ZABBIX

The Enterprise class Monitoring Solution for Everyone

WWW.ZABBIX.COM

Zabbix 3.0 Training Certified Specialist

Day 1

GETTING STARTED

Facilities

Introduction

- ✓ Background/company
- ✓ Experience with CLI Unix like systems
- ✓ Experience with Zabbix
- ✓ Experience with other monitoring solutions
- ✓ Current Zabbix deployments

Questions at any moment encouraged

DAILY SCHEDULE

Monday	Tuesday	Wednesday
10.00-11.30 Zabbix 3.0 Certified Specialist	**09.00-11.30 Zabbix 3.0 Certified Specialist**	**09.00-11.30 Zabbix 3.0 Certified Specialist**
11.30-11.45 Break	11.30-11.45 Break	11.30-11.45 Break
11.45-13.00 Zabbix 3.0 Certified Specialist	**11.45-13.00 Zabbix 3.0 Certified Specialist**	**11.45-14.00 Zabbix 3.0 Certified Specialist**
13.00-14.00 Break	13.00-14.00 Break	14.00-15.00 Break and Q/A session
14.00-15.30 Zabbix 3.0 Certified Specialist	**14.00-15.30 Zabbix 3.0 Certified Specialist**	**15.00-16.00 Advanced Topics**
15.30-15.45 Break	15.30-15.45 Break	**16.00-17.50 Certification and presentation of certificates**
15.45-17.50 Zabbix 3.0 Certified Specialist	**15.45-17.50 Zabbix 3.0 Certified Specialist**	

AGENDA

About Zabbix

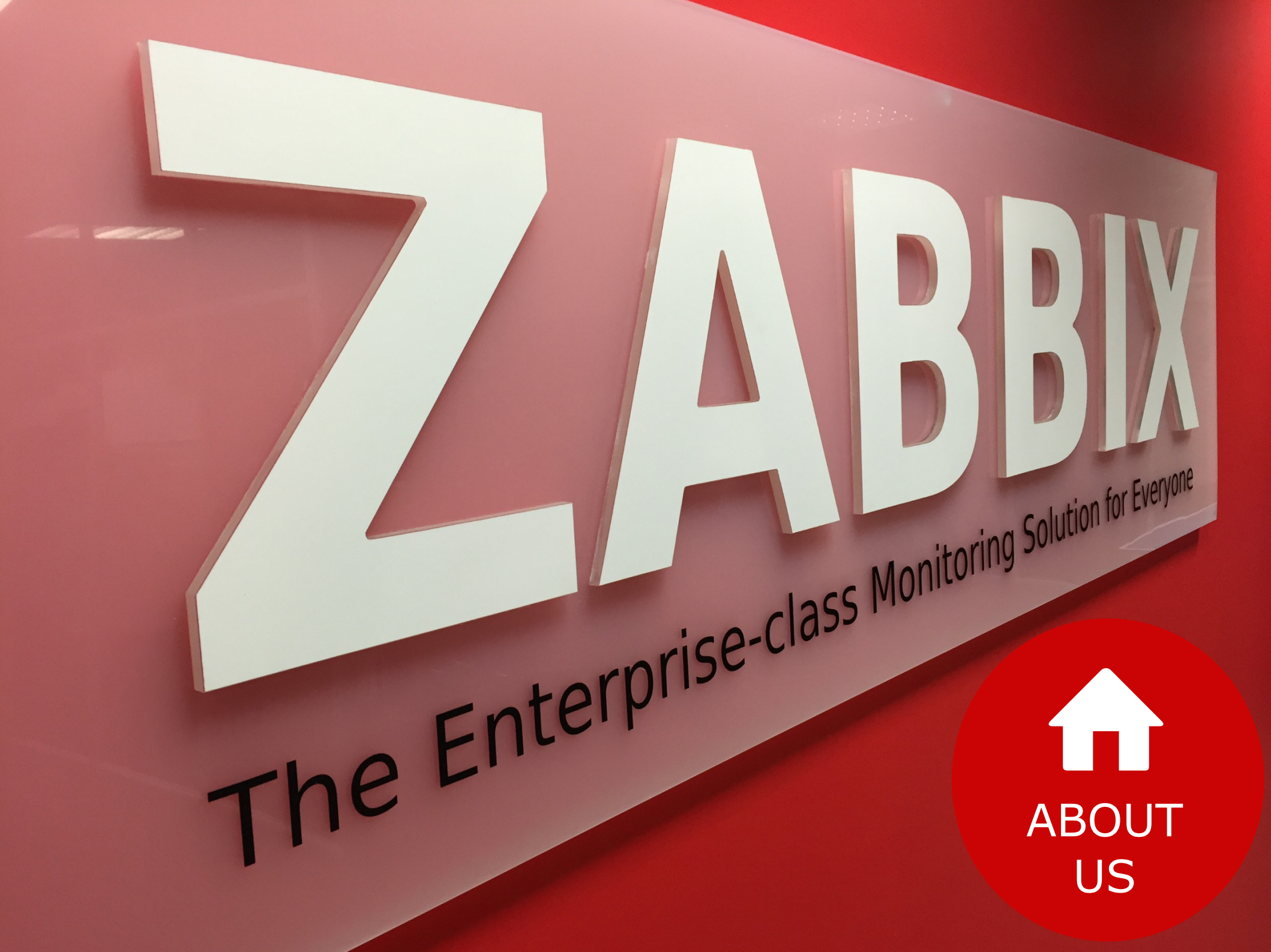
Architecture

Installation

Data collection

Items

Problem detection


ZABBIX
The Enterprise-class Monitoring Solution for Everyone
ABOUT US

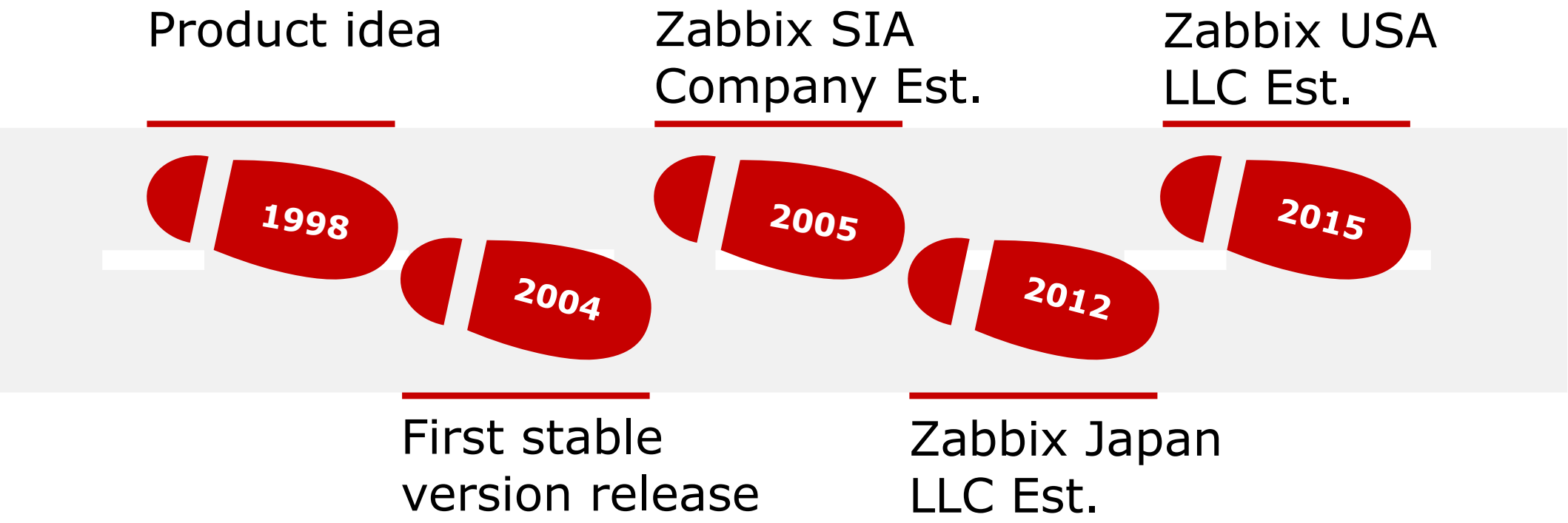
Zabbix is an enterprise level Open Source monitoring software

Product facts

- Supports virtually all platforms and methods of monitoring
- Scaling to Large Environments
- Distributed monitoring
- Pro active monitoring

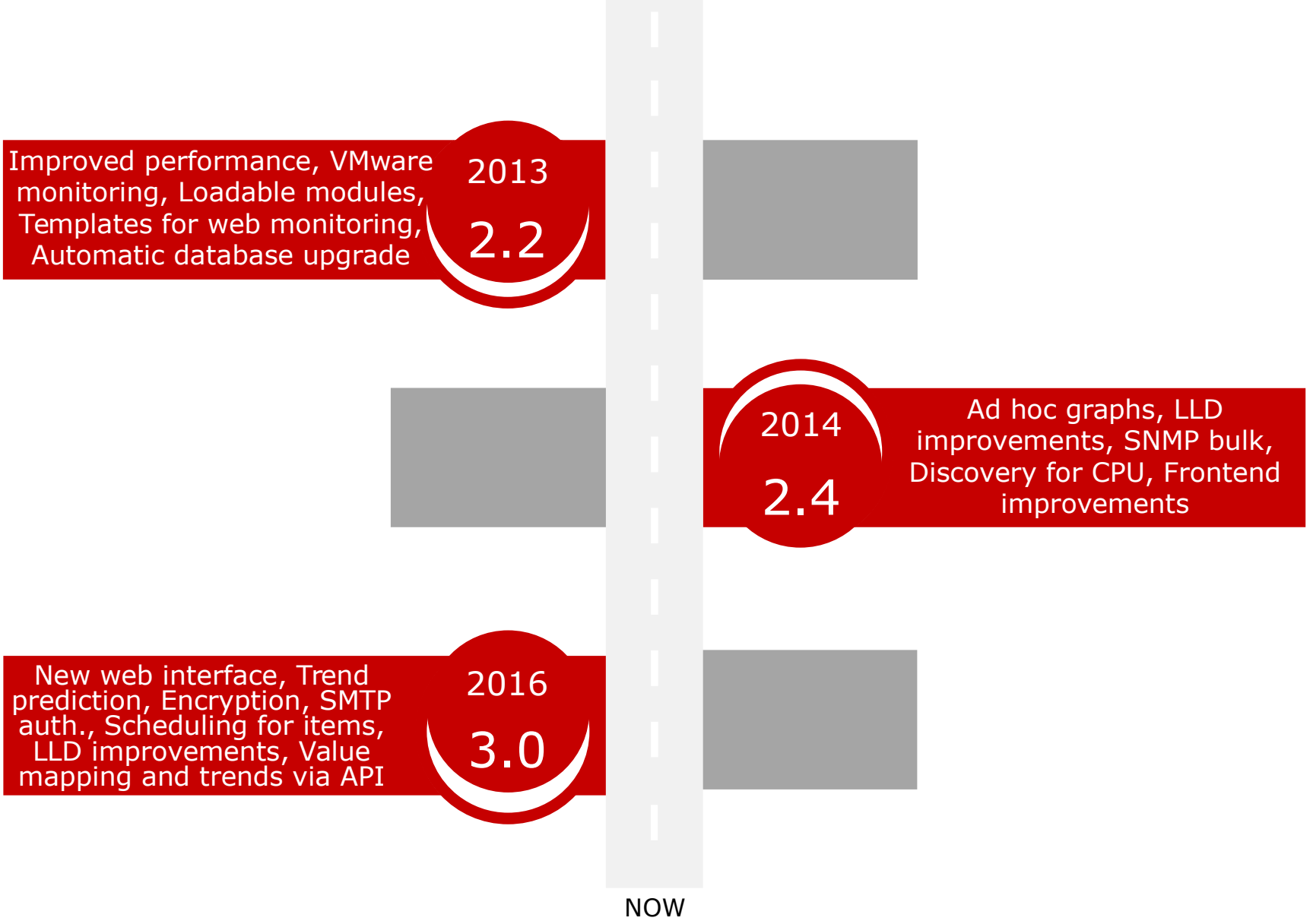
True Open Source,
no proprietary addons,
"professional" or "enterprise" versions

HISTORY OF ZABBIX



VERSIONS TIMELINE

1.0alpha1 (2001)
1.0 (2004)
1.1 (2006)
1.4 (2007)


Improved performance, VMware monitoring, Loadable modules, Templates for web monitoring, Automatic database upgrade
2013
2.2
2014
2.4
Ad hoc graphs, LLD improvements, SNMP bulk, Discovery for CPU, Frontend improvements
New web interface, Trend prediction, Encryption, SMTP auth., Scheduling for items, LLD improvements, Value mapping and trends via API
2016
3.0
NOW

COMPANY FACTS

- Established in 12 April, 2005 in Riga, Latvia.
- Privately held, no investors behind
- Locations

Headquarters: Riga, Latvia
Subsidiaries: Tokyo, Japan
New York, USA

MISSION & AIMS

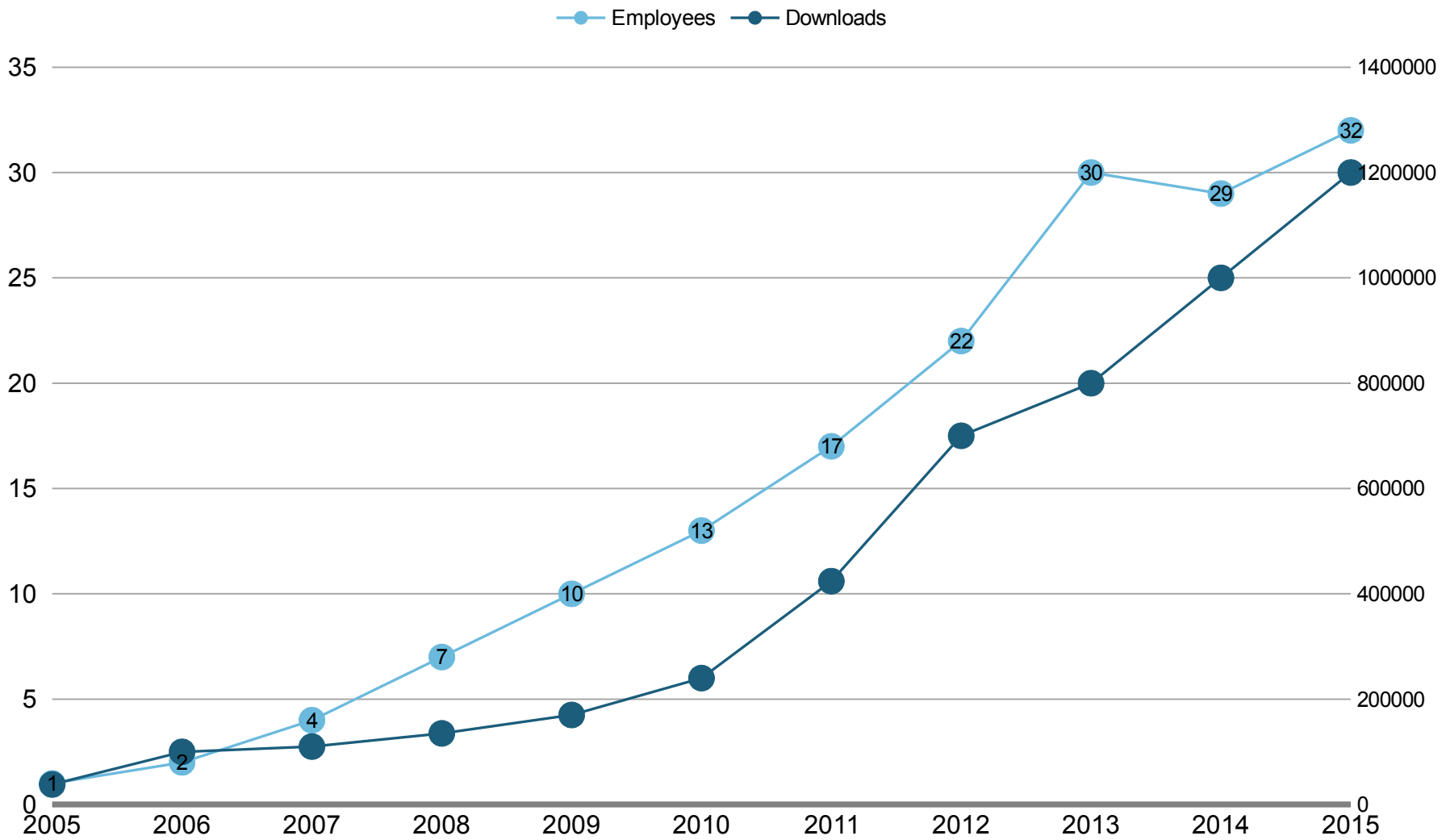
Our Mission

Focus on development and providing commercial services for Zabbix Software

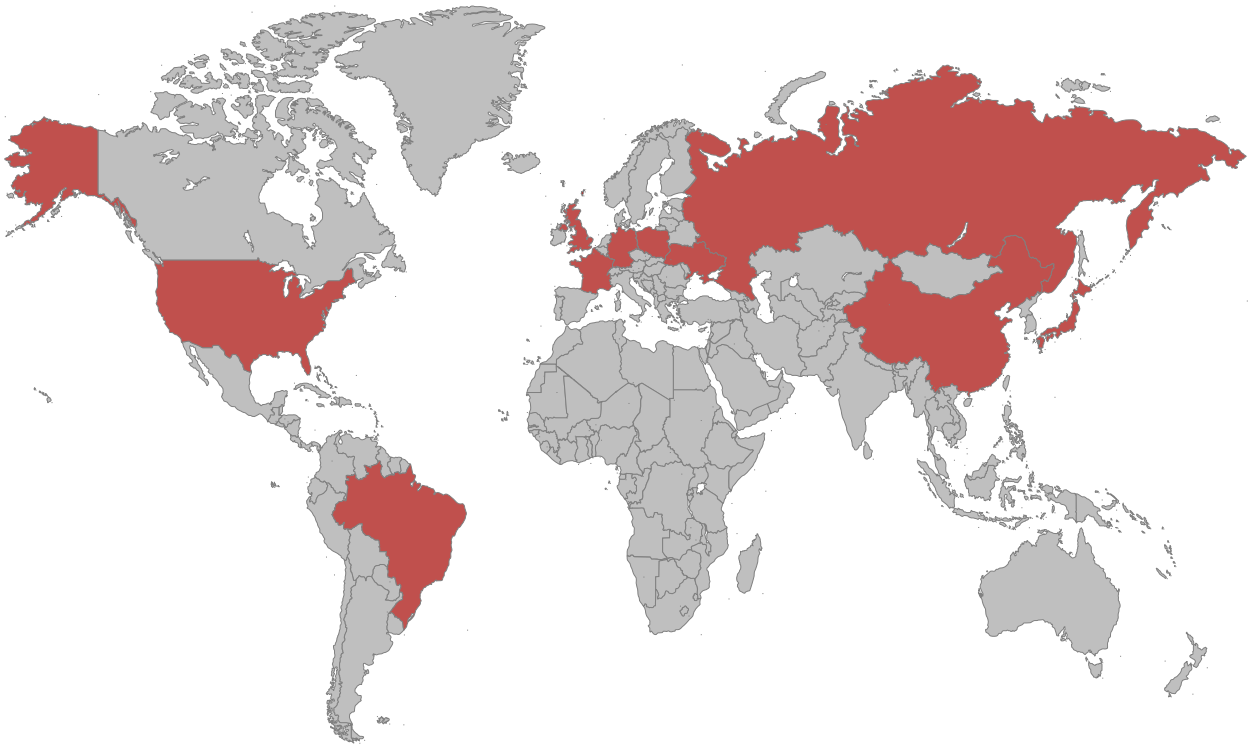
Aims and principles

Develop an exceptional monitoring platform
Listen to our users and our community
Be driven by technology and user needs

HISTORICAL DETAILS



TOP 10 DOWNLOADS 2015



Country	Total
United States	434,971
United Kingdom	193,038
China	68,604
Brazil	61,218
Russia	47,197
Japan	21,100
Germany	18,987
Poland	15,095
France	11,246
Ukraine	6,889

REASONS TO MONITOR

Hard to manage non-transparent environment

Cost of downtime is high

Minimize business impact

Know if your service is reachable

FUNCTIONALITY OF ZABBIX

Data gathering

Gathered using various methods, including SNMP, native agents, IPMI and others

Problem detection and alerting

Gathered data can be compared to thresholds and alerts sent out using different channels like email or SMS

Data storage

Once we have gathered the data it doesn't make sense to throw it away, so we will often want to store it for later analysis

Visualization

Humans are better at distinguishing visualized data, especially when there is huge amounts of data

DIFFERENT APPROACHES

Agent-less monitoring

- ICMP ping
- HTTP, SSH, IMAP, SMTP, other services
- Remote commands using Telnet and SSH

Monitoring with agents

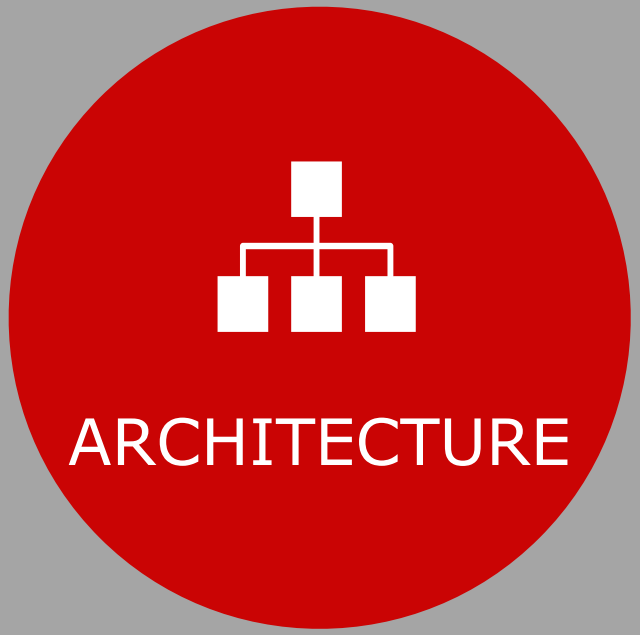
- Passive agents
  - SNMP, Zabbix Agent, IPMI
- Active agents
  - SNMP traps, Zabbix Agent

Centralized monitoring

- All configuration and management is done on one central Zabbix server

Distributed monitoring

- Reduce network load
- Survive link downtime


ARCHITECTURE

WHAT IS ZABBIX?

