

Monitoring 6000+ hosts in Zabbix

A Pseudo-DevOps journey

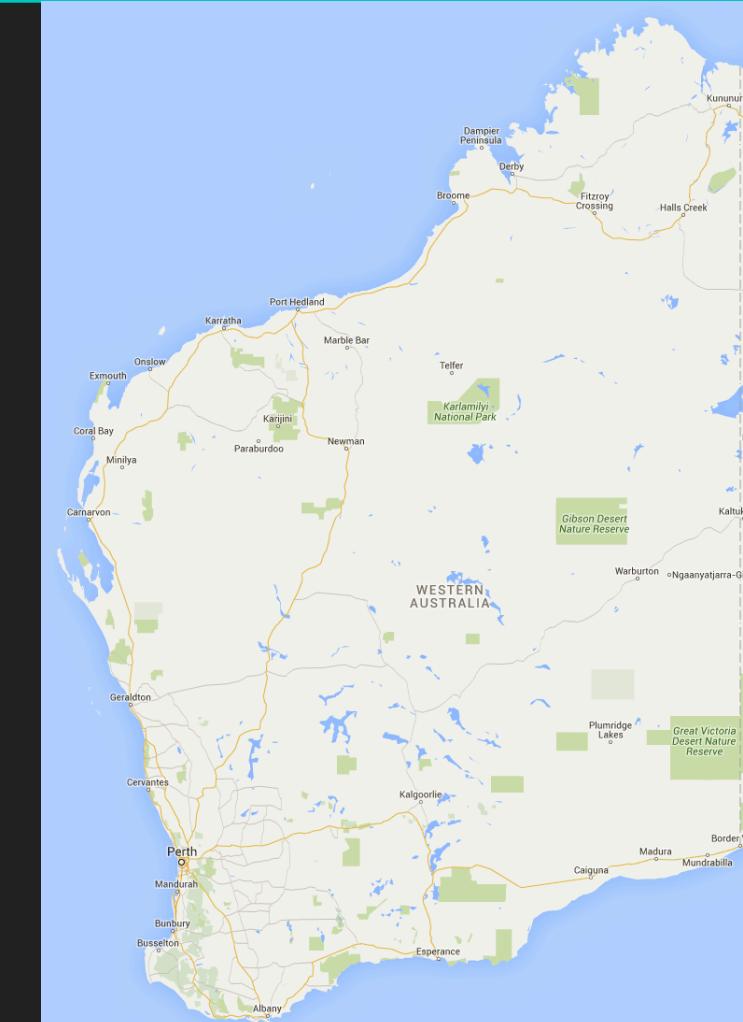
About me

- Senior Systems Engineer
Tools and Automation
Kinetic IT @ Department of Education
- Co founder of Passive Eye Ltd
- Open Source contributor



Dept. Education

- ~800 schools
 - ~400,000 end users
 - ~5000 SOE servers @ schools
 - ~1500 heterogeneous servers @ central office
 - Hub-spoke topology
 - Vast geographic distribution



Problem definition

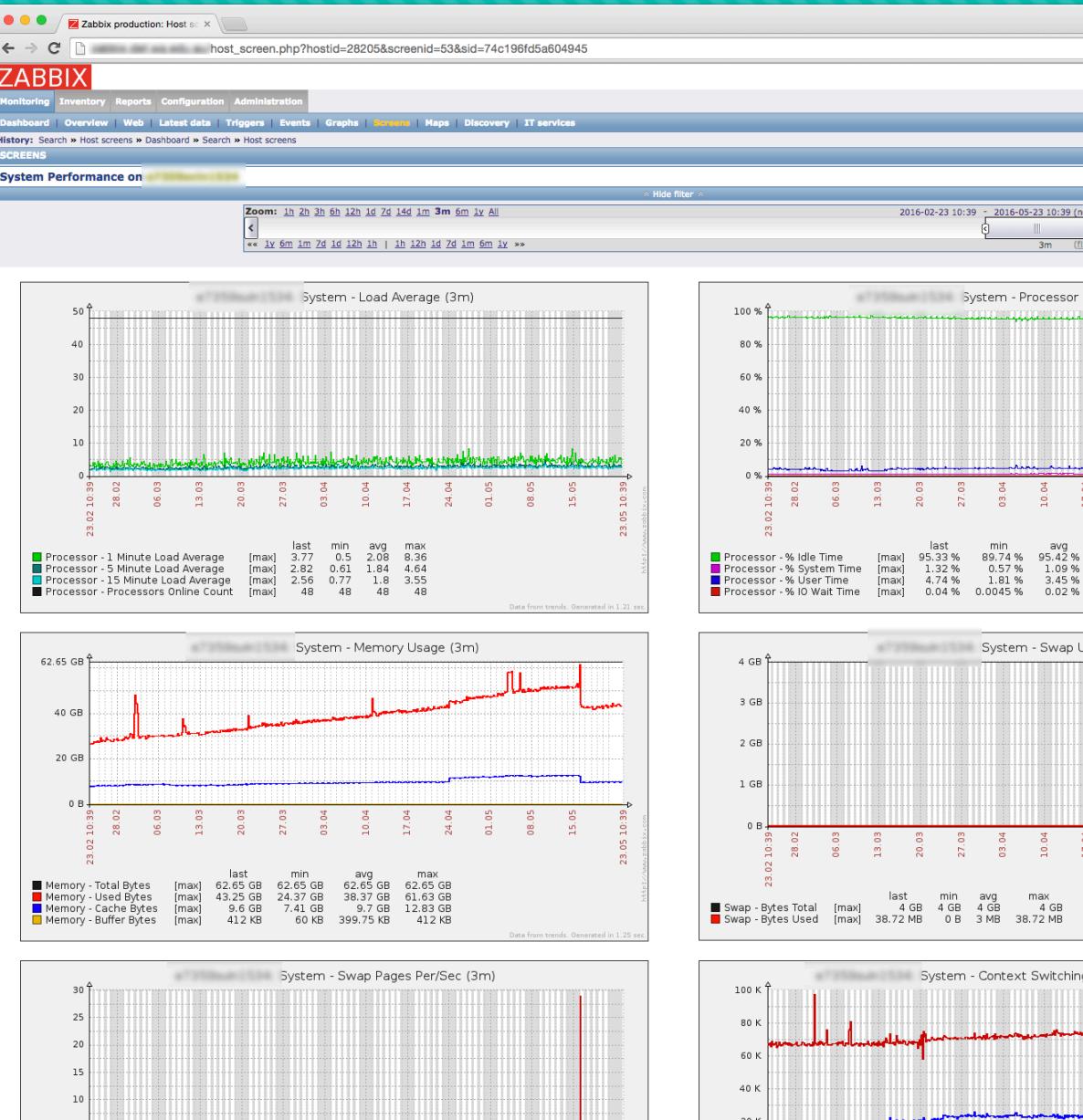
- Multiple, disconnected monitoring tools
 - Poor coverage
 - Lack of correlation
 - Duplication of effort
 - Inconsistent practice
 - Difficulty measuring SLAs

Requirements

- Single pane of glass
- Scalability
- Extensibility
- Ease of use
- Low costs licensing

