Development of the active monitoring system for the computer center at IHEP

V. Anshukov[1], V. Ezhova[1], V . Gusev[1], A. Kotliar[1] ,V. Kotliar[1*], G. Latyshev[1] ,A. Shishov,

[1]State Research Center of Russian Federation Institute for High Energy Physics, RU-142281, Protvino, Moscow region, Russia

E-mail: {Vladimir.Anshukov, Victoria.Ezhova, Victor.Gusev, Anna.Kotliar, Viktor.Kotliar, Grigory.Latyshev, Artur.Shishov}@ihep.ru

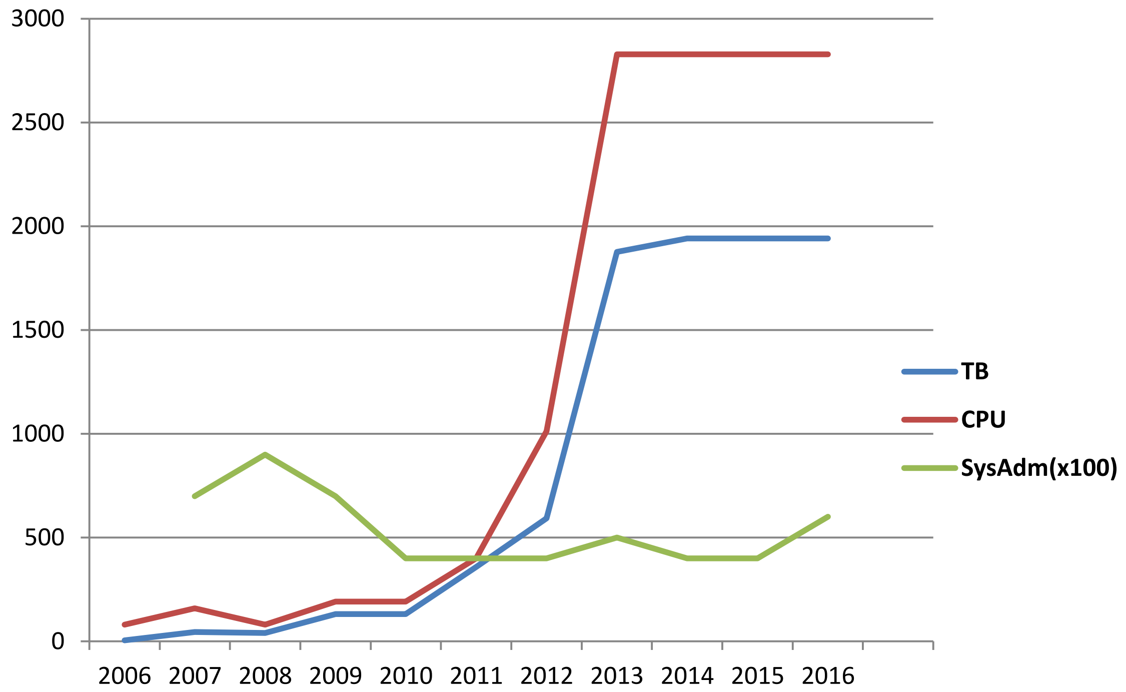
[*] Corresponding author

Computer center at IHEP is a complex system of many technologies gathered together. Among them distributed computing, high throughput networking, high reliable uninterruptable power systems, precision cooling systems. Monitoring and control such complex is a very difficult task. Even more difficult is to create self-optimization, self-healing and self-defense systems on top of the monitoring. As a first step it might be a creation of several databases to accumulate all information about center infrastructure, events, logs, statutes and then as second step a creation of an active monitoring system which could be able to perform simple tasks itself or make advices for the human interventions. The current status of the development of such system for the IHEP computer center described in this work.

Computer center at IHEP 50 years of history

1965 – Minsk-2, Minsk-22, M-220, BESM-4, BESM-6, Minsk-32, EC-1040, EC-1045
1972 – ICL
1977 – DEC 10
1991 - mVAX-II – mail
1993 – Internet
2003 – Grid-cluster
2011 – 10Gb/s

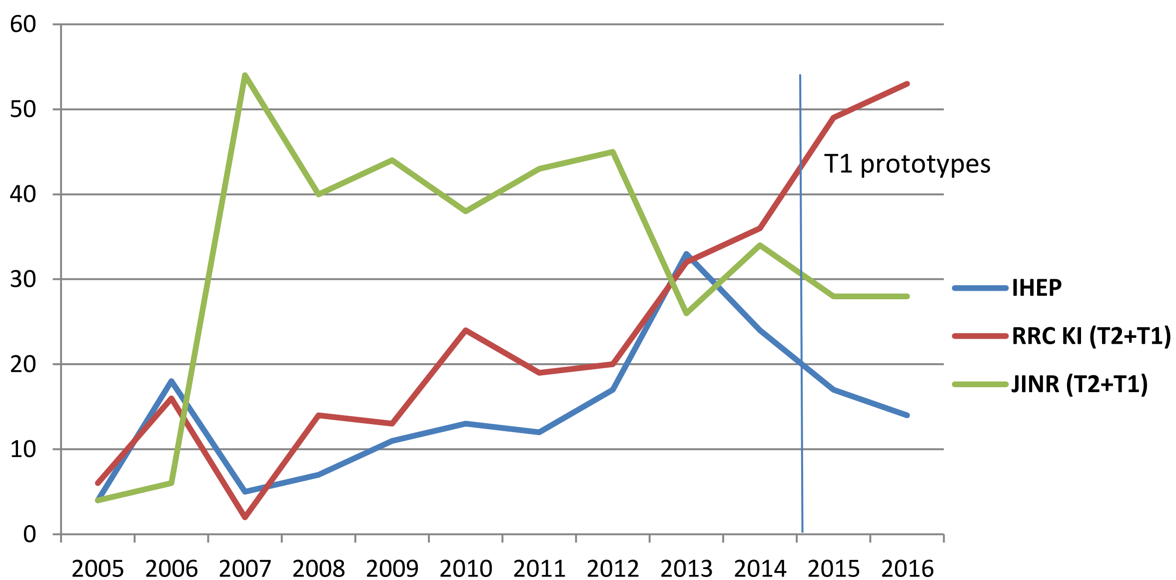
Computer center at IHEP resources evolution



