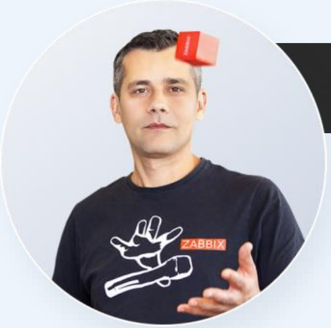
ZABBIX SUMMIT

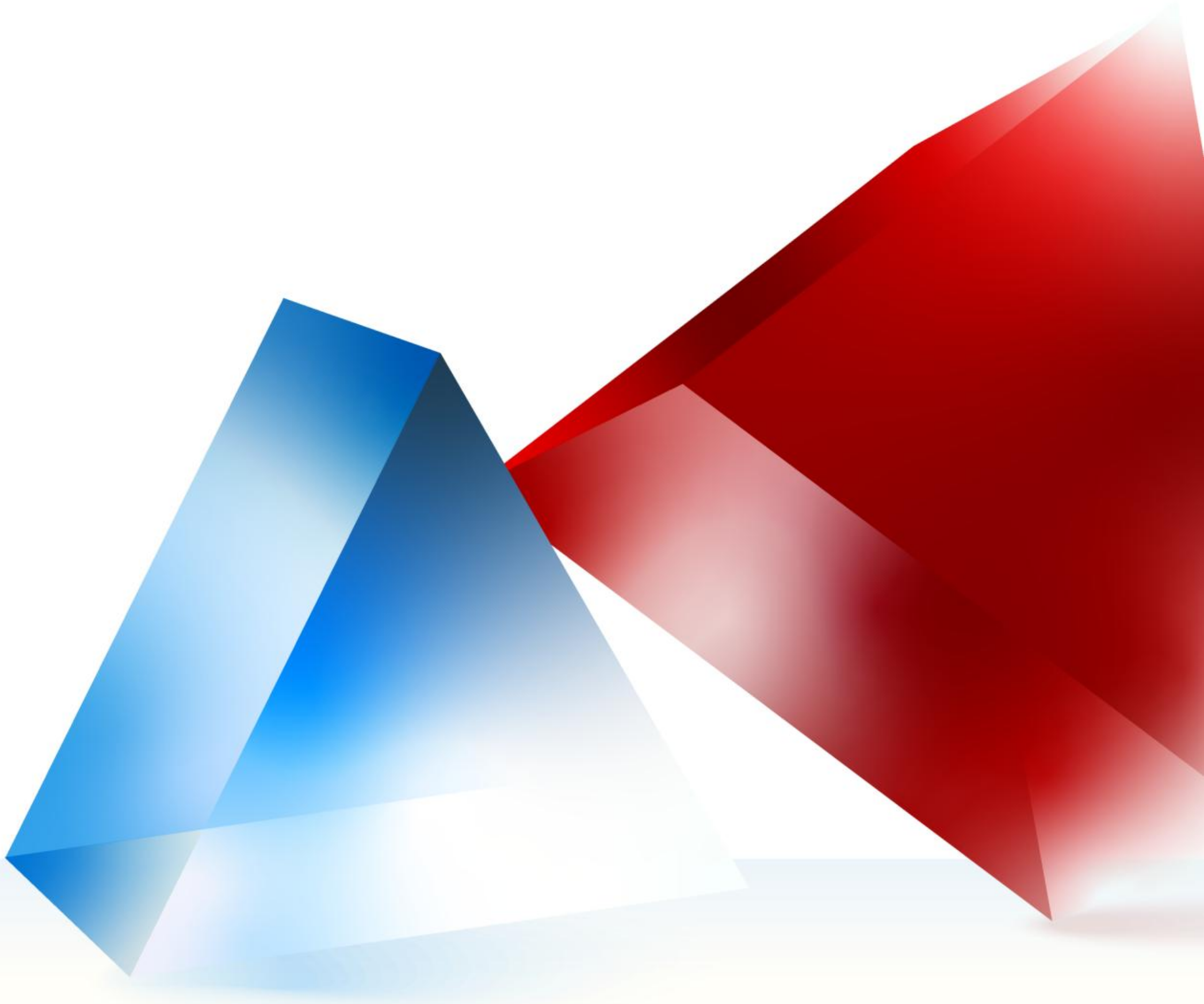
Data Collecting and Preprocessing Internals



Kaspars Mednis

Training Project Manager

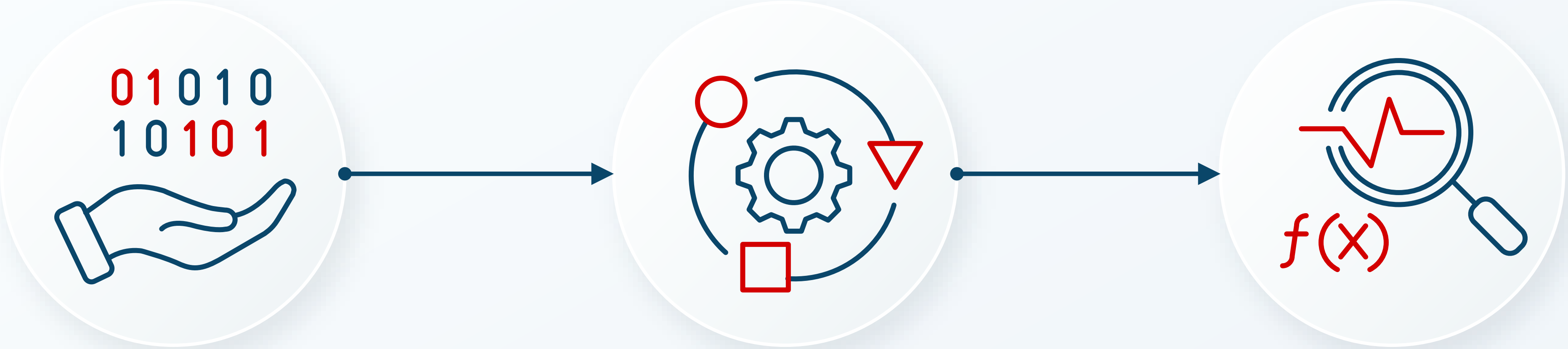
How it works



Zabbix data flow

The Zabbix data flow consists of multiple stages:

- Raw data collection using Hosts and Items
- Data preprocessing using Preprocessing steps
- Problem detection using Triggers



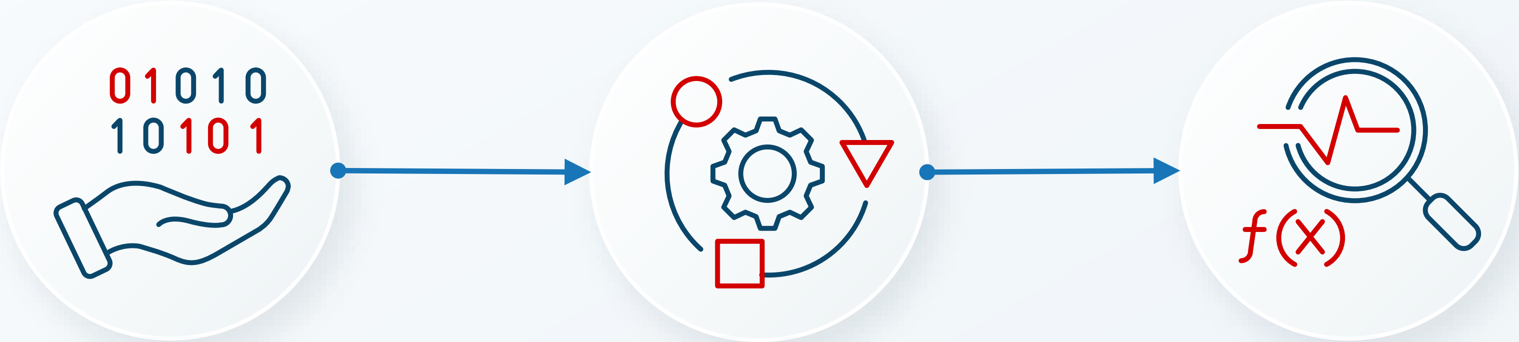
MAJOR IMPROVEMENTS

Prior to 7.0.17 / 7.4.2, all collected data was sent through preprocessing:

- This increased the load on the preprocessing manager
- Preprocessing performance issues affected every collected metric

Legend

All collected values



MAJOR IMPROVEMENTS

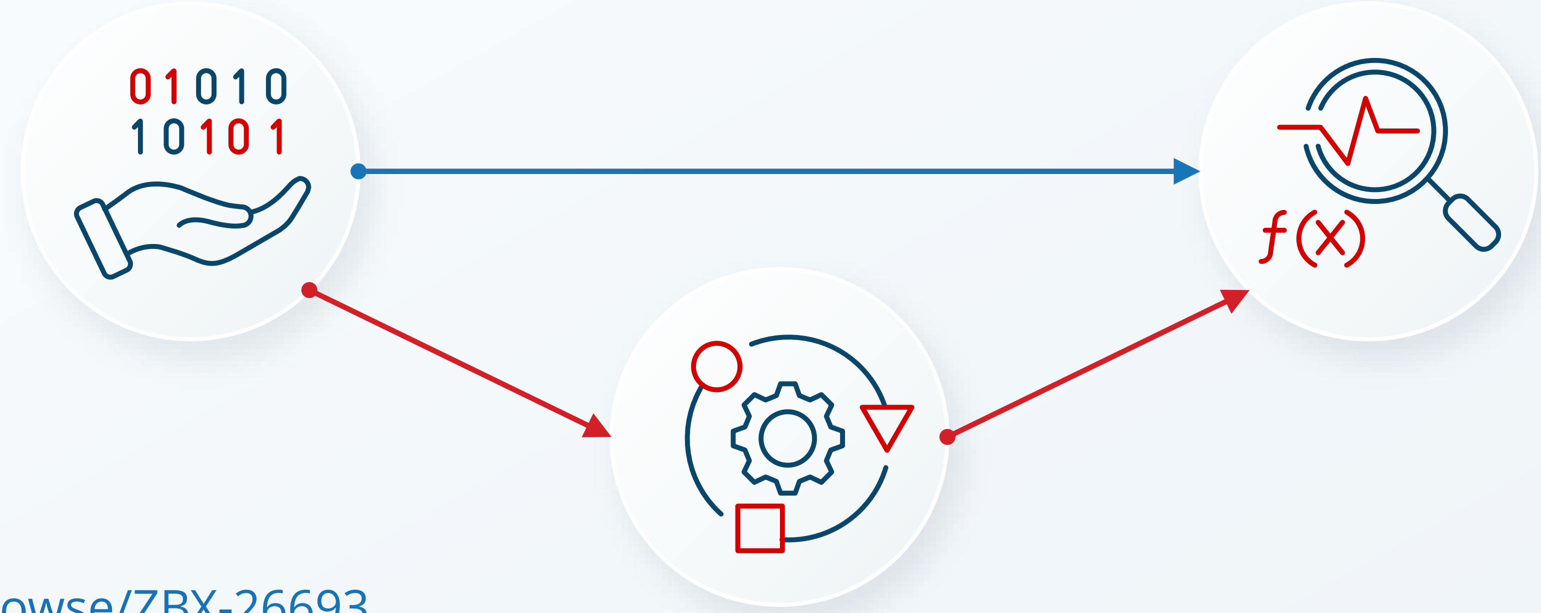
Starting from Zabbix 7.0.17 / 7.4.2 an improved logic was implemented:

- Only data with preprocessing steps is sent to the preprocessing manager
- Keep in mind that all dependent items use preprocessing

Legend

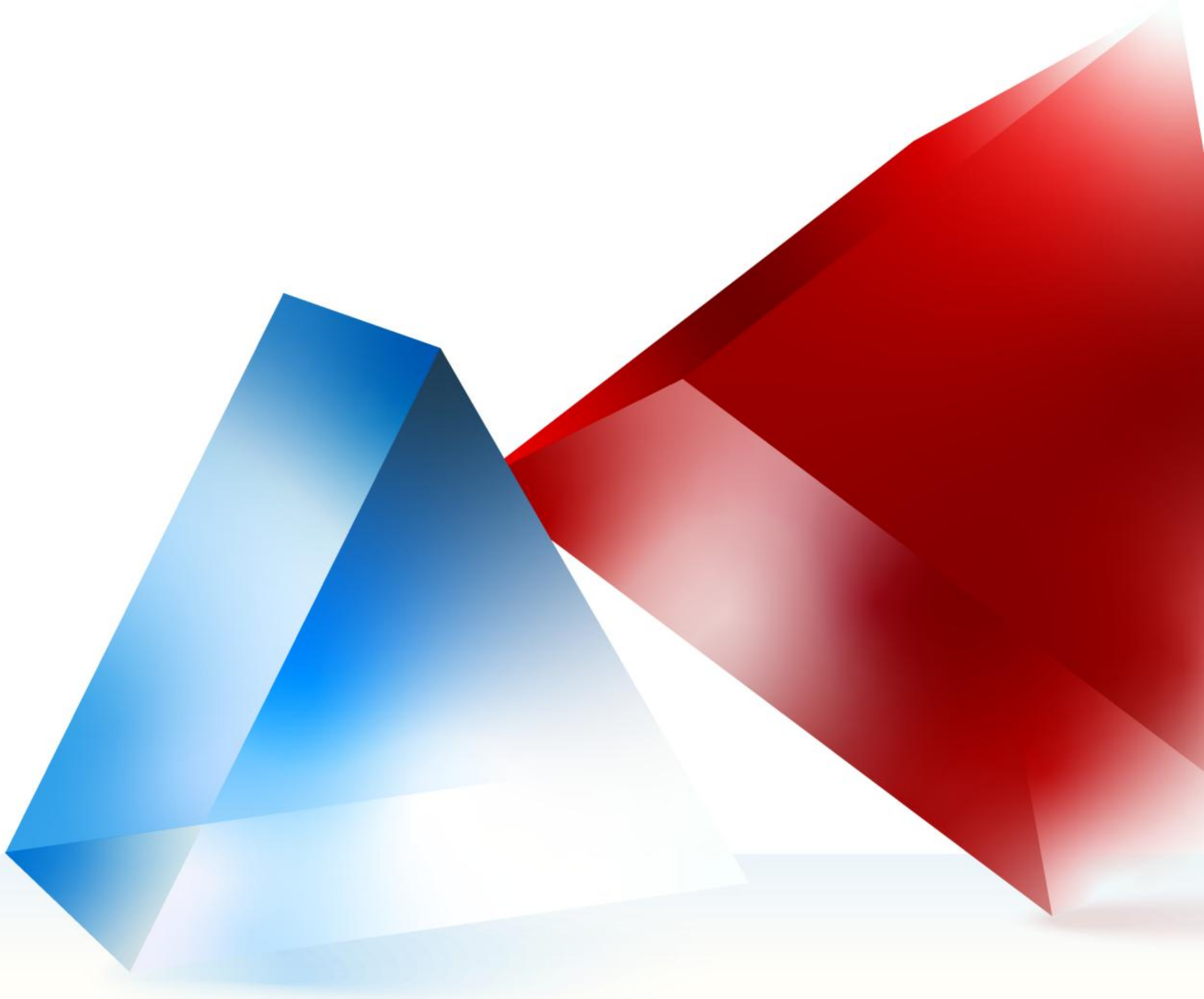
→ Data with preprocessing steps

→ Data without preprocessing steps



https://support.zabbix.com/browse/ZBX-26693

Data collection



DATA COLLECTION

Data collection in Zabbix is implemented using:

- Hosts represents physical device, logical device, or group of metrics
- Items represents individual metrics

Each item has its own individual parameters:

- Item type
- Update interval
- Timeout
- History storage period

ITEM TYPES

Zabbix items have 19 unique types:

0	Zabbix agent
2	Zabbix trapper
3	Simple check
5	Zabbix internal
7	Zabbix agent (active)
9	Web item
10	External check
11	Database monitor
12	IPMI agent
13	SSH agent
14	TELNET agent
15	Calculated
16	JMX agent
17	SNMP trap
18	Dependent item
19	HTTP agent
20	SNMP agent
21	Script
22	Browser

DATA COLLECTORS

Items are collected by data collector processes:

- Each item type is collected by its own data collector
- Some item types share the same data collector

Data collectors can be synchronous or asynchronous:

- Only Agent poller, HTTP poller, and SNMP poller are asynchronous data collectors
- All other data collectors are synchronous

SYNCHRONOUS VS ASYNCHRONOUS

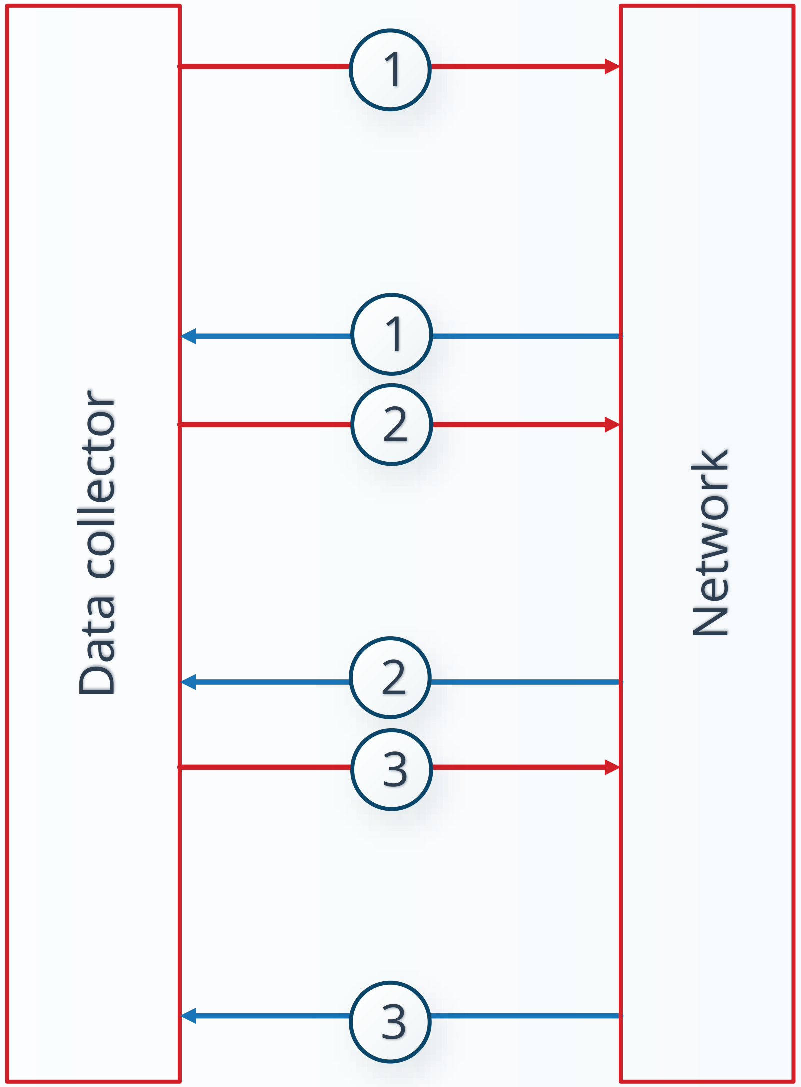
Synchronous data collectors collect one value at a time per process:

- Each data collector is a separate process
- Lots of processes increase CPU and memory usage

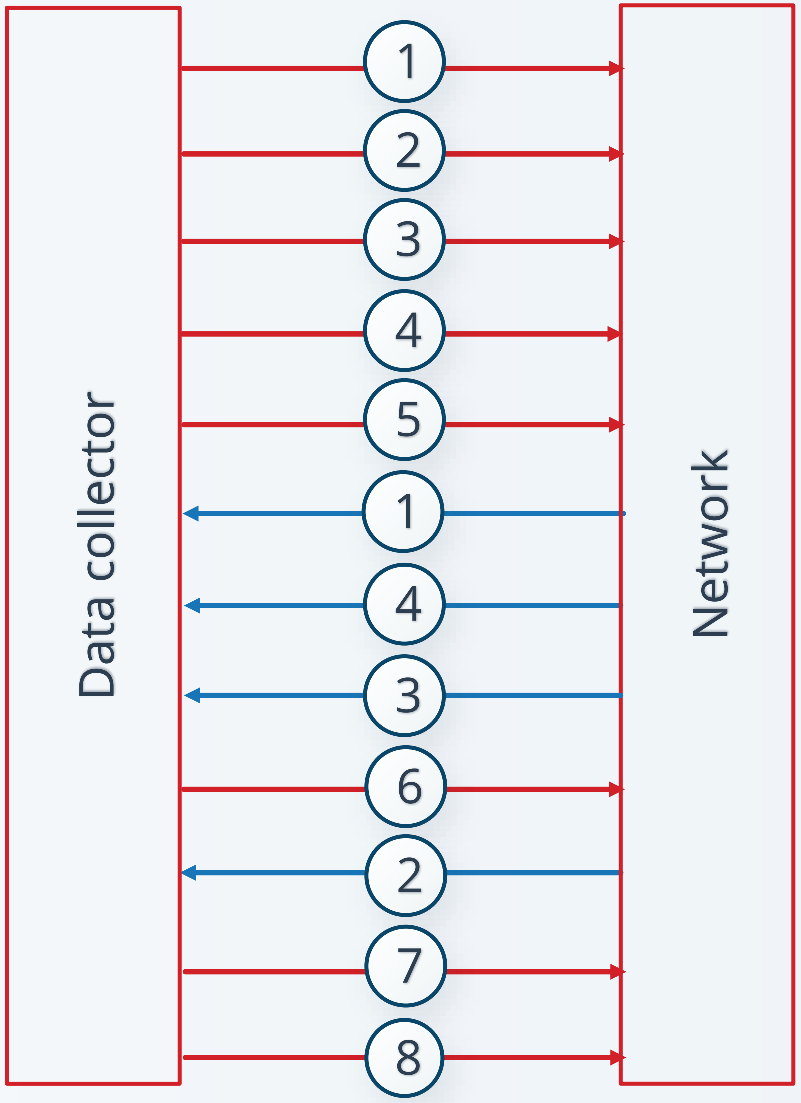
Asynchronous data pollers collect up to 1,000 values at a time per instance:

- Typically, only one to a few data collectors of each type are required
- Each data collector uses as many asynchronous threads as required

Synchronous data collection



Asynchronous data collection



SYNCHRONOUS DATA COLLECTORS

Each data collector has its own configuration parameter:

- Increase the number of collectors as the number of corresponding items increases
- Do not start unnecessary data collectors

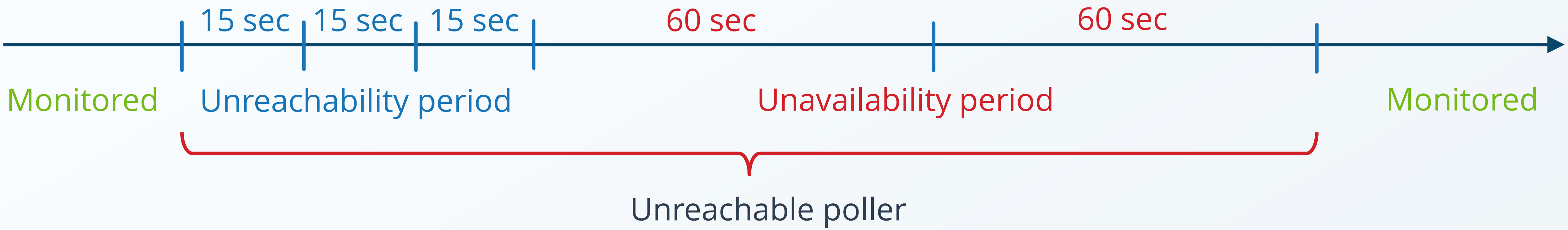
Use Zabbix server / proxy configuration files to tune:

```
### Option: StartPollers
#       Number of pre-forked instances of pollers.
# Mandatory: no
# Range: 0-1000
# Default: 5
StartPollers=20
```

UNREACHABLE POLLERS

Synchronous data collectors utilize Unreachable poller for unavailable hosts:

- Only a limited amount of data collectors are available for data collection
- Unreachable pollers will try to collect data from hosts which have connectivity issues



15 sec 15 sec 15 sec 60 sec 60 sec

Monitored Unreachability period Unavailability period Monitored

Unreachable poller

```
### Option: StartPollersUnreachable
#       Number of pre-forked instances of pollers for unreachable hosts
# Default: 1
StartPollersUnreachable=15
```

TUNING UNREACHABLE POLLERS

Unreachable poller configuration behavior can be tuned:

- Default settings are perfectly good for most environments
- Settings can be adjusted in the server or proxy configuration file

```
### Option: UnreachableDelay
#       How often host is checked for availability
UnreachableDelay=15

### Option: UnreachablePeriod
#       After how many seconds of unreachability treat a host as unavailable.
UnreachablePeriod=45

### Option: UnavailableDelay
#       How often host is checked during the unavailability period
UnavailableDelay=60
```

ASYNCHRONOUS DATA COLLECTORS

Asynchronous data collectors are controlled by two parameters:

- Number of asynchronous pollers started
- Maximum concurrent checks per poller

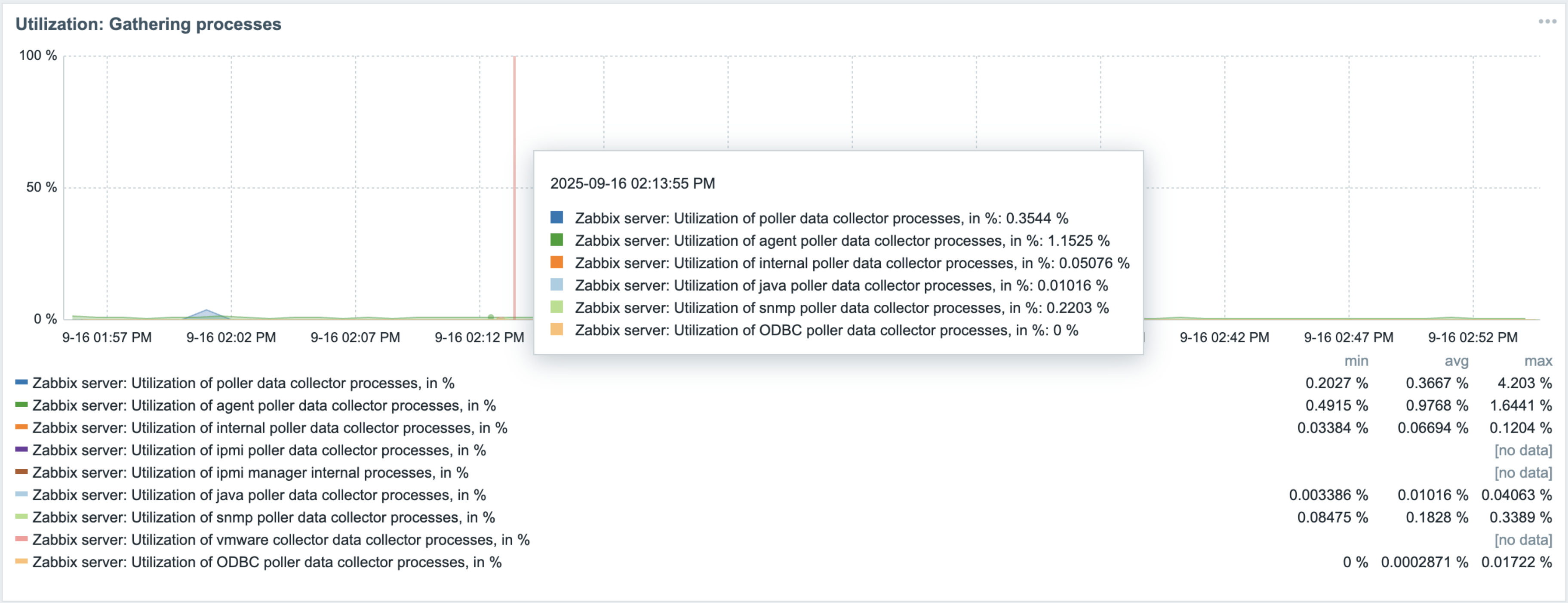
```
### Option: StartAgentPollers
#       Number of pre-forked instances of asynchronous Zabbix agent pollers.
# Mandatory: no
# Range: 0-1000
StartAgentPollers=1
```

```
### Option: MaxConcurrentChecksPerPoller
#       Maximum number of asynchronous checks that can be executed at once by each
# HTTP agent poller, agent poller or SNMP poller
# Range: 1-1000
MaxConcurrentChecksPerPoller=1000
```

PERFORMANCE TUNING

Use official Zabbix health templates to tune data collectors:

- Synchronous data collectors should be not more than 60% busy
- Asynchronous data collectors typically do not require too much fine-tuning



Utilization: Gathering processes

2025-09-16 02:13:55 PM

- ■ Zabbix server: Utilization of poller data collector processes, in %: 0.3544 %
- ■ Zabbix server: Utilization of agent poller data collector processes, in %: 1.1525 %
- ■ Zabbix server: Utilization of internal poller data collector processes, in %: 0.05076 %
- ■ Zabbix server: Utilization of java poller data collector processes, in %: 0.01016 %
- ■ Zabbix server: Utilization of snmp poller data collector processes, in %: 0.2203 %
- ■ Zabbix server: Utilization of ODBC poller data collector processes, in %: 0 %

	min	avg	max
▬ Zabbix server: Utilization of poller data collector processes, in %	0.2027 %	0.3667 %	4.203 %
▬ Zabbix server: Utilization of agent poller data collector processes, in %	0.4915 %	0.9768 %	1.6441 %
▬ Zabbix server: Utilization of internal poller data collector processes, in %	0.03384 %	0.06694 %	0.1204 %
▬ Zabbix server: Utilization of ipmi poller data collector processes, in %			[no data]
▬ Zabbix server: Utilization of ipmi manager internal processes, in %			[no data]
▬ Zabbix server: Utilization of java poller data collector processes, in %	0.003386 %	0.01016 %	0.04063 %
▬ Zabbix server: Utilization of snmp poller data collector processes, in %	0.08475 %	0.1828 %	0.3389 %
▬ Zabbix server: Utilization of vmware collector data collector processes, in %			[no data]
▬ Zabbix server: Utilization of ODBC poller data collector processes, in %	0 %	0.0002871 %	0.01722 %

DATA COLLECTORS

Asynchronous Synchronous Single instance

Zabbix agent	Agent poller	Web item	HTTP Poller
HTTP agent	HTTP agent poller	Browser	Browser poller
SNMP agent	SNMP agent poller	Calculated	History poller
SSH agent	Poller	Database monitor	ODBC poller
Simple check	Poller	IPMI agent	IPMI Poller
External check	Poller	JMX agent	JAVA poller
TELNET agent	Poller	Zabbix internal	Internal poller
Script	Poller	SNMP trap	SNMP trapper
Zabbix agent (active)	Trapper	Dependent item	*Master item*
Zabbix trapper	Trapper		

DATA COLLECTION BY PROXIES

Data collection by Zabbix proxies follows the same rules:

- Same set of data collectors is used
- Tune according to Zabbix proxy performance diagrams

Two data collectors are involved in communication between Zabbix server and proxies:

- Proxy pollers collects data from passive proxies
- Trappers collect data from active proxies

DATA COLLECTION BY AGENTS

Two generations of Zabbix agents currently are supported:

Zabbix agent	C based
Zabbix agent 2	Go based

Zabbix agent has a limited number of data collectors for passive checks:

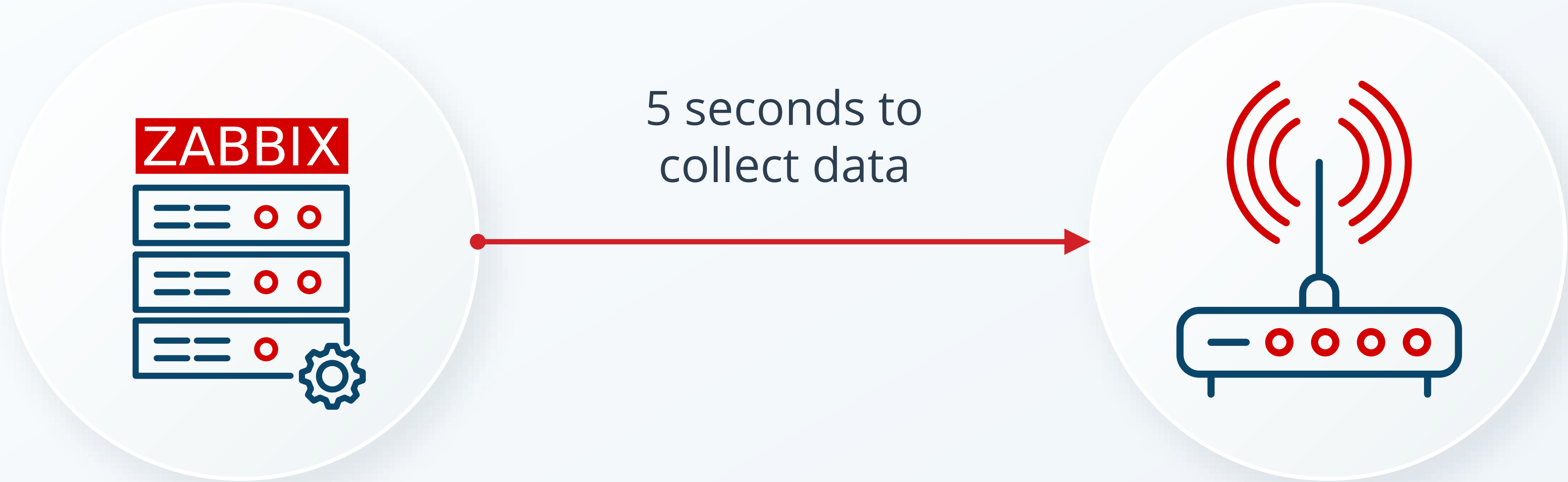
- Zabbix Agent 2 has a different architecture and does not require this tuning

```
### Option: StartAgents
#       The number of pre-forked instances of zabbix_agentd that process passive
# checks. If set to 0, passive checks are disabled and the agent will not listen on
# any TCP port.
# Range: 0-100
# Default: 10
StartAgentPollers=20
```

TIMEOUTS

Timeouts control how long Zabbix waits for data collection:

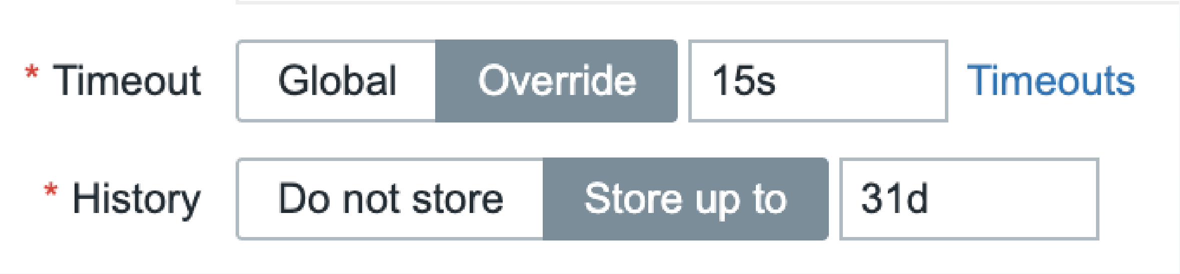
- Too short timeouts may lead to items not collecting data
- Too long timeouts may have a negative impact on performance



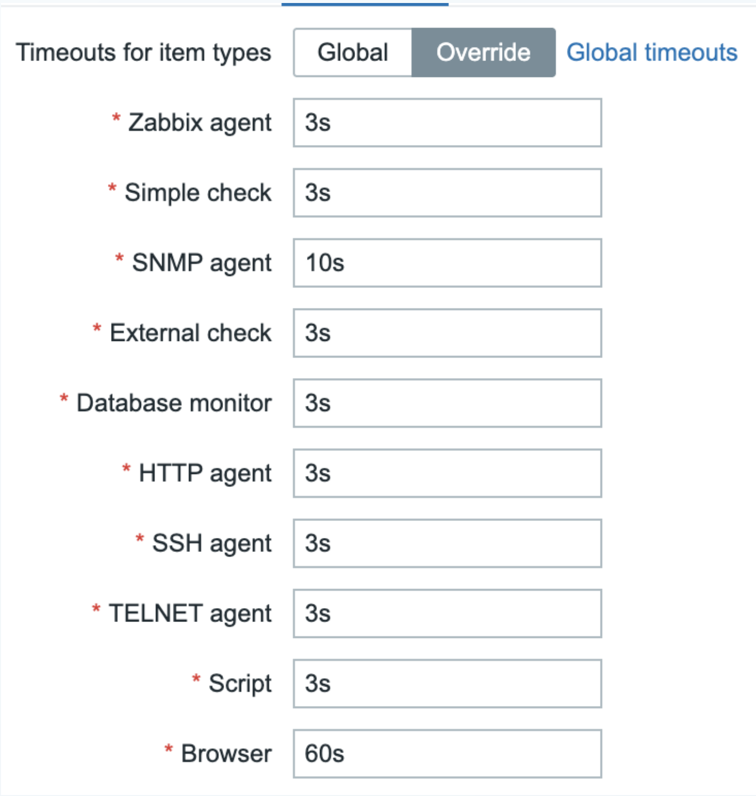
INDIVIDUAL TIMEOUTS

Zabbix 7.0 allows you to define data collection timeouts:

- On a global level per item type
- On a proxy level per item type
- For each individual item



* Timeout	Global [✓ Override]	15s	Timeouts
* History	Do not store [✓ Store up to]	31d	

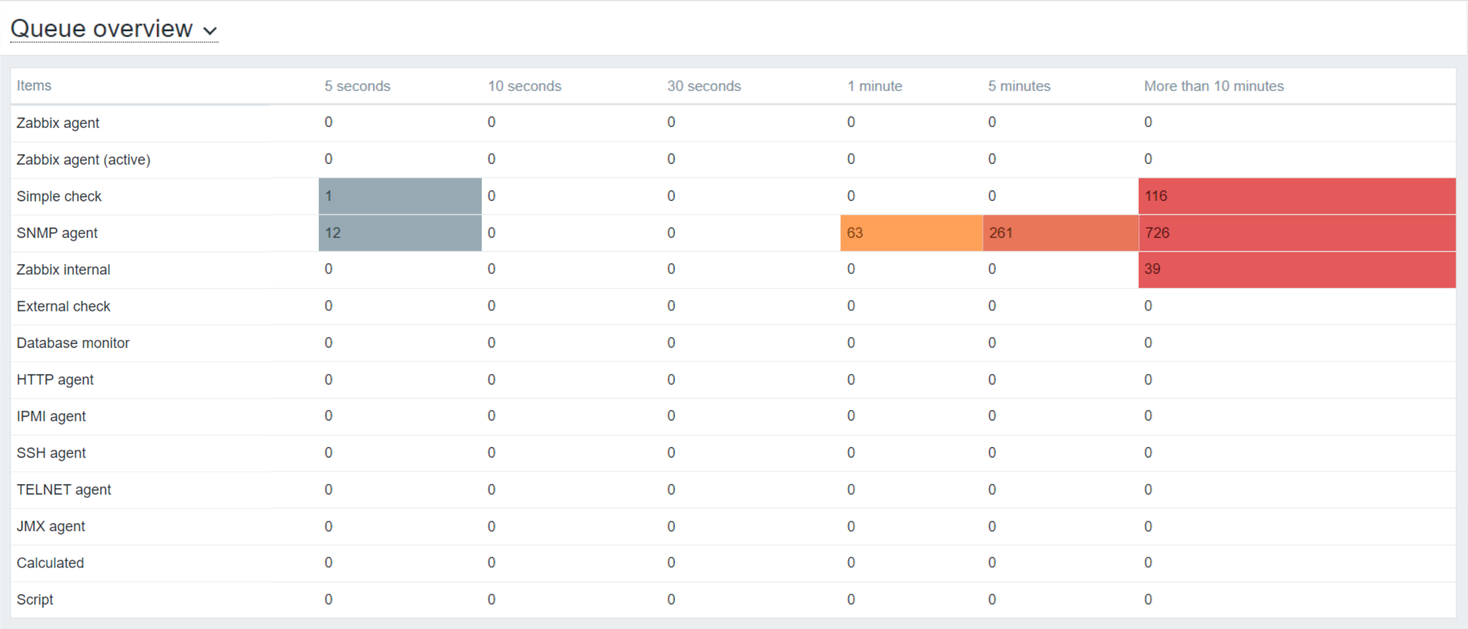


Timeouts for item types	Global [✓ Override]	Global timeouts
* Zabbix agent	3s	
* Simple check	3s	
* SNMP agent	10s	
* External check	3s	
* Database monitor	3s	
* HTTP agent	3s	
* SSH agent	3s	
* TELNET agent	3s	
* Script	3s	
* Browser	60s	

ZABBIX QUEUE

Data collection issues are displayed in the Administration > Queue page:

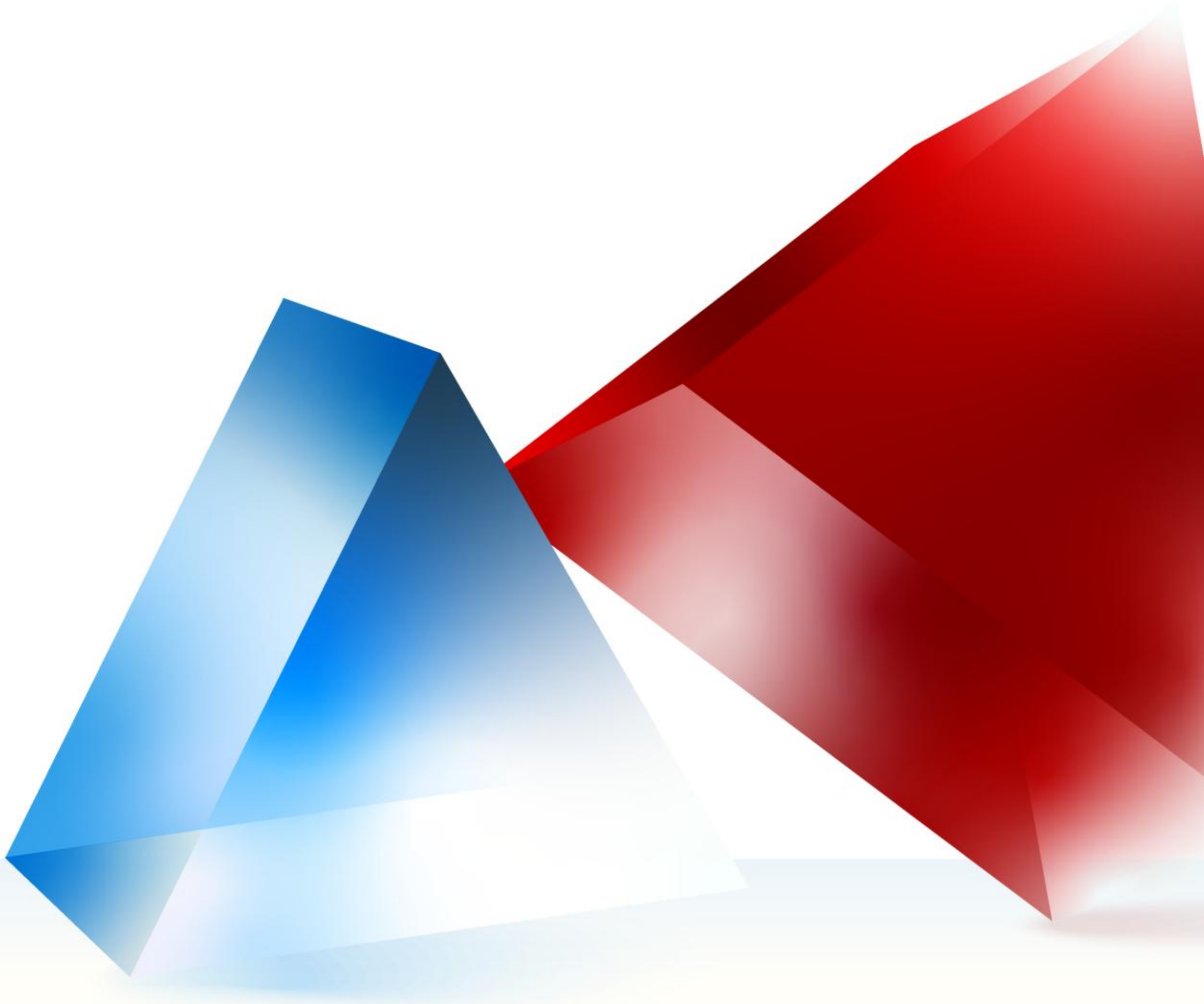
- Live information collected directly from Zabbix server
- This page only displays data collection delays



