

Automated **escalation**
ensures your production
gets the **service** it needs

ThruPut Manager[®] AE and Production Batch

This paper explores how ThruPut Manager AE (Automation Edition) addresses issues particular to z/OS production batch. The TM/Production Control Services™ option provides additional services for CA 7® Workload Automation users. However this paper will confine itself to the general case of ThruPut Manager AE without TM/PCS.

Production Batch in a JES2 Environment

Production batch workload is different from other batch workloads in a number of ways.

- It is usually submitted by a job scheduler. The job scheduler knows the job dependencies and submits the jobs in a logical order.
- Production is usually the most important batch workload. It takes precedence over development and other ad hoc workloads. Consequently it should only compete for resources with other production work, and possibly the online workload.
- Unlike other batch workloads, production can have financial penalties for missing deadlines, with no particular benefit for finishing early.
- Production workload is fairly predictable and well understood. It doesn't change that frequently and when it does change, it is subject to strict change control procedures.
- Production batch, like other batch, has a propensity for using external resources – indeed this is one reason why batch processes aren't written as online transactions in the first place. Fetching a resource,

...the auto-pilot engine makes sure your production workload gets the service you want it to, with the minimum of manual intervention.

such as staging a virtual volume, may be relatively time-consuming and delays the prompt execution of the job.

Automating Batch Service Management

In ThruPut Manager Automation Edition, a datacenter uses **policies** to describe their workload to ThruPut Manager's auto-pilot engine. A policy specifies the **service goals** and relative **batch importance** for each workload, together called a **Service Group**. It also controls the deployment of the JESplex and LPARs, in the form of **constraints** on jobs and drives. With these simple settings and the built-in heuristics, the auto-pilot engine makes sure your

production workload gets the service you want it to, with a minimum of manual intervention.

The following sections discuss how ThruPut Manager is pertinent to managing your production batch workload.

- **Automatic Escalation** explains how ThruPut Manager automatically adjusts the priority of a job while it's in the system, to give it the service it warrants with minimal operator intervention.
- **Resource Management** explains the benefits of ThruPut Manager's strategy of optimizing external resource management during queue time. Though astute storage management procedures for production jobs can mitigate delays due to external resources, the impact can still be significant.
- **Improved Throughput** explains how ThruPut Manager optimizes and automates the production workload to deliver it in less time.

8300 Woodbine Avenue
4th Floor
Markham ON Canada
L3R 9Y7

905.940.9404
905.940.5308 (fax)
marketing@mvssol.com
www.mvssol.com

MVS
solutions inc.

ThruPut
Manager[®] AE

Automatic Escalation

ThruPut Manager provides automatic escalation when a job waits too long in the queue. Your policy settings indicate what escalation you want for jobs in each Service Group. The escalation is relative to other jobs in the system at the same time. Though all jobs may enjoy escalation, production jobs are most likely to need the benefits of it. It is useful to look at three situations: “business as usual”, “system is stressed”, and “things are bad”.

Business as usual

In a Policy, each Service Group defines a TARGET and ACCEPTABLE time to be in the queue. These are values for what you would expect under normal conditions. Values for

Sample Service Groups

INVENTORY . PROD

TARGET = 1 minute
ACCEPTABLE = 3 minutes
Batch Importance = 2
CRITICAL = 7 minutes
Urgency = C2

INVENTORY . ADHOC

TARGET = 5 minute
ACCEPTABLE = 15 minutes
Batch Importance = 4
CRITICAL = <no value set>
Urgency = <no value set>

Figure 1: Using only five settings, a datacenter can design a set of Service Groups to differentiate its workload into the groups of jobs which warrant different relative levels of service. Once the Service Groups are specified, then ThruPut Manager's auto-pilot engine uses those settings to optimize the workload, automatically.

production jobs are probably tighter than for ad hoc compiles. This means a production job reaches its thresholds faster than jobs from other Service Groups. This is an “aging” strategy. Jobs that have aged more are given higher priority in the queue. Though some jobs travel faster than others, if a job waits long enough it will reach the ACCEPTABLE threshold.

If a production job has to stage external resources, it will still reach its thresholds faster and will be selected promptly once the resources are staged. All things being equal, production jobs are processed earlier than non-production jobs.

Under normal conditions all jobs should be processed within the ACCEPTABLE threshold.

System is stressed

If jobs are exceeding their ACCEPTABLE threshold, it suggests the system is stressed and ThruPut Manager knows it's time to give preference to the most important work – i.e., “use an importance strategy”. Each Service Group has a value between 1 and 5 to indicate its Batch Importance from a business point of view and this now comes into play.

Once past the ACCEPTABLE thresholds all the jobs with batch importance of 1 will be given higher priority than any other job with a batch importance between 2 and 5. Typically production jobs will have Batch Importance set to 1 or 2. (Our ad hoc compiles might have a batch importance of 4.) Within the group of jobs with the same batch importance, a job will “age” towards the CRITICAL threshold according to the CRITICAL value for its Service Group. Most production jobs will have a value for this attribute, set relative to other production jobs. (Non-productions jobs such as the ad hoc compiles, would omit the CRITICAL value, and would not age beyond the ACCEPTABLE threshold.)

The “system is stressed” situation should occur at most occasionally and, when it starts to happen too often, would indicate a lack of capacity.

Things are bad

When things are bad, say from an outage or unprecedented load, then some jobs must be run, while others can wait until the situation improves. ThruPut Manager invokes an “urgency” strategy under these circumstances.

Once a job “ages” past the CRITICAL threshold without being selected, it is placed at one of the three highest priorities, indicated by the setting in the Service Group of C1, C2, or C3. All C1 jobs are selected before any C2 and C3 jobs, regardless of the Service Group they belong to.