growth of the IHEP grid resources by year in TB and CPU

Current status: resourses

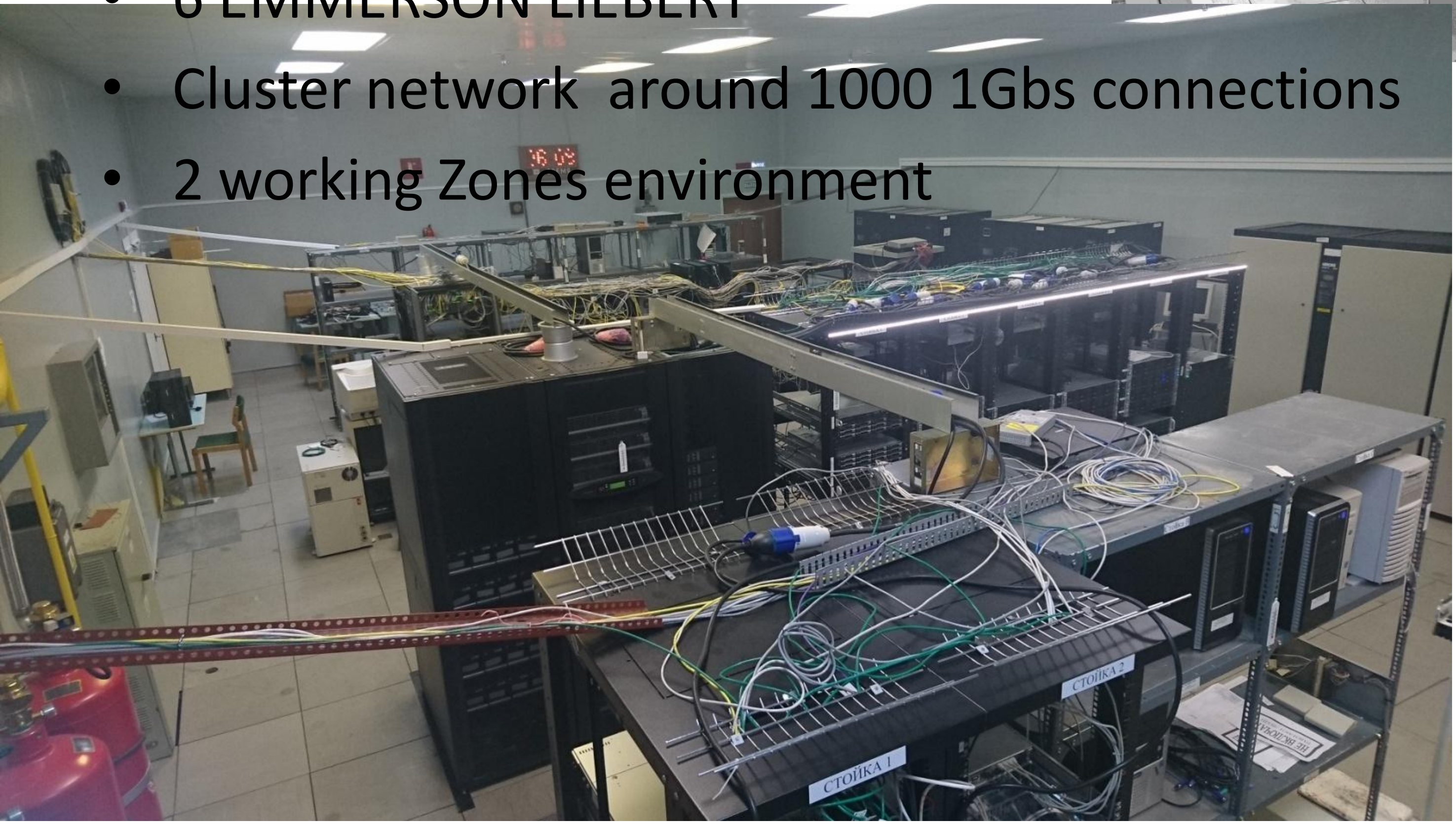
- 2828 CPU, 24390 HEP-SPEC06;
- 1942 TB: Atlas 1185, CMS 395, Alice 297, LHCb 65;
- 2x10Gb/s Internet channels
  (LHCONE shared with RDIG 10Gb/s);
- Manpower – 5 people;
- one of three big grid-sites in Russia:



Contibution in % to RDIG by normalised CPU time

WHAT to monitor?

