

# Tour of the Printer Provider Proxy Prototype

PWG WIMS-CIM Working Group

Rick Landau  
Dell, CTO Office  
2009/08/19 v0.1

# Goal of the Proxy Provider Proto

---



- Demonstrate the feasibility of the CIM printer-related classes and their properties
  - The printer classes are new, therefore considered “Experimental”
  - DMTF requires two independent implementations before a class can be promoted to “Final”
  - The Printer Provider Proxy Prototype is one of the two

# Doing a CIM Proxy Provider

---



- What is *this* CIM provider?
  - This provider implements a number of the new printer classes
- What's a proxy provider? An SNMP-to-CIM proxy agent
  - Reads SNMP data from a network printer
  - Republishes that data in CIM format
  - The cheese in the sandwich translates the data

# Corners Cut

---



- Just the cheese, no sandwich
- SNMP input
  - Reads MIB dump file taken from network printer
  - Dramatically easier and faster to develop under simulation
  - Dealing with network printers in real time makes testing much more difficult
- CIM output
  - Formatted text to stdout
  - No direct connection to CIMOM
  - Enumerates all instances of a requested class
  - Implements all plausible properties
- Static geometry of tables
  - Does not re-assess Alert table every time



# Printer Classes Implemented

---



- (All begin with CIM\_Print)
- InputTray
- OutputTray
- MediaPath
- Marker
- Channel
- Interpreter
- Interlock
- AlertRecord
- (Finisher class not implemented)
- (Printer-specific Associations)
- PrinterComponent
- AssociatedPrintSupply
- AssociatedPrintInterpreter
- (Other CIM Associations)
- ConcreteComponent
- Dependency
- UseOfLog
- LogManagesRecord

# Where's the Code?

---



- On the PWG FTP server
  - `ftp://ftp.pwg.org/pub/pwg/wims/cim/DellPrinterProviderProxyPrototype_v10.zip`
  - `ftp://ftp.pwg.org/pub/pwg/wims/cim/DellPrinterProviderProxyPrototype_README.txt`
- Includes all the test files you will see here

# Classes, Parents, Collections

---



- Corresponding directly to SNMP tables
  - Channel, InputTray, Interlock, Interpreter, Marker, MediaPath, OutputTray
- Corresponding to SNMP table or tables, sort of
  - Supply, AlertRecord
- Singletons
  - Printer, RecordLog
- Association classes
- The "El Fako" classes that carry data used by others
  - Console, ConsoleDisplay, Localization, Colorant

# Relationships in Dictionaries

---



- Python dict (= Perl hash) maps key --> value
- SNMP name <--> OID
- CIM class --> CIM properties
- CIM class --> SNMP table source
- Class, table, variable --> subscripts for SNMP access
- CIM property --> list of SNMP vars required to calc
- CIM property --> algorithm to calculate its value
- CIM property --> argument to calculation algorithm
- And seven more dicts and lists just to describe the structure of the Association classes



# Other Info in Dictionaries

---



- CIM property enum value --> text version
- Paper size name --> size tolerance interval

# How to Get CygWin

---



- [www.cygwin.com](http://www.cygwin.com)
- Click "Install CygWin now"
- Download the setup.exe file and run it
- Select "Install from Internet"
- Select "DOS/Text line endings"
- Choose a mirror site near you (bandwidth counts)
- Select Perl and Python from the Languages section of options; the rest should be standard
- Make sure you get awk and less, too

# How to Dump a MIB



- Get a Net-SNMP kit from the web
- Dump a printer

```
snmpwalk -v1 -cpublic -Oen <ip-address> 1.1
```

  - OIDs and enums must be in numeric form, **not** interpreted!
  - Need to see the data exactly as it would come back from an SNMP operation, not nicely interpreted for humans
- Process some irrelevant text out of the lines in the dump
  - Dot at beginning of line
  - Equal sign and datatype
  - Double quote for empty string values
  - Etc.
- Result should be

```
oid "\t" value newline
```

# Look at the Flow of Operations

---



- Where does the test information come from, where does it come out?
- How does control flow through all the internal object classes?
- General idea, then look at code

# Information Flow

---



- Ini files read into tables to be used at runtime
- CRequest.Enumerate creates all internal structures for a class
- Request populates all needed SNMP values, calculates all CIM property values
- Request collects output for all instances, all properties of each instance

# Control Flow of a Run

---



- User invokes shell script `./prov-test-<something>.sh`
  - Script supplies many (fixed) filenames
  - Optional args: SNMP dump filename, maybe classname
- Script invokes `./prov-test-<something>.py`
  - Puts positional filenames into a dictionary,
  - Creates a CProvider instance, calls `InitAll()` to absorb ini files
  - For each classname, enumerates instances
  - For each instance, collects output and prints

# Request Flow

---



- For each class, creates all instances (usually corresponding to SNMP table)
- For each instance, create all properties, populate values
  - First get SNMP vars needed, then calculate property value
- For each instance, collect printable form of property
  - Name = value
- Concatenate output for all instances of class

# Output Analyzed

---



- First get output for all dump files, takes a few minutes  
`./prov-test-allfiles.sh | tee prov-test-allfiles-<date>.log`
- Then, "analyze"
  - greps the file hundreds of times to isolate particular property values  
`./prov-test-analyze-log.sh prov-test-allfiles-<date>.log | tee prov-test-allfiles-analyzed-<date>.log`
- Examine output for differences in responses among printers



# Look at Some Files

---



- How to get CygWin
- All ini files
- Input for one printer
- Output for one printer
- Output for all printers
- “Analyzed” output all printers
- The code: scripts, tests, classes, calculations
- Trace mechanism: vars, code points, output

# Feedback!!

---



- PLEASE send feedback of any sort:
  - It works
  - It doesn't work
  - Suggested improvement
  - Here's another printer to try it on
  - Etc.
- Send to WIMS email reflector, [wims@pwg.org](mailto:wims@pwg.org)
- And/or to me, [Richard\\_Landau@dell.com](mailto:Richard_Landau@dell.com)

# Questions?

---