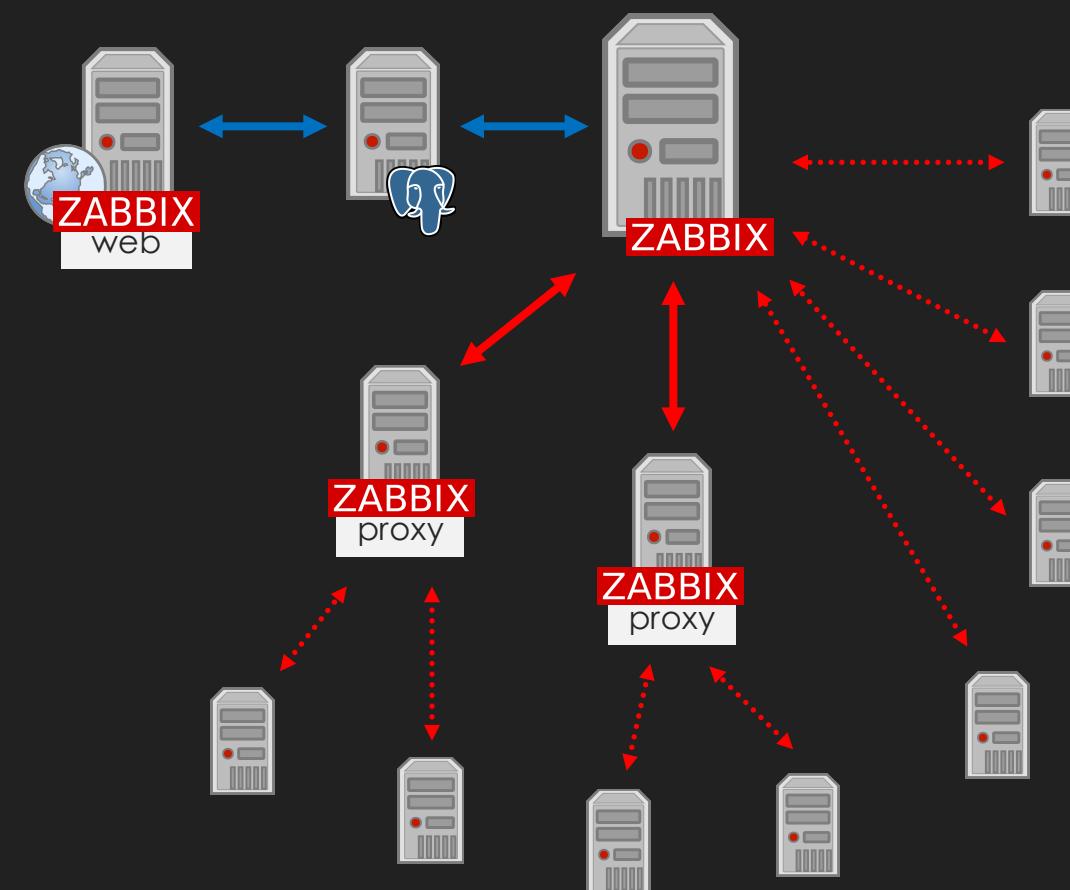
Enter Zabbix

- All-in-one
- Support for diverse devices
- Small footprint and scalable architecture
- Extensible API
- Configuration UI
- Free and open source + support



How Zabbix works

- Primary server
- Database
- Frontend
- Proxy Servers
- Agents
- Passive devices



How Zabbix works

- Items and LLD
- Active and passive checks
- Hosts and templates
- Triggers, Events and Actions
- Graphs, Screens and Maps

The screenshot shows the Zabbix configuration interface for a template named "Template Windows Server". The current view is the "Configuration" tab under "Host groups". A specific item is being edited, with its details shown in the main pane:

- Name:** System - Operating System Version
- Type:** Zabbix agent (active)
- Key:** wmi.get[root\clmv2,SELECT Version FROM CIM_OperatingSys]
- Type of information:** Text
- Update interval (in sec):** 86400
- History storage period (in days):** 365
- New application:** Windows - Memory, Windows - Network Interface, Windows - Paging File, Windows - Physical Disk, Windows - Processor, Windows - System
- Populates host inventory field:** -None-
- Description:** The operating systems version.
- Enabled:**

At the bottom of the screen, there are buttons for **Update**, **Clone**, **Delete**, and **Cancel**. The footer of the browser window displays "Zabbix 2.4.4 Copyright 2001-2015 by Zabbix SIA".

Planning

Scrum + Jira Agile

The screenshot shows the EMS JIRA application interface. At the top, there's a navigation bar with links for Dashboards, Projects, Issues, Agile, Create issue, and a search bar labeled "Quick Search". Below the navigation is a header for "Zabbix Monitoring Solution". The main area is a "Planning" board with two columns: "VERSIONS" and "EPICS".

VERSIONS:

- All issues
- v1.7.0 Grifter
- v1.8.0 Hulk
- v1.9.0 Invincible
- v1.10.0 Jubilee
- v1.11.0 Kilowog
- Issues without versions

EPICS:

- All issues
- Network Device Baseline Monitoring
- Directory Authentication SLAs
- Map views showing incidents
- Zabbix Maintenance Plan
- Server HW Baseline Monitoring
- Windows Baseline Monitoring v2
- Kill Nagios & schools-cacti
- Linux Baseline Monitoring v1
- ESXi Monitoring
- Datacentre UPS Monitoring
- Continuous Integration
- ESX Monitoring
- Storage Monitoring
- SM9 Integration KZBX-414 SM9 Integration
- Issues (19)
- Completed (3)
- Unestimated (5)
- Estimate (147)
- Create issue in epic
- Reporting
- Host Config Management
- Catchup smash

Issues List:

- KZBX-126 SMS alerts for network incident cat 1-2 (Network Device Baseline Monitoring)
- KZBX-153 Monitoring of monitoring data
- KZBX-155 incident assignments vary based on change status (ELS) (Host Config Management)
- KZBX-163 automatic disabling of monitoring via planned changes (SM9 Integration)
- KZBX-166 asset team receive incident when warranty/lease expiry in ext 30 days
- KZBX-169 automated "maintenance" mode for monitoring, linked to SOE Imp scr (Host Config Management)
- KZBX-170 RST shows links to get further details of an event, past or present
- KZBX-172 View flow on impact of a event
- KZBX-200 Record of technical and business owners in CI tool (Host Config Management)
- KZBX-206 Incident raised when device found but not in CI tool (Host Config Management)
- KZBX-209 Automatic network and server device discovery
- KZBX-214 Single Incident to represent multiple events (SM9 Integration)
- KZBX-218 SMS alerts when incident closed against certain CI's
- KZBX-219 SMS alerts when incident raised against certain CI's
- KZBX-233 Monitor McAfee NSP Sensor
- KZBX-496 Create monitoring to monitor zabbix externally
- KZBX-244 Filter SOEv4 Servers (Host Config Management)
- KZBX-285 Template for Satellite School Routers (Network Device Baseline Monitoring)
- KZBX-286 Button to quickly enable maintenance mode for a server
- KZBX-416 Windows Domain Controller Template (Windows Baseline Monitoring)
- KZBX-448 Debug output for BSM getEventData.pl to correlate run results with BSM reporting
- KZBX-451 Have all events be fed into kabana for log analysis and corrolation
- KZBX-454 Monitor school switches for ping (Network Device Baseline Monitoring)
- KZBX-362 Performance and capacity monitoring for storage SAN switches (Storage Monitoring)
- KZBX-282 Table Summary of School Health overlaid on map views (Map views showing incidents)
- KZBX-52 Add a list of available macros to the Action edit view
- KZBX-127 Auto Capture Cisco Security Patch info and feed into the Monthly Patch Summary Review Docu
- KZBX-128 Auto capture of MS Security Bulletin Details and feed into the Monthly Patch Summary Review!
- KZBX-129 Switch that WAP is plugged into (Network Device Baseline Monitoring)
- KZBX-131 # of connected clients by WAP (Network Device Baseline Monitoring)
- KZBX-167 SSS report showing users with long logon times related to profile size
- KZBX-173 email and sms alerts for affected service owner
- KZBX-172 service owner can configure alerts as they wish
- KZBX-183 SSS event subscription option for HDD trending to full in next 30 days
- KZBX-181 SSS event subscription option for HDD at 95% usage
- KZBX-182 SSS event subscription option for HDD at 90% usage

Issue Details: The right side of the screen shows detailed information for the selected issue KZBX-209: "Automatic network and server device discovery". It includes sections for Details (Status: OPEN, Component/s: None, Labels: TPOnDemand), People (Reporter: ARMSTRONG Ryan [ICT Strategic Service Delivery], Assignee: Unassigned), Dates (Created: 23/Jun/14 11:58 AM, Updated: 06/Oct/15 12:37 PM), Issue Links (Add Link), Description (As Ent Tools Engineer I want network devices and servers in the data centre to be automatically discovered So that...), Comments (Comment), and Attachments.

