sysfive
com securing performance

Zabbix Zero Downtime Upgrades

Case study in a highly-available

Zabbix Cluster

Background: Company

- sysfive.com GmbH:
  - Application Service Provider (Lotteries, Web Shops...)
  - Consulting Services (DevOps, Continuous Delivery...)
- Managing about a dozen platforms for various customers
- One Zabbix to rule them all!
  - Monitoring about 500 nodes
  - More than 50000 Items
  - More than 20000 Triggers
  - Strong integration in automated workflows (VM creation, deployment chains...)

Background: Zabbix

- Zabbix used for almost ten years now
- Historically grown setup
  - Running on “historic” hardware until early 2018
- 2018: Year of Upgrades (2.2 → 3.4, 3.4 → 4.0 ?)
  - Creating a completely new Zabbix infrastructure
  - Focus on High Availability
  - Plan for the next Upgrade
  - Make upgrades easy while having reliant monitoring!

Basic Thoughts I

- Sysfive.com needs:
  - **Distributed Monitoring Infrastructure (be save against network outages etc)**
  - **Centralised monitoring (one dashboard for Ops Team)**
- High Availability needs:
  - **Redundancy**
  - **Failover automatics**

Basic Thoughts II

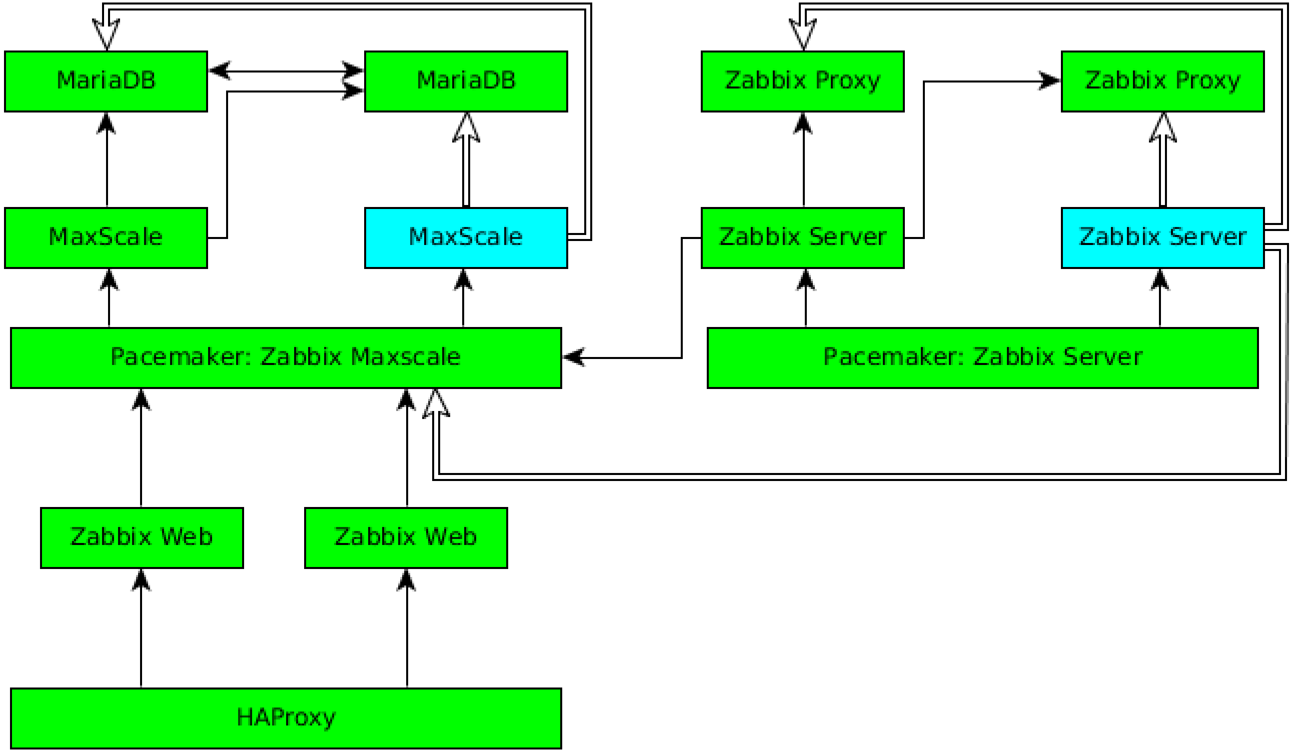
- Zero Downtime Upgrades need:
  - **Creating a new instance while running the old**
  - **Being able to downgrade as well (just in case!)**
  - **Allow switching between instances of Zabbix Server/Proxy**
  - **Ignore agent upgrades and do that later, version mix never created problems here**
- Conclusion:
  - **Use the redundant infrastructure to for upgrades**
  - **Accepting risk time for this allows reusing the same servers**

