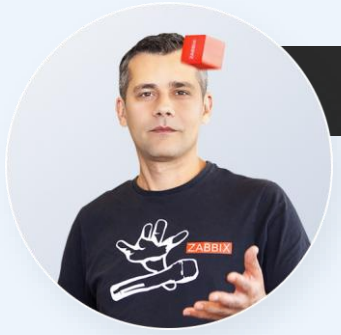
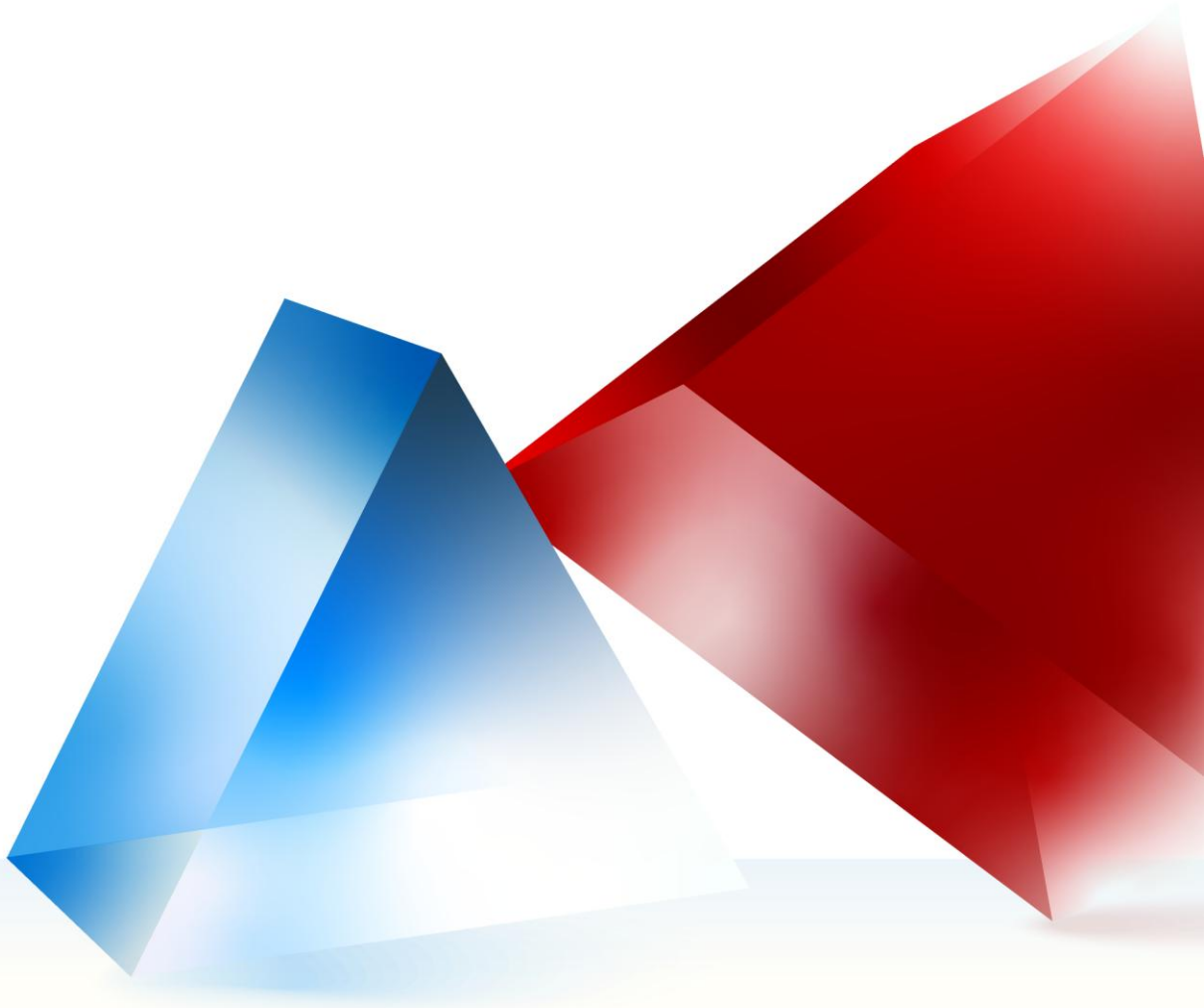
From Basics to Brilliance: LLD in action



Kaspars Mednis

Training Project Manager

What is LLD?



What is Low Level Discovery?

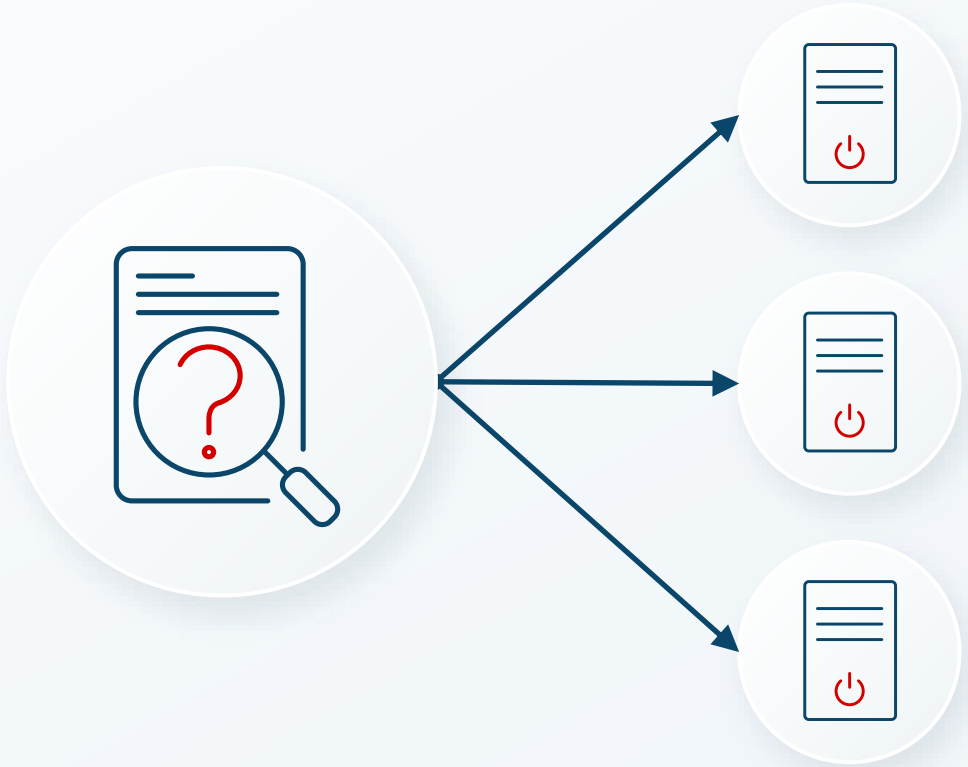
LLD is used to automatically discover entities inside a host:

- Network interfaces
- Filesystems
- CPU cores
- Windows and Linux services
- Physical disk drives
- Database tables
- And much more...

Low Level Discovery Hosts

LLD can even create new hosts from the discovery results:

- Virtual machines
- Docker containers
- Kubernetes pods
- Cloud resources
- And much more...



LLD in Zabbix

Most official Zabbix templates use LLD rules:

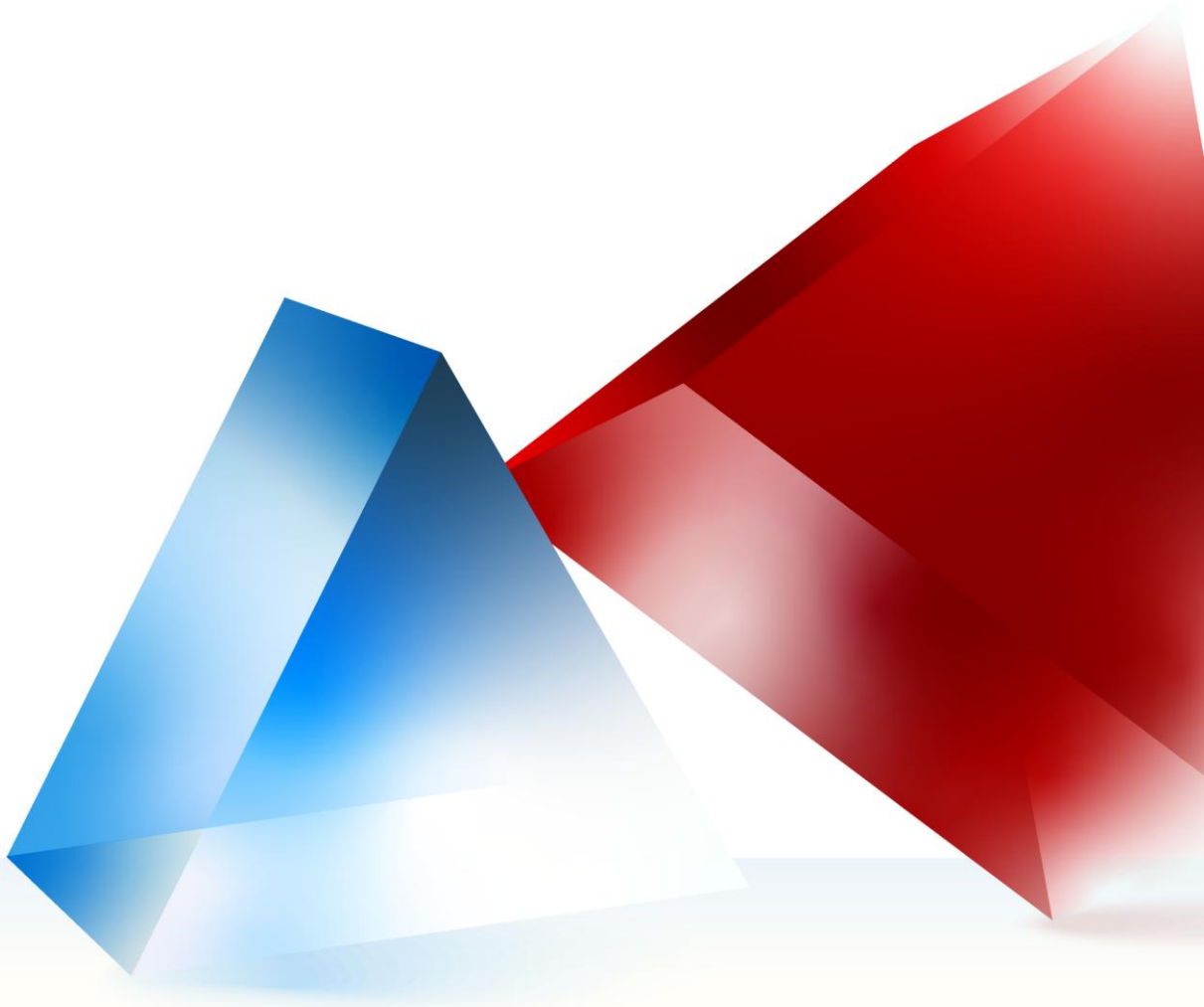
Operating systems	Linux, Windows, HP-UX
SNMP devices	Switches, routers, UPS, iLO
Databases	MySQL, PostgreSQL, Oracle
Cloud providers	AWS, Azure, GCP
VMWare	Hypervisors, Virtual Machines, Datastores

LLD in templates

Name ▲	Hosts	Items	Triggers	Graphs	Dashboards	Discovery	Web	Vendor	Version
Apache Tomcat by JMX	Hosts	Items 1	Triggers 1	Graphs	Dashboards 1	Discovery 4	Web	Zabbix	7.4-0
APC Smart-UPS 3000 XLM by SNMP	Hosts	Items 26	Triggers 22	Graphs 3	Dashboards 1	Discovery 6	Web	Zabbix	7.4-1
AWS by HTTP	Hosts	Items	Triggers	Graphs	Dashboards	Discovery 7	Web	Zabbix	7.4-4
Cisco IOS by SNMP	Hosts	Items 23	Triggers 8	Graphs	Dashboards 1	Discovery 8	Web	Zabbix	7.4-2
DELL PowerEdge R820 by SNMP	Hosts	Items 28	Triggers 10	Graphs 1	Dashboards 1	Discovery 12	Web	Zabbix	7.4-3
FortiGate by SNMP	Hosts	Items 53	Triggers 11	Graphs 5	Dashboards 3	Discovery 9	Web	Zabbix	7.4-2
HP iLO by SNMP	Hosts	Items 16	Triggers 9	Graphs	Dashboards	Discovery 13	Web	Zabbix	7.4-2
Linux by Zabbix agent	Hosts 1	Items 43	Triggers 15	Graphs 8	Dashboards 3	Discovery 3	Web	Zabbix	7.4-2
VMware FQDN	Hosts	Items 7	Triggers 2	Graphs	Dashboards 1	Discovery 5	Web	Zabbix	7.4-2
Windows by Zabbix agent	Hosts	Items 34	Triggers 13	Graphs 5	Dashboards 3	Discovery 4	Web	Zabbix	7.4-2

LLD rules

How LLD Works



Prototypes

Low level discovery rules use the entity prototypes (blueprints)