Building

- Discrete environments in Vagrant
 - Infrastructure as code
 - Discrete feature branches
 - Monolithic source repo

```
zabbix-scripts — vagrant@ ~ — ssh — 80x33
vagrant@ ~ $ vagrant ssh
~/Development/zabbix-scripts $ vagrant ssh
/Users/ Development/zabbix-scripts/Vagrantfile:3: warning: already initialized constant VAGRANTFILE_API_VERSION
/Users/.vagrant.d/Vagrantfile:5: warning: previous definition of VAGRANTFILE_API_VERSION was here
/Users/vagrant.d/boxes/doe-VAGRANTSPLIT-zabbix-1.7/0/virtualbox/include/_Vagrantfile:3: warning: already initialized constant VAGRANTFILE_API_VERSION
/Users/ Development/zabbix-scripts/Vagrantfile:3: warning: previous definition of VAGRANTFILE_API_VERSION was here
Last login: Mon May 16 11:58:51 2016 from 10.0.2.2

  
  
To reapply system configuration:  
vagrant provision # from host machine  
  
To refresh Puppet modules:  
cd /vagrant/puppet; /usr/local/bin/r10k puppetfile install -v  
  
Vagrant Zabbix URL (LDAP Login):  
http://localhost:8080  
  
vagrant@vagrant-box ~ $
```

Puppet Code

```
class ems::profile::zabbix::server {  
    UNREGISTERED  
    class ems::profile::zabbix::server {  
        require ::ems::profile::zabbix::globals  
  
        # install Zabbix server  
        class { '::kzbx::server' :  
            enable_pooling => false,  
            manage_config  => false, # we'll specify config next  
        }  
  
        # modify server config file  
        class { '::zabbix::server::config' :  
            # redirect alert/external scripts to the custom scripts repo  
            # this needs to be set here because alert_scripts_path and  
            # external_scripts_path are not inherited from zabbix::globals.  
            # database connection details are already inherited.  
            alert_scripts_path      => $::kzbx::params::alert_scripts_path,  
            external_scripts_path   => $::kzbx::params::external_scripts_path,  
  
            cache_size                => '2G',  
            history_cache_size        => '1G',  
            history_text_cache_size   => '1G',  
            log_file_size             => 100,  
            log_slow_queries          => 1000,  
            start_db_syncers          => 4,  
            start_discoverers         => 10,  
            start_http_pollers        => 5,  
            start_ipmi_pollers       => 5,  
            start_pingers              => 75,  
            start_pollers              => 400,  
            start_pollers_unreachable => 150,  
            start_proxy_pollers       => 1,  
            start_snmp_trapper        => 1,  
            start_timers               => 50,  
            start_trappers             => 50,  
            start_vmware_collectors   => 5,  
            timeout                   => 30,  
            tmp_dir                   => '/dev/shm',  
            trend_cache_size          => '1G',  
            unavailable_delay         => 60,  
            unreachable_period        => 300.  
    }  
}
```

Testing

- Bamboo
- Cucumber

```
zabbix-scripts — root@ms080027653f27:vagrant/cucumber — ssh • vagrant ssh — 132x46
~/Development/zabbix-scripts — root@ms080027653f27:vagrant/cucumber — ssh • vagrant ssh

@server
Feature: Zabbix server deployment

Scenario Outline: Environment variables are configured correctly # features/zabbix_server.feature:5
  Given I am running as 'root' # features/zabbix_server.feature:6
  When I query environment variable '<envVar>' # features/zabbix_server.feature:7
  Then the value appears to be a valid <envFormat> # features/zabbix_server.feature:8

Examples:
| envVar      | envFormat |
| ZBX_ENV     | string    |
| ZBX_USER    | string    |
| ZBX_GROUP   | string    |
| ZBX_HOME    | path      |
| ZBX_REPO    | path      |
| ZBX_URL     | url       |

Environment variable  is set is not a valid URL (http://localhost) (RuntimeError)
./features/step_definitions/step_zabbix_server.rb:27:in `^the value appears to be a valid (\w+)$'
features/zabbix_server.feature:17:in `Then the value appears to be a valid url'
features/zabbix_server.feature:8:in `Then the value appears to be a valid <envFormat>'

| ZBX_DB_HOST | string    |
| ZBX_DB_USER | string    |
| ZBX_DB_SCHEMA | string |
| ZBX_DB_DATABASE | string |

Scenario: Zabbix database is reachable # features/zabbix_server.feature:23
  Given Zabbix server is installed # features/step_definitions/step_zabbix_server.rb:38
    Zabbix server v2.4.4 (revision 52341) (23 February 2015)
    Compilation time: Aug 24 2015 14:28:05
  When I attempt to connect to the Zabbix database using the server settings # features/step_definitions/step_zabbix_server.rb:50
  Then the connection succeeds and the database version is available # features/step_definitions/step_zabbix_server.rb:54
    PostgreSQL 9.2.13 on x86_64-redhat-linux-gnu, compiled by gcc (GCC) 4.8.3 20140911 (Red Hat 4.8.3-9), 64-bit
    Zabbix database version: 2040000

@server
Feature: Zabbix server custom scripts

Scenario Outline: Script repositories are configured # features/zabbix_server_scripts.feature:5
  Given I am running as 'root' # features/zabbix_server_scripts.feature:6
  And '<repoName>' is deployed to '<repoPath>' # features/zabbix_server_scripts.feature:7
  When I check the configuration of the repo '<repoName>' # features/zabbix_server_scripts.feature:8
  Then the 'origin' remote is set to '<repoOrigin>' # features/zabbix_server_scripts.feature:9
  And the checked out branch is '<repoBranch>' # features/zabbix_server_scripts.feature:10

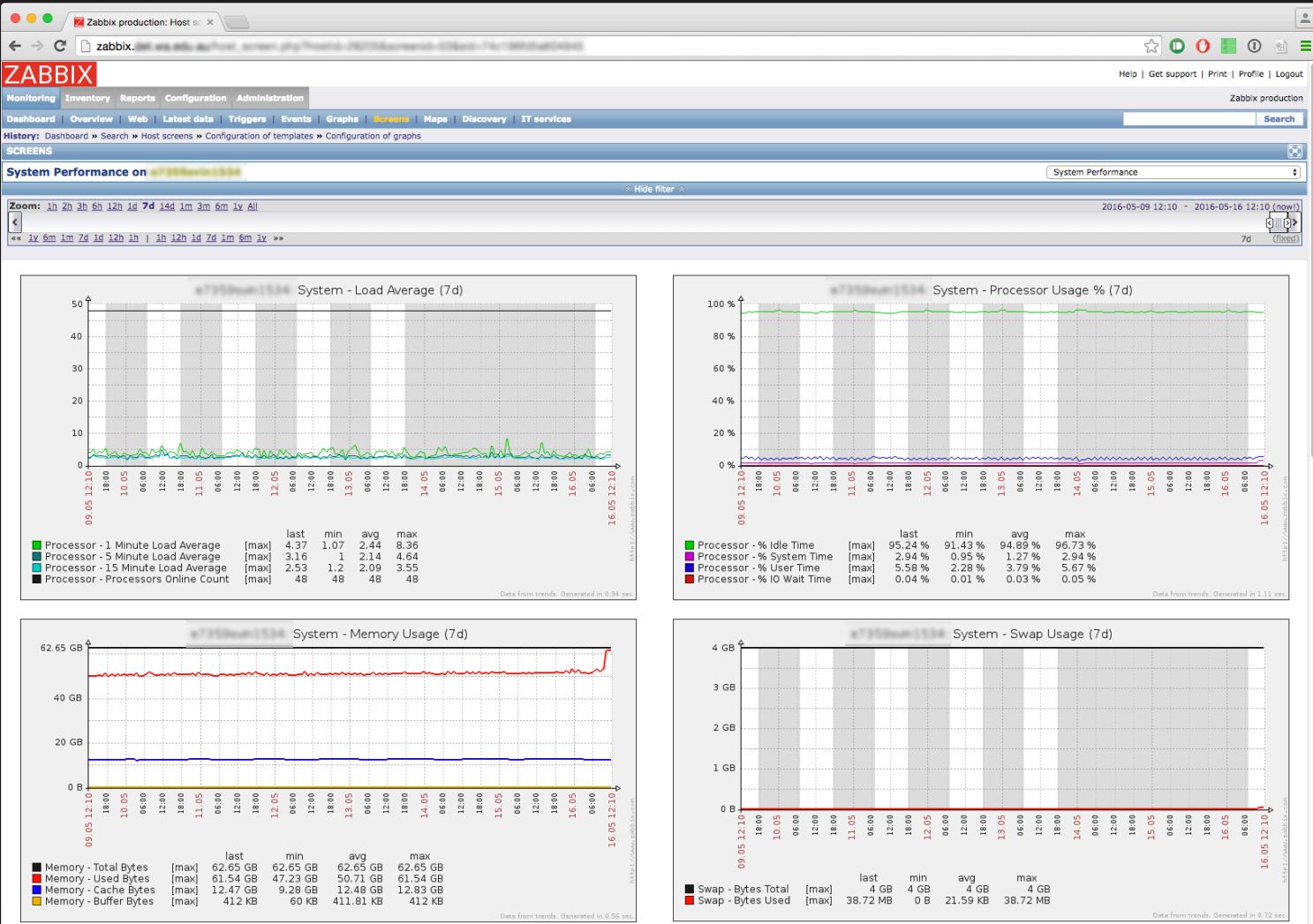
Examples:
| repoName    | repoPath   | repoOrigin | repoBranch |

```

Deployment

- Server, DB, Web
- No Proxies (so far)

Status of Zabbix		
Parameter	Value	Details
Zabbix server is running	Yes	:10051
Number of hosts (enabled/disabled/templates)	6573	6300 / 34 / 239
Number of items (enabled/disabled/not supported)	2533462	1331489 / 1190468 / 11505
Number of triggers (enabled/disabled [problem(ok)])	708956	424701 / 284255 [1147 / 423554]
Number of users (online)	229	11
Required server performance, new values per second	2156.07	-
Updated: 11:37:07		



