



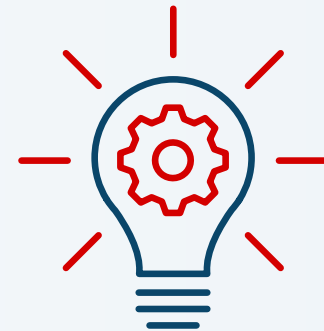
Zabbix 7.0 Performance Tuning Best Practices

Kārlis Saliņš

Technical Support Engineer

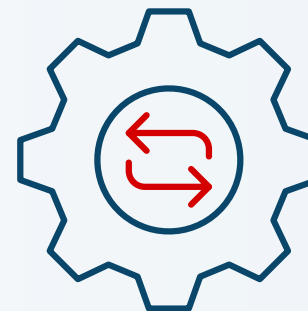
New things in Zabbix 7.0

- ▶ New processes
- ▶ Adjusted health monitoring
- ▶ Performance improvements
- ▶ Proxy load balancing



New processes

- ▶ Asynchronous processes
 - SNMPAgentPoller
 - AgentPoller
 - HTTPAgentPoller
 - Discoverer*
- ▶ MaxConcurrentChecksPerPoller
 - Maximum 1000 threads per worker
 - OS configuration required



*Discoverer process is changed, however, network discovery is still performed

```
#Edit service file
systemctl edit zabbix-server(or proxy)
#Add these lines, numbers can be different
[Service]
LimitNOFILE=100000
TasksMax=32768
#Reload systemctl daemon
systemctl daemon-reload
#Restart the service
systemctl restart zabbix-server(or proxy)
```

Increasing file descriptor amount

- ▶ Modification of service files is required
 - Zabbix server
 - Zabbix proxy

SNMPAgentPoller

- ▶ Allows asynchronous monitoring of SNMP devices
- ▶ Most default templates have been converted to use SNMPAgentPoller
- ▶ Custom templates need to be adjusted
- ▶ Works only with certain OID syntax
 - `walk[oid,oid,oid...]`
 - `get[oid]`



Converting a custom template

- Instead of discovery rule, a new walk item needs to be created

Discovery rule Preprocessing LLD macros Filters Overrides

* Name CPU discovery

Type SNMP agent ▼

* Key cpu.discovery

* SNMP OID ? discovery[{{#SNMPVALUE}},1.3.6.1.4.1.9.9.109.1.1.1.8]

* Update interval 1h



Item Tags Preprocessing

* Name CPU walk

Type SNMP agent ▼

* Key cpu.walk Select

Type of information Text ▼

* SNMP OID ? walk[1.3.6.1.4.1.9.9.109.1.1.1.8]

Converting a custom template

- Discovery rule needs to be converted to dependent discovery rule

Discovery rule Preprocessing LLD macros Filters Overrides

* Name CPU discovery

Type SNMP agent ▼

* Key cpu.discovery

* SNMP OID ? discovery[{#SNMPVALUE},1.3.6.1.4.1.9.9.109.1.1.1.8]

* Update interval 1h



Discovery rule Preprocessing LLD macros Filters Overrides

* Name CPU discovery

Type Dependent item ▼

* Key cpu.discovery

* Master item Cisco ASA v by SNMP: CPU walk ✕ Select

Converting a custom template

- Discovery rule needs to be converted to dependent discovery rule

Discovery rule Preprocessing LLD macros Filters Overrides

* Name CPU discovery

Type SNMP agent ▼

* Key cpu.discovery

* SNMP OID ? discovery[{#SNMPVALUE},1.3.6.1.4.1.9.9.109.1.1.1.8]

* Update interval 1h



Discovery rule Preprocessing LLD macros Filters Overrides

* Name CPU discovery

Type Dependent item ▼

* Key cpu.discovery

* Master item Cisco ASA v by SNMP: CPU walk ✕ Select

Converting a custom template

- Use preprocessing to convert walk output to JSON on LLD rule

The screenshot shows the Zabbix configuration interface for an LLD rule, specifically the 'Preprocessing' tab. The interface is dark-themed. At the top, there are tabs for 'Discovery rule', 'Preprocessing 1', 'LLD macros', 'Filters', and 'Overrides'. The 'Preprocessing 1' tab is active. Below the tabs, there is a section for 'Preprocessing steps'. It contains a list with one item: '1: SNMP walk to JSON'. To the right of this list is a table for 'Parameters'. The table has four columns: 'Field name', 'OID prefix', 'Format', and 'Action'. There is one row in the table with the following values: '{#SNMPVALUE}', '1.3.6.1.4.1.9.9.10', 'Unchanged', and a dropdown arrow. To the right of the 'Action' column is a 'Remove' button. Below the table is an 'Add' button. At the bottom of the interface are five buttons: 'Update', 'Clone', 'Test', 'Delete', and 'Cancel'.