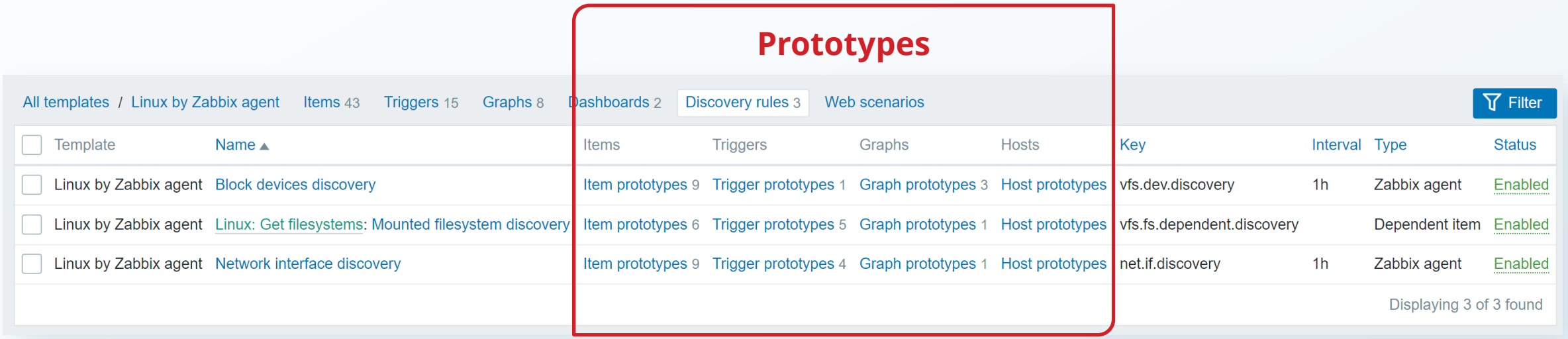
Item prototypes

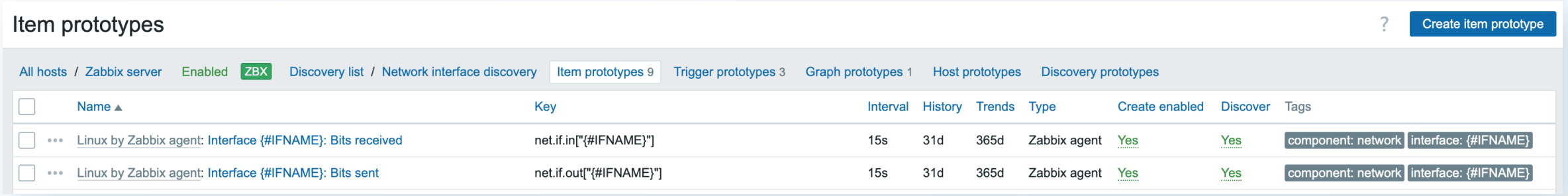
Trigger prototypes

Graph prototypes

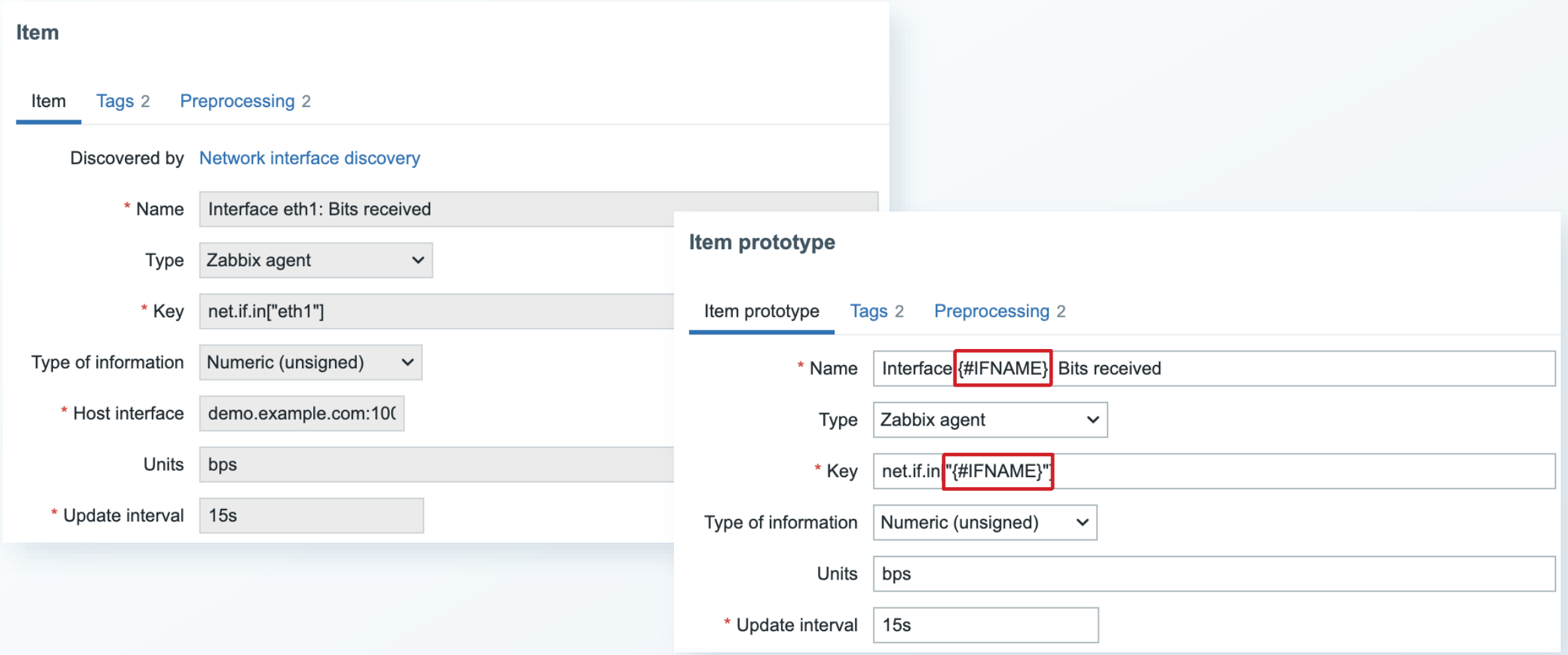
Host prototypes

LLD rules and prototypes





Item vs Item prototype



LLD rule

- Traffic in on **{#IFNAME}**
- Traffic out on **{#IFNAME}**
- High traffic in on **{#IFNAME}**
- High traffic out on **{#IFNAME}**
- Traffic graph on **{#IFNAME}**

Network interface **eth0**

- Traffic in on **eth0**
- Traffic out on **eth0**
- High traffic in on **eth0**
- High traffic out on **eth0**
- Traffic graph on **eth0**

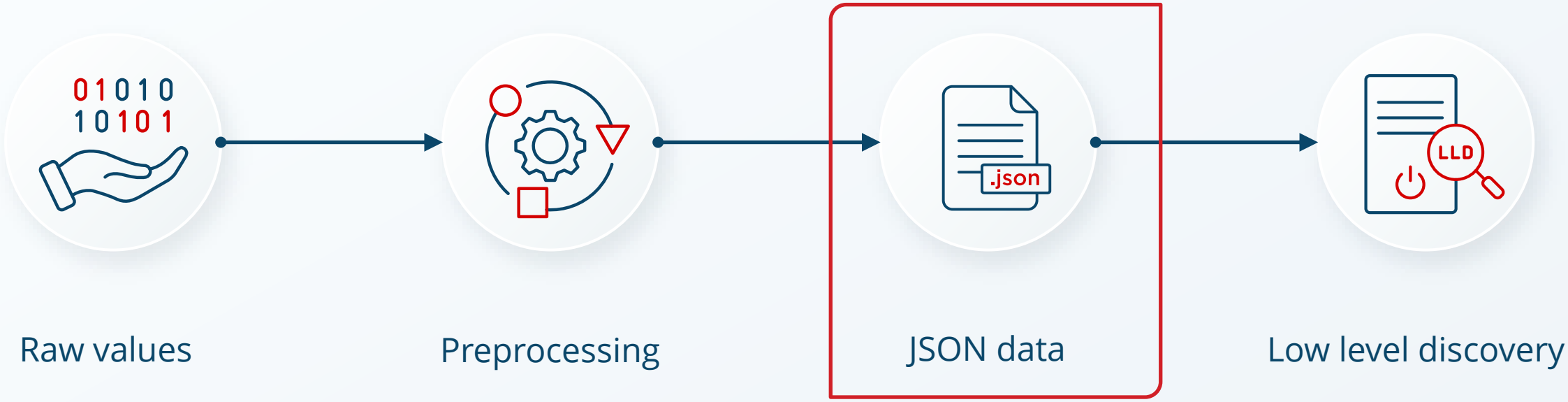
Network interface **eth1**

- Traffic in on **eth1**
- Traffic out on **eth1**
- High traffic in on **eth1**
- High traffic out on **eth1**
- Traffic graph on **eth1**

LLD data flow

LLD is based on data in JSON format:

- Data collector collects data for discovery (any format)
- Preprocessing transforms raw data to JSON
- Zabbix creates entities based on the JSON structure



JSON data structure

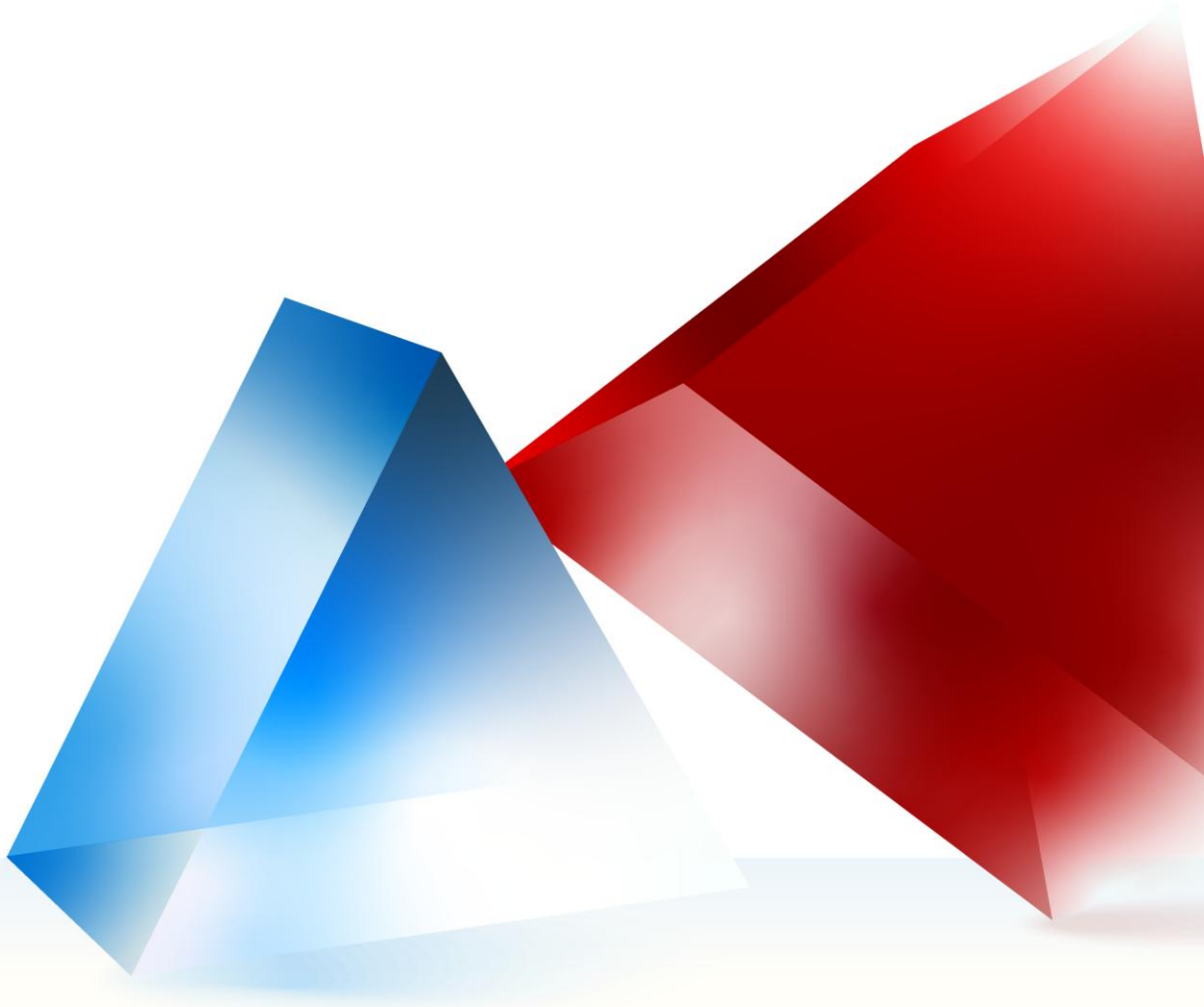
JSON data in Zabbix format

```
[
   {
      "{#FSNAME}": "/sys",
      "{#FSTYPE}": "sysfs"
   },{
      "{#FSNAME}": "/",
      "{#FSTYPE}": "xfs"
   },{
      "{#FSNAME}": "/root",
      "{#FSTYPE}": "rootfs"
   }
]
```

Free-form JSON data

```
[
  {
    "TABLE_NAME": "acknowledges",
    "DATA_LENGTH": "16384",
    "INDEX_LENGTH": "49152"
  },
  {
    "TABLE_NAME": "actions",
    "DATA_LENGTH": "16384",
    "INDEX_LENGTH": "32768"
  }
]
```

