Federated Observability

with ZABBIX and a telemetry bus

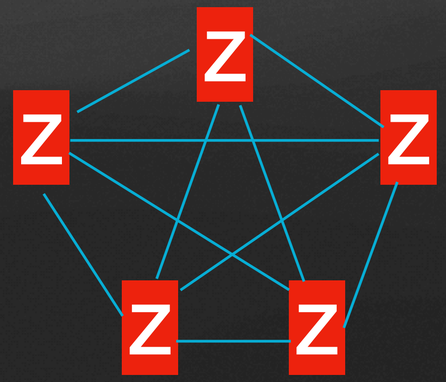
Vladimir Ulogov

vladimir.ulogov@amadeus.com

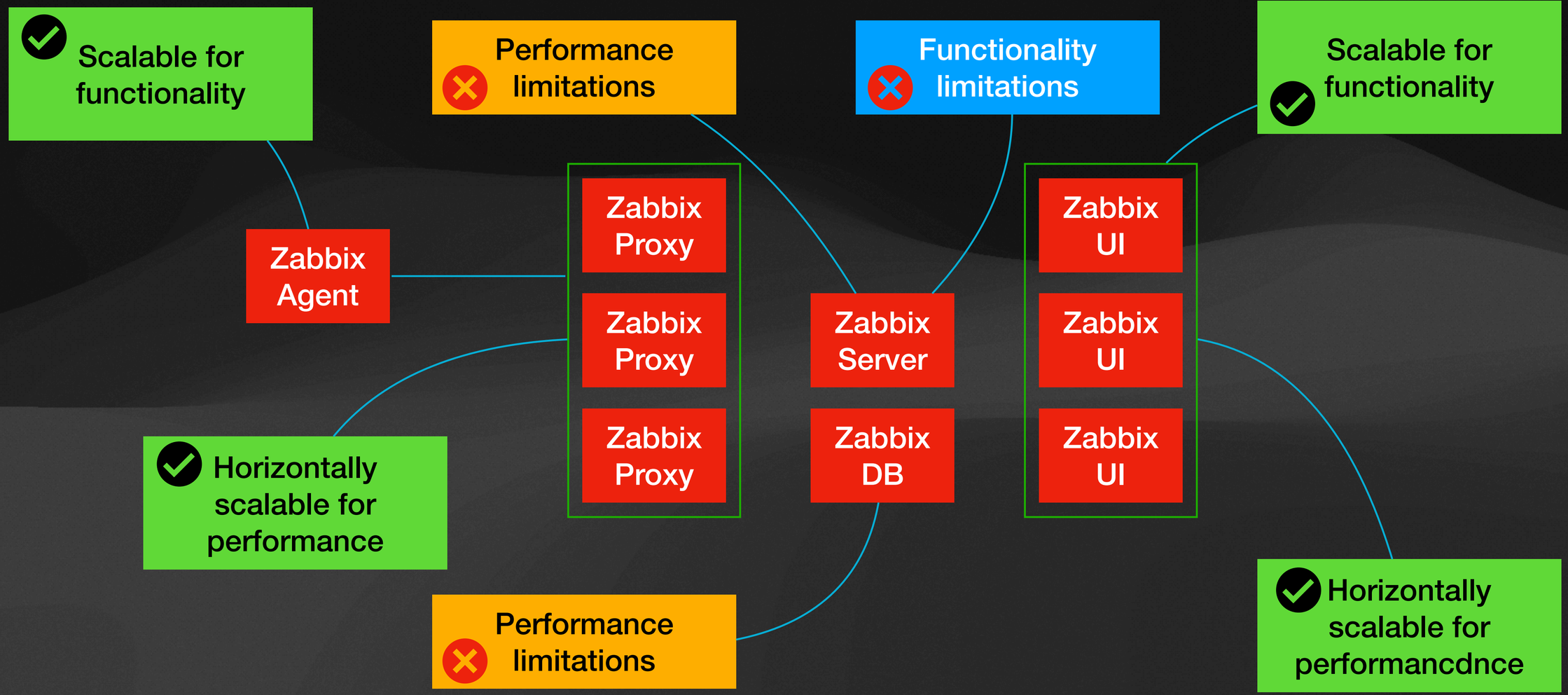
ABOUT

In this presentation, I will address strategies for expanding the capabilities of Zabbix. This entails enhancing scalability, broadening functionality, and refining the management, processing, and analysis of telemetry and observability data to align with specific operational demands and requirements.

The concept of "federated observability" entails consolidating the capabilities inherent in multiple platform instances, processing pipelines, and bespoke computation and analysis to unify their impact.



USE CASE


Scalable for functionality
Performance limitations
Functionality limitations
Scalable for functionality
Zabbix Agent
Zabbix Proxy
Zabbix Proxy
Zabbix Proxy
Zabbix Server
Zabbix DB
Zabbix UI
Zabbix UI
Zabbix UI
Horizontally scalable for performance
Performance limitations
Horizontally scalable for performancdnce

USE CASE

Where Zabbix shines

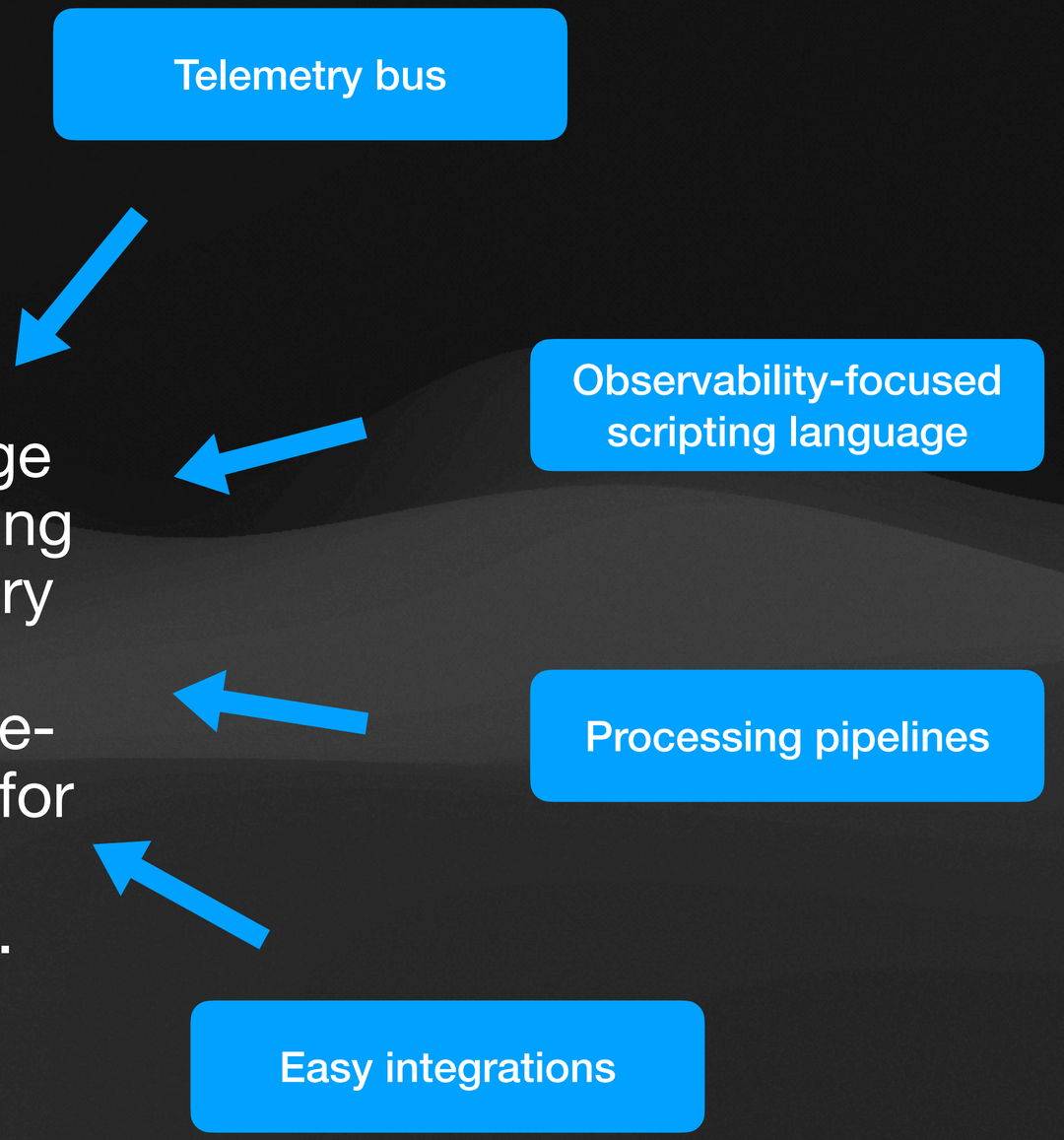
Reliable, one-shop, “include all bells and whistles” solution for Monitoring and Observability if you are satisfied with several concepts and limitations.

