

## So You've Moved to an i5 Platform?

by [Stephen Linam and Becky Schmieding](#)

Here's what you need to know about system firmware and concurrent firmware maintenance.

*Published August 2005*

As you transition to IBM's iSeries Power5 hardware platform (i5), you'll notice changes. Not only does the i5 have some new concepts, but the documentation and menus also use different terminology. This article explains these new terms and concepts. In addition, it introduces you to a newly provided function called "concurrent firmware maintenance." Let's first look at the components of system firmware.

### ***The System Firmware Components***

Previously, all you really needed to know about your AS/400 was its operating system, OS/400, which included the Systems License Internal Code (SLIC). However, as you move to the new i5 systems, you'll need to get acquainted with the system components that comprise "firmware." What is firmware? [A simplistic definition of firmware is found on VoxTechnologies Corporation's glossary Web page:](#) "Firmware is low level software that controls the system hardware." But the components of i5 firmware do a lot more than this simple definition suggests (see Figure 1).

- The Flexible Service Processor (FSP) firmware provides diagnostics, initialization, configuration, run-time error detection, and correction.
- The Power Hypervisor (PHYP) firmware, which is based on the iSeries hypervisor, provides VLAN, virtual I/O, and subprocessor partitioning support.
- The Partition Firmware (PFW) supports the pSeries Power Architecture Platform Requirements+ (PAPR+) interface.
- The Hardware Management Console (HMC) firmware provides converged platform configuration, management, and services.
- The System Power Control Network (SPCN) firmware interfaces with bulk power for power monitoring and control.
- The Bulk Power Control (BPC) firmware controls each bulk power unit in CEC and towers. This firmware is specific to the i595 system.

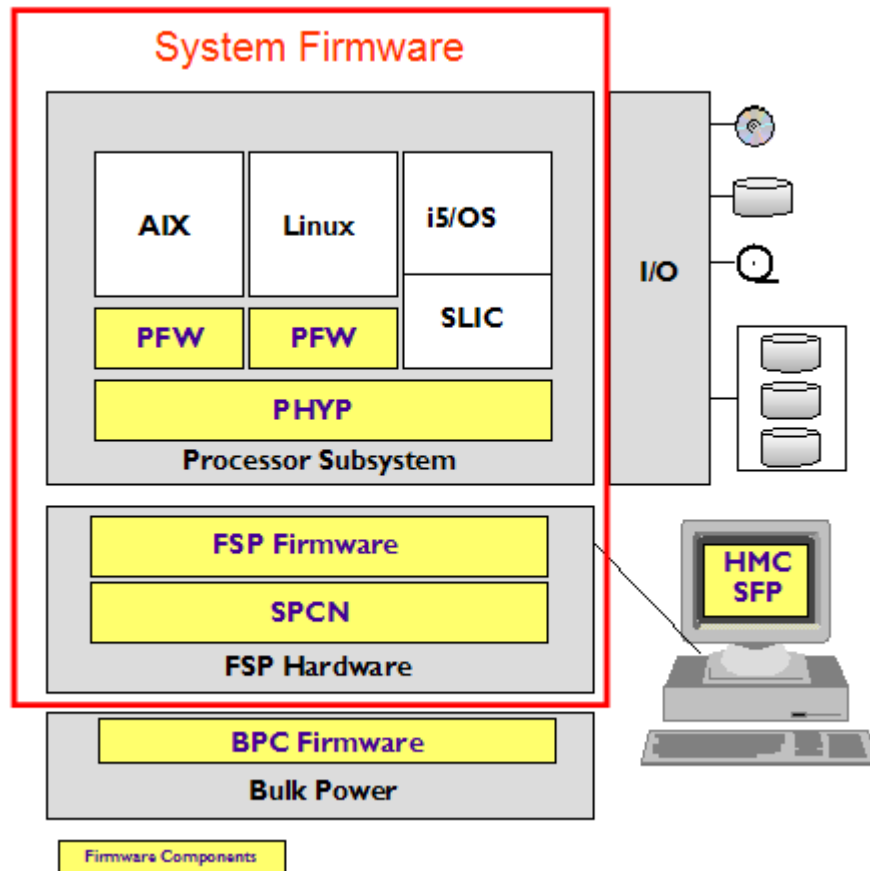


Figure 1: These are the system firmware components. (Click images to enlarge.)

