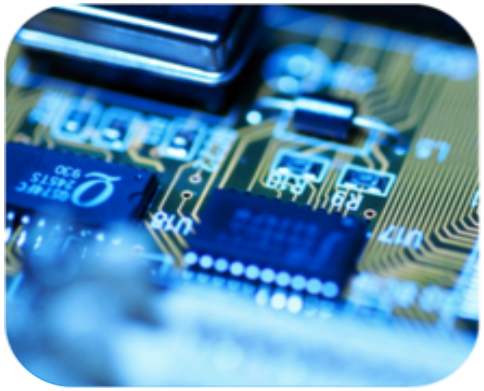
ChinaNetCloud

Running the World's Internet Servers

Zabbix

Loving the Database



By Steve Mushero

September, 2014

Greetings

I’m Steve

I’m from Shanghai, China

We have a big Internet there

We have a big business

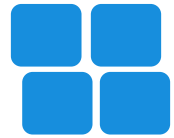
We have a big monitoring system

That’s Zabbix

We love the Database

Let me tell you more about it . . .

Our System & Database



Zabbix 1.8.3 – Going to 2.2 now

1250 hosts, 300K items, 550 new values/sec

MySQL Percona 5.5 w/ very optimized config

- Will go to 5.6 when 5.6 is more stable
- No query cache

Main DB about 200GB in size

History is about 2 billion rows of data, 157GB size

- Trends is 500 million, 37GB size

Backups take 8 hours (including compress/encrypt)

Has 48GB of Innodb Buffer (will expand to 64+ soon)

- Will go to 96, 128, 256 over time

Generally runs well, except I/O bottleneck

Zabbix Database

MySQL for us

Not Postgres/Oracle – No idea about them

1.8 & 2.2 Versions

Very familiar with 1.8.3

Will try to focus on 2.2

Overall, very logical

Easy to use/query

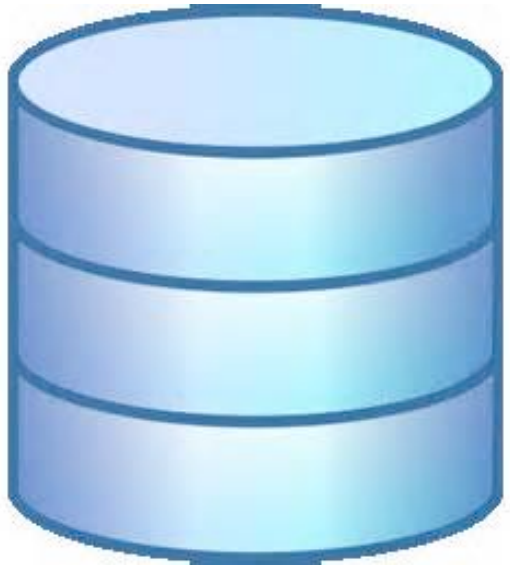
Very powerful

Some weird things

Some complex things

Key values in defines.inc.php

I have a 2.2 ERD Model for you !