Zabbix is a software solution for monitoring performance and availability of IT infrastructure (but not limited to that)

- ✓ Network devices
- ✓ OS resources
- ✓ Middleware
- ✓ Applications
- ✓ Services
- ✓ Anything

KEY PRINCIPLES OF DEVELOPMENT

Keep things simple (KISS)

Be efficient: use as few system resources as possible (memory/CPU usage)

Very high performance and high quality product

Low number of third-party dependencies

IMPORTANT DECISIONS

Frontend

- ✓ Open and customisable

Everything is stored in a relational database

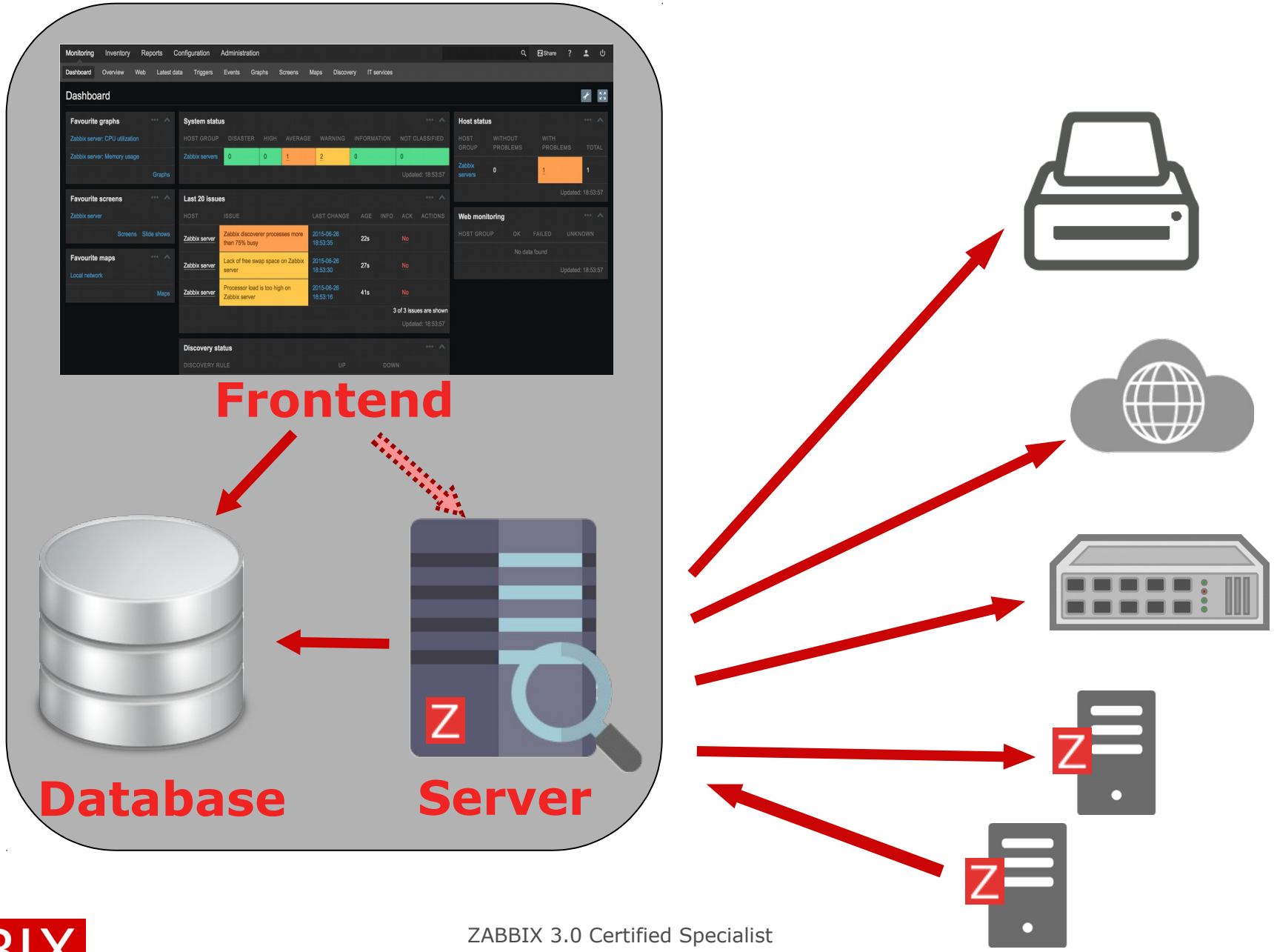
C language for server, proxy and agent

- ✓ The best performance
- ✓ The lowest footprint and resource usage
- ✓ Linux agent uses less than a megabyte of RAM (736K on 64bit; excluding shared libraries)

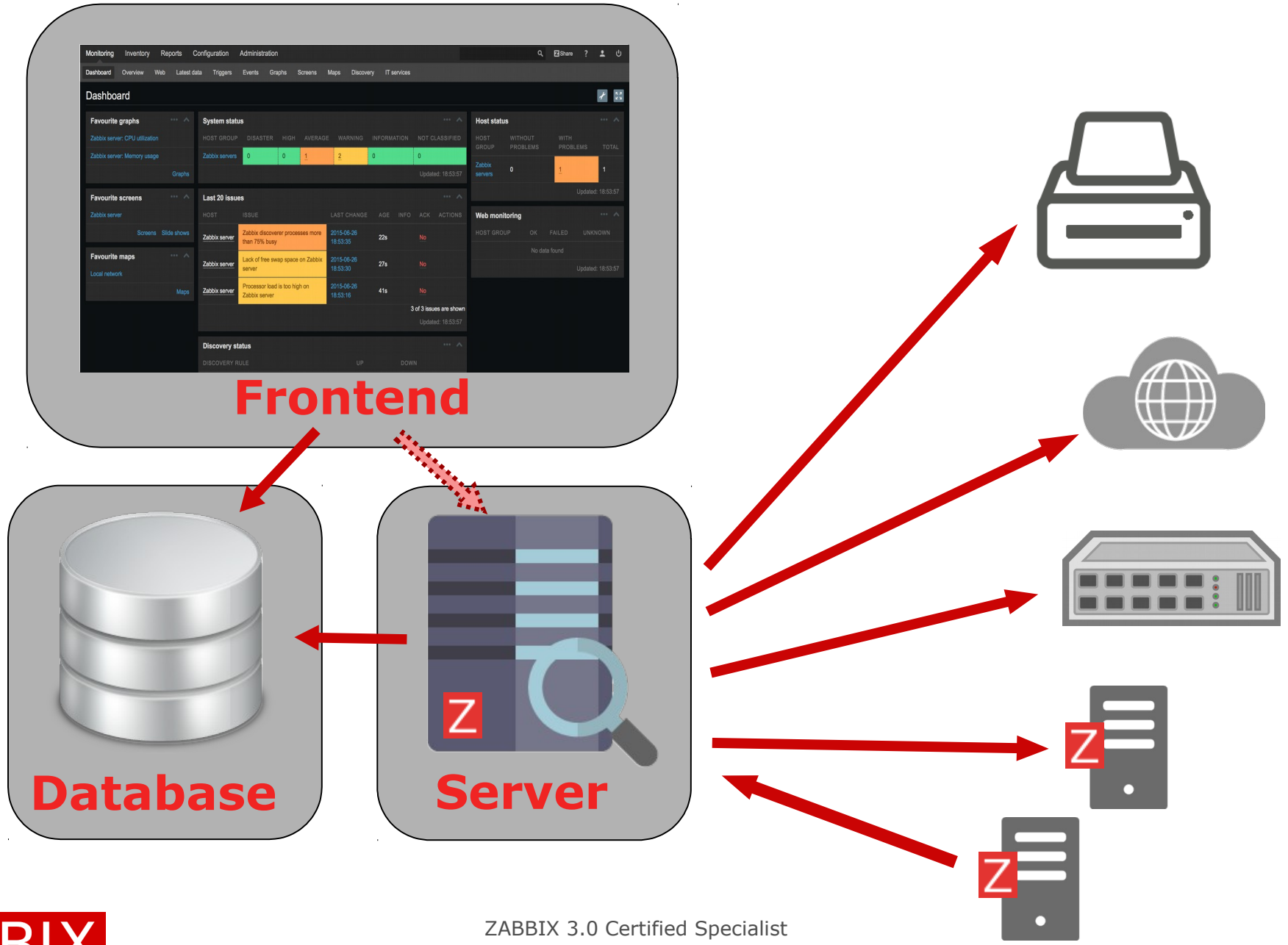
Can be used in embedded environment

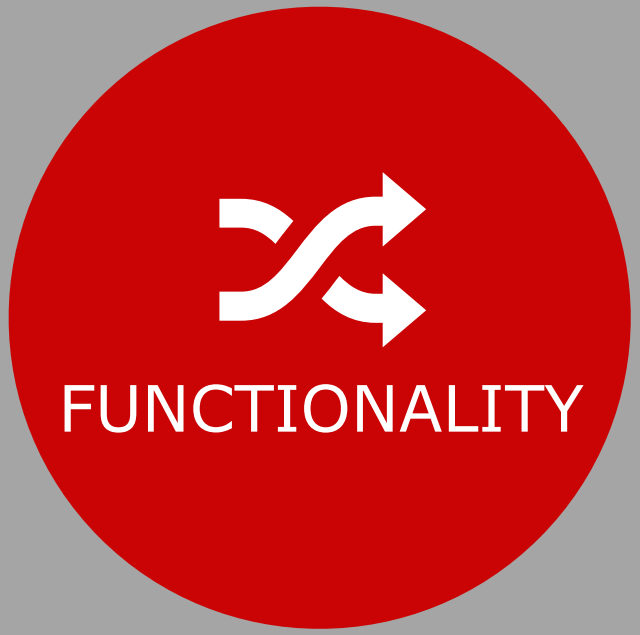
- ✓ SQLite, very small footprint

BASIC ARCHITECTURE



DISTRIBUTED COMPONENTS




FUNCTIONALITY

FUNCTIONALITY 1

Real-time monitoring

- ✓ Performance monitoring
- ✓ Availability monitoring
- ✓ Integrity monitoring
- ✓ Flexible notification conditions
- ✓ Alerts (email, SMS, Jabber)

Trend prediction

- ✓ Future value
- ✓ Time

Visualization

- ✓ Graphing
- ✓ Mapping
- ✓ User-defined views (screens)
- ✓ Filterable dashboard

Alerting (email, SMS, Jabber)

Encryption (certificate, PSK)

IT services / SLA monitoring

- ✓ Hierarchical IT Services
- ✓ Real-time SLA reporting

Flexibility

- ✓ Easily extensible agent
- ✓ Any notification methods
- ✓ Server runs on any Unix platform
- ✓ Zabbix sender

Pro-active monitoring

- ✓ Automatic execution of remote commands
- ✓ Manually executable commands

Aggregate monitoring

- ✓ Monitoring for a group of hosts

FUNCTIONALITY 2

High-performance native agents

Network discovery

Active agent auto-registration

Inventory

- ✓ Automatic collection

Web frontend for configuration (drag'n'drop maps/screens etc)

Authentication

- ✓ Native
- ✓ LDAP based
- ✓ HTTP authentication

Escalations

- ✓ Unlimited number of levels

Zabbix proxy

- ✓ Active/passive

Web monitoring

VMware monitoring

Performance improvement techniques

- ✓ Data buffering on agent side
- ✓ Caches on server side

User group setting

Regular expression builder

IPv6 support

IPMI monitoring

Maintenance

UTF8

API

ODBC monitoring

Java gateway (direct JMX monitoring)


SERVER AND
FRONTEND
INSTALLATION

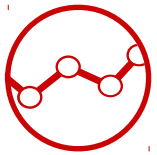
ZABBIX SERVER REQUIREMENTS

Operating system

- Linux
- Solaris
- AIX
- HP-UX
- FreeBSD
- OpenBSD

Database

- MySQL
- MySQL forks
- PostgreSQL
- Oracle
- SQLite
- IBM DB2



Additional libraries

- SNMP: NET-SNMP
- Web: libcurl
- SSH: libssh2
- IPMI: OpenIPMI
- Jabber: lib-iksemel
- VMware: libxml2
- ODBC: unixODBC
- Encryption: OpenSSL

WHAT'S RECOMMENDED

Hardware

- ✓ Multi-core 64bit CPU

Operating system

- ✓ Linux

Database engine

- ✓ MySQL with InnoDB engine

Why MySQL?

- ✓ Open source
- ✓ Most widely used backend

Distribution – what's mostly used

- ✓ RedHat, CentOS, SUSE, Debian, Ubuntu

Distribution and DB choice

- ✓ Use what you are familiar with

INSTALLING SERVER FROM PACKAGES

RHEL/CentOS

Install Zabbix Server

```
# rpm -ivh http://repo.zabbix.com/zabbix/3.0/rhel/\
7/x86_64/zabbix-release-3.0-1.el7.noarch.rpm

# yum install zabbix-server-mysql
```

or

```
# yum install zabbix30-server-mysql
```

OpenSUSE/SLES

```
# zypper install zabbix-server-mysql
```

INSTALLING SERVER: ALTERNATIVE WAY (SOURCES)

Configure and compile

```
# ./configure --help

# ./configure <parameter1> <parameter2> …

# make install
```

Example:

```
# ./configure --enable-server --with-mysql --with-net-snmp ...
# make install
```

CREATING DATABASE

Install MySQL Server

```
# yum install mysql-server
```

Create Zabbix database and user

```
# mysql
mysql> create database zabbix character set utf8 collate utf8_bin;
mysql> grant all privileges on zabbix.* to zabbix@localhost \
identified by 'M35s#ShtCL';
```

Load files

```
# cd /usr/share/doc/zabbix-server-mysql-3.0.0
# zcat create.sql.gz | mysql -uroot -p zabbix
```

FINALISING SERVER INSTALLATION

Configure Zabbix server

```
# vi /etc/zabbix/zabbix_server.conf
DBHost=localhost
DBName=zabbix
DBUser=zabbix
DBPassword=M35s#ShtCL
```

Start Zabbix server

```
# service zabbix-server start
```

or

```
# systemctl start zabbix-server
```

FRONTEND REQUIREMENTS

Component	Requirement
Back-end	Apache, lighthttpd, nginx Any other with support of PHP
Browser	Mozilla Chrome Safari MS Internet Explorer Opera

FRONTEND - PHP REQUIREMENTS

Component	Requirement
PHP version	**5.4**.0 or higher
PHP database support	php-mysql, php-sqlite, php-pgsql, php-sqlora, php-ibm_db2
PHP modules	php-bcmath, php-gd 2.0, php-net-socket, php-mbstring, PNG/JPEG/FreeType support, php-xml, php-gettext, php-ldap
Other requirements	Some distributions might split out PHP core features in packages like php5-ctype, php-session or php5-xml/php5-dom

PHP CONFIGURATION

Component	Requirement
PHP memory limit	128 MB
PHP post max size	16 MB
PHP upload max filesize	2 MB
PHP max execution time	300 seconds
PHP max input time	300 seconds
PHP Timezone	Europe/Riga America/Chicago http://php.net/manual/en/timezones.php

INSTALLING FRONTEND FROM PACKAGES

RHEL/CentOS

```
# yum install zabbix-web-mysql
```

OpenSUSE/SLES

```
# zypper install zabbix-phpfrontend
```

INSTALLING FRONTEND: ALTERNATIVE WAY (SOURCES)

From the source

```
# cp -a frontends/php <htdocs>/zabbix
```

Common <htdocs> locations:

/usr/local/apache2/htdocs

/srv/www/htdocs

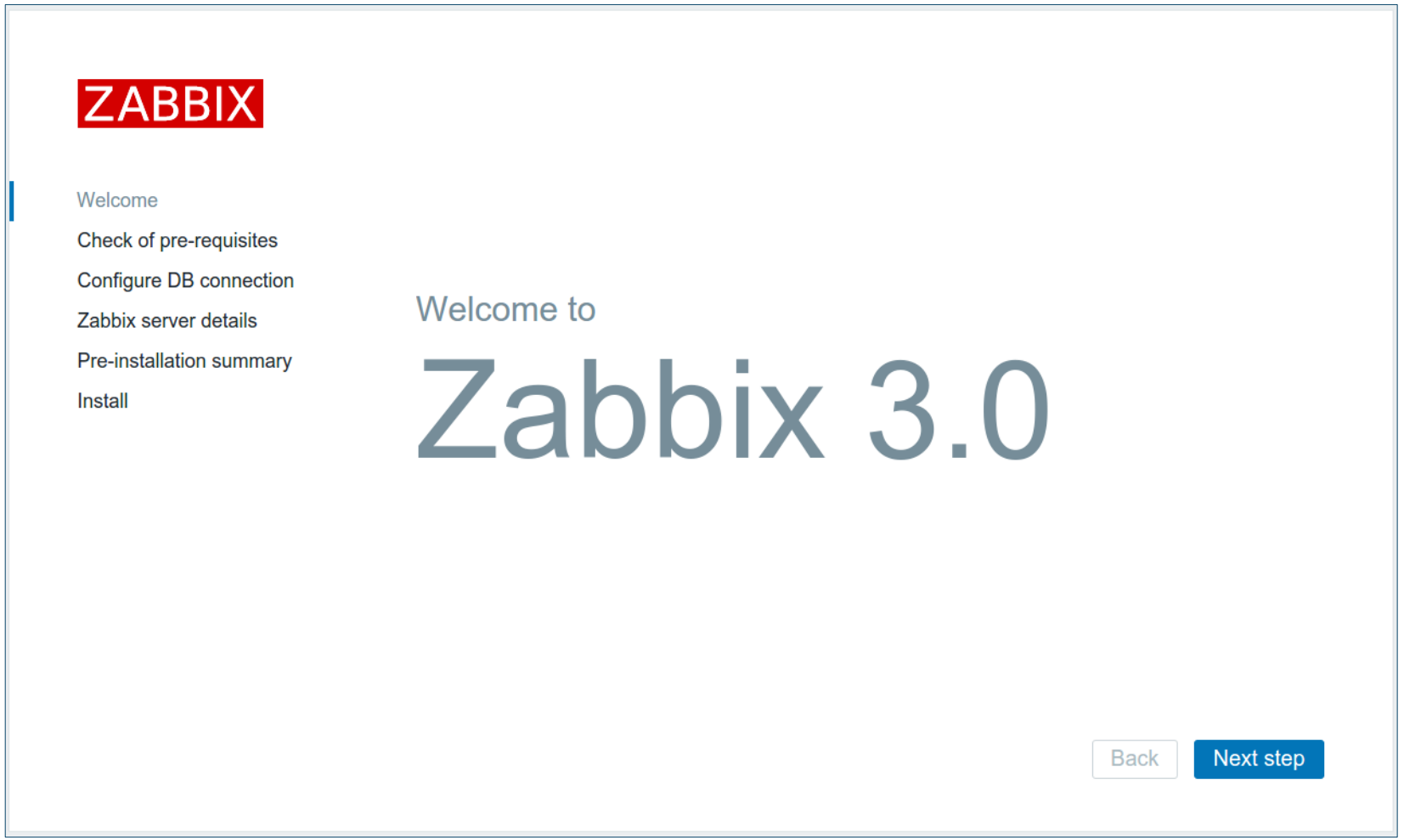
/var/www/html

/var/www

FRONTEND: CONFIGURATION WIZARD

Access frontend with a web browser:

<DNS or IP>/zabbix



FRONTEND CONFIGURATION: ALTERNATIVE WAY

```
# cp conf/zabbix.conf.php.example conf/zabbix.conf.php
```

Edit the new file

```
$DB["TYPE"]     = "MYSQL";
$DB["SERVER"]   = "localhost";
$DB["PORT"]     = "0";
$DB["DATABASE"] = "zabbix";
$DB["USER"]     = "zabbix";
$DB["PASSWORD"] = "M35s#ShtCL";
$ZBX_SERVER     = "localhost";
$ZBX_SERVER_PORT = "10051";
$ZBX_SERVER_NAME = "";
```