- Single server instance performance
- Agree to extend your functionality through Zabbix API.
- Data submission is through Zabbix Trapper or API.
- Collection functionality extension is traditionally based on Agent.
- Single DB instance performance

USE CASE

ZBUS extends Zabbix capabilities

Provide a reliable transmission and storage telemetry bus for enabling the same set of telemetry to multiple Zabbix instances. Provide a one-shop scripting solution for creating computational and analytical pipelines.



USE CASE

Zabbix is an excellent monitoring and observability platform deserving a better scalability.

Functional horizontal scalability and pipelines will simplify the adoption of advanced analysis practices.

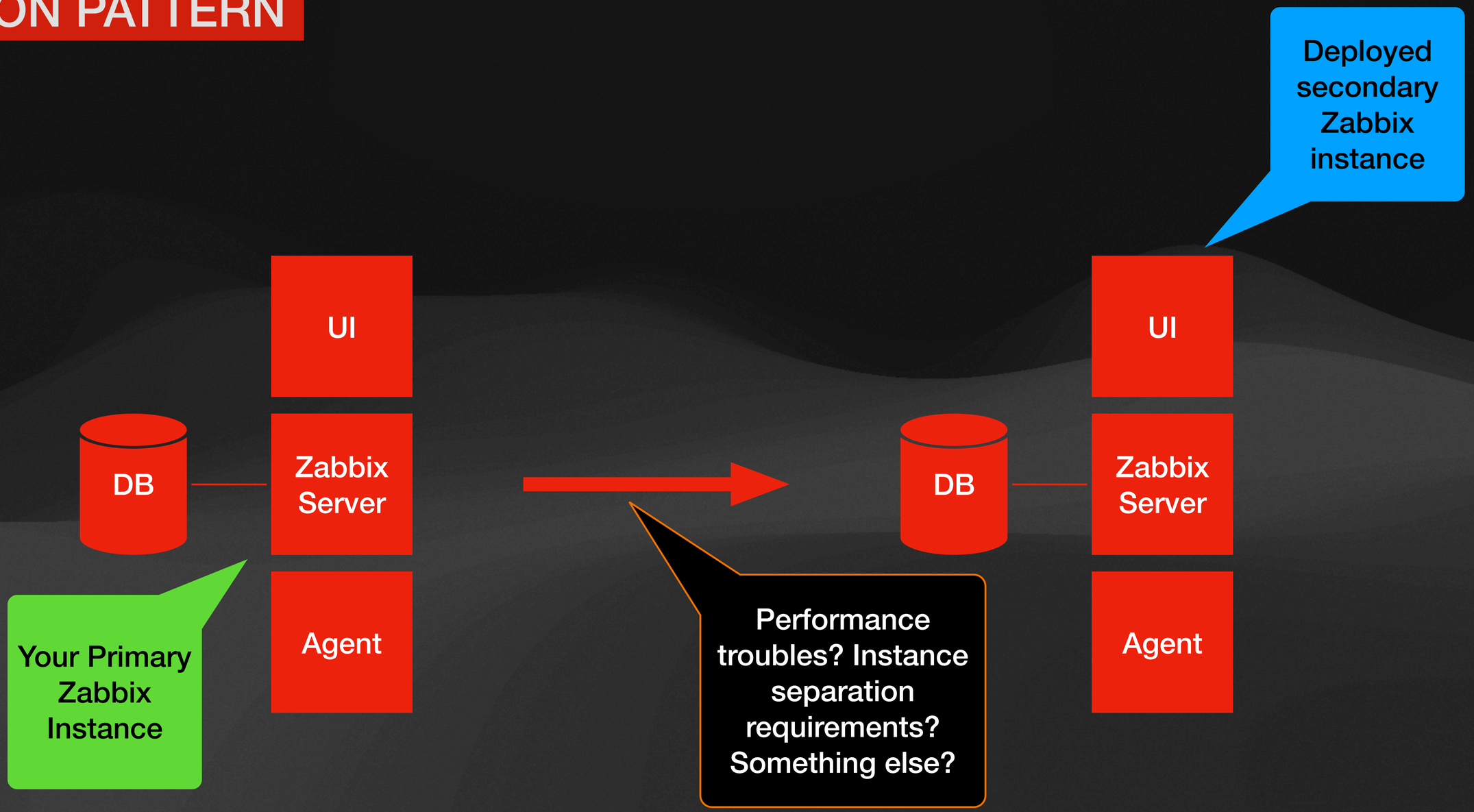
ZBUS extends Zabbix capabilities

Where Zabbix shines

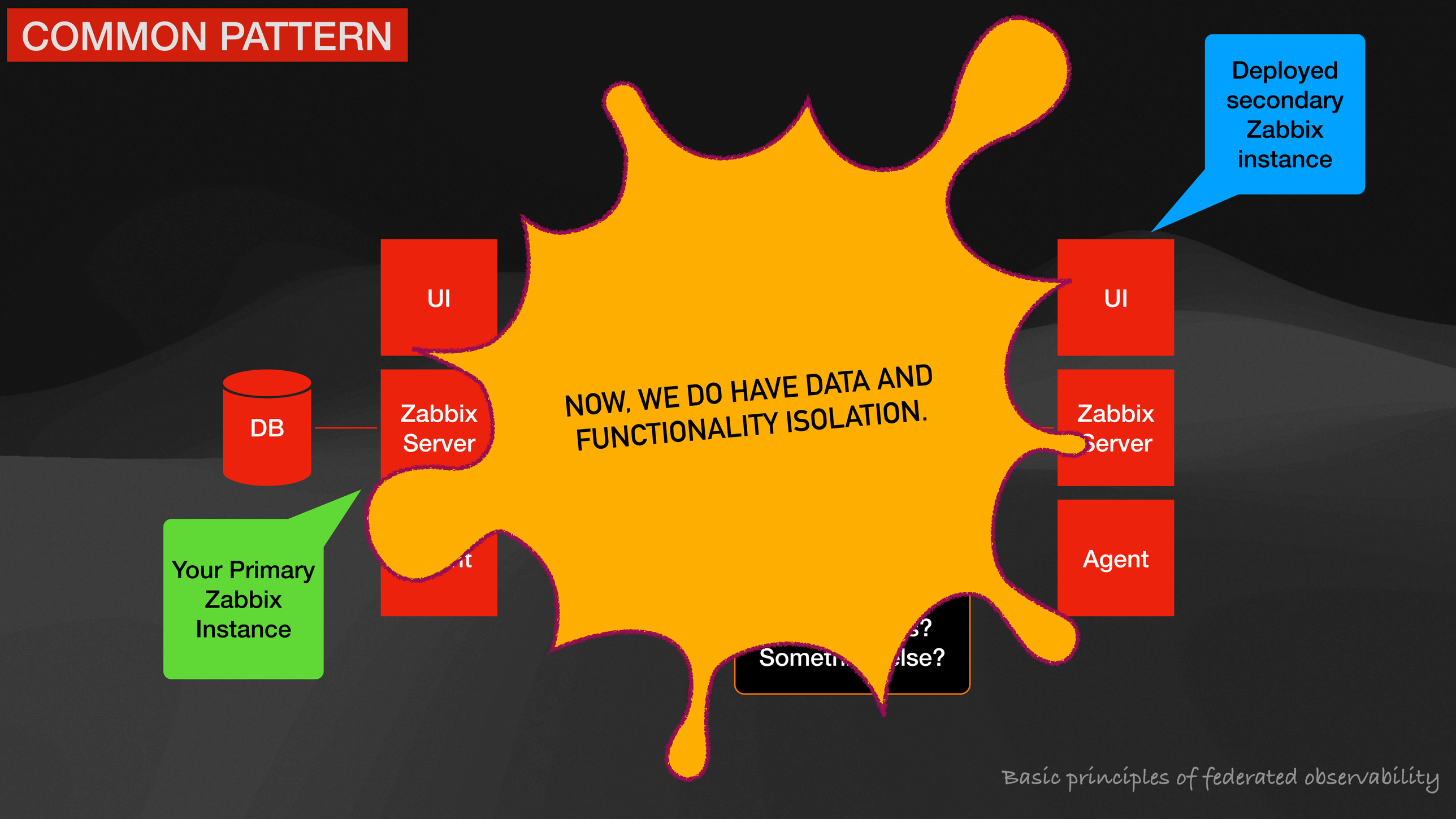
Zabbix provides all tools and APIs to implement true horizontal scalability.