Database Overview

~111 Tables

Well-structured & logical

All about foreign keys for joins

All useful joins on ID

objectid, e.g. hostid, itemid

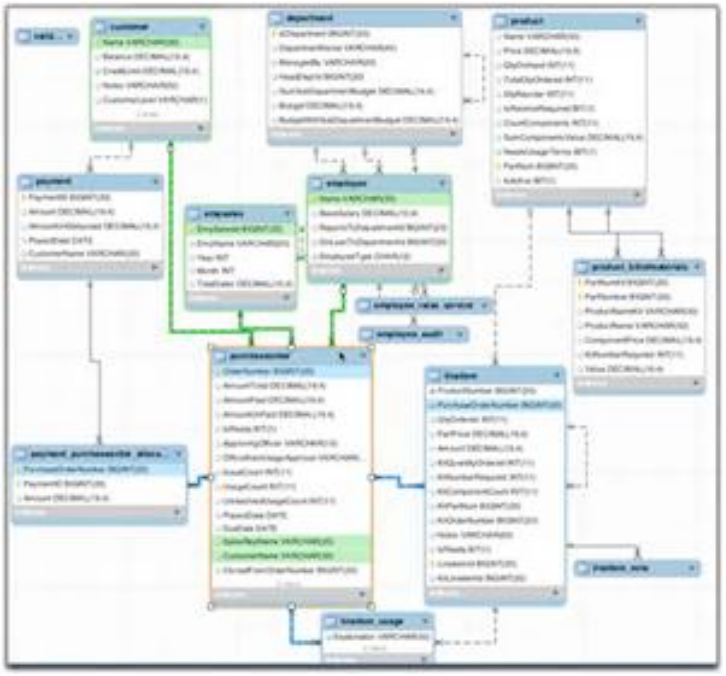
All timestamps are epoch time

Always called 'clock' in any table

Use MySQL unix_timestamp() and from_unixtime()

Several billion-row tables

Careful with these !



Key Subsystems

Hosts (& Templates)

Items (& Templates)

Functions

Triggers

Events

Users

Trends & History

Macros, Profiles, HostGroups

Graphs, Screens

Queue

Core Subsystems

Need to know/understand in detail

Hosts (& Templates)

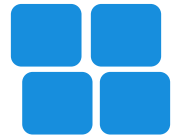
Items (& Templates)

Triggers (& Functions)

Events

Trends & History

Hosts, Items & Templates I



Critical to understand the relationships

Hosts & Items are the core of the system

Hosts have items

Items drive everything

Agent Type, Data Type, Intervals, etc.

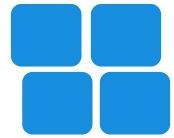
SQL to look at a host & items:

```
SELECT host, key_, description, from_unixtime(lastclock), lastvalue
FROM items i JOIN hosts h ON i.hostid = h.hostid
WHERE h.status = 0
AND i.status = 0
```

HOST	KEY	DESCRIPTION	LASTCLOCK	LASTVALUE
srv-nc-webdav1	agent.ping	agent-ping	2014-08-19 02:50:44	1
srv-nc-webdav1	agent.version	Version of zabbix_agent(d)	2014-08-19 02:25:13	1.8.3
srv-nc-webdav1	apache[busyworkers]	Apache Busy Workers	2014-08-19 02:50:16	1.000000

items.lastvalue removed in 2.2 – Pull from history table

Hosts, Items & Templates II



Templates are just special hosts (status=3)

Templated items are special items

Templateid=0 and hosts.status=3

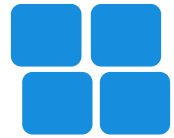
A host's non-template items also templateid=0 so careful

On a host, an item is from a template if templateid>0

Templateid is for an ITEM, not the template

Must join (host's item to template's item to template) to get template name

Hosts, Items & Templates III



Hosts have attached Templates

One or more per host

Host's templated items are copied from Template

Template Items are COPIED to the host
VERY important to understand this relationship
Important to understand what can be changed at host level

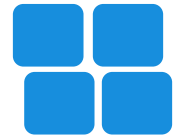
So 10 hosts with a template of 10 items

110 items total in the system (10 + 10x10)

HOST	HOST_STATUS	ITEMID	TEMPLATEID	KEY_
NC_Template_Linux	3	24130	0	agent.version
srv-nc-webdav1	0	23110	24130	agent.version
srv-nc-dns1	0	23314	24130	agent.version

```
SELECT host, h.status, itemid, templateid, key_
FROM items i JOIN hosts h ON i.hostid = h.hostid
WHERE (h.status = 0 OR h.status = 3)
AND i.status = 0
AND (i.itemid = 24130 OR i.templateid = 24130)
ORDER by templateid
```

Triggers, Events, Functions, and Items I



Another key set of relationships

Events are Trigger status changes

- Basically the alerts you see on dashboard
- Drive actions, emails, dashboard

Triggers are logic that finds problems

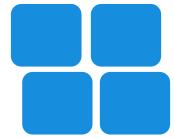
Contain the logic Expression

- String with fomula

Based on Functions

- Functions are the Zabbix functions

Triggers, Events, Functions, and Items II



Functions contain items and the function

Last, avg, etc.

