



WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Languages
Català
Deutsch
Español
فارسی
한국어
日本語
Svenska

Edit links

Article **Talk**

Read **Edit** View history

Search

Multilevel feedback queue

From Wikipedia, the free encyclopedia
(Redirected from [Multi level feedback queue](#))

In [computer science](#), a **multilevel feedback queue** is a [scheduling](#) algorithm. Solaris 2.6 Time-Sharing (TS) scheduler implements this algorithm.^[1] The Mac OS X and Microsoft Windows schedulers can both be regarded as examples of the broader class of multilevel feedback queue schedulers.^[2] This scheduling algorithm is intended to meet the following design requirements for **multimode systems**:

- Give preference to short jobs.
- Give preference to [I/O bound](#) processes.
- Separate processes into categories based on their need for the processor.

The Multi-level Feedback Queue scheduler was first developed by Corbato et al. in 1962, and this work, along with other work on Multics, led the ACM to award Corbato the [Turing Award](#).^[3]

Contents [hide]

- 1 Process Scheduling
 - 1.1 Algorithm
 - 1.2 Scheduling parameters
- 2 See also
- 3 References

Process Scheduling [edit]

Unlike [multilevel queue](#) scheduling algorithm where processes are permanently assigned to a queue, multilevel feedback queue scheduling allows a process to move between queues. This movement is facilitated by the characteristic of the CPU burst of the process. If a process uses too much CPU time, it will be moved to a lower-priority queue. This scheme leaves I/O-bound and interactive processes in the higher priority queues. In addition, a process that waits too long in a lower-priority queue may be moved to a higher priority queue. This form of [aging](#) also helps to prevent [starvation](#) of certain lower priority processes.

Algorithm [edit]

Multiple [FIFO](#) queues are used and the operation is as follows:

- A new process is inserted at the end (tail) of the top-level [FIFO](#) queue.
- At some stage the process reaches the head of the queue and is assigned the [CPU](#).
- If the process is completed within the [time quantum](#) of the given queue, it leaves the system.
- If the process voluntarily relinquishes control of the CPU, it leaves the queuing network, and when the process becomes ready again it is inserted at the tail of the same queue which it relinquished earlier.
- If the process uses all the quantum time, it is [pre-empted](#) and inserted at the end of the next lower level queue. This next lower level queue will have a time quantum which is more than that of the previous higher level queue.
- This scheme will continue until the process completes or it reaches the base level queue.
 - At the base level queue the processes circulate in [round robin](#) fashion until they complete and leave the system. Processes in the base level queue can also be scheduled on a [first come first served](#) basis.^[4]
 - Optionally, if a process blocks for I/O, it is 'promoted' one level, and placed at the end of the next-higher queue. This allows I/O bound processes to be favored by the scheduler and allows processes to 'escape' the base level queue.

For scheduling, the scheduler always start picking up processes from the head of the highest level queue. If the highest level queue has become empty, then only will the scheduler take up a process from the next lower level queue. The same policy is implemented for picking up in the subsequent lower level queues. Meanwhile, if a process comes into any of the higher level queues, it will preempt a process in the lower level queue.

Also, a new process is always inserted at the tail of the top level queue with the assumption that it would be a short time consuming process. Long processes will automatically sink to lower level queues based on their time

consumption and interactivity level. In the multilevel feedback queue, a process is given just one chance to complete at a given queue level before it is forced down to a lower level queue.

Scheduling parameters [edit]

In general, a multilevel feedback queue scheduler is defined by the following parameters:^[4]

- The number of queues.
- The scheduling algorithm for each queue which can be different from FIFO.
- The method used to determine when to promote a process to a higher priority queue.
- The method used to determine when to demote a process to a lower priority queue.
- The method used to determine which queue a process will enter when that process needs service.

See also [edit]

- Multilevel queue
- Lottery scheduling
- Scheduling (computing)
- Fair-share scheduling
- Round-robin scheduling

References [edit]

- ↑ <http://pages.cs.wisc.edu/~remzi/solaris-notes.pdf>
- ↑ Operating Systems and Middleware: Supporting Controlled Interaction, Max Hailperin, 2007, p. 61
- ↑ [Scheduling: The Multi-Level Feedback Queue](#)
- ↑ ^a ^b Silberschatz, Abraham; Galvin, Peter Baer; Gagne, Greg (2008). *Operating system concepts* (8th ed.). Hoboken, N.J.: Wiley. p. 198. ISBN 0470128720.
- ↑ Arpaci-Dusseau, Remzi H.; Arpaci-Dusseau, Andrea C. (2014). *Operating Systems: Three Easy Pieces [Chapter Multi-level Feedback Queue]* (PDF). Arpaci-Dusseau Books.
- ↑ Kleinrock, L.; Muntz, R. R. (July 1972). "Processor Sharing Queueing Models of Mixed Scheduling Disciplines for Time Shared System". *Journal of the ACM* **19** (3): 464–482. doi:10.1145/321707.321717 .

Categories: Processor scheduling algorithms

This page was last modified on 9 August 2015, at 22:34.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