System Architecture

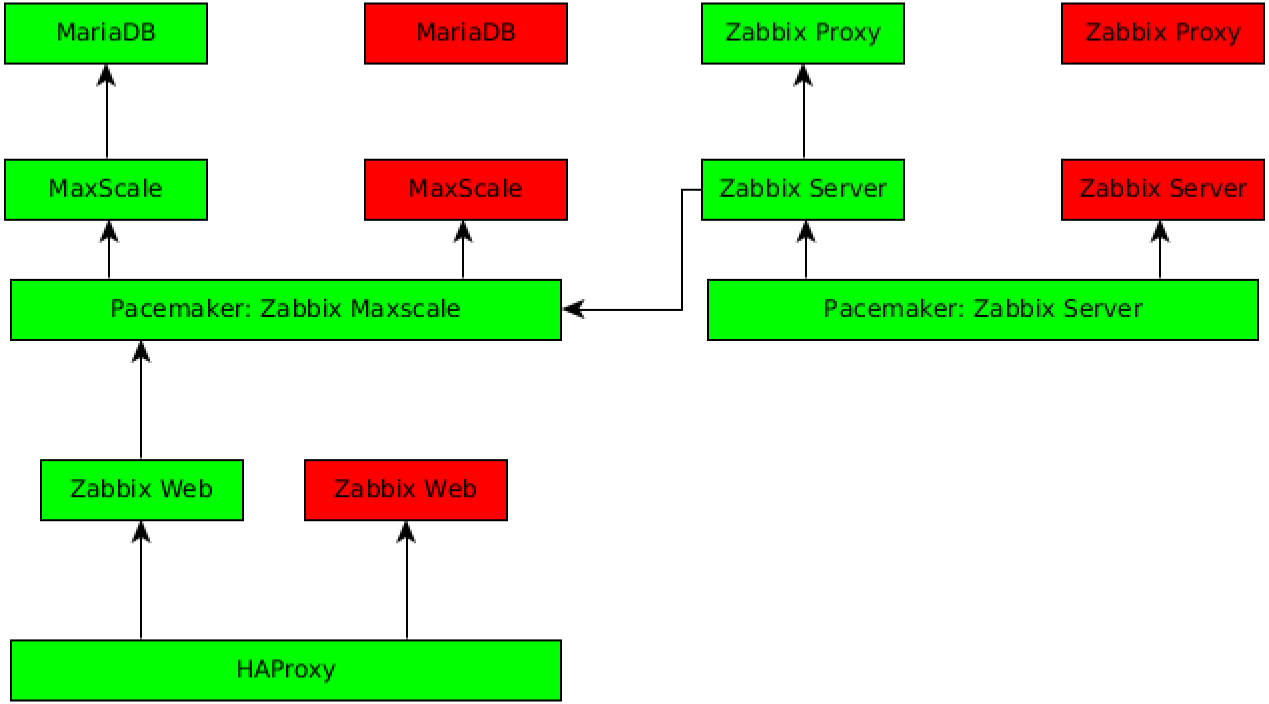
- Distribution
  - **Redundant Proxies in each data center**
- Failover Tools
  - → **Pacemaker**
  - → **Ansible (not automatically triggered!)**
- Load Balancing
  - → **MaxScale (for MariaDB)**
  - → **HAProxy (for Zabbix Web/NGINX)**
- DB Replication
  - → **MariaDB/Galera (Master-Master Replication)**