Performance

Integrations

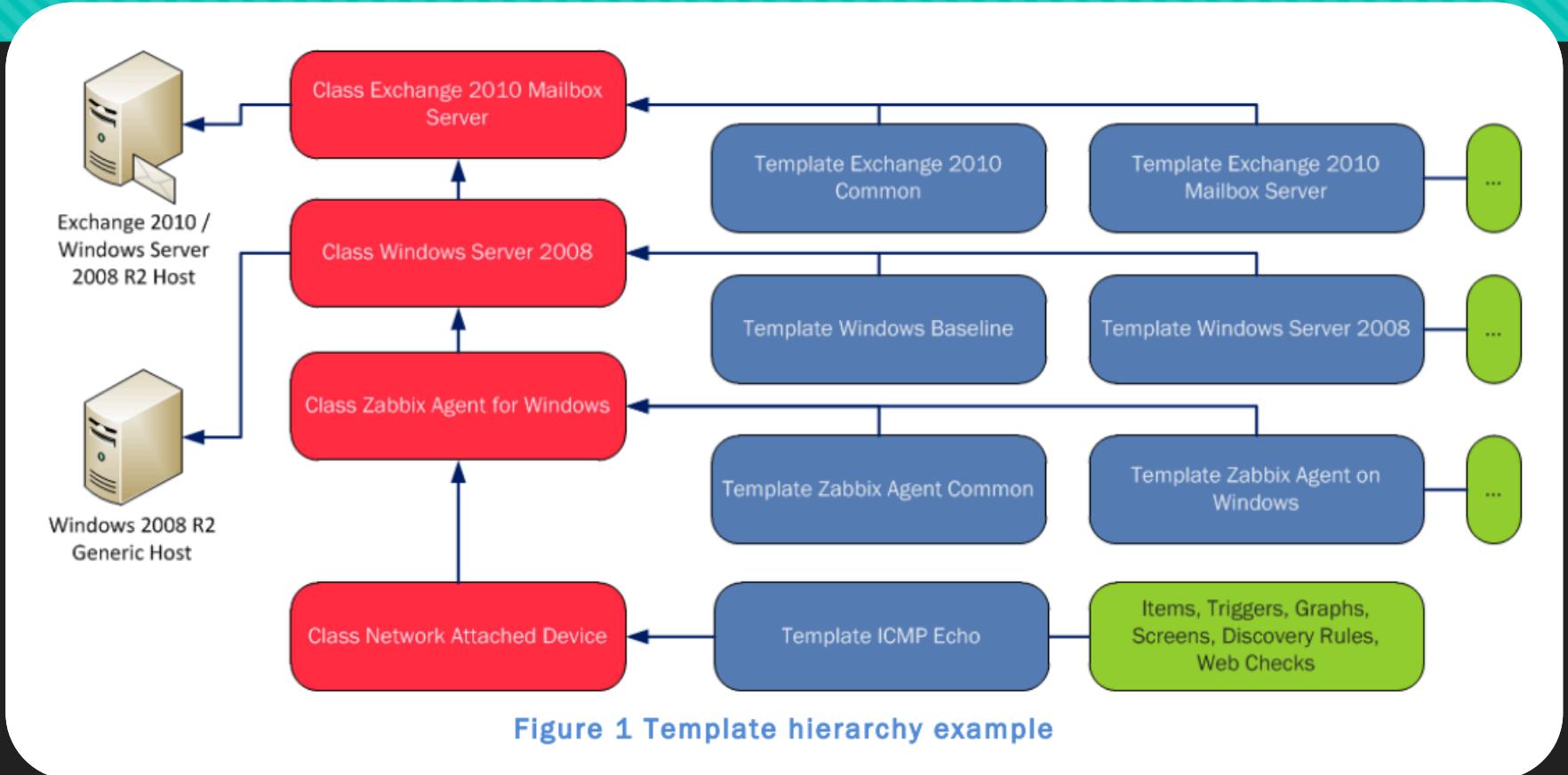
- Active Directory
- CMDB
- Service Management
- ICT Dashboard

Autonomy

- Host registration
- Low level discovery
- User provisioning
- Remediation scripts
- Data housekeeping
- Incidents and escalations

Template hierarchy

- Host
- Class
- Templates
- Items, etc.

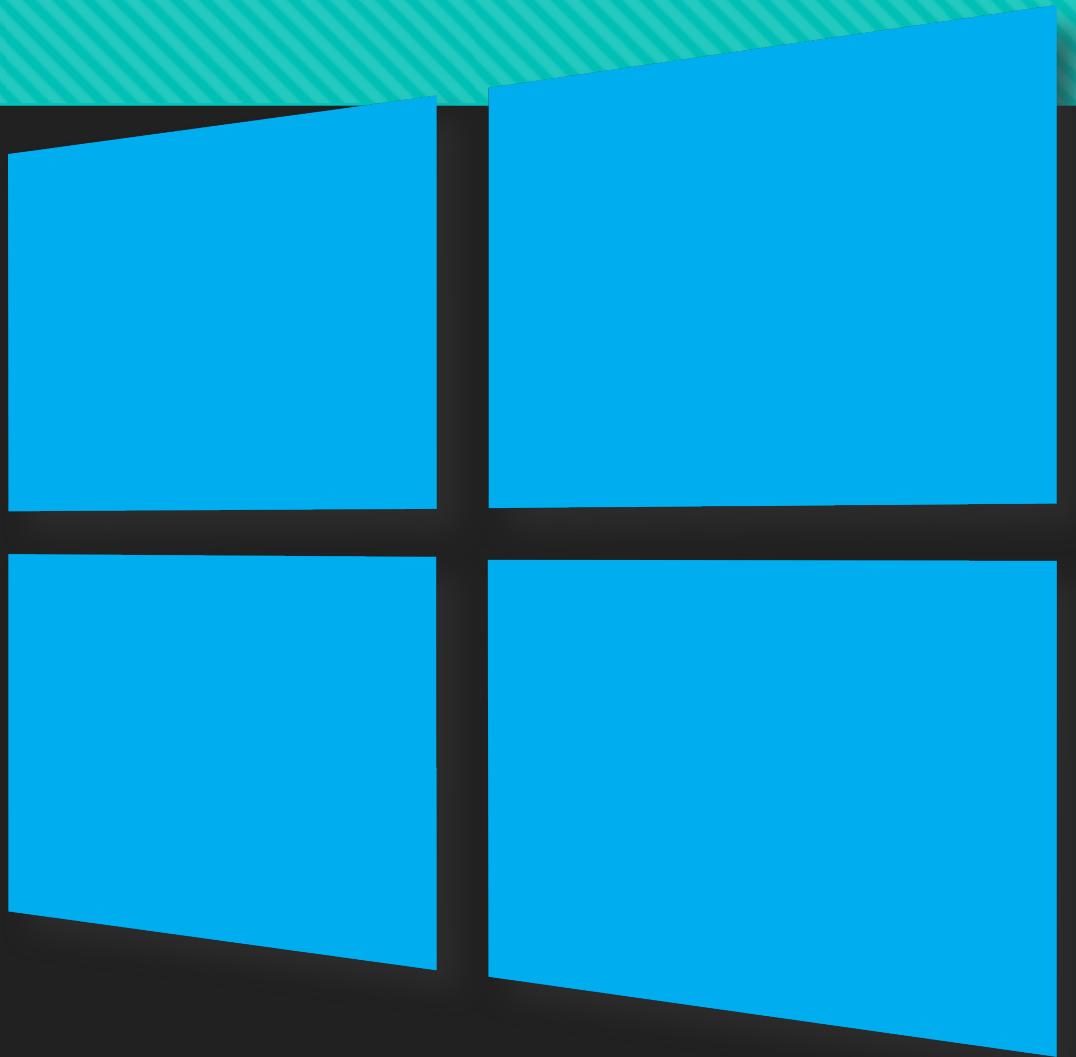


Windows monitoring

- OOB Support:
 - WMI queries
 - Performance counter probes
 - Event Log monitor
 - Service state

Windows monitoring

- Customisations
 - Hostname casing and format
 - Service discovery
 - Performance counter discovery
 - Failover Cluster discovery
 - Persistent disk and volume identification



Windows monitoring

- Tools
 - MSI installer package
 - Test PowerShell script
 - Performance Counter template builder

