

Comments on a Second Reading of CIM_Printer

Rick Landau
Dell Inc.
Richard_Landau@dell.com
2005/08/11

1 Overview

Here is a revised list of my comments after review on the concall today.

The comments fall into several general categories.

- Structural problems: the CIM_Printer element seems to be a blending of the Printer MIB and IPP models of "printer." It contains many properties that relate to the behavior of print services w.r.t. print jobs, properties that don't fit well within the "printer" object. I think we want the CIM model of the several printing-related objects to come closer to the PWG model, where the printer device is described in the Printer MIB, and Print Service, Queue, and Job are separate objects with their own properties.
- Vague or meaningless properties: the CIM_Printer element contains properties that cannot be implemented reliably or interoperably. In some cases, their semantics are vague or ambiguous for fundamental reasons; some others may be rescued with improved semantic descriptions.
- Errors in MappingStrings or ModelCorrespondence attributes: a few MappingStrings point to the wrong MIB.
- Improvements needed to MappingStrings or ModelCorrespondence attributes: additional MappingStrings or ModelCorrespondences would help implementers and future editors.
- Read/write attributes: the Device_Printing.mof does not currently identify what properties might be writable to consumers; some clearly are not, such as status. CIM properties by default are read-only. Printer MIB, at least, identifies some objects as potentially writable.
- Mutable and Modifiable attributes to be added to descriptions: documenting mutability and modifiability (same as write, above) would be useful guidance to developers.

2 Details

2.1 *Structural Suggestions*

- Move MaxCopies to somewhere, most likely CIM_PrintService.

Moving requires that we deprecate the property in the CIM_Printer element, include in the description a pointer to the new location of the property, and define the new property in the correct target element.

- Move DefaultCopies to somewhere, most likely CIM_PrintService.
- Move MaxNumberUp somewhere, most likely CIM_PrintService.
- Move DefaultNumberUp somewhere, most likely CIM_PrintService.
- Move MaxSizeSupported to somewhere, most likely CIM_PrintService.

2.2 *Vague or Meaningless Properties*

Deprecate meaningless properties.

- Fix the properties CIM_Printer.HorizontalResolution and VerticalResolution.

Horizontal and vertical resolution do not make sense for printers that (a) can print page images in portrait and landscape orientations, and (b) can feed paper by the short edge or the long edge. This does not require a complex or exotic printer: any modern home or office inkjet or laser printer can render "vertical" and "horizontal" completely ambiguous. Printer MIB uses prtMarkerAddressabilityFeedDir and ...XFeedDir to specify the resolution(s) of the printer unambiguously. CIM_Printer should use the same model.

Note that CIM models resolution only in pixels per inch. Printer MIB and IPP support both inches and centimeters. That's okay, we don't need no steenkin' meter sticks 'round here.

Suggestion: Correct the descriptions for these two properties to use the semantics of the corresponding Printer MIB properties. Define Vertical to be FeedDir and Horizontal to be XFeedDir. The units can probably remain at pixels per inch.

Deprecate most CurrentXxxx properties. The meaning of "current" is not clear, and, after some discussion, we do not think that it is useful. Example: "CurrentPaperType" might make sense for a continuous form printer loaded with a particular size and color of paper;

but it cannot be measured, even instantaneously, for a complex printer with multiple media sources and a long paper path. "Current" is not interesting; "Default" is.

- Deprecate CurrentPaperType. Consumers should refer to DefaultPaperType instead.
- Deprecate CurrentLanguage.
- Deprecate CurrentMimeType.
- Deprecate CurrentCapabilities.

Add descriptive semantics for some properties whose behavior is not clear.

- CurrentCharSet and CurrentNaturalLanguage need clear descriptions of
 - Their impact on the operation of the printer; and
 - How their values are changed; and
 - When a changed value takes effect.

2.3 *Errors in MappingStrings*

- CIM_Printer.PrinterStatus is mapped to Printer-MIB.hrPrinterStatus, change to HostResources-MIB.
- Same for DetectedErrorState.

2.4 *Improvements to MappingStrings and ModelCorrespondences*

- CIM_Printer.DefaultPaperType should map to Printer-MIB.prtInputDefaultIndex and prtInputMediaName.
- TimeOfLastReset should map to MIB-2.sysUptime. The transformation is not trivial and direct; it does require some arithmetic and access to the current real time.

2.5 *Write Attribute (Modifiability)*

Suggestions for properties that should be modifiable. In the MOF, we would add the "write" attribute to the property definition.

PaperTypesAvailable

DefaultPaperType
DefaultMimeType (if this turns out to be a sensible property)
DefaultCapabilities
MaxCopies (when moved)
DefaultCopies (when moved)
MaxNumberUp (when moved)
DefaultNumberUp (when moved)
MaxSizeSupported (when moved)
CurrentCharSet
CurrentNaturalLanguage
AvailableJobSheets

2.6 Mutability

A property is mutable if its value can change. This might be a useful piece of metadata for developers considering what data to cache. But how fast can a value change?

- The WSDM MUWS Part 1 spec says, "Mutability is defined as an indication of whether the value of a property can change over time." I had hoped for something more enlightening.
- (The C++ use of "mutable" makes a particular data item writable within an otherwise constant object, which is much closer to what we mean by "modifiable." Other programming languages use the term similarly.)

We could usefully invent a more rigorous definition of mutability. (Or adopt one if we can find one, of course.)

Simple example:

- The value of this property may change at any time without notice, e.g., status, page count.
- The value of this property will not change unless the device is reset, but may change across power or reset cycles, e.g., factory specified values such as version numbers in the ROMs.

Trickier example:

- The value of this property may change at any time without notice, e.g., status, page count.
- The value of this property may change by human intervention, e.g., default something.
- The value of this property may change across power or reset cycles, e.g., factory specified values such as version numbers in the ROMs.
- The value of this property is determined by the physical structure of the device and will not change without a screwdriver.

Yes, the advent of hot-plug capabilities will make some of these definitions vary over time.

Suggestions for properties whose mutability should be documented for the benefit of developers.

PrinterStatus
DetectedErrorState
ErrorInformation
PaperTypesAvailable
DefaultPaperType
DefaultMimeType (subject to the same questions as above)
JobCountSinceLastReset
TimeOfLastReset
DefaultCapabilities
MaxCopies (when moved)
DefaultCopies (when moved)
MaxNumberUp (when moved)
DefaultNumberUp (when moved)
MaxSizeSupported (when moved)
CurrentCharSet
CurrentNaturalLanguage
AvailableJobSheets

2.7 Other

- The description clause of PaperTypesAvailable refers to RFC1759; update to 3805.