IBM

server

New IBM xSeries Firmware and Device Driver Update Naming Convention Whitepaper

Written By:	Mike Nolterieke
Revision Level:	06
Last Revised:	6/21/2006 9:31 PM

Table of Contents:

Introduction

IBM xSeries is changing the way we build and release firmware and device driver updates. We will no longer be using a part number as the file name of the updates. Instead, an architecture has been developed that will allow customers an easier method to identify an update and its version based on the file name itself.

Customers will start see IBM xSeries updates being released over the next few months that follow this architecture.

This whitepaper provides the following information:

- Why we are changing
- An overview of the new architecture
- Possible values of the architected fields where applicable
- Examples of new file names using the new architecture

Why Make The Change?

Currently most xSeries firmware and driver updates are released to the IBM web site as a part number, 46Y3541.exe for example. This has been an issue for our customers and we have received several requests to change it.

Customer Complaints

Customers have been complaining about us using part numbers for a long time. Their main complaints are:

- They really mean nothing
- They are not at all descriptive/intuitive
- They are not displayed/used in any of our VPD
- All of the VPD that IBM displays uses build numbers/ID's
- It is very hard for internal IBM and our customers to correlate a part number with a specific build if/when there is a problem
- There is no easy way to determine that what we thought was released really was

Architecture Overview

Names have a maximum defined length of 64 characters.

All characters in the name must be lowercase. No upper case characters are allowed.

No blank spaces or "special" characters (See Appendix A: Special Characters) are allowed in the name.

The name will be constructed by concatenating the fields defined below in order. During the concatenation, all fields will be separated by a single underscore "_". There is no "_" character between the last field and the extension.

All fields are mandatory and must be used while constructing the name. This allows for tools to be developed that can easily parse the names.

The contents of the defined fields are not absolute. New values can and will be added as needed.

Architecture Exceptions

1. Linux device drivers which are released as RPM's normally already have a long, descriptive name. Therefore, they are not required to implement this architecture. However, it is strongly recommended that this architecture be implemented for RPM's where possible.

2. Vendor updates, such as BladeCenter I/O Module updates, that IBM does not develop or repackage are not required to implement this architecture. Most of these updates already have descriptive names with the version number in them and/or follow a naming convention defined by the vendor which is also used in updates for other non-IBM hardware they sell.

Architected Fields

Field 1 – Vendor:

This field identifies the vendor that provided the update using the company's stock symbol. This field is variable in length from 2 to a maximum of 5 characters.

For vendors that do not have an official stock symbol, an acceptable abbreviation will be used.

Field 2 – Type:

This field defines the type of file/update. This field is variable in length from 2 to a maximum of 3 characters. There are currently only six options available for this field as specified in the following table:

Type	Code
Device Driver	dd
Firmware	fw
Tools / Utilities	utl
Libraries	lib
Services	svc
Software	sw
Boot and Code	bt
ASU Definition File	def

Field 3 – Component:

This field defines the component for the update. This field is variable in length from 2 to a maximum of 7 characters. Updates that are longer than 7 characters will be abbreviated according to the following table:

Note: This table is current as of the writing of this document. Items in this table are subject to change and additional items may be added as well.

Device	Abbreviation
Network Interface Controller	nic
Integrated Systems Management Processor	ismp
Integrated Systems Management Processor 2	ismp2
Baseboard Management Controller	bmc
Remote Supervisor Adapter	rsa
Remote Supervisor Adapter II	rsa2
Management Module	mm
Advanced Management Module	amm
BladeCenter Switch Module	bcsw
BladeCenter I/O Module	bcio
Host Bus Adapter	hba
BIOS	bios
ROM Diagnostics	diag
Dynamic System Analysis	dsa
Video	video
Advanced Settings Utility	asu
Advanced Settings Utility Definition File	asudef
OSA SMBridge	smbr
Hard Disk Drive	hdd
Tape Drive	tape
Fibre Channel	fc
BladeCenter Switch	bcsw
System Update Pack	sup
iSCSI	iscsi
Automatic Server Restart	asr
IPMI	ipmi
IPMI Mapping Layer	mlayr
AMD Driver Pack	amddp
Electronic Service Agent	esa
Configuration Utilities	cutil
IBM Director	dir
IBM Director Extensions	direx

