Monitoring system of the "Pi of the Sky" experiment.

Arkadiusz Ćwiek
arkadiusz.cwiek@fuw.edu.pl

National Centre for Nuclear Research
in Otwock-Świerk, Poland

Project POLISH-SWISS ASTRO PROJECT supported by a grant from Switzerland through the Swiss Contribution to the enlarged European Union

Presentation plan

- Introduction
- Why we need monitoring?
- Icinga and why we used it?
- Impelemtation in Pi of the Sky
- To do
- Conclusions

Introduction

Pi of the Sky:

- network of robotic telescopes,
- all telescopes operate fully autonomously, without human intervention.

Currently, there are two telescopes:

- Working prototype aparature in San Pedro de Atacama in Chile,
- The first telescope of the final systemin in the INTA El Arenosillo test centre in Mazagón near Huelva, Spain.

Before the end of this year should be another 3 new units in the INTA.

Functionality expected from a monitoring system

- Monitoring any network and all its resources
  - Network services: SSH, HTTP, SNMP, PING, etc.
  - Host resources: CPU load, disk utilization, system logs, etc.
  - Server components: Switches, routers, temperature and humidity sensors, etc
  - **Any device or service** for which we are able to write a plugin that reads the parameters and returns them.
- Notify – when issues arise and are resolved

These requirements are met by Icinga

Why we need monitoring?

- The system work you need to keep control.
- In each telescope, we should check:
  - status of the dome
  - status of the computers that support the telescope (internet connection, harddisk, CPU temperature etc.)
  - Status mount.
  - Status of each of the 4 (INTA) cameras, such as:
    - chip temperature
    - frame size
    - number of stars
    - Astrometry
    - etc.

At 5 units, this can not be manually browse, we need a dedicated monitoring system.

What is Icinga?

- fork of Nagios, backward compatible.
- enterprise grade open source monitoring
  - licensed under GPL V2 and is free to use, distribute and modify,
- monitoring network and any network resource
  - notifies the user of errors and recoveries
  - generates performance data for reporting
- scalable and extensible,
  - can monitor complex, large environments across dispersed locations,

Why we used the Icinga and not Nagios?
What does Icinga have that Nagios doesn’t?

- Retains all the existing features of its Nagios.
- Added many long awaited patches and features requested by the users.
- Icinga can store data in:
  - text files,
  - MySQL,
  - **PostgreSQL**,
  - Oracle.
- Nagios supports only text files and a MySQL database.

Why we used the Icinga and not Nagios?
cont.

- Icinga was forked from Nagios with a clear objective to preserve its open source nature,
- All its versions are available under GPL and will always remain to be.
- Is developed by a large team of monitoring enthusiasts from around the world.
- Bug fixes and new features are constantly released
- Public roadmap is testimony to the team’s commitment.

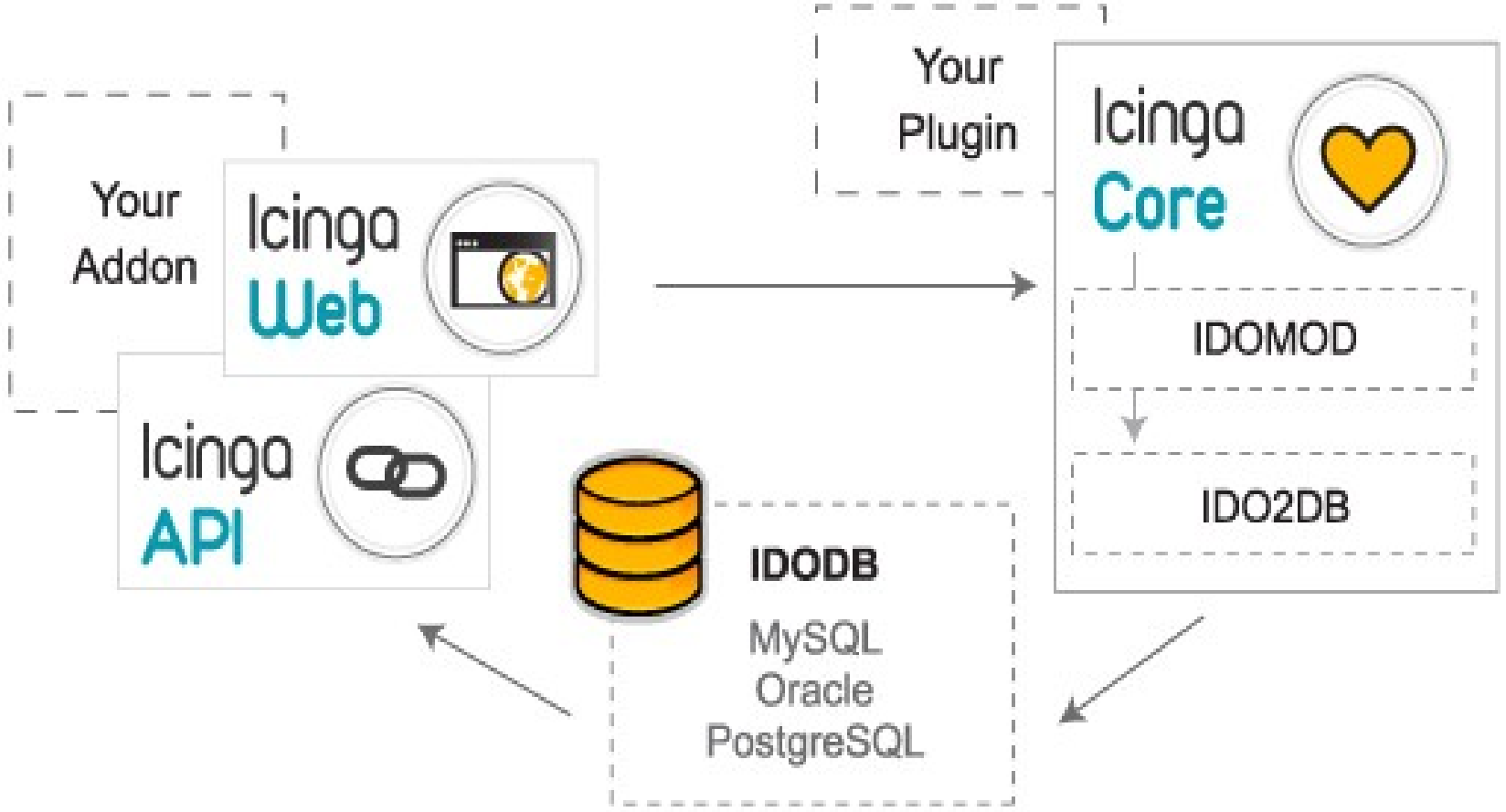
Enterprise grade extensibility & authentication