Items link to hosts, etc.

Triggers can be multi-host

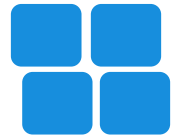
This complicates logic

Hard to link Trigger to a Host – big SQL

Example:

TRIGGERID	EXPRESSION	DESCRIPTION
10048	{1003079}/{1003080}*100<10	Lack of free memory on server {HOSTNAME}
10056	{1003227}>300	Too many processes on {HOSTNAME}

FUNCTIONID	ITEMID	TRIGGERID	FUNCTION	PARAMETER
1003079	10090	10048	last	0



Now that you understand all that

We'll talk about some tables

Hosts

Core table

Hosts are Hosts

Templates are also Hosts

hosts.status = 3

Proxies are also Hosts

Hosts.status = 5

Don’t confuse with agent on proxy host

Hosts are Enabled/Disabled

hosts.status = 0 or 1

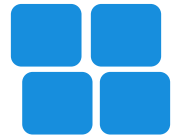
Hosts can be in unreachable state

hosts.status = 2

Not clear this is fully used

Join items to hosts to get host/template name

Hosts – SQL



List active hosts

```
SELECT hostid, proxy_hostid, host, ip, port, status
FROM hosts WHERE status = 0
ORDER BY host
```

HOSTID	STATUS	HOST	IP	PORT	STATUS
10057	0	srv-nc-web1	60.139.13.43	40067	0
10058	0	srv-nc-web2	223.173.38.47	13050	0
10059	0	srv-nc-web3	223.213.91.96	20450	0

Items

Core table

Linked to hosts on hostid

Have Type

Agent, SNMP, Internal, Simple, IPMI, ec.

Enabled/Disabled, Error

items.status = 0 or 1

Can be in error, items.status = 3

Lastclock tells last collect time in 1.8

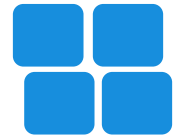
Very useful for pulling data, stats, issues

Not clear where this went in 2.2

LastValue has last value in 1.8

In 2.2 you must get from Trend table, annoying

Items



Either Host or Template Level

If from Template, COPIED from Template

Some fields can be changed at host level
- Enabled, Interval, History/Trend Retention, Application, Group

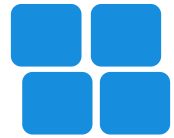
But OVERWRITTEN if you update the Template

templateid = itemid on the Template

```
SELECT host, h.status, itemid, templateid, key_
FROM items i JOIN hosts h ON i.hostid = h.hostid
WHERE h.status = 0
AND i.status = 0
ORDER by host, key_
```

HOST	STATUS	ITEMID	TEMPLATEID	KEY
srv-nc-def1	0	227843	22934	agent.ping
srv-nc-def1	0	216864	44130	agent.version
srv-nc-def1	0	216864	0	local.thing

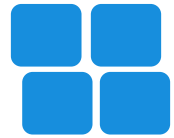
Items – Get Data



Get data from items (Ver 1.8)

```
/* Get small swap servers with swap used */
SELECT host, i.lastvalue AS Swap_Size, 100-ii.lastvalue as Swap_Used
FROM items i JOIN hosts h ON i.hostid = h.hostid
JOIN items ii on h.hostid = ii.hostid
WHERE i.templateid = 24172 /* Swap size */
AND i.lastvalue > (1 * 1024 * 1024 * 1024)
AND i.lastvalue < (88 * 1024 * 1024 * 1024)
AND h.status = 0
AND i.status = 0
AND ii.status = 0
AND ii.templateid = 154766 /* swap % free */
AND (100-ii.lastvalue) > 10
ORDER BY Swap_Used DESC
```

