

## Contents

[1.Using Raspberry Pi For ESXi Server Monitoring](#)[2.About Me](#)[3.Caveats](#)[4.Outline](#)[5.Objectives For This Monitoring Systems](#)[6.Our Plan](#)[7.Hardware Option - Example 1](#)[8.Hardware Option - Example 2](#)[9.Hardware Option - Example 3](#)[10.Hardware Option - Example 4](#)[11.We Chose Raspberry Pi](#)[12.Talking To VMWare](#)[13.SNMP](#)[14.CIM - Looks Like Hard Work](#)[15.WS API](#)[16.Perl](#)[17.PHP](#)[18.RAW Output](#)[19.Few Problems With This](#)[20.My Chosen Solutions](#)[21.My Chosen Solutions](#)[22.Sanitized Output](#)[23.Still Not Quite Right](#)[24.Still Not Quite Right Again!](#)[25.Finding Your Dongle](#)[26.Talking To Your Dongle](#)[27.Getting To Know Your Dongle](#)[28.Getting To Know Your Network](#)[29.Sending An SMS](#)[30.AT Command Help](#)[31.Disabling The Annoying Fake Cd Drive](#)[32.SMS Server Tools](#)[33.Setting Up The Pi](#)[34.Installing SMS Server Tools 3](#)[35.Checking SMS Server Tools](#)[36.Sending A Test SMS](#)[37.Installing PHP](#)[38.Bits And Peices](#)[39.Configuration](#)[40.Code](#)[41.Future](#)[42.Questions?](#)

# 1. Using Raspberry Pi For ESXi Server Monitoring

Ben Jefferson <ben.jefferson@brighter-connections.com>



[www.brighter-connections.com](http://www.brighter-connections.com)

## 2. About Me

- At heart I'm web developer
- But I am also...

```
<plug>
    Tech. Director at brighter-connections.com
</plug>

<plug>
    Author of WhatToGive.com
</plug>
```

### 3. Caveats

- I'm a bugger for reinventing wheels
- Noone should use what I am presenting!
- Use openNMS, icinga2, mauve or possibly Nagios
  
- BUT... you might use some of the ingredients
- You might...
  - Use the SMS infrastructure outlined with OpenNMS / Mauve
  - Plug the ESXi monitoring stuff into your existing monitoring system
  - Mess about with a Raspberry Pi for your latest hacking project
  - Have a tiny bit more understanding of how you might monitor ESXi

## 4. Outline

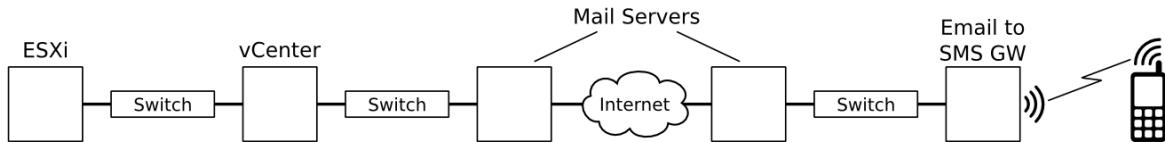
- Key objectives for this monitoring system
- Hardware options
- Talking to VMware
- Talking to 3G dongles
- Putting it all together

## 5. Objectives For This Monitoring Systems

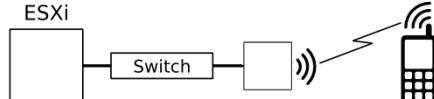
- Reliability
  - ...through built in redundancy
    - Redundant alerting channels
    - Redundant monitoring hardware
  - ...through simplicity
    - As few hardware elements as possible
    - As few software elements as possible
    - As few external systems as possible
  - All solid state components
  - Not so chatty as to be ignored...but with some kind of heartbeat
- SMS based
  - Reliable
  - Uses very different infrastructure to the systems being monitored
  - Intrusive enough to be noticed
  - Ubiquitous
- Self-monitoring
- Pluggable
- Low cost

## 6. Our Plan

- If you want SMS alerts you currently have to do something like this...



- We wanted something more like this...



## 7. Hardware Option - Example 1

Mele A1000

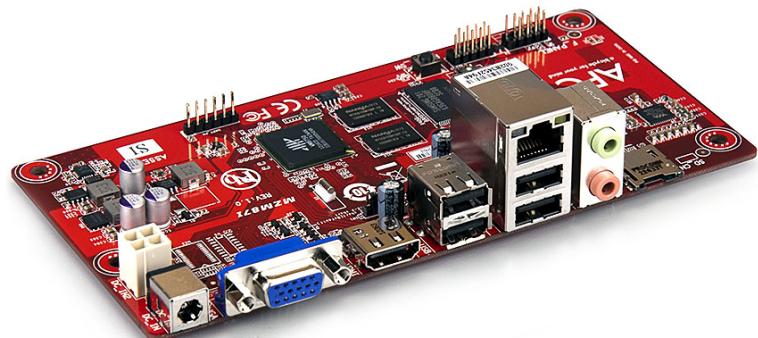
- Designed as a set top box
- Price: £60
- Supplier: Tricky to get in the UK
- 1.2GHz AllWinner Cortex A8 ARM processor
- HDMI
- VGA
- Ethernet



## 8. Hardware Option - Example 2

APC 8750

- General purpose minimal computer
- Home: <http://apc.io/>
- Purchase: <http://uk.rs-online.com/web/p/single-board-computers/7689782/>
- Price: £50
- VIA ARM 800MHz Processor
- 2GB onboard flash
- 512MB of RAM
- HDMI & VGA
- Ethernet



## 9. Hardware Option - Example 3

MK802 micro-PC

- Home:



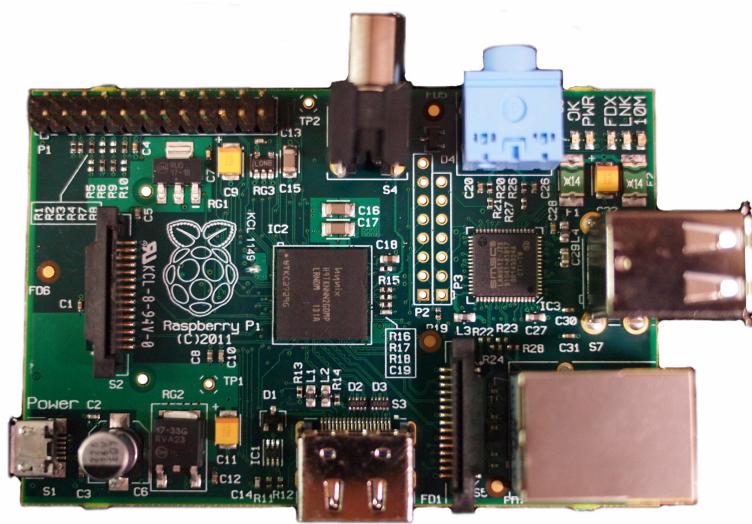
<https://www.miniand.com/products/MK802%20Android%20Mini%20PC>

- Purchase: <http://www.cloudsto.com/component/virtuemart/rikomagic/rikomagic-mk802-ii-detail.html?Itemid=0>
- Price: £50
- 1.5GHz AllWinner Cortex A10 ARM processor
- 512MB of RAM
- WiFi
- NO Ethernet

## 10. Hardware Option - Example 4

Raspberry Pi

- Home:  
<http://www.raspberrypi.org/>
- Purchase:



<http://raspberrypi.rsdelivers.com/default.aspx?cl=1>

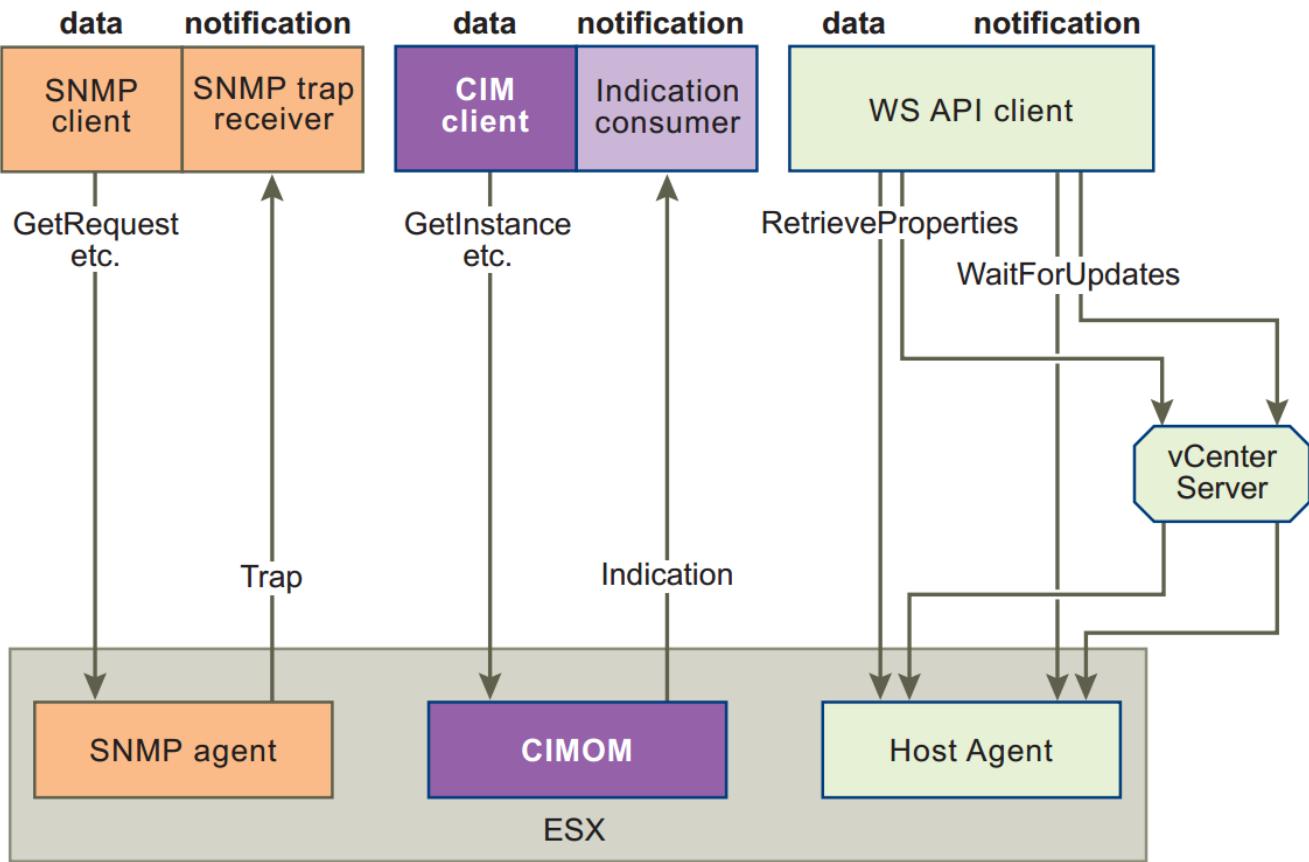
- Price: £30
- BCM2835 700MHz ARM1176JZFS processor
- 512MB of RAM
- Ethernet
- Composite video

## 11. We Chose Raspberry Pi

- Large community of developers
- One of the cheapest options
- Wide range of accessories
  - Cases
  - Power supplies
  - Battery power
- Looked like fun

## 12. Talking To VMWare

- [http://www.vmware.com/files/pdf/techpaper/hwhm41\\_technote.pdf](http://www.vmware.com/files/pdf/techpaper/hwhm41_technote.pdf)