- From one to thousands of hosts, services, instances
  - Icinga’s plugin and modular structure offers limitless extensibility.
- All Nagios extensions are Icinga compatible,
- Icinga’s own Doctrine abstraction layer, REST and plugin APIs make designing new extensions easier.
- Highly refined authentication system which enables user access, notifications and views to be customised to the detail of server groups, servers and services per individual.

Icinga architecture

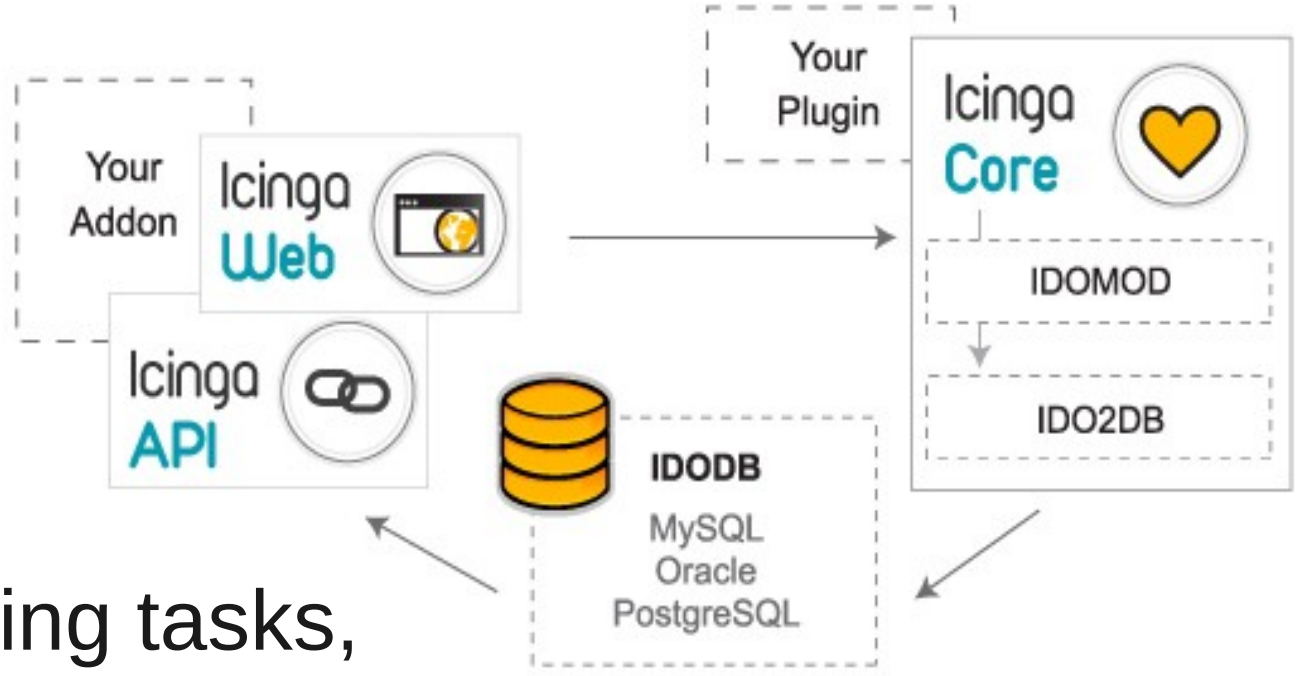
Icinga consists of 3 components which work in tandem:

- Core,
- API,
- Web.



