ZABBIX TIPS & TRICKS

Kaspars Mednis, ZABBIX

ZABBIX SUMMIT '18

1. {$USER_MACROS}

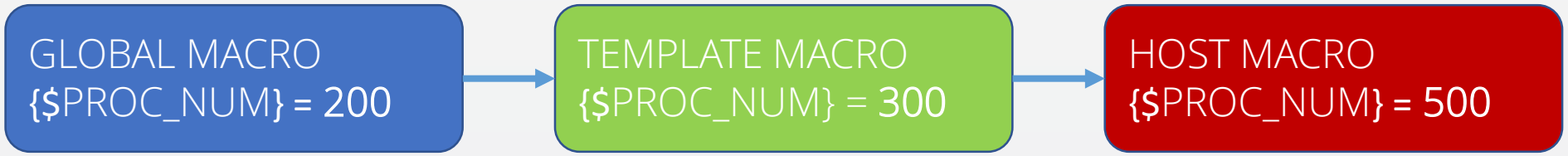
Zabbix Tips and Tricks

What are {$USER_MACROS} ?

They are **variable names** to store different information

- trigger thresholds
- different filters
- credentials
- etc.....

They have multiple levels, each new one overriding the previous



Fixed thresholds in templates

Different servers may have different **threshold values** for items like CPU load, free memory size, number of processes etc.

{Template OS: system.cpu.load[percpu,avg1].avg(5m)}>**5**
{Template OS: vm.memory.size[available].last(0)}<**20M**
{Template OS: proc.num[].avg(5m)}>**300**

TEMPLATE MACRO

HOST MACRO

{$PROC_NUM} = 300

Standard host

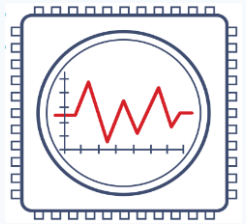
{$PROC_NUM} = 300

Heavily loaded host

{$PROC_NUM} = 500

Lightly loaded host

{$PROC_NUM} = 200



Dynamic thresholds in templates

ZABBIX SUMMIT 18

Use **{$USER_MACROS}** as **threshold values** for items like CPU load, free memory size, number of processes etc.

{Template OS: system.cpu.load[percpu,avg1].avg(5m)}>**{$CPU_LOAD}**
{Template OS: vm.memory.size[available].last(0)}<**{$MEMORY_FREE}**
{Template OS: proc.num[].avg(5m)}>**{$PROC_NUM}**

Fixed port numbers in templates

Different servers have different **port numbers** for tcp/udp ports like ssh, http, https etc.

{Template OS: net.tcp.service[ssh,**22**]
{Template OS: net.tcp.service[http,**80**]
{Template OS: net.tcp.service[https,**443**]

TEMPLATE MACRO

HOST MACRO

{$HTTP_PORT} = 80

Standard host

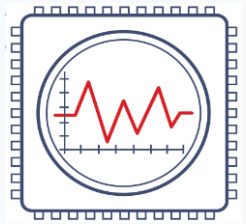
{$HTTP_PORT} = 80

Non standard host 1

{$HTTP_PORT} = 8080

Non standard host 2

{$HTTP_PORT} = 8000



Dynamic port numbers in templates

Use **{$USER_MACROS}** as **port numbers** for tcp/udp ports like ssh, http, https etc.

{Template OS: net.tcp.service[ssh,**{$SSH_PORT}**]
{Template OS: net.tcp.service[http,**{$HTTP_PORT}**]
{Template OS: net.tcp.service[https,**{$HTTPS_PORT}**]

Fixed thresholds for LLD (Low Level Discovery) items

Different mount points will have different **size**, hence different low space warning **thresholds**

• /boot	small size	**100M**
• /	medium size	**10G**
• /data	large size	**1TB**

How context based macros works

TEMPLATE MACRO

{$LOW_SPACE} = 1G

{$LOW_SPACE:"/boot"} = 10M

{$LOW_SPACE:"/"} = 500M

{$LOW_SPACE:"/var"} = 5G

HOST MACRO

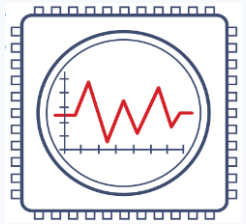
{$LOW_SPACE} = 1G

{$LOW_SPACE:"/boot"} = 10M

{$LOW_SPACE:"/"} = 500M

{$LOW_SPACE:"/var"} = 10G

{$LOW_SPACE:"/data"} = 100G



Use context based macro thresholds for LLD items

ZABBIX SUMMIT 18

Different mount points will have different **size**, so use **context based macros** to tune your triggers

• /boot	small size	**{$LOW_SPACE:"/boot"}**
• /	medium size	**{$LOW_SPACE:"/"}**
• /data	large size	**{$LOW_SPACE:"/data"}**

Can be used for **Windows drive names** also

Different services on different servers

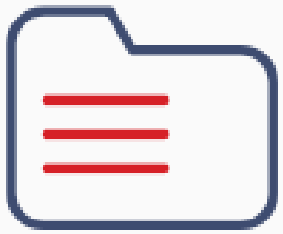
Different servers have different services, which need to be monitored

- Server 1 DHCP client, Windows defender
- Server 2 DHCP client, Windows defender, MS Exchange
- Server 3 DHCP client, RDP Service, RPC

Just @Global regular expressions can not be used because there are **too many combinations**

TEMPLATE MACRO

HOST MACRO



{$SERVICES} =none

Standard host

{$SERVICES} = none

OR

@SERVICES

Host with exchange

{$SERVICES}= (exchange)

OR

@SERVICES

GLOBAL REGULAR EXPRESSION

Host with Tomcat and WSUS

@SERVICES = (DNS Client|DHCP Client)

{$SERVICES}= (Tomcat|WSUS)

OR

@SERVICES

Services filter example

ZABBIX SUMMIT 18

HOST MACRO

Host macros | Inherited and host macros

Macro		Value	
{$SERVICES}	⇒	(Tomcat|WSUS)	Remove

Add

Update | Clone | Full clone | Delete | Cancel

GLOBAL REGULAR EXPRESSION

Name: Services

Expressions

Expression type	Expression	Delimiter	Case sensitive	Action
Result is TRUE	^(DNS|DHCP)$			Remove

Add

Add | Cancel

Discovery rule | Filters

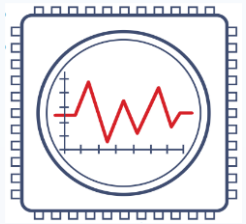
Type of calculation: Or — A or B

Filters

Label	Macro		Regular expression	Action
A	{#SERVICE.NAME}	matches	{$SERVICES}	Remove
B	{#SERVICE.NAME}	matches	@SERVICES	Remove

Add

Update | Clone | Delete | Cancel



Use a combination of {$USER_MACROS} and @global regular expressions

ZABBIX SUMMIT 18

Define {$USER_MACRO} on Host level

Use {$USER_MACRO} as **filter expression**

Global regular expressions and filters can be **combined**

Different customers use different credentials

- Different customers may have **different credentials**
  - for SNMPv2 communities
  - For SNMPv3 usernames/passwords
  - For SSH passwords
  - For WEB login passwords

Templates may be cloned for each customer, potentially leaving you with **too many templates** to manage.