This will make Zabbix a strong player in the market occupied by traditional SaaS observability solutions.

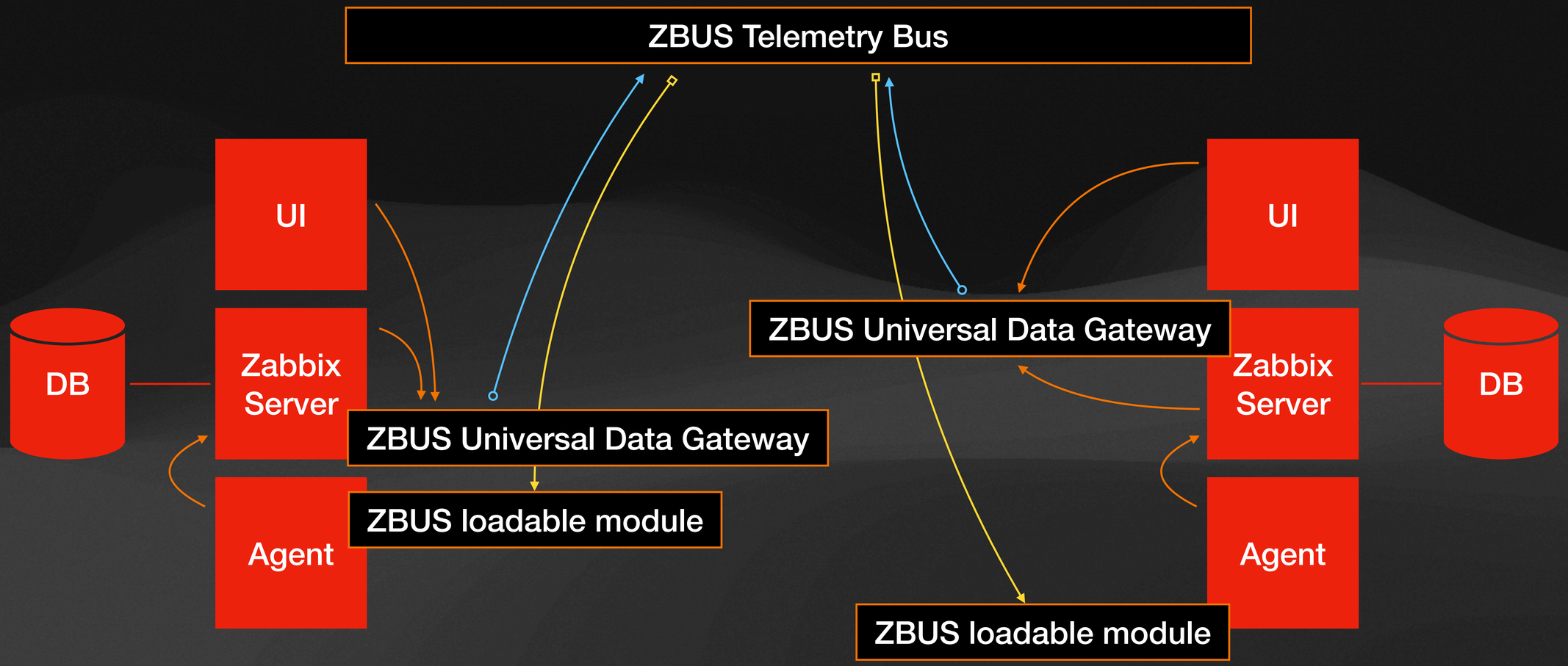
COMMON PATTERN



COMMON PATTERN



BASIC ARCHITECTURE



QUICK INTRO

```
% zenohd --adminspace-permissions=rw --cfg='plugins/storage_manager/storages/zbus:{key_expr:"zbus/**",volume:"memory"}'
[2024-03-18T21:25:44Z INFO  zenohd] zenohd v0.10.0-rc built with rustc 1.72.0 (5680fa18f 2023-08-23)
[2024-03-18T21:25:44Z INFO  zenohd] Initial conf: {"id":"7d409904acee372ca4c2e9e1aae8c73e","metadata":null,"mode":"router","connect":{"endpoints":[]},"listen":{"endpoints":["tcp/[::]:7447"]},"scouting":{"timeout":null,"delay":null,"multicast":{"enabled":
true,"address":null,"interface":null,"autoconnect":null,"listen":null},"gossip":{"enabled":null,"multihop":null,"autoconnect":null}},"timestamping":{"enabled":null,"drop_future_timestamp":null},"queries_default_timeout":null,"routing":{"router":{"peers_f
ailover_brokering":null},"peer":{"mode":null}},"aggregation":{"subscribers":[],"publishers":[]},"transport":{"unicast":{"accept_timeout":10000,"accept_pending":100,"max_sessions":1000,"max_links":1,"lowlatency":false},"multicast":{"join_interval":2500,"m
ax_sessions":1000},"qos":{"enabled":true},"link":{"protocols":null,"tx":{"sequence_number_resolution":"32bit","lease":10000,"keep_alive":4,"batch_size":65535,"queue":{"size":{"control":1,"real_time":1,"interactive_high":1,"interactive_low":1,"data_high":
2,"data":4,"data_low":2,"background":1},"backoff":100},"threads":3},"rx":{"buffer_size":65535,"max_message_size":1073741824},"tls":{"root_ca_certificate":null,"server_private_key":null,"server_certificate":null,"client_auth":null,"client_private_key":nul
l,"client_certificate":null,"server_name_verification":null},"unixpipe":{"file_access_mask":null},"compression":{"enabled":false}},"shared_memory":{"enabled":false},"auth":{"usrpwd":{"user":null,"password":null,"dictionary_file":null},"pubkey":{"public_k
ey_pem":null,"private_key_pem":null,"public_key_file":null,"private_key_file":null,"key_size":null,"known_keys_file":null}}},"adminspace":{"permissions":{"read":true,"write":true}},"plugins_search_dirs":[],"plugins":{"rest":{"__required__":true,"http_por
t":"8000"},"storage_manager":{"storages":{"zbus":{"key_expr":"zbus/**","volume":"memory"}}}}}}
[2024-03-18T21:25:44Z INFO  zenohd] Loading required plugin "rest"
[2024-03-18T21:25:44Z INFO  zenohd] Loading  plugin "storage_manager"
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime] Using PID: 7d409904acee372ca4c2e9e1aae8c73e
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::1]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::aede:48ff:fe00:1122]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::1481:b530:77ce:2750]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::fc82:78ff:fe11:1338]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::fc82:78ff:fe11:1338]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::ee46:8fe4:a2cd:2387]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::a610:31f4:6f1d:2f32]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::ce81:b1c:bd2c:69e]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::5ec4:f9b9:397b:bebb]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/[fe80::3de0:9636:53d2:5f5b]:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] Zenoh can be reached at: tcp/192.168.86.228:7447
[2024-03-18T21:25:44Z INFO  zenoh::net::runtime::orchestrator] zenohd listening scout messages on 224.0.0.224:7446
[2024-03-18T21:25:44Z INFO  zenohd] Starting required plugin "rest"
[2024-03-18T21:25:44Z INFO  zenohd] Successfully started plugin rest from "/usr/local/lib/libzenoh_plugin_rest.dylib"
[2024-03-18T21:25:44Z INFO  zenohd] Starting  plugin "storage_manager"
[2024-03-18T21:25:44Z INFO  zenohd] Successfully started plugin storage_manager from "/usr/local/lib/libzenoh_plugin_storage_manager.dylib"
[2024-03-18T21:25:44Z INFO  zenohd] Finished loading plugins
```