Items – Get Data



items.lastvalue & lastclock removed in 2.2 – Pull from history

VERY long query – about 5 pages (many parts removed):

```
SELECT (case when (i.value_type = 0)
          then (select  history.value from history
                where  (history.itemid = i.itemid)
                order by history.clock desc limit 1)
          when
             (i.value_type = 1) …
             (i.value_type = 2) …
             (i.value_type = 3) …
             (i.value_type = 4) …
end) AS lastvalue,
       (case  when (i.value_type = 0)
          then (select history.clock from  history
                where (history.itemid = i.itemid)
                order by history.clock desc limit 1)
          when
             (i.value_type = 1)
             (i.value_type = 2)
             (i.value_type = 3)
             (i.value_type = 4)
end) AS lastclock
    from  items i where itemid = 23110
```

Zabbix Queue

Items can be in a 'queue'

Seen on Admin|Queue screen

Not a real queue !

Just a list of late items

May change in Version 2.2

Now() > (lastclock + interval)

Max 'queue' is # of active items

Stuff can get stuck if error & host disabled

Hosts disabled but items enabled

Lots of good SQL for reports

Queue size, oldest items, queue by host

Queue by proxy for graphs / triggers

Triggers I

Core table

Linked to functions

Functions link to items
Items link to hosts

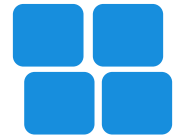
Functions

Last, min, max, sum, nodata, etc.

Enabled/Disabled, Error

triggers.status = 0 or 1

Triggers II



Value – OK or Problem

- Also UNKNOWN in 1.8
- Behavior changed in 2.0/2.2

Priority is here

URL is here

Templateid tells you if came from template

- ID is of parent Trigger in the template

Dependencies are here, complicated

- Trigger_up is trigger we depend on
- Trigger_down is dependent trigger (this trigger?)
- Trigger Level – 0 for no dependency

Events

The Alerts you see on Dashboard

Basically a trigger changing status

Also include auto-discovery, etc.

Triggered events

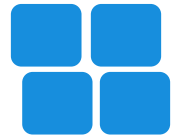
- Source = 0, Object = 0
- Objectid will match the trigger
- Status tells you Trigger status

We tie Alert Tickets to this

Note: Server re-creates events on restart

- Copies over ACKs
- Very annoying if you tie things to eventid
- We have special PHP to rebuild this relationship

Events – ACK & Duration



ACKs set flag in Event DB row

And ACK data in acknowledges table

Finding Event duration is HARD

- Basically scan forward for next OK event
- Slow and messy
- Important for metrics

```
select distinct SUBSTRING_INDEX(from_unixtime(e.clock),' ',1) AlertDate,
    SUBSTRING_INDEX(from_unixtime(e.clock),' ',-1) AlertTime, (select
    floor((eb.clock-e.clock)/60) from events eb where value = 0 and eb.eventid >
    e.eventid and eb.objectid=e.objectid order by eb.eventid limit 1 ) as Duration,
    h.host, t.description, t.priority, from_unixtime(a.clock) ACK, u.name,
    a.acknowledgeid, a.message, floor((a.clock-e.clock)/60) response_time from
    triggers t join functions f on f.triggerid=t.triggerid right join items i on
    f.itemid=i.itemid join hosts h on h.hostid=i.hostid right join events e on
    t.triggerid=e.objectid left join acknowledges a on a.eventid=e.eventid left join
    users u on u.userid=a.userid where e.value=1 and t.triggerid<>19072 and
    e.clock>unix_timestamp('2010-04-01 09:00:00') and
    e.clock<unix_timestamp('2010-04-01 18:00:00') order by e.clock;
```

History & Trends

This is the data we collect

Drives the graphs

Data first goes to History tables

By type – uint, text, double, etc.

