

## TPF5 Summary

This is an attempt to summarise what we currently know, or believe we know, about TPF5. IBM has not yet announced the product and refers to it internally as NBR (Next Big Release). If/when it is announced, its name might not be TPF5 (e.g. Websphere TPF?). When something is announced it may or may not contain everything below.

There is nothing revolutionary about the changes in TPF5, but it is a significant evolutionary step to realign TPF with the more advanced current mainframe hardware architecture capabilities. The major step is moving from 31 bit to 64 bit addressing for main memory. In addition, a move towards massively parallel processing is taken with the extension to 32-way tightly-coupled and 32-way loosely-coupled support. Such architectural change brings little functional enhancement to the system, but does necessitate a major rewrite of the control program kernel and surrounding software. IBM is reputed to be investing \$100M in this effort.

It is claimed that have also been a number of changes made to improve the performance of the system in some areas, but nevertheless it is expected there will be a 5% to 10% drop in overall performance compared to TPF4.1

### **Improvements**

Support for

- Very large amounts of main memory (heap, globals, and VFA) available through 64 bit addressing
- Up to 32 tightly-coupled I-streams in a single processor image
- More flexible I-stream scheduling for ECB's

Primarily, this means that a single processor multi-I-stream TPF5 system can be extremely powerful.

### **Concerns**

- Support is more complex and possibly more expensive as an installation now requires 4 operating systems: TPF5, Linux, z/OS, and VM.
- To take advantage of 64 bit addressing, assembler programs must be modified to use 8 byte addresses and 64 bit instructions.
- Loosely-coupled systems gain little or nothing from very large global and/or VFA areas.
- By how much will the software licence costs escalate?
- The cost and complexity of a TPF5 migration.

### **General Information about TPF5**

- General availability is 3<sup>rd</sup> quarter (September?) 2005.
- TPF4.1 will be supported up to 1 year after TPF5 general availability.
- The last TPF4.1 PUT with functional enhancements will be PUT 18.

- TPF4.1 PUT 19/20 will only contain fixes and, for example, migration tools.
- There will be no more TPF4.1 PUT's after PUT 20. Fixes will then only be published on the internet.
- To get to TPF5 the user must have TPF4.1 PUT 16 or later.

### **Some facilities made obsolete**

- Support for 24 bit addressing mode (programs and globals)
- Obsolete software (old communications, e.g. SLC) and obsolete hardware (3705, 3350, etc) support is being dropped
- Fast link macros (introduced for performance reasons) dropped: all macros now issue an SVC.

### **31 and 64 bit addressing is supported.**

- The TPF5 CP will run in 64 bit mode. All CP user code must use 64 bit instructions.
- ECB programs can run in 31 bit or 64 bit addressing mode.
- New heap storage available in 64 bit mode for ECB programs and the CP. New macros/C-calls will be supplied to use it.
- New format-2 globals<sup>1</sup> can be defined for 31 and/or 64 bit addressing. Enables enormously big tables to be defined in 64 bit mode.
- 64 bit storage only accessible by programs running in 64 bit mode.
- Storage usually allocated in 1MB lumps instead of 4KB lumps.

### **Keypoints are changed**

- Keypoints can be up to 48KB.
- CTKA will contain up to 8 different storage configurations and TPF5 will pick the first one that fits the physical configuration.
- CTKD is now a user keypoint. Formerly it was used for SLC (now obsolete).

### **ECB program handling is changed.**

- Changes made to program linkage (ENTR/BACK) to improve performance of C programs.
- Programs now generally loaded on demand<sup>2</sup> and no longer defined as core or file resident.
- Assembler programs can be greater than 4KB.
- Assembler programs can be made baseless (R8 not needed) by using the Branch Relative z/Architecture instructions.
- An entry (ECB/program) can be defined so that it can be dispatched on any available I-stream at any time instead of being forced to remain on the same I-stream.

### **Application environment is being changed from z/OS to Linux**

- The emphasis is now on developing programs in C/C++.
- The GCC<sup>3</sup> compiler will be used.
- All C programs need to be recompiled and will run in 64 bit mode.

### **VFA handling changed**

- VFA uses 64 bit addressing so it can enormous.
- There is only one block size and list – 4KB. Every record in VFA is placed into a separate 4KB block.

### **Miscellaneous changes**

- Lots of minor enhancements in diagnostics, such as storage is tagged with names and printed dump output better formatted.
  - Pools can be available in 1052 state.
- 

<sup>1</sup> Format-1 globals are the current TPF4.1 globals.

<sup>2</sup> We are not certain what this actually implies.

<sup>3</sup> GNU Compiler Collection. Compiler freely available under GPL. See [gcc.gnu.org](http://gcc.gnu.org)