Downtime-Free Upgrade of Central Zabbix Instance on Atea Managed Services

Vuk Kadenic

Solution Architect

Atea Sverige AB, Sweden

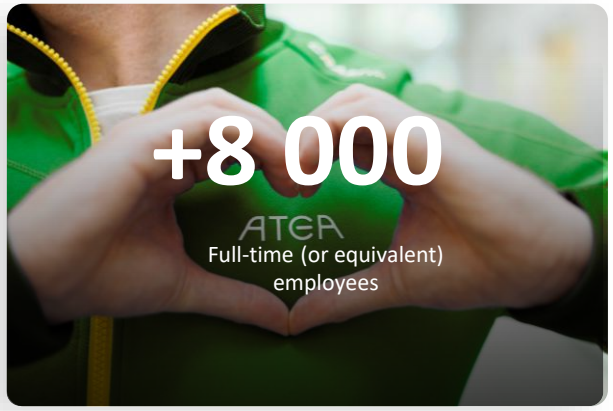
ZABBIX SUMMIT

OCTOBER 8 · 10, 2025

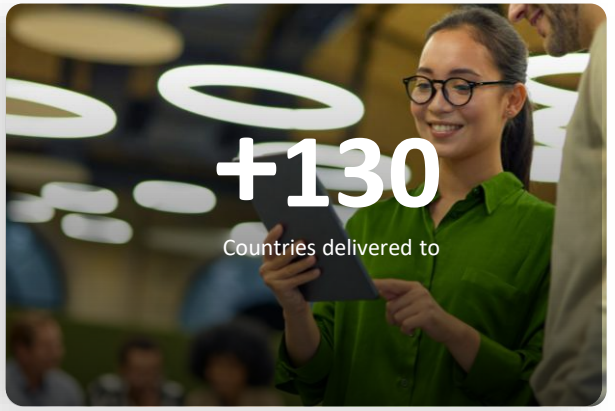
RIGA · LATVIA

ATEA

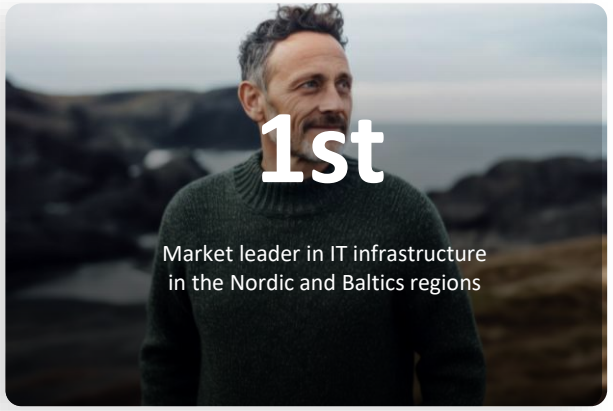
What is Atea?