- 2828 CPU (150 servers);
- 1942 TB (50 servers);
- 2 APC symmetra + 30 small UPS + 26PDU
- 6 EMMERSON LIEBERT
- Cluster network around 1000 1Gbs connections
- 2 working Zones environment



HOW is monitored?

Depends on needs there are several monitoring systems:

- Nagios to check computer centre services: 3500 services;
- Splunk and cerntral syslog to get and analyse all logs: 591,328,937 events for 4 years ;
- collectl – realtime monitoring: 160servers on one screen;
- ESK for engineering infrustructure: 2 symmetra + 1 cooling system;
- Munin rrd monitoring for ipmi sensors on all nodes;
- Self-build PNPI monitoring scripts for sensors over IPMI ;
- pmact + cacti for network traffic monitoring;
- Self-guard local monitoring on servers for IPMI events and temperature limits;
- Self-guard local monitoring on servers for UPS events;
- Big Red Button for safetly switching all servers manually;
- Self-build accounting system.

HOW is monitored? 2

GRID monitoring for all services:

- dCache internal monitor for stuck transfers, big queues, enabled pools;
- Regional nagios for Russian Grid-sites;
- CERN check_mk for all LHC experiments;
- Operation portal security dashboard;
- Midmon nagios for GRID services;
- Cluster jobs efficiency monitoring;
- CMS dashboard for readiness tests;
- Alice site monitoring for xrootd and vobox services;
- Atlas panda jobs monitoring for atlas specific problems on cluster (production and analyse jobs);
- GRID accounting system.

Current problems

Too much monitoring systems:

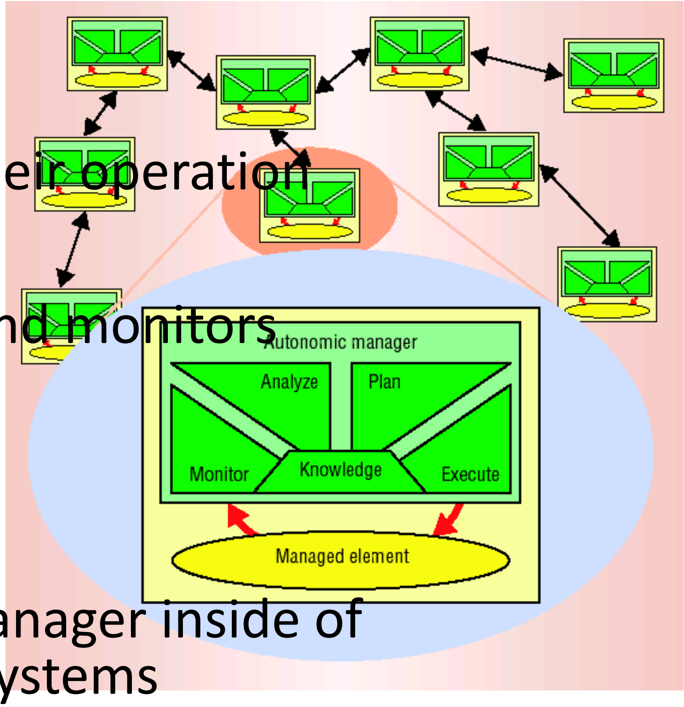
- difficult to maintain (support, upgrade, backup);
- difficult to use (need computer center control room with many monitors);
- difficult to use programms for anlalyse data and make desisions (no single API, no single DB);
- all systems have their data format and storage (rrd, mysql, nosql, text, JSON,...)

Limitations:

- No algorithms for clever alarms (usually only threshold limits)
- No data analyse interface (when you need to check new parameters which were not monitored)