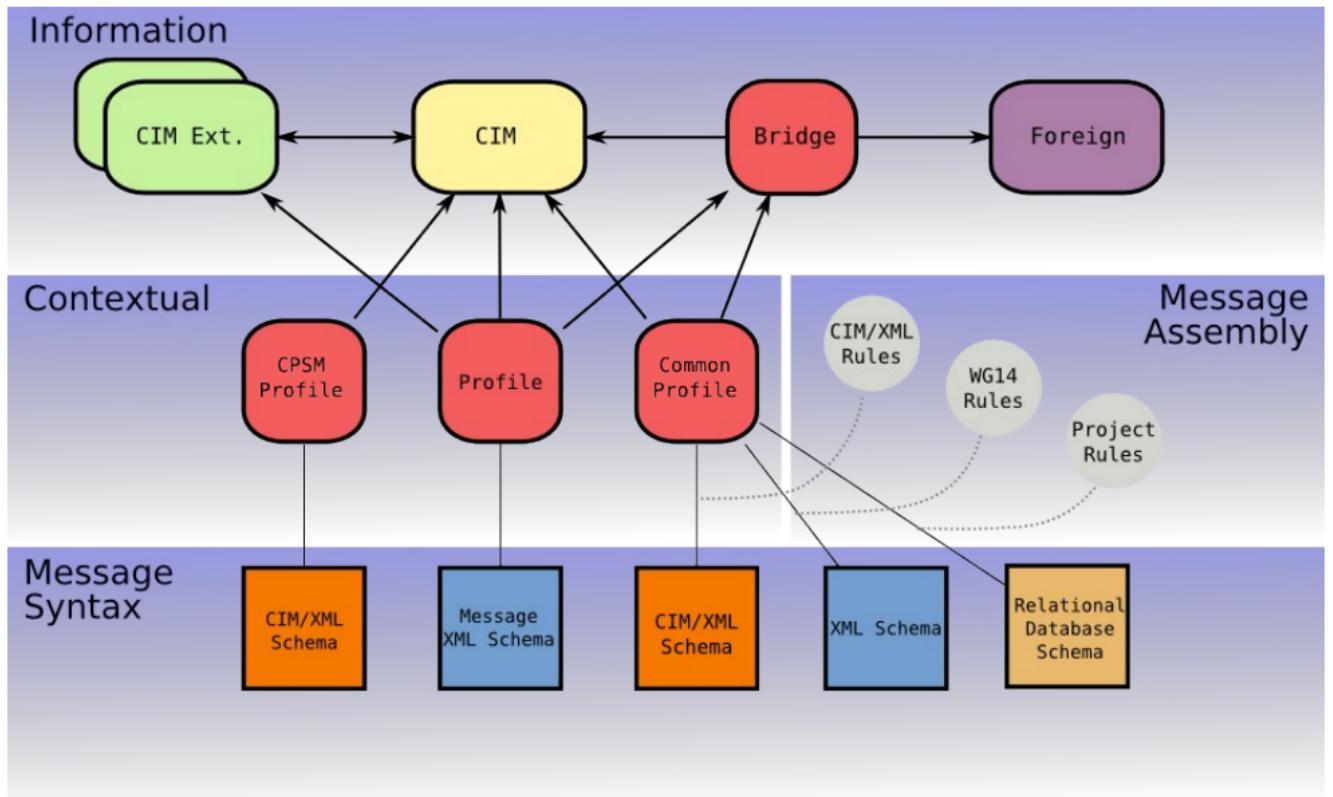
- 3 Interface options
  - SNMP
  - CIM
  - WS API (aka SOAP)

## 13. SNMP

- NOT an exhasutive search but...
- Googling through up stuff like...
  - Use Vmware VCenter server to monitor ESXi hosts and send out SNMP traps to intermapper. This will only work if you are using VCenter server though. Additionally, **the traps sent out are not well documented so a complex trap handler will likely have to be written.**
  - (<http://forums.intermapper.com/viewtopic.php?p=2981&sid=9fa8f74356fd05d6f720eec843076dd1>)
  - SNMP is not enabled in ESXi for the reason that there is not much there to query and you can use the CIM queries that I mentioned in the previous post to look at this instead.
  - (<http://www.matthewjwhite.co.uk/blog/2010/02/10/esxi-enabling-snmp/>)
- Several people seemed to have some success with 3rd party drivers (HP/Dell) but this looked very proprietary

# 14. CIM - Looks Like Hard Work

## Models and Dependencies



- [cimtool.org](http://cimtool.org)
  - read and merge CIM and local UML models in XMI form
  - generate equivalent OWL ontologies
  - generate XML schemas, OWL and RDFS ontologies for profiles
  - validate instances against profiles (including very large CIM/XML instances)
- <http://files.cimtool.org/CIMUGPresentation.pdf>
- Perl - CPAN
  - The DMTF::CIM class provides object-oriented access to a CIM schema and, when created using a protocol module such as DMTF::CIM::WSMan, permits the use of DSP0223 generic operations against a target.
- Python
  - pywbem