+8 000
ATEA
Full-time (or equivalent) employees


88
Offices


+130
Countries delivered to

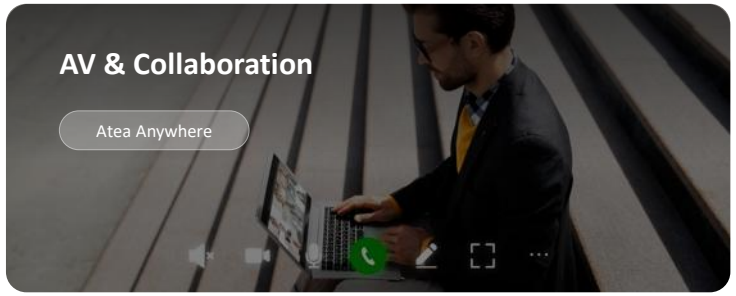

3rd
Largest IT infrastructure company in Europe

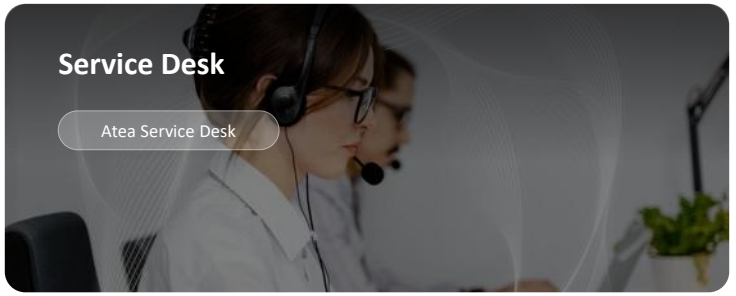

1st
Market leader in IT infrastructure in the Nordic and Baltics regions

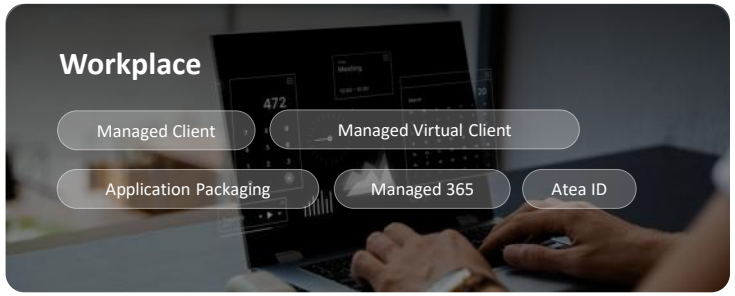

35
NOK billion revenue

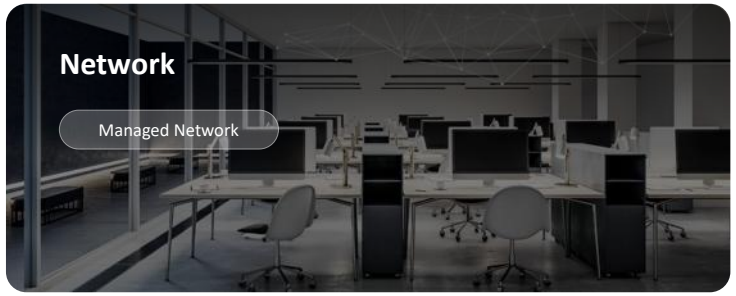
Atea Managed Services - AMS

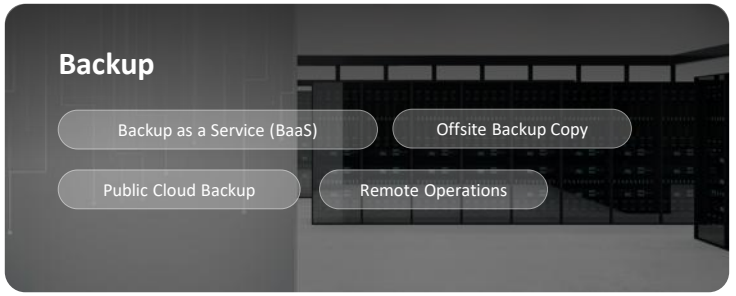
- 1500+ employees across the Nordic and Baltic