Server moves to Trends tables

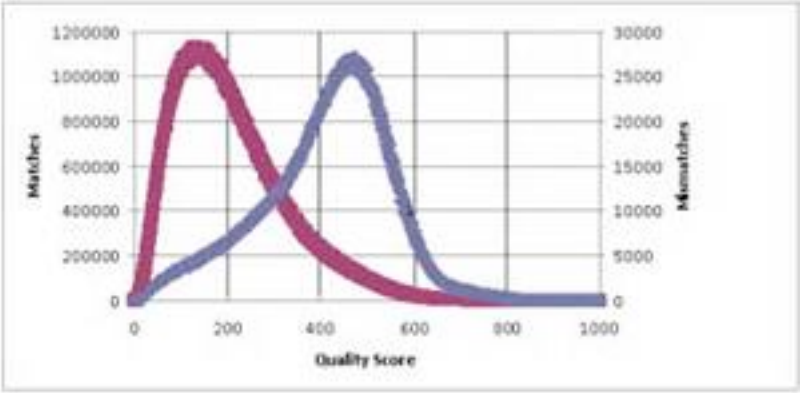
On schedule based on Item config
Summarizes each hour
Saves min, max, average

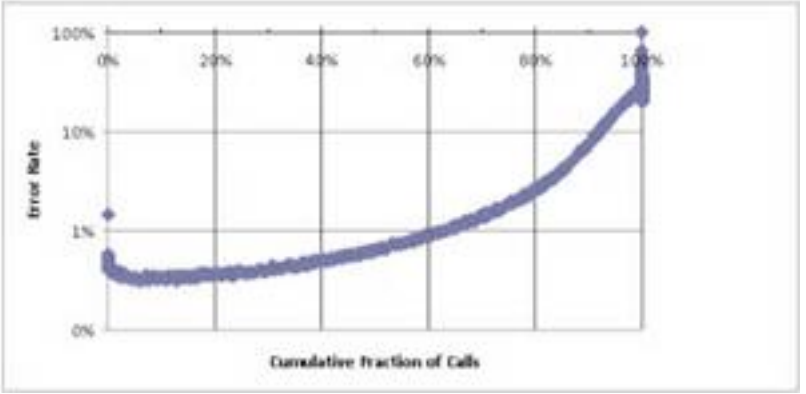
Trends table purged by Housekeeper

Can partition in DB for much faster purge
Special SQL can also purge, but I/O heavy

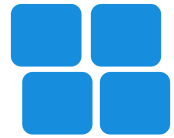
SQL in History/Trends painful

Lots of random I/O
Ideally data fits in server RAM or have SSD





Web Checks



A bit complex – 4 tables

Link into the rest of the system

Httptest

Test name, status, and interval, last check, and what seems like response time, and error text

httpstep tables

Step name, URL, timeout, response code, required text

Httptestitem

Links to items, creates two items of type 9 for each test - Download Speed & Failed Step (type 2 & 3 in this table)

httpstepitem

Links to items, creates 3 items of type 9 for each step - Download Speed, Response Time, Response Code