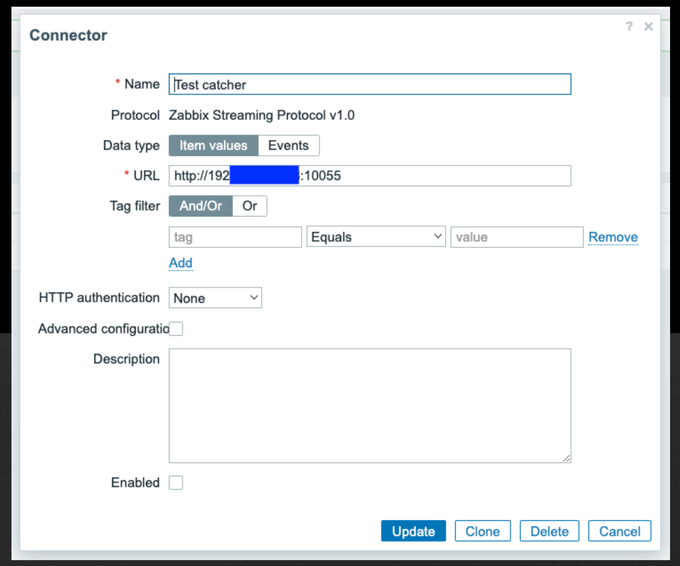
ZENOH daemon a.k.a. Telemetry bus

```
Starting Zabbix Agent [Zabbix server]. Zabbix 6.4.8 (revision ecda9311a92).
Press Ctrl+C to exit.

 26335:20240318:170031.421 Starting Zabbix Agent [Zabbix server]. Zabbix 6.4.8 (revision ecda9311a92).
 26335:20240318:170031.421 **** Enabled features ****
 26335:20240318:170031.421 IPv6 support:           NO
 26335:20240318:170031.421 TLS support:           YES
 26335:20240318:170031.421 **************************
 26335:20240318:170031.421 using configuration file: ./zabbix_agentd.conf
[2024-03-18T23:00:31Z DEBUG zbus_module::init] Initialize ZBUS module
 26335:20240318:170031.436 loaded modules: zbus_module.so
 26335:20240318:170031.436 agent #0 started [main process]
 26336:20240318:170031.436 agent #1 started [collector]
 26337:20240318:170031.437 agent #2 started [listener #1]
 26338:20240318:170031.437 agent #3 started [listener #2]
 26339:20240318:170031.437 agent #4 started [listener #3]
 26340:20240318:170031.437 agent #5 started [active checks #1]
```

Loadable Module Interfacing Zabbix Agent with Telemetry bus

QUICK INTRO

```
[redacted] % ./target/debug/zbusdg -dd  --zabbix-api http://192.[redacted]/zabbix gateway --zbus --zabbix --zabbix-token [redacted]
[2024-07-09T21:20:50Z DEBUG zbusdg::stdlib::channel] Initializing default pipes
[2024-07-09T21:20:50Z DEBUG zbusdg::stdlib::threads] Running STDLIB::threads init
[2024-07-09T21:20:50Z DEBUG zbusdg::stdlib::threads] Thread engine has been configured with 16 threads
[2024-07-09T21:20:50Z DEBUG zbusdg::stdlib::alerts] Running STDLIB::alerts init
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_loader_logs_categorization] zbus_loader_logs_categorization::run() reached
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_loader_logs_categorization] Logs category training data not provided
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd] Execute ZBUSDG
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_rhai] zbus_rhai::run() reached
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_rhai] Processing will not be scripted
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway] Filtering disabled
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway] Analythical collection and enchancing is OFF
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway] Logs analythical enchancing is OFF
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway_processor] PROCESSOR ZABBIX thread has been started
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway_zbus_sender] ZBUS sender thread has been started
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway_zbus_sender] Multicast discovery enabled
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway_zbus_sender] ZENOH bus set to: tcp/127.0.0.1:7447
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway_catcher_zabbix] Starting zabbix catching thread #0
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway_zbus_sender] ZENOH listen set to default
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway_zbus_sender] ZENOH configured in PEER mode
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway_zbus_sender] ZENOH config is OK
[2024-07-09T21:20:50Z DEBUG zbusdg::cmd::zbus_gateway_zbus_sender] Published telemetry will not be aggregated
```



Run ZBUS Universal Data Gateway to export Zabbix telemetry to the bus.

… and configure Zabbix Connector.