Preprocessing steps	Name	Parameters								
1:	SNMP walk to JSON	<table border="1"><thead><tr><th>Field name</th><th>OID prefix</th><th>Format</th><th>Action</th></tr></thead><tbody><tr><td>{#SNMPVALUE}</td><td>1.3.6.1.4.1.9.9.10</td><td>Unchanged</td><td>Remove</td></tr></tbody></table>	Field name	OID prefix	Format	Action	{#SNMPVALUE}	1.3.6.1.4.1.9.9.10	Unchanged	Remove
Field name	OID prefix	Format	Action							
{#SNMPVALUE}	1.3.6.1.4.1.9.9.10	Unchanged	Remove							

Buttons: Add, Update, Clone, Test, Delete, Cancel

Converting a custom template

- Convert item prototypes to dependent items which will take values from walk item

Item prototype Tags 1 Preprocessing

* Name CPU [#{SNMPINDEX}] Utilization

Type SNMP agent

* Key cpu.util[#{SNMPINDEX}] Select

Type of information Numeric (float)

* SNMP OID ? 1.3.6.1.4.1.9.9.109.1.1.1.8.#{SNMPINDEX}



Item prototype Tags 1 Preprocessing

* Name CPU [#{SNMPINDEX}] Utilization

Type Dependent item

* Key cpu.util[#{SNMPINDEX}] Select

Type of information Numeric (float)

* Master item Cisco ASAv by SNMP: CPU walk X Select Select prototype

Converting a custom template

- ▶ Use preprocessing to extract specific values from the master item on the item prototype

Item prototype

[Item prototype](#) [Tags 1](#) [Preprocessing 1](#)

Preprocessing steps ?

	Name	Parameters
1:	SNMP walk value	1.1.1.8.{#SNMPINDEX}
		Unchanged

Add

Type of information Numeric (float)

Converting a custom template

- Use preprocessing to limit LLD rule execution time

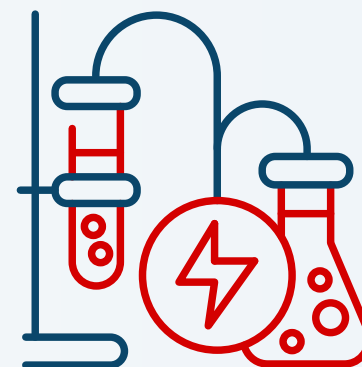
Discovery rule Preprocessing 2 LLD macros Filters Overrides

Preprocessing steps	Name	Parameters												
1:	SNMP walk to JSON	<table><thead><tr><th>Field name</th><th>OID prefix</th><th>Format</th><th>Action</th></tr></thead><tbody><tr><td>{#SNMPVALUE}</td><td>1.3.6.1.4.1.9.9.109</td><td>Unchanged</td><td>Remove</td></tr><tr><td colspan="4">Add</td></tr></tbody></table>	Field name	OID prefix	Format	Action	{#SNMPVALUE}	1.3.6.1.4.1.9.9.109	Unchanged	Remove	Add			
Field name	OID prefix	Format	Action											
{#SNMPVALUE}	1.3.6.1.4.1.9.9.109	Unchanged	Remove											
Add														
2:	Discard unchanged with heartbeat	12h												
Add														

Proxy memory buffer

Available proxy memory buffer methods (**ProxyBufferMode**):

- ▶ «disk»
 - All data gets stored in DB
 - Default for old environments after upgrade
- ▶ «memory»
 - All data gets stored in memory (RAM)
 - No protection against data loss
- ▶ «hybrid»
 - **Recommended**
 - Uses memory in most cases
 - Data loss protection using DB
 - Default for new installations



Proxy load balancing

- ▶ Proxy groups managed by Zabbix server
- ▶ Auto rebalancing of hosts
- ▶ In case of issues, host gets automatically assigned to a working proxy



Proxy LLD by Zabbix server

Zabbix server health template:

- ▶ Does LLD of proxies that are connected to the server
- ▶ Creates basic items and triggers
- ▶ Shows various statistics from proxies