What we want: theoretically

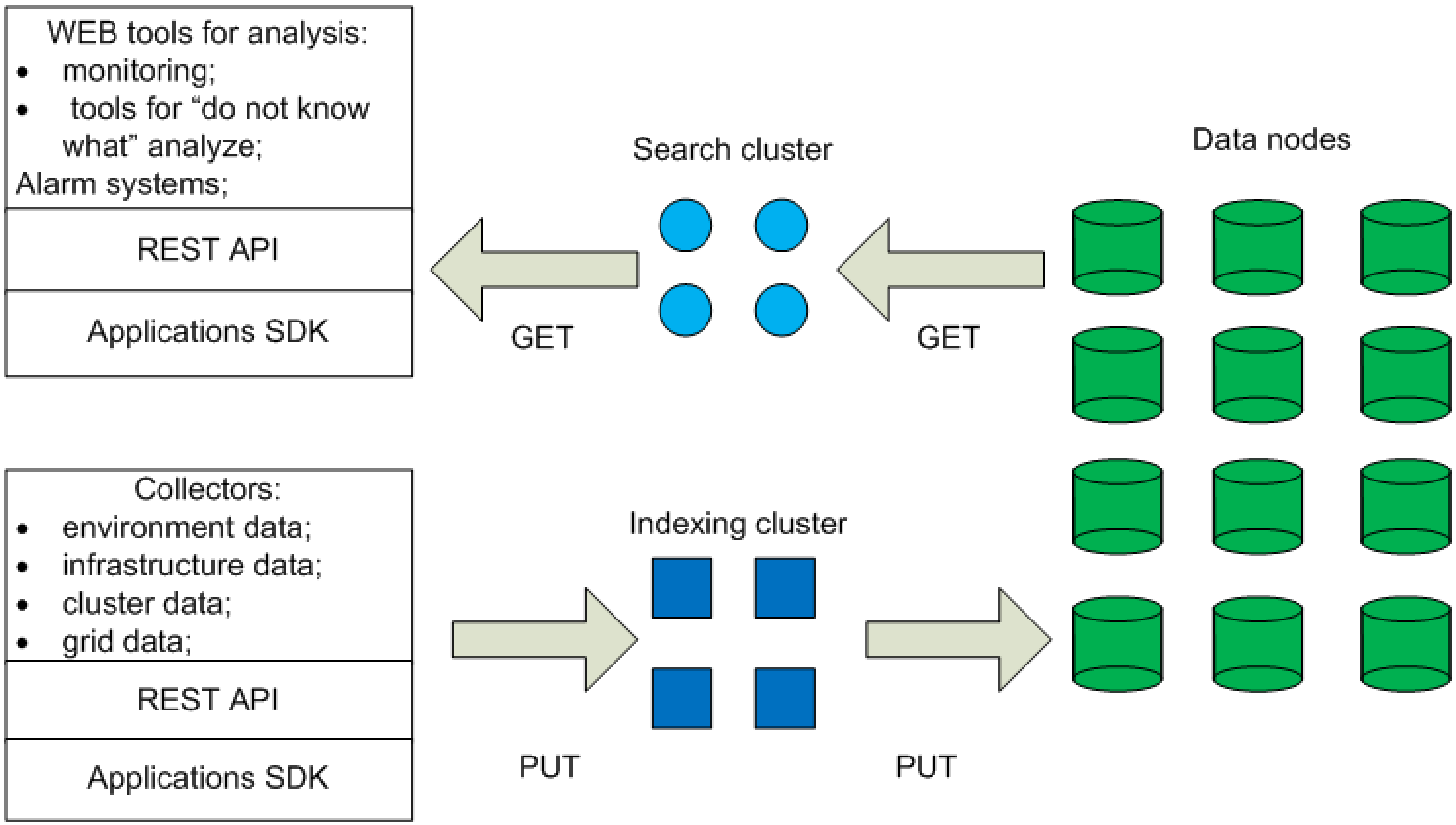
- Four aspects of self-management
  - Self-configuration
    - Configure themselves automatically
    - High-level policies (what is desired, not how)
  - Self-optimization
    - Hundreds of tunable parameters
    - Continually seek ways to improve their operation
  - Self-healing
    - Analyze information from log files and monitors
  - Self-protection
    - Malicious attacks
    - Cascading failures
- Monitor is a core part of the Autonomic manager inside of autonomic element for self-management systems



What we want: practically

- Collect information for all available sensors inside computer center for offline or online analyze;
- Make all collected information easy reachable for programs, scripts, humans;
- Develop the logic for the self-defense computer center from: power cuts, cooling problems, fire
- Develop the logic for automatic detection for anomalies in hardware equipment (UPS, Cooling)
- Develop the logic for data center optimization: job efficiency, cooling efficiency