QUICK INTRO

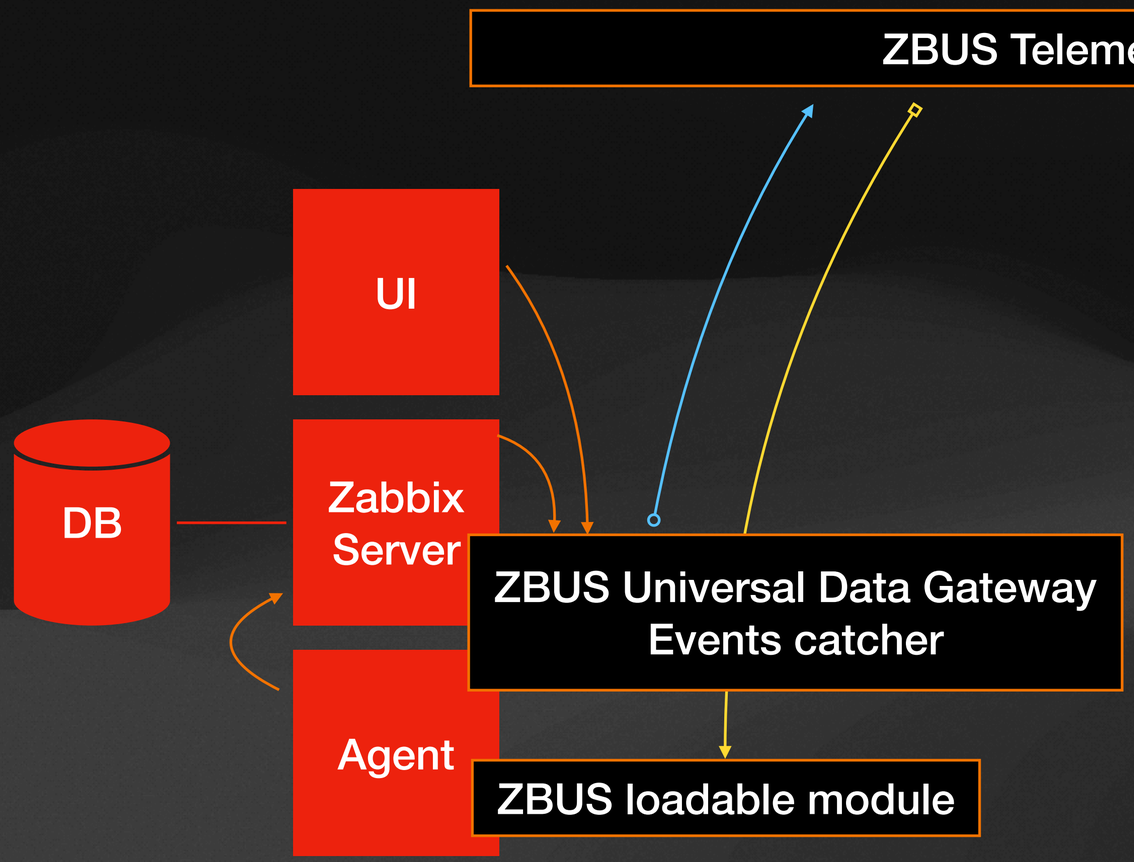
```
[redacted] % zabbix_get -s 127.0.0.1 -k "zbus.get[zbus/metric/v2/local/system.cpu.load/all/avg1]"
{"body":{"details":{"destination":"zbus/metric/v2/local/system.cpu.load/all/avg1","details":{"contentType":0,"data":0.0
5,"detailType":""},"origin":"Zabbix server","properties":{"name":"Linux: Load average (1m avg)","tags":null,"zabbix_clo
ck":1719114249,"zabbix_host_name":"Zabbix server","zabbix_item":"system.cpu.load[all,avg1]","zabbix_itemid":42249,"zabb
ix_ns":478352324,"zbus_item":"/system.cpu.load/all/avg1"}}},"headers":{"compressionAlgorithm":null,"cultureCode":null,"
encryptionAlgorithm":null,"messageType":"telemetry","route":"local","streamName":"local","version":"v2"},"id":"gwiF8G5_
p02eKmIkbe08Q"}
[redacted] % zabbix_get -s 127.0.0.1 -k "zbus.get[zbus/metric/v2/local/system.cpu.load/all/avg1]"
{"body":{"details":{"destination":"zbus/metric/v2/local/system.cpu.load/all/avg1","details":{"contentType":0,"data":0.0
5,"detailType":""},"origin":"Zabbix server","properties":{"name":"Linux: Load average (1m avg)","tags":null,"zabbix_clo
ck":1719114249,"zabbix_host_name":"Zabbix server","zabbix_item":"system.cpu.load[all,avg1]","zabbix_itemid":42249,"zabb
ix_ns":478352324,"zbus_item":"/system.cpu.load/all/avg1"}}},"headers":{"compressionAlgorithm":null,"cultureCode":null,"
encryptionAlgorithm":null,"messageType":"telemetry","route":"local","streamName":"local","version":"v2"},"id":"gwiF8G5_
p02eKmIkbe08Q"}
[redacted] % zabbix_get -s 127.0.0.1 -k "zbus.query[zbus/metric/v2/local/system.cpu.load/all/avg1, $.body]"
{"details":{"destination":"zbus/metric/v2/local/system.cpu.load/all/avg1","details":{"contentType":0,"data":0.05,"detai
lType":""},"origin":"Zabbix server","properties":{"name":"Linux: Load average (1m avg)","tags":null,"zabbix_clock":1719
114249,"zabbix_host_name":"Zabbix server","zabbix_item":"system.cpu.load[all,avg1]","zabbix_itemid":42249,"zabbix_ns":4
78352324,"zbus_item":"/system.cpu.load/all/avg1"}}}
[redacted] % zabbix_get -s 127.0.0.1 -k "zbus.query[zbus/metric/v2/local/system.cpu.load/all/avg1, $.body.details]"

{"destination":"zbus/metric/v2/local/system.cpu.load/all/avg1","details":{"contentType":0,"data":0.05,"detailType":""},
"origin":"Zabbix server","properti  :{"name":"Linux: Load average (1m avg)","tags":null,"zabbix_clock":1719114249,"zab
bix_host_name":"Zabbix server"  bix_item":"system.cpu.load[all,avg1]","zabbix_itemid":42249,"zabbix_ns":478352324,"z
                                vg1"}}
                              27.0.0.1 -k "zbus.query[zbus/metric/v2/local/system.cpu.load/all/avg1, $.body.details.d

                              etailType":""}
                              127.0.0.1 -k "zbus.query_float[zbus/metric/v2/local/system.cpu.load/all/avg1, $.body.det
```

Then check if you can see Zabbix telemetry published on the telemetry bus by sending a query through the Zabbix Agent.

IMPLEMENTATION

Now, instead of isolated instances, you have a federated observability architecture. Multiple Zabbix instances can individually contribute collected and computed telemetry, making this data available to all members.

BASIC ARCHITECTURE



ZBUS Universal Data Gateway can capture information about events generated by Zabbix servers that are members of the federation and publish events on the Telemetry bus, where they can be accessed as regular telemetry.

EXTRAS


```
{
  "body": {
    "details": {
      "destination": "/system.cpu.switches",
      "details": {
        "analythical_data": {
          "geometric_mean": 554.3513488865332,
          "harmonic_mean()": 553.2152732871378,
          "markov_chain_forecast": [
            587.5782702861025,
            566.6332505132751,
            551.265542705476,
            526.4156638506256,
            539.763143540589,
            571.3996695299844,
            533.6376695277129,
            553.9396148478114,
            542.7016869447555
          ],
          "maximum": 746.8642289349323,
          "mean": 555.597475933712,
          "minimum": 503.599956788435,
          "n_of_samples": 128,
          "quadratic_mean": 556.967770594719,
          "statistical_oscillator": -2.955105620820346,
          "std_dev": 39.198807211034705,
          "tsf_next": 545.6179761122715,
          "variance": 1536.5464867678663
        },
        "contentType": 0,
        "data": 536.9516607478183,
        "detailType": ""
      },
      "origin": "Zabbix server",
      "properties": {
        "name": "Linux: Context switches per second",
        "tags": null,
        "zabbix_clock": 1717741041,
        "zabbix_host_name": "Zabbix server",
        "zabbix_item": "system.cpu.switches",
        "zabbix_itemid": 42261,
        "zabbix_ns": 623252060
      }
    }
  },
  "headers": {
    "compressionAlgorithm": null,
    "cultureCode": null,
    "encryptionAlgorithm": null,
    "messageType": "telemetry",
    "route": "local",
    "streamName": "local",
    "version": "v2"
  },
  "id": "oaBWozCz5hJL_s4Wu7CvB"
}
```


Universal Data Gateway can enhance telemetry with statistical and forecast computation.

You can programmatically filter and transform telemetry in real-time.

```
fn filter(data) {
  true
}

fn transformation(data) {
  data.body.details.added_by_transformation = "Transformation routine been here";
  data
}
```

