HiperDispatch Logical Processors and Weight Management - Part 1

Fabio Massimo Ottaviani - EPV Tehnologies

In the last few years, the power and number of the physical processors available on mainframe hardware has greatly increased, allowing the concentration of a much higher number of LPARs than before on a single machine.
A side effect of this high number of LPARs is an increase of the number of defined logical processors compared to the number of physical CPs.
These factors tend to reduce the probability for a logical processor to be re-dispatched to the same physical processor and therefore reuse instructions and data previously loaded in the Level 1 cache (the amount of cache memory dedicated to each processor).
A L1 cache miss will cause data and instructions to be loaded from the Level 2 cache (the cache memory shared among all the physical processors packaged in a book). Performance degradation and overhead occurs when this happens because the access to L2 cache will require more CPU cycles to be performed. However if the logical processor has been dispatched to a different physical processor which belongs to a different book, the required instructions and data have to be loaded from the previously used L2 cache and performance degradation and overhead can be much worse.
HiperDispatch has been designed to minimise the number of L1 cache misses and, if a L1 cache miss occurs, to maximise the probability of finding instructions and data in the L2 cache of the book where the logical processor is dispatched.
To reach this goal, a new weight called polarization weight has been introduced; polarization weight is a key element in the HiperDispatch design because it is the way z/OS uses to give PR/SM indications on how logical processors should be dispatched to the physical processors.
This paper, using real life examples, will discuss:

- PR/SM and IRD logical processors and weight management (Part 1);
- HiperDispatch logical processors and weight management (Part 2).