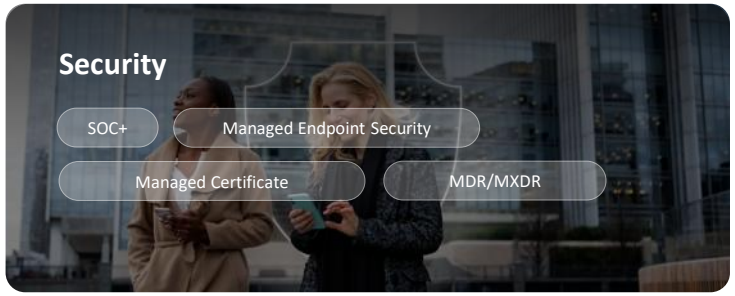


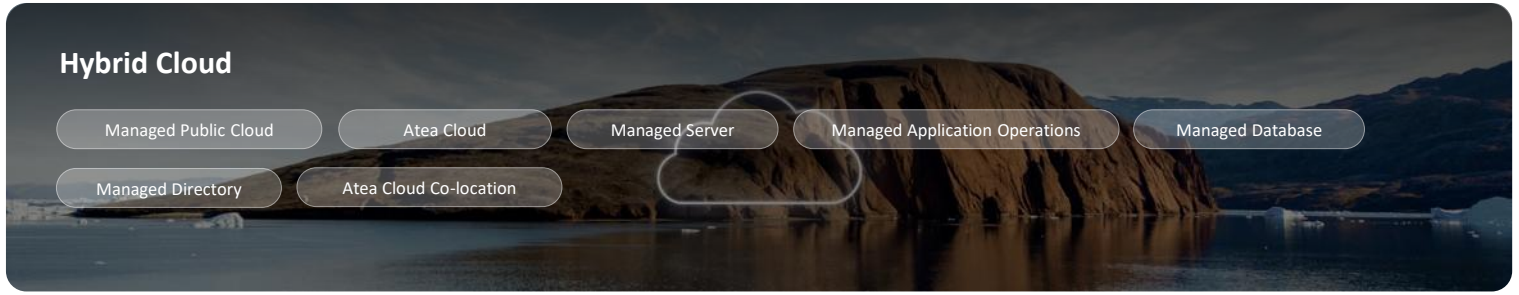




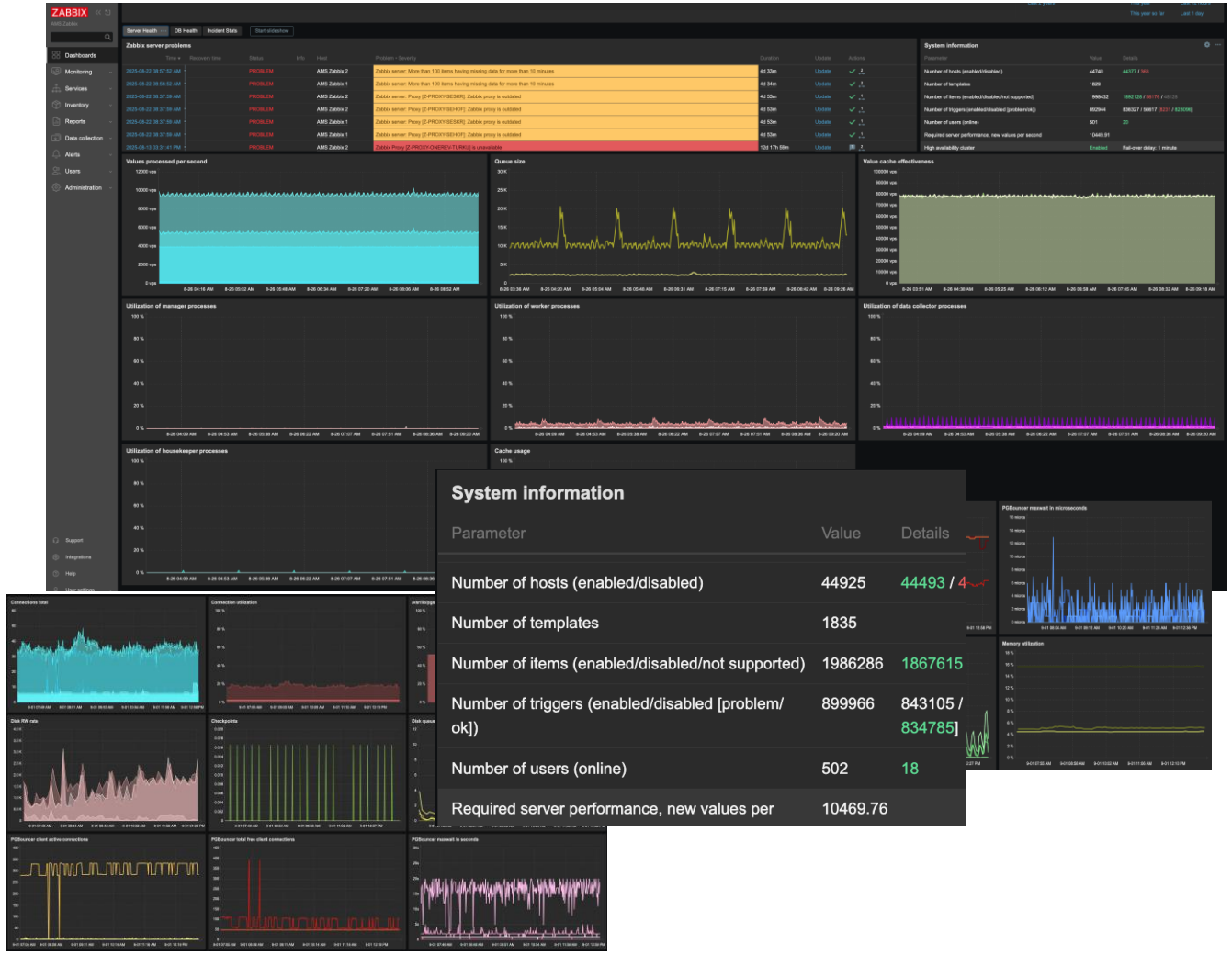








AMS Zabbix environment size



- 350+ monitored customers
- 45K+ monitored hosts
- 500+ active users
- 2M+ monitored items
- 10K+ new values per sec
- 160+ Zabbix proxies
- High Availability setup

How did Atea choose Zabbix?