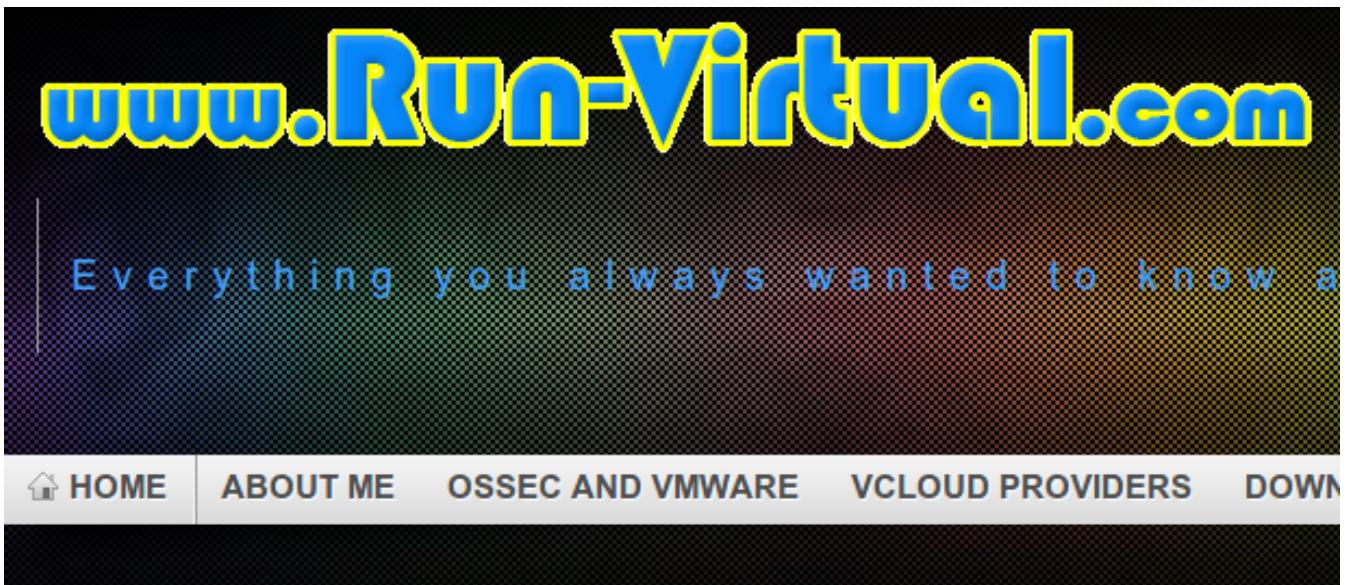
## 15. WS API

- SOAP
  - Not my favourite but manageable and reasonably familiar
  - Based on HTTP
  - Allows access to VMWares "Managed Object Browser"
- Looks like the best option for me so... what language to use?

## 16. Perl

- This example script looked promising
  - This script allows you to monitor the host hardware Health Status located on the configuration tab under Hardware section of an ESX(i) host. The script will look at all available components health such as CPU, MEMORY, STORAGE, CHASSIS, etc. to ensure everything is operating at normal status.
  - (<http://communities.vmware.com/docs/DOC-11641>)
- But...
  - use VMware::VIRuntime;
  - use VMware::VILib;
  - use Net::SMTP;
- Couldn't find anyone using the VMware Perl modules on Raspberry Pi
- CPAN is convoluted at the best of times
- Running it on the Pi took ages and then crashed the Pi!

## 17. PHP



### PHP vSphere (5.x) programming

It has been a while I updated my PHP examples, the previous code was designed for VI3 (a long time back).

Not too much has changed, ~~but~~ somethings you should be aware of. PHP5 has a build in soap library, which I do not use, I like to use nusoap, so make sure you use the nusoap\_client, instead of soap\_client. I also have a small library that contains 2 functions, that I found where easier done by creating raw soap messages, instead of creating them with variables. One is for the most imports Retrieve Properties. I build it with a very extensive traversal spec, so it should be easily be able to find anything.

- This example looked interesting
  - [http://www.run-virtual.com/?page\\_id=893](http://www.run-virtual.com/?page_id=893)
- PHP is a language I am familiar with
- nuSoap is pure PHP so no dependencies and nothing to compile
- Very quickly managed to get this working
- Couple of tweaks
  - Change \$objecttype to "HostSystem";
  - Add print\_r's to dump output
- This gave useful results...

## 18. RAW Output

---

```
[1] => Array
(
    [obj] => host-1856
    [propSet] => Array
        (
            [0] => Array
                (
                    [name] => name
                    [val] => 192.168.1.113
                )

            [1] => Array
                (
                    [name] => runtime
                    [val] => Array
                        (
                            [connectionState] => connected
                            [powerState] => poweredOn
                            [standbyMode] => none
                            [inMaintenanceMode] => false
                            [bootTime] => 2012-11-28T15:28:14.375663Z
                            [healthSystemRuntime] => Array
                                (
                                    [systemHealthInfo] => Array
                                        (
                                            [numericSensorInfo] => Array
                                                (
                                                    [0] => Array
                                                        (
                                                            [name] => System Board 9 P
                                                            [healthState] => Array
                                                                (
                                                                    [label] => Green
                                                                    [summary] => Senso
                                                                    [key] => green
                                                                )
                                                                [currentReading] => 21400
                                                                [unitModifier] => -2
                                                                [baseUnits] => Watts
                                                                [sensorType] => power
                                                        )
                                                        )
                                                    )
                                                )
                                            )
                                        )
                                    )
                                )
                            )
                        )
                    )
                )
            )
        )
    )
)
[1] => Array
```

```
(  
    [name] => Other 12 Temp 30  
    [healthState] => Array  
        (  
            [label] => Green  
            [summary] => Senso  
            [key] => green  
        )  
  
    [currentReading] => 6800  
    [unitModifier] => -2  
    [baseUnits] => Degrees C  
    [sensorType] => temperatur  
)  
  
[2] => Array  
(  
    [name] => Drive Backplane  
    [healthState] => Array  
        (  
            [label] => Green  
            [summary] => Senso  
            [key] => green  
        )  
)
```

## 19. Few Problems With This

- The structure is a bit overcomplicated
- I want to apply tests to specific parameters but...
  - Many of the parameter names contain version numbers like this...
    - VMware scsi-mptsas 4.23.01.00-5vmw.500.0.0.469512 2011-08-19 01:59:20.000
  - Some of the parameters seem to contain state data as well
    - Power Domain 1 Temp 8 --- Normal

## 20. My Chosen Solutions

- Represent the data as path => value pairs
- Run regular expressions on the paths to remove extraneous bits

```
/propSet/runtime/val/healthSystemRuntime/systemHealthInfo/numericSensorInfo/Power [
```

- becomes

```
/Power Domain 1 Temp 8 --- Normal/healthState/label
```

- Split off any status data from the parameter name e.g.

```
/Power Domain 1 Temp 8 --- Normal/healthState/label
```

- becomes

```
/Power Domain 1 Temp 8
```

- Store the status in a new "status" sub-entry e.g.

```
/Power Domain 1 Temp 8/status = Normal
```

## 21. My Chosen Solutions

- Apply regular expressions to remove things that look like version numbers e.g.

```
/VMware ehci-ehci-hcd 1.0-3vmw.500.0.0.469512 2011-08-19 01:58:52.000/
```

- becomes

```
/VMware ehci-ehci-hcd
```

- Store the status in a new "versionData" sub-entry e.g.

```
/VMware ehci-ehci-hcd/VersionData = 1.0-3vmw.500.0.0.469512 2011-08-19 01:58:52
```

- Output now looks like this...