TEMPLATES

TEMPLATE SNMP

{$SNMP_COMMUNITY} = public

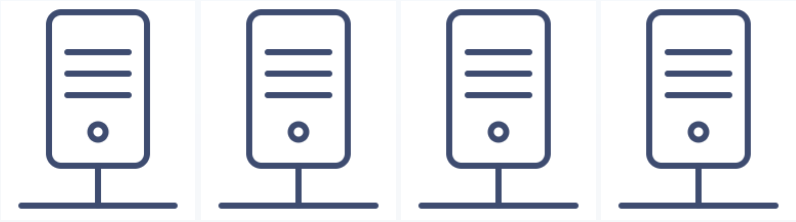
TEMPLATE Customer 1

{$SNMP_COMMUNITY} = secret1

TEMPLATE Customer 2

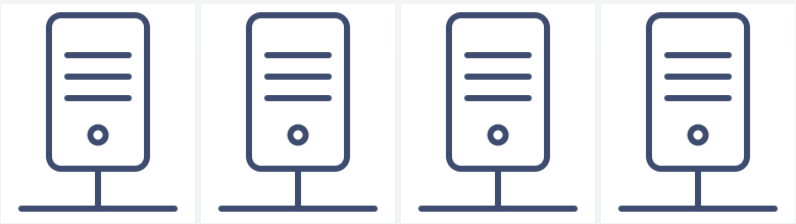
{$SNMP_COMMUNITY} = secret2

Customer 1 hosts



{$SNMP_COMMUNITY} = public

Customer 2 hosts



{$SNMP_COMMUNITY} = public

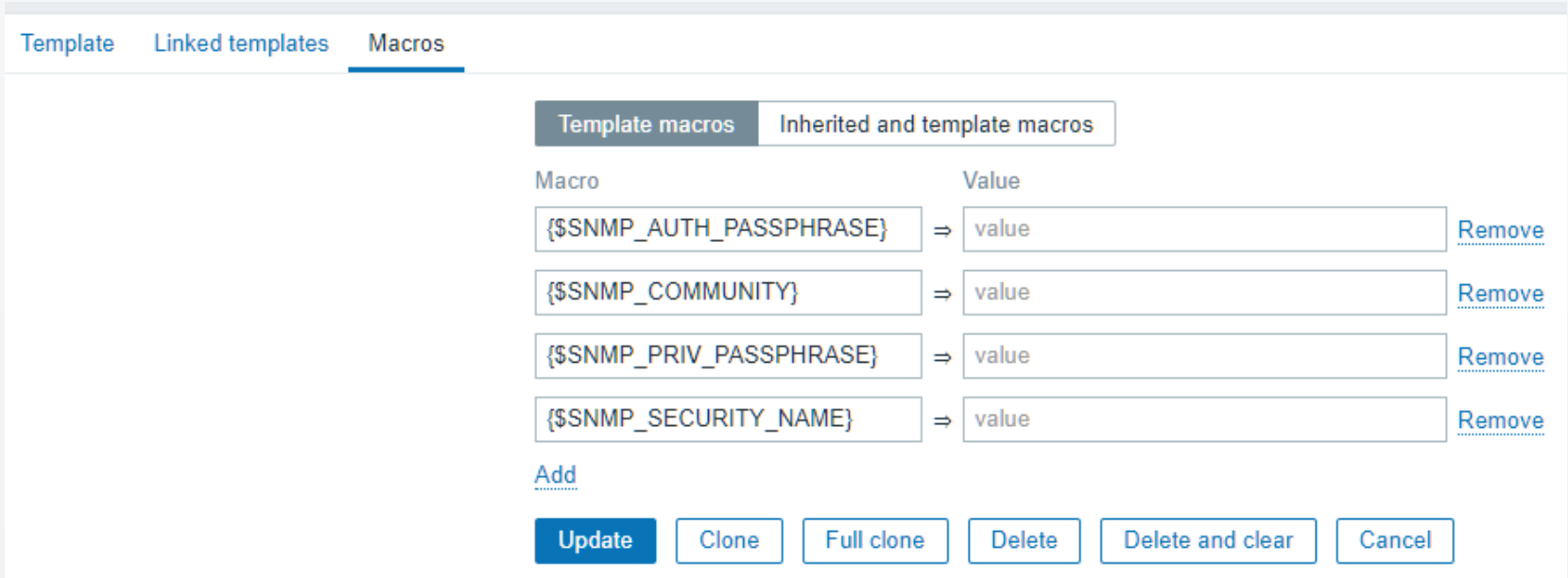
Credentials storage example

ZABBIX SUMMIT 18

Use {$USER_MACRO} as **password storage**

Define as many {$USER_MACRO} as needed

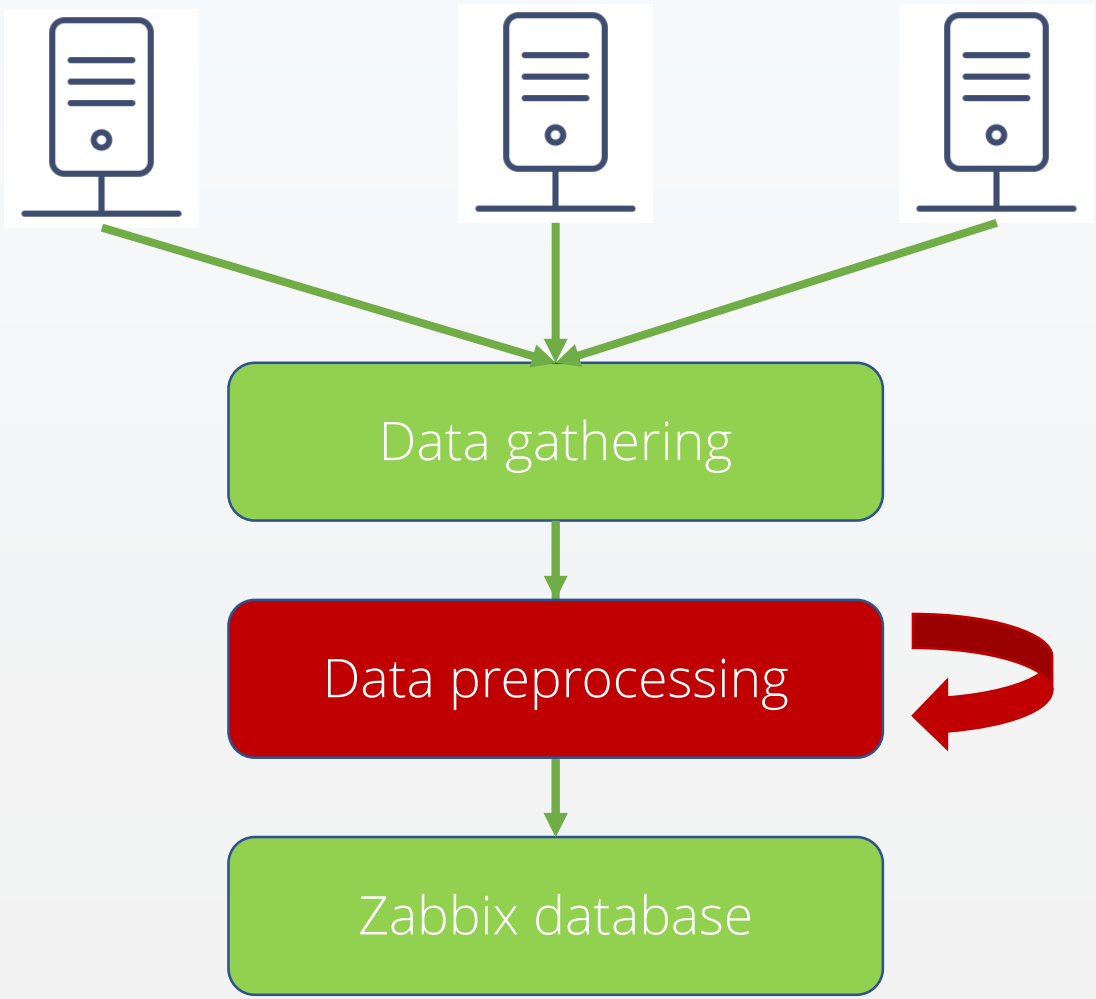
Macros still can be **overridden** on host level



2. Preprocessing

Zabbix Tips and Tricks

How preprocessing works ?



Need to divide or multiply values

- Convert Bytes to Bits
- Convert Bits to Bytes
- Convert milliseconds to seconds
- etc....

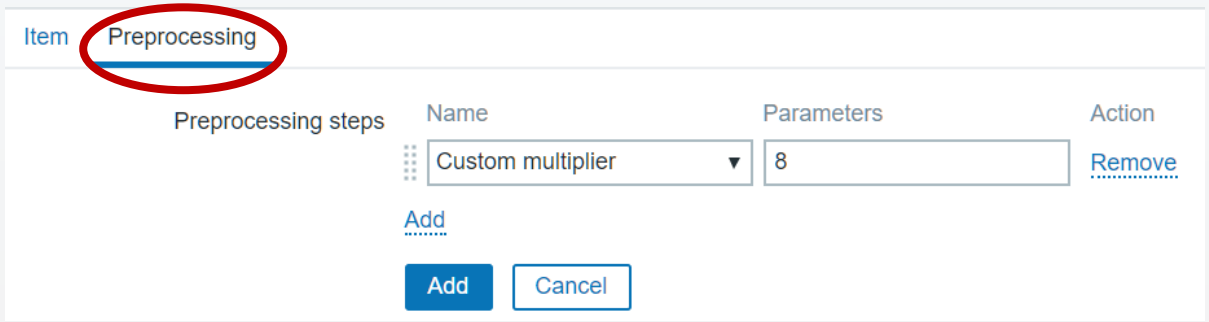
Item units may be used for visualisation, but we want to **change** the stored data.

Use Custom multiplier preprocessing

ZABBIX SUMMIT 18

Convert bits to bytes using **multiplier 8**

Convert bytes to bits using **multiplier 0.125**



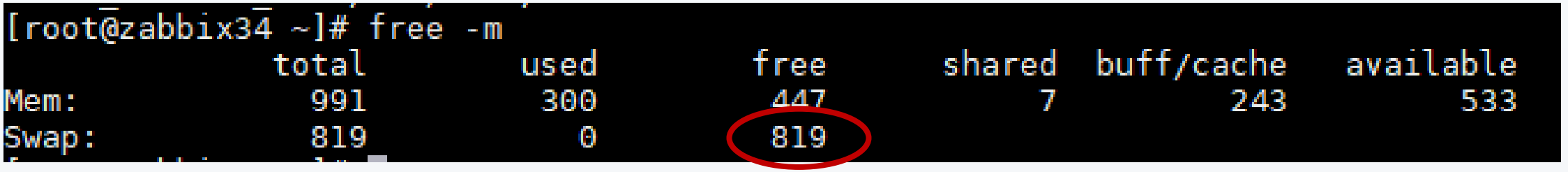
Recent Zabbix versions uses new **Preprocessing tab** for this purpose