*Remember to retrieve the latest health templates after upgrading to a new major Zabbix version!

```
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: Certificate
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: Compatibility
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: Compression
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: Host count
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: Item count
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: Last seen, in seconds
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: Mode
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: PSK
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: Required VPS
... Zabbix proxy discovery: Zabbix proxies stats: Proxy [zabbix-proxy]: Stats
... Zabbix proxy discovery: Proxy [zabbix-proxy]: Stats: Proxy [zabbix-proxy]: Unencrypted
```

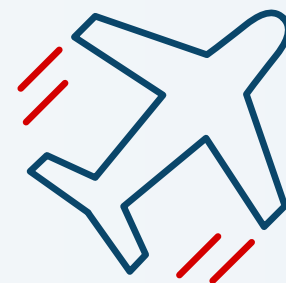
Configuration updates

- ▶ Configuration updates are now **incremental**
- ▶ Zabbix server reloads configuration every 10s
- ▶ Zabbix proxy reloads configuration every 10s
- ▶ Active agent reloads configuration every 5s

Version/Component	Zabbix server	Zabbix proxy	Active agent
6.0	1 minute	1 hour	2 minutes
7.0	10 seconds	10 seconds	5 seconds

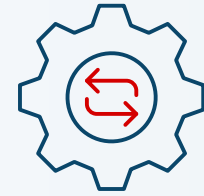
Performance improvements

- ▶ Faster frontend permission checks
- ▶ Immediate maintenance
- ▶ Faster trigger execution
- ▶ Logging of SNMP v3 duplicate engine ID
- ▶ Configuration file validation
- ▶ Item timeouts

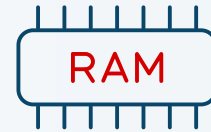


Zabbix internal process tuning

► Workers



► Caches



► Managers



Tuning workers

- ▶ Most worker usage needs to be between 40-60%
- ▶ Zabbix server and proxy have almost the same set of workers

Alerter

Escalator

HTTP poller

History syncer

History poller

Timer

Text

LLD worker

Text

Java poller

Text

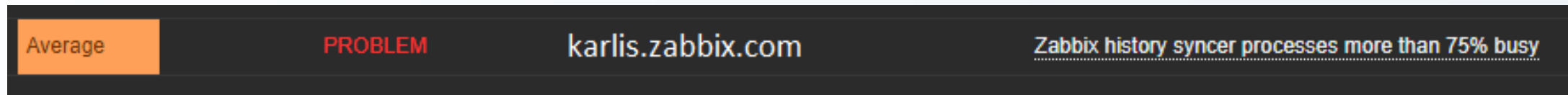
Poller

History syncers

- ▶ Writes data into the database
- ▶ Calculates triggers
- ▶ 1 History syncer can deal with ~1000 NVPS



History syncers



Causes:

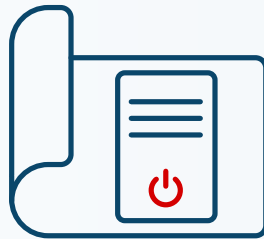
- ▶ Data cannot be written in the DB fast enough
- ▶ Lots of triggers to be calculated
- ▶ Most often DB related

Fixes:

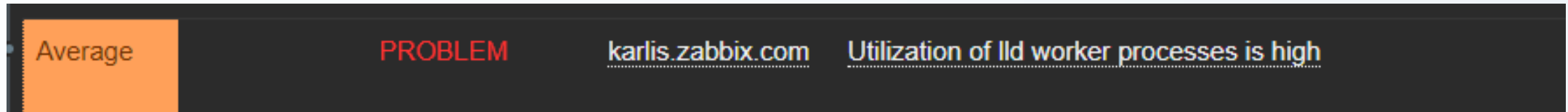
- ▶ Tune and check DB performance
- ▶ Check triggers
- ▶ Improve hardware
- ▶ Increase history syncer amount (1 history syncer = 1000 NVPS!)

LLD workers

- ▶ Performs low-level discovery
- ▶ High impact on DB performance
- ▶ Only on Zabbix server
- ▶ Usually, best not to increase above the default amount



LLD workers



Causes:

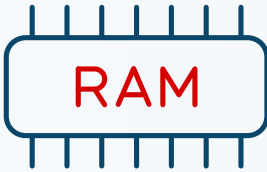
- ▶ Frequent execution of low-level discovery rules
- ▶ Most often DB related

Fixes:

- ▶ Tune and check DB performance
- ▶ Increase update interval for low-level discovery rules
- ▶ Increase “Discard unchanged with heartbeat” period for dependent low-level discovery rules
- ▶ Increase LLD workers (Every LLD worker causes huge load on DB!)

