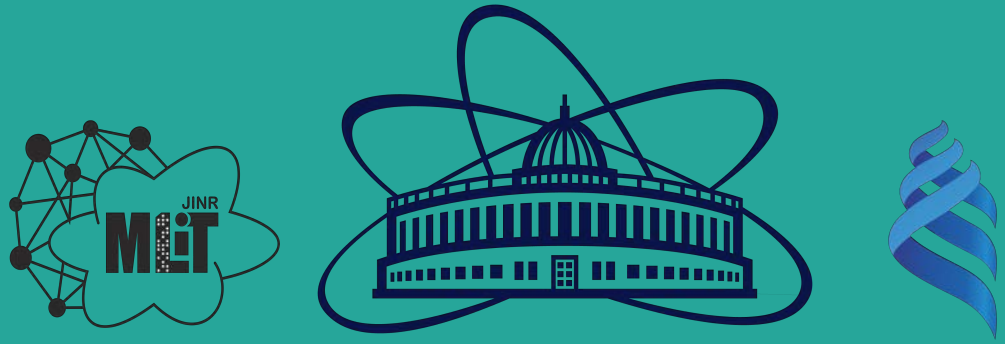


The system for monitoring computing cluster resources

Authors: Gennadii Karpov, Maksim Skazkin, FEFU, Vladivostok;
Maksim Zuev, Dmitry Belyakov,
MLIT, JINR, Dubna.
AYSS, 2024

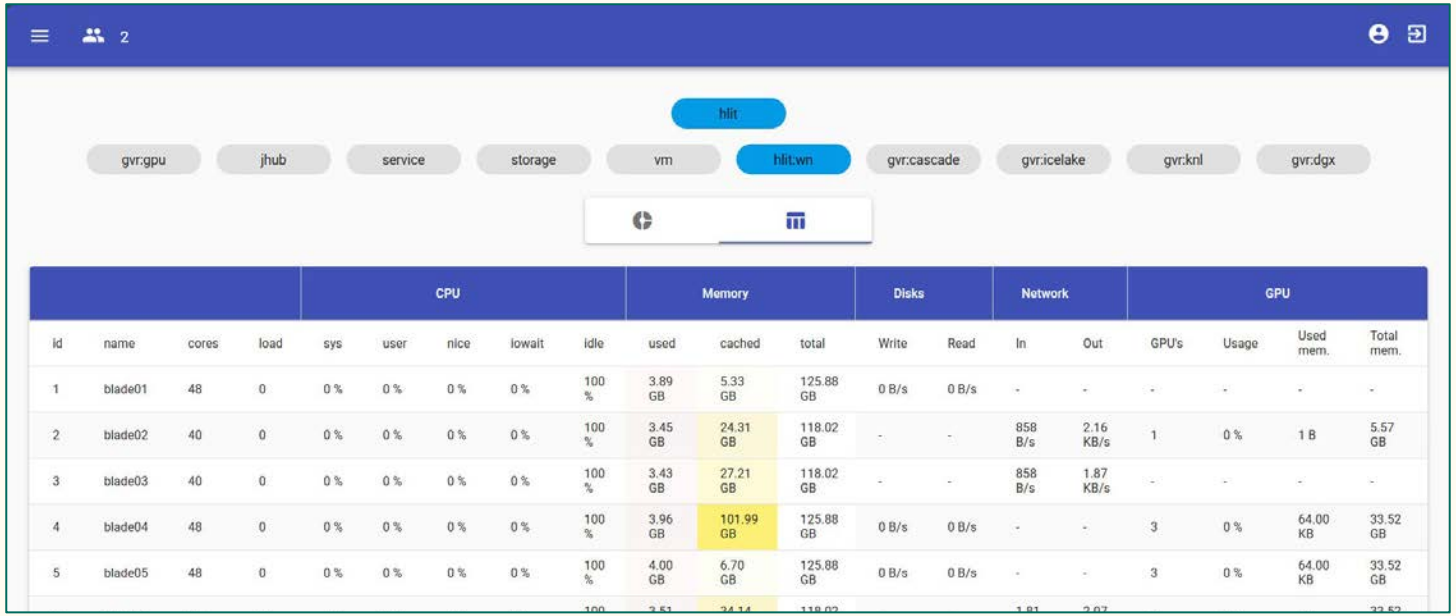
HybriLIT

The heterogeneous platform HybriLIT is part of the Multifunctional Information and Computing Complex (MICC), Laboratory of Information Technologies, JINR, Dubna.