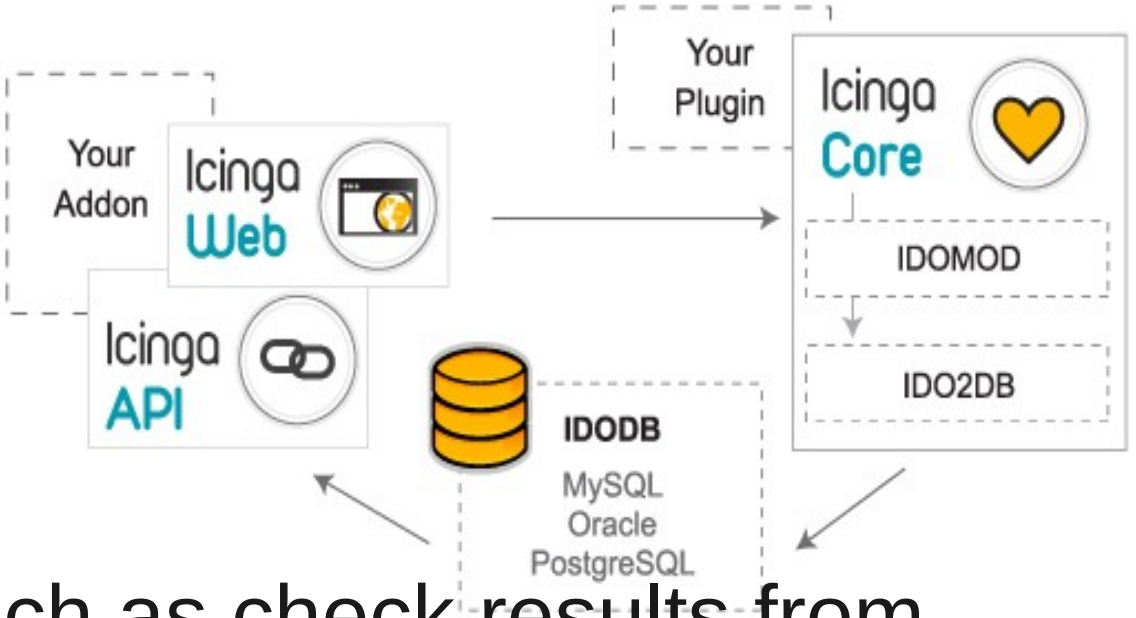
And common database, the IDODB (Icinga Data Out Database).

Icinga Core:



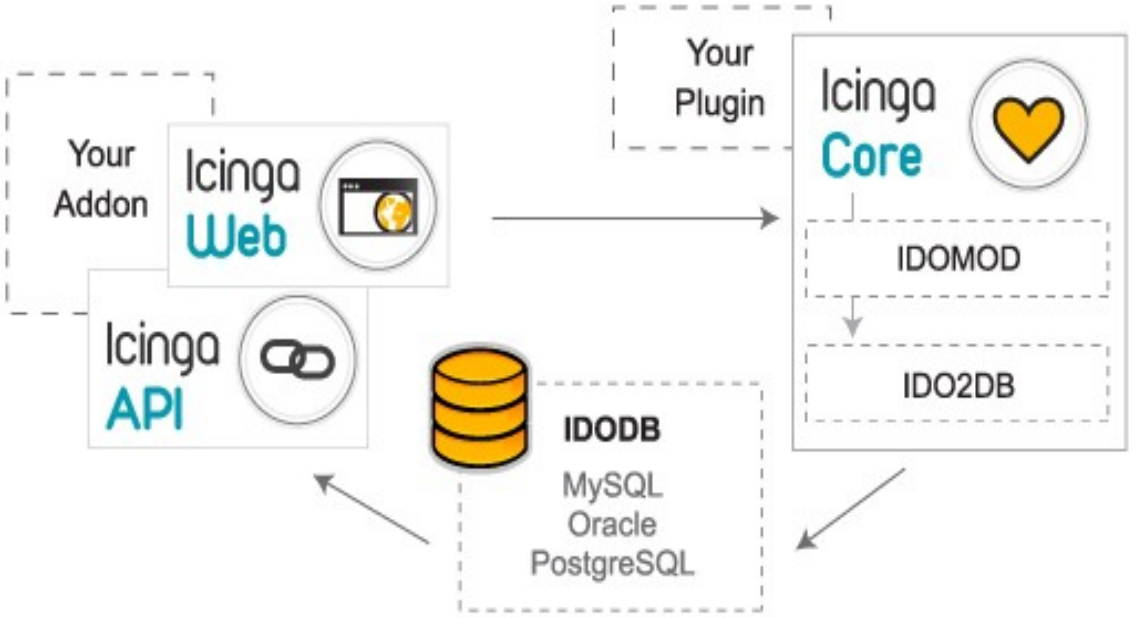
- manages monitoring tasks,
- receiving check results from various plugins.
- communicates these results to the IDODB through the IDOMOD interface and IDO2DB service daemon over SSL encrypted TCP sockets.