Now, let's look at some of the terminology used by writers, developers, and service personnel when talking about the system firmware.

### ***Firmware Terminology Defined***

To really discuss firmware with anyone, you need to know the terminology. The following is an explanation of the most common terms and concepts.

The first set of terms is "p-side" and "t-side." These are analogous to i5/OS A side and B side, respectively. The p-side is the permanent side; it usually contains the older level of firmware code. The t-side is the temporary side; it usually contains the newer level of firmware code.

You also need to know the difference between a "fixpack" and a "release." A fixpack is a means by which fixes to firmware are provided in an "installable entity." Every fixpack contains PHYP, PFW, and FSP firmware, and fixpacks usually contain multiple fixes. A release is an "installable entity," and it is the means by which IBM delivers new function in the firmware. Releases are similar to fixpacks, but they contain new function. A release is distributed and installed similarly to a fixpack. The installation of a firmware release is called an "upgrade," while the installation of a fixpack is called an

"update." There are two types of firmware (fixpack) updates:

- Inband update: Firmware fix updates are initiated by the i5/OS PTF process.
- Outband update: Firmware fix updates are initiated by the HMC.

The installation process of a fixpack has five steps:

1. You must "retrieve" the fixes by obtaining them from a repository location.
2. You must install the firmware on the system. Installing is the process that writes on flash the code that is to be loaded into memory at the next system IPL. What is flash? Flash is a nonvolatile storage location in FSP where firmware is contained for the next activation.
3. You must IPL. There are two types of IPL, partition IPL and system IPL. Partition IPL is the IPL of an operating system (e.g., i5/OS) within a partition. The IPL of a partition does not impact the status of any other partition or the system itself. System IPL, sometimes called a "white button" IPL, requires all partitions to be ended and the system to be rebooted.
4. You must "activate" the firmware by loading the firmware code level to be run into system memory. When people talk about an "activated level," they are referring to the level of firmware code running in system memory.
5. You must request that the system "accept" the firmware by copying the installed level of firmware code to the p-side flash. When people talk about an "accepted level," they are referring to the lowest firmware code level on the two flash sides; usually, this is on the p-side code level.

As with i5/OS, you have the opportunity to rollback to a previous level. In firmware, this is called "remove," which means to back off an installed firmware code level that was previously accepted by copying the flash from the p-side to the t-side. This function is currently available only through the advanced function path, not through the firmware install wizard. We'll cover this more in a future article.

Next, firmware fixes can be one of the following types:

- Concurrent: Fixes can be applied and activated concurrently (i.e., no system IPL is required).
- Deferred: Fixes can be applied but will not go into effect until the next system IPL. An example of a firmware fix that might be delayed is a fix to the IPL path.
- Disruptive: When a *release* or *fixpack* is installed, a system IPL will be required.

These definitions should give you a fairly good understanding of some of the basic concepts of firmware installation. But let's look at some additional firmware concepts.

## ***Firmware Distribution Concepts***

All firmware install fixpacks are labeled as follows: PPNNSSS\_FFF\_DDD:

- PP = package identifier; 01 = managed system and 02 = power subsystem.
- NN = machine type model group; SF = system firmware and BP = bulk power.
- SSS = fixpack code stream indicator, i.e., a GA level (e.g., 230)

[Reprinted with permission from MC Mag Online, published by MC Press, LP; <http://www.mcpressonline.com>.]

- FFF = fixpack number for current fixpack level
- DDD = fixpack number for the last disruptive fixpack level

A fixpack consists of a fixpack cover letter XML file and a firmware code RPM file (for example, 01SF230\_001\_001.xml and 01SF230\_001\_001.rpm). Firmware fixpacks are installed through the HMC via the Licensed Code Update menu.