The heterogeneous platform consists of the Govorun supercomputer and the HybriLIT training and test polygon.

Existing solution

«Salsa» monitoring



				CPU					Memory			Disks		Network		GPU			
id	name	cores	load	sys	user	nice	iowait	idle	used	cached	total	Write	Read	In	Out	GPU's	Usage	Used mem.	Total mem.
1	blade01	48	0	0 %	0 %	0 %	0 %	100 %	3.89 GB	5.33 GB	125.88 GB	0 B/s	0 B/s	-	-	-	-	-	-
2	blade02	40	0	0 %	0 %	0 %	0 %	100 %	3.45 GB	24.31 GB	118.02 GB	-	-	858 B/s	2.16 KB/s	1	0 %	1 B	5.57 GB
3	blade03	40	0	0 %	0 %	0 %	0 %	100 %	3.43 GB	27.21 GB	118.02 GB	-	-	858 B/s	1.87 KB/s	-	-	-	-
4	blade04	48	0	0 %	0 %	0 %	0 %	100 %	3.96 GB	101.99 GB	125.88 GB	0 B/s	0 B/s	-	-	3	0 %	64.00 KB	33.52 GB
5	blade05	48	0	0 %	0 %	0 %	0 %	100 %	4.00 GB	6.70 GB	125.88 GB	0 B/s	0 B/s	-	-	3	0 %	64.00 KB	33.52 GB

The purpose of the work and the tasks

The purpose of the work:

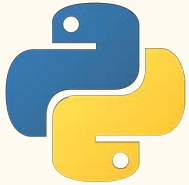
- Develop a monitoring system based on a modern technology stack, with advanced functionality and the possibility of support and modernization.

Tasks for developing of the monitoring system:

- Define the functionality based on requirements of the product;
- Develop the architecture;
- Develop the design of the web application;
- Develop data exchange protocols between clients, sensors and the server;
- Implement authentication and authorization functionality;
- Deploy the product on resources of Heterogeneous platform HybriLIT.