Though both come packaged (previously known as IDOUtils) with the Core; they are single standing components which can be separated to distribute the data and processes across multiple servers.



Icinga API:

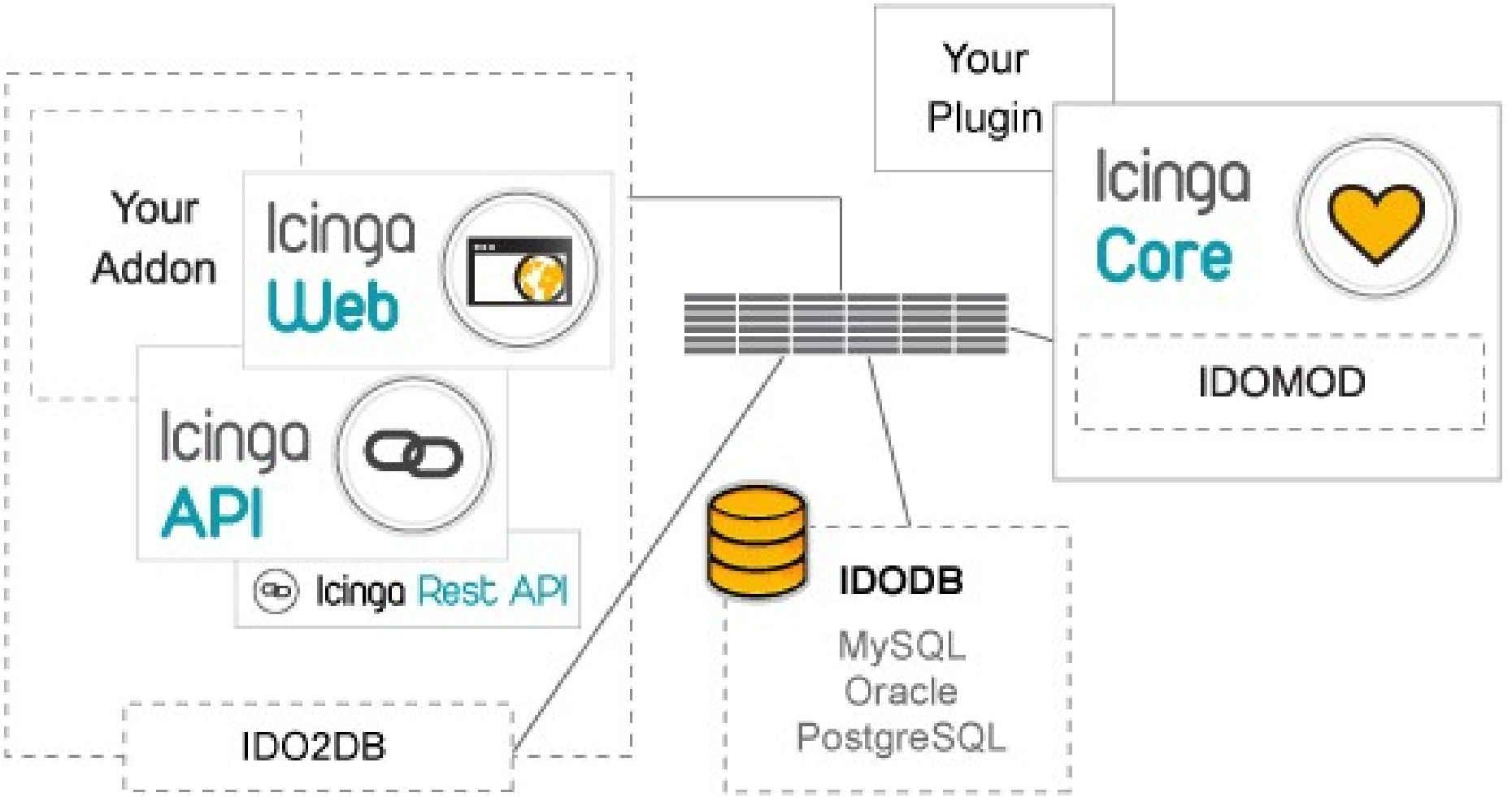
- Fetches information such as check results from the core, through the IDODB.
  - Icinga can support the most popular relational database management systems.
- PHP based, the Icinga API is free from complex data schemas, and has no dependencies on other libraries or frameworks.
- Supports various interfaces, from database PHP-PDO output to pipe and SSH input.



