



**ITMO UNIVERSITY**

Saint Petersburg, Russia

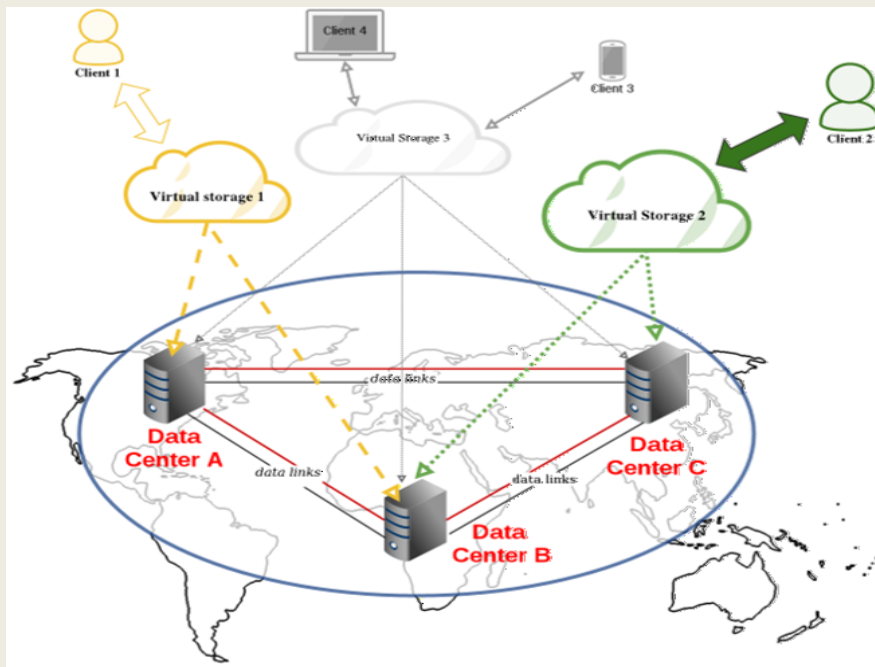
# APPROACHES TO THE AUTOMATED DEPLOYMENT OF THE CLOUD INFRASTRUCTURE OF GEOGRAPHICALLY DISTRIBUTED DATA CENTERS

Contract # 03.G25.31.0229 “Development of new technological components for management systems of geographically distributed Data Centers”

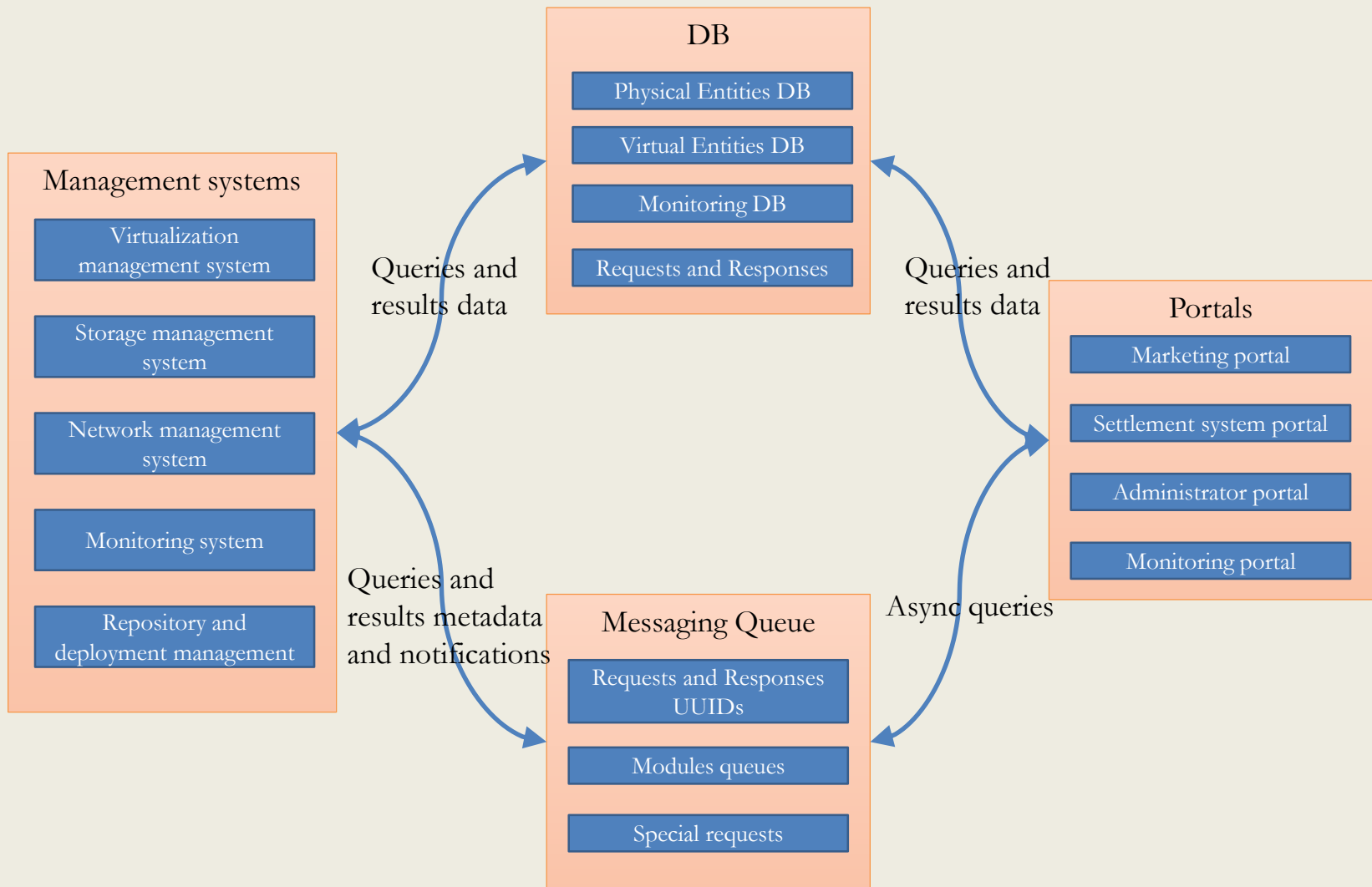
**Reporter: Petr Fedchenkov**  
**[pvfedchenkov@corp.ifmo.ru](mailto:pvfedchenkov@corp.ifmo.ru)**