Technology stack. Backend

Language



Asynchronous support

asyncIO

System data

psutil

Web framework

FastAPI

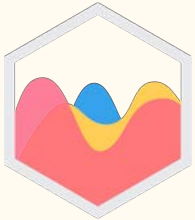
Proxy and server

NGINX

Technology stack. Frontend

Reactive framework

Charts



Data storage

Data streaming

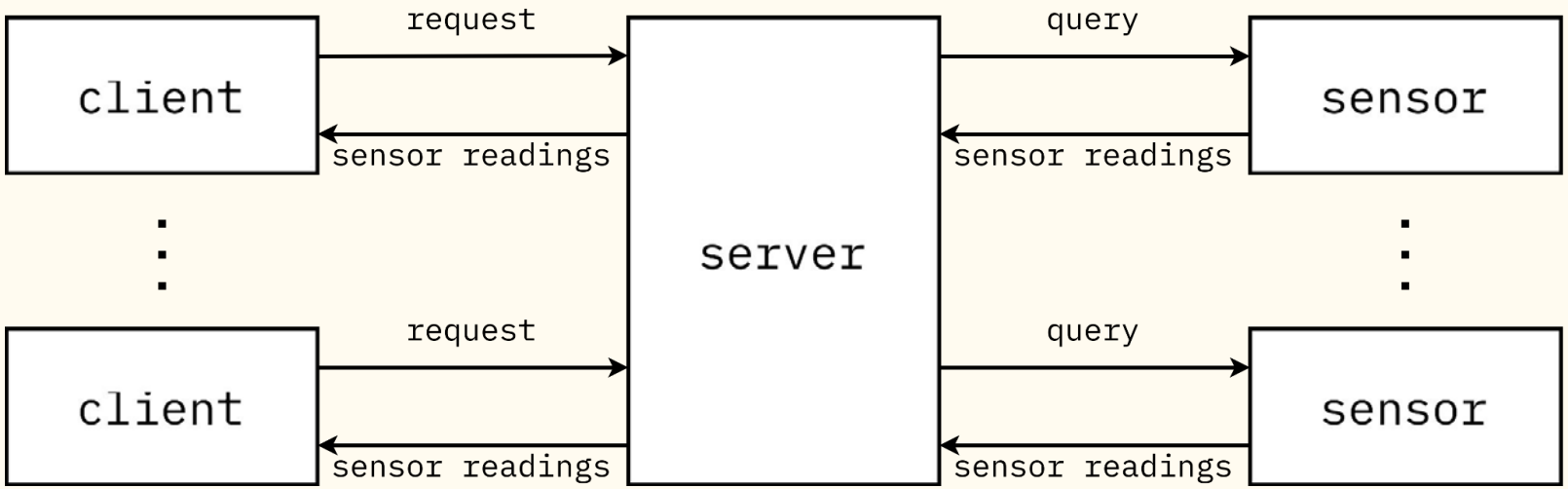
Language

Design

HTTP requests

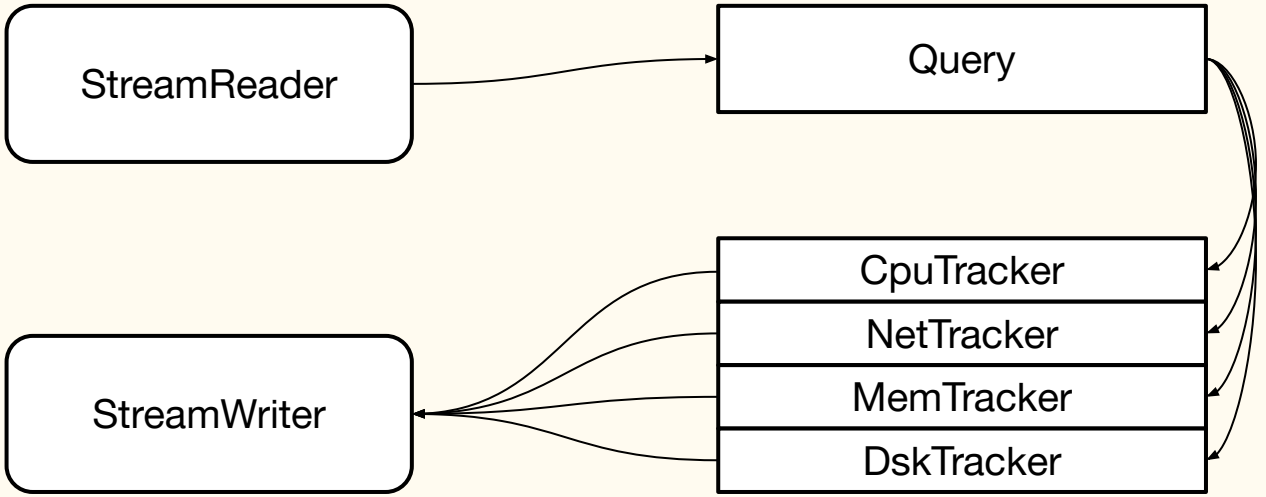
Architecture

overall architecture of the system



Architecture. Sensor

sensor architecture



Architecture. Protocol

server-sensor protocol

server

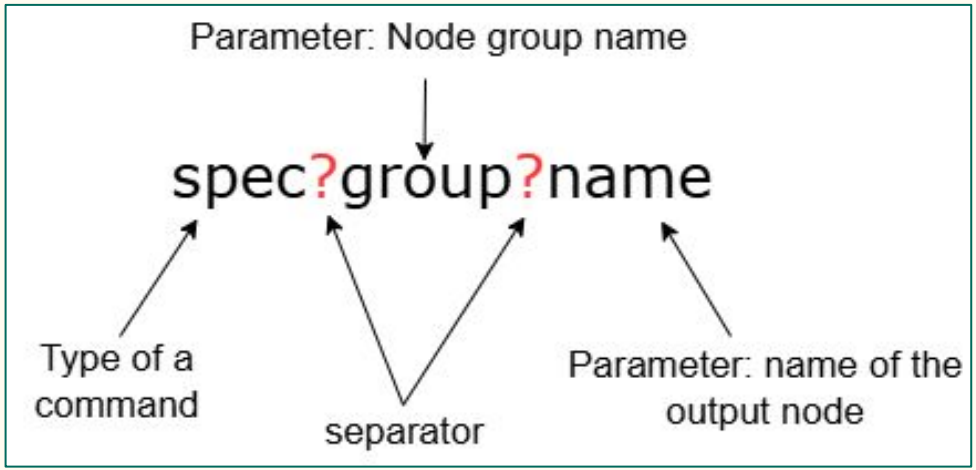
```
{
  “interval”: 1,
  “cpu_fields”: [
    “system”,
    “user”,
    “iowait”,
    “idle”
  ],
  “net_fields”: [
    “recv”,
    “sent”
  ],
  ...
}
```


```
{
  “cpu”: {
    “system”: 2.3,
    “user”: 5.4,
    “iowait”: 10,
    “idle”: 83.3
  },
  “net”: {
    “recv”: 10424,
    “sent”: 239
  }
  ...
}
```