AGENT
INSTALLATION

AGENT AVAILABILITY

INSTALLING AGENT

Install Zabbix Agent

```
# yum install zabbix-agent
```

Configure zabbix_agentd.conf

- Server
- ServerActive
- Hostname

Start Zabbix Agent

```
# service zabbix-agent start
```

Often pre-compiled

Configure and compile sources

INSTALLING WINDOWS AGENT

Install as a Windows service

```
cmd> zabbix_agentd.exe --config \
"C:\Program Files (x86)\Zabbix agent\zabbix_agentd.conf" --install
```

Run agent

```
cmd> zabbix_agentd.exe --start
```

Sample config: zabbix_agentd.win.conf

PRACTICAL SETUP

Install MySQL server

Install Zabbix Server, Frontend and Agent

Create Zabbix DB

Configure Zabbix server

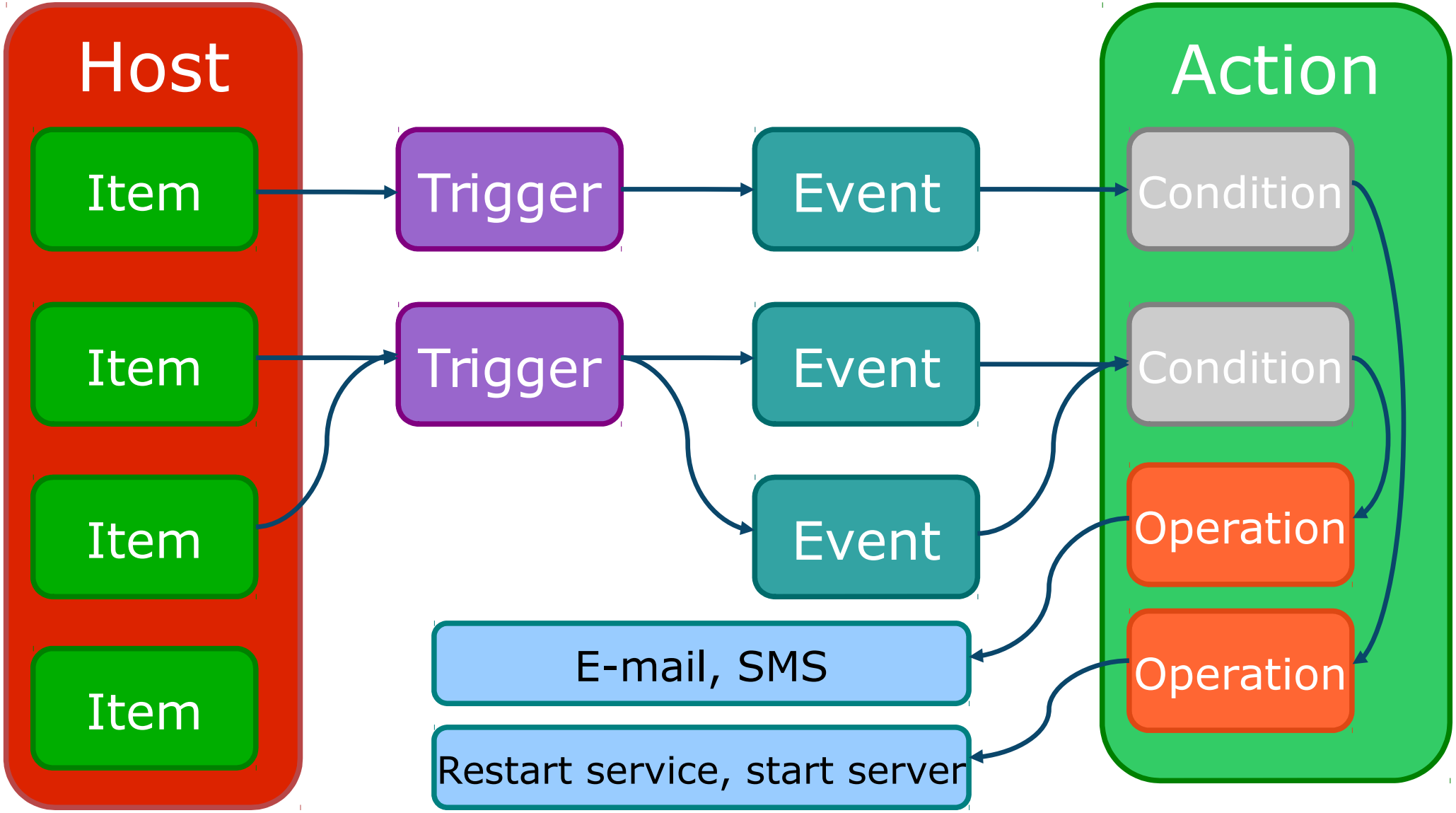
Configure Frontend

Start web server, Zabbix server and agent

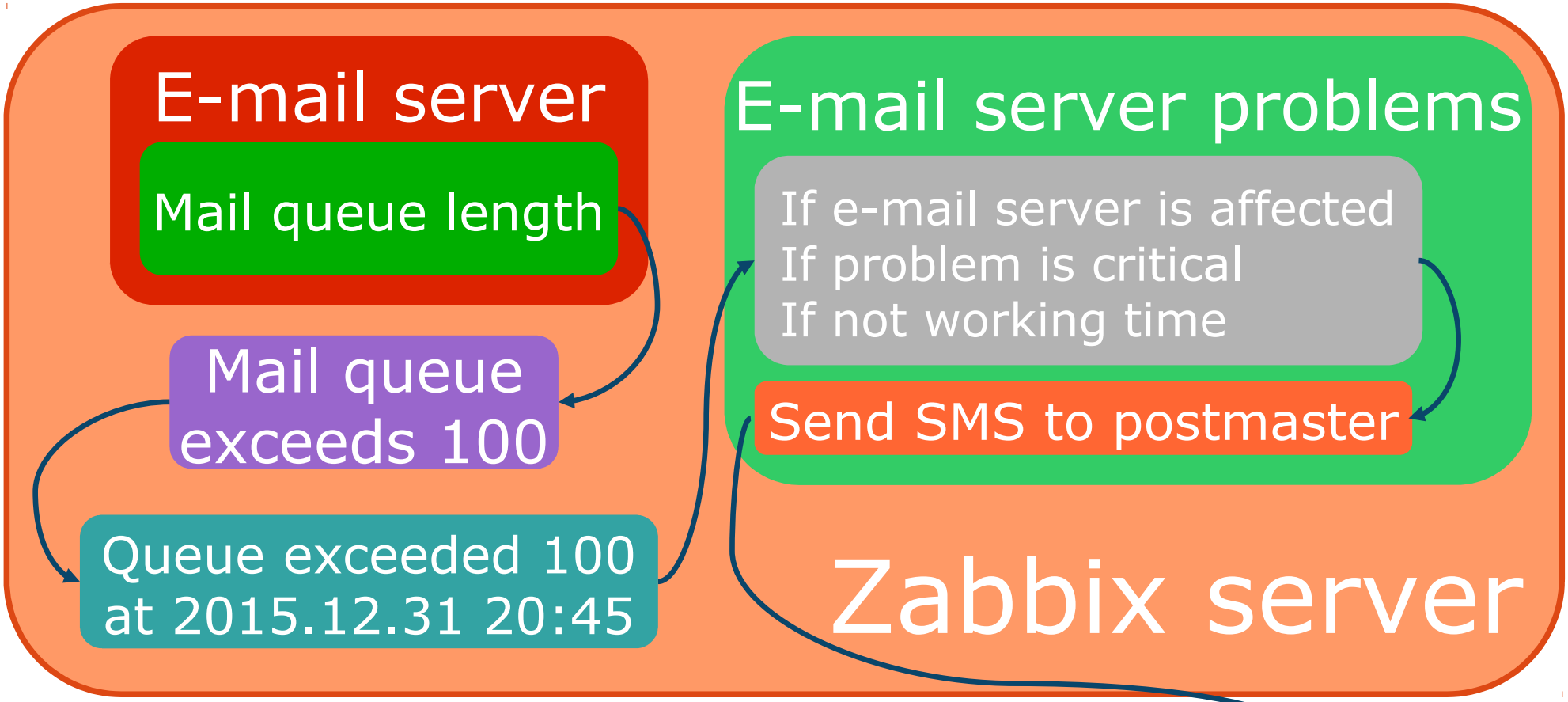
ZABBIX DEFINITIONS

Component	Requirement
Host	Any network attached device having IP or DNS name
Host Group	Logical grouping of hosts
Item	Source of information / metric
Trigger	Logical expression representing problem condition
Template	A set of entities (items, triggers, etc) ready to be applied to one or several hosts
Application	Grouping of items in a logical group
Event	Element state change
Action	A flexible set of conditions Automatically executed set of operations
Operation	Different types: notification, remote command, add/remove host, template linkage

TRYING THEM TOGETHER

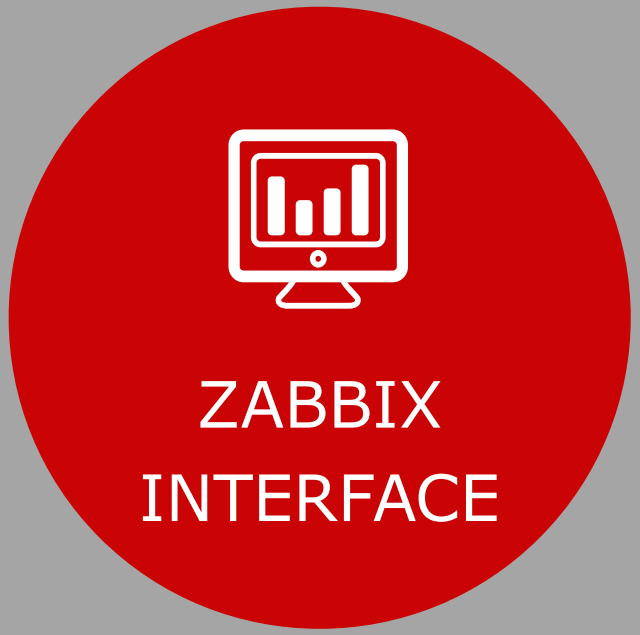


REAL LIFE EXAMPLE



Host | Trigger | Condition | Operation

Item | Event | Action | Actual course taken


ZABBIX
INTERFACE

BASICS

Five menu levels

Monitoring
Inventory
Reports
Configuration
Administration

Permissions depend on user type

Normal user: Monitoring, Inventory and Reports

Zabbix Administrator: Monitoring, Inventory, Reports and Configuration

Zabbix Super Administrator: Monitoring, Inventory, Reports, Configuration and Administration

Global search

List selection and mass updates

Special user: Guest

Unauthorized user, used for login

Full-screen mode for many views (special icon)

Theming, language selection

DASHBOARD

Consists of two parts

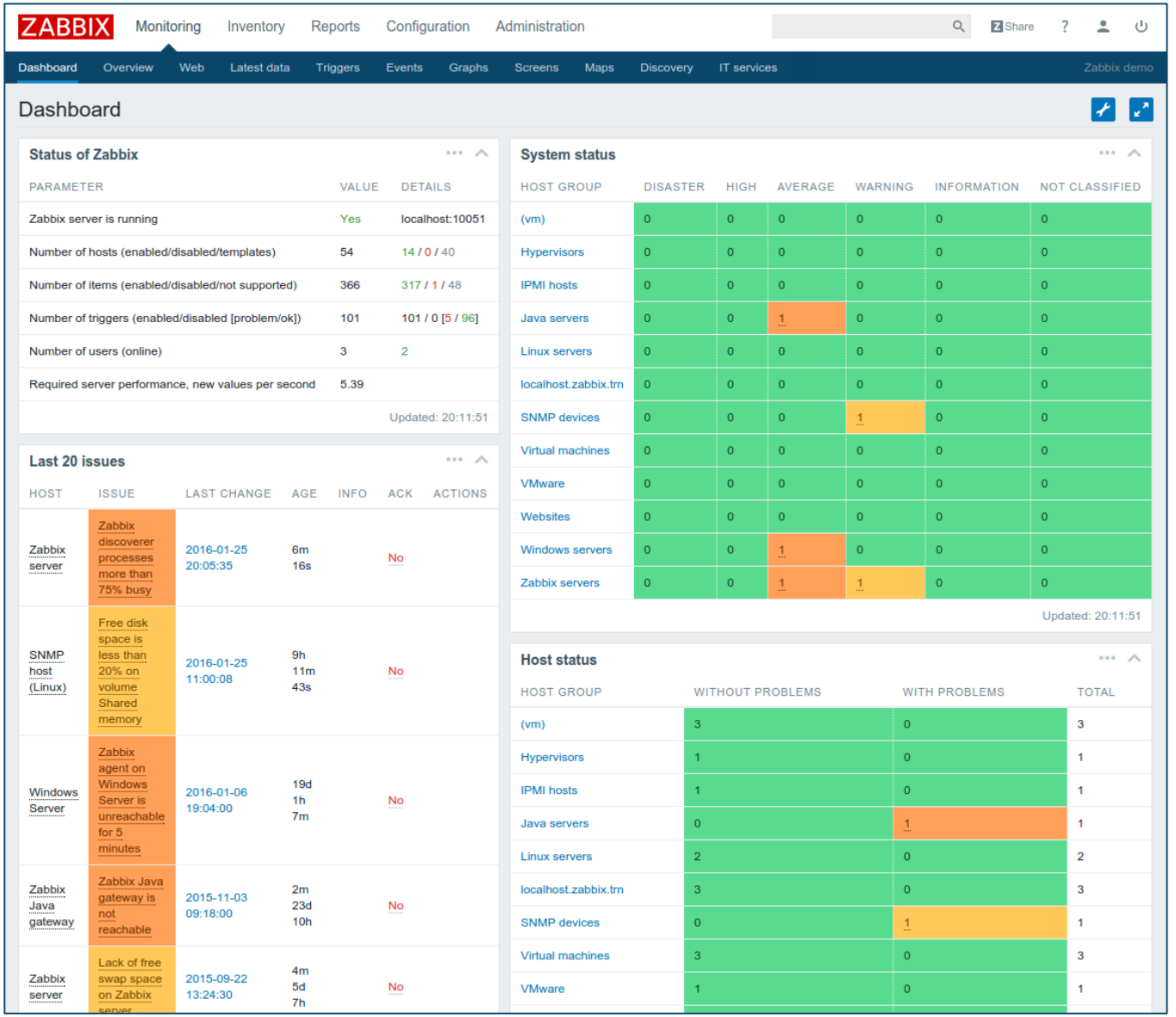
- Favorites
- High level statistics

Rearrange widgets

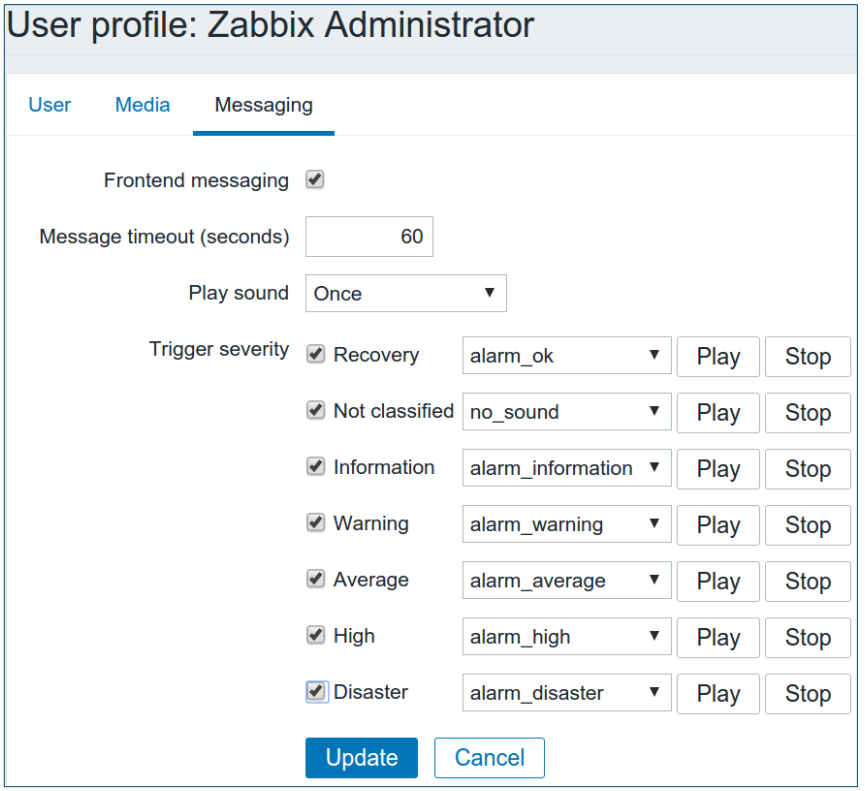
Hide/Show panels

Filter by hostgroup, maintenance, trigger severity and name

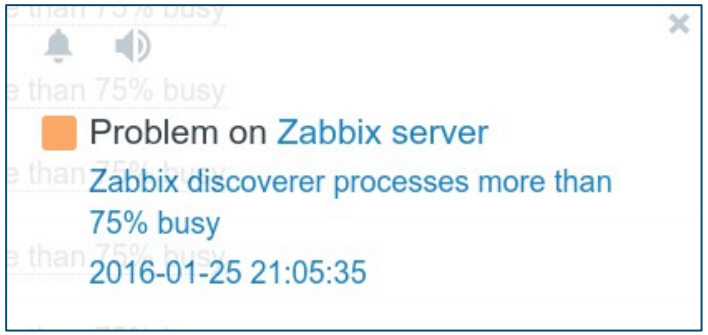
Show Unacknowledged triggers



GLOBAL ALARMS

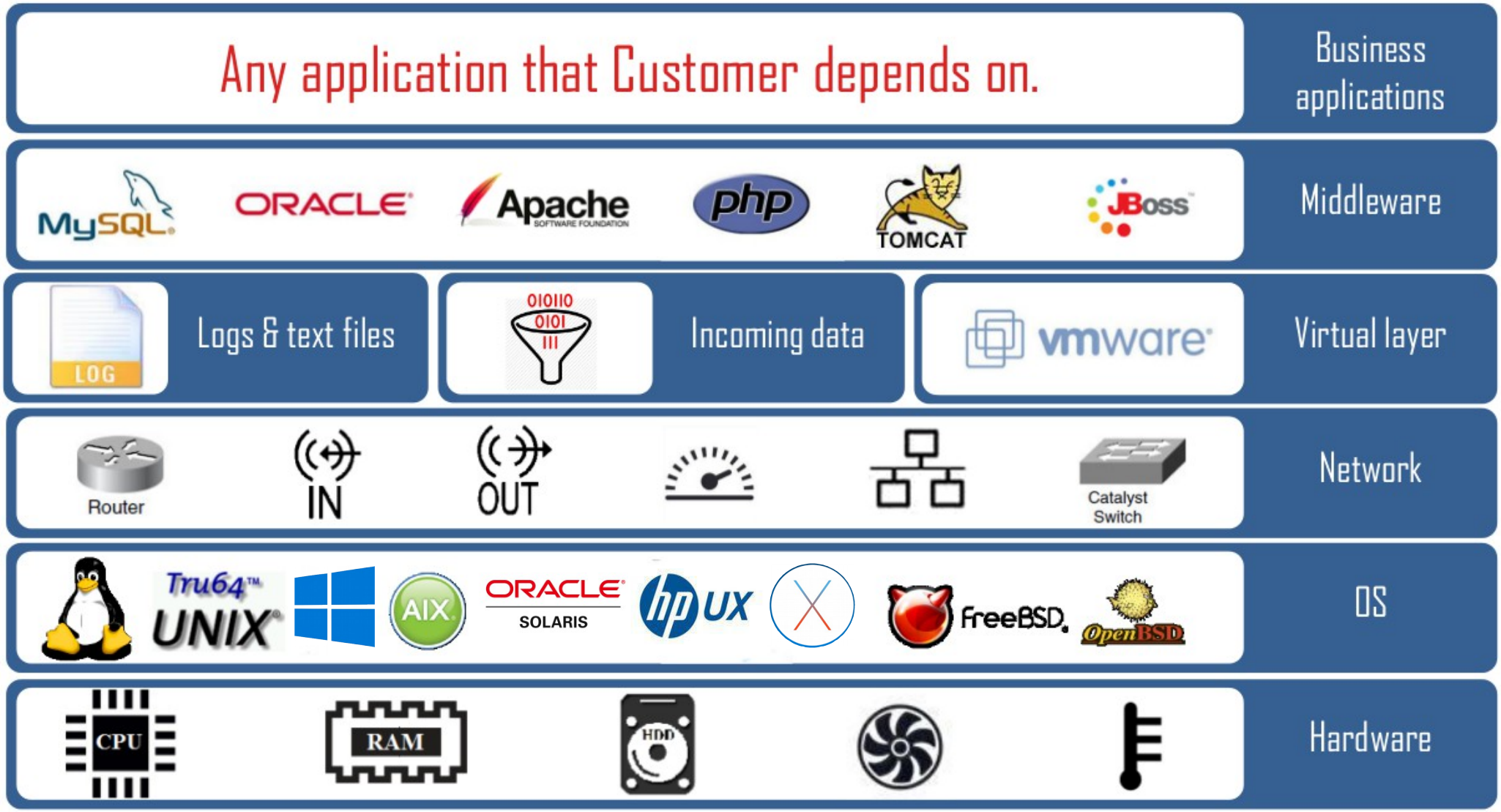


- Per user (can't be set by admin)
- Timeout
- Play sound once/10 sec/forever
- Different sounds depending on severity
- Snooze/mute




DATA
COLLECTION

ALL LEVELS OF IT INFRASTRUCTURE



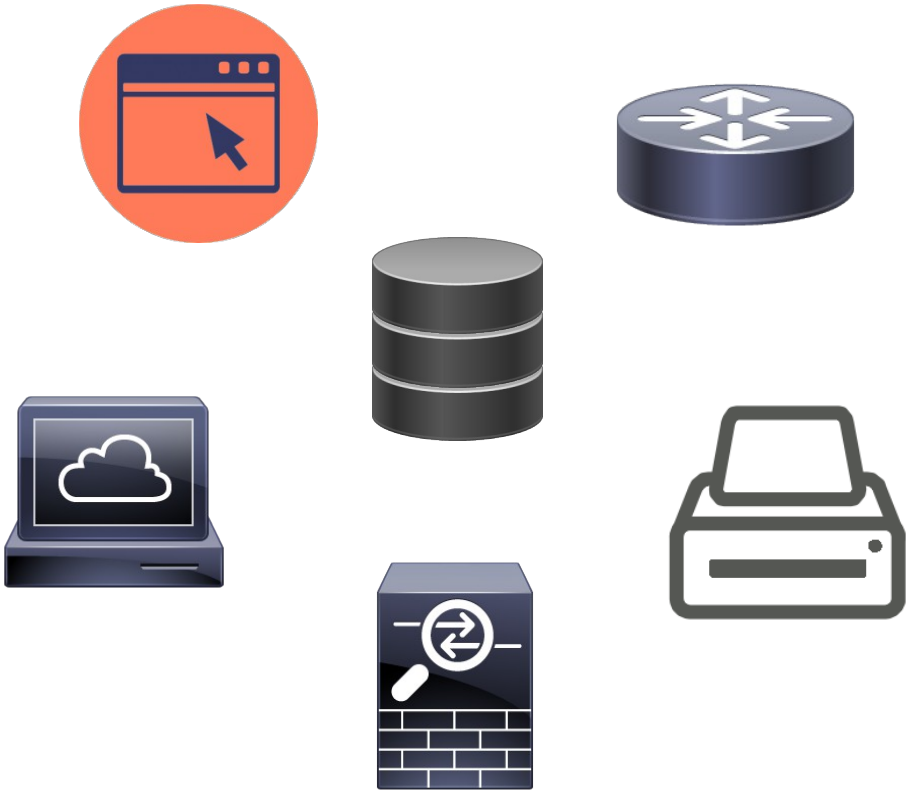
WHAT'S A HOST?