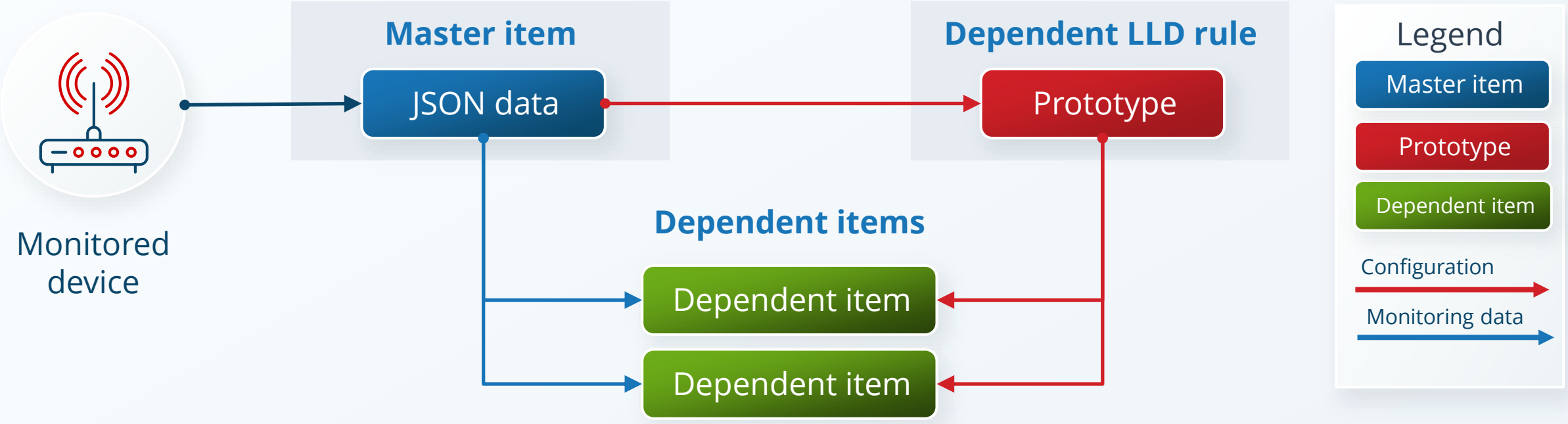
Dependent LLD



Dependent LLD

Dependent LLD utilizes the same JSON data for two purposes:

- Configuration of items and triggers discovered by LLD rule
- The actual monitoring data collected by items



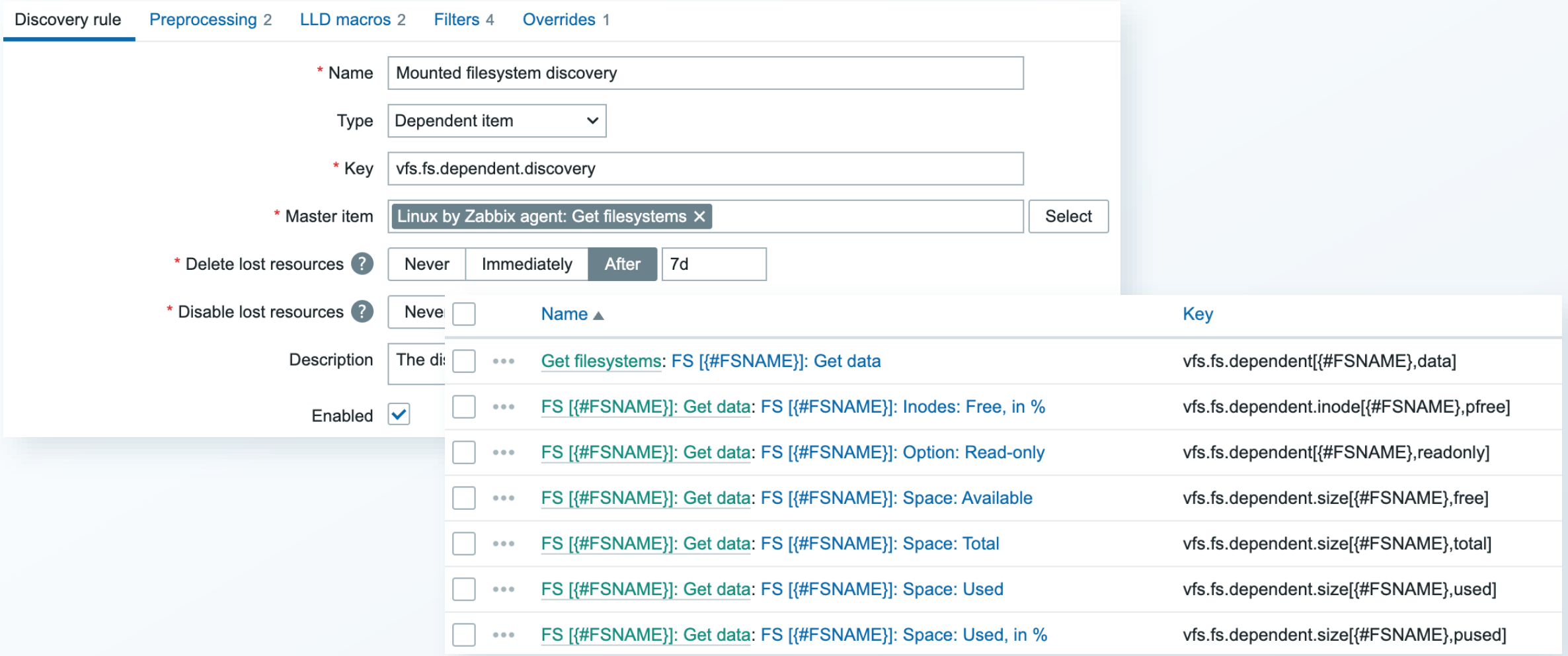
JSON Example

Legend

Configuration data

Item values

How it looks in Zabbix



BENEFITS

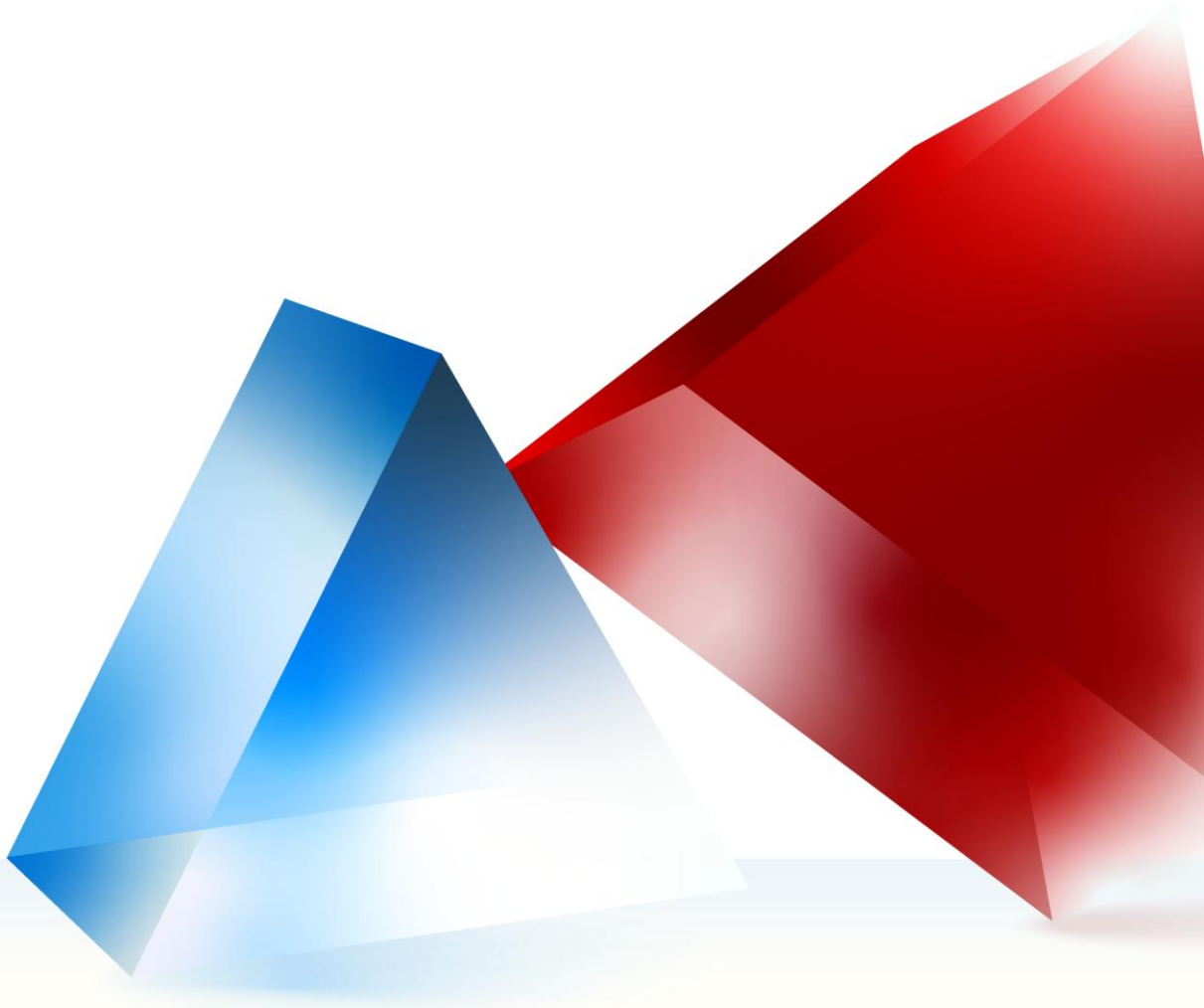
Dependent LLD rules provides lots of benefits:

- All required data is collected in a single request
- Fewer connections are opened from Zabbix to monitored devices
- All monitoring data are collected at the same time and are consistent

Keep in mind:

- Dependent items use preprocessors to extract data from a master item
- Dependent LLD rule executes every time data is collected by the master item
- Use discard unchanged data with heartbeat preprocessing step on LLD rules

Nested LLD



Pre-7.4 limitations

Previous Zabbix versions were limited to a single LLD level:

- Enough in most simple use cases
- Impossible to implement more complex scenarios

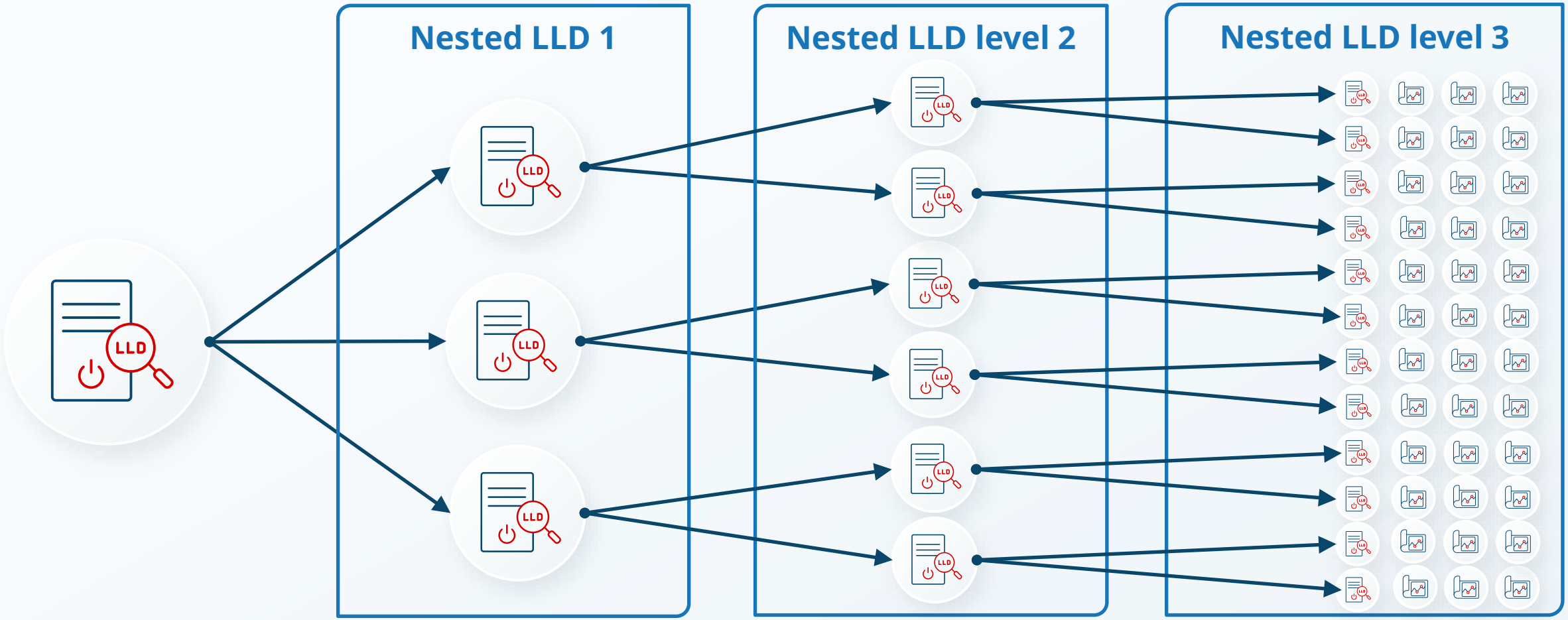
Nested LLD

A parent LLD rule discovers entities and spawns child LLD rules:

- More complex scenarios can be implemented
- Allows users to fully utilize multi-level JSON data

Multiple levels of nested LLD

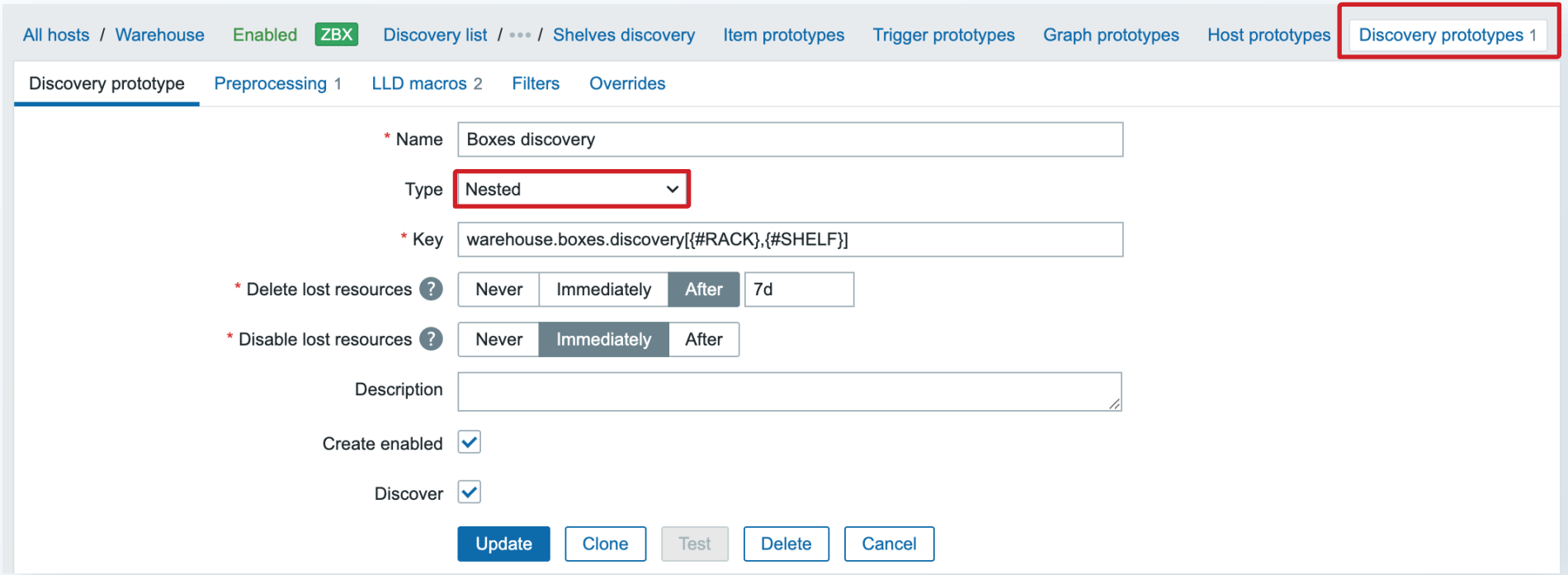
Unlimited levels of nested LLD rules are supported:



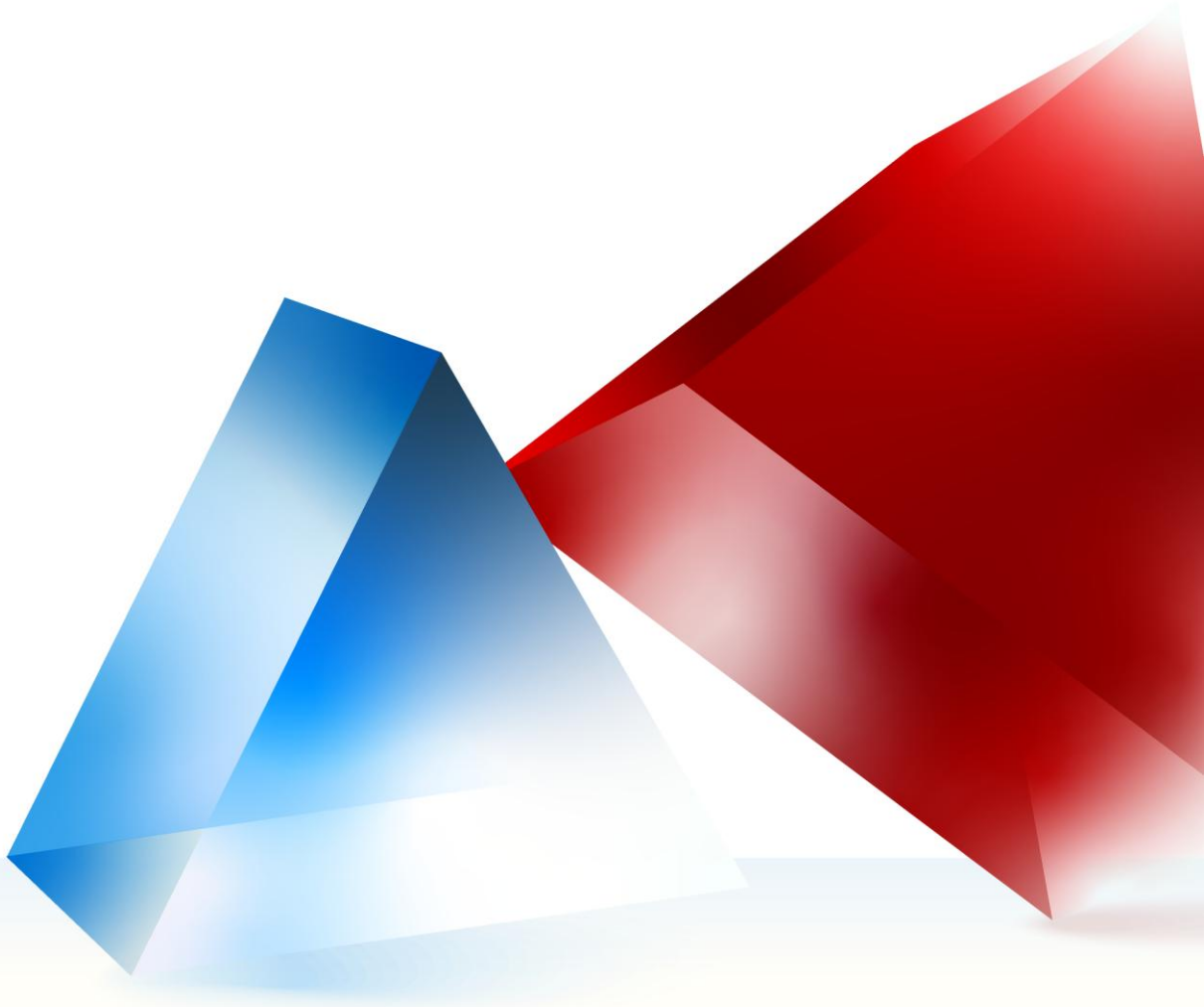
How nested LLD works

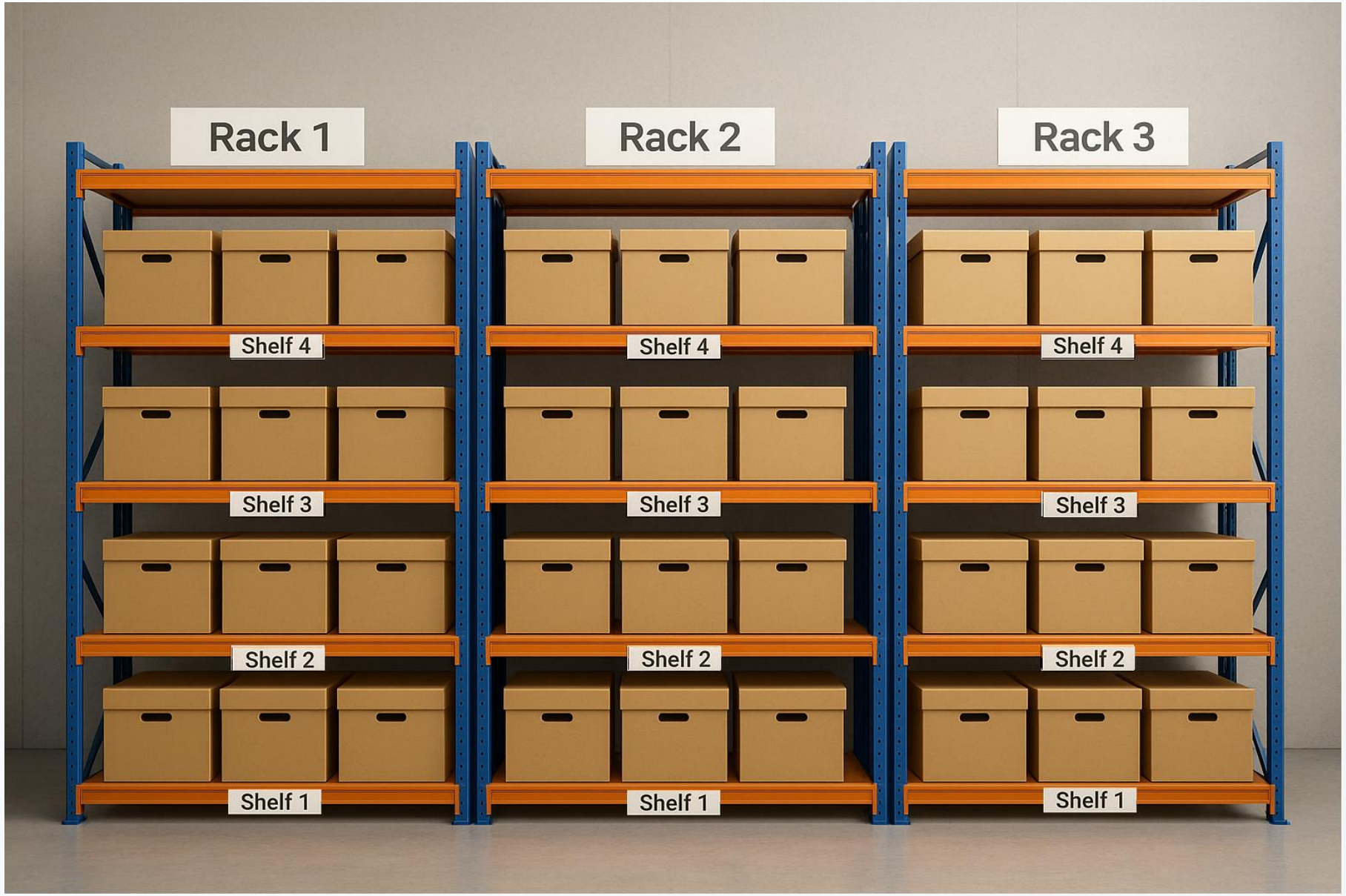
Each nested LLD level inherits its portion of JSON data from the previous level:

- Nested LLD rules are created using discovery prototypes
- Discovery prototype have a new "nested" type