IBM Director Plug-ins	dirpu
IBM Director Upward Integration Modules	dirui
Remote Deployment Manager	rdm
ServerGuide	sg
ServerGuide Scripting Toolkit	sgtk
UpdateXpress CD1	uxcd1
UpdateXpress CD2	uxcd2
UpdateXpress CD3	uxcd3
Update Manager	um
ServeRAID Application CD	srapp
ServeRAID Support CD	srsupp
ServeRAID Configuration Utility	cutil
ServeRAID SCSI	ips
ServeRAID SATA/SAS	aacraid
ServeRAID 8e SAS (Razor)	hr94xx
ServeRAID 8e SATA (ICH7R)	adpahci
ServeRAID 7e SATA (Silicon Image 3512)	aarsi3x
ServeRAID 7e SATA (ICH5R/ICHS)	aarich
Adaptec aic79xx SCSI	aic79xx
Adaptec adp94xx SAS	adp94xx
Message Passing Technology	mpt
LSI 1020/1030 SCSI	mptscsi
LSI 1064/1064e SAS	mptsas
LSI IDE RAID	megaide
DS4000	ds4k
Remote Disk CLI	rdcli
CPLD	cpld

		New IBM xSeries Firmware and Device Driver Update Naming Convention Whitepaper
IBM	**e server**	**Page 9 of** 17 Written By: Mike Nolterieke

Field 4 – Version:

This field defines the unique VPD string and/or version number. This field is variable in length from 3 to a maximum of 16 characters. This field may contain any characters that are not considered "special" (See Appendix A: Special Characters).

All components built by IBM will use the string which matches what our systems display as the VPD minus the country code for that component. If the component is a third party based update, the string "-pq" will be appended to the operating system string in field 5.

For example, IBM built BIOS for the x345 displays GEE132AUS as the VPD. Therefore, the string GEE132A will be used in this field for all of the deliverables. Examples of the deliverables are as follows:

> DOS Based Image, ReadMe, and XML Descriptor:
> > ibm_fw_bios_gee132a_anyos_i386.img
> > ibm_fw_bios_gee132a_anyos_i386.txt
> > ibm_fw_bios_gee132a_anyos_i386.xml
>
> WFlash Based Image, ReadMe, and XML Descriptor:
> > ibm_fw_bios_gee132a_windows_i386.exe
> > ibm_fw_bios_gee132a_windows_i386.txt
> > ibm_fw_bios_gee132a_windows_i386.xml
>
> Windows Third Party Based Image, ReadMe, and XML Descriptor:
> > ibm_fw_bios_gee132a_windows-pq_i386.exe
> > ibm_fw_bios_gee132a_windows-pq_i386.txt
> > ibm_fw_bios_gee132a_windows-pq_i386.xml
>
> LFlash Based Image, ReadMe, and XML Descriptor:
> > ibm_fw_bios_gee132a _linux_i386.sh
> > ibm_fw_bios_gee132a _linux_i386.txt
> > ibm_fw_bios_gee132a _linux_i386.xml
>
> Linux Third Party Based Image, ReadMe, and XML Descriptor:
> > ibm_fw_bios_gee132a_linux-pq_i386.sh
> > ibm_fw_bios_gee132a_linux-pq_i386.txt
> > ibm_fw_bios_gee132a_linux-pq_i386.xml

ASU Definition File, ReadMe, and XML Descriptor:
ibm_fw_asudef_gee132a_anyos_i386.exe
ibm_fw_asudef_gee132a_anyos_i386.txt
ibm_fw_asudef_gee132a_anyos_i386.xml

IBM built components that produce multiple images for the same operating system, such as the DOS update for Diagnostics, will append a dash "-" followed by the diskette number to the build ID. For example:

DOS Based Images, ReadMe, and XML Descriptors:
ibm_fw_diag_geyt12a-1_anyos_i386.img
ibm_fw_diag_geyt12a-2_anyos_i386.img
ibm_fw_diag_geyt12a-1_anyos_i386.xml
ibm_fw_diag_geyt12a-2_anyos_i386.xml
ibm_fw_diag_geyt12a_anyos_i386.txt

Any components that have multiple formats for the same operating system, such as DOS diskette and ISO images, which also require separate ReadMe files will append a dash "-" followed by some short descriptive text that differentiates the files to the operating system field. For Example:

DOS Based BMC diskette image, ISO image, ReadMe's, and XML Descriptor:
ibm_fw_bmc_z2bt04a_anyos_i386.img
ibm_fw_bmc_z2bt04a_anyos_i386.txt
ibm_fw_bmc_z2bt04a_anyos_i386.iso
ibm_fw_bmc_z2bt04a_anyos-cd_i386.txt
ibm_fw_bmc_z2bt04a_anyos_i386.xml