Windows monitoring

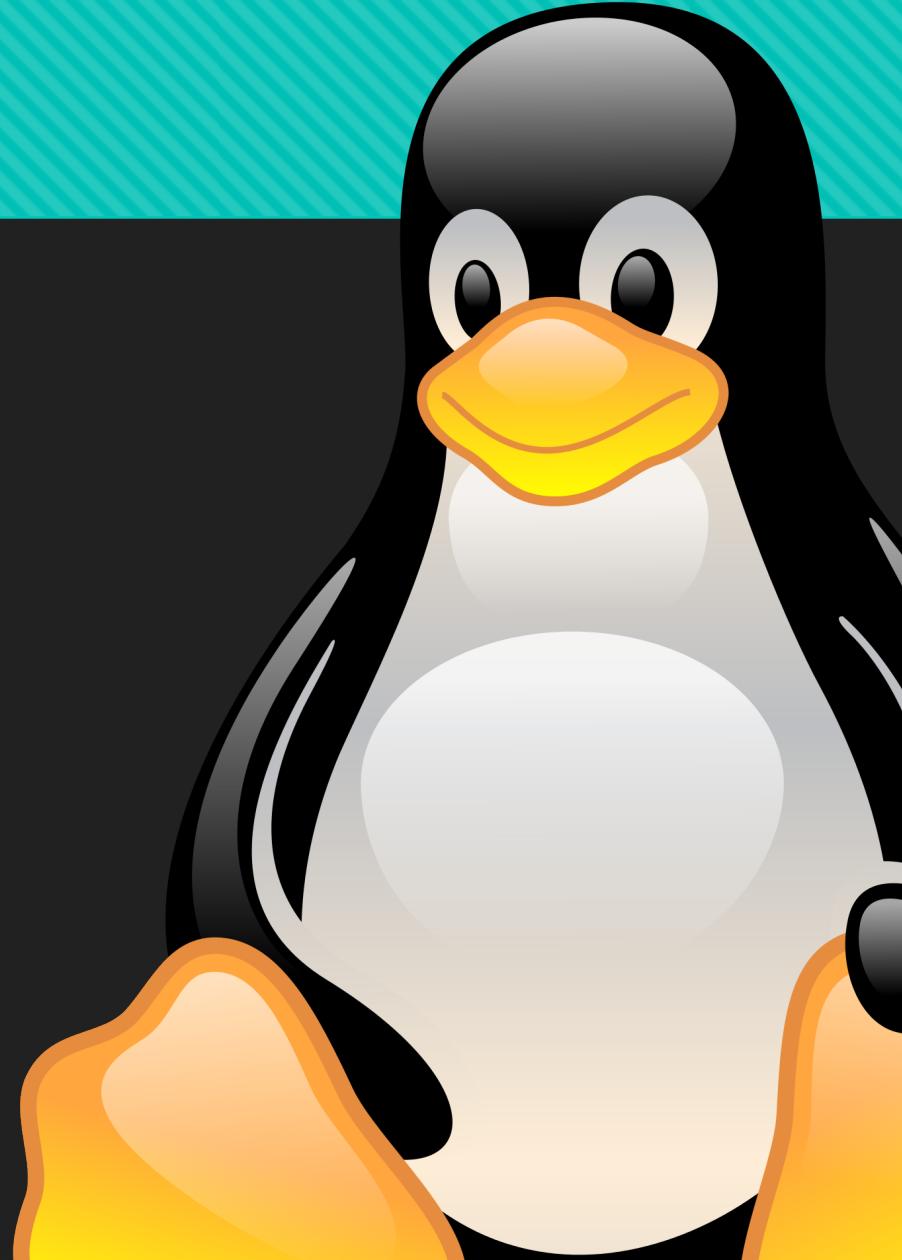
- Convert-PerfCountToZabbixTemplate.ps1
 - Counter Set > Application
 - Multi-instance counter > Discovery Rule + Prototypes
 - Single-instance counter > Item check



```
> ./Convert-PerfCountersToZabbixTemplate.ps1 -CounterSet Processor | Out-File template.xml
```

Linux monitoring

- Modules
 - Extensions for Linux kernel
 - PostgreSQL
- Packaging
- Test script



SNMP monitoring

- CRAC
- UPS
- Dell iDRAC
- IPS
- Mail gateways

SNMP monitoring

- mib2zabbix.pl
 - Tree nodes > Applications
 - OID Tables > Discovery Rules + Prototypes
 - OID Scalars > Item checks
 - OID Traps > Item + Trigger + snmptrap config
 - Enums > Value Maps

```
$ ./mib2zabbix.pl --template -oid=.1.3.6.1.2.1.25 --name="Host resources"
```

Application monitoring

The screenshot shows the Zabbix web interface for monitoring a Microsoft SQL Server instance. The top navigation bar includes links for Monitoring, Inventory, Reports, Configuration, Administration, Dashboard, Overview, Web, Latest Data, Triggers, Events, Graphs, Screens, Maps, Discovery, and IT services. The main content area is titled 'LATEST DATA' and displays a table of items for the host group 'CO - Microsoft SQL Servers 2014'. The table columns include Name, Inter..., History, Trends, Type, Last check, Last value, Change, and Info. The table lists several items under categories like 'MSSQL 2014 Broker Activation', 'MSSQL 2014 Catalog Metadata', and 'MSSQL 2014 Database'. Each item row contains a checkbox, a name, and various performance metrics such as size in KB and last checked time.

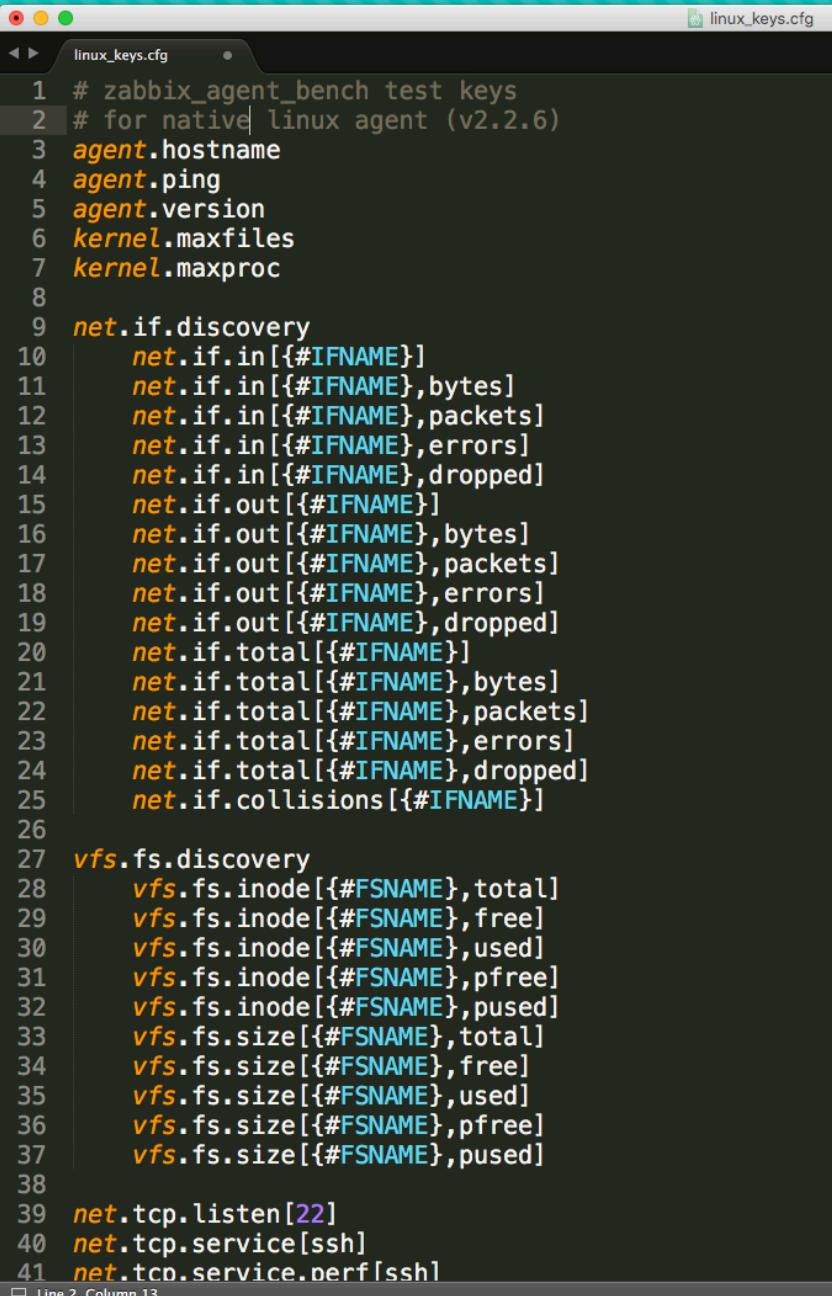
- Microsoft Exchange
- Microsoft SCCM
- Microsoft SQL Server
- Microsoft Active Directory
- PostgreSQL Server
- EMC Avamar
- HP BPM
- Squid Proxy
- Zabbix Server
- ...