# Contemporary computer clouds and near future

- Number of Data Centers is steadily growing
- Demands to integrate resources of many DCs are growing as well.
  - Pros:
    - Ability to redistribute resources in according to changing demands;
    - More reliable service even in case when whole data center is down;
  - Cons:
    - It is needed special management system to control multiple DCs;
    - Special attention to data links between DCs s required
- The variety of services and their different versions leads to the need for centralized deployment and provisioning



# Logical structure of cloud

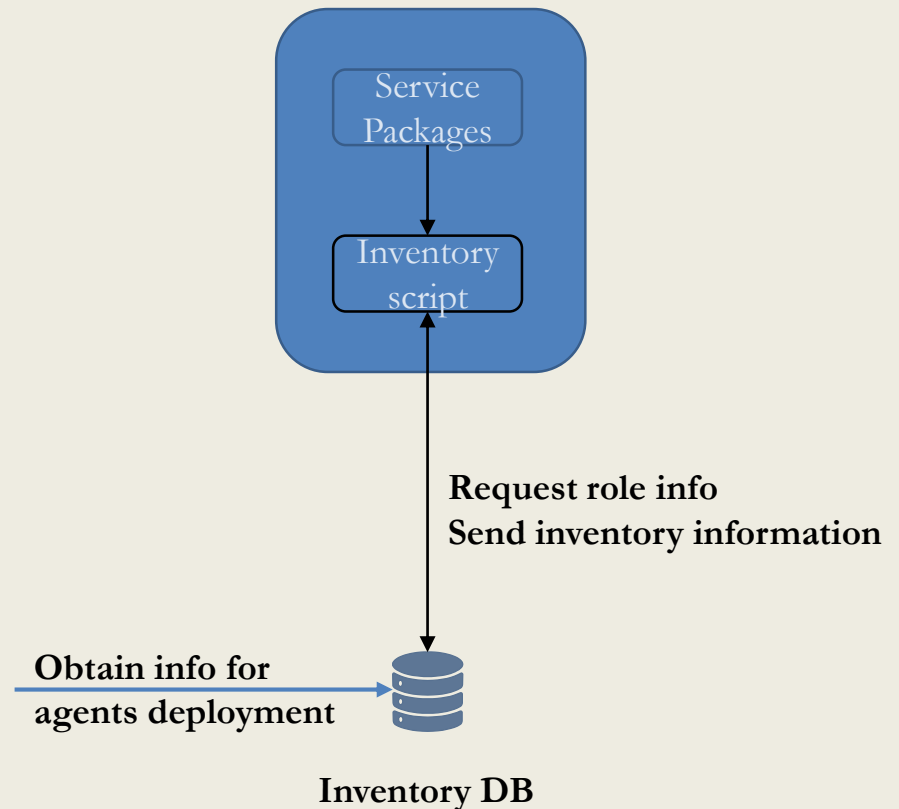


# Deployment preparation

- Preparing repository accessibility
- Prepare DNS and DHCP
- Preparing of system images using separate repositories for different types of services:
  - Storage
  - Computing
  - Networking
  - Agents
- Preparing of monitoring and DB facilities and messages queue service
- Preparing Inventory DB:
  - Hardware information
  - MAC-addresses
  - Logical distribution