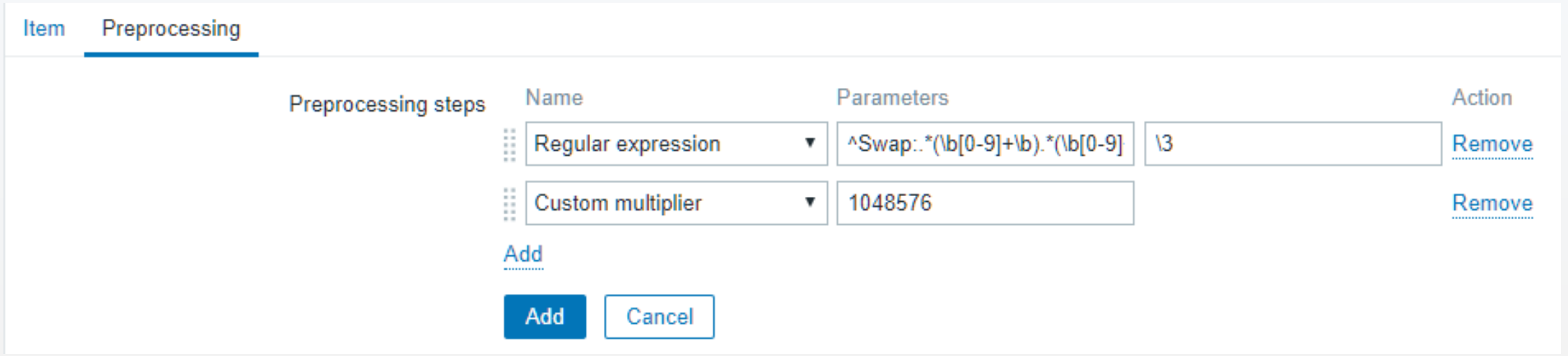
Need to extract numerical data from text

- Linux Memory
- Linux Free Space report
- Any other report which contains numerical data

```
[root@zabbix40 ~]# free -m
              total        used        free      shared  buff/cache   available
Mem:            991         357         383           7         250         477
Swap:           819           0         819
```

Use regular expression preprocessing





Extract data using **PCRE REGEX patterns**

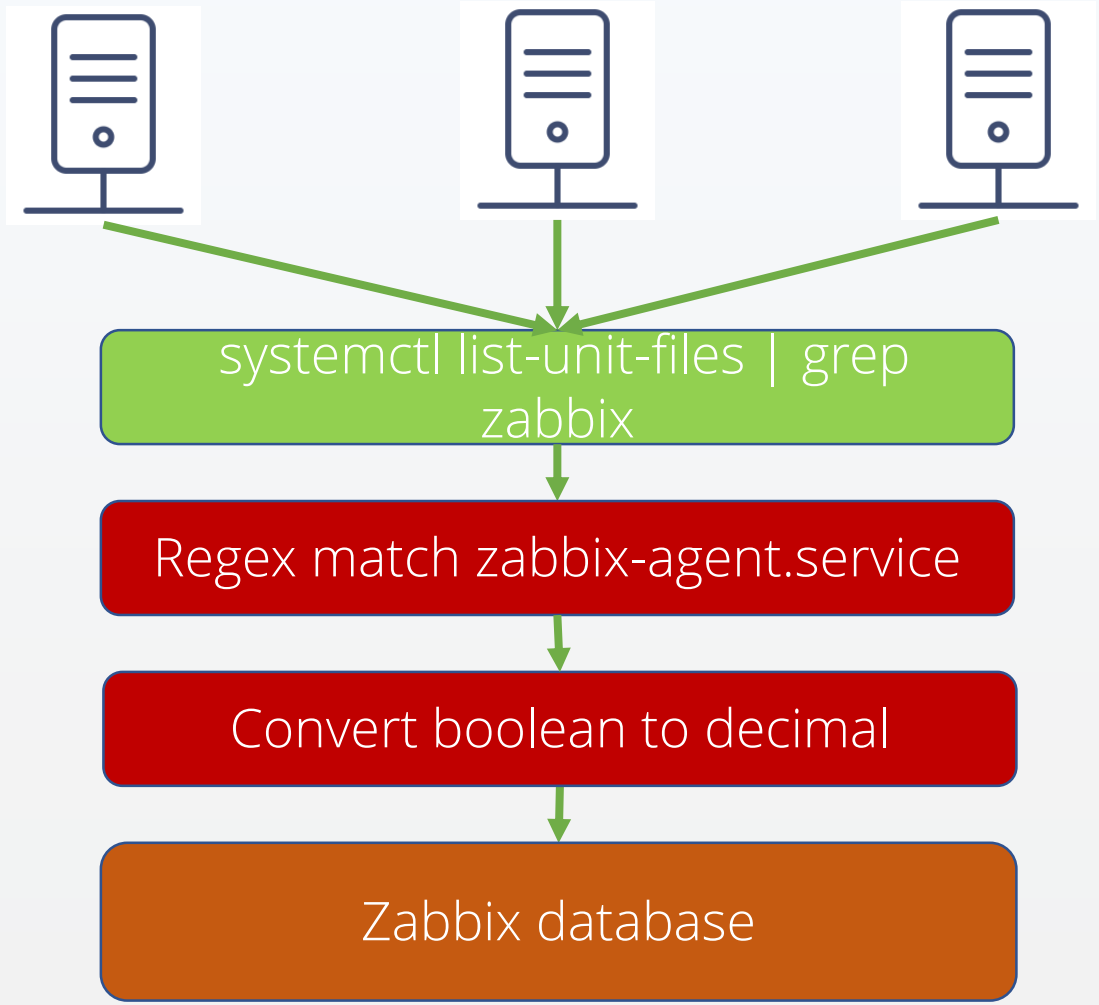
Extracted data can be processed on next steps

Need to convert Boolean text data to decimal

- systemctl reports services as enabled or disabled
- we want to store them as 1 or 0 for trigger functions or graphical representation

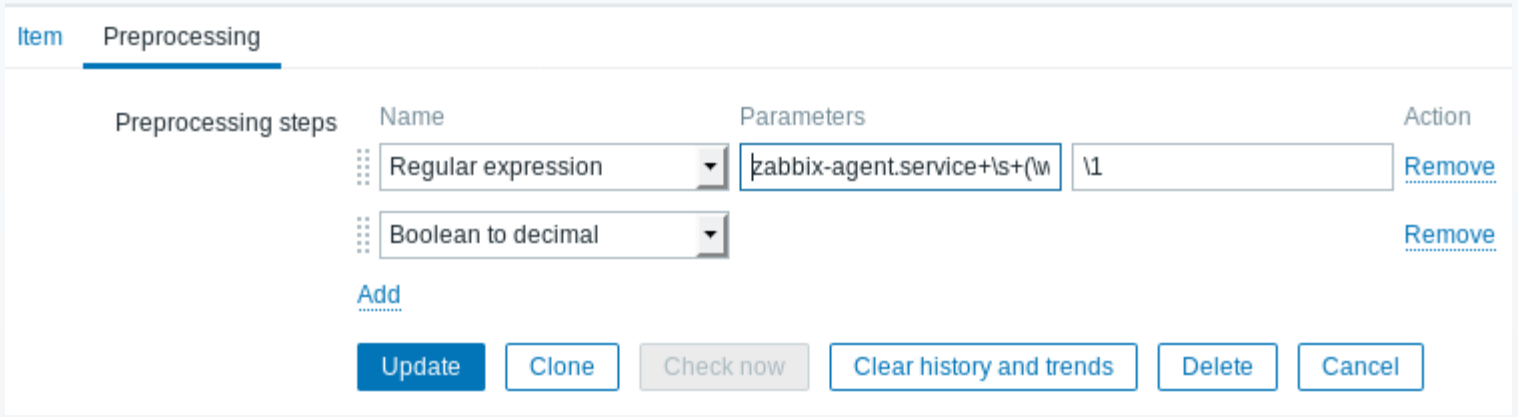
```
[root@zabbix40 ~]# systemctl list-unit-files | grep zabbix
zabbix-agent.service                          enabled
zabbix-java-gateway.service                   disabled
zabbix-server.service                         enabled
```

Use Boolean to decimal preprocessing

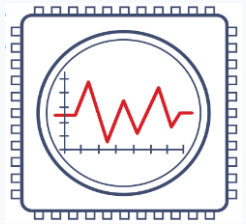


1	0
true	false
y	n
yes	no
On	off
enabled	disabled
up	down

Boolean to decimal configuration



Wizard	Name ▲	Triggers	Key	Interval	History	Trends	Type	Applications	Status
•••	Enabled Zabbix services		ssh.run[zabbix_enabled,{HOST.CONN}]	10s	0		SSH agent	Zabbix	Enabled
•••	Enabled Zabbix services: Zabbix Agent state		Zabbix_Agent_enabled		90d	365d	Dependent item	Zabbix	Enabled
•••	Enabled Zabbix services: Zabbix java gateway state		Java_gateway_enabled		90d	365d	Dependent item	Zabbix	Enabled
•••	Enabled Zabbix services: Zabbix Server state		Zabbix_Service_enabled		90d	365d	Dependent item	Zabbix	Enabled



Boolean to decimal latest data

ZABBIX SUMMIT '18

Host	Name ▲	Last check	Last value	Change	
Zabbix HOST	**Zabbix** (4 Items)				
	Enabled Zabbix services				
	Zabbix Agent state	2018-10-04 14:45:43	1		Graph
	Zabbix java gateway state	2018-10-04 14:45:43	0		Graph
	Zabbix Server state	2018-10-04 14:45:43	1		Graph

- Historical information is not stored for master item
- Gathered data can be used in graphs and triggers

3. Dependent items

Zabbix Tips and Tricks

Need to extract all numerical data from text

ZABBIX SUMMIT 18

- Linux memory, free space report or any other report which contains numerical data
- With standard items we will need to make **9(!)** checks to gather all data
- This results in additional network traffic and CPU usage on the host

```
[root@zabbix40 ~]# free -m
              total        used        free      shared  buff/cache   available
Mem:            991         357         383           7         250         477
Swap:           819           0         819
```