ZABBIX 20 YEARS

-2021, AMS Sweden

- Multiple vendors and multiple instances of monitoring systems - a disconnect
- Evaluation of 8 vendors, 3 in final selection, landed on Zabbix

2021-2022 AMS Sweden

- Consolidated on Zabbix
- Single pane of glass, integrations to ITSM, invoicing and reporting
- Zabbix 5.0

2022, One AMS

- Always play with A team
- Monitoring Roadmap for all countries
- 3 system designs in final selection
- 120+ evaluation points, technical and economical
- Landed on Zabbix

2023, One AMS

- Implementation started
- Zabbix 5.0 -> 6.0

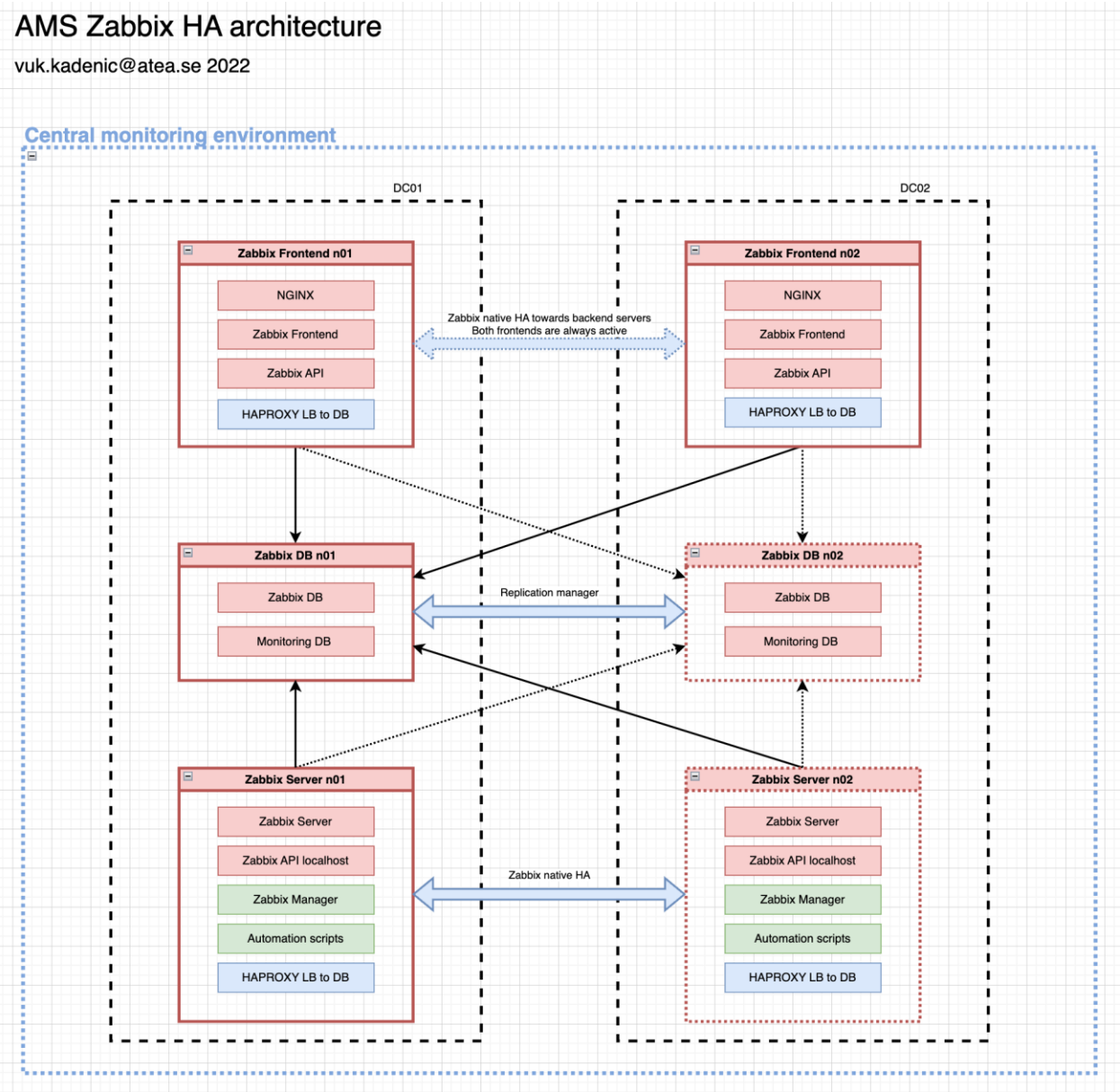
2025, One AMS

- 4/5 of migration to Zabbix complete
- Zabbix 6.0 -> 7.0

ATEA

Back to our theme!