Queue overview ⌄

Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Zabbix agent	0	0	0	0	0	0
Zabbix agent (active)	0	0	0	0	0	0
Simple check	1	0	0	0	0	116
SNMP agent	12	0	0	63	261	726
Zabbix internal	0	0	0	0	0	39
External check	0	0	0	0	0	0
Database monitor	0	0	0	0	0	0
HTTP agent	0	0	0	0	0	0
IPMI agent	0	0	0	0	0	0
SSH agent	0	0	0	0	0	0
TELNET agent	0	0	0	0	0	0
JMX agent	0	0	0	0	0	0
Calculated	0	0	0	0	0	0
Script	0	0	0	0	0	0

Preprocessing



PREPROCESSING

Preprocessing allows us to transform collected raw metrics:

- Apply custom multipliers to transform to different units
- Extract values from larger data blocks (JSON, YAML, XML, etc.)
- Validate data using validation rules
- Handle errors with custom error handling routines
- Implement your own logic with custom JavaScript code

PREPROCESSING ON PROXIES

Preprocessing is implemented on both Zabbix server and proxies

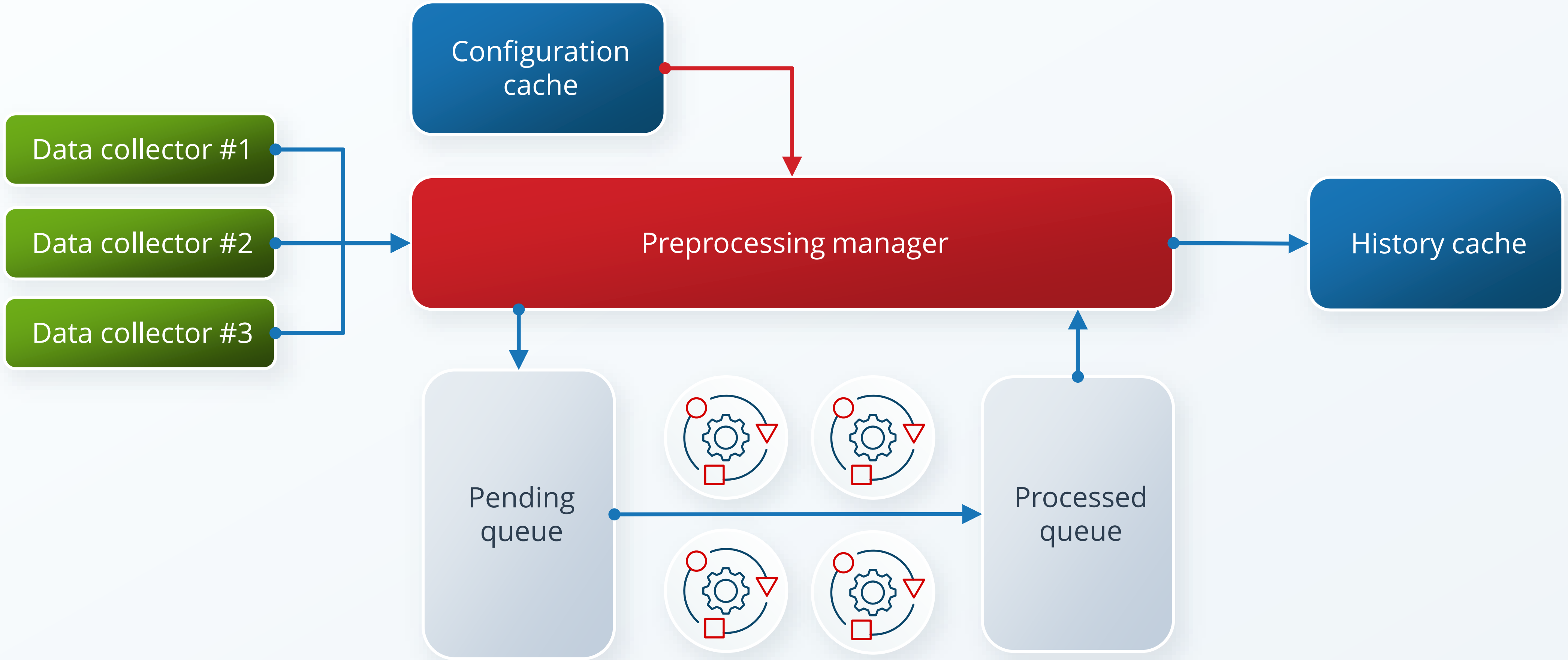
- Values collected by Zabbix server are always preprocessed by Zabbix server
- Values collected by Zabbix proxy are always preprocessed by the same proxy

Usage of proxies allows us to offload Zabbix server with horizontal scaling:

- All templates with dependent items use preprocessing to extract data
- Processing lots of large objects may consume significant CPU time

PREPROCESSING MANAGER

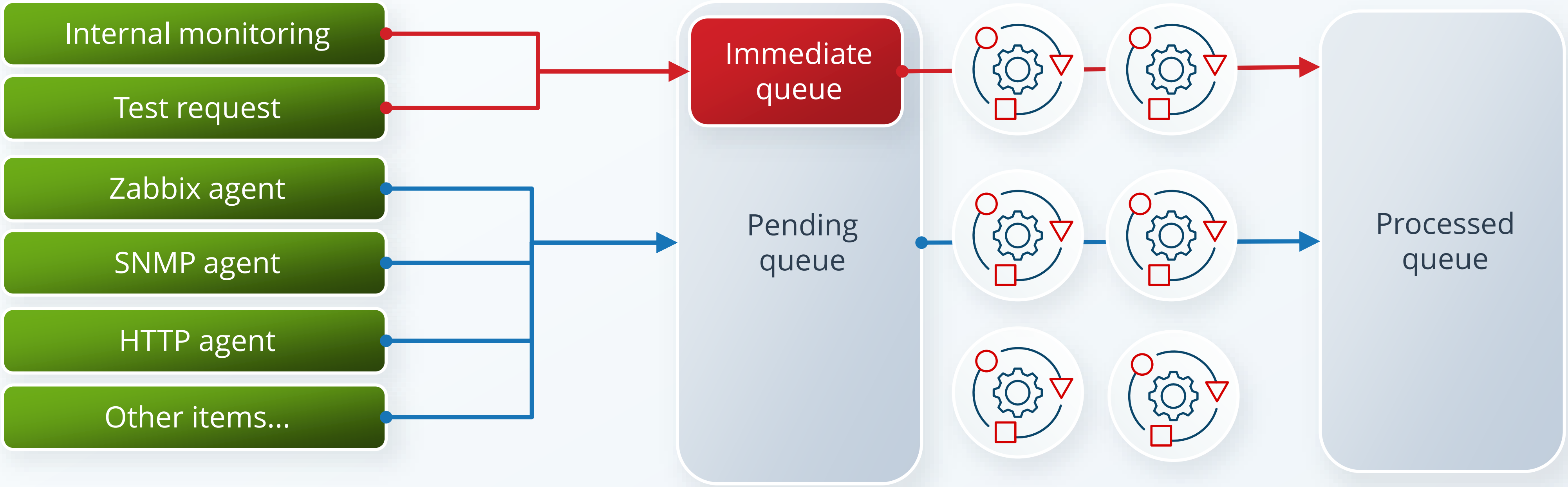
Preprocessing is managed by the preprocessing manager process:



ITEM PRIORITIES

Items in the pending queue have different priorities:

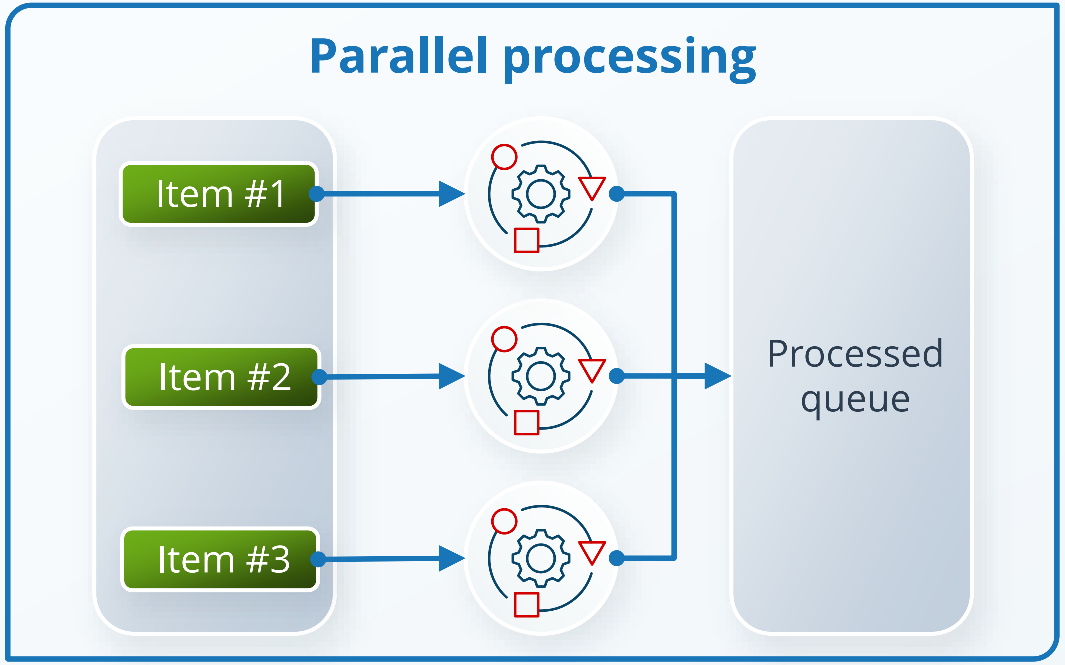
- The immediate queue is processed first using any available workers
- The pending queue is processed with the remaining workers