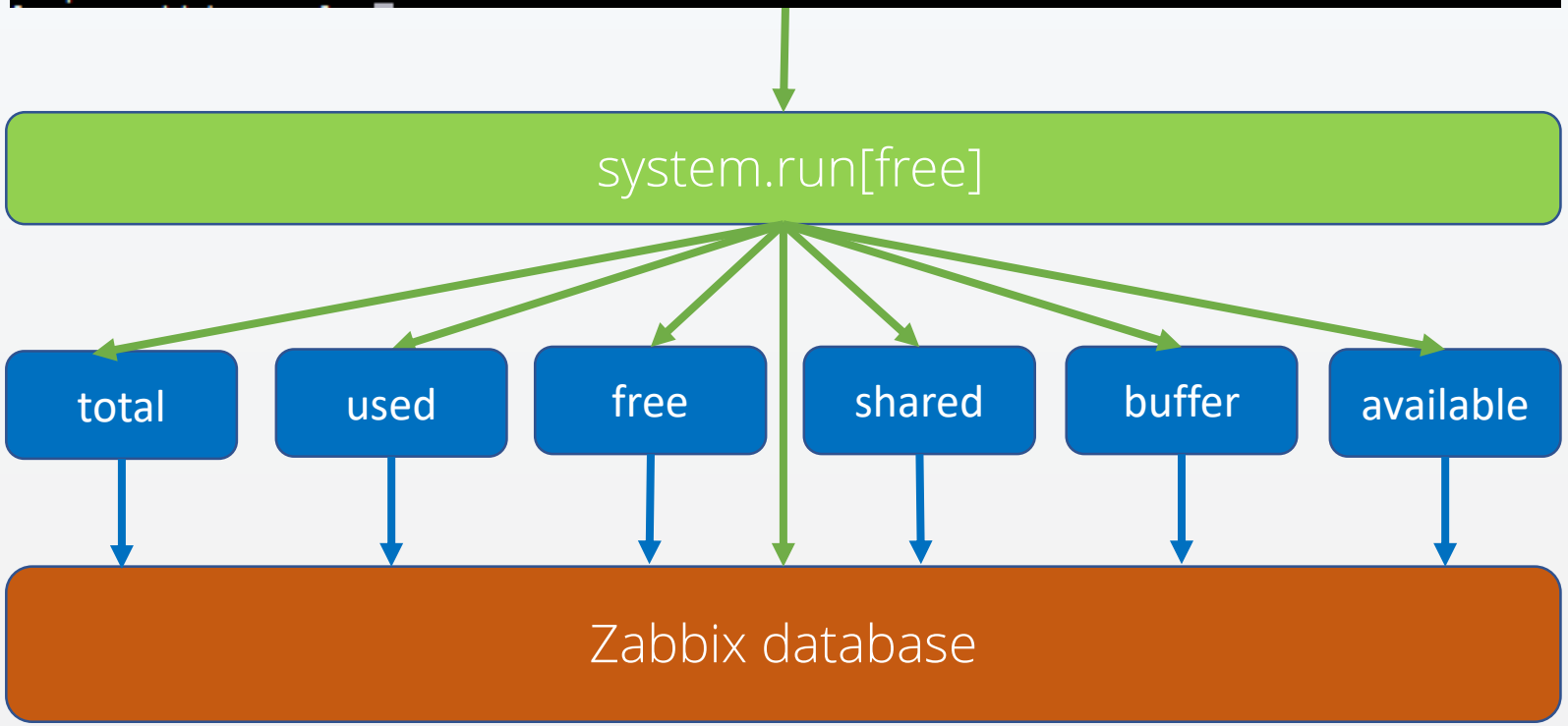
Use Boolean to decimal preprocessing step

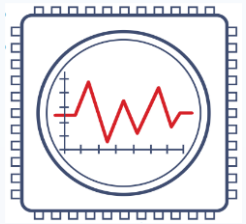
Source data

Regular text item

Dependent numeric items

```
[root@zabbix40 ~]# free
              total        used        free      shared  buff/cache   available
Mem:        1015500      476388       90012        7444      449100      370960
Swap:        839676           0      839676
```





Use dependent items

ZABBIX SUMMIT 18

Gather all data with regular text item:

Timestamp	Value
2018-09-04 09:40:46	total used free shared buff/cache available Mem: 67359399936 28717457408 6309957632 7909801984 32331984896 29861924864 Swap: 25769799680 0 25769799680

Create dependent items for each entry with regex preprocessing:

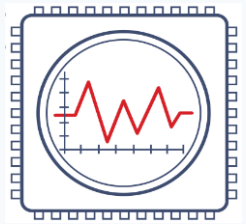
Item | Preprocessing

Preprocessing steps

Name	Parameters		Action
Regular expression	^Swap:.*(\b[0-9]+\b).*(\b[0-9]	\3	Remove
Custom multiplier	1048576		Remove

Add

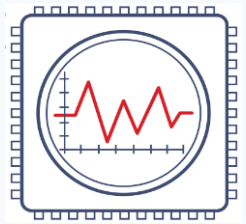
Add | Cancel



Dependent items configuration

ZABBIX SUMMIT '18

☐	•••	Memory Report	ssh.run[check.memory] 1m	0	
☐	•••	Memory Report: Memory Shared Size	ssh.ram.shared	90d	720d
☐	•••	Memory Report: Memory Total Size	ssh.ram.total	90d	720d
☐	•••	Memory Report: Memory Used Size	ssh.ram.used	90d	720d
☐	•••	Memory Report: Swap Free Size	ssh.swap.free	90d	720d
☐	•••	Memory Report: Swap Total Size	ssh.swap.total	90d	720d
☐	•••	Memory Report: Swap Used Size	ssh.swap.used	90d	720d



Dependent items latest data

ZABBIX SUMMIT 18

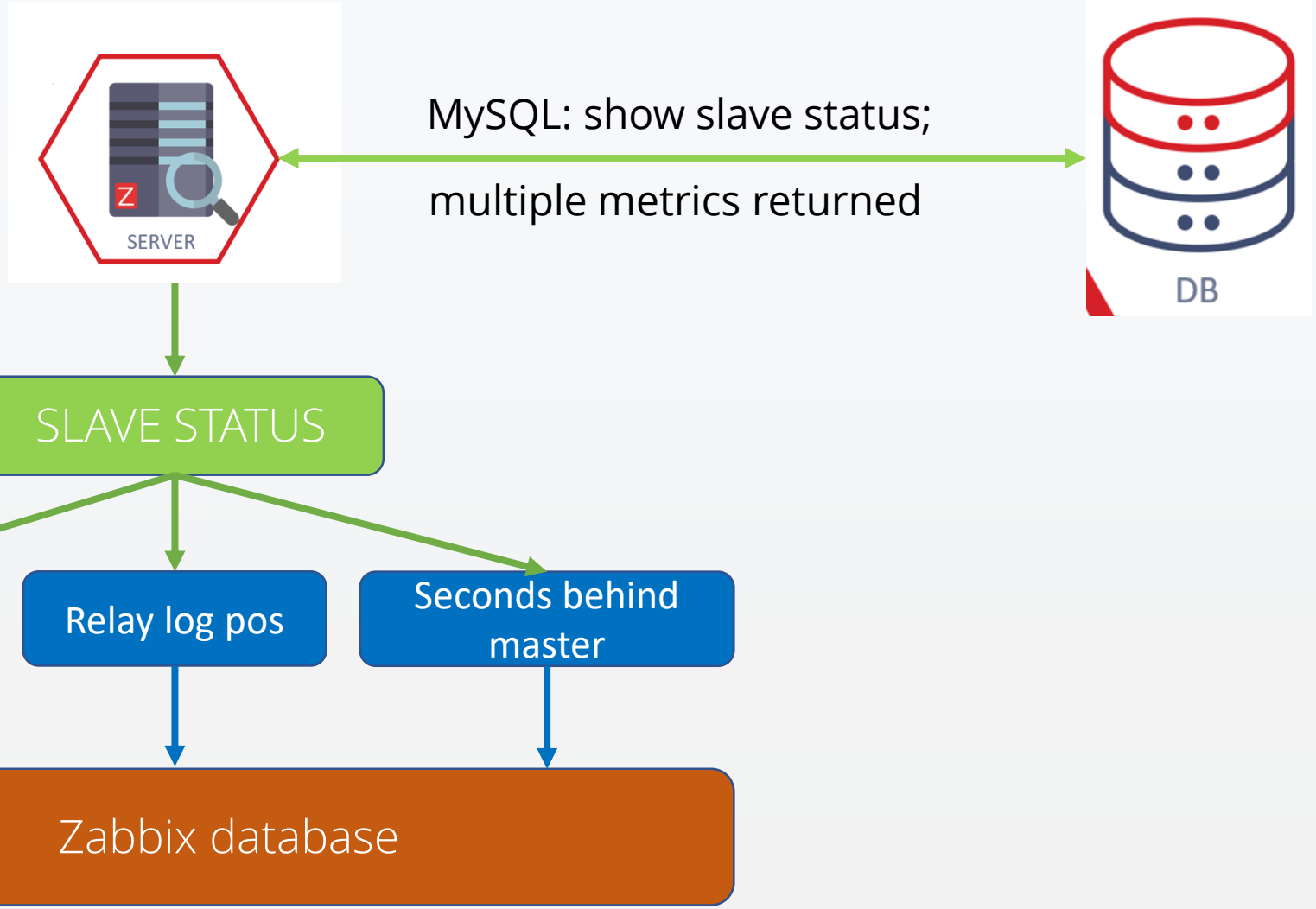
▼	**Memory** (10 Items)							
☐	Memory Available Size ssh.ram.available		90d	720d	Depend…	2018-09-04 08:4…	28.57 GB	+28.58 MB
☐	Memory Buff/cache Size ssh.ram.buff		90d	720d	Depend…	2018-09-04 08:4…	32.4 GB	+34.14 MB
☐	Memory Free Size ssh.ram.free		90d	720d	Depend…	2018-09-04 08:4…	4.42 GB	-5.81 MB
☐	Memory Report ssh.run[check.memory]	1m	0		SSH ag…			
☐	Memory Shared Size ssh.ram.shared		90d	720d	Depend…	2018-09-04 08:4…	7.43 GB	-32 KB
☐	Memory Total Size ssh.ram.total		90d	720d	Depend…	2018-09-04 08:4…	62.73 GB	
☐	Memory Used Size ssh.ram.used		90d	720d	Depend…	2018-09-04 08:4…	25.91 GB	-28.33 MB