Tuning caches

- ▶ Most cache usage should be between 40% and 60%
- ▶ Zabbix server and proxy have almost the same set of caches



Value cache

Configuration cache

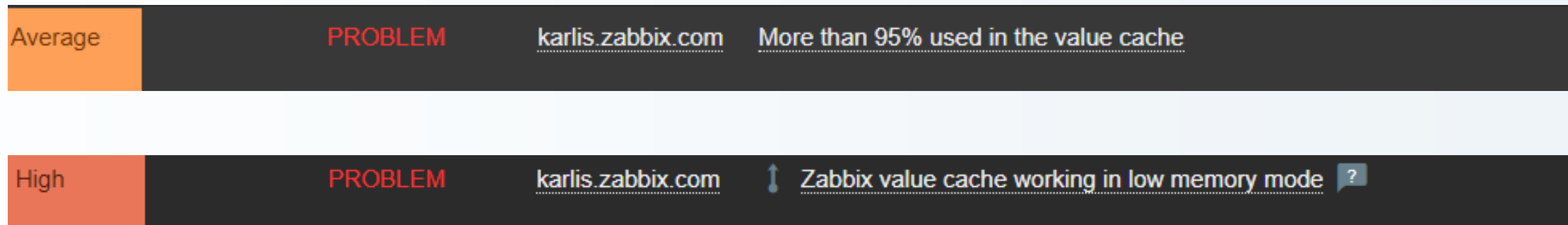
Trend cache

Value cache

- ▶ Stores values for easier accessibility
- ▶ Is used for trigger calculation
- ▶ Is used for calculated items
- ▶ Almost limitless
- ▶ Never should be full!



Value cache



Causes:

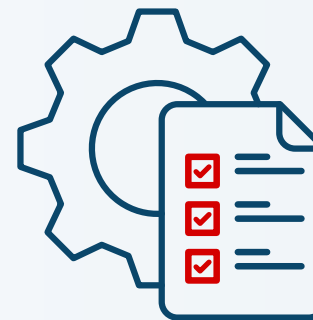
- ▶ Lots of triggers
- ▶ Lots of functions that use forecasting
- ▶ Lots of calculated items

Fixes:

- ▶ Increase ValueCacheSize (Make sure there is enough RAM on the server!)
- ▶ Adjust forecast/trigger periods

Configuration cache

- ▶ Stores configuration
- ▶ Almost limitless
- ▶ Never should be full, if it gets full, Zabbix server/proxy crashes!



Configuration cache

Average

PROBLEM

karlis.zabbix.com

More than 75% used in the configuration cache

Causes:

- ▶ Lots of hosts, items, triggers

Fixes:

- ▶ Increase CacheSize (Make sure there is enough RAM on the server!)

History and history index cache

History cache

- ▶ Stores values after they are preprocessed
- ▶ History syncer takes the values from this cache and writes them to the database
- ▶ 2 GB limit
- ▶ Never should be full and should be as empty as possible!

History index cache

- ▶ Indexes history cache
- ▶ $\frac{1}{4}$ size of history cache

History and history index cache

Average	PROBLEM	karlis.zabbix.com	More than 75% used in the history cache
Average	PROBLEM	karlis.zabbix.com	More than 75% used in the history index cache

Causes:

- ▶ Data cannot be written in the DB fast enough
- ▶ Most of the time issues arise together with history syncer

Fixes:

- ▶ Tune and check DB performance
- ▶ Increase HistoryCacheSize and HistoryIndexCacheSize(Make sure there is enough RAM on the server - this only applies if your DB is big)

Tuning other processes and caches

- ▶ Most process worker and cache usage should be maintained between 40% and 60%
- ▶ If the default settings are being used, it's acceptable for the usage to be lower
- ▶ There is no “best” configuration, every Zabbix environment needs to be tuned individually



Tuning managers

- ▶ Manager processes cannot be tuned!
- ▶ High manager usage typically happens when worker usage grows
- ▶ Needs to be checked what is causing the issues – diaginfo, reading log files, etc.