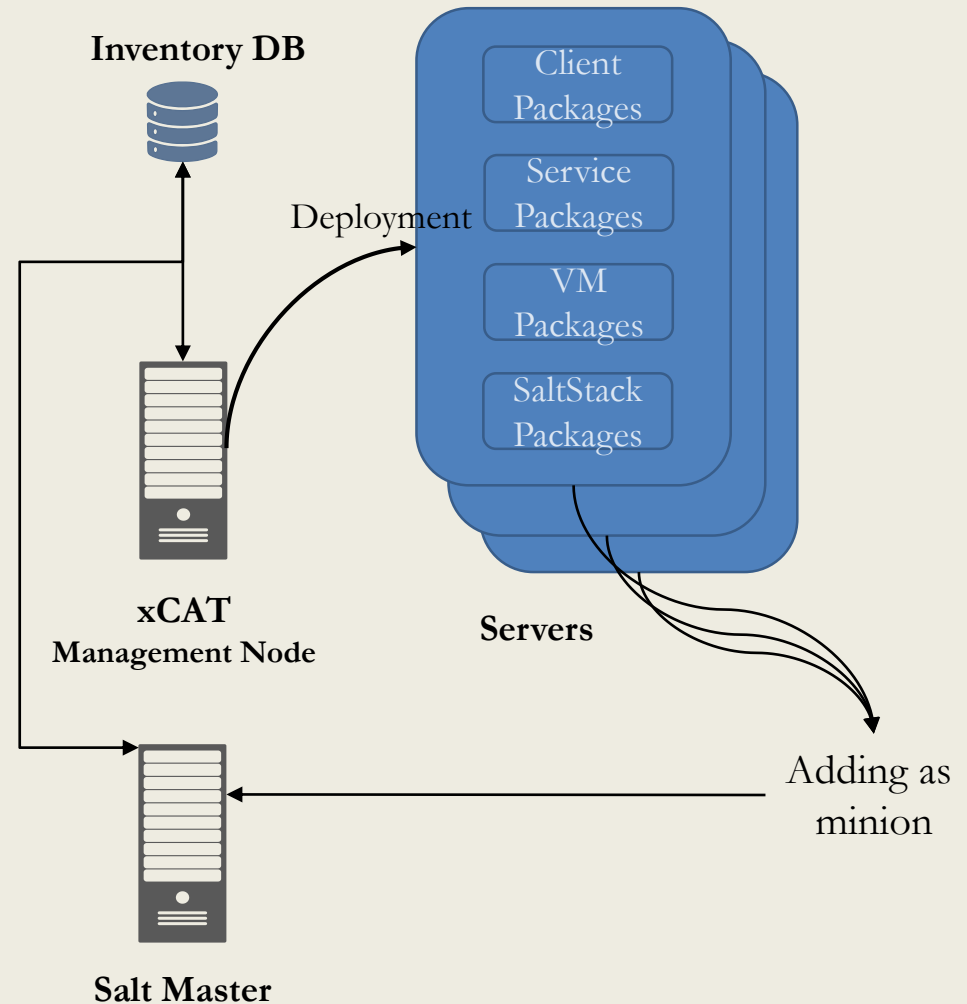
# Inventory of servers

- Scripts for provision Inventory DB
- Works under base OS
- Automatically add info about NICs, disks, RAM and CPU
- Uses under base server and VMs