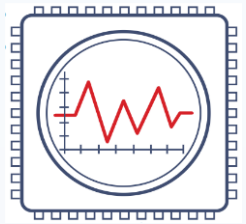
Need to monitor MySQL replication status


```
MariaDB [(none)]> show slave status \G
*************************** 1. row ***************************
Slave_IO_State: Waiting for master to send event
Master_Port: 3306
Connect_Retry: 60
Master_Log_File: master2-bin.000001
Read_Master_Log_Pos: 926945751
Relay_Log_File: master1-relay-bin.000002
Relay_Log_Pos: 207526
Relay_Master_Log_File: master2-bin.000001
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Seconds_Behind_Master: 0
```

Using regular items ODBC report must be gathered **multiple times,** resulting in **unnecessary** traffic, CPU usage and DB connections.

Use SQL queries and dependent items

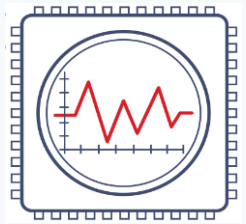




Dependent items configuration

ZABBIX SUMMIT 18

Wizard	Name ▲	Triggers	Key	Interval	History	Trends
•••	MysqL DB replication status		mysql.slavestatus	30s	0	
•••	MysqL DB replication status: Read_Master_Log_Pos		Read_Master_Log_Pos		90d	365d
•••	MysqL DB replication status: Relay_Log_File		Relay_Log_File		90d	
•••	MysqL DB replication status: Relay_Log_Pos		Relay_Log_Pos		90d	365d
•••	MysqL DB replication status: Seconds_Behind_Master	Triggers 1	Seconds_Behind_Master		90d	365d
•••	MysqL DB replication status: Slave_IO_Running	Triggers 1	Slave_IO_Running		90d	365d
•••	MysqL DB replication status: Slave_IO_State		Slave_IO_State		90d	
•••	MysqL DB replication status: Slave_SQL_Running	Triggers 1	Slave_SQL_Running		90d	365d



Dependent items latest data report

ZABBIX SUMMIT '18

Host groups: type here to search — Select

Hosts: Zabbix_DB_01 ✖ type here to search — Select

Application: MySQL Replication — Select

Name:

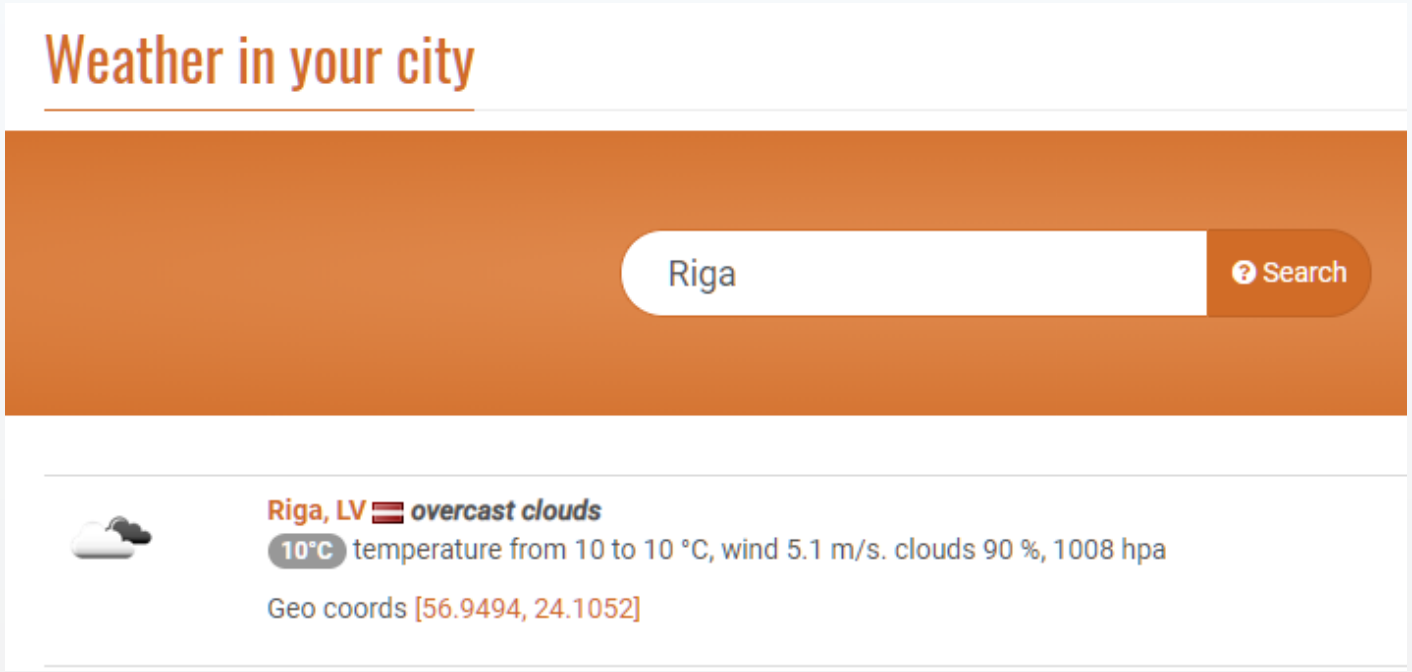
Show items without data: ☑

Show details: ☐

Apply | Reset

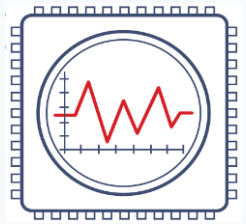
Name ▲	Last check	Last value
MySQL Replication (8 Items)		
MysqL DB replication status		
Read_Master_Log_Pos	2018-10-04 11:30:30	926945751
Relay_Log_File	2018-10-04 11:30:30	master1-relay-bin.000002
Relay_Log_Pos	2018-10-04 11:30:30	207526
Seconds_Behind_Master	2018-10-04 11:30:30	0
Slave_IO_Running	2018-10-04 11:30:30	1
Slave_IO_State	2018-10-04 11:30:30	Waiting for master to send event
Slave_SQL_Running	2018-10-04 11:30:30	1

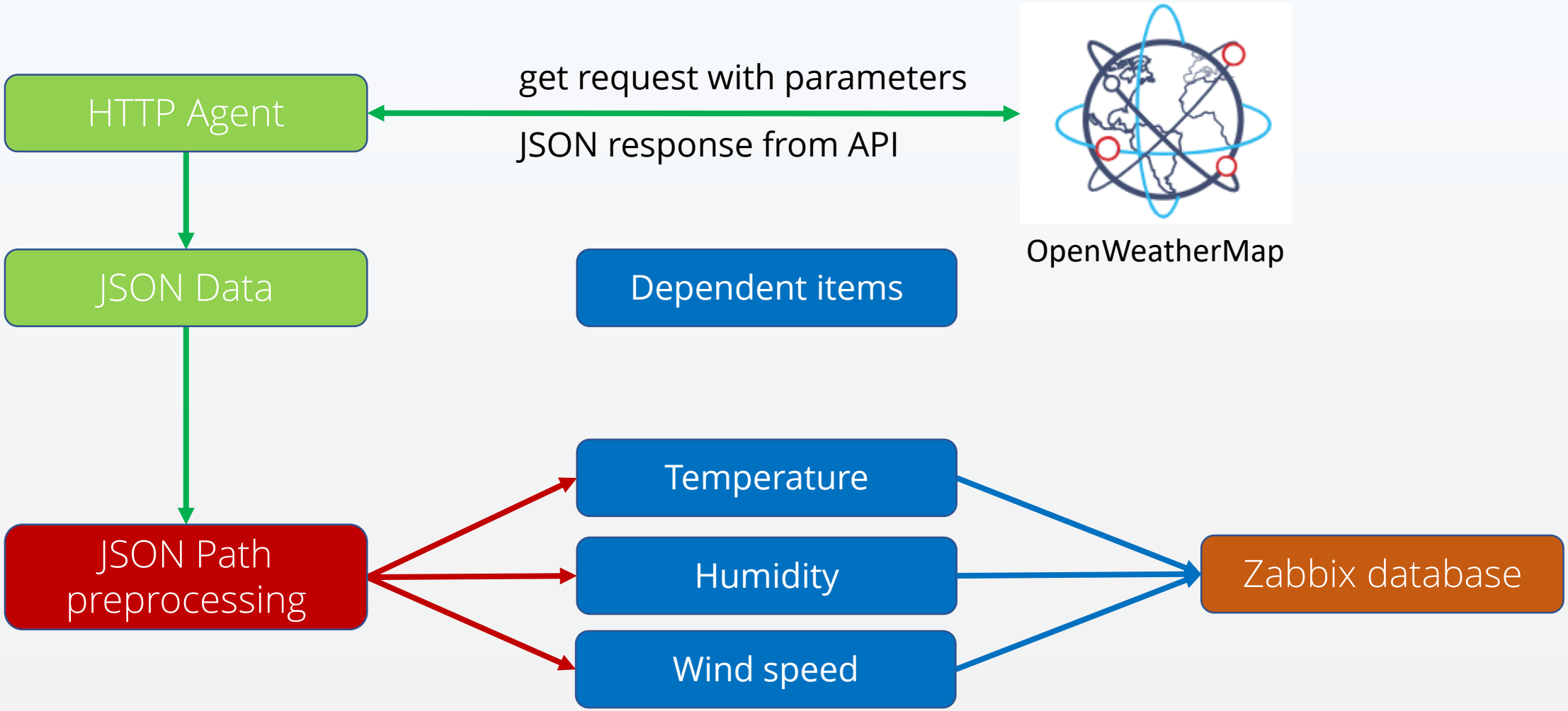
Need to gather weather data for your location