Host is the device you wish to monitor

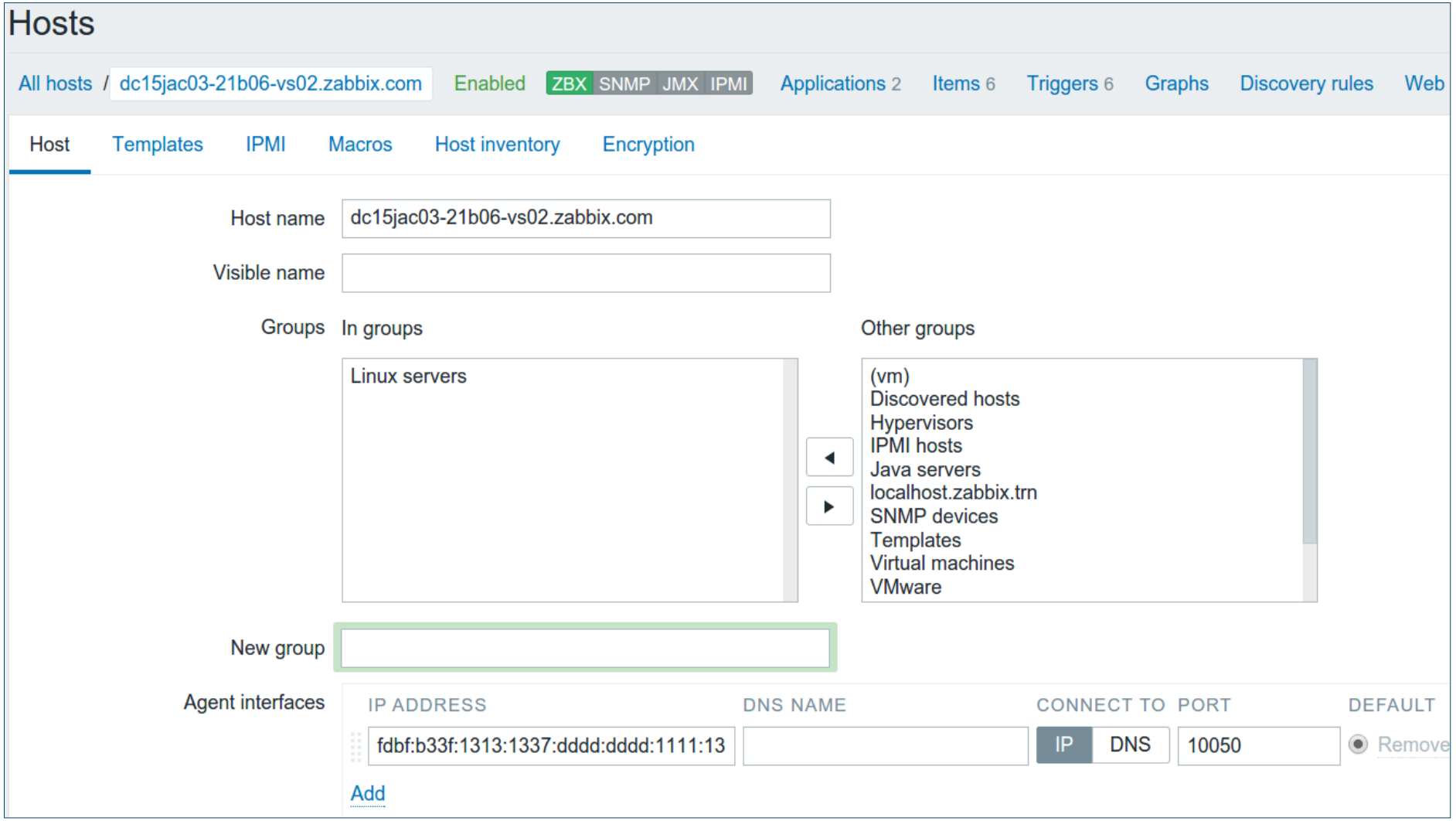
Examples:

- ✓ Server
- ✓ Switch
- ✓ UPS
- ✓ Application
- ✓ Database
- ✓ Website

...anything



HOSTS


Hosts
All hosts / dc15jac03-21b06-vs02.zabbix.com
Enabled
ZBX SNMP JMX IPMI
Applications 2
Items 6
Triggers 6
Graphs
Discovery rules
Web
Host
Templates
IPMI
Macros
Host inventory
Encryption
Host name
dc15jac03-21b06-vs02.zabbix.com
Visible name
Groups
In groups
Other groups
Linux servers
(vm)
Discovered hosts
Hypervisors
IPMI hosts
Java servers
localhost.zabbix.trn
SNMP devices
Templates
Virtual machines
VMware
New group
Agent interfaces
IP ADDRESS
DNS NAME
CONNECT TO
PORT
DEFAULT
fdbf:b33f:1313:1337:dddd:dddd:1111:13
IP
DNS
10050
Remove
Add

HOST PROPERTIES

Name

Visible name

Groups

New group

Interfaces - IP (recommended) / DNS

- Agent
- SNMP
- JMX
- IPMI

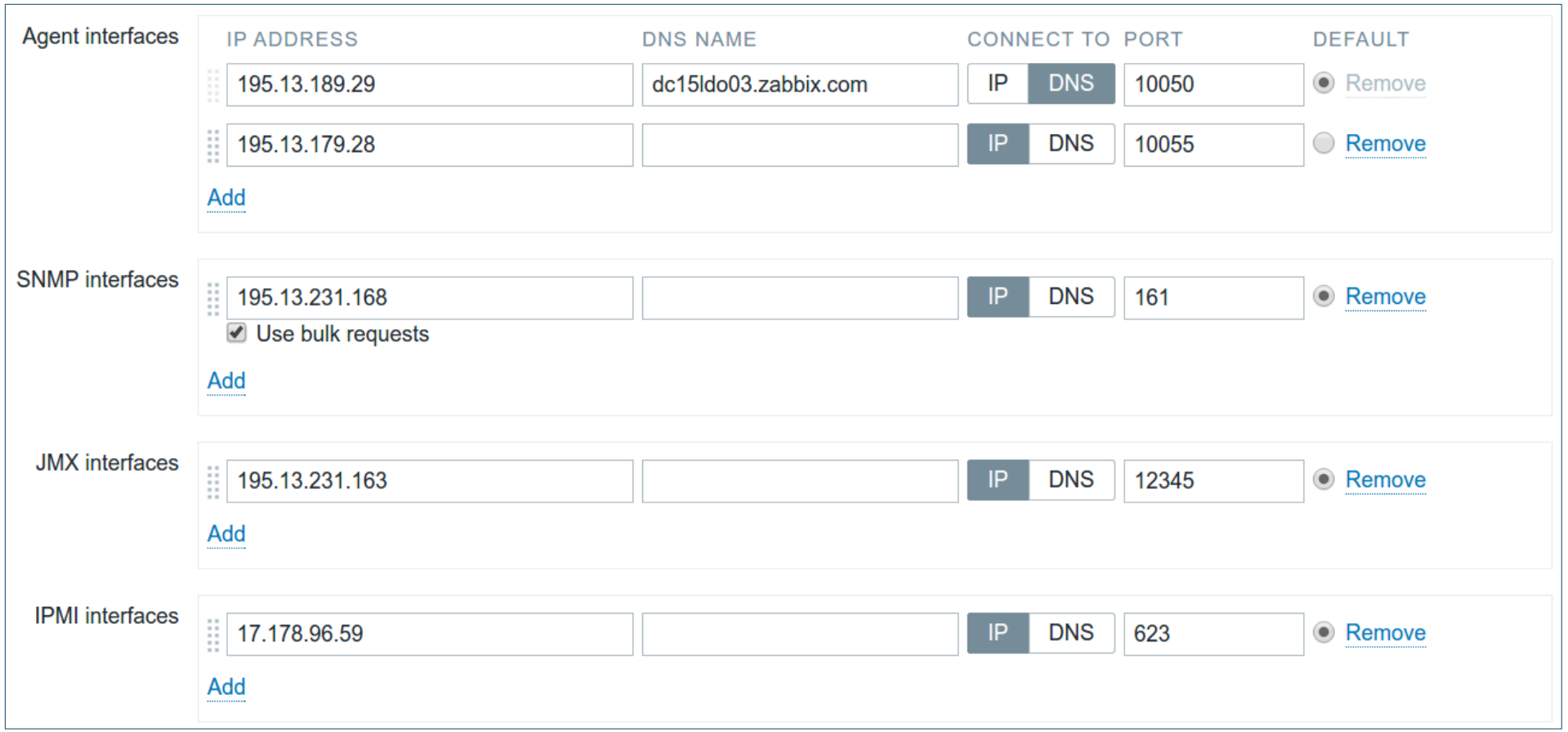
Monitored by proxy

Status

Other tabs:

- Templates
- IPMI
- Macros
- Inventory

HOST INTERFACES



PRACTICAL SETUP

Create "Training servers" host group in the frontend

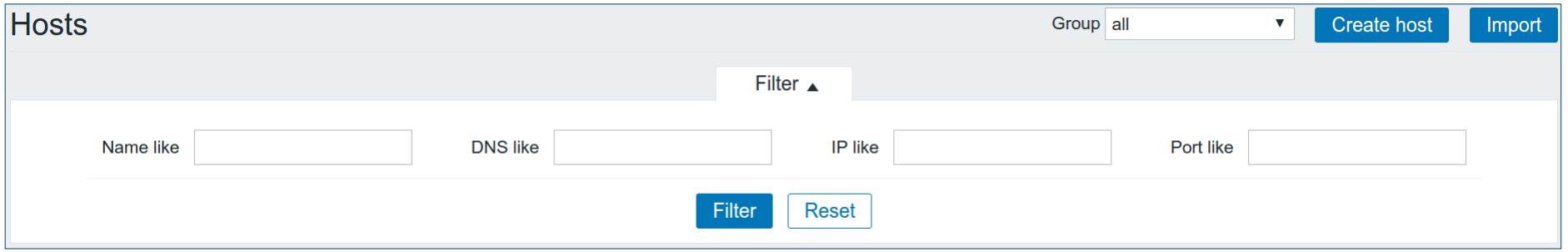
Create a new host

Use your VM name as host name

Put it in the "Training servers" host group

HOST CONFIGURATION VIEW

Host filter

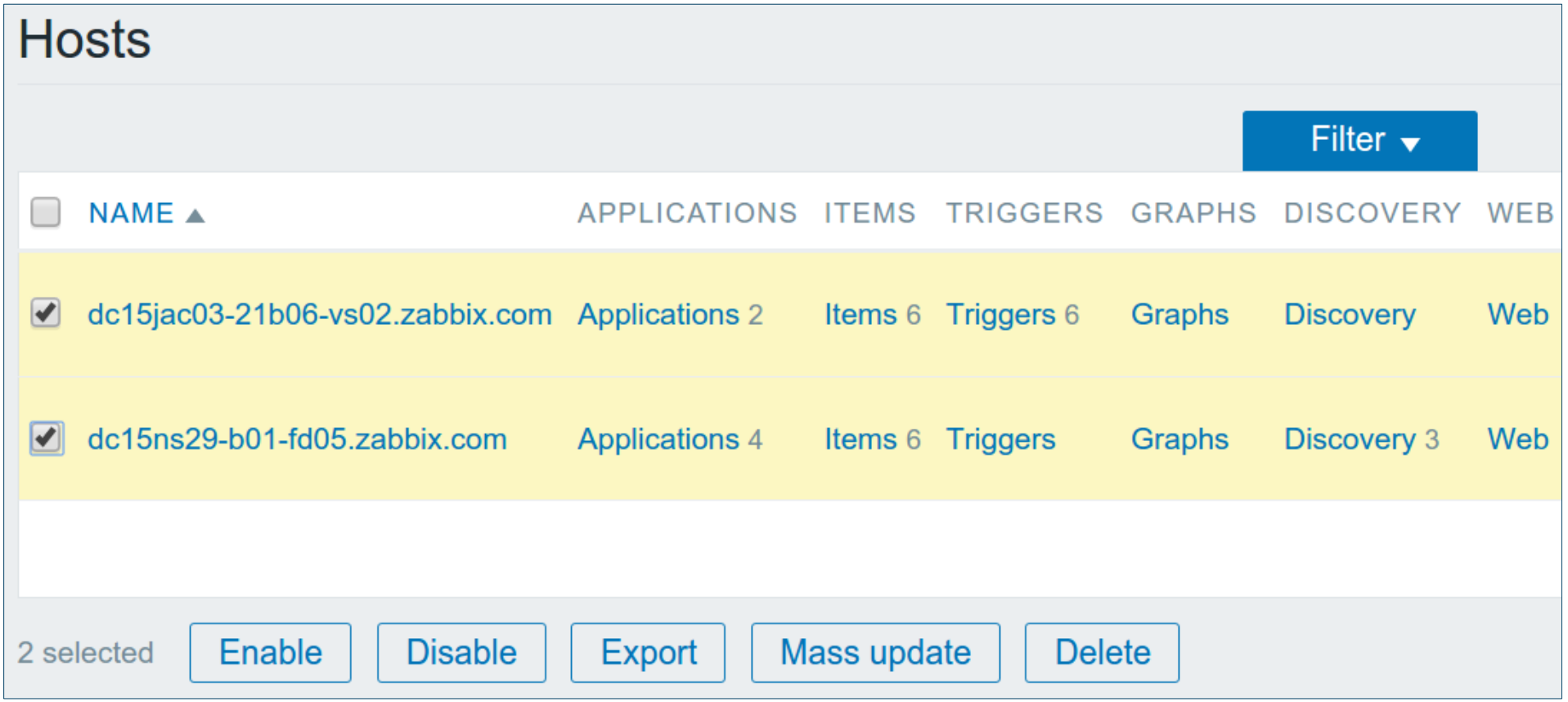


Links to entity configuration

NAME	APPLICATIONS	ITEMS	TRIGGERS	GRAPHS	DISCOVERY	WEB
dc3ns27-b01-fd04.zabbix.com	Applications 4	Items 6	Triggers	Graphs	Discovery 3	Web
dc15jac03-21b06-vs02.zabbix.com	Applications 2	Items 6	Triggers 6	Graphs	Discovery	Web

MASS EDITING FOR HOSTS

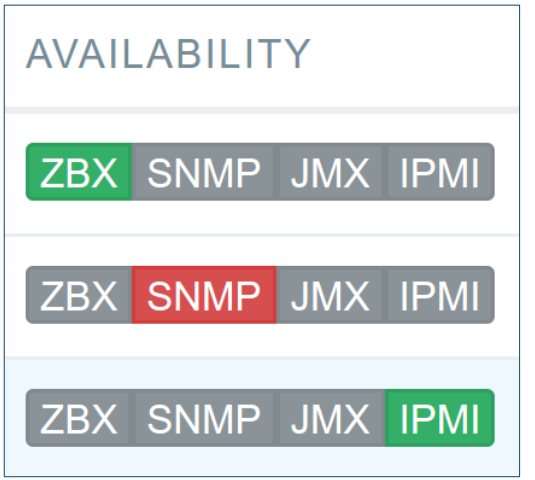
Configuration → Hosts



WHAT IS HOST AVAILABILITY?

Availability is kept for 4 different types of checks separately:

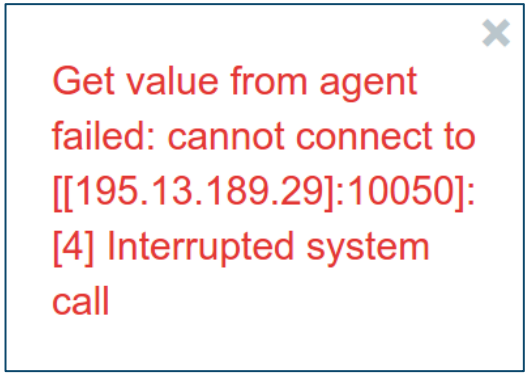
- ✓ Zabbix passive agent
- ✓ SNMP
- ✓ JMX
- ✓ IPMI



Error messages are preserved for each

Calculated by server internally

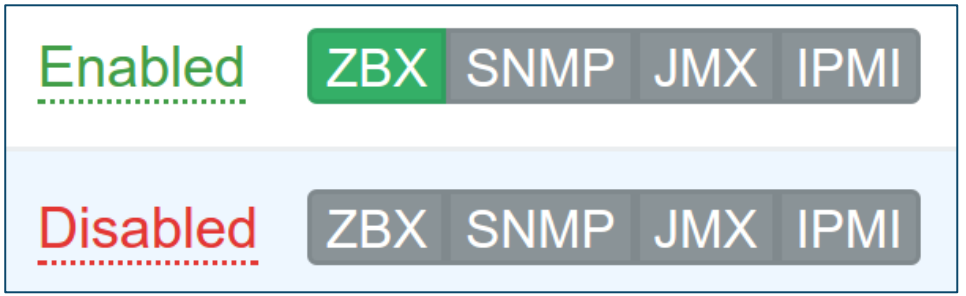
Shown in the list and host properties



HOST AVAILABILITY STATUS

Zabbix server will set the host availability icon to gray:

- there are no enabled items on the corresponding interface
- host is set to be monitored by proxy, a different proxy or by server
- host is monitored by a proxy that appears to be offline
- host is disabled

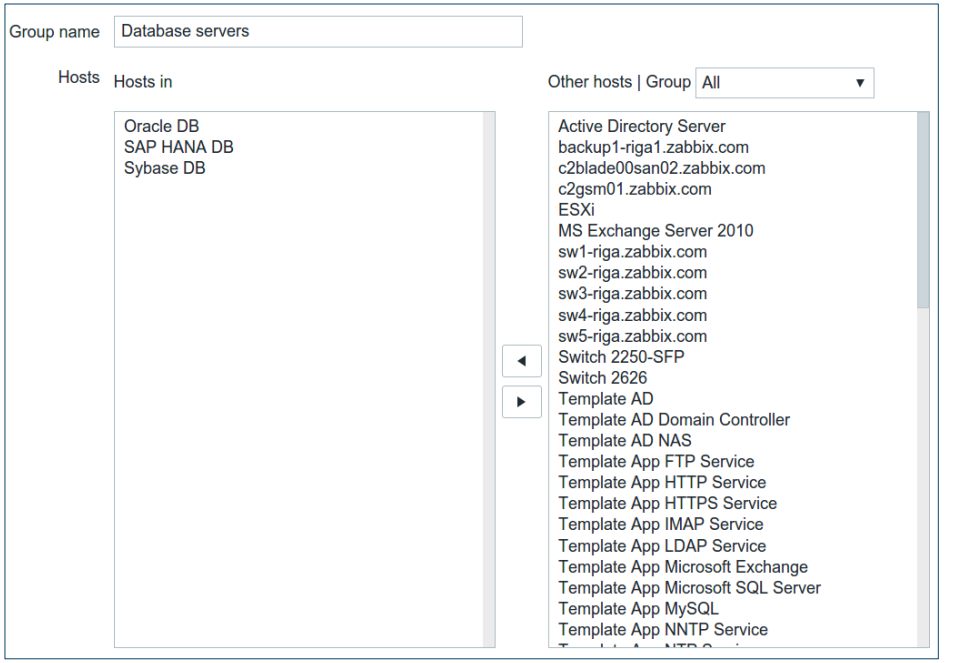


HOST GROUPS

Many hosts can belong to one group

Host can belong to any number of groups

Groups are used for permissions



SAP HANA servers	Hosts 1	Templates	SAP HANA DB
Service servers	Hosts 1	Templates	ADServer Riga
SL Servers	Hosts 3	Templates	backup1-riga1.zabbix.com, c2blade00san02.zabbix.com, c2gsm01.zabbix.com
Soft Routers	Hosts 2	Templates	pe1-kgn1.zabbix.com, pe1-mgn1.zabbix.com
Sybase servers	Hosts 1	Templates	Sybase DB


ITEMS

WHAT'S AN ITEM?