System Architecture

sysfive com securing performance



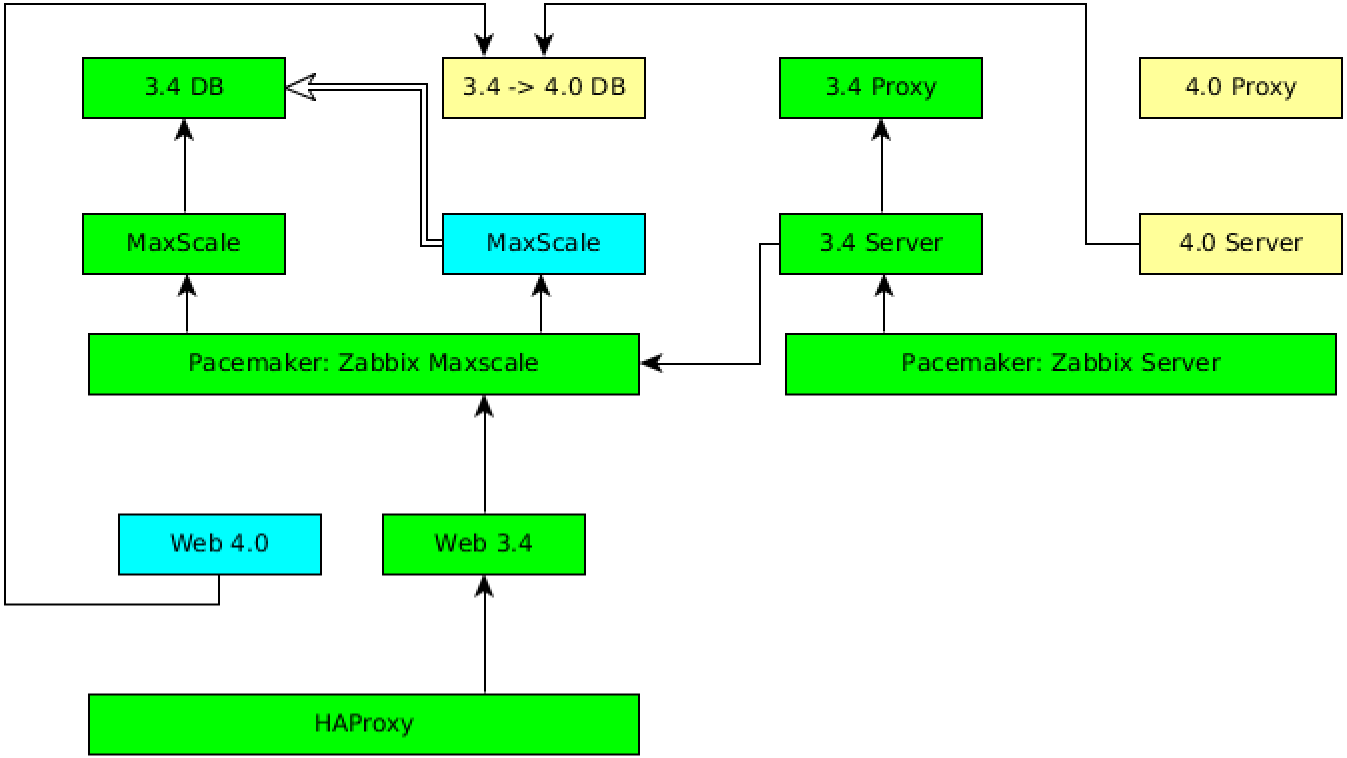
Redundancy


MariaDB
MariaDB
Zabbix Proxy
Zabbix Proxy
MaxScale
MaxScale
Zabbix Server
Zabbix Server
Pacemaker: Zabbix Maxscale
Pacemaker: Zabbix Server
Zabbix Web
Zabbix Web
HAProxy