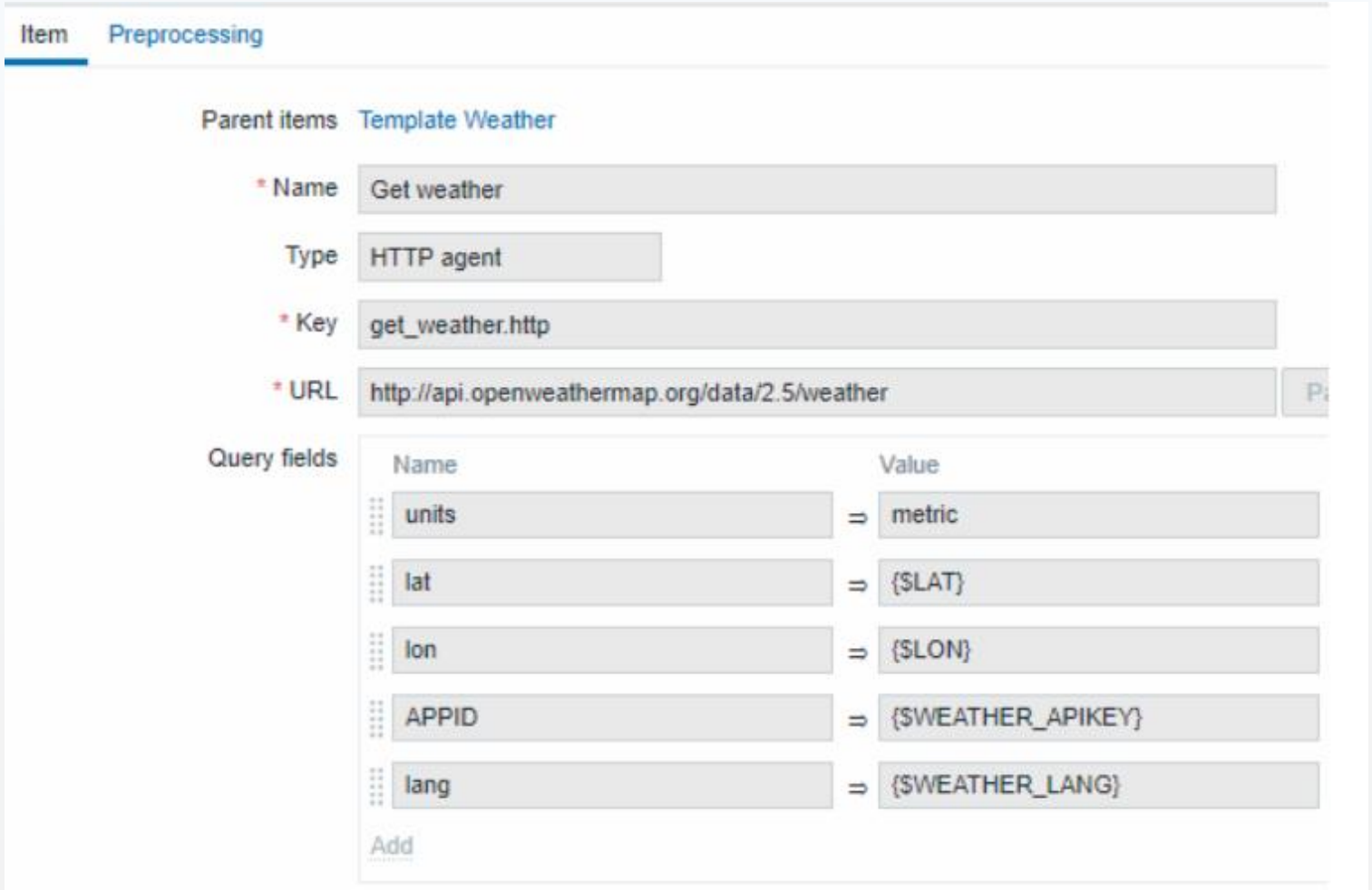
- Want to get temperature, humidity, wind speed data
- Custom curl scripts can be used, but they are **complicated**

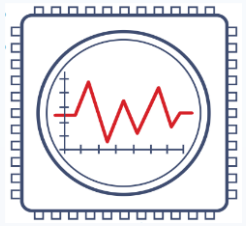
Use OpenWeatherMap API and JSON preprocessing





Use HTTP agent with {$USER_MACROS}



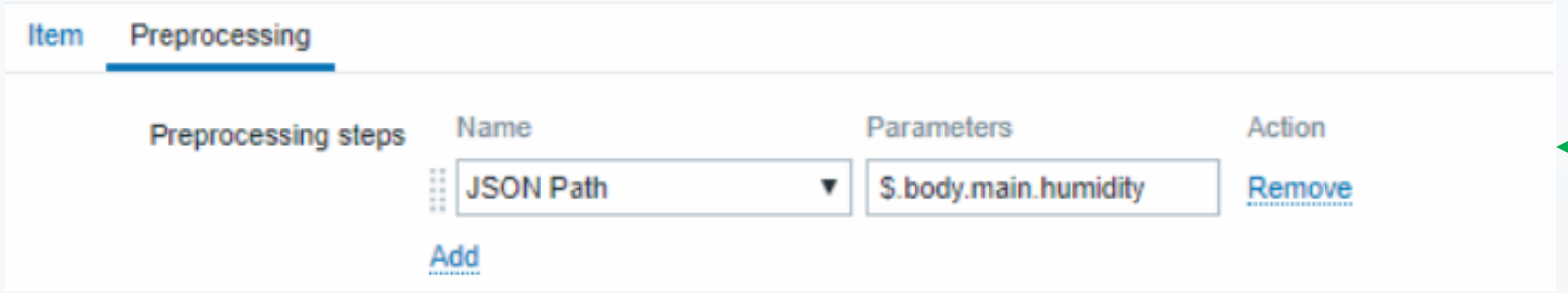


JSON data returned as response


```
{
    "body": {
        "coord": {
            "lon": 40.01,
            "lat": 56.11
        },
        "weather": [{
            "id": 801,
            "main": "Clouds",
            "description": "few clouds",
            "icon": "02n"
        }],
        "base": "stations",
        "main": {
            "temp": 15.14,
            "pressure": 1012.6,
            "humidity": 66,
            "temp_min": 15.14,
            "temp_max": 15.14,
            "sea_level": 1030.91,
            "grnd_level": 1012.6
        },
        "wind": {
            "speed": 1.86,
            "deg": 246.001
```


$.body.wind.speed

Use preprocessing to process JSON



JSON Path preprocessing

Host	Name ▲	Inter...	History	Trends	Type	Last check	Last value
weather	**Weather** (8 Items)						
	Get weather get_weather.http	10m	1d		HTTP agent	2018-05-17 01:23:45	{"body":{"coord":{"lon...
	Get weather HTTP response code get_weather.http_code		7d	0	Depende...	2018-05-17 01:23:45	OK (200)
	Humidity humidity		90d	365d	Depende...	2018-05-17 01:23:45	66 %
	Temperature temp		90d	365d	Depende...	2018-05-17 01:23:45	15.14 C
	Weather weather		90d		Depende...	2018-05-17 01:23:45	Clouds
	Weather condition id weather.condition.id		7d	0	Depende...	2018-05-17 01:23:45	801
	Weather description weather.description		90d		Depende...	2018-05-17 01:23:45	few clouds
	Wind speed wind.speed		90d	365d	Depende...	2018-05-17 01:23:45	1.86 m/s

Dependent Items

4. Low Level Discovery

Zabbix Tips and Tricks

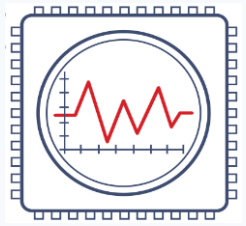
Need to discover custom SNMP metrics

- Different printer models are used
- Want to discover all printer metrics
  - Supplies level
  - Paper trays
  - Number of printed pages

Use SNMP discovery LLD rules

DISCOVERY RULE	ITEM PROTOTYPES
Supplies discovery[{#SNMPVALUE},.1.3.6.1.2.1.43.11.1.1.6]	Supplies description .1.3.6.1.2.1.43.11.1.1.6.{#SNMPINDEX}
Supplies capacity .1.3.6.1.2.1.43.8.2.1.8	Supplies capacity .1.3.6.1.2.1.43.11.1.1.8.{#SNMPINDEX}
Supplies level .1.3.6.1.2.1.43.8.2.1.9	Supplies level .1.3.6.1.2.1.43.11.1.1.9.{#SNMPINDEX}

All SNMP trees used in this example use **the same indexing** !!!