EXTRAS

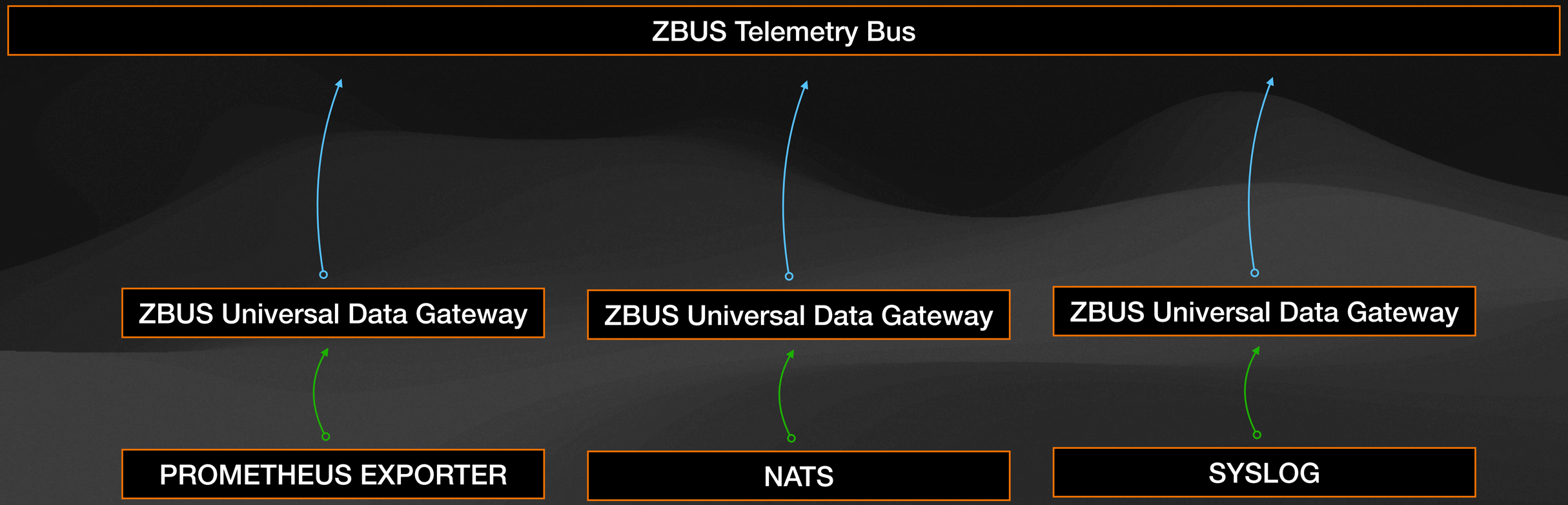

```
"body": {
  "details": {
    "destination": "/vfs.dev.queue_size/sda",
    "details": {
      "analytical_data": {
        "anomalies": [
          0.02509447745978832,
          0.024656997993588448,
          0.024203620851039886,
          0.03446614742279053,
          0.03890187665820122
        ],
        "breakouts": [],
        "geometric_mean": 0.014386080810498149,
        "harmonic_mean()": 0.013943866730837967,
        "markov_chain_forecast": [
          0.0114241307670302,
          0.011594340257044256,
          0.014165685076108332,
          0.0250944765446439,
          0.014356744228746485,
          0.011565748255395558,
          0.018633149727116105,
          0.02465699755148972,
          0.024092730835433614
        ],
        "maximum": 0.03890187781176513,
        "mean": 0.014926412867644349,
        "minimum": 0.00954994344793434,
        "n_of_samples": 128,
        "quadratic_mean": 0.015597732139859084,
        "statistical_oscillator": 29.61101465767979,
        "std_dev": 0.004544534826173887,
        "tsf_next": 0.017988799694544592,
        "variance": 0.00002065279678630732
      },
      "contentType": 0,
      "data": 0.02105294733584284,
      "detailType": ""
    },
    "origin": "Zabbix server",
    "properties": {
      "name": "sda: Disk average queue size (avgqu-sz)",
      "tags": null,
      "zabbix_clock": 1718000338,
      "zabbix_host_name": "Zabbix server",
      "zabbix_item": "vfs.dev.queue_size[sda]",
      "zabbix_itemid": 46084,
      "zabbix_ns": 568778998
    }
  }
},
"headers": {
  "compressionAlgorithm": null,
  "cultureCode": null,
  "encryptionAlgorithm": null,
  "messageType": "telemetry",
  "route": "local",
  "streamName": "local",
  "version": "v2"
},
```


While sampling data in real-time, Universal Data Gateway can detect anomalies and breakouts in telemetry.

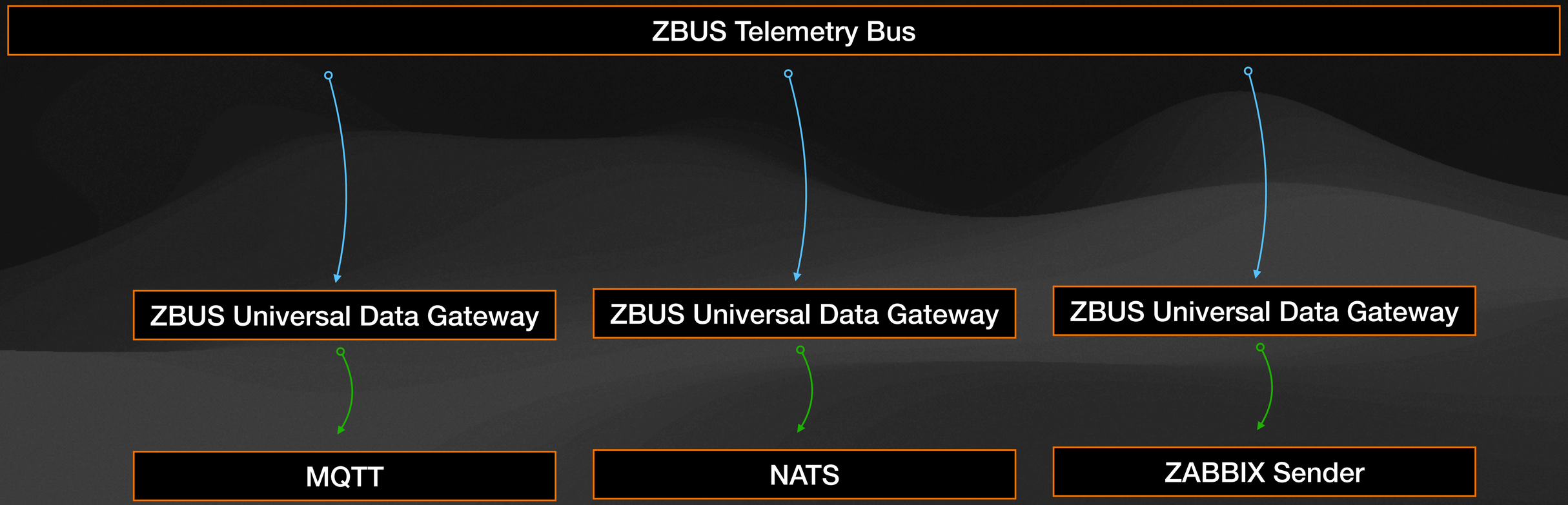
… also, a ZBUS UDG can categorize log entries while accepting data from Zabbix and sending it to a telemetry bus.


```
{
  "body": {
    "details": {
      "destination": "zbus/log/syslog",
      "details": {
        "analytical_data": {
          "category": "linux"
        },
        "contentType": 2,
        "data": "syslog[85644]: Session closed to user root",
        "detailType": ""
      },
      "origin": "home.lan",
      "properties": {
        "syslog_appname": "unknown",
        "syslog_facility": 0,
        "syslog_procid": "unknown",
        "syslog_severity": 7,
        "syslog_version": 0,
        "timestamp": 1718680619000000000,
        "zabbix_item": "log[/var/log/syslog]"
      }
    }
  },
  "headers": {
    "compressionAlgorithm": null,
    "cultureCode": null,
    "encryptionAlgorithm": null,
    "messageType": "telemetry",
    "route": "local",
    "streamName": "local",
    "version": "v2"
  },
  "id": "iWAtouXGdUc35_3NWa67P"
}
```