Only the jobs you want to give precedence to when you can't run all the work should have a CRITICAL value set.

Thus production jobs get the service they need under normal circumstances with automatic escalation as the system degrades and automatic recovery as the system improves.

When things improve

Once the outage is over, and there are no longer any C1, C2, or C3 jobs in the queue, then the jobs which are above the ACCEPTABLE threshold will be selected, those with Batch Importance 1 before those with Batch Importance 2, 3, 4, and 5. Once there are no jobs above their ACCEPTABLE threshold, then the job closest to its ACCEPTABLE threshold is selected, and so on to the TARGET threshold.

Thus production jobs get the service they need under normal circumstances with automatic escalation as the system degrades and automatic recovery as the system improves.

Resource Management

Pre-initiation resource fetching

In an attempt to expedite processing, some datacenters have a “zero queue time” strategy for their production workload. They believe that even if there are external resources to be fetched, it’s better to get the job selected as soon as possible.

At MVS Solutions, we believe tying up an initiator while you are fetching external resources is counter-productive. Over many years of experience, we have proven that the queue is the time to fetch resources, then when the job is ready for execution the initiator takes over. While the resources are being fetched, the initiator can be providing execution services for other jobs which don’t need, or already have, their external resources. Based on the policy specification, the next available initiator always selects the highest priority job that is ready to execute. No further delays are experienced because the external resources are fetched in the queue rather than after initiation. This strategy delivers better service for high priority work than the “zero queue time” approach, and delivers better overall throughput for the entire workload.

Prioritized fetching

In addition to queue time fetching of resources, ThruPut Manager provides for prioritized fetching. For instance, if two jobs are trying to recall HSM archived datasets, it gives precedence to the job with the higher Batch Importance.

Simplified maintenance of the scheduler database

Some datacenters use job scheduler features, such as negative dependencies, to control external resources and avoid contention. ThruPut Manager does this with much less specification effort and does it better since ThruPut Manager can detect and automatically avoid contention at runtime.

Improved Throughput

Better throughput is unquestionably good for production. The same work can be run with fewer resources and at a lower cost. There is more slack in the batch window and ultimately this means that hardware upgrades can be deferred. Some customers have told us they have deferred upgrades by 6-12 months by deploying ThruPut Manager.

When better throughput is achieved through optimization and automation of batch workload management, then you have the extra benefit of less dependence on manual operator intervention. Operator training is reduced and procedure maintenance is simplified. While that is a direct benefit in and of itself, the benefit from the user point of view is the increased consistency of response and service. ThruPut Manager increases throughput in the following ways:

Removes delays

As explained earlier, ThruPut Manager removes initiator delays by staging external resources during queue time. Unlike traditional JES staging, ThruPut Manager stages all the resources for a job in parallel rather than sequentially, further reducing the delay. This both prepares the job to execute earlier and improves initiator utilization, thereby allowing more jobs to finish in the same elapsed time.

Optimizes execution

ThruPut Manager optimizes execution by deploying its dynamic TM-managed initiators to stop and start initiators as needed (See the *ThruPut Manager AE and Dynamic Initiators* white paper). It also verifies a WLM Service Class is receiving service before starting another job in that Service Class. ThruPut Manager manages dataset

contention automatically, giving preference to higher priority work, and avoiding dataset contention “ripple”. This means production jobs finish earlier, leaving more slack in your batch window.

ThruPut Manager pre-emptively captures information for later reporting, including performance against the original policy settings, reasons for delays, and the number of Service Units consumed.

Optimize LPAR /JESplex Utilization

There is a tendency for datacenters to run production and non-production on separate LPARS/JESplexes in an effort to give production the preference it needs. ThruPut Manager can achieve this without such extreme measures – and realize the benefits of consolidating workloads. For instance, a datacenter could allow up to, say, 5 production jobs that have reached the CRITICAL threshold to start on their “online” image. If the job was not yet CRITICAL, or it wasn’t “production”, then it would run elsewhere. This same feature could be used to limit the number of non-production jobs running on a “production” image. This allows more efficient use of resources without sacrificing production service.

Reduce interference from non-production batch jobs

Besides the prioritized fetching of external resources, ThruPut Manager has other features that prevent interference from non-production workload:

- **Contention from TSO Users:**
If a TSO user is holding a dataset, ThruPut Manager can send a message asking them to release it, and can, under some circumstances, repossess it.
- **Reserve and Cap Drives:**
When you describe your configuration to ThruPut Manager, you can both **reserve** and **cap** drives for specific workloads. Reserving drives allows you to have drives that can only be used for production. The capping feature allows several workloads to

share drives, but limits any one workload from ‘hogging’ them. You can manipulate these values to get the service you need for production, while allowing other work to use resources as they are available.

Benefits of ThruPut Manager AE

ThruPut Manager AE has many features to enforce standards, reduce reliance on end-user compliance, distribute workload, and save on software license costs, to name a few. All of these contribute to a healthy batch management environment, and in many cases the impact is greater for production batch. ThruPut Manager AE is the product of choice to automate and optimize your z/OS production batch service.

This document assumes the reader is familiar with ThruPut Manager AE (Automation Edition). Only certain highlights of the product have been discussed here. For further information, please contact us as noted on the first page.

ThruPut Manager is a registered trademark of MVS Solutions Inc. The names of the optional components of ThruPut Manager used in this document are trademarks of MVS Solutions Inc. Other trademarks and registered trademarks used in this document are the property of their respective owners and are to be regarded as appearing with the appropriate ™ or © symbol.

© MVS Solutions Inc. 2009. All rights reserved.

910