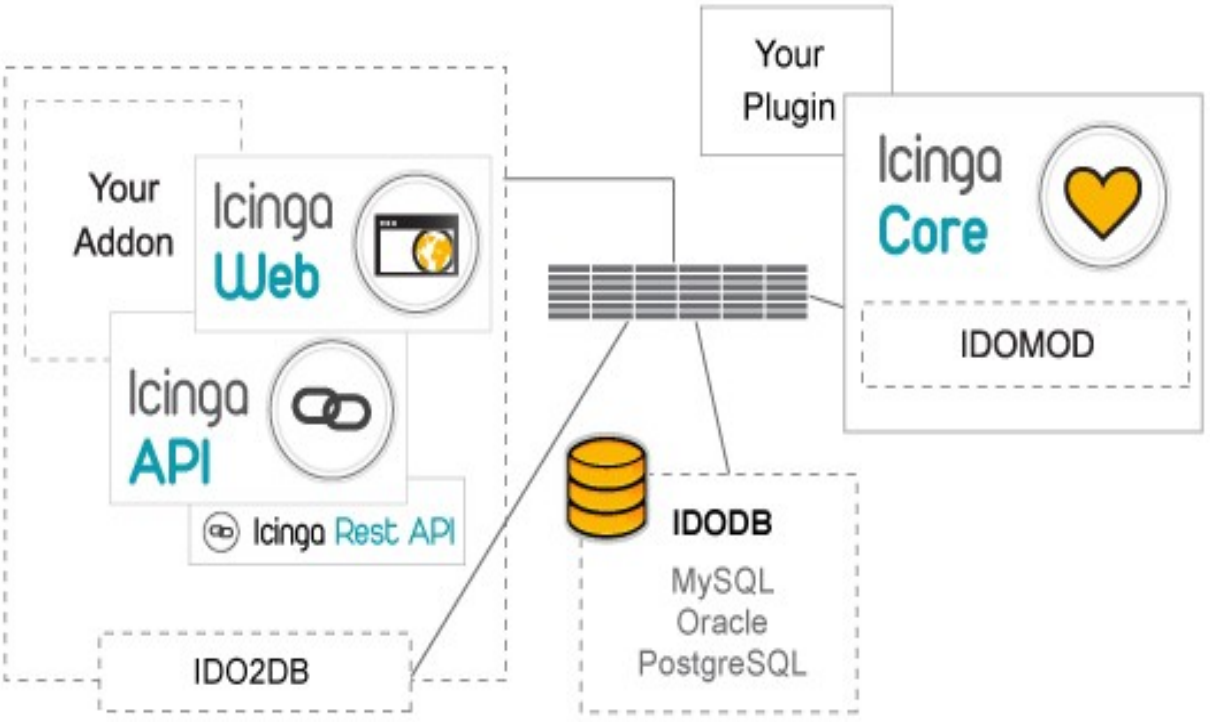
Icinga Web:

- Icinga Web is in essence an online portal to view Icinga monitoring results and send commands to Icinga Core.
- Here host and service status, history, notifications and status maps are available to keep a check on the health of your network in real-time.
- A graphing addon such as PNP or GrapherV2 can be integrated to Icinga Web to generate performance charts for reporting.