Queue

- ▶ Values that have not yet arrived
- ▶ Fixable
- ▶ Many different reasons why it appears

5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
0	1	0	1	0	24663
0	0	0	0	0	22919
0	0	0	1	0	28858
0	1	0	7	11	8998
0	0	0	3	7	113535
0	0	0	0	0	108601
0	0	0	2	8	137555
0	0	0	1	0	103860
0	0	0	0	0	14893
0	1	0	0	0	1024

How to

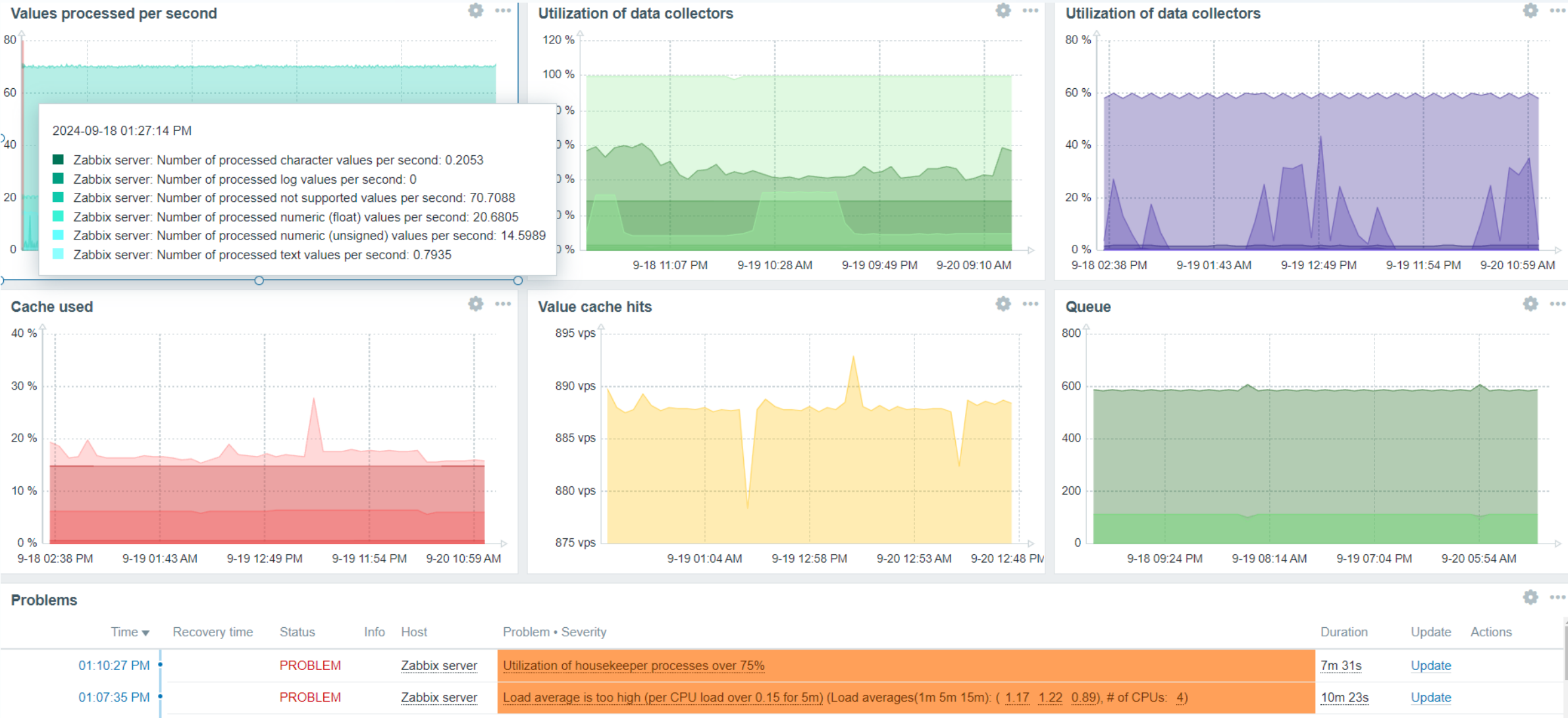
- ▶ Use **built-in health templates**
- ▶ Dashboards of server/proxy performance
- ▶ Monitor OS metrics of server/proxy using Zabbix agent
- ▶ Remember to update the templates when upgrading to next major release