Other

Note Item History & Trends set to 30 & 90, but can't seem to be edited anywhere

Graphs, Screens & Slideshows

Pretty part of the system

Basic tables

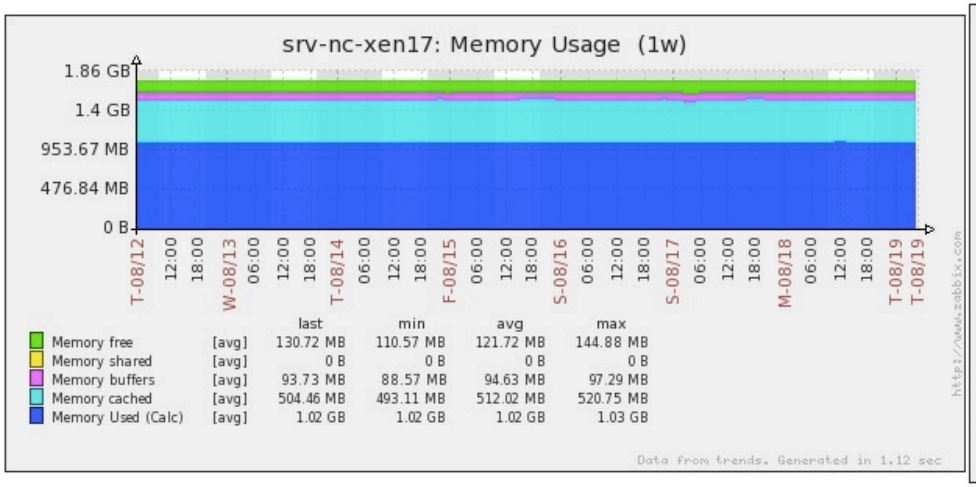
screens & screen_items

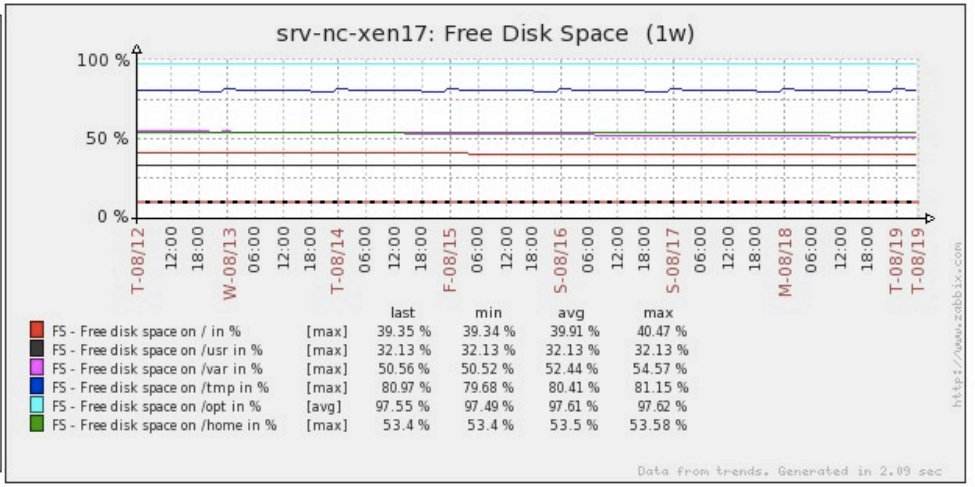
Use resourcetype to know how to link

Each type can link to underlying graph, item, etc.

Complex security

Based on item/host permissions





Users & Security - I

users table

alias field is the actual user name

usrgrp table

All roles/permissions tied here, API, GUI, etc.

rights table

Links usrgrp to server groups with RO, R/W permission

profiles table

Use not clear as thousands of rows per user

Think its drives dashboard and other modules settings

sessions table

Basic user session tied to cookie

```
mysql> update users set passwd=MD5('somepassword') where alias='Admin';
```

Users & Security - II

Enable/Disable

Seems to disable by adding to disabled group

Passwords

MD5 has WITH TRAILING CARRIAGE RETURN, so use -n on echo:

```
$ echo -n "somepassword" | md5sum 9c42a1346e333a770904b2a2b37fa7d3
```

But easier to use MySQL function:

```
mysql> update users set passwd=MD5('somepassword') where alias='Admin';
```

Refresh field – For all screens

Defaults very low, set to 0 or much higher

Otherwise heavy load on DB

Users & Security

Hosts a User has rights on

```
SELECT ug.usrgrpid, ug.name AS user_group, g.name as host_group, host FROM
    users u JOIN users_groups ugl ON u.userid = ugl.userid JOIN usrgrp ug ON
    ugl.usrgrpid = ug.usrgrpid JOIN rights r ON ug.usrgrpid = r.groupid JOIN groups g
    ON r.id = g.groupid JOIN hosts_groups hg ON g.groupid = hg.groupid JOIN hosts
    h on hg.hostid = h.hostid where u.alias = 'steve.mushero' /* userid 116 */ /*
    group 15 is 24x7 user */ AND r.permission in (2,3);
```