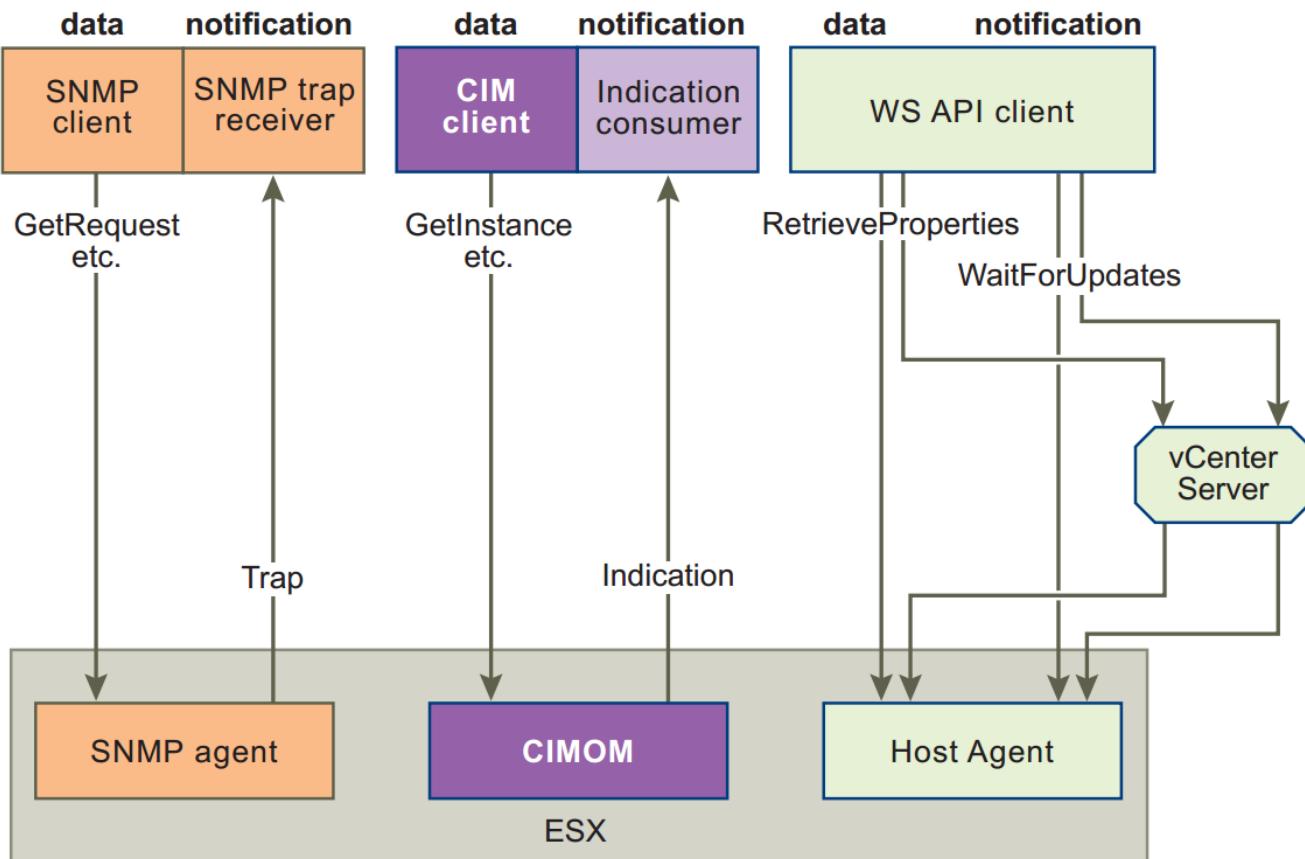
## 22. Sanitized Output

```
/name => 192.168.1.112
/obj => host-1846
/connectionState => connected
/powerState => poweredOn
/standbyMode => none
/inMaintenanceMode => false
/bootTime => 2012-11-28T14:09:37.377Z
/System Board 9 Power Meter/name => System Board 9 Power Meter --- Device enabled
/System Board 9 Power Meter/healthState/label => Green
/System Board 9 Power Meter/healthState/summary => Sensor is operating under normal co
/System Board 9 Power Meter/healthState/key => green
/System Board 9 Power Meter/currentReading => 20200
/System Board 9 Power Meter/unitModifier => -2
/System Board 9 Power Meter/baseUnits => Watts
/System Board 9 Power Meter/sensorType => power
/System Board 9 Power Meter/status => Device enabled
/Other 12 Temp 30/name => Other 12 Temp 30 --- Normal
/Other 12 Temp 30/healthState/label => Green
/Other 12 Temp 30/healthState/summary => Sensor is operating under normal conditions
/Other 12 Temp 30/healthState/key => green
/Other 12 Temp 30/currentReading => 6600
/Other 12 Temp 30/unitModifier => -2
/Other 12 Temp 30/baseUnits => Degrees C
/Other 12 Temp 30/sensorType => temperature
/Other 12 Temp 30/status => Normal
/Drive Backplane 1 Temp 29/name => Drive Backplane 1 Temp 29 --- Normal
/Drive Backplane 1 Temp 29/healthState/label => Green
/Drive Backplane 1 Temp 29/healthState/summary => Sensor is operating under normal con
/Drive Backplane 1 Temp 29/healthState/key => green
/Drive Backplane 1 Temp 29/currentReading => 3500
/Drive Backplane 1 Temp 29/unitModifier => -2
/Drive Backplane 1 Temp 29/baseUnits => Degrees C
/Drive Backplane 1 Temp 29/sensorType => temperature
/Drive Backplane 1 Temp 29/status => Normal
/Power Supply 1: Running/Full Power-Enabled/name => Power Supply 1: Running/Full Power
/Power Supply 1: Running/Full Power-Enabled/healthState/label => Green
/Power Supply 1: Running/Full Power-Enabled/healthState/summary => Sensor is operating
/Power Supply 1: Running/Full Power-Enabled/healthState/key => green
/Power Supply 1: Running/Full Power-Enabled/currentReading => 0
/Power Supply 1: Running/Full Power-Enabled/unitModifier => 0
/Power Supply 1: Running/Full Power-Enabled/baseUnits =>
/Power Supply 1: Running/Full Power-Enabled/sensorType => power
/Power Supply 2: Running/Full Power-Enabled/name => Power Supply 2: Running/Full Power
/Power Supply 2: Running/Full Power-Enabled/healthState/label => Green
```

```
/Power Supply 2: Running/Full Power-Enabled/healthState/summary => Sensor is operating
/Power Supply 2: Running/Full Power-Enabled/healthState/key => green
/Power Supply 2: Running/Full Power-Enabled/currentReading => 0
/Power Supply 2: Running/Full Power-Enabled/unitModifier => 0
/Power Supply 2: Running/Full Power-Enabled/baseUnits =>
/Power Supply 2: Running/Full Power-Enabled/sensorType => power
```