## **Current Firmware Maintenance (CFM)**

You may have heard of the concept of "current firmware maintenance (CFM)," but what really is CFM? Current firmware maintenance is the ability to deploy firmware updates (remember the definition of updates; this does not mean new releases of firmware) on a running system without rebooting partitions or disturbing applications. CFM covers only the components of Partition Hypervisor (PHYP), Flexible Service Processor (FSP), and Partition Firmware (PFW).

Bulk Power Control (BPC) firmware updates are done using a separate fixpack and are always concurrent with respect to the server. HMC firmware updates are also done using a separate fix process and are also concurrent with respect to the servers that the HMC manages. To employ CFM is as easy as selecting a new option on the existing firmware update menus (we'll cover this in a future article). However, there are some requirements to implement CFM.

The first requirement is that you must have system firmware fixpack SF230\_126 installed on your system, as this is the release and fixpack that delivered CFM.

Many of you like to use i5/OS PTFs to install all system fixes; however, if you want to use CFM, you will need to modify your process. CFM can only be accomplished through the HMC. In other words, inband update support for firmware fixes (see above for definition) does *not* support CFM. This means that every time you install firmware fixes through the i5 PTF method, you will need to IPL your system. Is it any wonder IBM recommends customers use their HMCs to install fixes?

## **How Does CFM Work?**

Firmware fixpacks can be installed through the HMC. There are two steps to the process of installing fixes:

1. You must apply the firmware (think of this as updating flash).
2. You must activate the firmware (think of this as allowing the system to use the new firmware).

CFM is being able to apply and activate the firmware without IPLing the system.

You are probably wondering whether all firmware fixpacks can be installed concurrently. Unfortunately, the answer is no. Some fixes can be applied and activated concurrently, others can be applied but not activated until the next IPL (i.e., deferred), while still others cannot be applied without an IPL (i.e., disruptive). As we said above, most fixpacks will contain multiple fixes; for a deferred fixpack, any concurrent fixes will be activated immediately, and only the deferred fixes will wait until the next IPL to be activated. For disruptive fixpacks, none of the fixes will be activated until the system IPL. IBM makes every attempt to make firmware fixpacks non-disruptive, but sometimes firmware fixes need to be applied that cannot be activated without system disruptions. IBM's creation of a

[Reprinted with permission from MC Mag Online, published by MC Press, LP; <http://www.mcpressonline.com>.]

disruptive fixpack means that IBM has determined that the risk or consequence of a problem clearly outweigh the inconvenience caused by a disruptive activation.

The developers will tell you that "fixpacks are sequential and cumulative, but not mandatory sequential." But what does that mean? See Figure 2 below. If Level 80 is the GA level of code, then Level 81 is the GA level of code plus some set of fixes. And Level 82 is the Level 81 level of code plus some fixes. And so on. While the fixpacks are cumulative (i.e., each is a superset of the previous), you do not need to install them in order; you can go directly from Level 81 to Level 84 without having to install Levels 82 and 83. By doing this, you get the fixes that were added to Levels 82, 83, and 84.

But more importantly, what is the implication of this on CFM? It implies that the ability to apply and activate firmware concurrently depends both on the current firmware level on the system and the fixpack. Let's look again at Figure 2. As you can see, fixpack Levels 81, 82, and 84 are concurrent (i.e., they don't require IPL). However, Level 83 is a disruptive fixpack. If your system currently has Level 81 installed and you want to move to Level 82, you can install Level 82 using the concurrent firmware method (i.e., you do not need to IPL the system to install Level 82 firmware). However, if you want to move from Level 81 to Level 83, you would incur an IPL because Level 83 is a disruptive fixpack. Once you're on Level 83, the move to Level 84 fixpack would again be concurrent. However, if you were on Level 81 and wanted to move to Level 84, which is marked concurrent, you would incur an IPL because the fixes from Level 83, which are not on your system, require an IPL even though the fixes added in Level 84 do not.