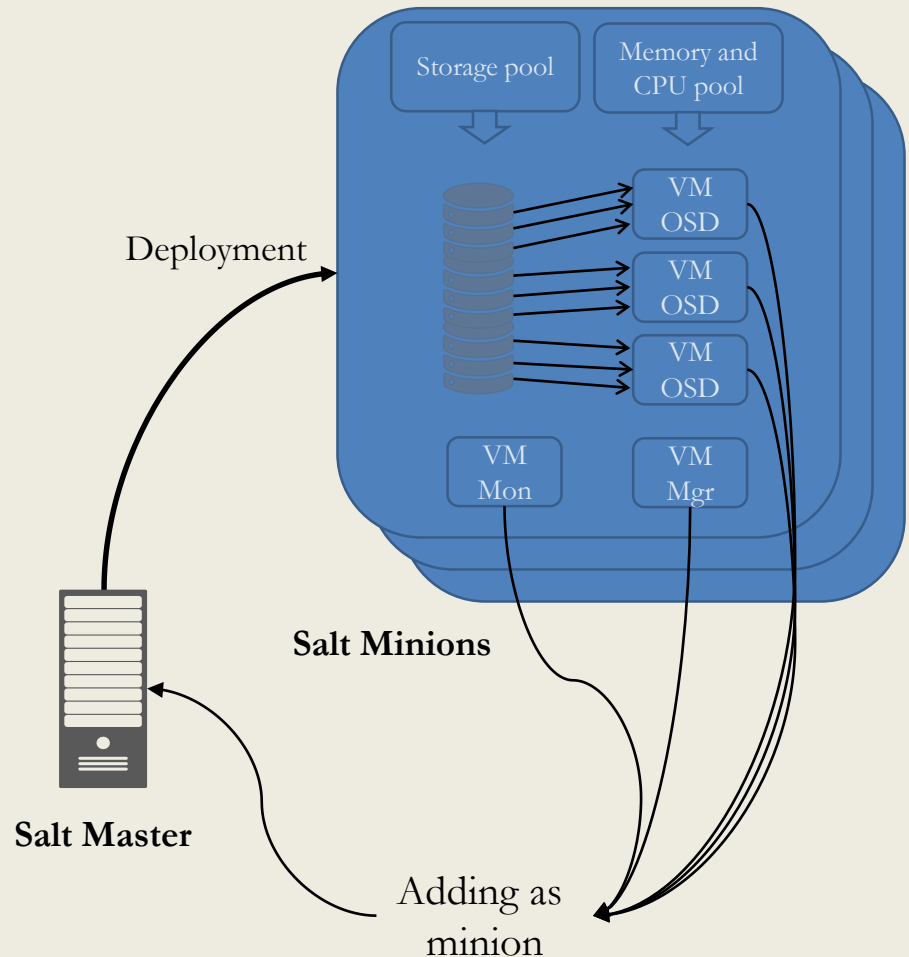
# Base system deployment

- Separate project repositories for different types of services
- Deployment of base system (Naulinux distribution based on Scientific Linux)
- Automating the installation of packages after deployment for agents
- Installing Salt Minion packages

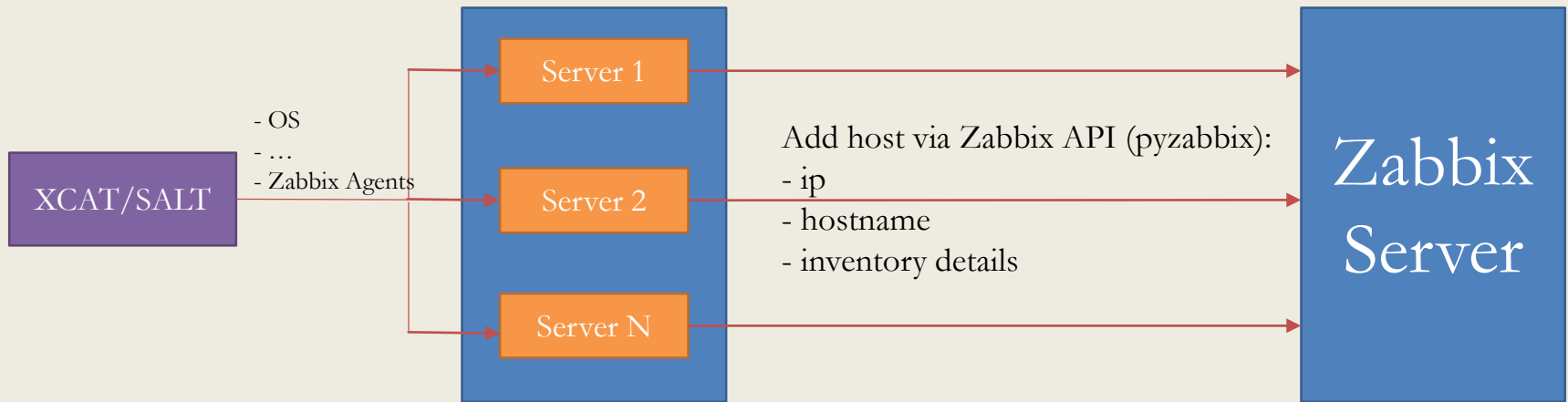


# Storage subsystem deployment

- Automatically provision of node`s disks
- Automatically filtering of disks and checking their accessibility
- Check for the availability of the necessary hardware resources for optimal use of the server
- Group disks for use as an OSDs
- Installing required software, preparing Salt state files and deployment scripts
- Adding VMs as minions to Salt