What we want: architecture



Development: where we are

WEB tools for analysis:
Kibana

Alarm systems:
Symmetra alarms

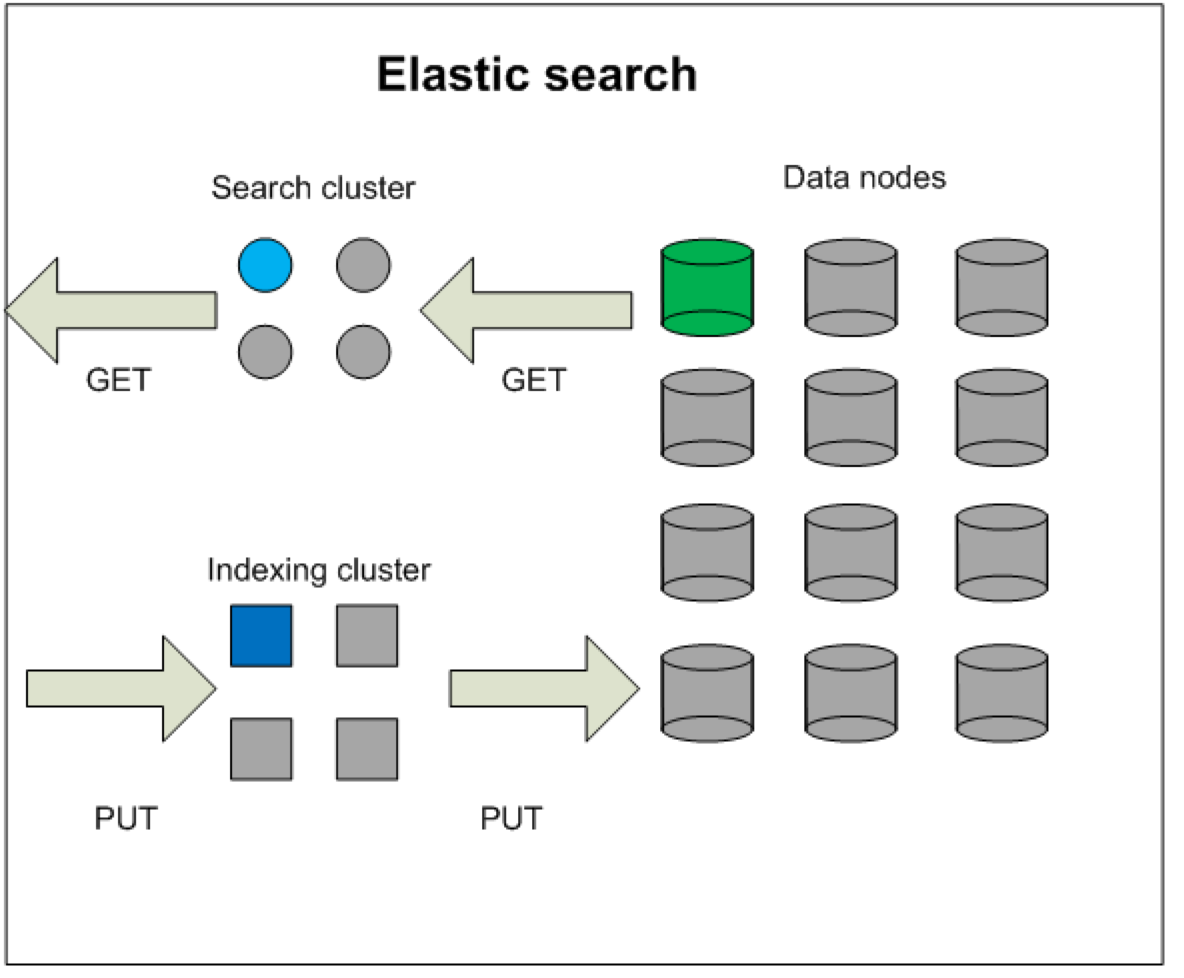
REST API

Applications SDK

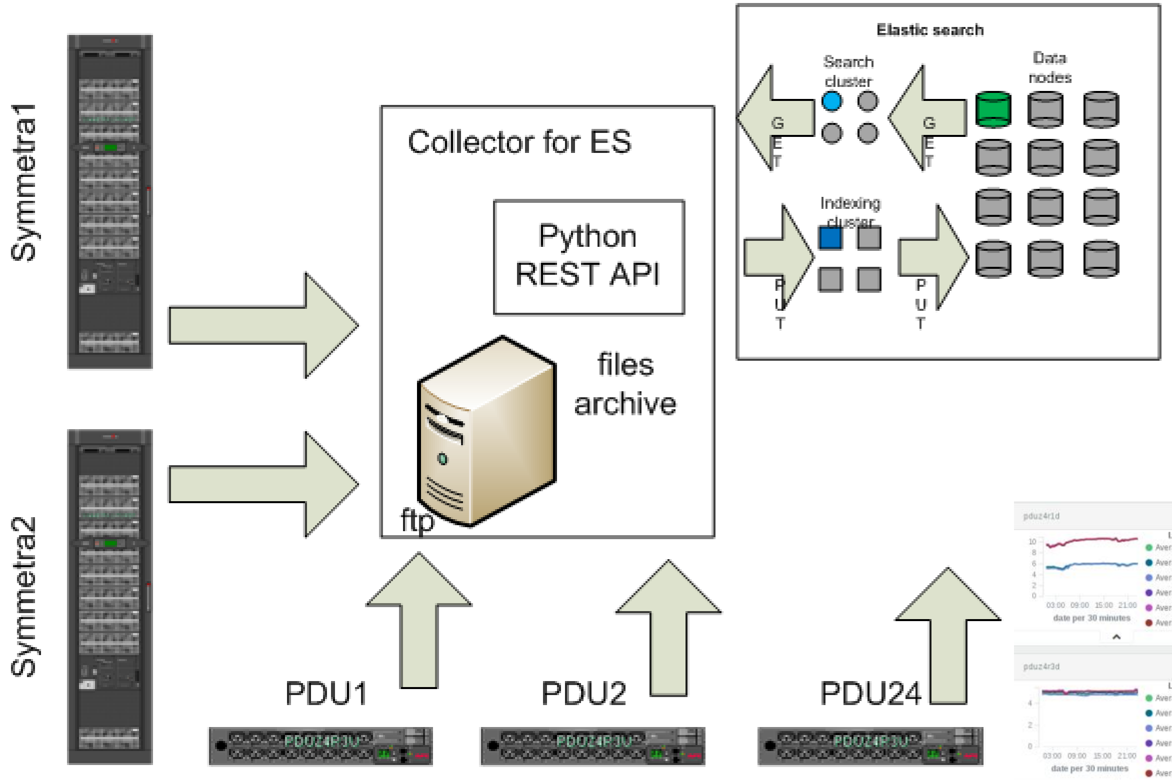
Collectors:
- environment data;
- infrastructure data; **symmetra, pdu, cooling system, ups**
- cluster data;**gpu usage, xen usage**
- grid data;