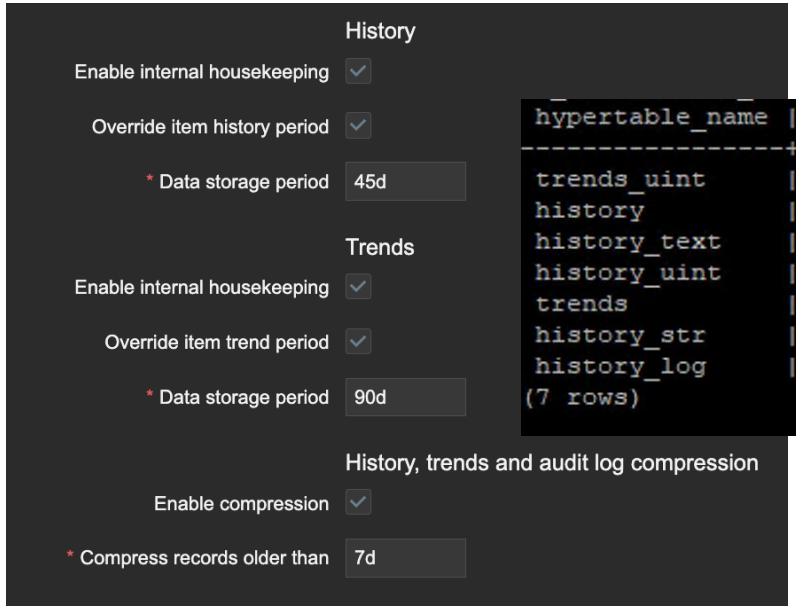
- Central Zabbix cluster
  - Native HA
- PostgreSQL cluster
  - TimescaleDB
  - Table partitioning
  - ~1.1TB compressed
  - 3rd Reporting node
  - Configuration backup



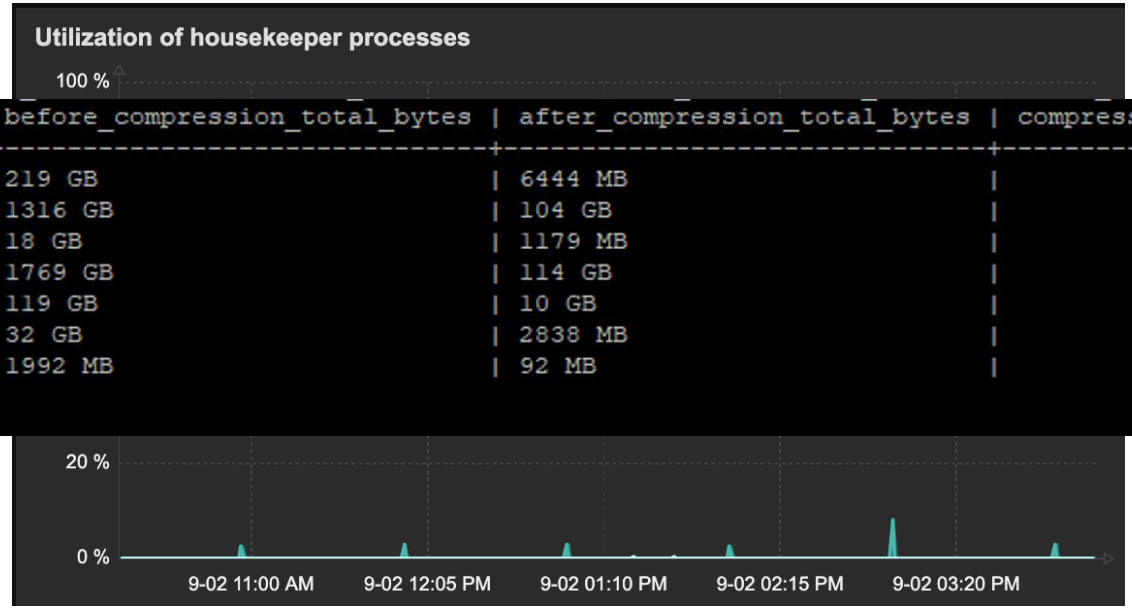
TimescaleDB table partitioning

- Removes the first major bottleneck when scaling up
- Significant performance boost, data compression
- "**Resistance**" to table structure changes impacts upgrade procedure