LLD rule and item prototypes

ZABBIX SUMMIT '18

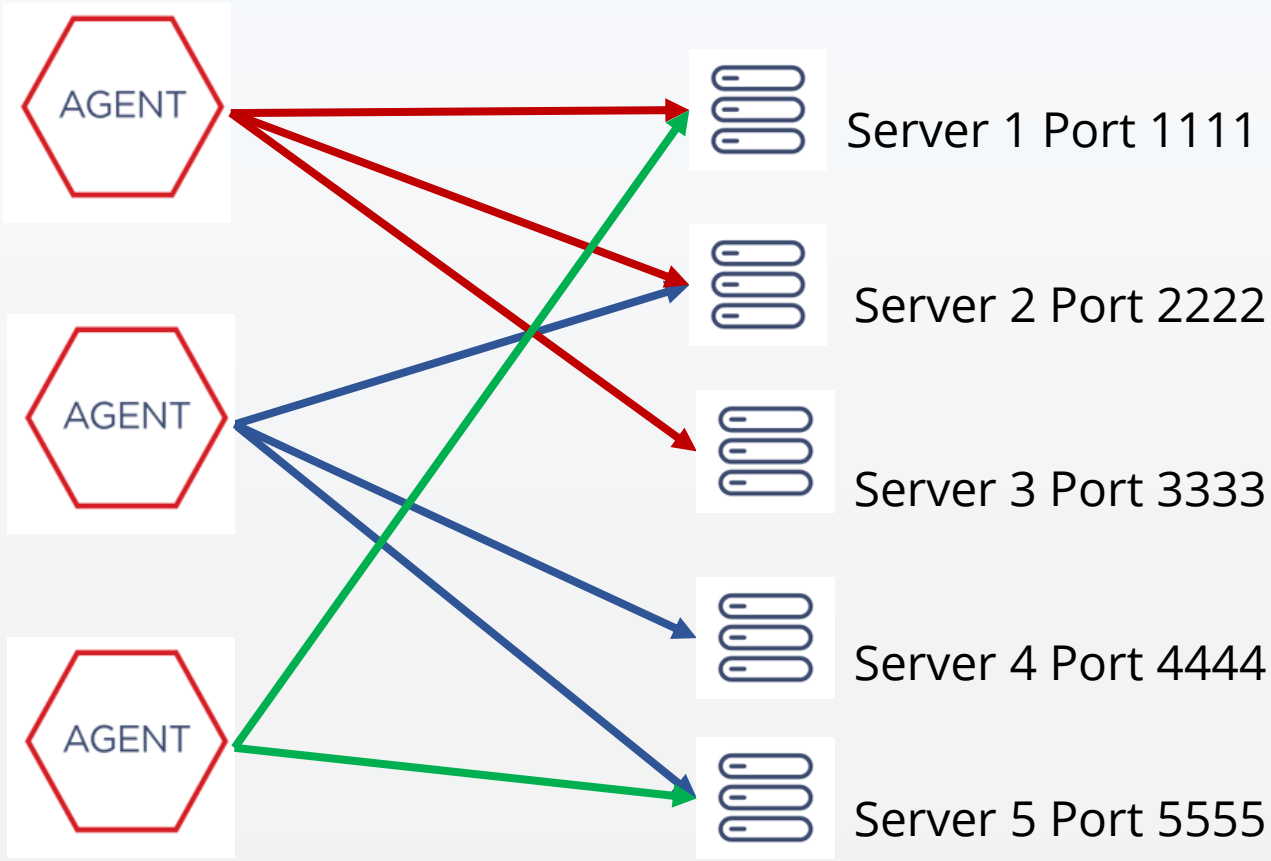
Discovery rule | Filters

* Name	Supplies
Type	SNMPv2 agent
* Key	prtMarkerSuppliesDescription
* SNMP OID	discovery[{#SNMPVALUE},.1.3.6.1.2.1.43.11.1.1.6]
* SNMP community	{$SNMP_COMMUNITY}

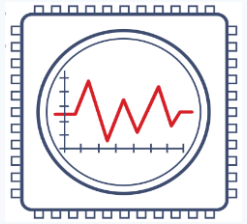
Item prototype | Preprocessing

* Name	Level of supplies $1
Type	SNMPv2 agent
* Key	prtMarkerSuppliesLevel[{#SNMPVALUE}] Select
* SNMP OID	.1.3.6.1.2.1.43.11.1.1.9.{#SNMPINDEX}
* SNMP community	{$SNMP_COMMUNITY}

Check remote port accessibility on other hosts



- net.tcp.port key is used
- there are numerous servers and ports
- each agent has different server list to check



Use custom LLD script

```
#/bin/bash
IFS=':, ' read -r -a array <<< "$1"
idx=0
echo {\"data\":[
while [ -n "${array[$idx]}" ]; do
        echo -n \{\"{#R_IP}\":\""${array[$idx]}"\",\"{#R_PORT}\":\""${array[$idx+1]}"\"\}
        let idx=$idx+2
        [ -n "${array[idx]}" ] && echo "," || echo
done
echo ]}
exit
```

- Simple **bash script** works out of box on most platforms
- Uses **{$USER_MACRO}** as input, returns **JSON object**

Create discovery rule

ZABBIX SUMMIT 18

Name	Remote TCP connection discovery
Type	External check
Key	check_ports_remote_lld.sh[{$LLD_REMOTE_CHECK}]
Update interval	15m

Custom intervals

Type	Interval	Period	Action
Flexible / Scheduling	50s	1-7,00:00-24:00	Remove

Add

Item prototypes are created automatically

ZABBIX SUMMIT 18

Host | Templates | IPMI | Macros | Host inventory | Encryption

Host macros | Inherited and host macros

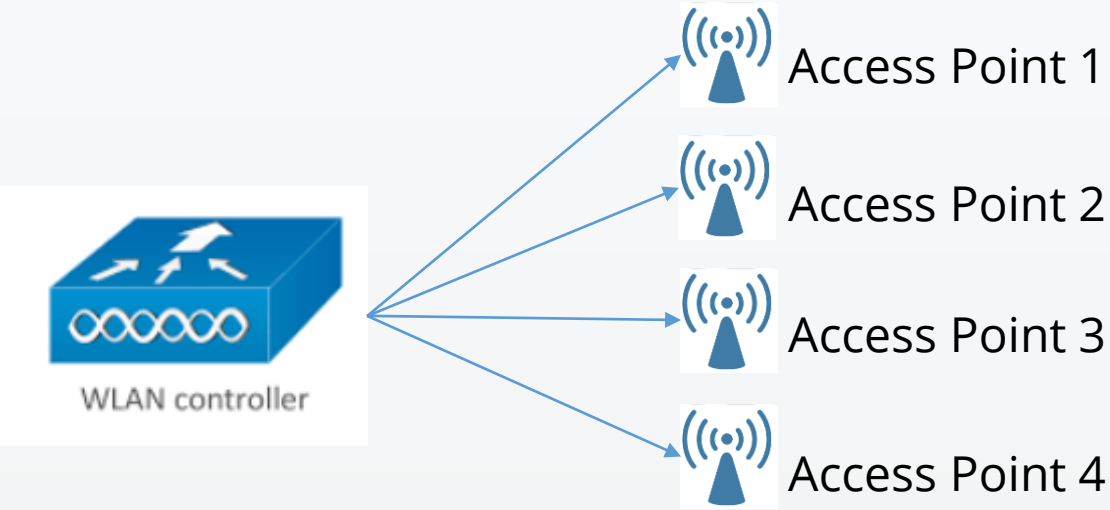
Macro		Value	
{$LLD_REMOTE_CHECK}	⇒	10.74.181.89:8080,10.74.181.80:8080,10.74	Remove

Add

Update | Clone | Full clone | Delete | Cancel

Remote TCP connection discovery: Remote TCP service to 10.74.181.79:8080]	Triggers 3	net.tcp.service.perf[tcp,10.74.181.79,8080]
Remote TCP connection discovery: Remote TCP service to 10.74.181.80:8080]	Triggers 3	net.tcp.service.perf[tcp,10.74.181.80,8080]
Remote TCP connection discovery: Remote TCP service to 10.74.181.81:8080]	Triggers 3	net.tcp.service.perf[tcp,10.74.181.81,8080]
Remote TCP connection discovery: Remote TCP service to 10.74.181.93:8080]	Triggers 3	net.tcp.service.perf[tcp,10.74.181.93,8080]

All Access Point data is collected on controller only



- WLAN controller returns full SNMP info about every Access Point
- Access Points are used by different customers
- We want to ping the physical Access Points also

Create discovery rule and host prototype to gather Access Point data

ZABBIX SUMMIT 18

Name: WLC AP data

Type: SNMPv2 agent

Key: bsnAPName

SNMP OID: discovery[{#AP_NAME},1.3.6.1.4.1.14179.2.2.1.1.3,{#AP_IP_ADDRESS},1.3.6.1.4.