---

## 23. Still Not Quite Right



- The data looks good ... BUT IT NEVER UPDATES!
- Turns out this is a VMWare bug
  - The ESX hosts are not configured by default to update the health data in the VCenter server
  - This is fixed by setting <throttleInterval>120</throttleInterval> in the vpxa config
  - <http://communities.vmware.com/thread/402899?tstart=0>
  - [http://kb.vmware.com/selfservice/microsites/search.do?language=en\\_US&cmd=displayKC&externalId=2017770](http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2017770)
  - N.B. in ESXi 5 the config file has moved to /etc/vmware/vpxa/vpxa.cfg

## 24. Still Not Quite Right Again!

- Now the data changes ... a bit ... BUT NOT WHEN I PULL A POWER SUPPLY!
- The health state on the power supplies doesn't change!
- Couldn't figure out what was going on so had to resort to querying ESX servers directly
- This means config will have to enumerate ESX servers at present
- Or connect to VCenter first to get list of servers and then iterate across servers

## 25. Finding Your Dongle

- 3G dongles are easy to get hold of
- Reasonable Linux support
- Run dmesg to see if your dongle is detected

---

```
[307337.252224] usb 2-1.1: new high speed USB device using ehci_hcd and address 4
[307337.368492] scsi7 : usb-storage 2-1.1:1.2
[307337.507254] usbcore: registered new interface driver usbserial
[307337.507283] USB Serial support registered for generic
[307337.507366] usbcore: registered new interface driver usbserial_generic
[307337.507368] usbserial: USB Serial Driver core
[307337.547922] USB Serial support registered for GSM modem (1-port)
[307337.548022] option 2-1.1:1.0: GSM modem (1-port) converter detected
[307337.548248] usb 2-1.1: GSM modem (1-port) converter now attached to ttyUSB0
[307337.548273] option 2-1.1:1.1: GSM modem (1-port) converter detected
[307337.548447] usb 2-1.1: GSM modem (1-port) converter now attached to ttyUSB1
[307337.548470] option 2-1.1:1.3: GSM modem (1-port) converter detected
[307337.548590] usb 2-1.1: GSM modem (1-port) converter now attached to ttyUSB2
[307337.548607] usbcore: registered new interface driver option
[307337.548609] option: v0.7.2:USB Driver for GSM modems
[307338.360945] scsi 7:0:0:0: CD-ROM           HSPA      USB SCSI CD-ROM 2.31 PQ:
[307338.361540] scsi 7:0:0:1: Direct-Access     ZTE       MMC Storage   2.31 PQ:
[307338.365140] sr1: scsi-1 drive
[307338.365367] sr 7:0:0:0: Attached scsi CD-ROM sr1
[307338.365620] sr 7:0:0:0: Attached scsi generic sg2 type 5
[307338.366075] sd 7:0:0:1: Attached scsi generic sg3 type 0
[307338.371099] sd 7:0:0:1: [sdb] Attached SCSI removable disk
```

---

## 26. Talking To Your Dongle

- Test with minicom first by running...

```
> dmesg | grep ttyUSB

[ 6.267902] usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB0
[ 6.506995] usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB1
[ 6.655585] usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB2

> sudo apt-get install minicom

> minicom -D /dev/ttyUSB#
```

---

```
Welcome to minicom 2.6.1

OPTIONS: I18n
Compiled on Apr 28 2012, 19:24:31.
Port /dev/ttyUSB0

Press CTRL-A Z for help on special keys

AT
OK
```

## 27. Getting To Know Your Dongle

- Get modem make and model

```
> AT+CGMI  
< ZTE INCORPORATED  
> AT+CGMM  
< K3570-Z
```

- Get sim phone number

```
> AT+CNUM  
< +CNUM: "My Number", "+447426186634", 145
```

- N.B. This is just stored in the SIM phonebook and need not be correct!!!

- Get IMEI number

```
> AT+CGSN  
< 862222001561068
```

- Checking if the dongle is locked

```
> AT+ZSEC?  
< +ZSEC: 3,0
```

- 0 = Initializing the encryption (Insignificant SEC\_ITEMS)
  - 1 = Network Lock error. (Insignificant SEC\_ITEMS)
  - 2 = Network Locked
  - 3 = Unlocked or correct MCC/MNC

## 28. Getting To Know Your Network

- Are we connected to the network?

```
> AT+CREG?  
< +CREG: 0,1
```

- 0 = not registered new operator to registered and not searching
- 1 = registered, home network
- 2 = not registered, currently searching a new operator to register with

- What networks can we see?

```
> AT+COPS=?  
< +COPS: (1,"3","3","23433",0),(3,"vodafone UK","voda UK","23415",0),(3,"T-Mobi
```

- This can take a while to respond

- Which network are we connected too

```
> AT+COPS?  
< +COPS: 1,0,"3 UK",2
```

- Check what SMS service the we are currently using

```
> AT+CSCA?  
< +CSCA: "+447782000800",145
```

- You can Google this to check if it is right

## 29. Sending An SMS

- Set text mode...

```
> AT+CMGF=1  
< OK
```

- Send a message...

```
> AT+CMGS="+447985034436"  
> Test Message
```

- Use Ctrl-Z to end message

```
< +CMGS: 14
```

## 30. AT Command Help

- Varies from modem to modem
- Some fairly comprehensive lists
  - [http://www.3gpp.org/ftp/Specs/archive/27\\_series/27.007/27007-3d0.zip](http://www.3gpp.org/ftp/Specs/archive/27_series/27.007/27007-3d0.zip)
  - <http://www.zeeman.de/wp-content/uploads/2007/09/ubinetics-at-command-set.pdf>
  - <http://www.activexperts.com/mobile-messaging-component/at/>
  - <http://m2msupport.net/m2msupport/software-and-at-commands-for-m2m-modules/>
  - <http://www.shapeshifter.se/2008/04/30/list-of-at-commands/>
- Getting more error information
  - +CEER
- Checking which commands are supported
  - +CLAC

