible work is allowed to run on GCPs.

14. **Use of CAPPED service class and resource group.**
    This service class has a discretionary goal and is defined to a resource group, RGCAPPED, which has a max capacity of 1 service unit/second. It has no classification rules, but is available to manually move work into it to limit resource consumption for some reason. Examples are looping or runaway tasks impacting more important workloads.

15. **MOBILE Reporting Attribute.**
    WLM allows you to classify and measure mobile workloads that are eligible for Mobile Workload Pricing. You can tag any of your work as MOBILE with the WLM Reporting Attribute within the classification rules. As an example, this policy defines a subset of CICS transactions with the qualifier TRN3 with this attribute.

    The default Reporting Attribute for all transactions is NONE.

16. **Goal Management for CICS/IMS - TRANSACTION, REGION, or BOTH**
    From a z/OS perspective, z/OS is dispatching CICS and IMS regions, not individual transactions. This is why in the STC rules there is an option to manage a region to a transaction goal, region goal, or BOTH. The default is TRANSACTION for all workloads.

    Managing CICS/IMS regions to transaction goals means that when transactions are active the CICS/IMS region is acting as a 'SERVER' service class and its own goal is ignored. If transaction goals are defined but only planned on being used for monitoring, then the REGION option is needed for these regions.

    There is no one correct answer for which management type to use. If your system has multiple members of a sysplex and each partition is a different size, it may be easier to run with transaction goals. Transaction goals are also easier to understand and report on to management or others without the technical knowledge of what a velocity goals means.

Environments where transactions can be unpredictable and no classification rule can be used to separate transactions with different response times may be better suited to use REGION goals. We do advise defining transaction goals and report classes to get the most out of RMF management as this will give you transaction response time and other information without needing to go to other monitors.

The BOTH option in WLM was implemented to allow CICS TORs, which typically have low resource consumption, to be managed to the higher of the transaction goal or region goal specified to it. Typically, this will give CICS TORs a higher dispatching priority than other CICS regions and help with overall transaction flow.

Resources and references

"z/OS Workload Manager: What Are You Thinking?", IBM Washington Systems Center

z/OS MVS Planning – Workload Management V2R5

z/OS WLM homepage