REST API

Applications SDK



Development: where we are power



Distribution Output Current

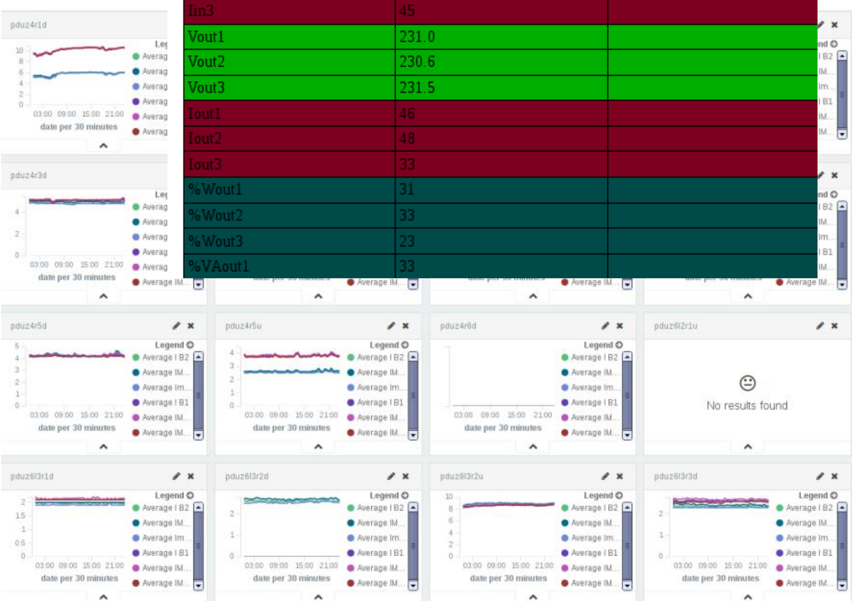
L1-2min	45.1	
L2-3min	46.7	
L3-1min	33.2	
L1-2max	46.3	
L2-3max	47.7	
L3-1max	34.1	

Total Output Load

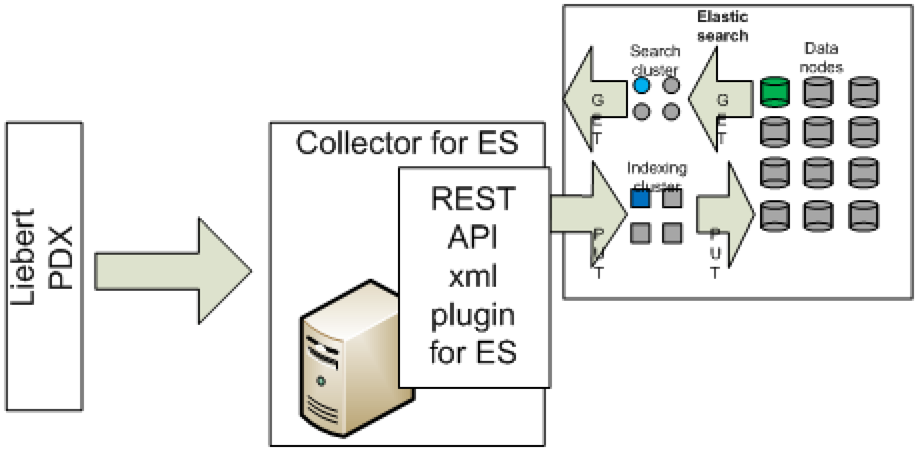
kVA	28.6	
kW	27.5	
PF	0.96	

Symmetra 160K

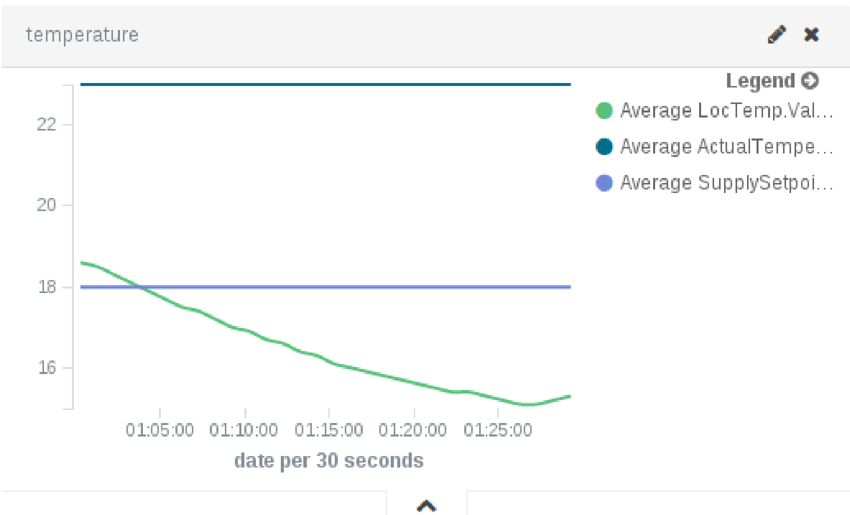
Vmin1	229.1	
Vmin2	229.1	
Vmin3	226.1	
Vmax1	231.7	
Vmax2	231.2	
Vmax3	228.8	
Iin1	44	
Iin2	43	
Iin3	45	
Vout1	231.0	
Vout2	230.6	
Vout3	231.5	
Iout1	46	
Iout2	48	
Iout3	33	
%Wout1	31	
%Wout2	33	
%Wout3	23	
%VAout1	33	