	Last check	Last value	Change
Database size	29m 3s	1.03 TB	+2.79 GB

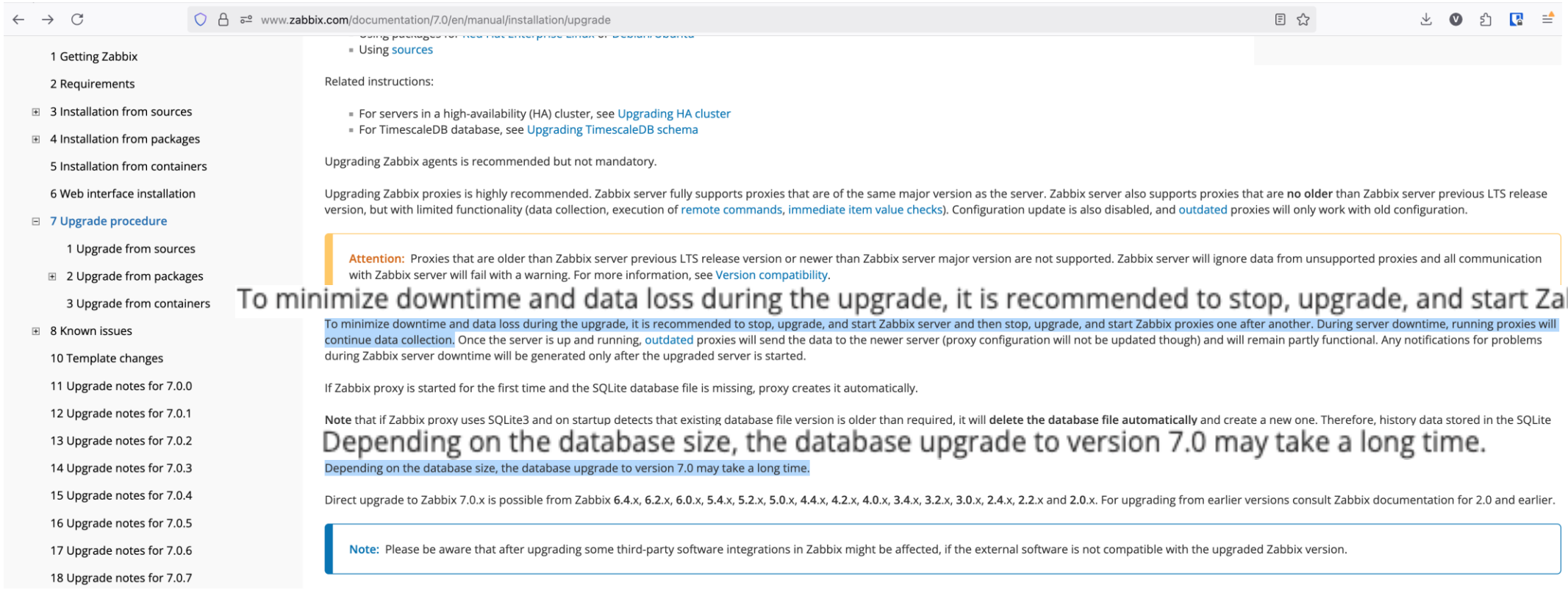



```
 hypertable_name | before_compression_total_bytes | after_compression_total_bytes | compression_rate
-----------------+--------------------------------+-------------------------------+------------------
 trends_uint     | 219 GB                         | 6444 MB                       |               34
 history         | 1316 GB                        | 104 GB                        |               12
 history_text    | 18 GB                          | 1179 MB                       |               15
 history_uint    | 1769 GB                        | 114 GB                        |               15
 trends          | 119 GB                         | 10 GB                         |               11
 history_str     | 32 GB                          | 2838 MB                       |               11
 history_log     | 1992 MB                        | 92 MB                         |               21
(7 rows)
```



Major upgrade – Zabbix documentation

- Zabbix Server is **down** until DB schema upgrade finishes
  - Zabbix Proxies buffer the collected data and deliver it once the server is up
  - Best case scenario = No data loss, but alerts are delayed



www.zabbix.com/documentation/7.0/en/manual/installation/upgrade

- Using sources

Related instructions:

- For servers in a high-availability (HA) cluster, see Upgrading HA cluster
- For TimescaleDB database, see Upgrading TimescaleDB schema

Upgrading Zabbix agents is recommended but not mandatory.

Upgrading Zabbix proxies is highly recommended. Zabbix server fully supports proxies that are of the same major version as the server. Zabbix server also supports proxies that are **no older** than Zabbix server previous LTS release version, but with limited functionality (data collection, execution of remote commands, immediate item value checks). Configuration update is also disabled, and outdated proxies will only work with old configuration.

Attention: Proxies that are older than Zabbix server previous LTS release version or newer than Zabbix server major version are not supported. Zabbix server will ignore data from unsupported proxies and all communication with Zabbix server will fail with a warning. For more information, see Version compatibility.