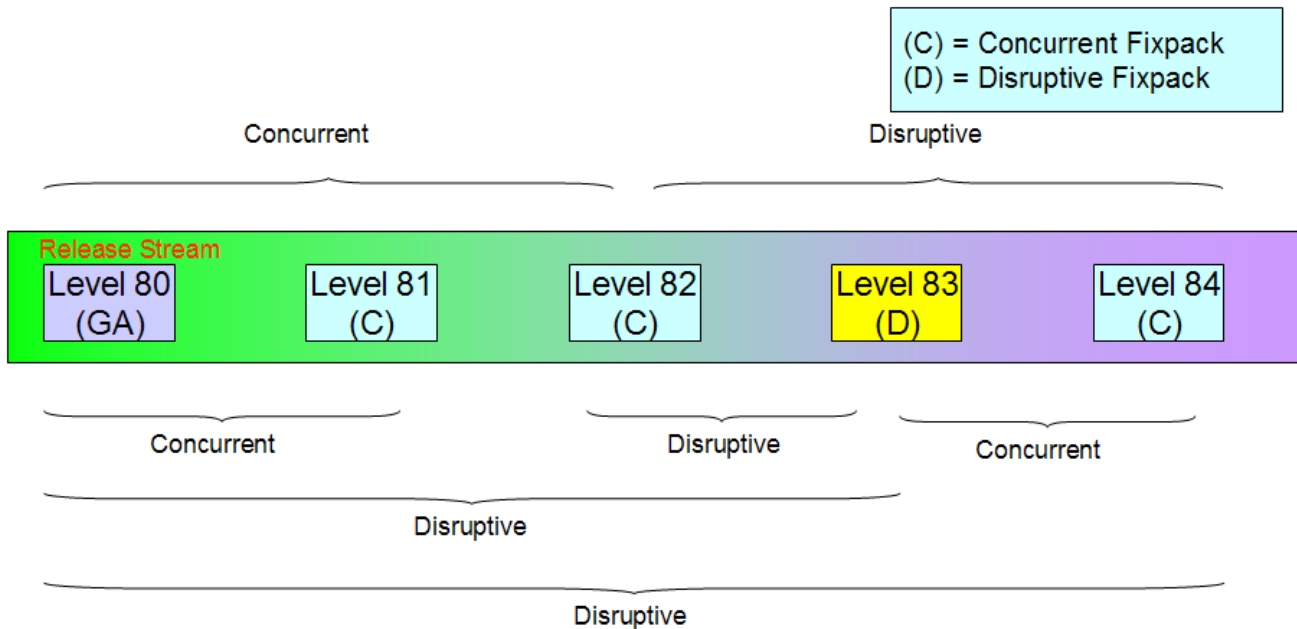


Figure 2: Figure 2: Understand the difference between concurrent and disruptive firmware updates.

Finally, current firmware updates are only available within a release boundary; in other words, all firmware release updates are disruptive and are installed via a different process than fixpacks.

[Reprinted with permission from MC Mag Online, published by MC Press, LP; <http://www.mcpressonline.com>.]

## ***CFM--It's a Good Thing***

We hope this article has given you a better understanding of firmware concepts and terminology. We also hope that you now understand what CFM is and what it isn't. And finally, we hope that you understand the value CFM brings in reducing (but not eliminating) the downtime associated with maintaining firmware. Stayed tuned: In our next article, we'll discuss how to install firmware releases and fixpacks as well as how to determine if the fixpack installation will be disruptive or concurrent.

**Stephen Linam** is a Senior Technical Staff Member in IBM Global Firmware Development. He was an architect of the P5 Concurrent Firmware Maintenance Support. Before that, he worked in architecture and development for P5 and pSeries P4 firmware and the AIX operating system. Before joining IBM in 1984, he developed applications in RPG. Stephen can be reached at [slinam@us.ibm.com](mailto:slinam@us.ibm.com).

**Becky Schmieding** is an IBM Certified Executive Project Manager at IBM Rochester on the Product Definition Team (PDT) responsible for the E2E upgrade process. She is a noted speaker and author on PC5250, iSeries Navigator, and the iSeries Access family of products. She can be contacted at [schmieb@us.ibm.com](mailto:schmieb@us.ibm.com).