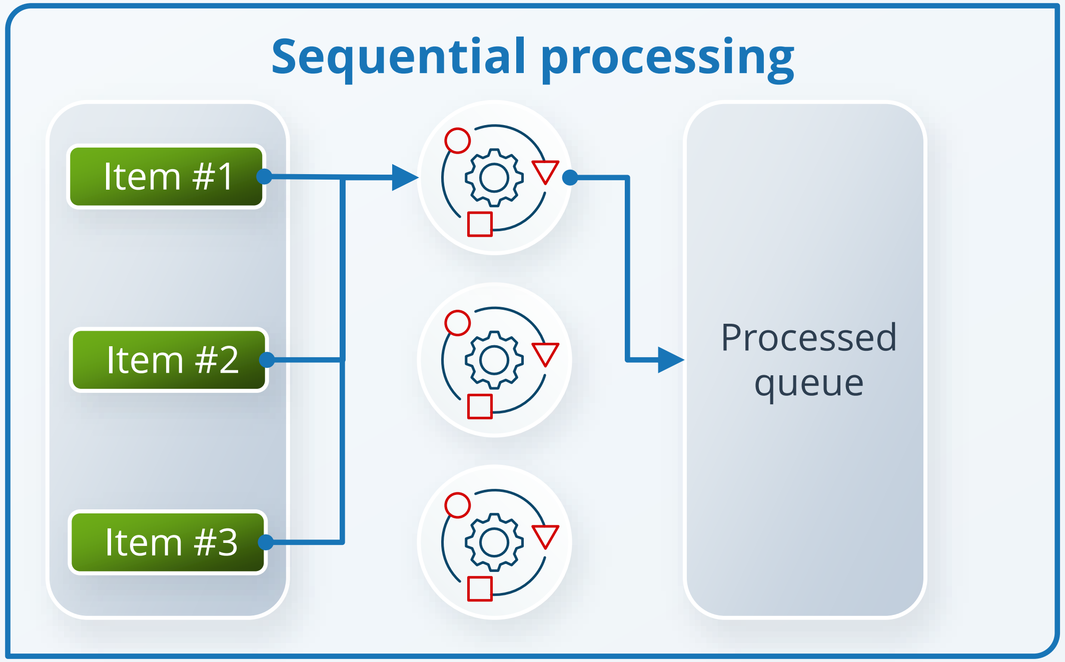


PARALLEL PROCESSING

Most values from the pending queue are processed in parallel

- Only some specific scenarios require sequential processing
- For example, a master item is preprocessed first, and then all dependent items

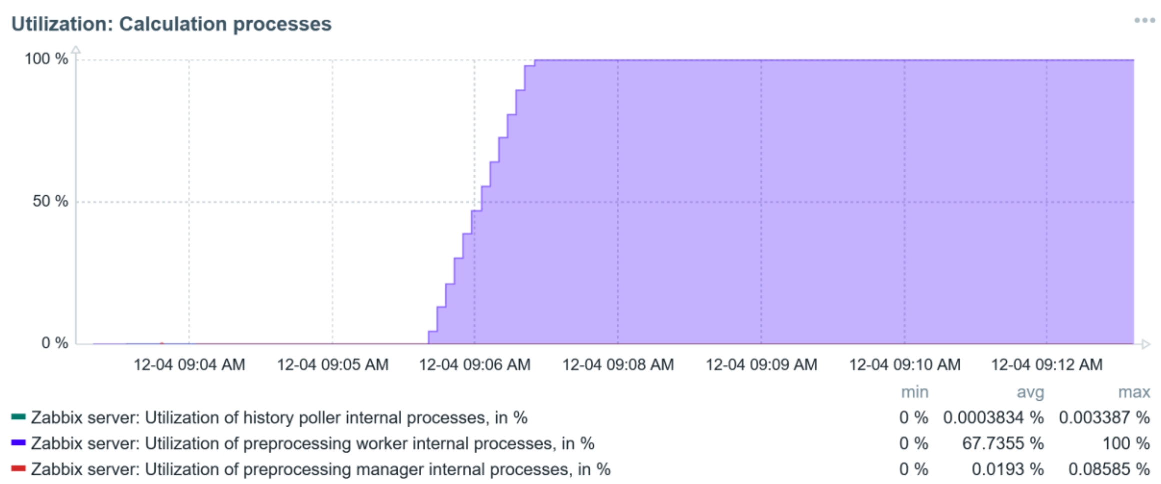




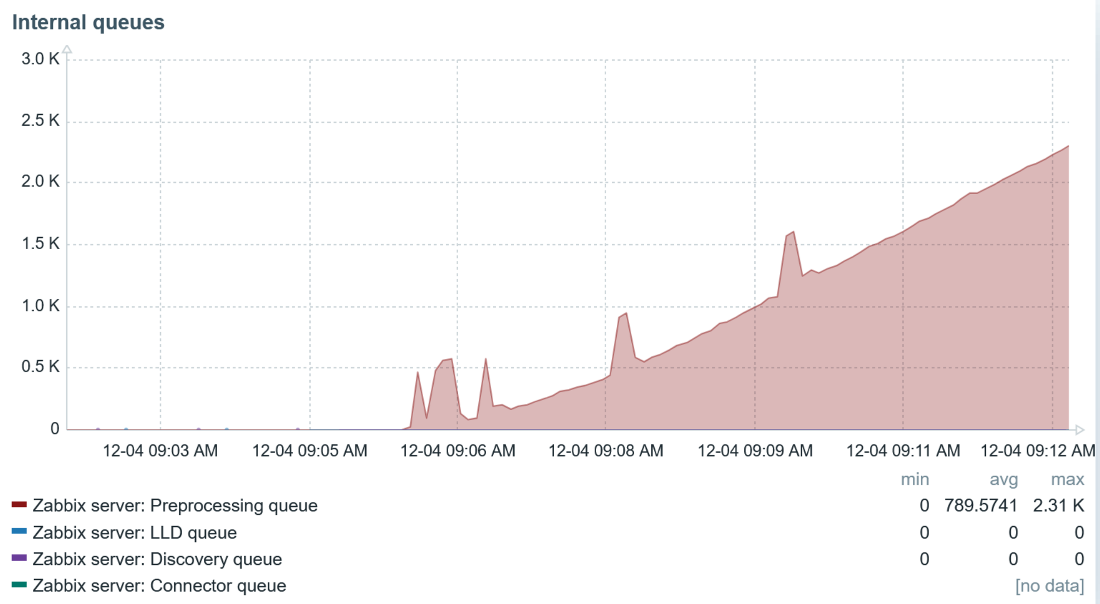
HEALTH TEMPLATES

Zabbix health templates contain the following internal metrics:

- Utilization of preprocessing manager and worker internal processes, in %
- Preprocessing queue



	min	avg	max
🟩 Zabbix server: Utilization of history poller internal processes, in %	0 %	0.0003834 %	0.003387 %
🟦 Zabbix server: Utilization of preprocessing worker internal processes, in %	0 %	67.7355 %	100 %
🟥 Zabbix server: Utilization of preprocessing manager internal processes, in %	0 %	0.0193 %	0.08585 %

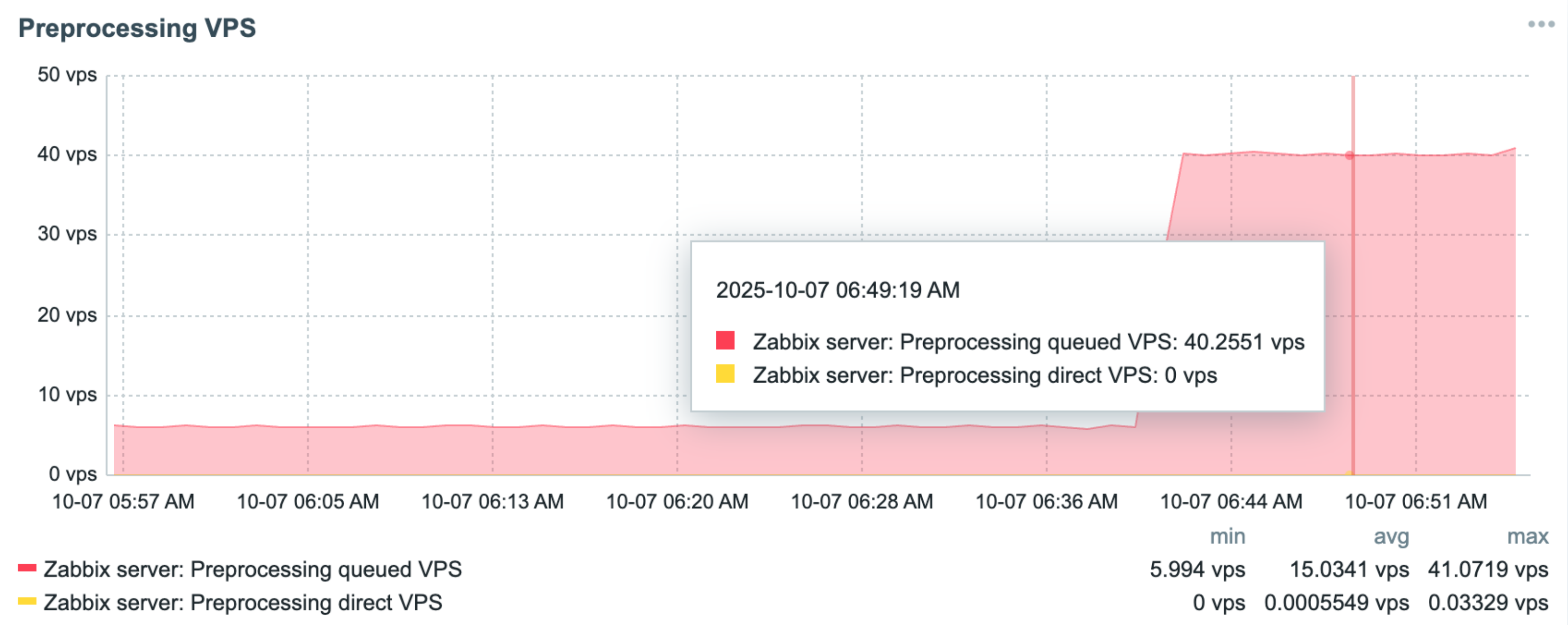


	min	avg	max
🟥 Zabbix server: Preprocessing queue	0	789.5741	2.31 K
🟦 Zabbix server: LLD queue	0	0	0
🟪 Zabbix server: Discovery queue	0	0	0
🟩 Zabbix server: Connector queue			[no data]

Preprocessing metrics

The Zabbix health template includes new metrics: zabbix[preprocessing]

- Queued values that require preprocessing
- Direct values that did not require preprocessing (only LLD)