Icinga distributed architecture



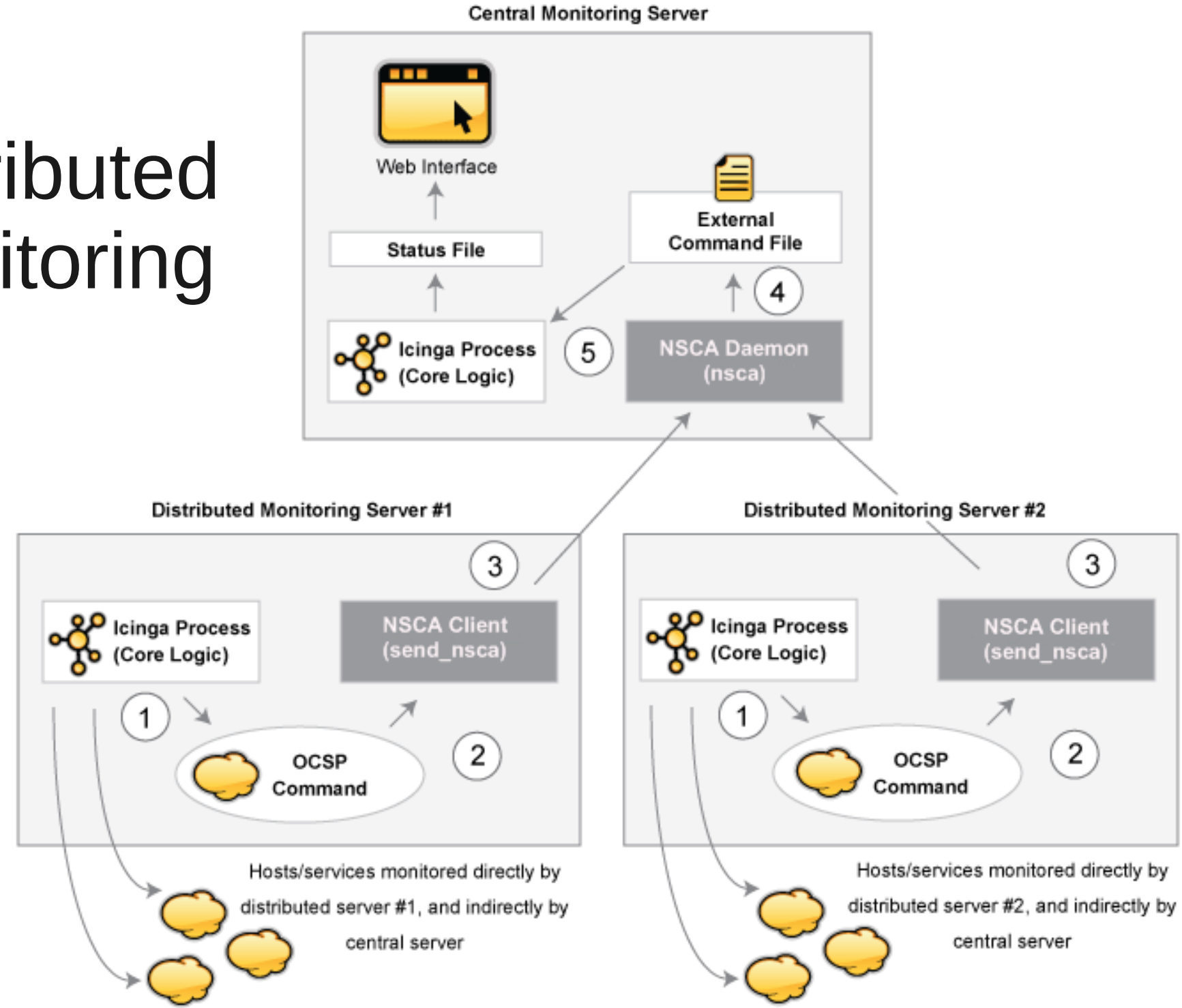
Icinga distributed architecture



Thanks to the loose bundling of Icinga Core, Web, API and database, these components can be distributed and connected by a switch or any other intermediary.

Such systems ensure that the monitoring system itself is fail safe, so should one component fall out, it can be replaced without disturbing the system as a whole.

Distributed Monitoring



Monitoring system of the "Pi of the Sky" experiment.

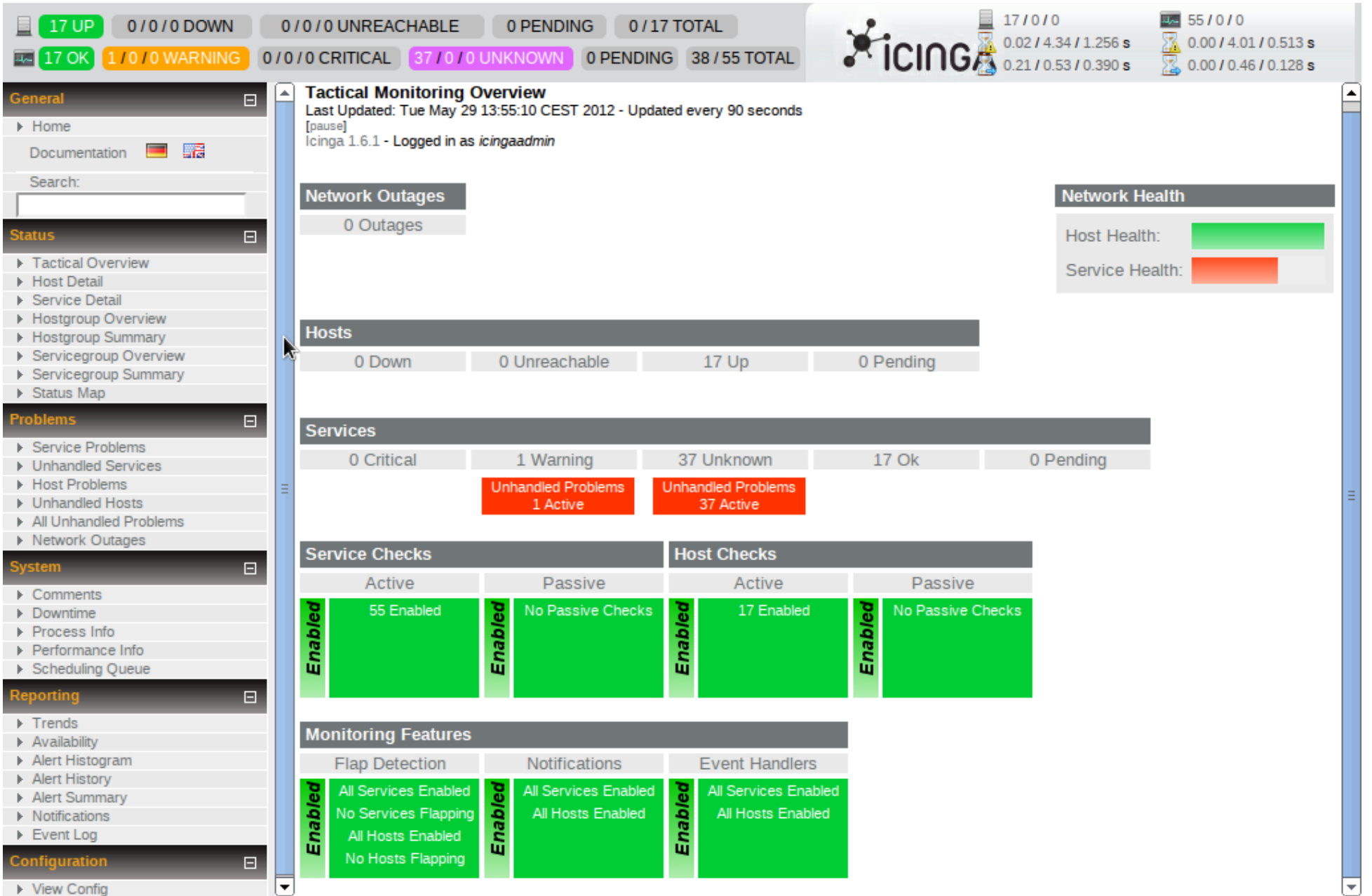
- First version of Icinga plugin to monitor different system components written.
  - Parses status files which are created by different system tasks (DAQ, mount, piman)
  - Standard plugins used to monitor network and computers
- Icinga monitoring impemented for telescopes in:
  - INTA (Spain)
  - SPdA (Chile)