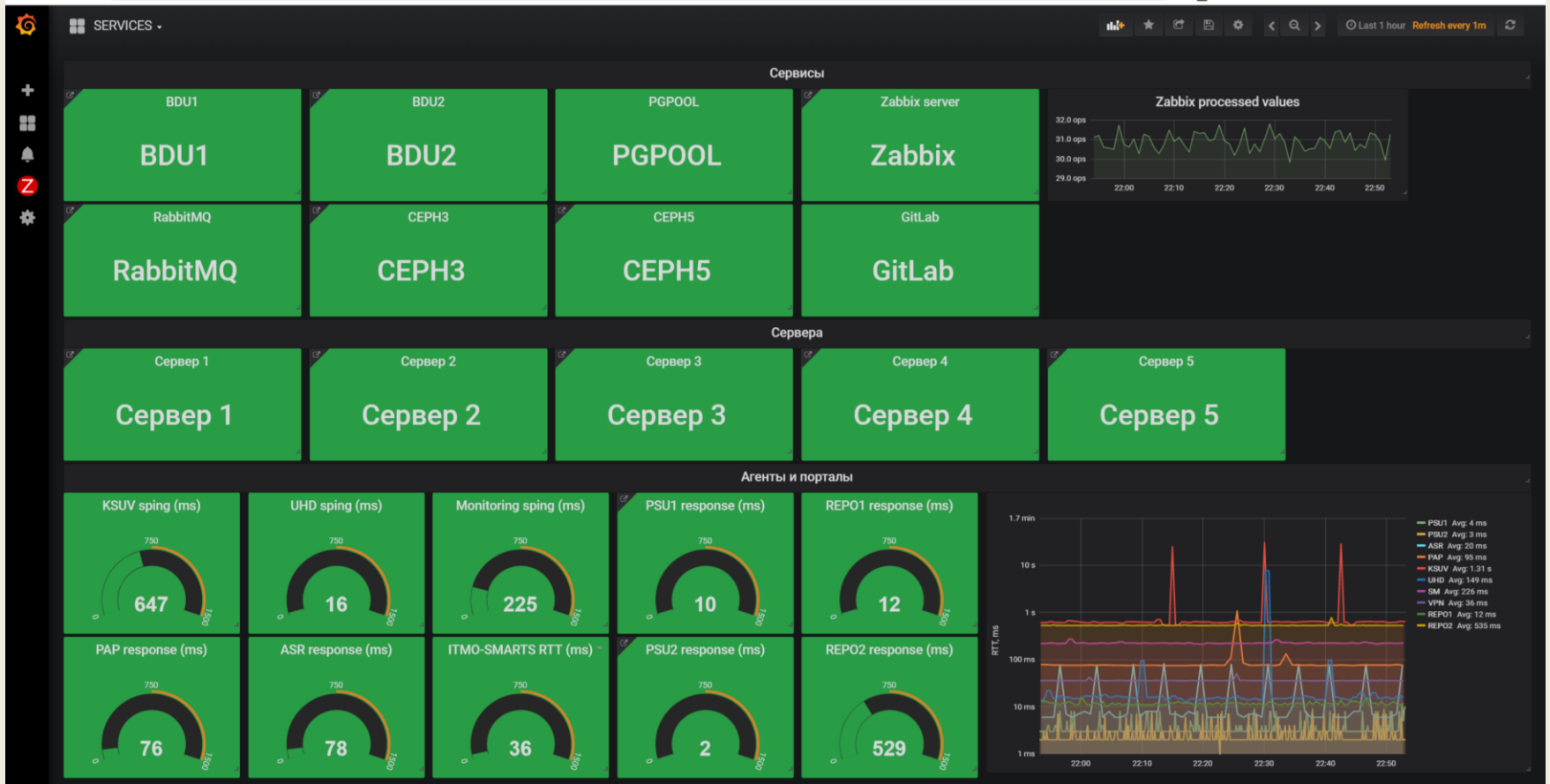
# Configuration sequence for Monitoring system



- Install OS & services
- Install Zabbix agent
- Configure Zabbix agent
- Send Zabbix agent configuration and host information to Zabbix server via Zabbix API
- Host starts being monitored



# Monitoring system



# Inventory, state monitoring, detriment estimation



- Notify about problems on server
- Find what services are damaged
- Calculation of possible damage
- Send information about services for failover
  - Verify the possibility of recovery / transfer of services
  - Fire redeployment if needed

# Current status

- The system management is under heavy debugging.
- Experimental distributed storage deployment is available for testing.
- Monitoring system is working.