Example LLD scenario

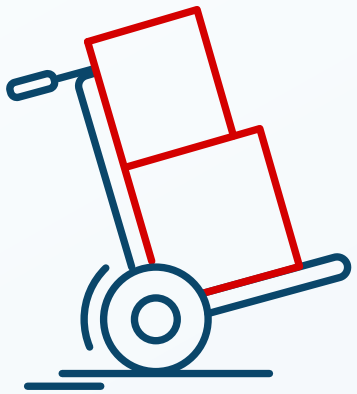


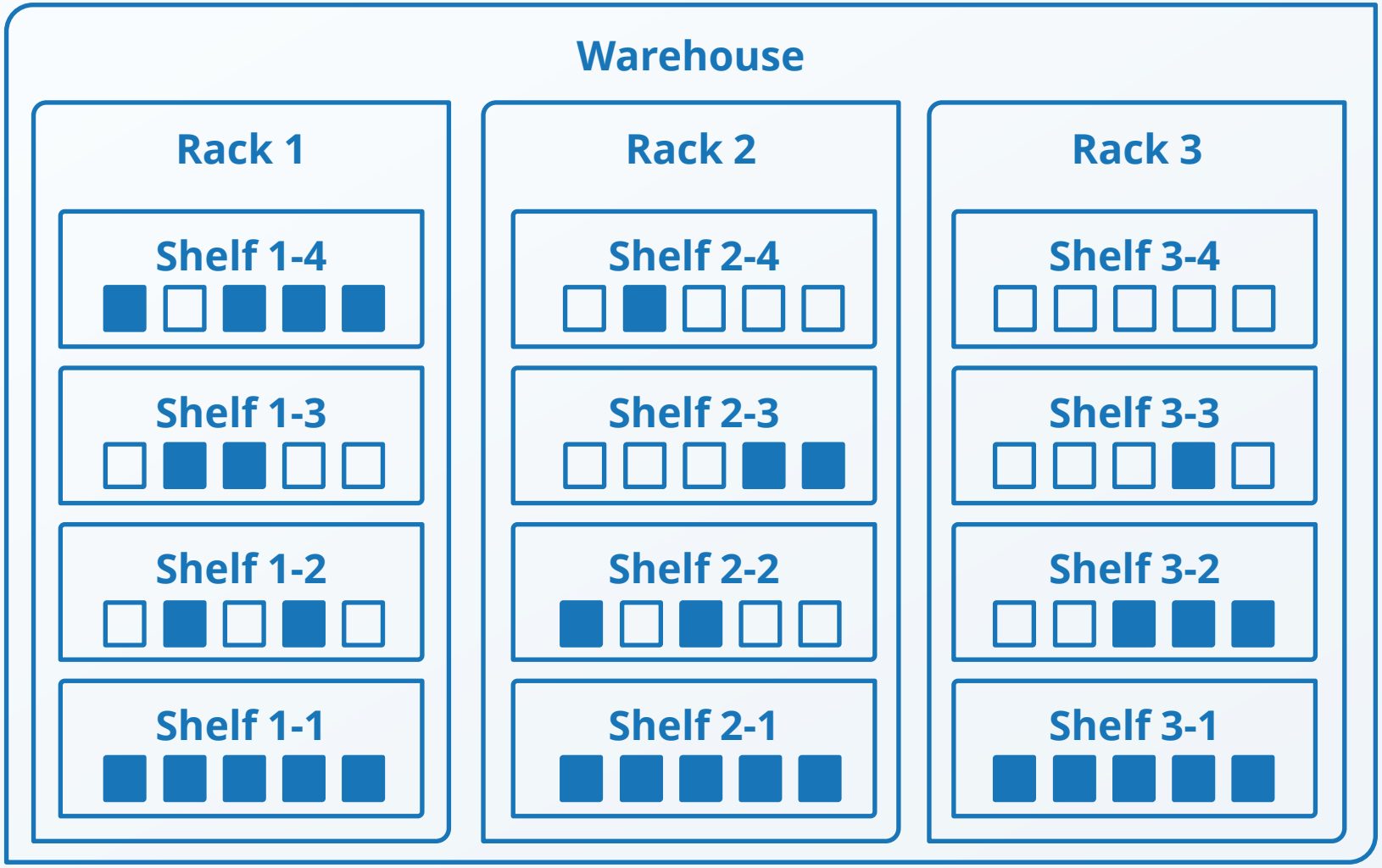


Example scenario

Warehouse

- LLD Level 1: Racks
- LLD Level 2: Shelves
- LLD Level 3: Boxes





Example scenario JSON

Data is represented as a hierarchical JSON:

- Level 1: Racks
- Level 2: Shelves
- Level 3: Boxes


```
{
  "warehouse": {
    "racks": [
      {
        "rack": "Rack 1",
        "shelves": [
          {
            "shelf": "Shelf 1-4",
            "boxes": [
              {
                "box": "Box 1-4-1",
                "content": 1
              },
              {
                "box": "Box 1-4-2",
                "content": 0
              },
              {
                "box": "Box 1-4-3",
                "content": 1
              },
              {
                "box": "Box 1-4-4",
                "content": 0
              },
              {
                "box": "Box 1-4-5",
                "content": 0
              }
            ]
          },
```

Nested LLD Screenshot

Warehouse > Racks discovery > Shelves discovery > Boxes discovery

Host	Name ▲	Items	Triggers	Graphs	Hosts	Discovery rules	Key	Interval	Type
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 1,Shelf 1-1]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 1,Shelf 1-2]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 1,Shelf 1-3]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 1,Shelf 1-4]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 2,Shelf 2-1]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 2,Shelf 2-2]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 2,Shelf 2-3]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 2,Shelf 2-4]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 3,Shelf 3-1]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 3,Shelf 3-2]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 3,Shelf 3-3]		Nested
Warehouse	Shelves discovery: Boxes discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	warehouse.boxes.discovery[Rack 3,Shelf 3-4]		Nested
Warehouse	Warehouse data: Racks discovery	Item prototypes	Trigger prototypes	Graph prototypes	Host prototypes	Discovery prototypes 1	warehouse.racks.discovery		Dependent item
Warehouse	Racks discovery: Shelves discovery	Item prototypes	Trigger prototypes	Graph prototypes	Host prototypes	Discovery prototypes 1	warehouse.shelves.discovery["Rack 1"]		Nested
Warehouse	Racks discovery: Shelves discovery	Item prototypes	Trigger prototypes	Graph prototypes	Host prototypes	Discovery prototypes 1	warehouse.shelves.discovery["Rack 2"]		Nested
Warehouse	Racks discovery: Shelves discovery	Item prototypes	Trigger prototypes	Graph prototypes	Host prototypes	Discovery prototypes 1	warehouse.shelves.discovery["Rack 3"]		Nested

Nested LLD result

Host	Name ▲	Last check	Last value	Change
Warehouse	Box content [Rack 1/Shelf 1-1/Box 1-1-1]	16s	0 Box	-1 Box
Warehouse	Box content [Rack 1/Shelf 1-1/Box 1-1-2]	16s	0 Box	
Warehouse	Box content [Rack 1/Shelf 1-1/Box 1-1-3]	16s	1 Box	
Warehouse	Box content [Rack 1/Shelf 1-1/Box 1-1-4]	16s	0 Box	
Warehouse	Box content [Rack 1/Shelf 1-1/Box 1-1-5]	16s	1 Box	
Warehouse	Box content [Rack 1/Shelf 1-2/Box 1-2-1]	16s	0 Box	
Warehouse	Box content [Rack 1/Shelf 1-2/Box 1-2-2]	16s	1 Box	
Warehouse	Box content [Rack 1/Shelf 1-2/Box 1-2-3]	16s	0 Box	
Warehouse	Box content [Rack 1/Shelf 1-2/Box 1-2-4]	16s	1 Box	
Warehouse	Box content [Rack 1/Shelf 1-2/Box 1-2-5]	16s	0 Box	
Warehouse	Box content [Rack 1/Shelf 1-3/Box 1-3-1]	16s	1 Box	
Warehouse	Box content [Rack 1/Shelf 1-3/Box 1-3-2]	16s	0 Box	
Warehouse	Box content [Rack 1/Shelf 1-3/Box 1-3-3]	16s	1 Box	
Warehouse	Box content [Rack 1/Shelf 1-3/Box 1-3-4]	16s	0 Box	
Warehouse	Box content [Rack 1/Shelf 1-3/Box 1-3-5]	16s	1 Box	+1 Box
Warehouse	Box content [Rack 1/Shelf 1-4/Box 1-4-1]	16s	0 Box	
Warehouse	Box content [Rack 1/Shelf 1-4/Box 1-4-2]	16s	1 Box	
Warehouse	Box content [Rack 1/Shelf 1-4/Box 1-4-3]	16s	0 Box	
Warehouse	Box content [Rack 1/Shelf 1-4/Box 1-4-4]	16s	1 Box	+1 Box

Official Zabbix template

The first official Zabbix template with nested LLD has been developed:

- Template Proxmox VE by HTTP
- Will be officially available with Zabbix 8.0
- https://support.zabbix.com/browse/ZBXNEXT-10051