2025-10-07 06:49:19 AM

🟥 Zabbix server: Preprocessing queued VPS: 40.2551 vps

🟨 Zabbix server: Preprocessing direct VPS: 0 vps

	min	avg	max
🟥 Zabbix server: Preprocessing queued VPS	5.994 vps	15.0341 vps	41.0719 vps
🟨 Zabbix server: Preprocessing direct VPS	0 vps	0.0005549 vps	0.03329 vps

Advanced diagnostics

The --diaginfo runtime command can be used to troubleshoot preprocessing:

- Can be used on both Zabbix server and proxies
- Shows the top items in preprocessing queue

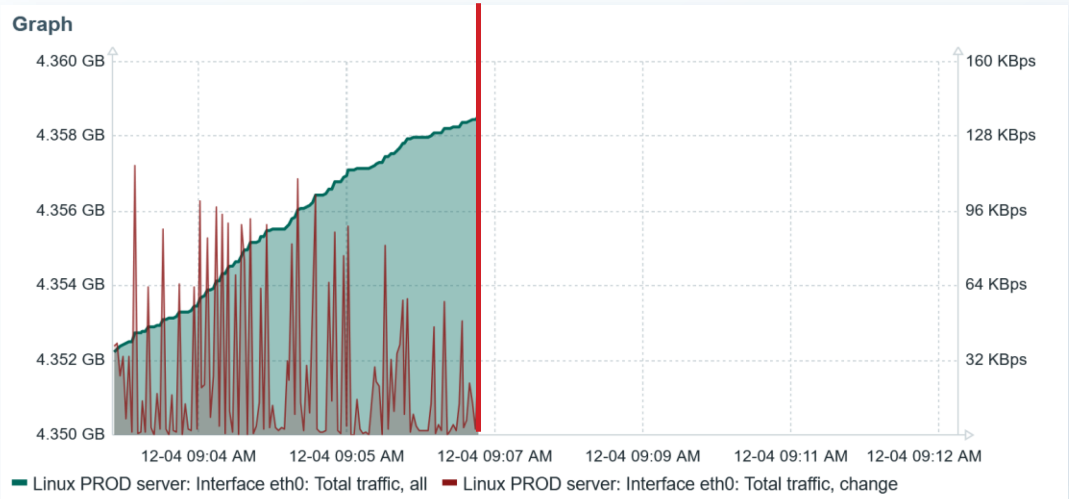
```
== preprocessing diagnostic information ==
Cached items:425 pending tasks:1042045 finished tasks:0 task sequences:0
queued count:1057004 queued size:92757706 direct count:36 direct size:311
7 history size:13765 time:0.001981
Top.sequences:
Top.peak:
  itemid:69188 tasks:2795
  itemid:69182 tasks:1248
  itemid:69634 tasks:279
  itemid:69196 tasks:63
  itemid:69340 tasks:11
```

POTENTIAL PROBLEMS

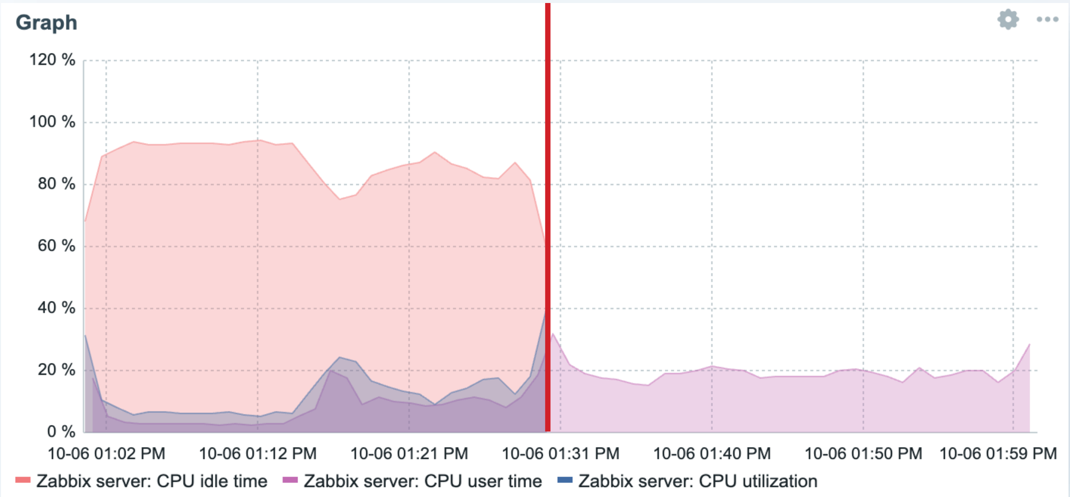
If preprocessing has performance issues, metrics will be delayed:

- In older releases all metrics were affected
- Since version 7.0.17 / 7.4.2 only metrics with preprocessing steps should be delayed