IBM built components that have ODM crisis recovery deliverables will append "cr" to the VPD string. For example:

ODM DOS based BIOS crisis recovery image:
ibm_fw_bios_pxe124acr_anyos_i386.img

Vendors define their own format for this field and may use version numbers that do not include any special characters (See Appendix A: Special Characters) such as 7.10, 2.2.21, etc., however, if the component is a third party based update, the string "-pq" must be appended to the operating system string in field 5. For example:

Broadcom Third Party Based Windows NIC Firmware:
brcm_fw_nic_3.45.12_windows-pq_i386.exe

If more information is needed than is provided in fields 1 – 3 to properly identify hardware, a prefix will be added to the build ID and/or version number with a short descriptive string followed by a dash. For example:

Broadcom Windows NIC Device Driver:
brcm_dd_nic_3.45.12_windows_i386.exe

Broadcom Windows NIC Device Driver for Chipset 5704 Only:
brcm_dd_nic_5704-3.45.12_windows_i386.exe

Broadcom Linux NIC Device Driver for Chipset 5721 Only:
brcm_dd_nic_5721-3.45.12_linux_i386.rpm

Field 5 – Operating System:

This field defines the supported operating system. This field is variable in length from 3 to a maximum of 15 characters. This field is used to define the minimum or specific level/version of an operating system that is required. If the component is a third party based update, the string "-pq" will be appended to the string used in this field.

Below is a hierarchy of operating systems and releases. The value selected for the file name is the highest level that applies to the update.

Note: This hierarchy is current as of the writing of this document. Items are subject to change and additional items may be added as well.

- anyos = Supports all or is OS independent
  - dos = DOS
  - windows = Generic Windows
    - winsrvr = Generic Windows Server OS'
      - win2k = Windows 2000
        - win2ksp3 = Windows 2000 SP 3
      - win2k3 = Windows 2003
        - win2k3r2 = Windows 2003 R2
      - win2k3dc = Windows 2003 DataCenter
    - winwrk = Generic Windows Workstation OS'
      - winxp = Windows XP
  - winpe = Windows PE
    - winpe2k5 = Windows PE 2005
  - linux = Generic Linux
    - linux2.4 = Generic Linux 2.4 Kernel
    - linux2.6 = Generic Linux 2.6 Kernel
    - sles = Generic SUSE Linux Enterprise
      - sles8 = SUSE Linux Enterprise Server 8.0
      - sles9 = SUSE Linux Enterprise Server 9.0
        - sles9sp3 = SUSE Linux Enterprise Server 9 SP 3
        - sles10 = SUSE Linux Enterprise Server 10.0
    - nld = Generic Novell Linux Desktop
      - nld9 = Novell Linux Desktop 9.0
      - nld10 = Novell Linux Desktop 10.0
    - rhel = Generic RedHat Enterprise Linux
      - rhel3 = RedHat Enterprise Linux 3.0
        - rhel3u4 = RedHat Enterprise Linux 3 Update 4
      - rhel4 = RedHat Enterprise Linux 4.0
      - rhel5 = RedHat Enterprise Linux 5.0

- netware   = Generic Novell NetWare
    - netware65   = Novell NetWare 6.5
- solaris   = Sun Solaris
    - solaris10   = Sun Solaris 10.0
- vmware   = VMWare
    - vmwesx   = VMWare ESX
        - vmwesx25   = VMWare ESX 2.5
        - vmwesx3   = VMWare ESX 3.0
- xen   = Xen

Field 6 – Processor Architecture

This field defines the processor architecture. This field is variable in length from 3 to a maximum of 7 characters.

Below is a hierarchy of processor architectures. You should select the highest level that applies to your update.

- ~~anycpu~~   = ~~Any CPU~~
- noarch   = No architecture
  - i386   = Any 32 bit compatible running in 32-bit modes (includes AMD64 and EM64T)
    - ~~intel32~~   = ~~Any Intel 32 bit (including 64 bit extensions)~~
      - ~~intl386~~   = ~~Intel 32 bit only~~
      - ~~em64t~~   = ~~Intel 64 bit extensions for i386 only~~
    - ~~amd32~~   = ~~Any AMD 32 bit (including 64 bit extensions)~~
      - ~~amd386~~   = ~~AMD 32 bit only~~
      - ~~x64~~   = ~~AMD 64 bit extensions for i386 only~~
  - i586   = Any Pentium compatible running in 32-bit mode
  - i686   = Any P4 class compatible running in 32-bit mode
  - x86-64   = Any AMD64 or Intel EM64T running in 64-bit mode
  - 32-64   = Any Intel or AMD 32-bit or 64-bit processor
  - ia64   = Intel 64-bit Itanium family only
  - ~~amd64~~   = ~~AMD 64 bit only~~
  - anyppc   = Any Power based
    - ppc64   = Power based 64 bit only
  - sparc   = Any SPARC based

Notes:

The lines that are gray and have the strikethrough have been deprecated and will no longer be used. They are being left in the document for completeness and backwards compatibility.