Config and User Profiles

Refresh rates and DB load

Dashboard refresh rates

```
UPDATE profiles p JOIN users u ON p.userid = u.userid
SET p.value_int = 300
WHERE idx LIKE 'web.dahsboard.rf_rate.%'
AND u.alias not LIKE 'cust%';
```

Audit System

Great, but useless GUI

Data very fine-grained / detailed

audit_log & audit_log_details

Actions & Resources

See code for list of values

General SQL to get info:

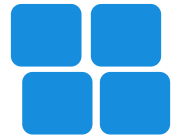
SELECT alias, from_unixtime(a.clock), a.action, a.resourcetype, a.details, a.resourceid, a.resourcename, ad.table_name, ad.field_name, CAST(ad.oldvalue AS UNSIGNED) AS oldvalue, CAST(ad.newvalue AS UNSIGNED) AS newvalue

FROM auditlog a LEFT JOIN auditlog_details ad ON a.auditid = ad.auditid JOIN users u ON a.userid = u.userid

WHERE resourceid = 109482 AND field_name = 'status'

AND from_unixtime(clock) > **'2013-05-01'**

Audit System



Big select to get details on hosts

```
SELECT alias, from_unixtime(a.clock), CASE a.action WHEN 0 THEN "Added"
WHEN 1 THEN "Updated"
WHEN 2 THEN "Deleted" ELSE CAST(a.action AS CHAR) END AS action, CASE
    a.resourcetype
WHEN 4 THEN "Host"
WHEN 13 THEN "Item ?"
WHEN 15 THEN "Item"
ELSE CAST(a.resourcetype AS CHAR) END AS resource_type, a.details, a.resourceid,
    a.resourcename, ad.table_name, ad.field_name, CAST(ad.oldvalue AS CHAR),
    CASE ad.newvalue
WHEN 0 THEN "Enable”
WHEN 1 THEN "Disable" END as newvalue
FROM auditlog a LEFT JOIN auditlog_details ad ON a.auditid = ad.auditid JOIN users
    u ON a.userid = u.userid
WHERE from_unixtime(clock) > '2012-01-01' /* AND alias LIKE 'matt%' */
AND a.resourcetype = 4 /* Host */ AND (field_name = 'status' OR field_name IS
    NULL) /* AND ad.newvalue = 1 /* 1 = Disable */ /*
AND resourcename LIKE '%web17%'*/;
```

Safety Reports

We have dozens of these, such as:

- Items that differ from template
- Missing templates
- Disable items/hosts, forget to enable
- Alerts with no URL/Wiki
- Hosts missing profile data
- Items disabled conflict with trigger
- Web alerts with no trigger
- Web alerts with long/short timeouts
- Hosts in wrong, duplicate, conflicting groups
- Servers in Zabbix, not core system

Many are quite complex, big SQL

We will post these on-line

- After updating for Ver 2.2

Housekeeper

Done by server every hour

Very slow – Item by Item

Thousands per second

A LOT of I/O (mostly read)

We have SQL to do in bulk

But too heavy load on I/O system
We'll see on SSD, or all data in RAM

Backups

Backup, of course, but big

Ours are 8 hours

200GB of data

At 1TB this is not manageable

Need incrementals

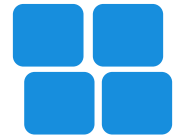
Maybe recent data only

Do from slave

You can backup config only

Ignore history*, trends*, events, audit*, acknowledges

Summary



We love the Zabbix database

So should you

Learn how everything connects

Focus on key / core tables

Have fun

Thanks from ChinaNetCloud

Pioneers in OaaS – Operations as a Service

Shanghai Headquarters:

X2 Space 1-601, 1238 Xietu Lu

Shanghai, 200032 China

T: +86-21-6422-1946 F: +86-21-6422-4911

Beijing Office:

Lee World Business Building #305

57 Happiness Village Road, Chaoyang District

Beijing, 100027 China

Silicon Valley Office:

California Avenue

Palo Alto, 94123 USA