To minimize downtime and data loss during the upgrade, it is recommended to stop, upgrade, and start Zabbix server

To minimize downtime and data loss during the upgrade, it is recommended to stop, upgrade, and start Zabbix server and then stop, upgrade, and start Zabbix proxies one after another. During server downtime, running proxies will continue data collection. Once the server is up and running, outdated proxies will send the data to the newer server (proxy configuration will not be updated though) and will remain partly functional. Any notifications for problems during Zabbix server downtime will be generated only after the upgraded server is started.

If Zabbix proxy is started for the first time and the SQLite database file is missing, proxy creates it automatically.

Note that if Zabbix proxy uses SQLite3 and on startup detects that existing database file version is older than required, it will **delete the database file automatically** and create a new one. Therefore, history data stored in the SQLite

Depending on the database size, the database upgrade to version 7.0 may take a long time.

Depending on the database size, the database upgrade to version 7.0 may take a long time.

Direct upgrade to Zabbix 7.0.x is possible from Zabbix **6.4**.x, **6.2**.x, **6.0**.x, **5.4**.x, **5.2**.x, **5.0**.x, **4.4**.x, **4.2**.x, **4.0**.x, **3.4**.x, **3.2**.x, **3.0**.x, **2.4**.x, **2.2**.x and **2.0**.x. For upgrading from earlier versions consult Zabbix documentation for 2.0 and earlier.

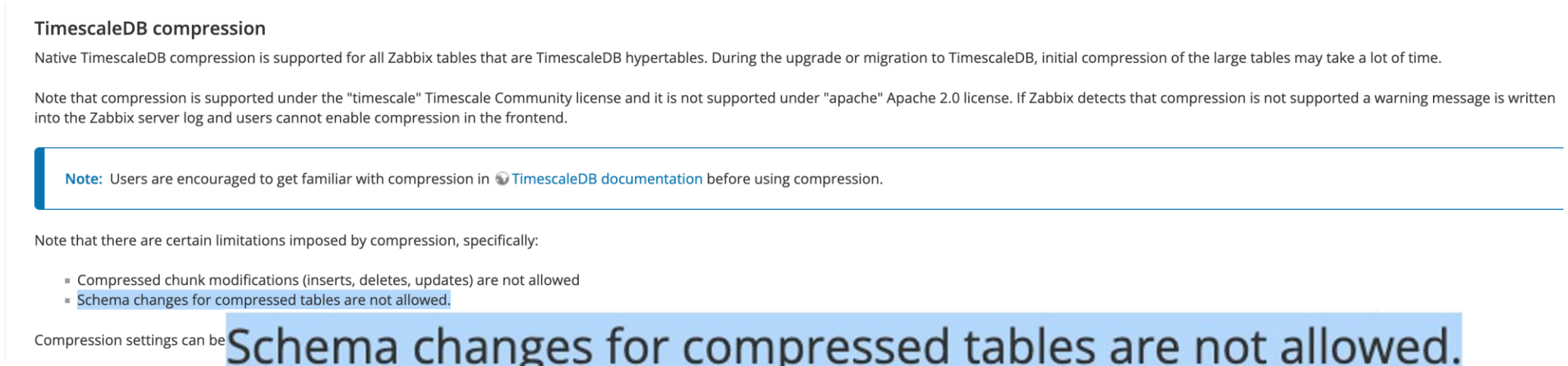
Note: Please be aware that after upgrading some third-party software integrations in Zabbix might be affected, if the external software is not compatible with the upgraded Zabbix version.

Pitfalls during major version upgrade

- Proxies run out of storage space
  - Data loss
  - May also require restoration of the proxy to make collection run again
- Upgrade takes longer time than Zabbix Proxies are configured to buffer
  - Data is rotated, FIFO principle, so data loss
  - Parts of collected data would be missing, showing as gaps on graphs
  - Proving the delivered SLA and KPIs becomes problematic
- Database upgrade fails
  - Constraints preventing changes of table structure (custom views, keys, triggers), lack of prerequisites (previous upgrades)
  - Can be fixed, but adds to downtime
  - Rollback using standby database instance and starting over after the issues are corrected may be necessary
  - You did make a db backup and/or have a standby instance, right?

What if I have TimescaleDB setup?

- Decompression of compressed tables is required
  - Disk space consideration, compression factor is not trivial
  - Disk R/W load consideration
  - Time consideration
- Check version compatibility!



TimescaleDB compression

Native TimescaleDB compression is supported for all Zabbix tables that are TimescaleDB hypertables. During the upgrade or migration to TimescaleDB, initial compression of the large tables may take a lot of time.

Note that compression is supported under the "timescale" Timescale Community license and it is not supported under "apache" Apache 2.0 license. If Zabbix detects that compression is not supported a warning message is written into the Zabbix server log and users cannot enable compression in the frontend.