SNMP community: {$SNMP_COMMUNITY}

Host | Groups | Templates | Host inventory | Encryption

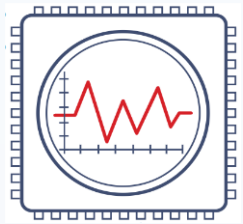
Host name: {#AP_IP_ADDRESS}

Visible name: {#AP_NAME}

Create enabled: ☑

Update | Clone | Delete | Cancel

Link the same template to all hosts



ZABBIX SUMMIT 18

Host | Groups | Templates | IPMI | Macros | Host inventory | Encryption

Linked templates

Name
Template SNMP Cisco Wireless Access Point

Update | Clone | Delete | Cancel

Data is still gathered from WLC

ZABBIX SUMMIT 18

Discovery rule | Filters

* Name	WLC AP data
Type	SNMPv2 agent
* Key	bsnAPName
* SNMP OID	discovery[{#AP_NAME},1.3.6.1.4.1.14179.2.2.1.1.3,{#AP_IP_ADDRESS},1.3.6.1.4.
* SNMP community	{$SNMP_COMMUNITY}

Actual AP data is filtered out

ZABBIX SUMMIT '18

Discovery rule | Filters

Filters

Label	Macro		Regular expression	Action
A	{#AP_NAME}	matches	^{HOST.NAME}$	Remove

Add

Update | Clone | Delete | Cancel

Use **LLD filters** to filter out only data for particular access point

Result – hosts generated for each Access Point automatically

ZABBIX SUMMIT 18

	Applications	Items	Triggers	Graphs	Discovery	Web	Interface
WLC AP data: APc47d.4f3a.8181	Applications 2	Items 8	Triggers 1	Graphs 1	Discovery 1	Web	172.16.0.4: 161
WLC AP data: APc84c.75ee.212d	Applications 2	Items 8	Triggers 1	Graphs 1	Discovery 1	Web	172.16.0.4: 161
WLC AP data: APc84c.757a.11a7	Applications 2	Items 8	Triggers 1	Graphs 1	Discovery 1	Web	172.16.0.4: 161
WLC AP data: APd0d0.fd2e.17ce	Applications 2	Items 8	Triggers 1	Graphs 1	Discovery 1	Web	172.16.0.4: 161
WLC AP data: PMO	Applications 2	Items 8	Triggers 1	Graphs 1	Discovery 1	Web	172.16.0.4: 161

Name ▲	Last check	Last value
AP Data (6 Items)		
APc47d.4f3a.8181 Ap If No Of Users 2.4 GHz	2018-09-28 01:12:12	3 users
APc47d.4f3a.8181 Ap If No Of Users 5 GHz	2018-09-28 01:12:12	0 users
APc47d.4f3a.8181 AP If Phy Channel Number 2.4 GHz	2018-09-28 01:10:12	1
APc47d.4f3a.8181 AP If Phy Channel Number 5 GHz	2018-09-28 01:10:12	36
APc47d.4f3a.8181 AP Ip Address	2018-09-28 00:35:12	172.16.0.49

Benefits of using host prototypes

ZABBIX SUMMIT '18

- Managed by Zabbix **LLD internal mechanism**
  - Added as needed
  - Removed or changed automatically
- Templates and host groups are assigned **automatically**
- Host group names can be generated **dynamically** from LLD macros
- Can assign permissions to **restrict access** only to part of the data using host groups and user groups

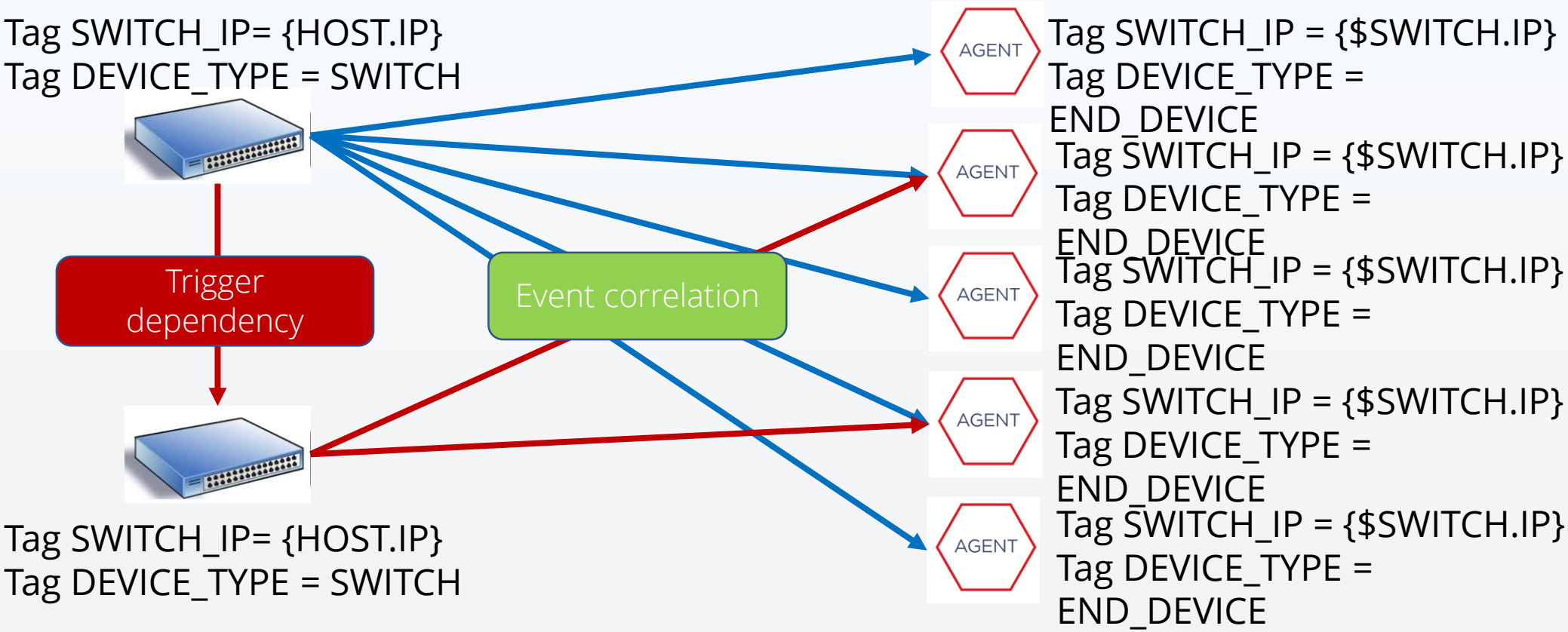
5. Event Correlation

Zabbix Tips and Tricks

Suppress device alarms when switch goes down

- All end devices are connected to **different switches**
- Dependencies can be used, but there are **too many devices** connected to each switch
- Devices can be **moved between switches**, dependencies must be relinked in this case

Use global event correlation with tags



{$SWITCH.IP} macro is generated by API script with device IP to MAC resolution via ARP tables, which are gathered by Zabbix

Create global event correlation rule

ZABBIX SUMMIT 18

* Name	Switch and Device correlation
Type of calculation	And/Or ▼ A and B and C

* Conditions

Label	Name	Action
A	Old event tag *SWITCH_IP* equals new event tag *SWITCH_IP*	Remove
B	Old event tag *DEVICE_TYPE* equals *SWITCH*	Remove
C	New event tag *DEVICE_TYPE* equals *END_DEVICE*	Remove

Operations

Details	Action
Close new event	Remove

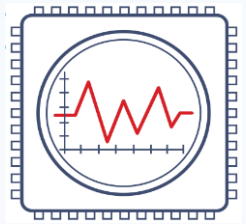
6. Action management

Zabbix Tips and Tricks

Postpone alerts for 30 minutes

ZABBIX SUMMIT '18

- When problem happens, actions are executed immediately by default
- “Fake” operation steps can be created, but this does not look nice

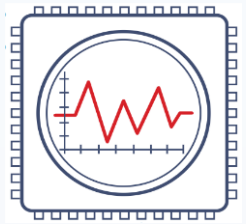


Use Zabbix built in scheduler

ZABBIX SUMMIT 18

OPERATION STEPS	OPERATIONS	START IN	DURATION
Operation step 1	Do nothing	Immediately	default
Operation step 2	Send message to operator	After 30 minutes	default
Operation step 3	Send message to manager	After 1 hour	1 hour
Operation step 4	Execute remote command	After 2 hours	default

Default step duration: 30 min



Zabbix built-in scheduler in details

ZABBIX SUMMIT '18

Action | Operations | Recovery operations | Update operations

* Default operation step duration: 30m

Default subject: Problem: {EVENT.NAME}

Default message:

```
Problem started at {EVENT.TIME} on {EVENT.DATE}
Problem name: {EVENT.NAME}
Host: {HOST.NAME}
Severity: {EVENT.SEVERITY}

Original problem ID: {EVENT.ID}
{TRIGGER.URL}
```

Pause operations for suppressed problems: ☑

Operations

Steps	Details	Start in	Duration	Action
2	**Send message to user groups:** Operators via all media	00:30:00	Default	Edit Remove
3	**Send message to user groups:** Managers via all media	01:00:00	1h	Edit Remove
4	**Run remote commands on current host**	02:00:00	Default	Edit Remove

New

* At least one operation, recovery operation or update operation must exist.

Update | Clone | Delete | Cancel

THANK YOU !

ZABBIX SUMMIT '18