Risk mitigation

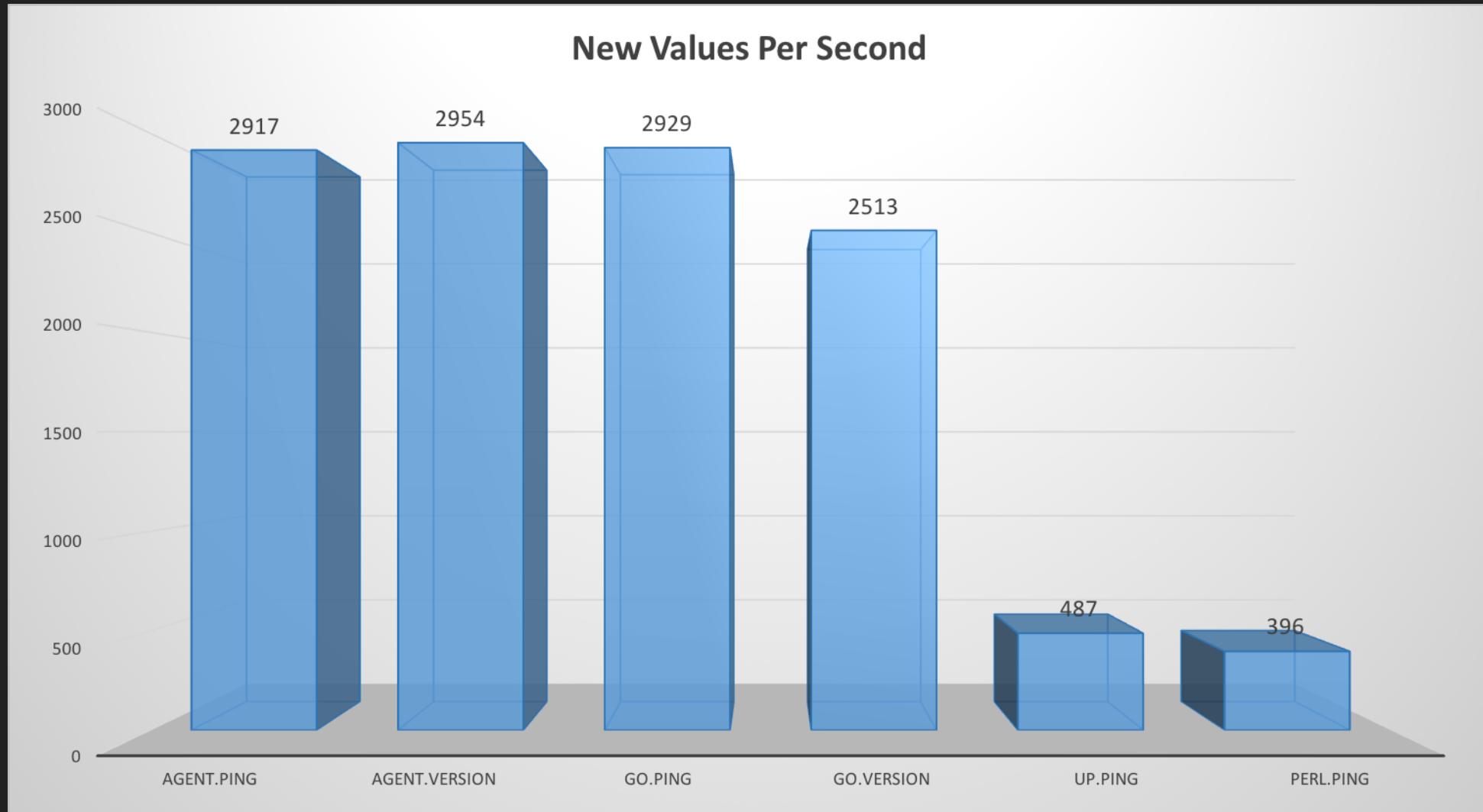
- Document in code
- Source control
- Clearly defined interfaces
- Quality gates
- Upstream contribution
- Change the hiring criteria to avoid SPOF

Agent stress test

- Critical to finding:
 - Memory leaks
 - Race conditions
 - Impact on system
 - Regressions
- Validate efficiency improvements



```
1 # zabbix_agent_bench test keys
2 # for native linux agent (v2.2.6)
3 agent.hostname
4 agent.ping
5 agent.version
6 kernel.maxfiles
7 kernel.maxproc
8
9 net.if.discovery
10 net.if.in[{"#IFNAME"}]
11 net.if.in[{"#IFNAME"},bytes]
12 net.if.in[{"#IFNAME"},packets]
13 net.if.in[{"#IFNAME"},errors]
14 net.if.in[{"#IFNAME"},dropped]
15 net.if.out[{"#IFNAME"}]
16 net.if.out[{"#IFNAME"},bytes]
17 net.if.out[{"#IFNAME"},packets]
18 net.if.out[{"#IFNAME"},errors]
19 net.if.out[{"#IFNAME"},dropped]
20 net.if.total[{"#IFNAME"}]
21 net.if.total[{"#IFNAME"},bytes]
22 net.if.total[{"#IFNAME"},packets]
23 net.if.total[{"#IFNAME"},errors]
24 net.if.total[{"#IFNAME"},dropped]
25 net.if.collisions[{"#IFNAME"}]
26
27 vfs.fs.discovery
28 vfs.fs.inode[{"#FSNAME"},total]
29 vfs.fs.inode[{"#FSNAME"},free]
30 vfs.fs.inode[{"#FSNAME"},used]
31 vfs.fs.inode[{"#FSNAME"},pfree]
32 vfs.fs.inode[{"#FSNAME"},pused]
33 vfs.fs.size[{"#FSNAME"},total]
34 vfs.fs.size[{"#FSNAME"},free]
35 vfs.fs.size[{"#FSNAME"},used]
36 vfs.fs.size[{"#FSNAME"},pfree]
37 vfs.fs.size[{"#FSNAME"},pused]
38
39 net.tcp.listen[22]
40 net.tcp.service[ssh]
41 net.tcp.service.perf[ssh]
```



It's no magic bullet...

- Data aggregation
- Visualizations
- Alert Scripts

Future

- Zabbix v3 upgrade
- Better engagement from ITOps
- More devices and apps
- More automation
- Better use of data
- Enterprise Integration Patterns
- Cloud monitor

DevOps?

- Meta-software
- Agile delivery
- Infrastructure As Code
- Continuous Integration
- Theory of Constraints

Contrib

- PostgreSQL monitoring
- Agent benchmarking
- Windows MSI package
- Golang module adapter

<https://github.com/cavaliercoder/libzbxpgsql>

https://github.com/cavaliercoder/zabbix_agent_bench

<https://github.com/cavaliercoder/zabbix-msi>

<https://github.com/cavaliercoder/g2z>

Windows Counters

- Performance Counter IDs are non-persistent
 - Today G: is *PhysicalDisk(3 G:)*, tomorrow it is *PhysicalDisk(5)*
- Graphs and alerts break
- Mapping is not practical in scripting APIs