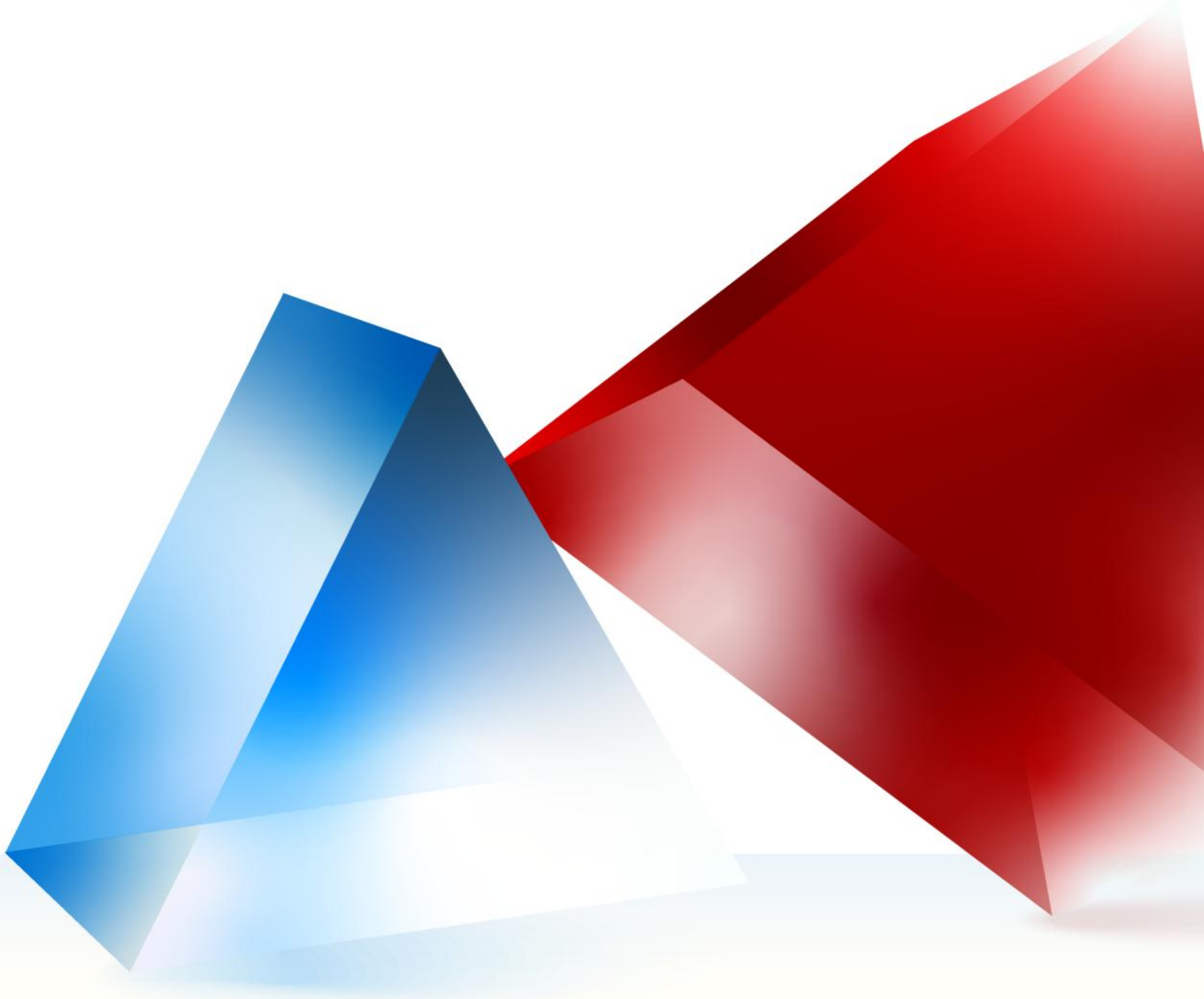
Before the new logic



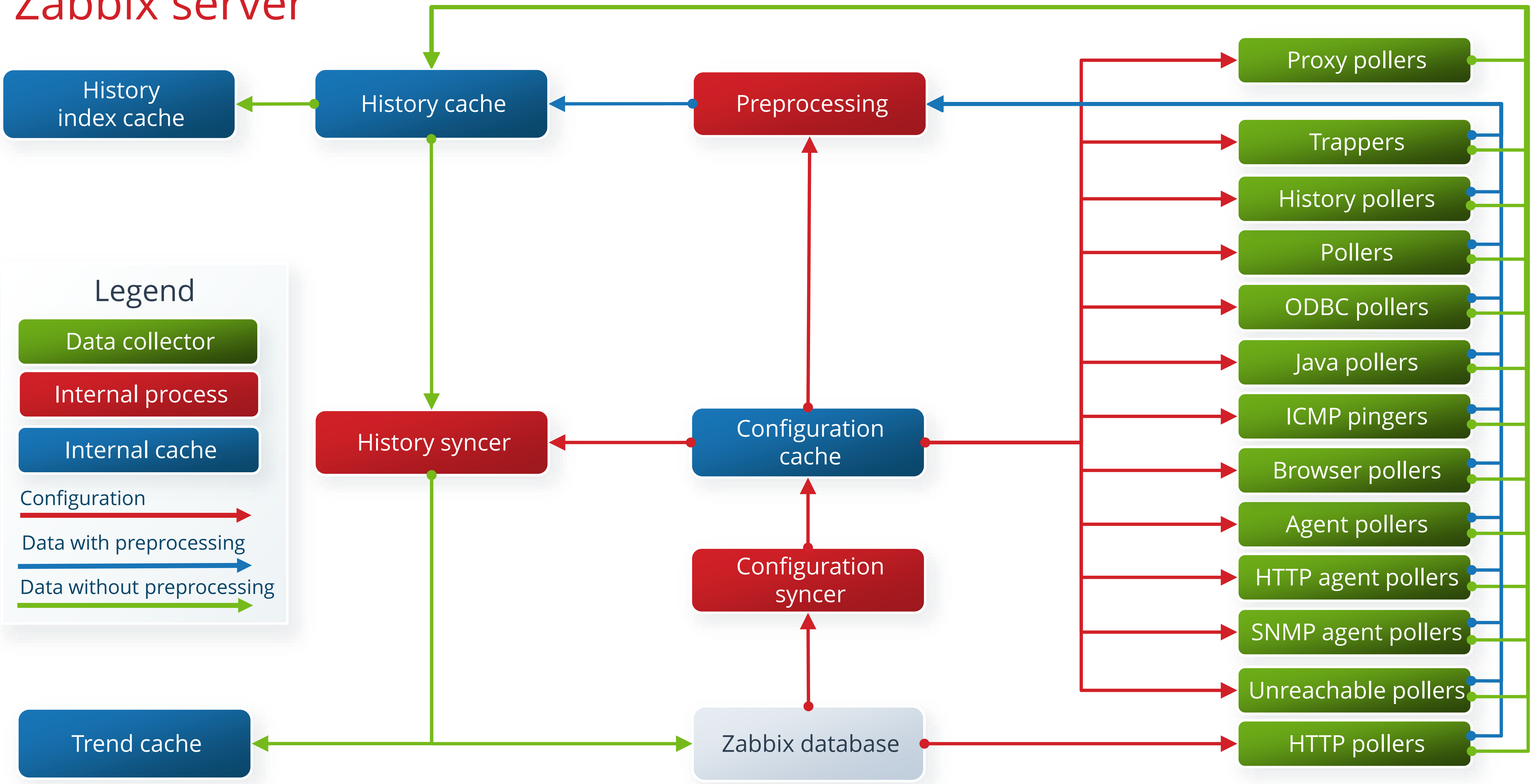
After the new logic

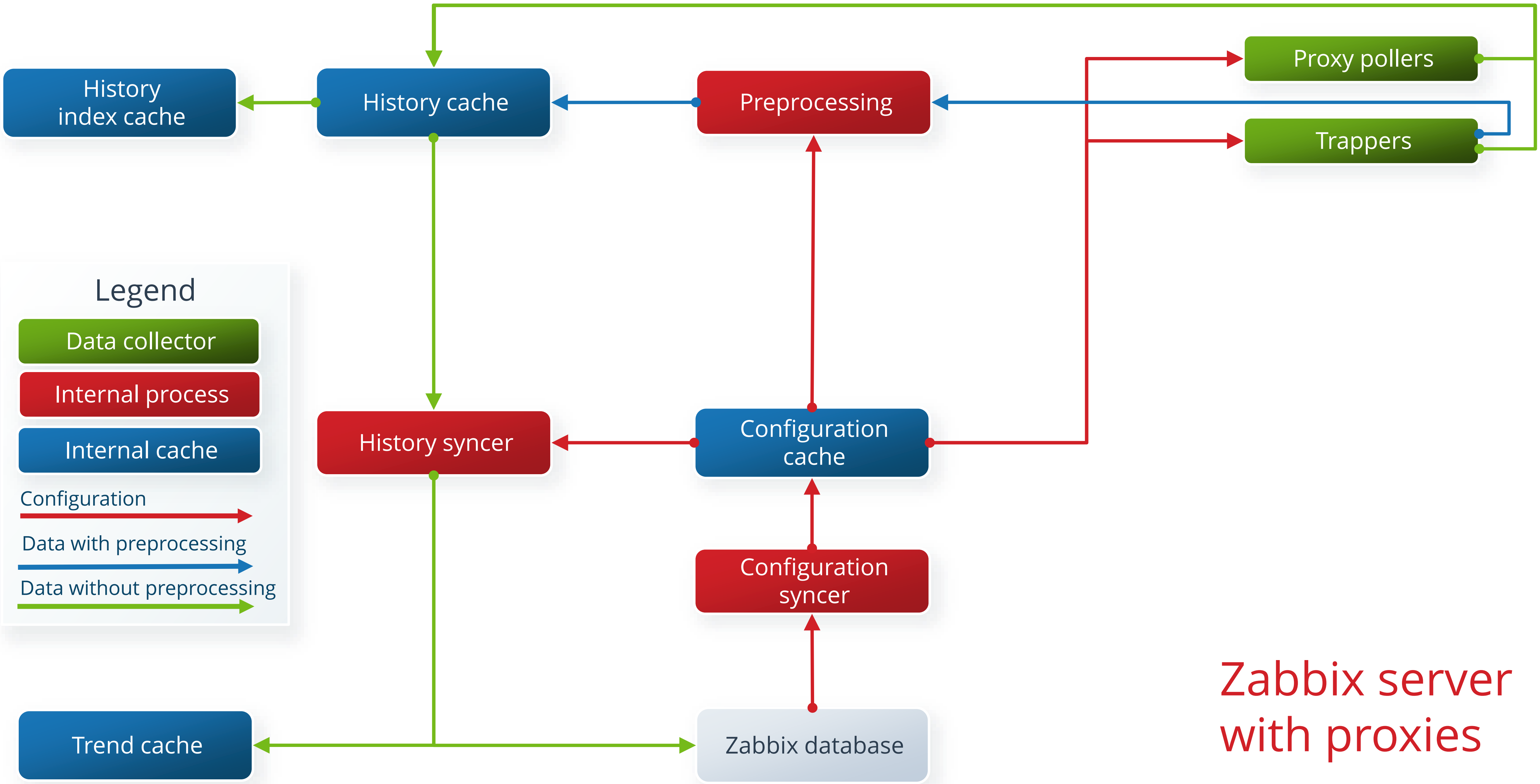


PUTTING IT ALL TOGETHER



Zabbix server





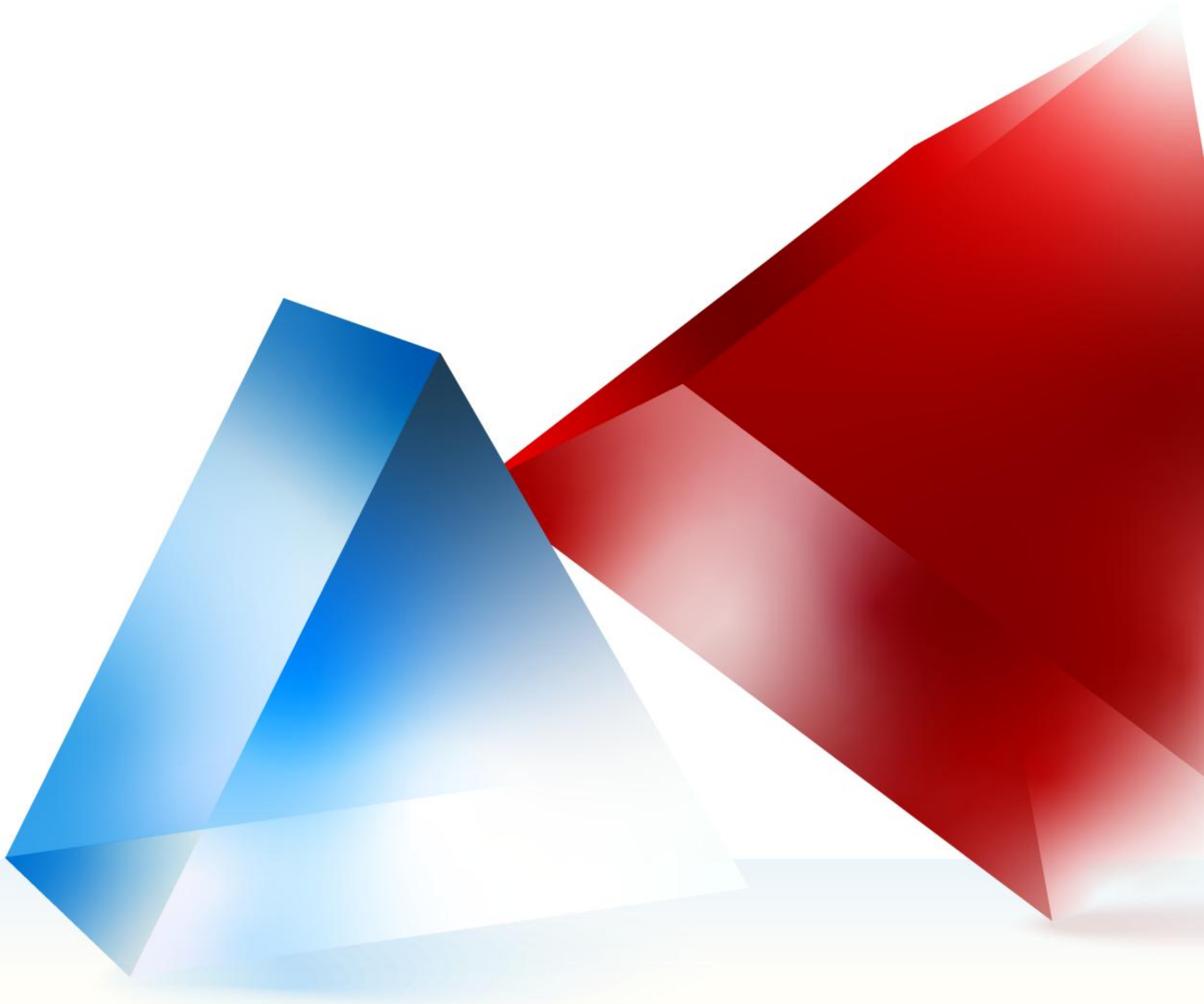
Zabbix server with proxies

SUMMARY

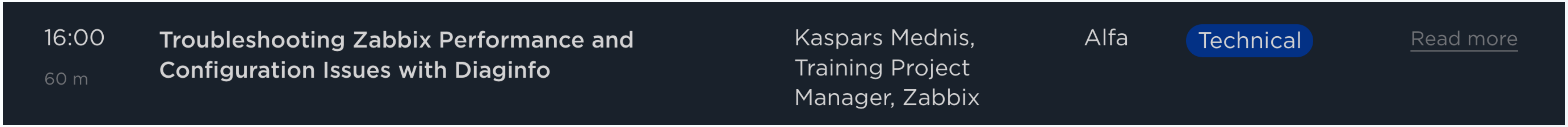
Tune your data collection process:

- Start enough data collectors
- Monitor the performance of data collectors using health templates
- Adjust timeouts if required
- Watch the preprocessing queue
- Increase preprocessors if required
- Use diagnostics to find problematic items
- Find bottlenecks if performance issues appear

Workshop



Workshop



16:00

60 m

Troubleshooting Zabbix Performance and Configuration Issues with Diaginfo

Kaspars Mednis, Training Project Manager, Zabbix

Alfa

We’ll guide you through multiple scenarios focused on troubleshooting:

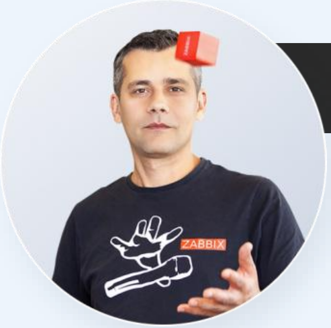
- Preprocessing issues
- Low level discovery issues
- Problematic items

You must bring your own laptop to the workshop:

- Individual virtual machines will be provided by Zabbix

ZABBIX SUMMIT

Thank You!



Kaspars Mednis

Training Project Manager