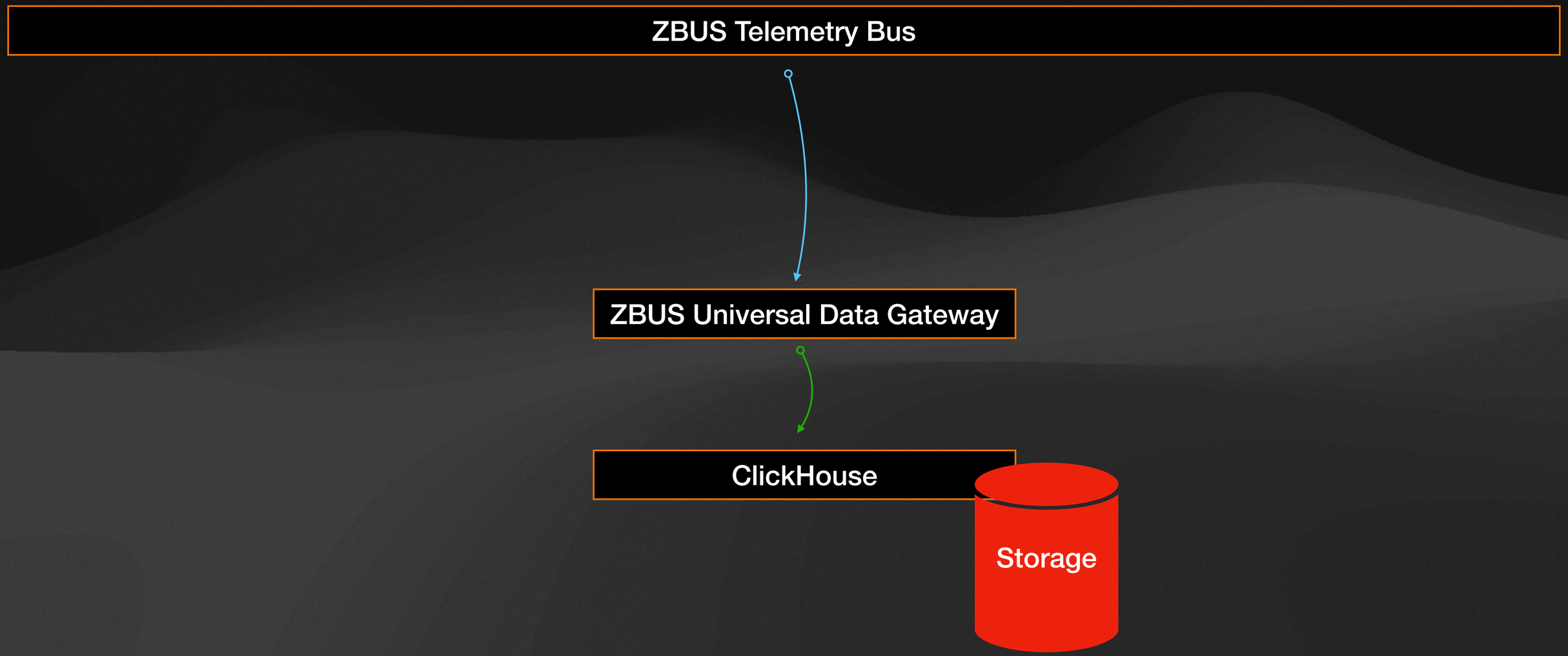
EXTRAS


ZBUS Telemetry Bus
ZBUS Universal Data Gateway
ZBUS Universal Data Gateway
ZBUS Universal Data Gateway
PROMETHEUS EXPORTER
NATS
SYSLOG

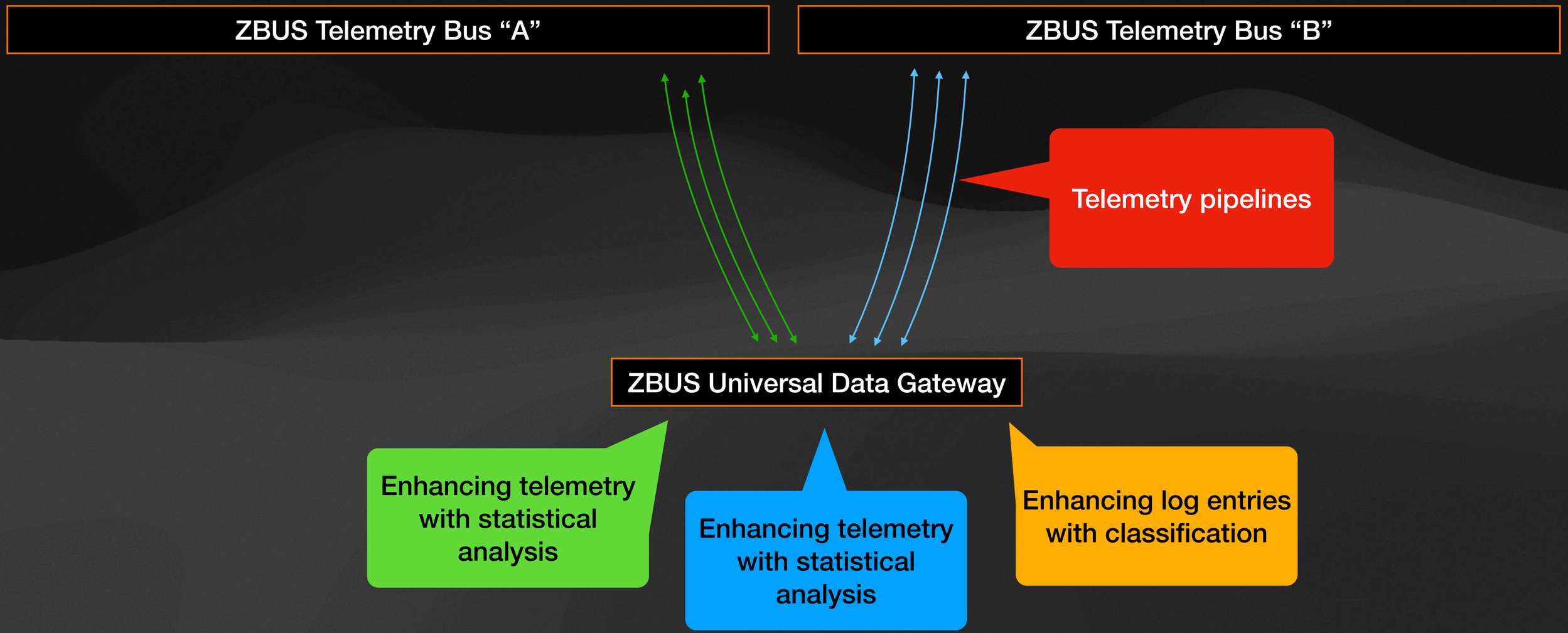
EXTRAS


ZBUS Telemetry Bus
ZBUS Universal Data Gateway
ZBUS Universal Data Gateway
ZBUS Universal Data Gateway
MQTT
NATS
ZABBIX Sender

EXTRAS


ZBUS Telemetry Bus
ZBUS Universal Data Gateway
ClickHouse
Storage

EXTRAS


ZBUS Telemetry Bus “A”
ZBUS Telemetry Bus “B”
Telemetry pipelines
ZBUS Universal Data Gateway
Enhancing telemetry with statistical analysis
Enhancing telemetry with statistical analysis
Enhancing log entries with classification