The distinction between Intel and AMD will be made with the supported system list, i.e. the update applies to the system (Intel or AMD) that it is listed under on the IBM web site.

The 32-64 entry in the hierarchy is provided for packages that include both 32-bit and 64-bit drivers in the same file.

File Extension:

The file extension remains the same as it is today. It is variable in length with either 3 or 4 characters. The extension will always be separated from the rest of the name by a period ".". The following table summarizes all known extensions:

Type	Extention
DOS disk image	.img
CD Image	.iso
Windows executable	.exe
Linux shell script	.sh
Linux RPM	.rpm
Linux tar file	.tar
Linux tar gzip	.tgz
Zipped file	.zip
Text file (readme)	.txt
Adobe Portable Document Format	.pdf
MS Word document	.doc
XML Descriptor	.xml
Packet file	.pkt
Linux GZip File	.gz
Cabinet File	.cab
HTML File	.htm

Naming Examples Using the Architecture

IBM DOS Based BIOS, ReadMe, and XML:
>   ibm_fw_bios_gee132a_anyos_i386.img
>   ibm_fw_bios_gee132a_anyos_i386.txt
>   ibm_fw_bios_gee132a_anyos_i386.xml

IBM RedHat Enterprise Linux (3 or 4) Based BIOS:
>   ibm_fw_bios_gee132a_rhel_i386.sh

IBM DOS Based Diagnostics (Disks 1 and 2), ReadMe, and XML:
>   ibm_fw_diag_geyt12a-1_anyos_i386.img
>   ibm_fw_diag_geyt12a-2_anyos_i386.img
>   ibm_fw_diag_geyt12a_anyos_i386.txt
>   ibm_fw_diag_geyt12a-1_anyos_i386.xml
>   ibm_fw_diag_geyt12a-2_anyos_i386.xml

Broadcom DOS Based NIC Firmware:
>   brcm_fw_nic_2.34_dos_i386.img

Broadcom Windows NIC Device Driver:
>   brcm_dd_nic_3.45.12_windows_i386.exe

Broadcom Windows NIC Device Driver for Chipset 5704 Only:
>   brcm_dd_nic_5704-3.45.12_windows_i386.exe

Broadcom Linux NIC Device Driver for Chipset 5721 Only:
>   brcm_dd_nic_5721-3.45.12_linux_i386.rpm

IBM Management Module Firmware:
>   ibm_fw_mm_bret82h_anyos_anycpu.zip

IBM RSA II Windows Based Firmware:
>   ibm_fw_rsa_goet24a_windows_i386.exe

IBM BMC Windows Based Firmware:
>   ibm_fw_ismp_kpbt11a_windows_i386.exe

Appendix A: Special Characters

The following is a list of "special" characters that must not be used within any of the fields defined in this architecture

- ~ (TILDE)
- ` (BACKWARDS SINGLE QUOTE)
- ! (EXCLAMATION POINT)
- @ (AT SYMBOL)
- # (POUND SIGN)
- $ (DOLLAR SIGN)
- % (PERCENT SIGN)
- ^ (CARAT)
- & (AMPERSAND)
- * (ASTERISK)
- ((OPEN PAREN)
-) (CLOSE PAREN)
- + (PLUS SIGN)
- = (EQUAL SIGN)
- { (OPEN BRACE)
- } (CLOSE BRACE)
- [(OPEN BRACKET)
-] (CLOSE BRACKET)
- | (VERITICAL BAR)
- \ (BACKSLASH)
- : (COLON)
- ; (SEMICOLON)
- " (DOUBLE QUOTE)
- ' (SINGLE QUOTE)
- < (LESS THAN)
- > (GREATER THAN)
- ? (QUESTION MARK)
- / (SLASH)
- , (COMMA)
- _ (UNDERSCORE)

Notes:

The "_" character must only be used for constructing the string as defined in the Architecture Overview section.

A blank space is also considered a "special" character and is not allowed.