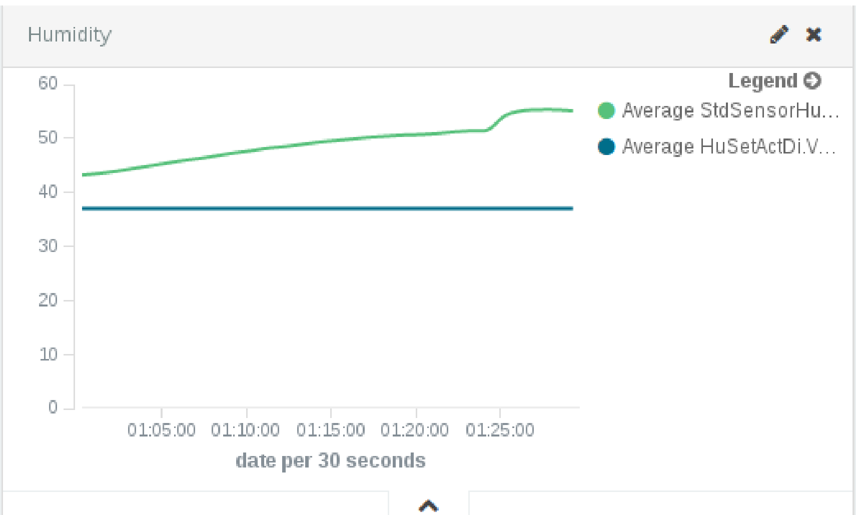


Development: where we are cooling



Time	LocTemp.Value.	StdSensorHumidity.Value.	ShowMaint.Value.	ShowCool.Value.
June 9th 2016, 01:29:01.000	15.3	55.1	off	off
June 9th 2016, 01:28:01.000	15.2	55.3	off	off
June 9th 2016, 01:27:01.000	15.1	55.3	off	off
June 9th 2016, 01:26:01.000	15.1	55.1	off	off
June 9th 2016, 01:25:01.000	15.2	54.2	off	off
June 9th 2016, 01:24:01.000	15.3	51.6	off	off
June 9th 2016, 01:23:02.000	15.4	51.4	off	off
June 9th 2016, 01:22:01.000	15.4	51.2	off	off
June 9th 2016, 01:21:01.000	15.5	50.9	off	off
June 9th 2016, 01:20:01.000	15.6	50.7	off	off
June 9th 2016, 01:19:01.000	15.7	50.6	off	off
June 9th 2016, 01:18:01.000	15.8	50.4	off	off
June 9th 2016, 01:17:01.000	15.9	50.2	off	off





Development: achievements

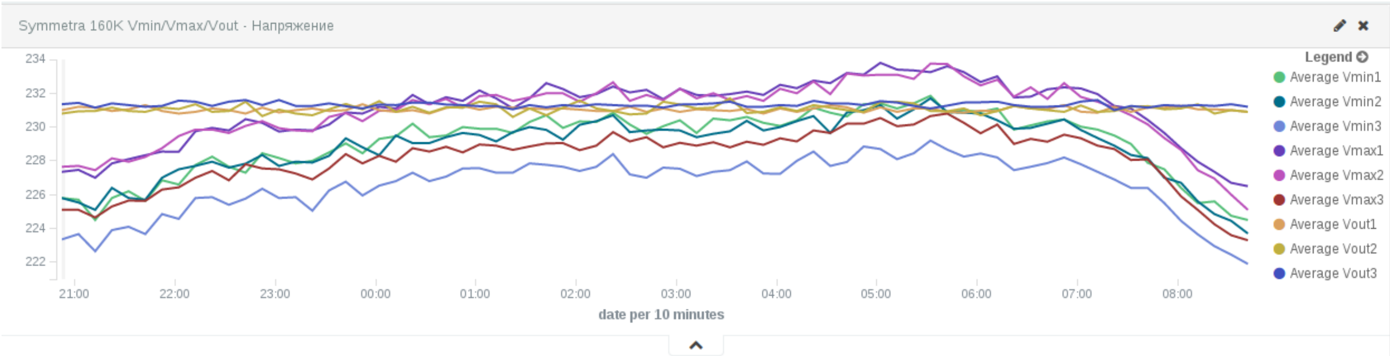
We can detect SYMMETRA battery problems:

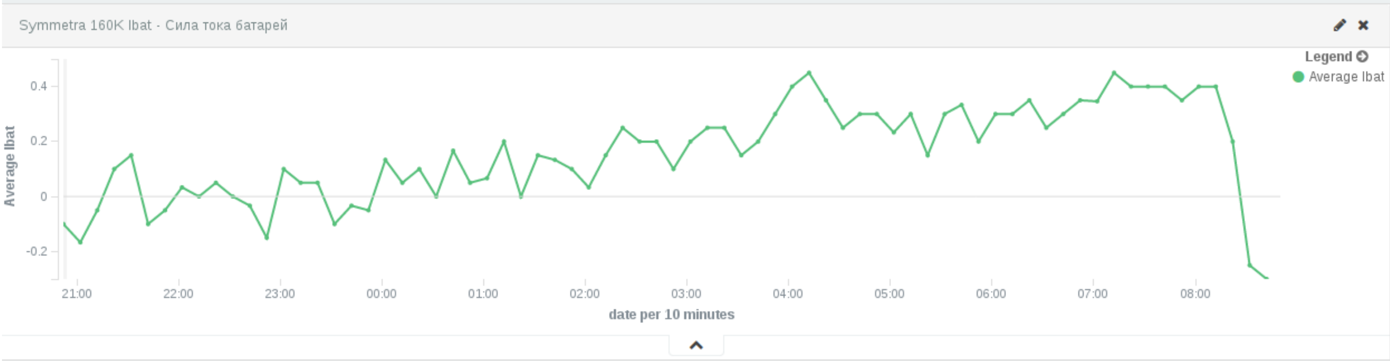
Mar 12 07:15:51 192.168.66.10 UPS: At least one faulty battery exists. 0x0119

Mar 12 07:15:51 192.168.66.10 UPS: A battery fault exists. 0x0207

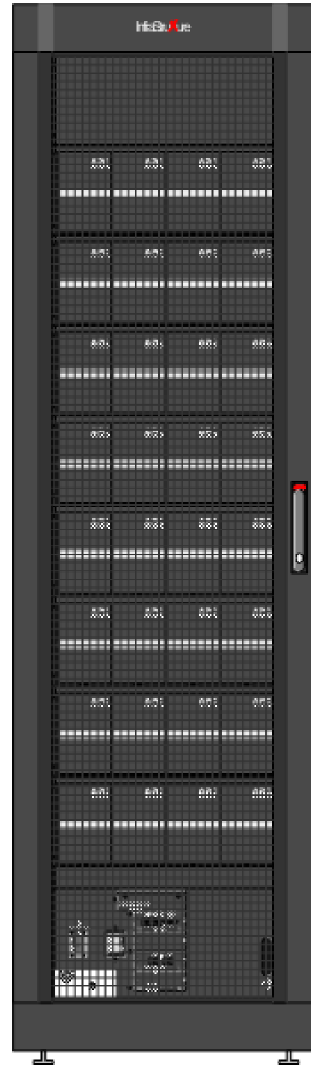
Mar 12 07:15:57 192.168.66.10 UPS: The internal battery temperature exceeds the critical threshold. 0x012C

Mar 12 07:17:50 192.168.66.10 UPS: The internal battery temperature no longer exceeds the critical threshold. 0x012D

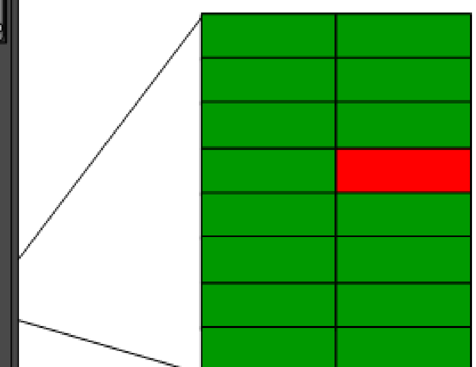




Development: achievements

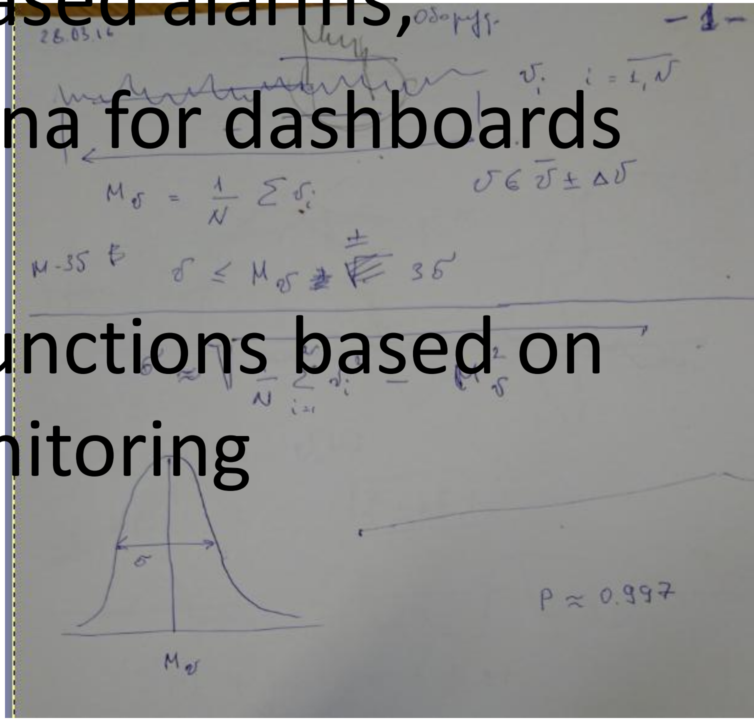


- Ibat >0
- Check temperature for battery blocks and find a bad one
- find\replace a bad battery in the block



Developments: plans

- Add more and more monitoring information to ES: nodes, accounting, storages, antiviruses;
- More and more threshold based alarms;
- Try graphana instead of kibana for dashboards and graphics;
- Start implement analyzing functions based on R – smart alarms, active monitoring



Thank you!

Any questions?