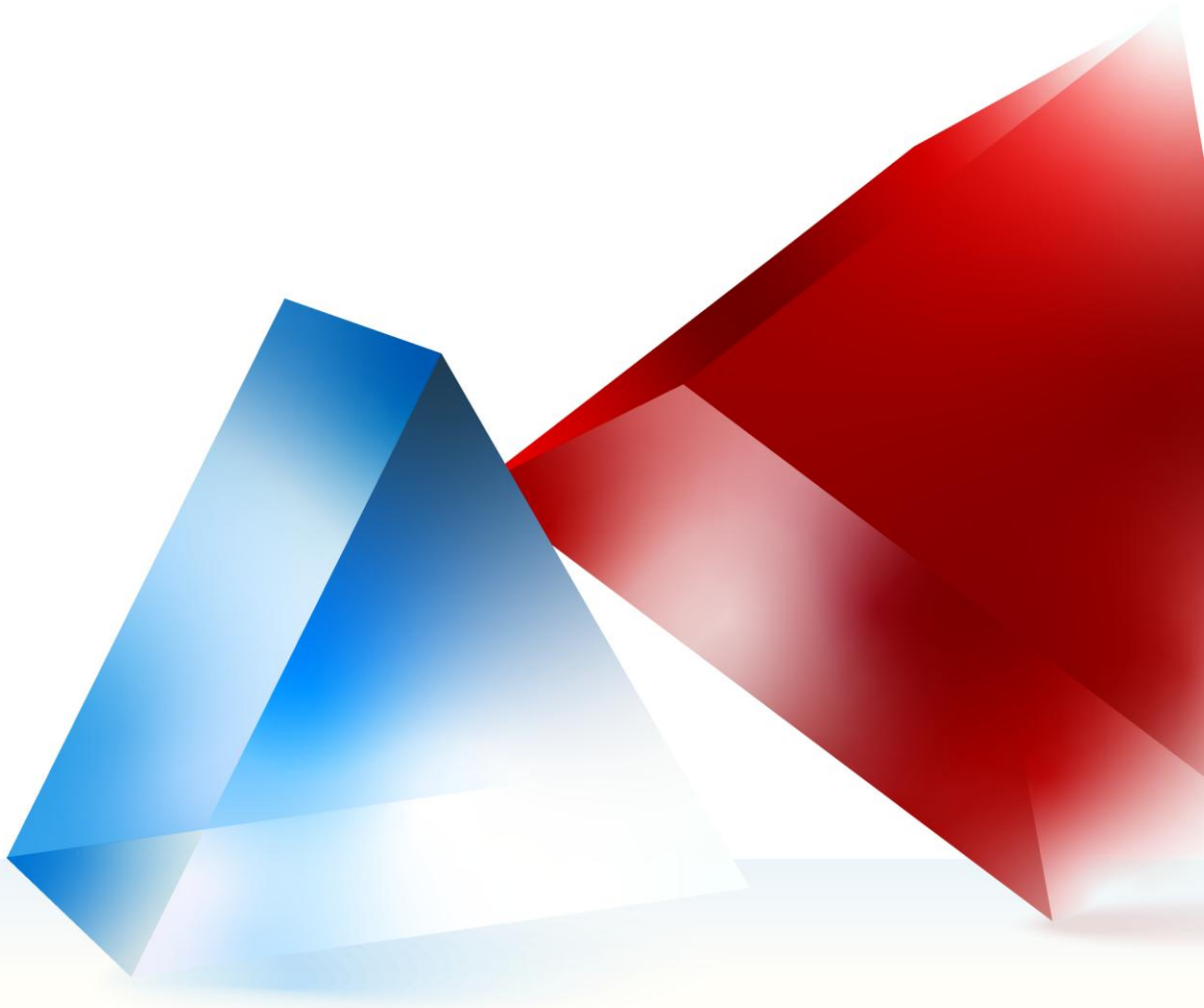
Template	Name ▲	Items	Triggers	Graphs	Hosts	Discovery rules	Key	Interval	Type	Status
Proxmox VE by HTTP	Cluster: Get resources: Proxmox LXC discovery	Item prototypes 18	Trigger prototypes 7	Graph prototypes 6	Host prototypes	Discovery prototypes	proxmox_ve.node.lxc.discovery[{#NODE.ID}]		Dependent item	Enabled
Proxmox VE by HTTP	Cluster: Get node data: Proxmox nodes discovery	Item prototypes 20	Trigger prototypes 9	Graph prototypes 4	Host prototypes	Discovery prototypes 3	proxmox_ve.nodes.discovery		Dependent item	Enabled
Proxmox VE by HTTP	Cluster: Get resources: Proxmox QEMU discovery	Item prototypes 17	Trigger prototypes 7	Graph prototypes 4	Host prototypes	Discovery prototypes 3	proxmox_ve.node.qemu.discovery[{#NODE.ID}]		Dependent item	Enabled
Proxmox VE by HTTP	Cluster: Get resources: Proxmox shared storage discovery	Item prototypes 5	Trigger prototypes	Graph prototypes	Host prototypes	Discovery prototypes	proxmox_ve.shared.storage.discovery		Dependent item	Enabled

All templates / Proxmox VE by HTTP Discovery list / Proxmox nodes discovery Item prototypes 20 Trigger prototypes 9 Graph prototypes 4 Host prototypes Discovery prototypes 3

Name ▲	Items	Triggers	Graphs	Hosts	Discovery rules	Key	Interval	Type	Create enabled	Discover
Node [{#NODE.NAME}]: Certificate discovery	Item prototypes 6	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	proxmox_ve.node.certificates.discovery[{#NODE.ID}]	{$PVE.PARAMS.INTERVAL.CERT}	HTTP agent	Yes	Yes
Node [{#NODE.NAME}]: Disks discovery	Item prototypes 6	Trigger prototypes 1	Graph prototypes	Host prototypes	Discovery prototypes	proxmox_ve.node.disks.discovery[{#NODE.ID}]	{$PVE.PARAMS.INTERVAL.DISK}	HTTP agent	Yes	Yes
Node [{#NODE.NAME}]: Storage discovery	Item prototypes 6	Trigger prototypes 2	Graph prototypes 1	Host prototypes	Discovery prototypes	proxmox_ve.node.storage.discovery[{#NODE.ID}]	{$PVE.PARAMS.INTERVAL.STORAGE}	HTTP agent	Yes	Yes

Displaying 3 of 3 found

Workshop info



Workshop

13:00 60 m	**Nested LLDs**	Aleksandrs Petrovs-Gavrilovs, Zabbix Trainer, Zabbix	Alfa	Technical

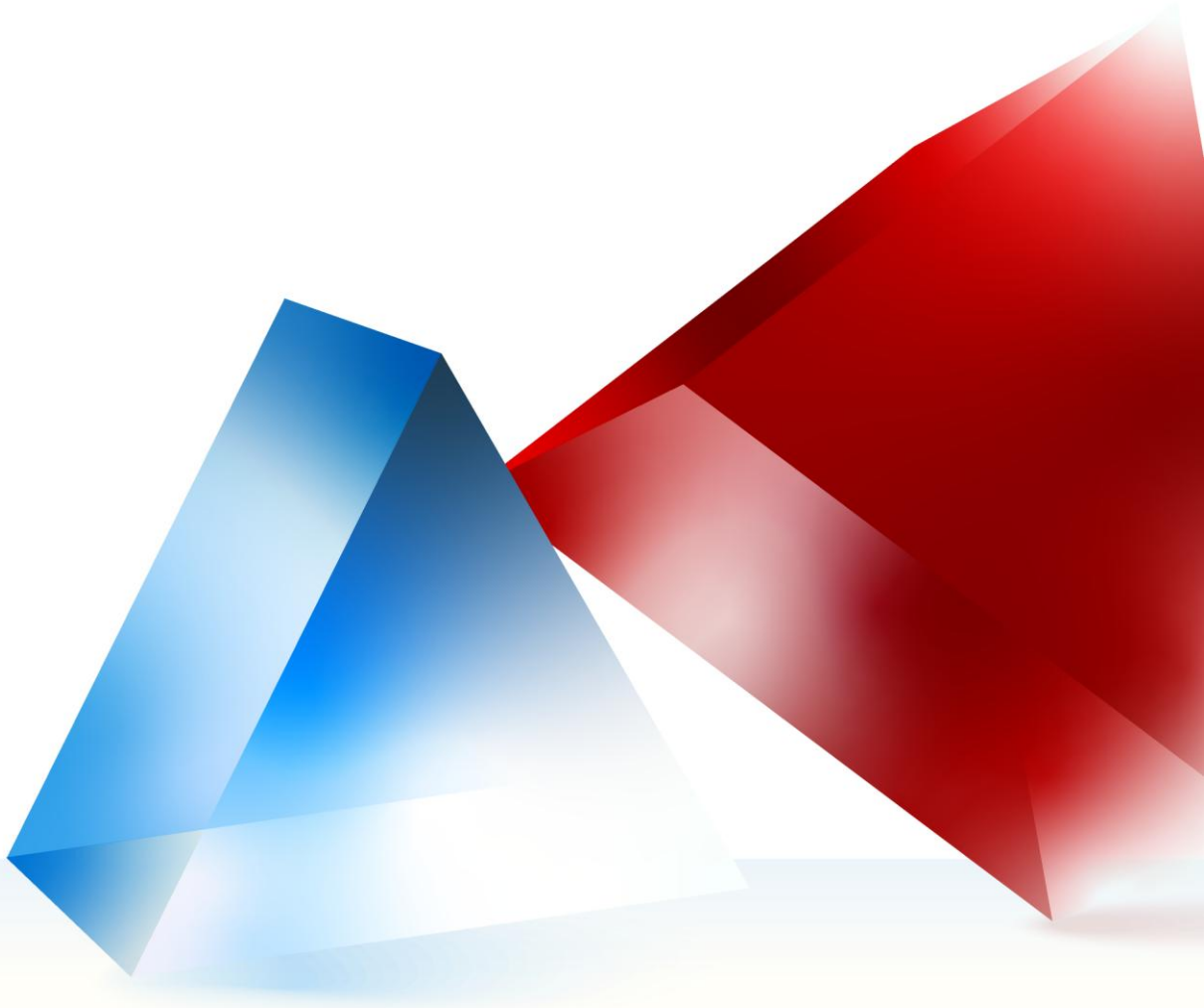
We'll guide you through a scenario focused on discovering and monitoring a warehouse, including:

- Multiple racks
- Multiple shelves per rack
- Multiple storage boxes per shelf

You must bring your own laptop to the workshop:

- Individual virtual machines will be provided by Zabbix

Training info

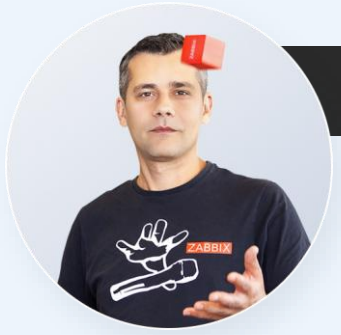


New one-day course

A new one-day course on Low Level Discovery (LLD) is coming soon!

- Available as both live training and a Zabbix Academy course
- Covers every aspect of LLD, from the basics to dependent LLD and nested LLD
- Learn how to create hosts from host prototypes
- Discover how to transform raw data into JSON for LLD discovery of nearly anything
- Includes live examples with Zabbix Agent, SNMP, and database discovery

Thank You!



Kaspars Mednis

Training Project Manager