To do:

- Creation of virtual machine disk image with installed Icinga, and necessary plugins, add-ons, and the initial configuration.
- Installation in the INTA and SPdA KVM(Kernel-based Virtual Machine) and so prepared virtual machines or direct install server Icinga.
  - parse status files localy instead of downloading them to Warsaw
  - store minitoring results localy in case of network problems
- Implement plugins for monitoring additional resources (eg. power supply voltages)
- Writing a plugin directly monitoring the condition of the dome and not by pressing the status file.
- Modifying the current plugin to collect data on the state service directly through CORBA. (This may require modifying the code of those services)

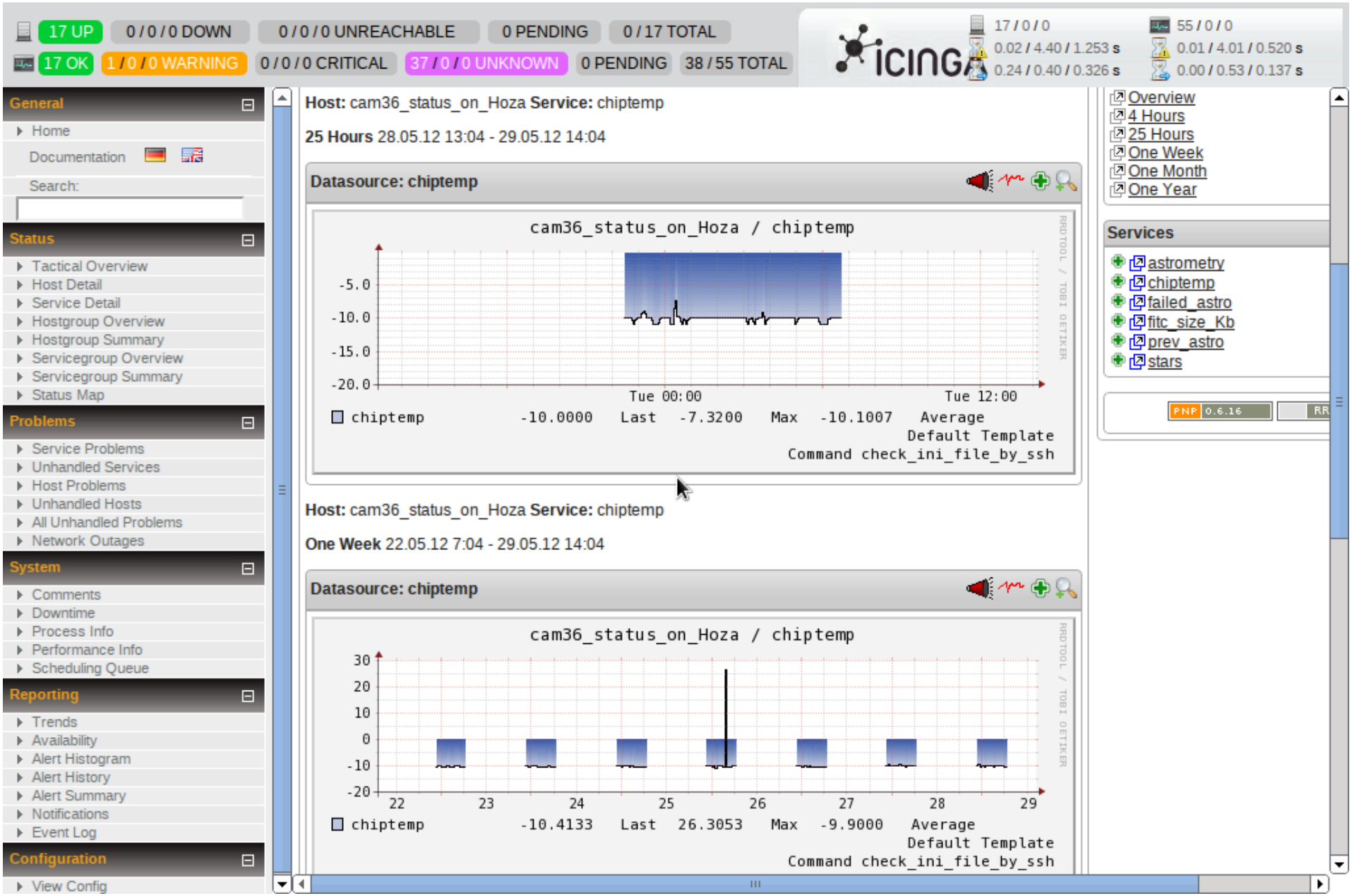
"Pi of the Sky" Monitoring

sample screen shots



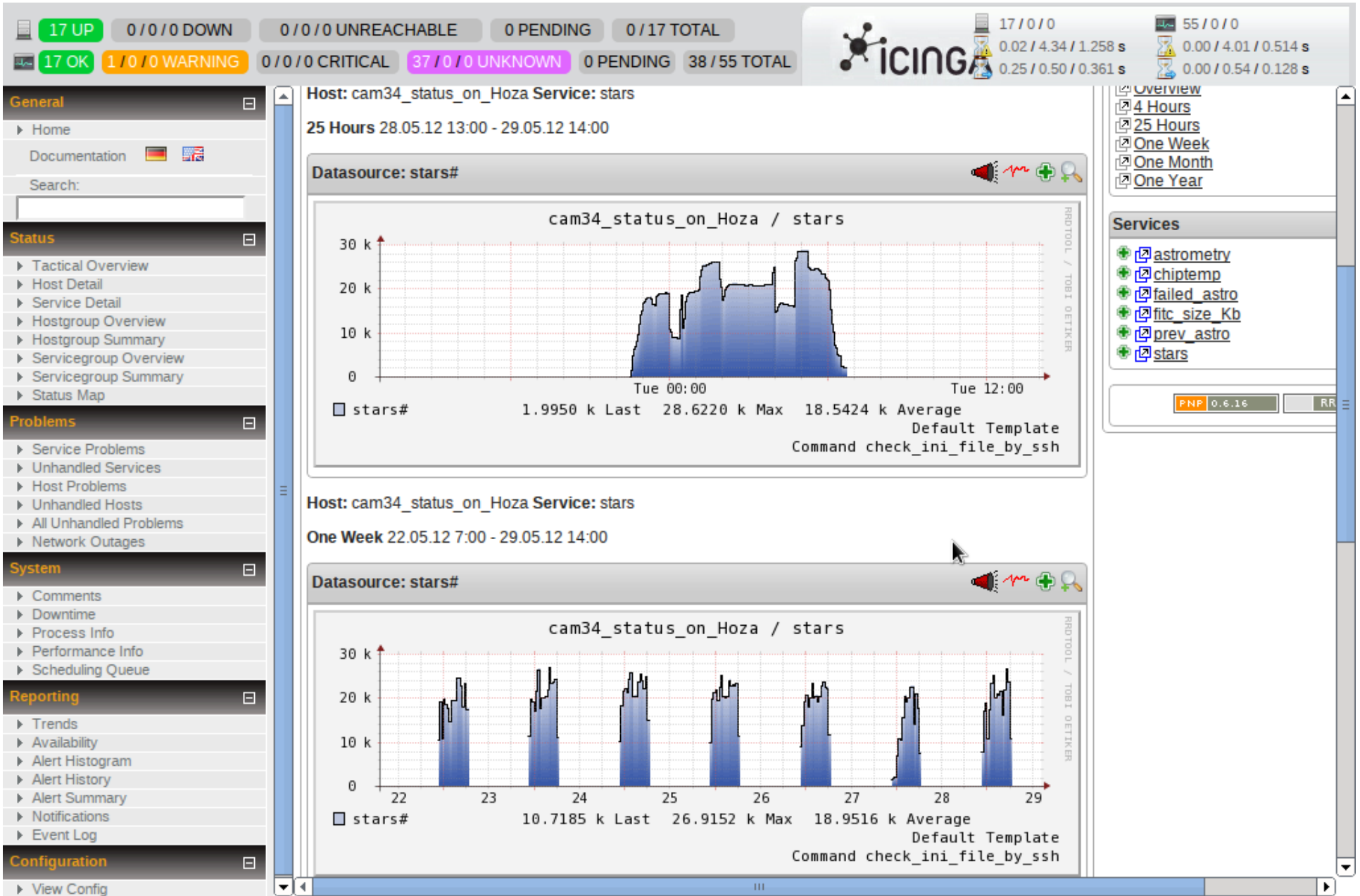
"Pi of the Sky" Monitoring

sample screen shots (cont.)



"Pi of the Sky" Monitoring

sample screen shots (cont.)



Summary.

- The system allows to continuously monitor vital parameters of detectors and computers which control them.
- It facilitates a fast response to anomalies or failures in system's performance.
- Automatic recording of parameters of all components of the "Pi of the Sky" system allows an efficient identification of possible causes of eventual failures.