## 31. Disabling The Annoying Fake Cd Drive

- For Huawei...
  - This can also be done with the usb-modeswitch package
  - Or use sdparm to issue a scsi eject
    - > AT^U2DIAG=0
  - Display mode modem that is currently active.
    - > AT ^ GETPORTMODE
  - Sets all settings to default.
    - > AT ^ SETPORT = "A1, A2; 1,2,3,7, A1, A2"
  - Turn off CD + SD
    - > AT ^ SETPORT = "A1, A2; 1,2,3,7"
  - <http://wiki.openwrt.org/doc/recipes/3gdongle>
  - <http://3g-modem.wetpaint.com/page/Huawei+AT-commands>
  - <http://www.imei-server.com/415-huawei-at-comands-for-3g-modems.html>
- For ZTE modems
  - > AT+ZCDRUN=8
    - < Close autorun state result(0:FAIL 1:SUCCESS):1
  - <http://firmware4u.blogspot.co.uk/2013/01/zte-modem-at-commands-to-enable-disable.html>

## 32. SMS Server Tools

"The SMS Server Tools 3 is a SMS Gateway software which can send and receive short messages through GSM modems and mobile phones."

# SMS Server Tools 3

<http://smstools3.kekekasvi.com/index.php>

- Just drop message files in a directory to send SMS's
  - /var/spool/sms/outgoing
- Also provides command line "sendsms <num> <message>" utility
- Written in C
  - no external libraries required
- This also handles receiving SMS's too
  - SMS appear as files in /var/spool/sms/incoming

## 33. Setting Up The Pi

- I tried the ArchLinux distro first but had various minor problems with it
- Raspbian proved much easier to setup
- Follow the steps detailed by my colleague Tim Fletcher
  - <http://blog.night-shade.org.uk/2013/03/setting-up-a-headless-networked-raspberry-pi/>
- Quick summary
  - Boot with wired connection to network
  - Automatically picks up DHCP
  - Use nmap to scan network for devices listening on port 22 with Mac addresses from Raspberry Pi Foundation

## 34. Installing SMS Server Tools 3

- There is already a package for this

```
> sudo apt-get install smstools
```

- Then we need to configure SMS Server Tools

```
> sudo vi /etc/smsd.conf
```

- replace this...

```
# loglevel = 7  
...  
[GSM1]  
#init =  
device = /dev/ttyS0  
incoming = yes  
#pin =  
baudrate = 19200
```

- with this....

```
loglevel = 7  
...  
[GSM1]  
#init =  
device = /dev/ttyUSB0  
incoming = no  
#pin =  
baudrate = 19200
```

## 35. Checking SMS Server Tools

- In separate window run...

```
> sudo tail -f /var/log/smstools/smsd.log
[...]
2013-03-06 22:03:01,7, GSM1: <- OK
2013-03-06 22:03:02,7, GSM1: -> AT+CIMI
2013-03-06 22:03:02,7, GSM1: Command is sent, waiting for the answer
2013-03-06 22:03:02,7, GSM1: <- 234200901833767 OK
2013-03-06 22:03:02,7, GSM1: -> AT+CGSN
2013-03-06 22:03:02,7, GSM1: Command is sent, waiting for the answer
2013-03-06 22:03:02,7, GSM1: <- 864592001561068 OK
2013-03-06 22:03:02,5, GSM1: CGSN: 864592001561068
2013-03-06 22:03:02,5, GSM1: IMSI: 234200901833767
2013-03-06 22:03:02,5, GSM1: Waiting for messages to send...
```

## 36. Sending A Test SMS

- Once smsd is up and running you can send a test SMS
  - Remember to put number in international format
  - You can avoid this with appropriate configuration of smsd
- The "sendsms" utility can be used from the command line

```
> sendsms 447985012345 'Hello World'  
--  
Text: Hello World  
To: 447985012345
```

## 37. Installing PHP

```
> sudo apt-get install php5-cli  
> sudo apt-get install php5-curl
```

- Configure PHP
  - Edit /etc/php/php.ini
  - Hash out open\_basedir
  - Configure timezone

```
[Date]  
; Defines the default timezone used by the date functions  
; http://php.net/date.timezone  
date.timezone = Europe/London
```

## 38. Bits And Peices

- USB disconnect issues
  - probably due to power draw and the polyfuse on the Pi

```
[ 4208.096841] option1 ttyUSB2: option_instat_callback: error -71
[ 4208.119099] option1 ttyUSB2: option_instat_callback: error -71
[ 4208.141476] option1 ttyUSB2: option_instat_callback: error -71
[ 4208.162317] usb 1-1.3: USB disconnect, device number 72
```

- Solution 1 (not ideal)

```
> sudo vi /etc/udev/rules.d/10-3g-dongle.rules
SUBSYSTEMS=="usb", KERNEL=="ttyUSB[0-15]", RUN+="/usr/sbin/service smstools res
```

- Solution 2 - buy a newer Raspberry Pi!