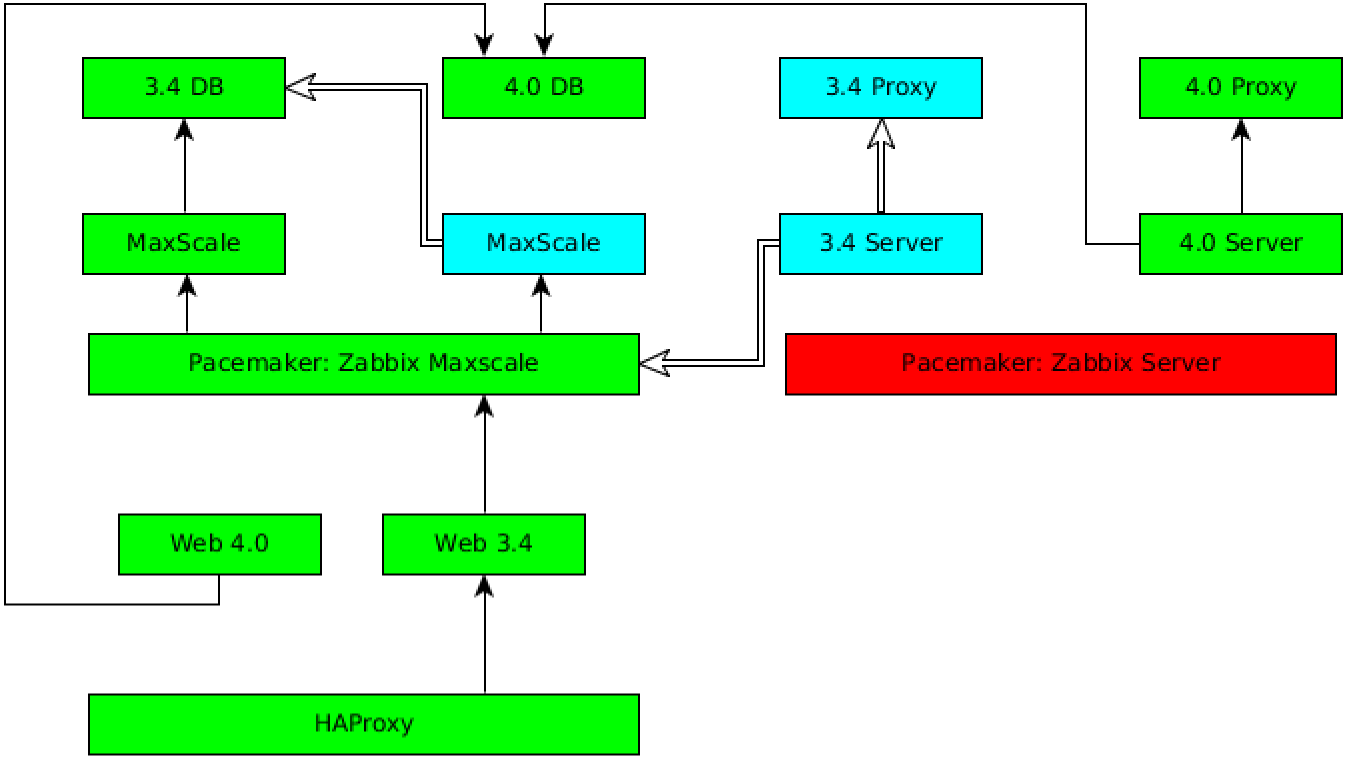
Upgrade workflow

- Split the cluster
- Upgrade one half at a time
- Smart and dynamic reconfiguration of Zabbix
- Be free which half to use (upgrade/rollback becomes a start/stop)
- Automate this!
  - **Use ansible to create upgrade/rollback plays**
  - **Create an ansible module for dynamic proxy/host handling**
    - Draft available on GitHub
    - Demonstration follows later in this talk

Upgrade Workflow



Evaluation Period



Demonstration

- Vagrant Setup:
  - **2 Servers**
  - **2 Proxies**
  - **4 Agents**
- Ansible Showcase:
  - **Move monitored hosts between proxies**

    → Demonstrating automated Ansible inventory creation

    → Use Ansible + Zabbix API for Zabbix Server reconfiguration

    → Use Ansible for Zabbix Agent reconfiguration

  - **https://github.com/stefanheykes/zabbix-api-demo**

Thank you for your attention!

Questions?