Item defines a metric which you would like to monitor

Examples:

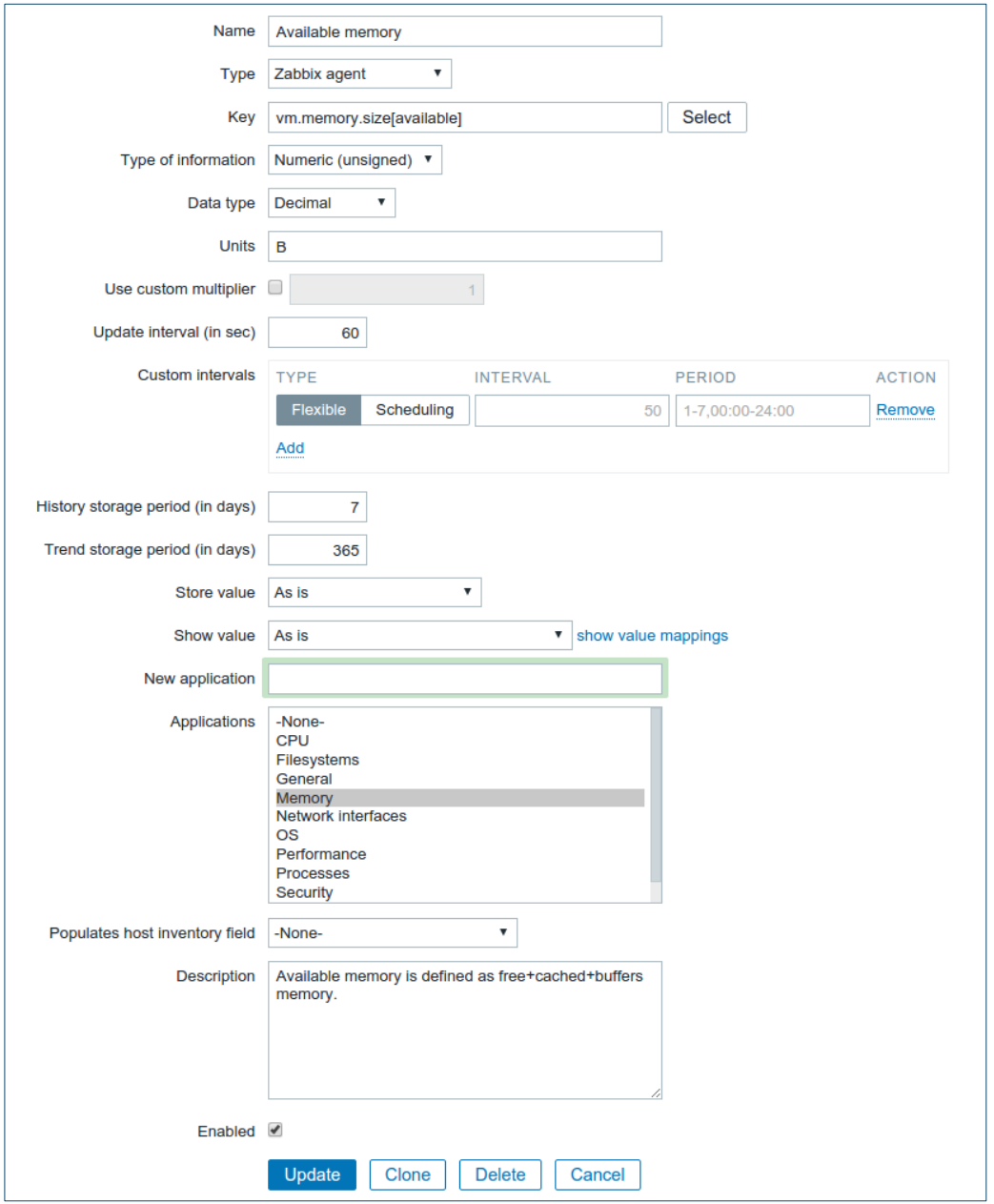
- ✓ CPU utilization
- ✓ DB status
- ✓ Temperature in a server room
- ✓ Number of users online for an application

...anything





ITEMS


Name
Available memory
Type
Zabbix agent
Key
vm.memory.size[available]
Select
Type of information
Numeric (unsigned)
Data type
Decimal
Units
B
Use custom multiplier
1
Update interval (in sec)
60
Custom intervals
TYPE
INTERVAL
PERIOD
ACTION
Flexible
Scheduling
50
1-7,00:00-24:00
Remove
Add
History storage period (in days)
7
Trend storage period (in days)
365
Store value
As is
Show value
As is
show value mappings
New application
Applications
-None-
CPU
Filesystems
General
Memory
Network interfaces
OS
Performance
Processes
Security
Populates host inventory field
-None-
Description
Available memory is defined as free+cached+buffers memory.
Enabled
Update
Clone
Delete
Cancel

ITEM TYPES

Zabbix agent

- ✓ Polled by Zabbix server

Zabbix agent (active)

- ✓ Processed by Zabbix agent
- ✓ Can be cached

Agent-less checks (simple)

- ✓ Performed by Zabbix server

SNMP agent/trapper

- ✓ All SNMP versions are supported

Zabbix trapper

- ✓ Used with Zabbix sender

Internal

- ✓ Zabbix health

IPMI

JMX

Aggregate

grpsum["MySQL Servers","vfs.fs.size[/,total]","last", "0"]

External check

script[parameters]

SSH

- ✓ Password and key authentication supported

Telnet

Database

Calculated

last("vm.memory.size[free]")+

last("vm.memory.size[buffers]")


ZABBIX AGENT
CHECKS

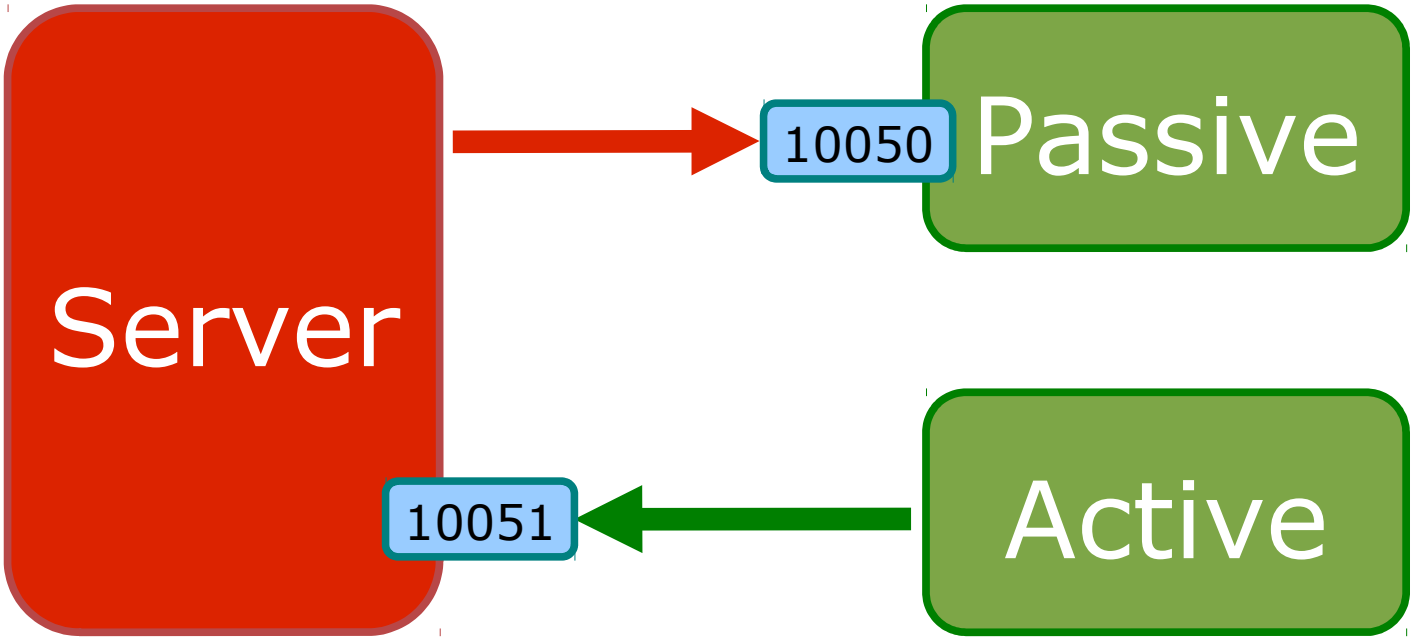
COMMUNICATIONS

JSON based protocol

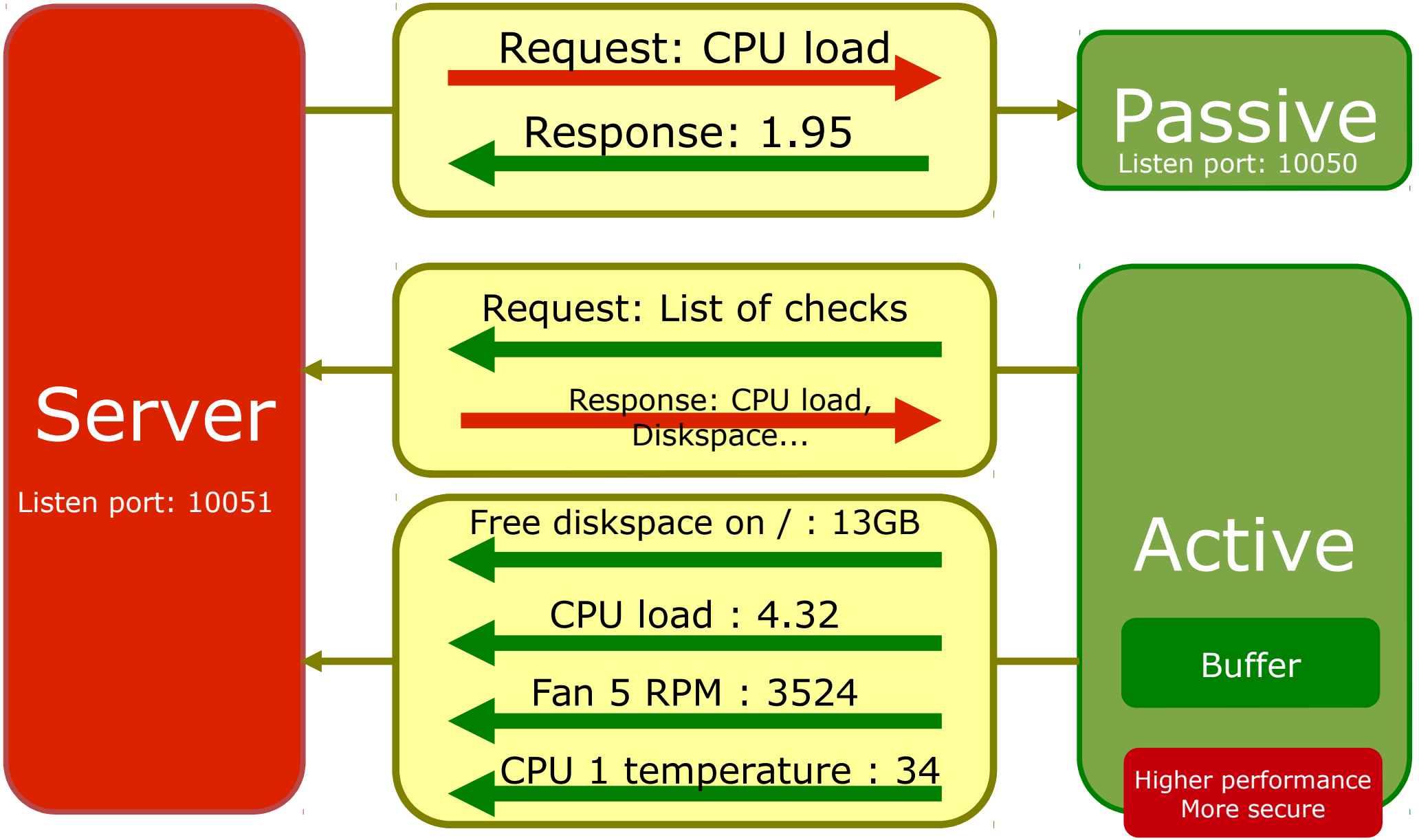
- 1.0, 1.1: very simple protocol
- 1.4: XML
- 1.6+: JSON

PASSIVE VS ACTIVE CHECKS

- ✓ Passive (pull)
- ✓ Active (push)



DATA FLOW


Server
Listen port: 10051
Request: CPU load
Response: 1.95
Passive
Listen port: 10050
Request: List of checks
Response: CPU load, Diskspace...
Free diskspace on / : 13GB
CPU load : 4.32
Fan 5 RPM : 3524
CPU 1 temperature : 34
Active
Buffer
Higher performance
More secure

PASSIVE VS ACTIVE - IDENTIFICATION

- Passive

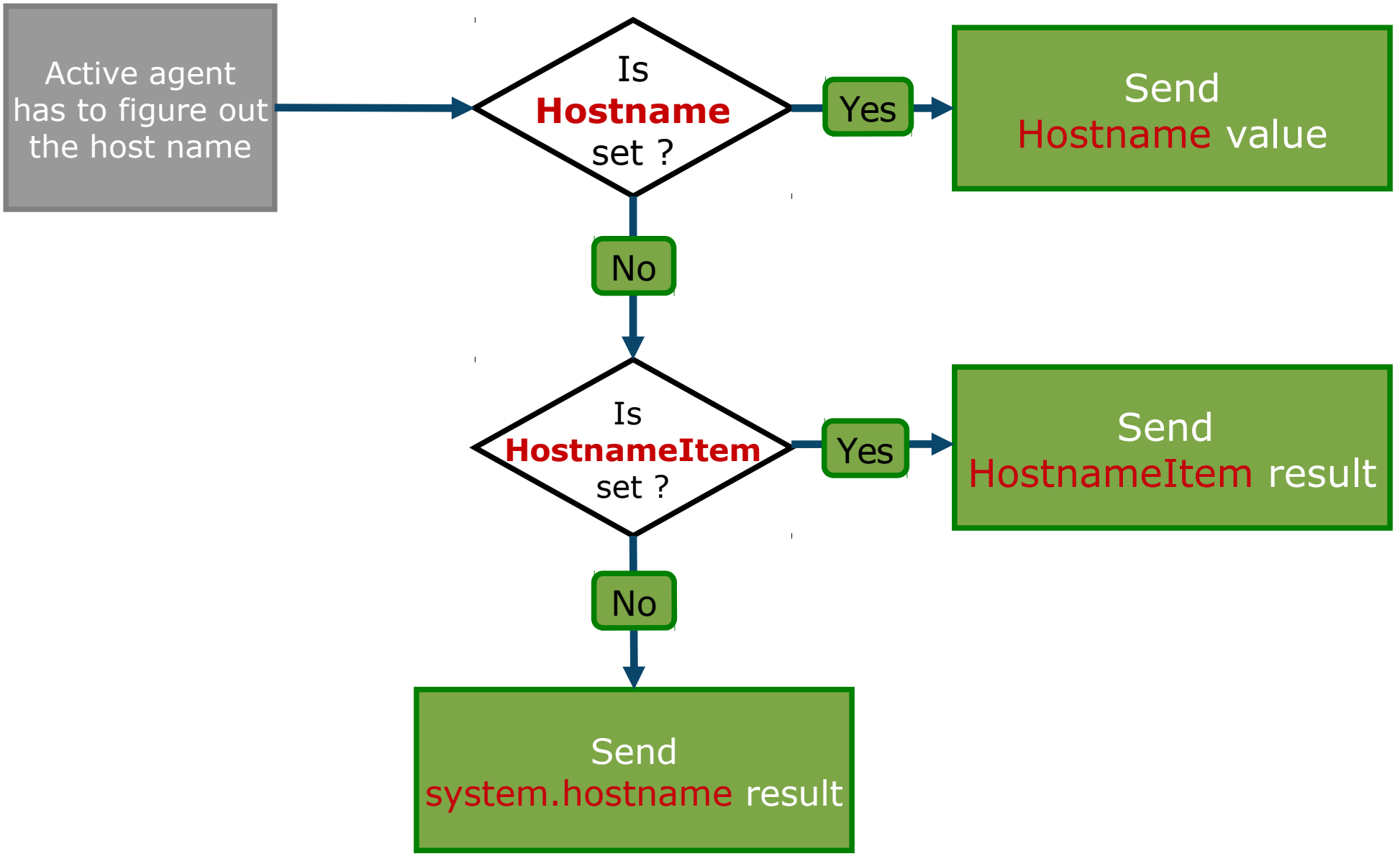
  IP address / DNS name

- Active

  Explicitly set Hostname

  HostnameItem if Hostname unset

  system.hostname by default


Active agent has to figure out the host name
Is Hostname set ?
Yes
Send Hostname value
No
Is HostnameItem set ?
Yes
Send HostnameItem result
No
Send system.hostname result

ITEM KEY

General syntax: key[parameter1,parameter2,parameter3]

Must be unique per host

Flexible / non-flexible (net.tcp.listen[631] / agent.ping)

Use $1, $2…$9 in the item name to refer to the first, second… ninth parameter of the item key

Use quotes

system.swap.size[/proc/swaps,free]

vs

system.swap.size["/proc/swaps,free"]

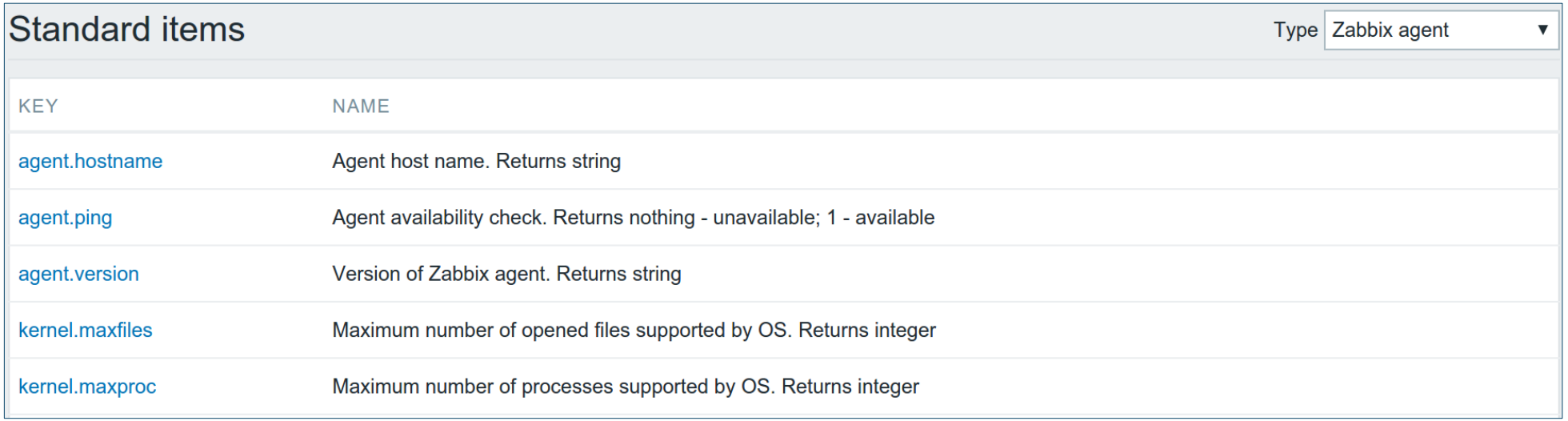
ITEM KEY IN MORE DETAILS

Free form string for SNMP & IPMI (OID & IPMI sensor matters)