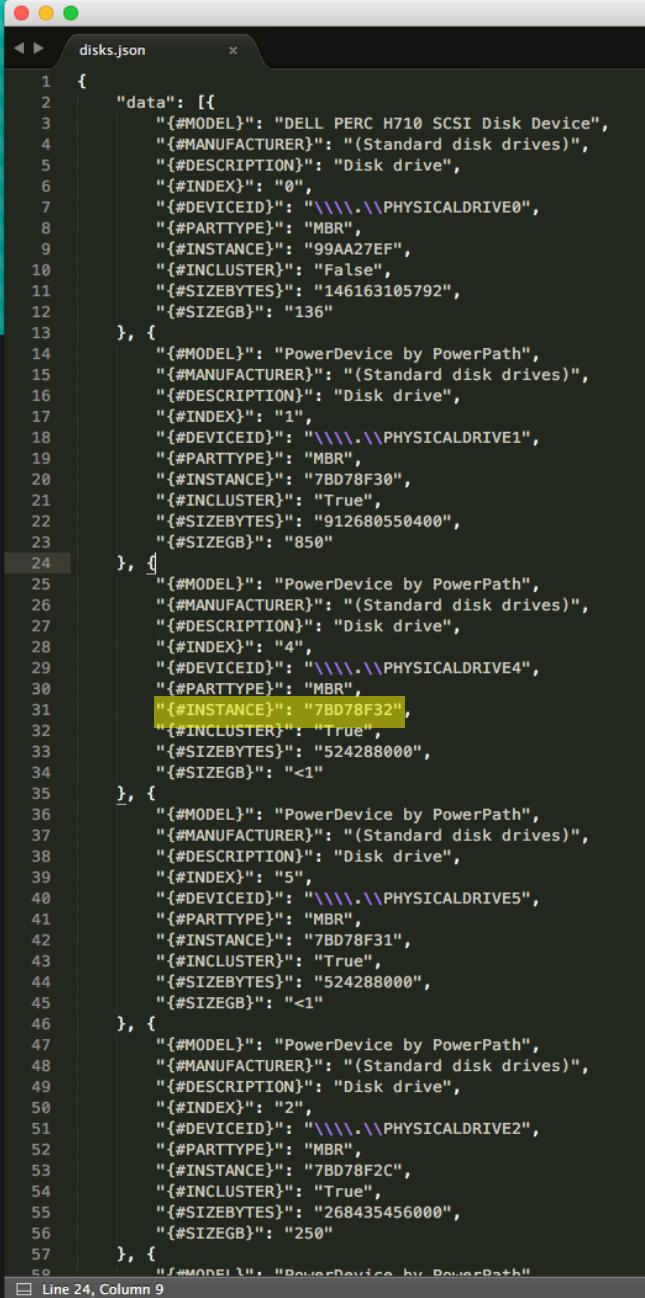
Windows physical disks

- Mutable performance counter ID:
PhysicalDisk(0 C:)
- Index ('0') changes on reboot, swap, failover, etc.
- The drive letter ('C:') is undocumented
- MBR Signatures and GPT GUIDs are more persistent



Windows physical disks

- Q: What runtime counter index maps to desired MBR/GUID?
- Identify via MBR Signature or GPT GUID
 - DeviceIOControl (`IOCTL_DISK_GET_DRIVE_LAYOUT_EX`)
- Get device index (`\.\.\PHYSICALDRIVE<i>`)
 - DeviceIOControl (`IOCTL_STORAGE_GET_DEVICE_NUMBER`)
- Iterate *PhysicalDisk* counters (ignore drive letter)
 - PdhEnumObjectItems



```
disks.json
[{"data": [{"#MODEL": "DELL PERC H710 SCSI Disk Device", "#MANUFACTURER": "(Standard disk drives)", "#DESCRIPTION": "Disk drive", "#INDEX": "0", "#DEVICEID": "\\\\.\\PHYSICALDRIVE0", "#PARTTYPE": "MBR", "#INSTANCE": "99AA27EF", "#INCLUSTER": "False", "#SIZEBYTES": "146163105792", "#SIZEGB": "136"}, {"#MODEL": "PowerDevice by PowerPath", "#MANUFACTURER": "(Standard disk drives)", "#DESCRIPTION": "Disk drive", "#INDEX": "1", "#DEVICEID": "\\\\.\\PHYSICALDRIVE1", "#PARTTYPE": "MBR", "#INSTANCE": "7BD78F30", "#INCLUSTER": "True", "#SIZEBYTES": "912680550400", "#SIZEGB": "850"}, {"#MODEL": "PowerDevice by PowerPath", "#MANUFACTURER": "(Standard disk drives)", "#DESCRIPTION": "Disk drive", "#INDEX": "4", "#DEVICEID": "\\\\.\\PHYSICALDRIVE4", "#PARTTYPE": "MBR", "#INSTANCE": "7BD78F32", "#INCLUSTER": "True", "#SIZEBYTES": "524288000", "#SIZEGB": "<1"}, {"#MODEL": "PowerDevice by PowerPath", "#MANUFACTURER": "(Standard disk drives)", "#DESCRIPTION": "Disk drive", "#INDEX": "5", "#DEVICEID": "\\\\.\\PHYSICALDRIVE5", "#PARTTYPE": "MBR", "#INSTANCE": "7BD78F31", "#INCLUSTER": "True", "#SIZEBYTES": "524288000", "#SIZEGB": "<1"}, {"#MODEL": "PowerDevice by PowerPath", "#MANUFACTURER": "(Standard disk drives)", "#DESCRIPTION": "Disk drive", "#INDEX": "2", "#DEVICEID": "\\\\.\\PHYSICALDRIVE2", "#PARTTYPE": "MBR", "#INSTANCE": "7BD78F2C", "#INCLUSTER": "True", "#SIZEBYTES": "268435456000", "#SIZEGB": "250"}, {"#MODEL": "PowerDevice by PowerPath"}]}
```

Line 24, Column 9

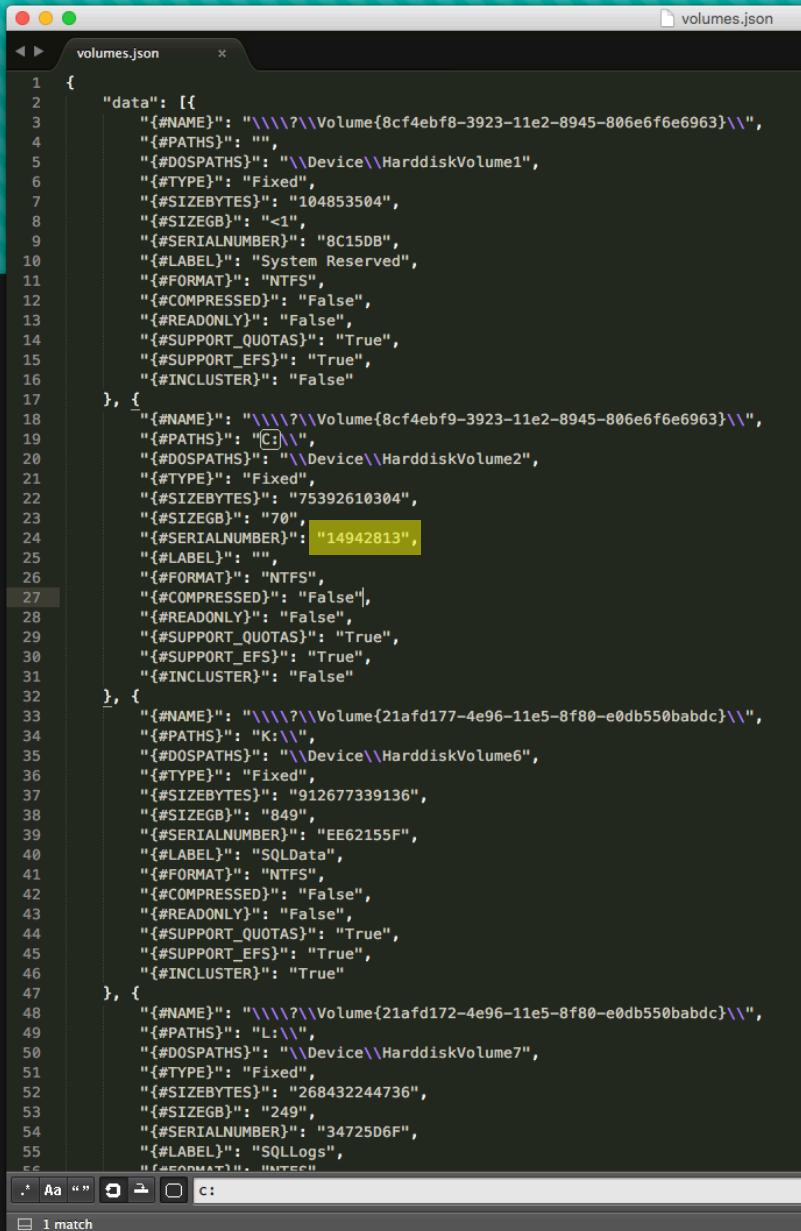
Sample code: <https://github.com/cavaliercoder/sysinv/blob/master/diskinfo.cpp>

Windows Volumes

- Performance counter ID: LogicalDisk(C: | HarddiskVolumeN)
- Drive letter is mutable
- N is mutable
- Volume GUIDs or Serials are more persistent