Note: Users are encouraged to get familiar with compression in TimescaleDB documentation before using compression.

Note that there are certain limitations imposed by compression, specifically:

- Compressed chunk modifications (inserts, deletes, updates) are not allowed
- Schema changes for compressed tables are not allowed.

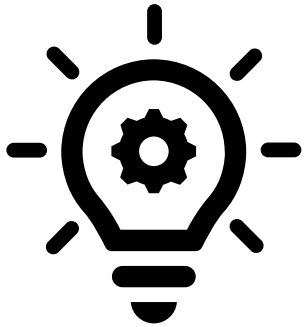
Compression settings can be

Schema changes for compressed tables are not allowed.

What does it all mean when upgrading?

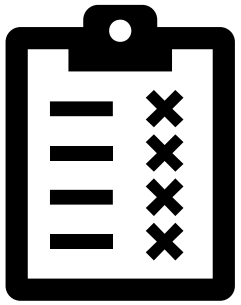
Requirements

- Uninterupted monitoring
- Uninterupted reports



Available solutions

- Upgrade all nodes at once
- Upgrade one node at a time



Downsides

- Alert delay
- Gaps in reports

A closer look at challenges

- How much upgrade time are we talking about here?
  - 200GB database with table partitioning and compression took 2.5 days for db schema upgrade
  - Approx. 500GB database without table partitioning and compression took approx. 20 minutes for db schema upgrade
  - Majority of the time is spent on decompress/compress
  - Even without compression, db schema upgrade can take significant time on large databases
  - Rough time estimate on AMS db size would be 2+ weeks with compression, hours if the db is first decompressed

Option 1 - All nodes down

- Disable the compression and wait for compressed chunks to be rotated out before attempting an upgrade
  - Proxies would buffer data – no gaps in reports
  - Disks on all db nodes would need to be expanded (hardware)
- Not a viable option
  - Hitting pause on monitoring for several hours (best case scenario, with decompressed database) is not acceptable in our case

Option 2 – Upgrade one node at a time

- Steps
  - Create new DB replica and promote it to master
  - Stop and upgrade one of the standby nodes of Zabbix Server
  - Point the upgraded Zabbix Server node to the new database node and start it – db schema starts upgrading and eventually completes
  - Point the rest of Zabbix Server nodes and all Frontend nodes to the new database, upgrade and restart them, then do post-upgrade tasks

    No alerts missed*, all notifications and automations are delivered but there is a gap in history for the time during the upgrade

    The gap can be minimized to several hours if the tables are decompressed first, but when the upgrade will finish cannot be predicted with enough accuracy

    Missing history data can be found in the “old” database instance

- Not acceptable due to the gap in history/reporting

Gap in history data – Impact on Business

- Missing proof of Service delivery through reports
  - But we did monitor, honest. Why can't they just trust us?
- Can be costly, or at the very least look unprofessional
  - Old db can be kept to fill in the gaps, but there is often a requirement on how the reports are delivered

So, how to solve this?

- Configuration backup – tables without collected data
  - Can be used to quickly restore monitoring functionality
  - DB with monitoring configuration is restored, Zabbix Server creates history and trend tables at startup
  - Upgrade of db schema with empty history and trend tables happens almost instantly
- Timing is everything

  - Identify breaking point between reporting periods, usually 1st day of the month 00:00
  - Plan configuration change freeze for a comfortable amount of time before the upgrade
  - Restore Zabbix db from configuration backup on the new db node
  - Point the standby Zabbix Server node to the new db node, restart it, stop it, upgrade it
  - At 00:00 stop the active Zabbix Server node and start the upgraded standby node, which upgrades the restored database and becomes active

What just happened?

- A fresh db node was created from configuration backup
- DB schema could be upgraded almost instantly because previously huge history and trend tables were now empty
- Monitoring on old Zabbix version stopped on the breaking point between reports
- Monitoring on new Zabbix version started on the breaking point between reports
- Old db can be used to crunch out previous month's reports
- New db will now be used to collect this month's reports
- What remains? Post upgrade tasks
  - Rebuild HA
  - Upgrade Zabbix Proxies and Agents

Considerations

- Alerts that were active on the old Zabbix Server node will appear as new at the next polling and trigger evaluation
  - New alerts will get escalated
- Active alerts that came in as traps will be missing from the new instance
  - Escalations have already been performed, so these are not missed*
  - Resolution traps will be ignored
- PostgreSQL and TimescaleDB were upgraded as well
  - These steps were left out to keep the presentation concise

QUESTIONS!

ATEA