EXTRAS

```
[redacted] % curl -X POST -H 'Content-Type: application/json' -d '{"jsonrpc":"2.0","id
":"id","method":"metrics","params":[]}' http://127.0.0.1:10060
{"jsonrpc":"2.0","result":["zbus/metric/v2/local/agent.ping","zbus/metric/v2/local/net.if.in/enp0s3","zbus/metric/v2
/local/net.if.in/enp0s3/dropped","zbus/metric/v2/local/net.if.in/enp0s3/errors","zbus/metric/v2/local/net.if.out/enp
0s3","zbus/metric/v2/local/net.if.out/enp0s3/dropped","zbus/metric/v2/local/net.if.out/enp0s3/errors","zbus/metric/v
2/local/proc.num","zbus/metric/v2/local/proc.num/run","zbus/metric/v2/local/system.cpu.intr","zbus/metric/v2/local/s
ystem.cpu.load/all/avg1","zbus/metric/v2/local/system.cpu.load/all/avg15","zbus/metric/v2/local/system.cpu.load/all/
avg5","zbus/metric/v2/local/system.cpu.switches","zbus/metric/v2/local/system.cpu.util","zbus/metric/v2/local/system
.cpu.util/guest","zbus/metric/v2/local/system.cpu.util/guest_nice","zbus/metric/v2/local/system.cpu.util/idle","zbus
/metric/v2/local/system.cpu.util/interrupt","zbus/metric/v2/local/system.cpu.util/iowait","zbus/metric/v2/local/syst
em.cpu.util/nice","zbus/metric/v2/local/system.cpu.util/softirq","zbus/metric/v2/local/system.cpu.util/steal","zbus/
metric/v2/local/system.cpu.util/system","zbus/metric/v2/local/system.cpu.util/user","zbus/metric/v2/local/system.loc
altime","zbus/metric/v2/local/system.swap.size/free","zbus/metric/v2/local/system.swap.size/pfree","zbus/metric/v2/l
ocal/system.swap.size/total","zbus/metric/v2/local/system.uptime","zbus/metric/v2/local/system.users.num","zbus/metr
ic/v2/local/vfs.dev.queue_size/sda","zbus/metric/v2/local/vfs.dev.read.await/sda","zbus/metric/v2/local/vfs.dev.read
.rate/sda","zbus/metric/v2/local/vfs.dev.read.time.rate/sda","zbus/metric/v2/local/vfs.dev.util/sda","zbus/metric/v2
/local/vfs.dev.write.await/sda","zbus/metric/v2/local/vfs.dev.write.rate/sda","zbus/metric/v2/local/vfs.dev.write.ti
me.rate/sda","zbus/metric/v2/local/vfs.file.contents","zbus/metric/v2/local/vfs.fs.dependent.inode/\\/pfree","zbus/m
etric/v2/local/vfs.fs.dependent.inode/\\boot/pfree","zbus/metric/v2/local/vfs.fs.dependent.size/\\/pused","zbus/metr
ic/v2/local/vfs.fs.dependent.size/\\/total","zbus/metric/v2/local/vfs.fs.dependent.size/\\/used","zbus/metric/v2/loc
al/vfs.fs.dependent.size/\\boot/pused","zbus/metric/v2/local/vfs.fs.dependent.size/\\boot/total","zbus/metric/v2/loc
al/vfs.fs.dependent.size/\\boot/used","zbus/metric/v2/local/vfs.fs.dependent/\\/data","zbus/metric/v2/local/vfs.fs.d
ependent/\\/readonly","zbus/metric/v2/local/vfs.fs.dependent/\\boot/data","zbus/metric/v2/local/vfs.fs.dependent/\\b
oot/readonly","zbus/metric/v2/local/vm.memory.size/available","zbus/metric/v2/local/vm.memory.size/pavailable","zbus
/metric/v2/local/vm.memory.size/total","zbus/metric/v2/local/vm.memory.utilization","zbus/metric/v2/local/zabbix/con
nector_queue","zbus/metric/v2/local/zabbix/host/agent/available","zbus/metric/v2/local/zabbix/lld_queue","zbus/metri
c/v2/local/zabbix/preprocessing_queue","zbus/metric/v2/local/zabbix/process/alert/manager/avg/busy","zbus/metric/v2/
local/zabbix/process/alert/syncer/avg/busy","zbus/metric/v2/local/zabbix/process/alerter/avg/busy","zbus/metric/v2/l
ocal/zabbix/process/availability/manager/avg/busy","zbus/metric/v2/local/zabbix/process/configuration/syncer/avg/bus
y","zbus/metric/v2/local/zabbix/process/connector/manager/avg/busy","zbus/metric/v2/local/zabbix/process/connector/w
orker/avg/busy","zbus/metric/v2/local/zabbix/process/discoverer/avg/busy","zbus/metric/v2/local/zabbix/process/escal
ator/avg/busy","zbus/metric/v2/local/zabbix/process/history/poller/avg/busy","zbus/metric/v2/local/zabbix/process/hi
story/syncer/avg/busy","zbus/metric/v2/local/zabbix/process/housekeeper/avg/busy","zbus/metric/v2/local/zabbix/proce
ss/http/poller/avg/busy","zbus/metric/v2/local/zabbix/process/icmp/pinger/avg/busy","zbus/metric/v2/local/zabbix/pro
cess/lld/manager/avg/busy","zbus/metric/v2/local/zabbix/process/lld/worker/avg/busy","zbus/metric/v2/local/zabbix/pr
ocess/odbc/poller/avg/busy","zbus/metric/v2/local/zabbix/process/poller/avg/busy","zbus/metric/v2/local/zabbix/proce
ss/preprocessing/manager/avg/busy","zbus/metric/v2/local/zabbix/process/preprocessing/worker/avg/busy","zbus/metric/
v2/local/zabbix/process/proxy/poller/avg/busy","zbus/metric/v2/local/zabbix/process/self-monitoring/avg/busy","zbus/
metric/v2/local/zabbix/process/service/manager/avg/busy","zbus/metric/v2/local/zabbix/process/task/manager/avg/busy"
,"zbus/metric/v2/local/zabbix/process/timer/avg/busy","zbus/metric/v2/local/zabbix/process/trapper/avg/busy","zbus/m
etric/v2/local/zabbix/process/trigger/housekeeper/avg/busy","zbus/metric/v2/local/zabbix/process/unreachable/poll
avg/busy","zbus/metric/v2/local/zabbix/queue","zbus/metric/v2/local/zabbix/queue/10m","zbus/metric/v2/local/zabbix/r
cache/buffer/pused","zbus/metric/v2/local/zabbix/tcache/cache/pitems","zbus/metric/v2/local/zabbix/tcache/cache/pmis
ses","zbus/metric/v2/local/zabbix/vcache/buffer/pused","zbus/metric/v2/local/zabbix/vcache/cache/hits","zbus/metric/
v2/local/zabbix/vcache/cache/misses","zbus/metric/v2/local/zabbix/vcache/cache/mode","zbus/metric/v2/local/zabbix/wc
ache/history/pused","zbus/metric/v2/local/zabbix/wcache/index/pused","zbus/metric/v2/local/zabbix/wcache/trend/pused
","zbus/metric/v2/local/zabbix/wcache/values","zbus/metric/v2/local/zabbix/wcache/values/float","zbus/metric/v2/loca
l/zabbix/wcache/values/log","zbus/metric/v2/local/zabbix/wcache/values/not/supported","zbus/metric/v2/local/zabbix/w
cache/values/str","zbus/metric/v2/local/zabbix/wcache/values/text","zbus/metric/v2/local/zabbix/wcache/values/uint"]
,"id":"id"}
[redacted] % curl -X POST -H 'Content-Type: application/json' -d '{"jsonrpc":"2.0","id
":"id","method":"sample","params":["zbus/metric/v2/local/net.if.in/enp0s3"]}' http://127.0.0.1:10060
{"jsonrpc":"2.0","result":[9488,8112,7456,7600,7400],"id":"id"}
[redacted] % curl -X POST -H 'Content-Type: application/json' -d '{"jsonrpc":"2.0","id
":"id","method":"last","params":["zbus/metric/v2/local/system.uptime"]}' http://127.0.0.1:10060
{"jsonrpc":"2.0","result":1824349,"id":"id"}
```

ZBUS Telemetry Bus

ZBUS Universal Data Gateway
JSON-RPC API

Enabling access to telemetry data published on a bus for any JSON-RPC client.

WHERE IS THE SOURCE?

The conference presentation format limits me from covering all aspects of the federated Zabbix platform. However, this brief overview may pique your interest in this topic. The full source code and some documentation is available here.

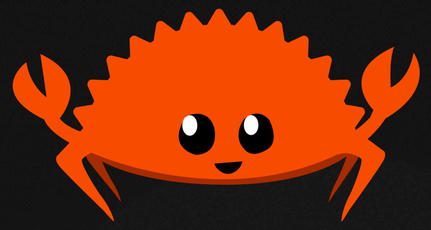
ZBUS Universal Data Gateway

https://github.com/vulogov/zbus_universal_data_gateway

https://github.com/vulogov/zabbix_zbus_module

Zabbix loadable module for accessing data on telemetry bus

REQUIREMENTS AND LICENSES



ZABBIX

The Rust and GCC compilers are necessary to compile both the ZBUS Universal Data Gateway and the Zabbix loadable module. Additionally, the compilation of the Zabbix loadable module requires access to the Zabbix source code. The software required for Zabbix Federated Observability is distributed under an Apache 2.0 license.

ABOUT ME

My name is Vladimir Ulogov. I am a seasoned software developer and software/systems architect with an extensive 30+ years of experience in the field of information technology. Over the course of my career, I have devoted 25 years to the specialized domain of monitoring and observability, with a significant portion of 15+ years purely dedicated to working with the Zabbix platform. Throughout my professional journey, I have accumulated 5 years of invaluable experience working for companies committed to the development of monitoring and observability solutions and associated technologies.

https://linkedin.com/in/vladimirulogov