Windows Volumes

- Q: Which runtime counter ID matches Volume GUID?
○ Find volumes with ID
 FindNextVolume
- Compare GUID/Sig against name returned by
 GetVolumeInformation
- Enumerate *LogicalDisk* counters with
 PdhEnumObjectItems
- Test mount paths (N:) returned by
 GetVolumePathNamesForVolumeName
- Test DOS Device Path (\Device\HarddiskVolumeN) returned by
 QueryDosDevice



```
1 {  
2     "data": [  
3         {"#NAME": "\\\\?\Volume{8cf4ebf8-3923-11e2-8945-806e6f6e6963}\\",  
4          "#PATHS": "",  
5          "#DOSPATHS": "\\\Device\\HarddiskVolume1",  
6          "#TYPE": "Fixed",  
7          "#SIZEBYTES": "104853504",  
8          "#SIZEGB": "<1",  
9          "#SERIALNUMBER": "8C15DB",  
10         "#LABEL": "System Reserved",  
11         "#FORMAT": "NTFS",  
12         "#COMPRESSED": "False",  
13         "#READONLY": "False",  
14         "#SUPPORT_QUOTAS": "True",  
15         "#SUPPORT_EFS": "True",  
16         "#INCLUSTER": "False"  
17     }, {  
18         "#NAME": "\\\\?\Volume{8cf4ebf9-3923-11e2-8945-806e6f6e6963}\\",  
19         "#PATHS": "[C:\\]",  
20         "#DOSPATHS": "\\\Device\\HarddiskVolume2",  
21         "#TYPE": "Fixed",  
22         "#SIZEBYTES": "75392610304",  
23         "#SIZEGB": "70",  
24         "#SERIALNUMBER": "14942813",  
25         "#LABEL": "",  
26         "#FORMAT": "NTFS",  
27         "#COMPRESSED": "False",  
28         "#READONLY": "False",  
29         "#SUPPORT_QUOTAS": "True",  
30         "#SUPPORT_EFS": "True",  
31         "#INCLUSTER": "False"  
32     }, {  
33         "#NAME": "\\\\?\Volume{21afdf77-4e96-11e5-8f80-e0db550babdc}\\",  
34         "#PATHS": "K:\\",  
35         "#DOSPATHS": "\\\Device\\HarddiskVolume6",  
36         "#TYPE": "Fixed",  
37         "#SIZEBYTES": "912677339136",  
38         "#SIZEGB": "849",  
39         "#SERIALNUMBER": "EE62155F",  
40         "#LABEL": "SQLdata",  
41         "#FORMAT": "NTFS",  
42         "#COMPRESSED": "False",  
43         "#READONLY": "False",  
44         "#SUPPORT_QUOTAS": "True",  
45         "#SUPPORT_EFS": "True",  
46         "#INCLUSTER": "True"  
47     }, {  
48         "#NAME": "\\\\?\Volume{21afdf172-4e96-11e5-8f80-e0db550babdc}\\",  
49         "#PATHS": "L:\\",  
50         "#DOSPATHS": "\\\Device\\HarddiskVolume7",  
51         "#TYPE": "Fixed",  
52         "#SIZEBYTES": "26843244736",  
53         "#SIZEGB": "249",  
54         "#SERIALNUMBER": "34725D6F",  
55         "#LABEL": "SQLlogs",  
56         "#FORMAT": "UNICODE"  
57     }]  
58 }
```

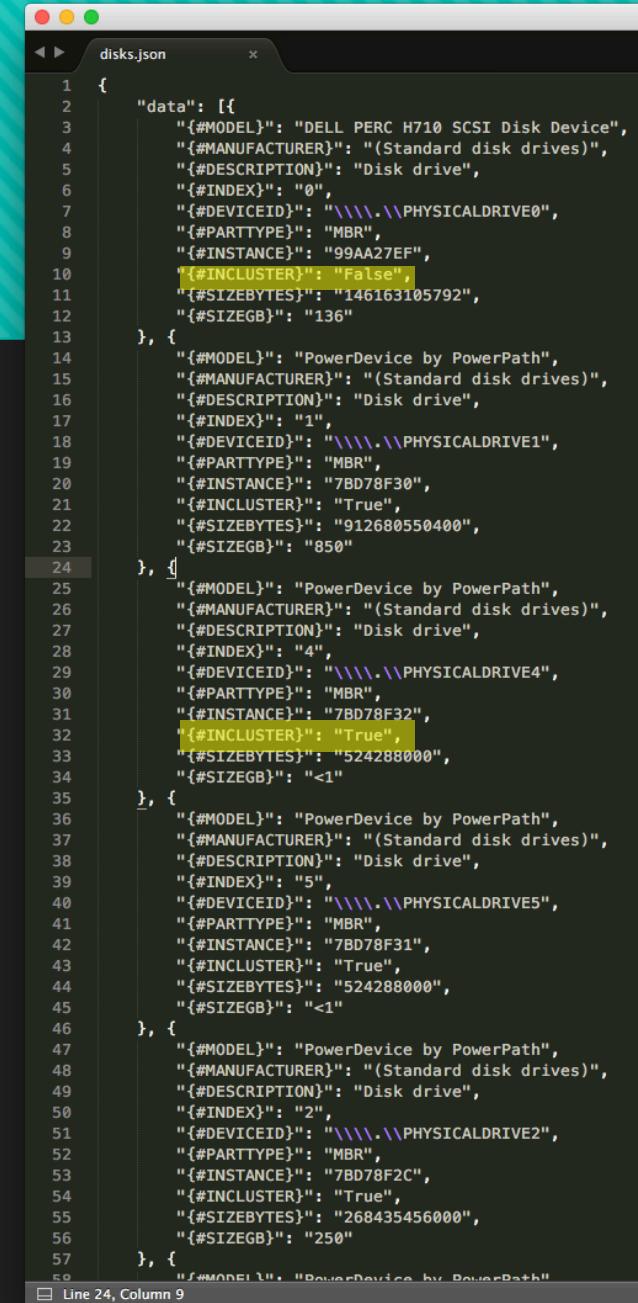
Sample code: <https://github.com/cavaliercoder/sysinv/blob/master/diskinfo.cpp>

Windows Failover Clusters

- Disks move between nodes
- Node disks are visible on cluster IP
- IDs and drive letters change

Windows Failover Clusters

- Q: Is a MBR/GUID listed as a cluster resource?
- Cluster API uses MBR Signature or GPT GUID!
- Enumerate “Physical Disk” resources in cluster with `ClusterEnum`
- Add a discovery parameter for clustered/non-clustered disks



```
disks.json
{
  "data": [
    {
      "#MODEL": "DELL PERC H710 SCSI Disk Device",
      "#MANUFACTURER": "(Standard disk drives)",
      "#DESCRIPTION": "Disk drive",
      "#INDEX": "0",
      "#DEVICEID": "\\\\.\\"PHYSICALDRIVE0",
      "#PARTTYPE": "MBR",
      "#INSTANCE": "99AA27EF",
      "#INCLUSTER": "False",
      "#SIZEBYTES": "146163105792",
      "#SIZEGB": "136"
    },
    {
      "#MODEL": "PowerDevice by PowerPath",
      "#MANUFACTURER": "(Standard disk drives)",
      "#DESCRIPTION": "Disk drive",
      "#INDEX": "1",
      "#DEVICEID": "\\\\.\\"PHYSICALDRIVE1",
      "#PARTTYPE": "MBR",
      "#INSTANCE": "7BD78F30",
      "#INCLUSTER": "True",
      "#SIZEBYTES": "912680550400",
      "#SIZEGB": "850"
    },
    {
      "#MODEL": "PowerDevice by PowerPath",
      "#MANUFACTURER": "(Standard disk drives)",
      "#DESCRIPTION": "Disk drive",
      "#INDEX": "4",
      "#DEVICEID": "\\\\.\\"PHYSICALDRIVE4",
      "#PARTTYPE": "MBR",
      "#INSTANCE": "7BD78F32",
      "#INCLUSTER": "True",
      "#SIZEBYTES": "524288000",
      "#SIZEGB": "<1"
    },
    {
      "#MODEL": "PowerDevice by PowerPath",
      "#MANUFACTURER": "(Standard disk drives)",
      "#DESCRIPTION": "Disk drive",
      "#INDEX": "5",
      "#DEVICEID": "\\\\.\\"PHYSICALDRIVE5",
      "#PARTTYPE": "MBR",
      "#INSTANCE": "7BD78F31",
      "#INCLUSTER": "True",
      "#SIZEBYTES": "524288000",
      "#SIZEGB": "<1"
    },
    {
      "#MODEL": "PowerDevice by PowerPath",
      "#MANUFACTURER": "(Standard disk drives)",
      "#DESCRIPTION": "Disk drive",
      "#INDEX": "2",
      "#DEVICEID": "\\\\.\\"PHYSICALDRIVE2",
      "#PARTTYPE": "MBR",
      "#INSTANCE": "7BD78F2C",
      "#INCLUSTER": "True",
      "#SIZEBYTES": "268435456000",
      "#SIZEGB": "250"
    }
  ]
}
```

Sample code: <https://github.com/cavaliercoder/sysinv/blob/master/cluster.cpp>

Thank you!

Ryan Armstrong

- Blog: cavaliercoder.com
- Twitter: @cavaliercoder
- GitHub: [cavaliercoder](https://github.com/cavaliercoder)