Name ▲	Hosts	Items	Triggers	Graphs	Dashboards	Discovery	Web	Vendor
Remote Zabbix proxy health	Hosts	Items 48	Triggers 35	Graphs 5	Dashboards 1	Discovery	Web	Zabbix
Remote Zabbix server health	Hosts	Items 67	Triggers 49	Graphs 11	Dashboards 2	Discovery 2	Web	Zabbix
Zabbix proxy health	Hosts 2	Items 48	Triggers 34	Graphs 5	Dashboards 1	Discovery	Web	Zabbix
Zabbix server health	Hosts 1	Items 67	Triggers 48	Graphs 11	Dashboards 2	Discovery 2	Web	Zabbix

Set up proxy monitoring

1. Set up the proxy
2. Create a host on the frontend
3. Set up the host to be monitored by itself
4. Link template «Zabbix proxy health»
5. Link Zabbix agent template (optional)

Health monitoring



Tuning frontend

- ▶ Frontend can become slow and unresponsive over time
- ▶ Lots of improvements in Zabbix 7.0
- ▶ Issues can be caused by web server or the DB



Tuning frontend

- ▶ Enable debug mode!
- ▶ Debug can be done for the whole page or individual widgets

```
Total time: 15.06485  
Total SQL time: 0.010517  
SQL count: 36 (selects: 35 | executes: 16)  
Peak memory usage: 4M  
Memory limit: 256M
```

Web server issue

```
Total time: 15.06485  
Total SQL time: 15.010517  
SQL count: 36 (selects: 35 | executes: 16)  
Peak memory usage: 4M  
Memory limit: 256M
```

Database issue

Tuning frontend

- ▶ Web server configuration can be changed
- ▶ /etc/php-fpm.d/zabbix.conf

```
pm = dynamic
pm.max_children = 50
pm.start_servers = 5
pm.min_spare_servers = 5
pm.max_spare_servers = 35
pm.max_requests = 200

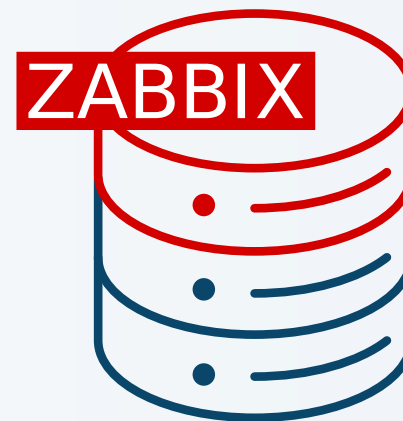
php_value[session.save_handler] = files
php_value[session.save_path] = /var/lib/php/session

php_value[max_execution_time] = 300

php_value[memory_limit] = 256M
php_value[post_max_size] = 16M
php_value[upload_max_filesize] = 20M
php_value[max_input_time] = 300
php_value[max_input_vars] = 10000
```

Tuning databases

- ▶ MySQL tunable parameters
 - `innodb_flush_log_at_trx_commit = 0`
 - `innodb_flush_method = O_DIRECT`
 - `optimizer_switch=index_condition_pushdown=off`
 - `innodb_buffer_pool_size=(75-80% of RAM if standalone DB or 60% if shared with the proxy)`
- ▶ PostgreSQL tunable parameters
 - Use PG Tune (<https://pgtune.leopard.in.ua/>)



Tuning triggers

- ▶ Huge amount of trigger changes in a short time period creates lots of events
- ▶ Lots of events = load on DB
- ▶ Regularly observe Reports > Top 100 triggers or dashboard widget

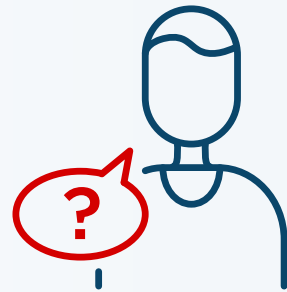
Host	Trigger	Severity	Number of problems
Karlis server	High CPU Load	Average	2465
Karlis laptop	Memory usage is too high	Average	1256
Karlis laptop	Windows requires new updates	Warning	657

Tuning triggers

- ▶ Use proper functions
- ▶ Adjust time intervals
- ▶ Alert fatigue
- ▶ LLD overrides allow to not discover triggers based on regexp
- ▶ Recovery expression
- ▶ Limit nodata() usage

Tuning templates

- ▶ Default templates are good, however, require attention
- ▶ Switch old SNMP templates to use asynchronous checks
- ▶ Think about update interval



The ZABBIX logo is displayed in white uppercase letters within a red rectangular box. The background of the slide features a dark blue globe with a network of glowing blue lines and dots, suggesting a global network or data flow.

ZABBIX

Thank you!

Kārlis Saliņš

Technical Support Engineer