- BCM Watchdog

- This provies a hardware watchdog that can trigger a reboot if the software fails
- <http://pi.gadgetoid.com/post/001-who-watches-the-watcher>

## 39. Configuration

```

<monitorConfig>
<!--
ALL times/intervals/delays etc are specified in seconds

logTimestampInterval controls how often a timestamp is included in the log file output
    set this to zero if you want a timestamp on every log line
logFile can be empty or "-" if you want logging to STDOUT
logLevel 0 => only errors and major events (e.g. startup)
logLevel 10 => debug
defaultInterval is the monitoring interval - this can be overridden on a per-monitor
attribute
-->
<global
    defaultInterval="600"
    logLevel="10"
    logTimestampInterval="0"
>

</global>

<alertMediums>
    <!-- The order of the entries implies which will be tried first -->
    <!-- When an alert needs to be sent each method will be tried until one non-i

    <!-- The email alert engine is very simple at present - there is nothing that
<email name="email" invisible="yes" />

    <!-- Not implemented yet but might be kind of cool -->
    <!-- At least one of the alert recipients would need to have twitter credentials
<twitter name="twitter" invisible="yes" />

    <!-- This will try and use BulkSMS first to send the message - this is a HTTP
    <!-- If you have internet connectivity this will probably be cheaper and more reliable
    <!-- Also it doesn't require a 3G dongle of course! -->
    <bulksms name="bulkSms1" invisible="no" username="" password="" />
    <!-- You might want to try more than one bulk SMS account in case one runs out
    <bulksms name="bulkSms2" invisible="no" username="" password="" />

    <!-- The modem entries define 3G dongles -->
    <smsServer name="3GSMS"
        outgoingDir="/var/spool/sms/outgoing"
        sentDir="/var/spool/sms/sent"
        failedDir="/var/spool/sms/failed"
        tmpDir="/tmp"
        flash="yes"
        timeout="90"
    >
        <!-- setupCommands contains any AT commands that should be fed to the
        <setupCommands>
        </setupCommands>
    </smsServer>

    <!-- Default logLevel is 0 -->
    <!-- You can also pass a file="" attribute to specify a file to write alerts
default is to write to main monitor log -->
    <log name="log" invisible="yes" logLevel="0"/>

```

```

</alertMediums>

<alertRecipients>
    <alertRecipient name="Ben">
        <details medium="3GSMS" number="YOUR PHONE NUMBER HERE" />
        <details medium="email" address="YOUR EMAIL HERE" />
    </alertRecipient>
</alertRecipients>

<monitors>
    <!-- default interval for monitors is 10mins -->

    <!-- The monitor block here defines what should happen if one of the alerting
    <!-- This is a special type of monitor and should only be included once -->

    <medium name="medium1" interval="">
        <recipient name="Ben" />
        <recipient name="Ed" />
    </medium>

    <random name="Daily health check" message="Daily monitoring system health che
interval="86400">
        <recipient name="Ben" mediums="3GSMS,email"/>
    </random>

    <!-- The tolerance states how many seconds failure can be tolerated -->
    <!-- bear in mind that the checks are still governed by the interval - thus
    <!-- means you still have to wait for the interval to expire and a second fai
alert is triggered. -->
    <vsphere interval="300" name="vSphere1" username="" password="" repeatLimit="
        <!-- Who to alert... -->
        <recipient name="Ben" />
        <recipient name="Ed" />

        <!-- Which ESX hosts to monitor... -->
        <!-- is username and/or password is added here these will override th
vsphere as a whole -->
        <includeHost name="IP OR NAME OF HOST 1"/>
        <includeHost name="IP OR NAME OF HOST 2"/>

        <!-- What to monitor... -->
        <!-- test attribute can be one of ne|eq|gt|lt|ge|le|re -->
        <!-- These are tests to check that everything is OK i.e. test failure

        <test name="System power draw too high" parameter="/System Board 9 Po
value="22900" type="lt" />
            <test name="Temperature sensor 12 too hot" parameter="/Other 12 Temp
value="6900" type="lt" />
                <test name="Temperature sensor 11 too hot" parameter="/Other 11 Temp
value="3600" type="lt" />
                    <test name="Temperature sensor 10 too hot" parameter="/Other 10 Temp
value="3600" type="lt" />
                        <test name="Temperature sensor 9 too hot" parameter="/Other 9 Temp 24
value="3800" type="lt" />
                            <test name="Temperature sensor 8 too hot" parameter="/Other 8 Temp 23
value="4200" type="lt" />
                                <test name="Temperature sensor 7 too hot" parameter="/Other 7 Temp 22
value="3500" type="lt" />
                                    <test name="Temperature sensor 6 too hot" parameter="/Other 6 Temp 21
value="3600" type="lt" />
                                        <test name="Temperature sensor 5 too hot" parameter="/Other 5 Temp 20
value="3800" type="lt" />

```

```
value="3400" type="lt" />
    <test name="Temperature sensor 4 too hot" parameter="/Other 4 Temp 19
value="3000" type="lt" />
    <test name="Temperature sensor 3 too hot" parameter="/Other 3 Temp 12
value="4600" type="lt" />
    <test name="Temperature sensor 2 too hot" parameter="/Other 2 Temp 11
value="3800" type="lt" />
    <test name="Temperature sensor 1b too hot" parameter="/Other 1 Temp 1
value="4800" type="lt" />
    <test name="Temperature sensor 1a too hot" parameter="/Other 1 Temp 1
value="2400" type="lt" />
    <test name="Temperature sensor 3 on expansion board too hot" parameter=
Expansion Board 3 Temp 15/currentReading" value="3700" type="lt" />
    <test name="Temperature sensor 2 on expansion board too hot" parameter=
Expansion Board 2 Temp 14/currentReading" value="3700" type="lt" />
    <test name="Temperature sensor 1 on expansion board too hot" parameter=
Expansion Board 1 Temp 13/currentReading" value="3400" type="lt" />
    <test name="Power supply 2 too hot" parameter="/Power Domain 2 Temp 9
value="3900" type="lt" />
    <test name="Power supply 1 too hot" parameter="/Power Domain 1 Temp 8
value="4300" type="lt" />
    <test name="Memory module 4 too hot" parameter="/Memory Module 4 Temp
value="3900" type="lt" />
```

## 40. Code

- Very much a work in progress
- Available at
  - <http://www.github.com/skwirrel/phpMonitor>

```
> git clone http://github.com/skwirrel/phpMonitor  
> ./install.sh  
> vi /etc/phpMonitor.conf.xml  
> update-rc.d phpMonitor defaults
```

## 41. Future

- Finish of the base alerting mechanisms (e.g. email, sms over http)
- Threading - or maybe distributed configuration
- SMS'ing each other to test SMS delivery
- Adding more monitors
- Assessing and improving scalability
- Battery backup

## 42. Questions?

Ben Jefferson <ben.jefferson@brighter-connections.com>