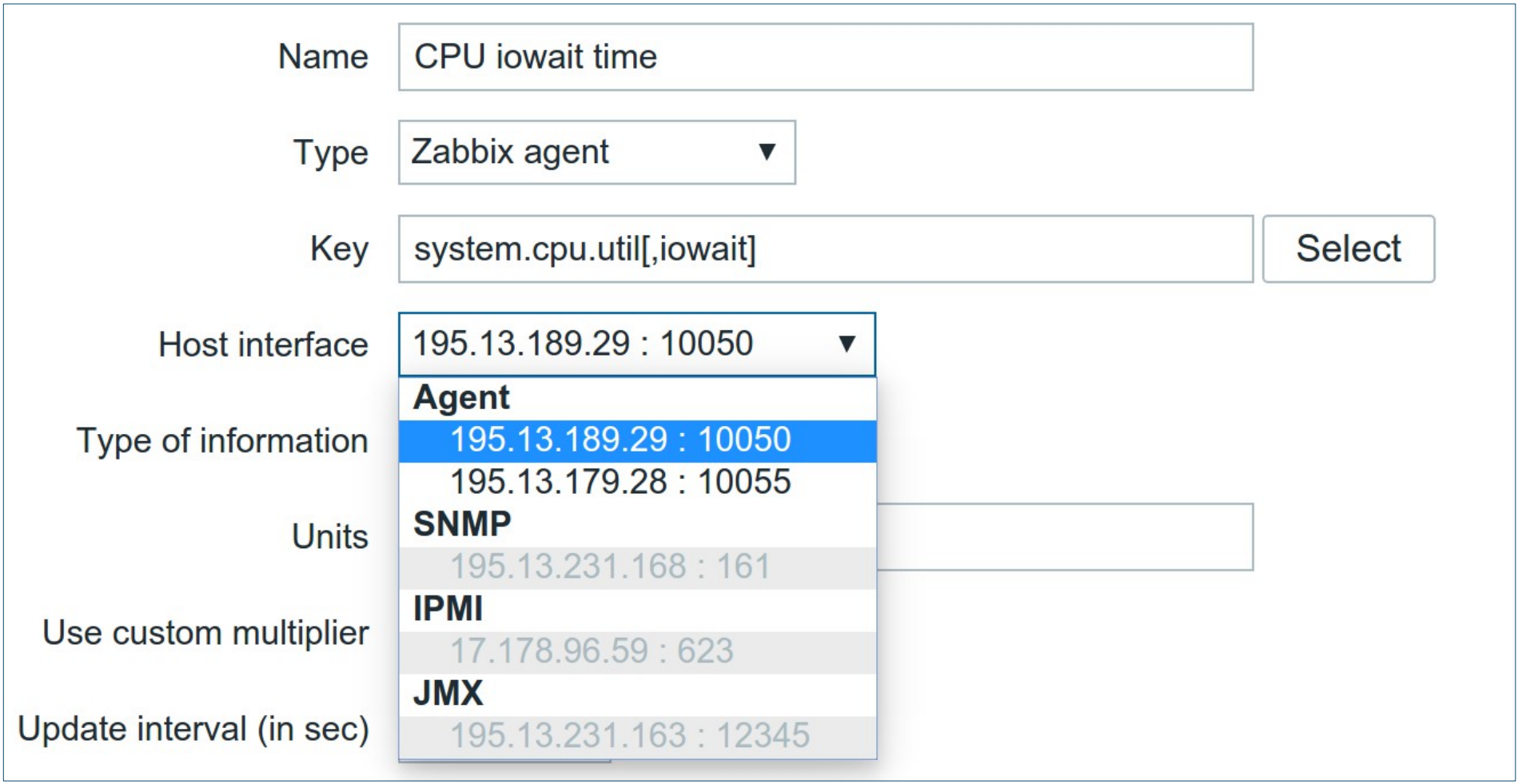
Quote parameters (proc.mem["httpd",apache,sum])

Quick reference in the frontend

See Zabbix manual for the list of supported keys



INTERFACES FOR ITEMS



ITEM UNITS

Units

For numeric data only

If set, K/M/G/T/P/E/Z/Y prefix fill be added:

- 5242880 B -> 5 MB

Special processing for:

- B, Bps, unixtime, uptime
- Unit blacklist for %, ms, RPM, rpm

ITEM PRE-PROCESSING

Store value

Store as is: no changes of received value

Delta (speed per second): calculate as (value-prevvalue)/(time-prevtime)

Useful for monitoring counters

Delta (simple change): calculate as (value-prevvalue)

Data type

Decimal, octal, hex, boolean

Multiplier

Calculated as value * multiplier

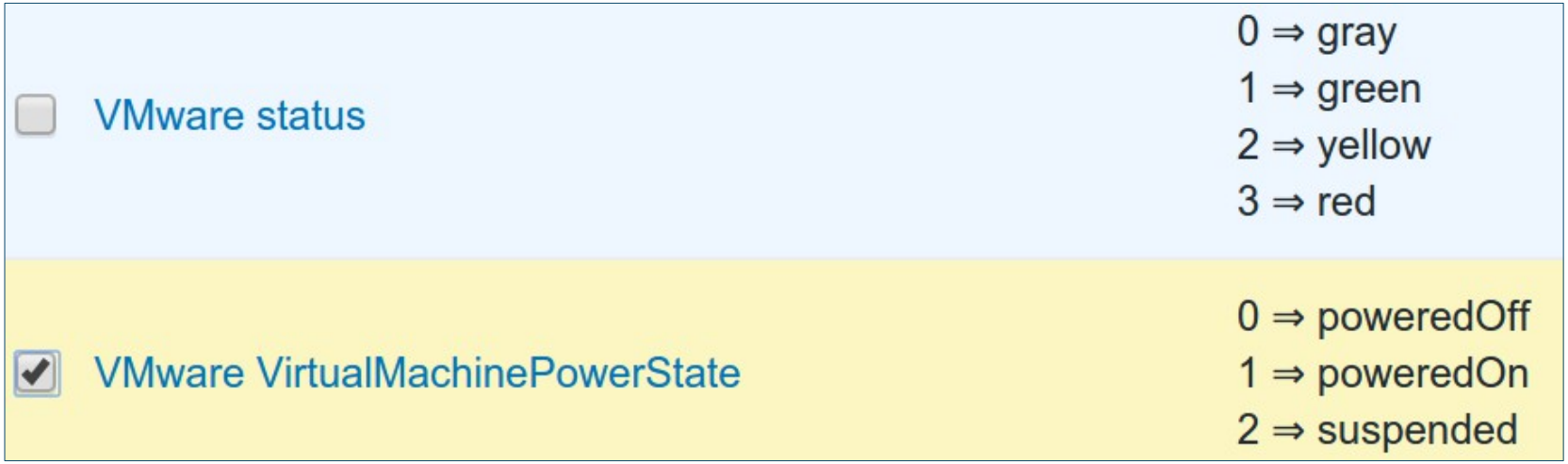
Use 0.125 to divide by 8

VALUE MAPPING

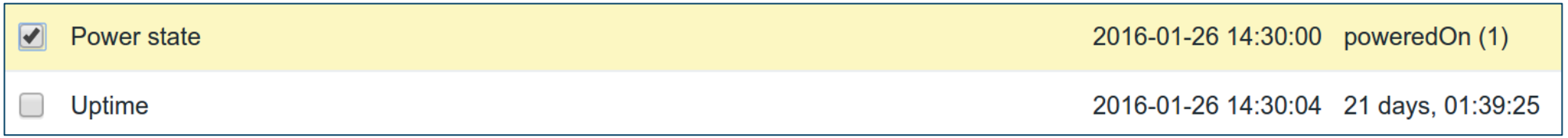
Used almost everywhere in the frontend and for notifications

Support of string values

Administration → General → Value mapping



Example: Monitoring → Latest data



CHANGES STORED DATA OR NOT?

Do not change stored data (value stored as-is)

Units

Value mapping

Change stored data

Data type

Store value

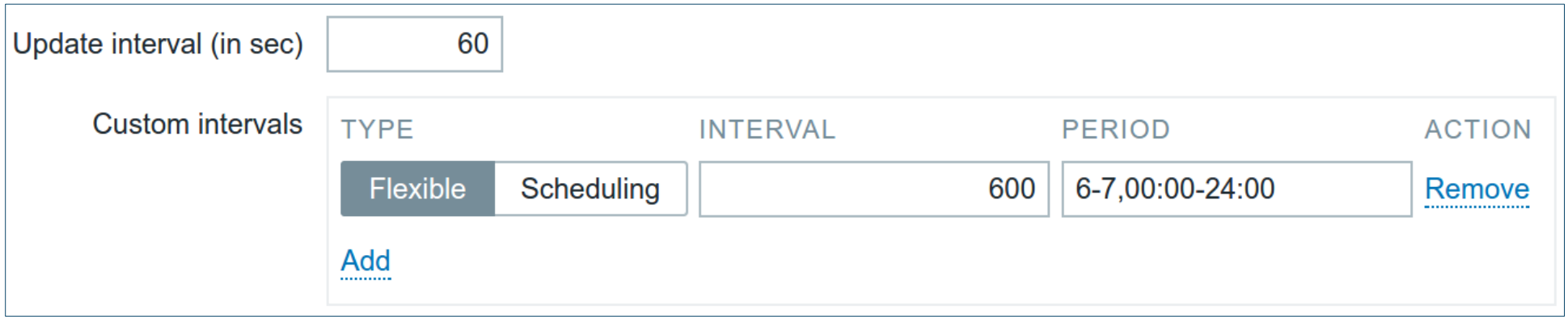
Multiplier

FLEXIBLE INTERVALS

Allows to override default item interval

If multiple flexible intervals overlap, smallest time is used

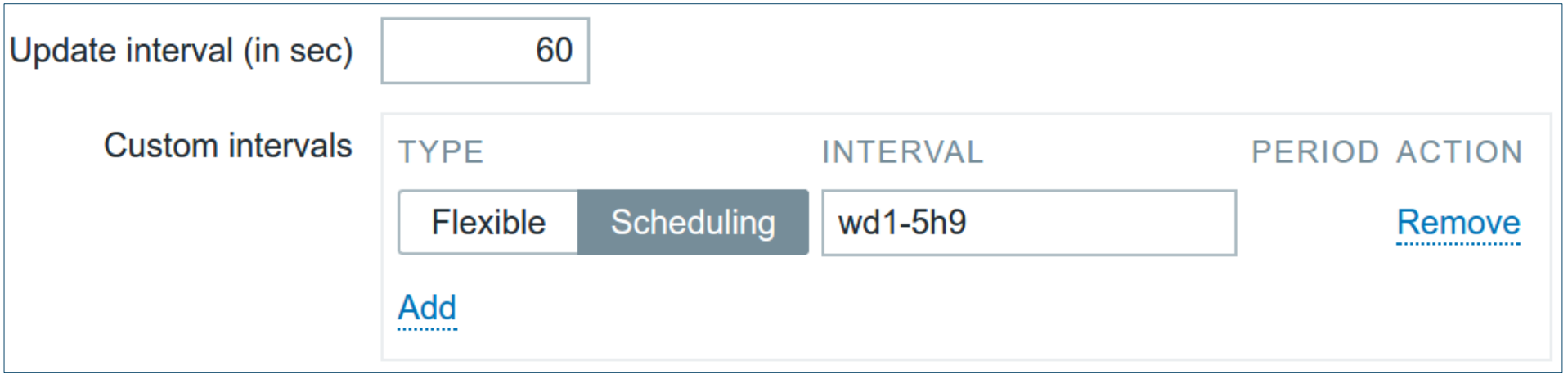
Default delay of 0 can be used together with a flexible interval to emulate scheduling on a specific time of day



Be aware: not supported as agent active check

EXECUTION AT A SPECIFIC TIME

Allows to check items at specific times

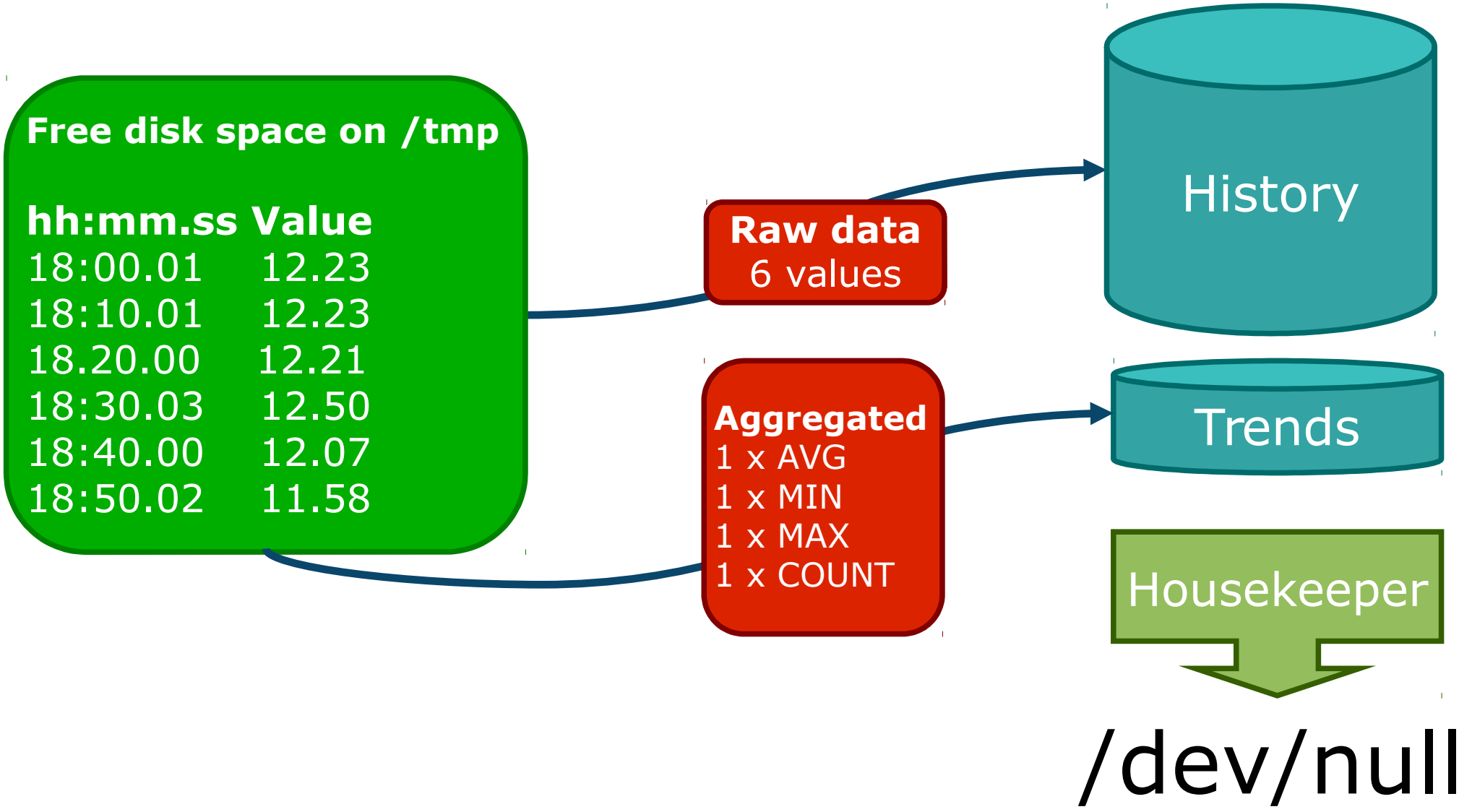


Examples:

wd1-5h9 - every Monday till Friday at 9:00
h9m/30;h10 - execute at 9:00, 9:30, 10:00
h9-10m10-40/30 - execute at 9:10, 9:40, 10:10, 10:40
md1wd1h9m30 - every 1st day of each month at 9:30 if it is Monday

Be aware: not supported as agent active check

HISTORY, TRENDS AND HOUSEKEEPER

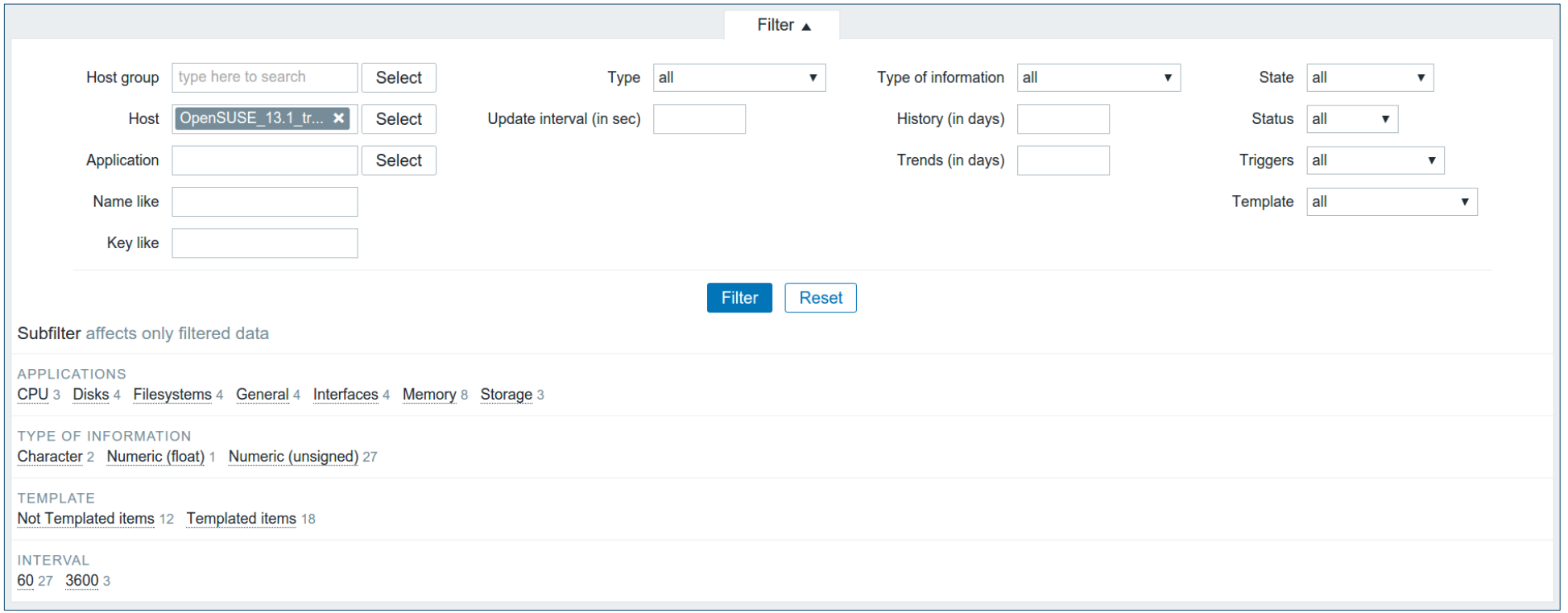


ITEM CONFIGURATION FILTER

Find items from multiple hosts

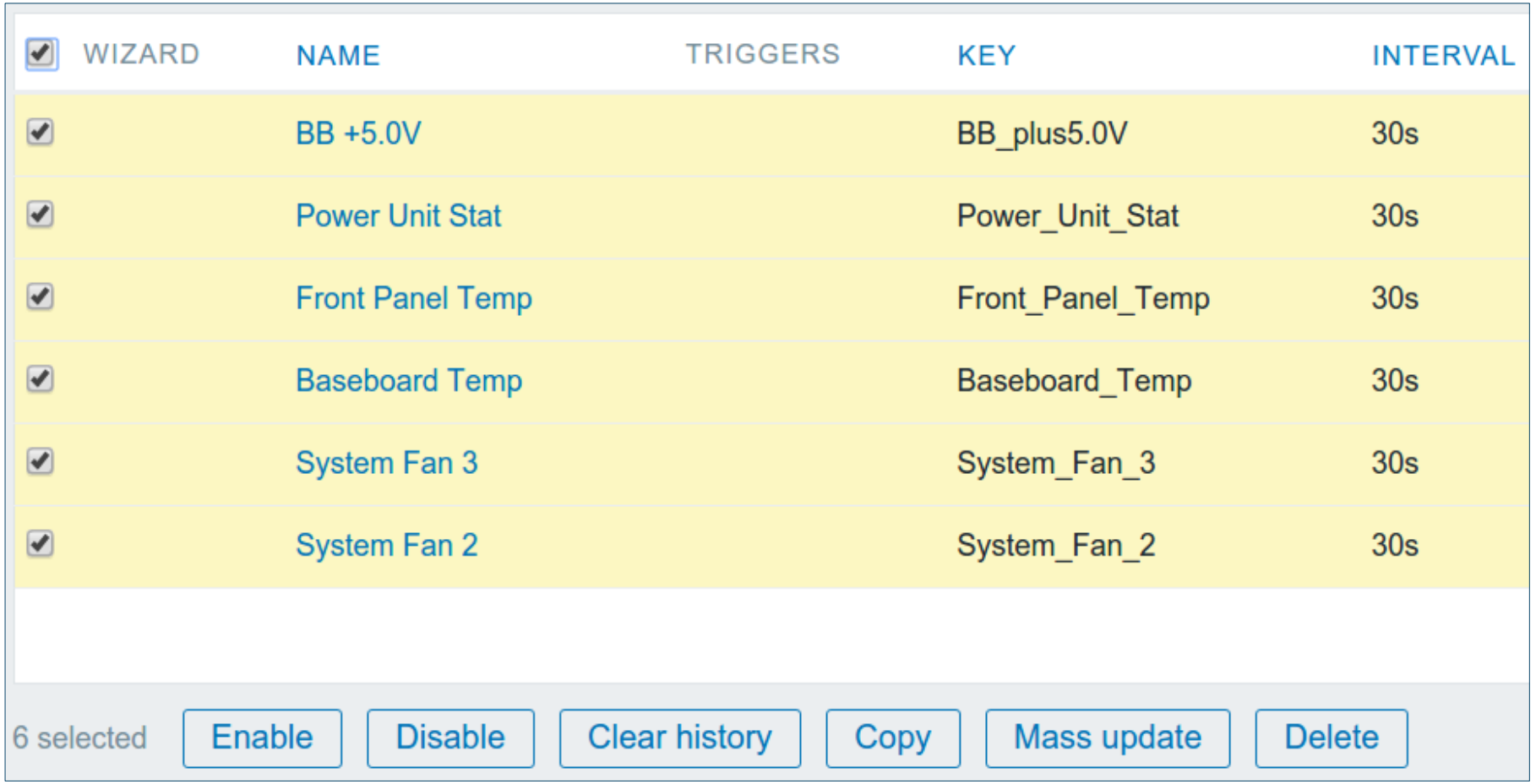
Find unsupported items

Further drill down by subfilter



MASS EDITING FOR ITEMS

Configuration → Hosts → <Host> → Items



PRACTICAL SETUP

Create three items on the host:

- "Incoming traffic on eth0" (bytes per second)
- "Outgoing traffic on eth0" (bytes per second)
- "CPU load"

Make sure that the items receive data

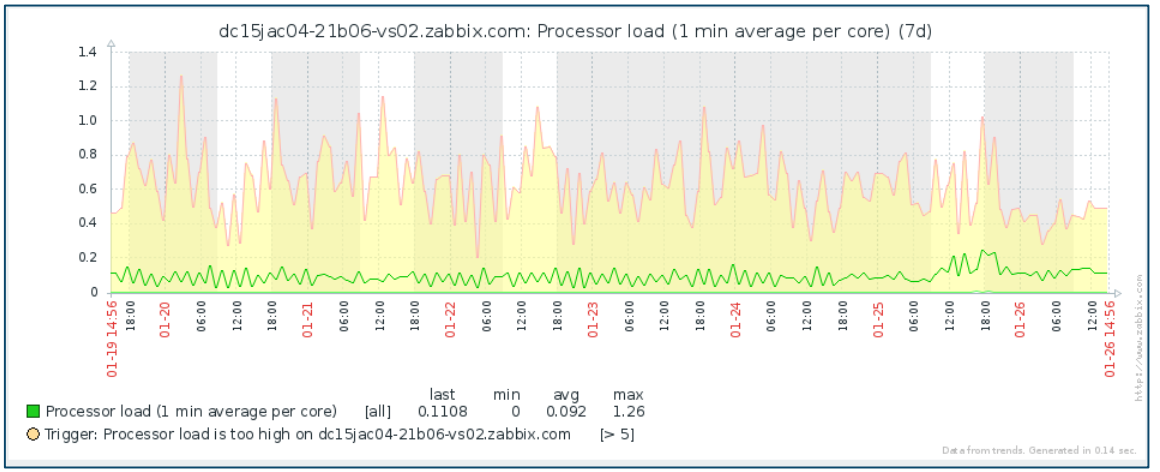
LATEST DATA

Performance data for a selected server/group

Simple graphs

Plain text information

Config details



☐	CPU idle time	2016-01-26 14:57:19	88.37 %	+0.97 %	Graph
☐	CPU interrupt time	2016-01-26 14:57:20	0 %		Graph
☐	CPU iowait time	2016-01-26 14:57:21	3.11 %	-2.29 %	Graph
☐	CPU nice time	2016-01-26 14:57:22	0 %		Graph

TIMESTAMP	VALUE
2016-01-26 14:59:21	4.9975
2016-01-26 14:58:21	5.6178
2016-01-26 14:57:21	3.1067
2016-01-26 14:56:21	5.3968
2016-01-26 14:55:21	3.2263

APPLICATIONS

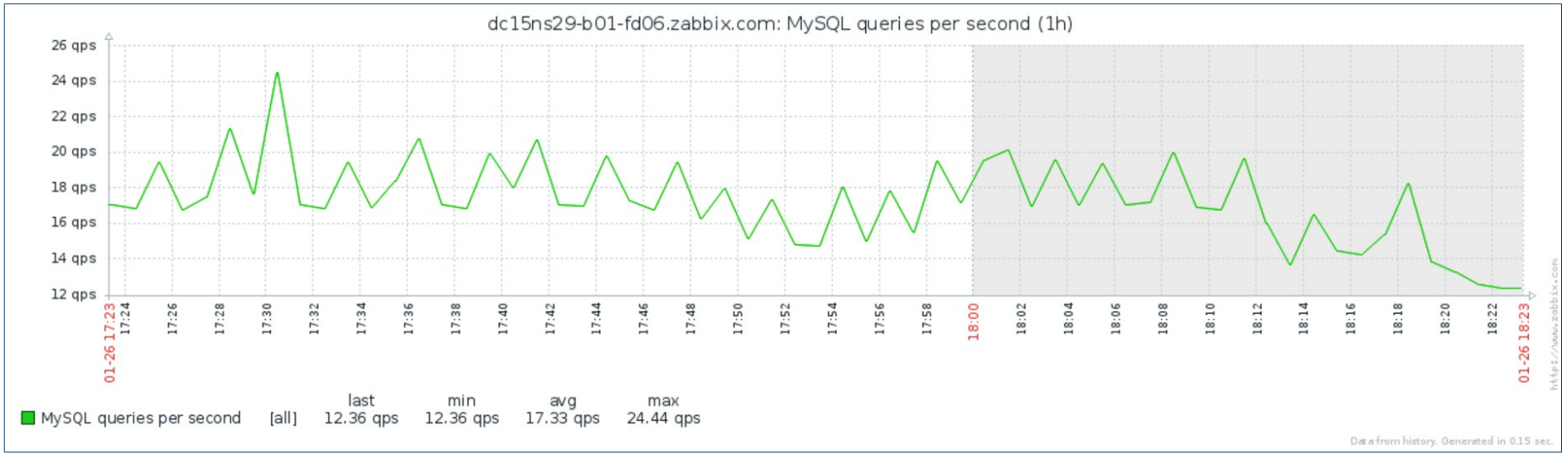
Group of items

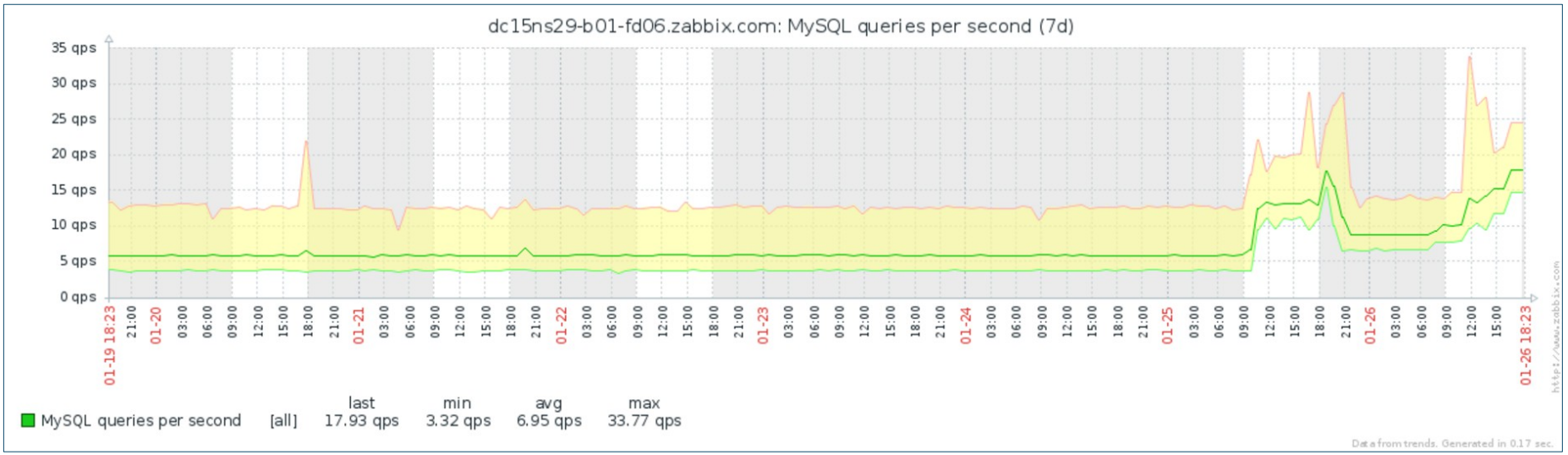
One application, many items

One item, many applications

NAME ▲	LAST CHECK	LAST VALUE
CPU (13 Items)		
Filesystems (5 Items)		
General (5 Items)		
Memory (5 Items)		
MySQL (15 Items)		
MySQL begin operations per second	2016-01-26 15:02:26	2 qps
MySQL bytes received per second	2016-01-26 15:02:24	1.18 KBps
MySQL bytes sent per second	2016-01-26 15:02:25	7.03 KBps

SIMPLE GRAPHS





OVERVIEW

Performance data for a group of servers

Displays problems

Quick navigation to Graphs and Plain text data

ITEMS	OPENSUSE_13.1_TR	WINDOWS SERVER
Average number of bytes read from the disk Hard disk 1	0 Bps	0 Bps
Average number of bytes written to the disk Hard disk 1	0 Bps	0 Bps
Average number of reads from the disk Hard disk 1	0	0
Average number of writes to the disk Hard disk 1	0	0
Ballooned memory	0 B	0 B
Cluster name		
Committed storage space	4.55 GB	4.9 GB
Compressed memory	0 B	0 B
CPU ready	11 %	15 %

COMMON ITEM KEYS

Area	Suggested key
Availability	agent.ping
Network performance	net.if.in/out[interface]
Remote services	net.tcp.service[service,<ip>,<port>]
Processes	proc.num[<name>,<user>,<state>,<cmdline>]
Disk space availability	vfs.fs.size[fs, <mode>]
Memory availability	vm.memory.size[<mode>]
Host name	system.hostname[<type>]
CPU load/utilization	system.cpu.load[] system.cpu.util[]


PROBLEM
DETECTION

WHAT IS A TRIGGER?

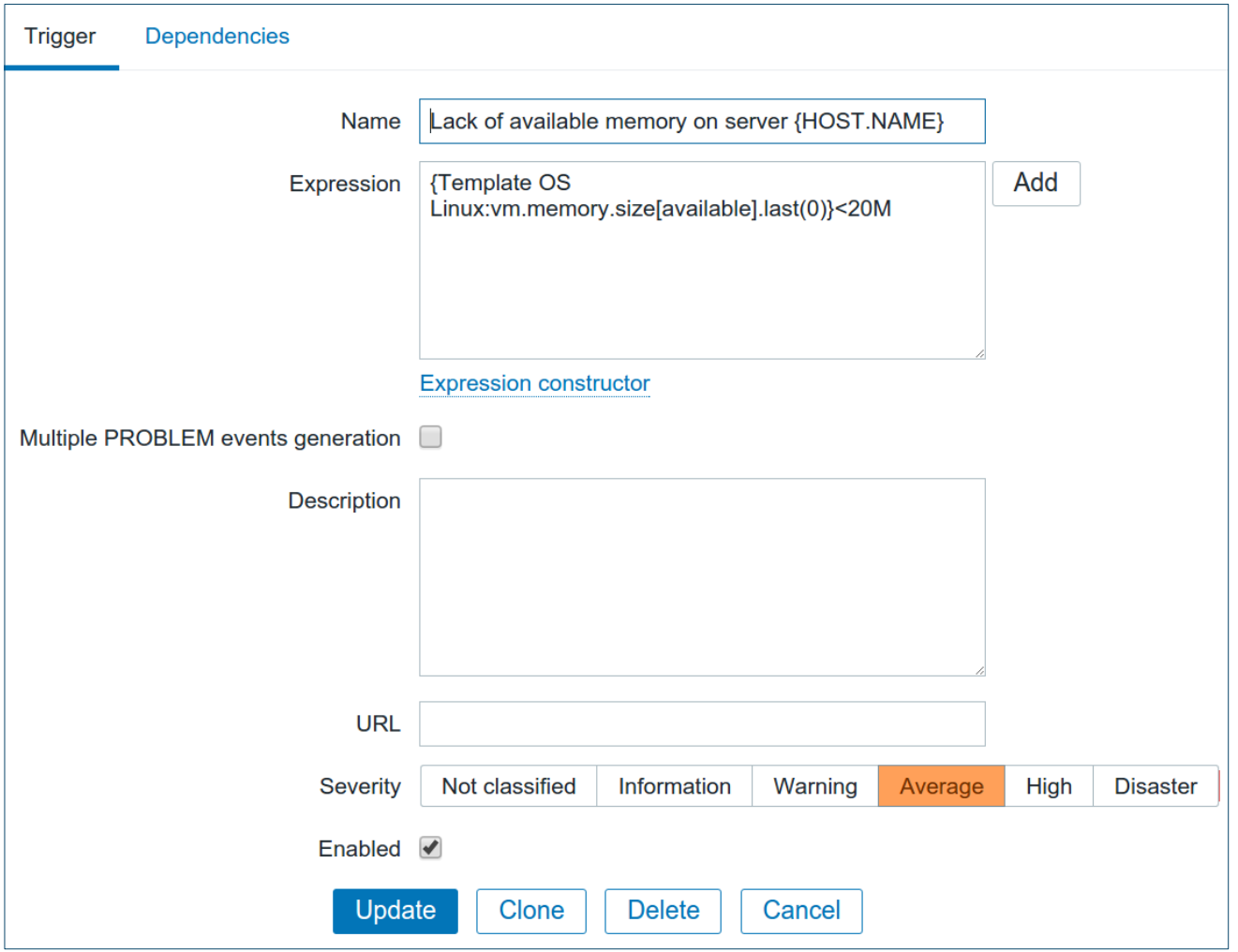
Trigger is a problem definition

Examples:

- ✓ CPU utilization is too high
- ✓ Host is unreachable using ICMP
- ✓ Database is down
- ✓ Application is not running

...anything

TRIGGER DEFINITION



- Name
- Expression
- Events generation
- Description
- Severity
- URL
- Enabled
- Dependencies

TRIGGER EXPRESSION SYNTAX

Syntax:

{host:key.function(param)}=0

{zabbix:system.cpu.load.min(300)}>10

Operators

\- + / * < > = <> >= <= or and

Refer to items from many hosts

{host1:item.fun(5m)}>10 and {host2:item.fun(5m)}>5 and {host3:item.fun(5m)}<3

Zabbix makes decisions based on all information available: latest and history

TRIGGER FUNCTIONS

Functions:

min, max, avg, last, diff, count, delta, time, etc

See Zabbix manual

Parameters:

<str> - normal parameters

{zabbix:system.cpu.load.min(10m)}>5

#<num> - number of checks

{zabbix:system.cpu.load.min(#10)}>5

Supported suffixes: s, m, h, d, w

LESS SENSITIVE TRIGGER EXPRESSIONS

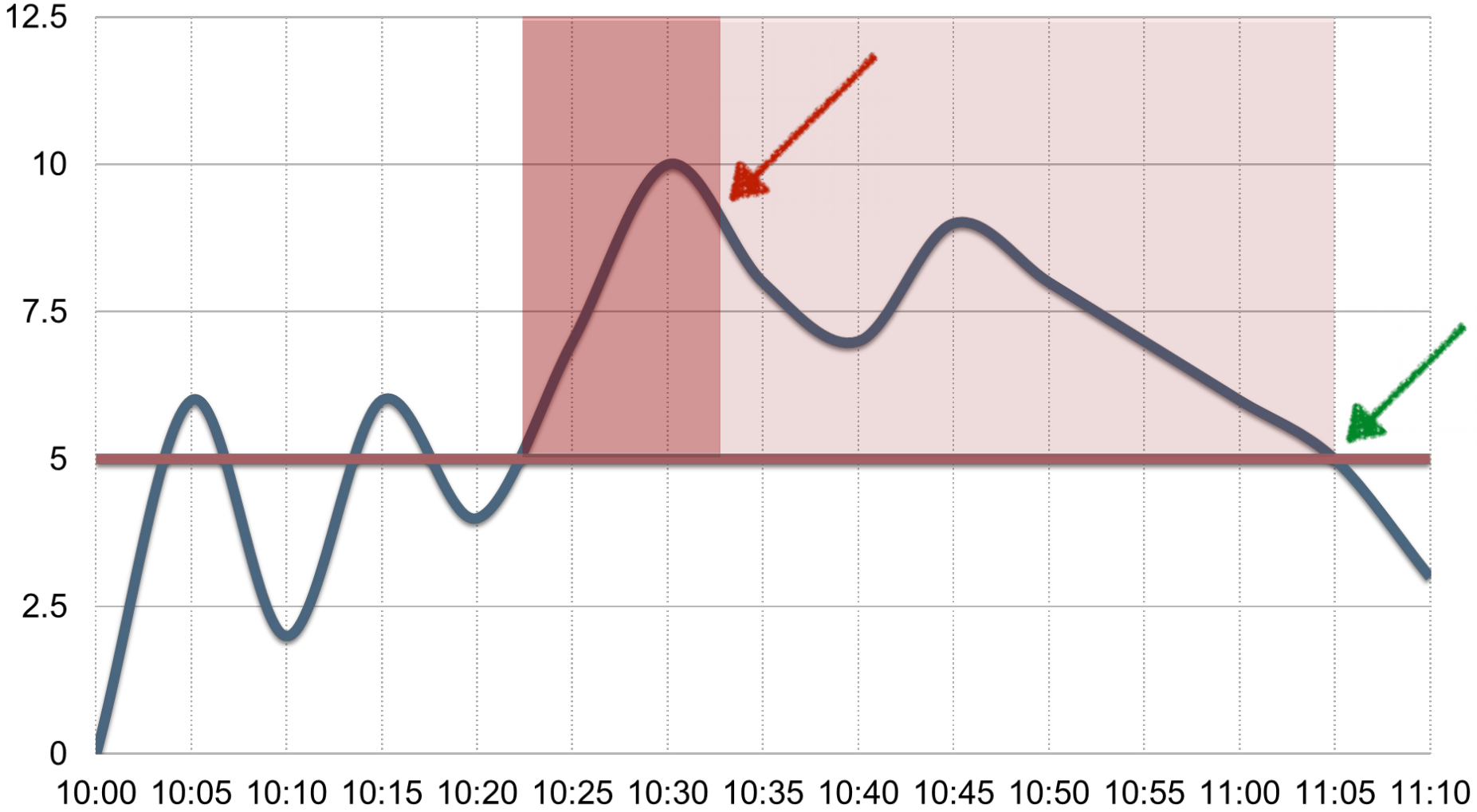
For example, use:

min(10m) > 5 for CPU load

min(#10) > 5 for CPU load

max(10m) = 0 for availability check

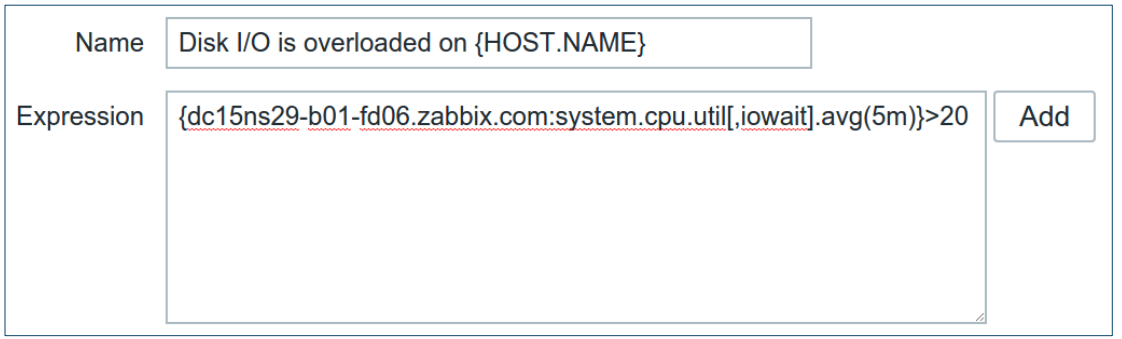
ANALYSE HISTORY



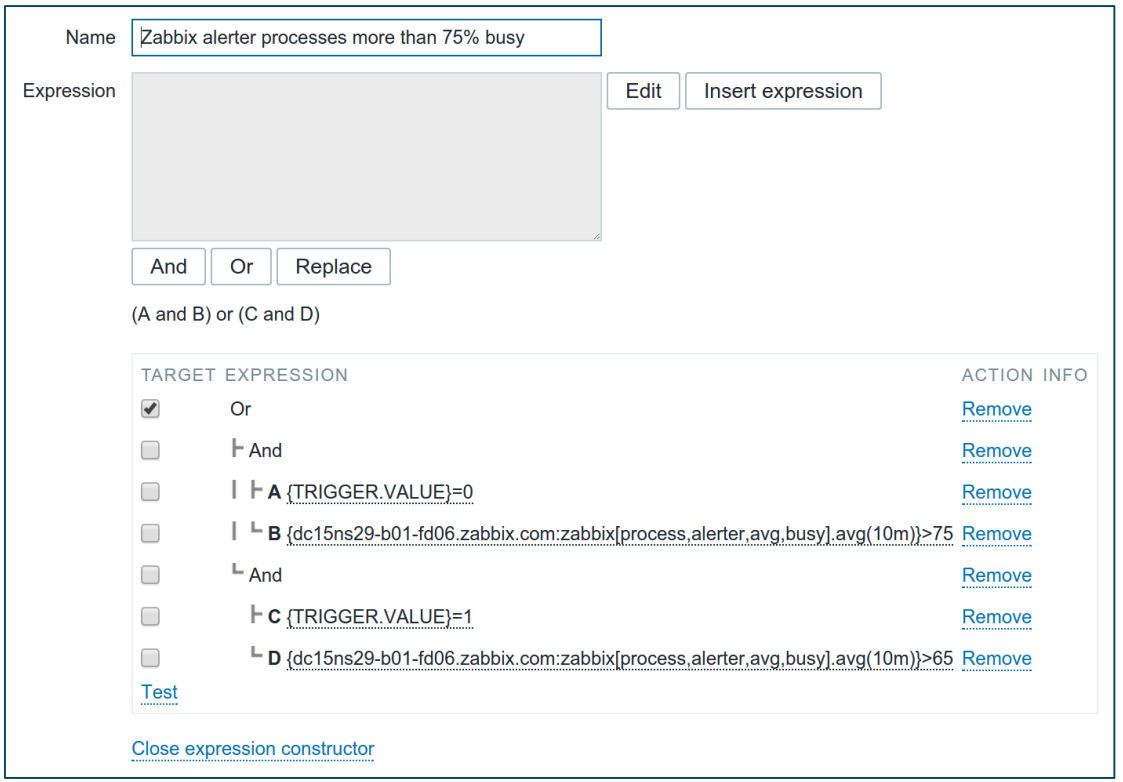
{server:system.cpu.load.min(10m)} > 5

TRIGGER EXPRESSION EDITING

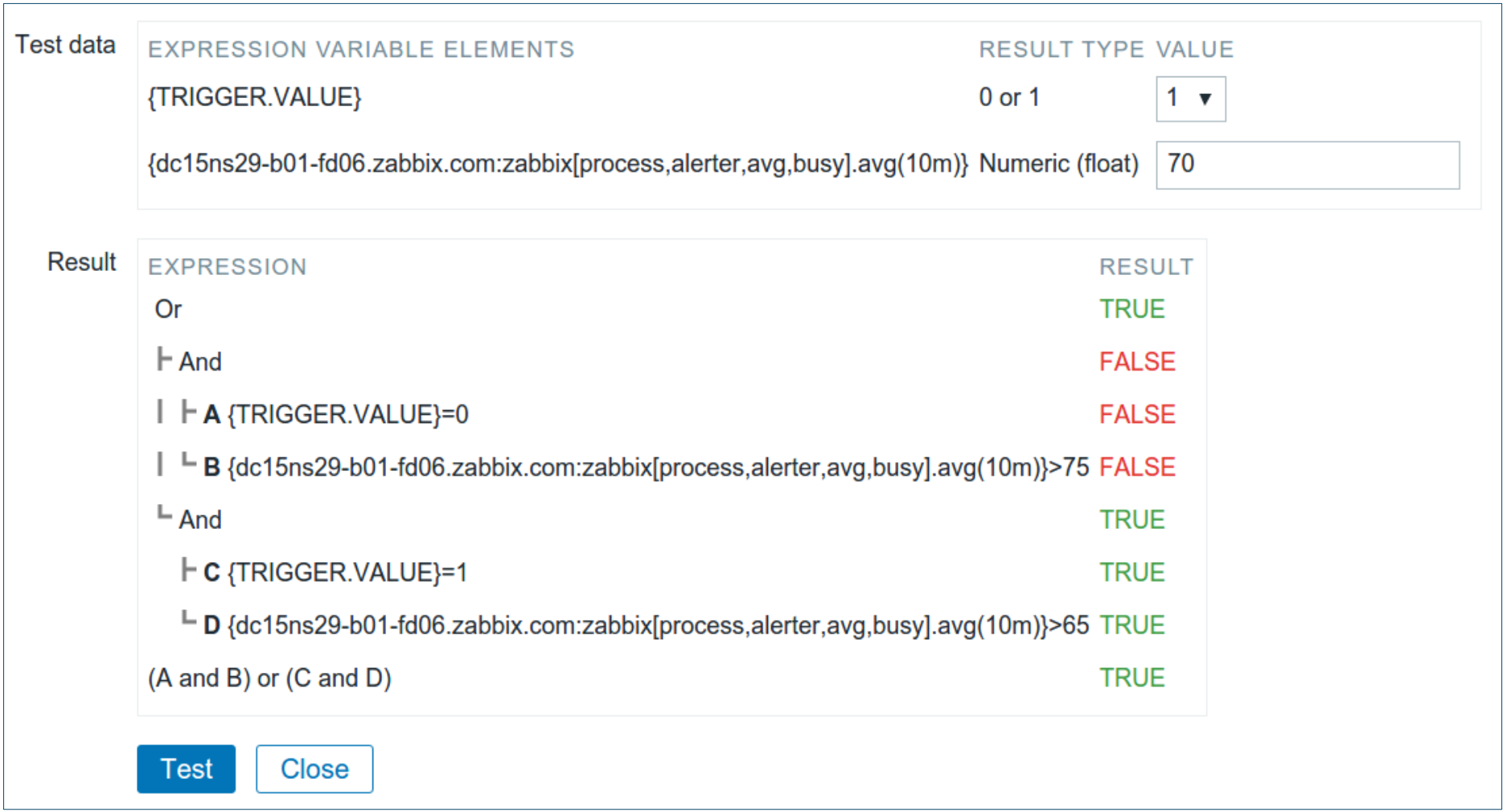
Classic



Constructor



EXPRESSION TESTING



TIPS AND TRICKS

No data for a period of time

No ping from agent within 5 minutes:
{host:agent.ping.nodata(5m)}=1

"Fuzzy" trigger, when? 

Server is unreachable

After trigger expression update

If Zabbix is unable to calculate trigger expression

Check for time difference

fuzzytime() function

Macros for trigger name

{HOST.NAME}

USER MACRO FUNCTIONALITY

Easier maintenance – one template and:

different item key parameters

net.tcp.service[ssh,{$SSH_PORT}]

different trigger expression values

{server:system.cpu.load[,avg1].last(0)} > {$CPU_LOAD}

Overwrites upstream – priority:

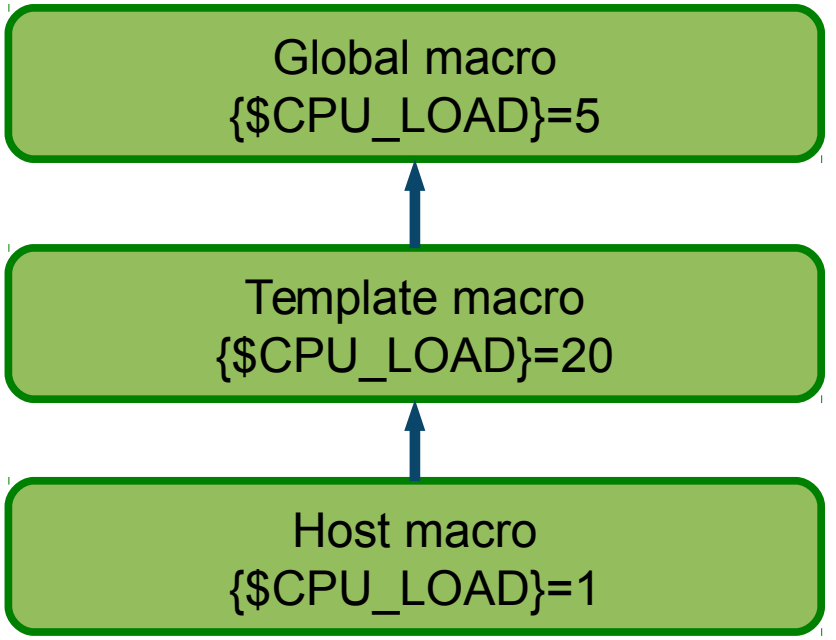
Host macro

Template macro

Global macro

Syntax:

{$NAME}



DEPENDENCIES BETWEEN TRIGGERS

Avoid notifications

Define dependencies between

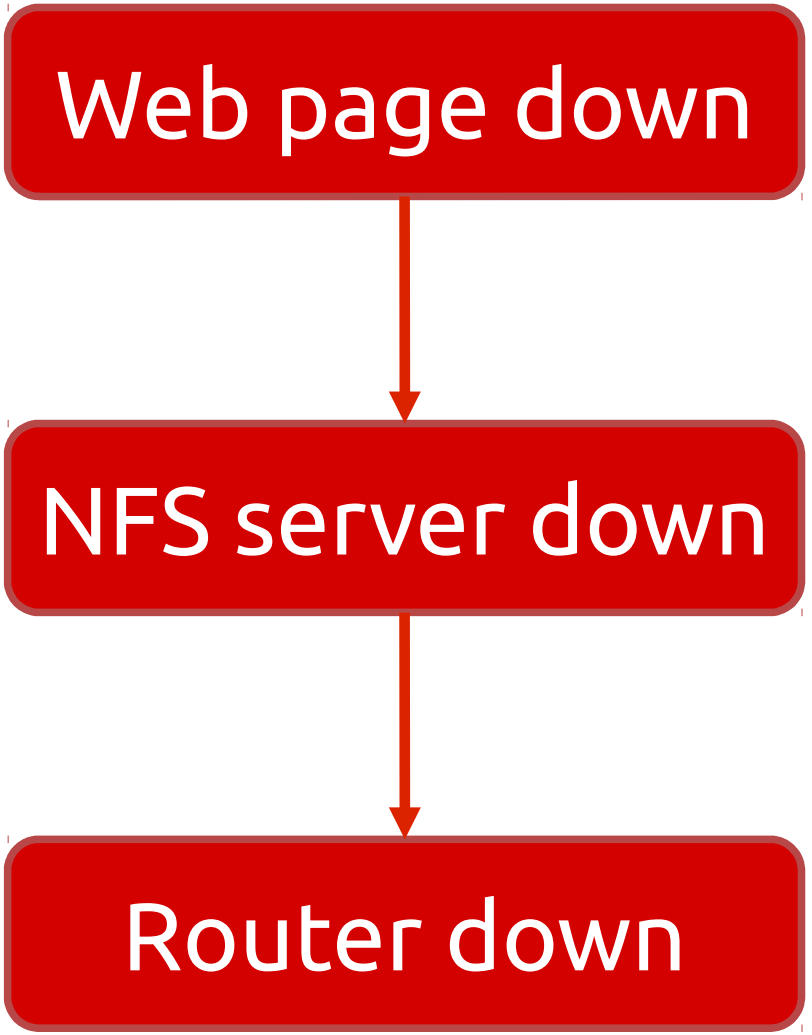
- Network devices
- Applications
- Resources
- … anything

Multiple levels

- Host → Switch 1 → Switch 2

Multiple dependencies

- Host → Router1
- Host → Router2



PRACTICAL SETUP

Create a new trigger on the host:

"CPU load too high on <macro>"

Use macro in the trigger name to display the host name

Use "cat /dev/urandom | md5sum" command to test it

STATUS OF TRIGGERS

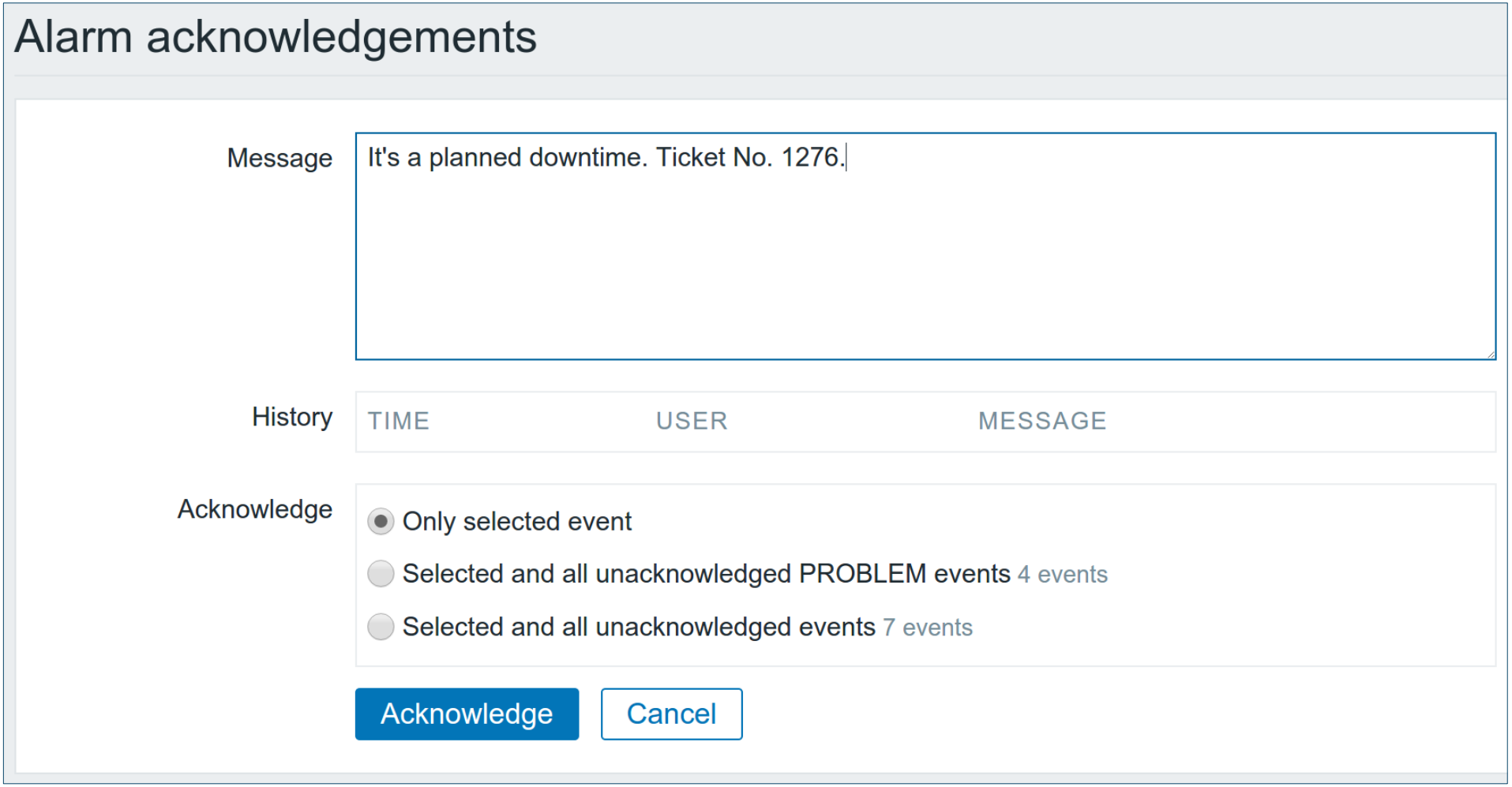
- Shows status of triggers and events
- Events can be acknowledged
- Can filter by min severity, age, name, application and host inventory

- Triggers can be expanded to show events
- If trigger has an active dependency it is not shown

SEVERITY	STATUS	INFO	LAST CHANGE ▼	AGE	ACK	HOST	NAME	DESCRIPTION
Average	PROBLEM		2016-01-26 19:05:35	15m 48s	Yes	dc15ns29-b01-fd06.zabbix.com	Zabbix discoverer processes more than 75% busy	Add
Warning	PROBLEM		2016-01-25 11:00:08	1d 8h 21m	No 11	SNMP host (Linux)	Free disk space is less than 20% on volume Shared memory	Add
Average	PROBLEM		2016-01-06 19:04:00	20d 17m	No 1	Windows Server	Zabbix agent on Windows Server is unreachable for 5 minutes	Add
Average	PROBLEM		2015-11-03 09:18:00	2m 24d 10h	Yes	Zabbix Java gateway	Zabbix Java gateway is not reachable	Add
Warning	PROBLEM		2015-09-22 13:24:30	4m 6d 6h	No 1	dc15ns29-b01-fd06.zabbix.com	Lack of free swap space on dc15ns29-b01-fd06.zabbix.com	Show

Displaying 5 of 5 found

BULK ACKNOWLEDGMENT



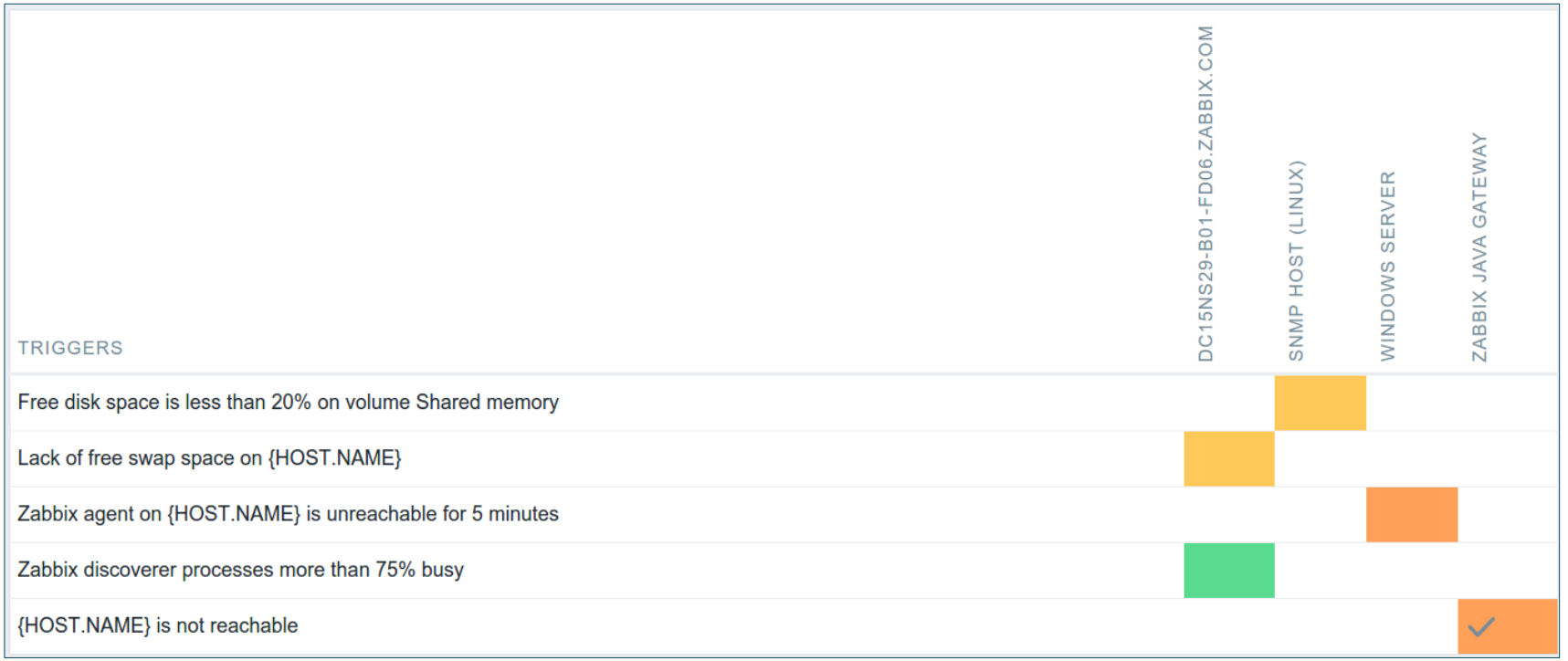
OVERVIEW

Status of a group of servers

Different colors for different trigger severities

Blinking on change

Quick navigation to Events and Graphs



MORE TRIGGER EXAMPLES

CPU load is too high and less than 100 users online:
{host:system.cpu.load.last()}>5 and {host:users.last()}<100

The "passwd" file was changed:
{host:vfs.file.cksum[/etc/passwd].diff()}>0

Someone is downloading a large file from the Internet
{host:net.if.in[eth0,bytes].min(5m)}>512M

QUESTIONS?

ZABBIX

The Enterprise class Monitoring Solution for Everyone

WWW.ZABBIX.COM

TIME FOR A BREAK :)