sensor

Architecture. Protocol

client-server protocol

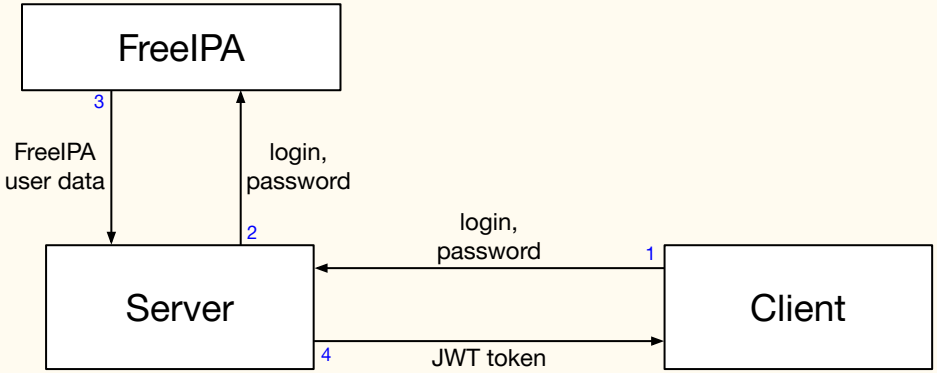


Structure of a request from a client

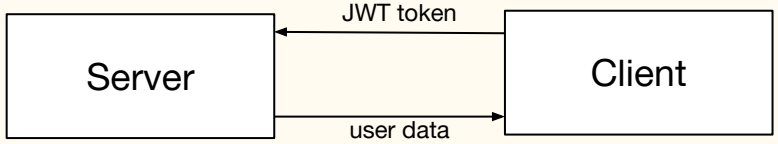
```
{
    "header": "spec!group!name",
    "field1": {
        ...
    },
    "field2": {
        ...
    }
}
```

Example of a JSON response from the server

Architecture. Authorization



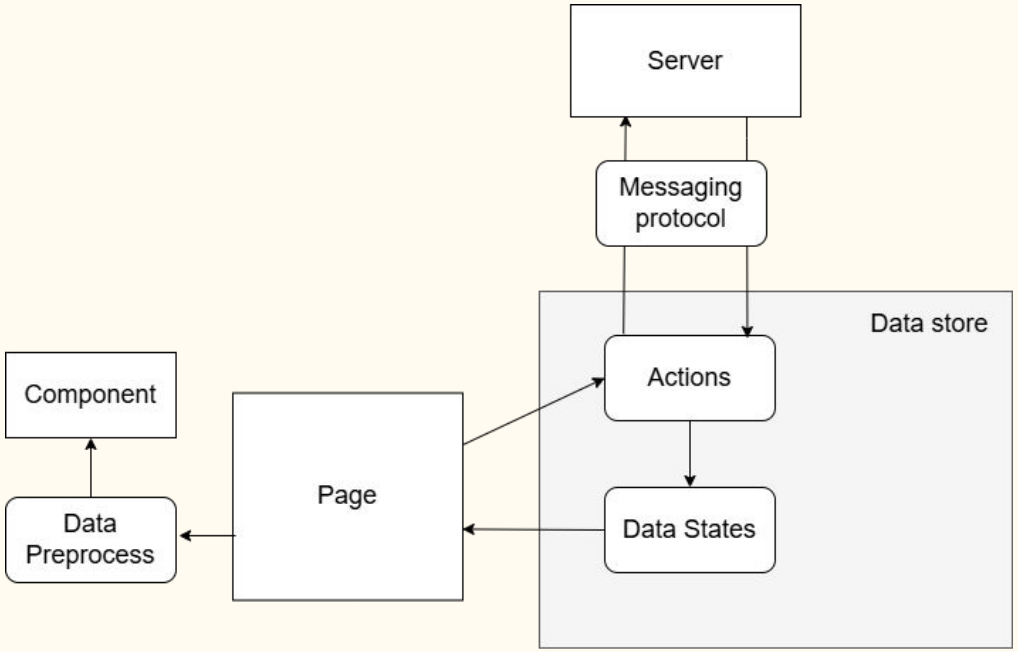
Initial authorization



Reauthorization

Architecture. Frontend

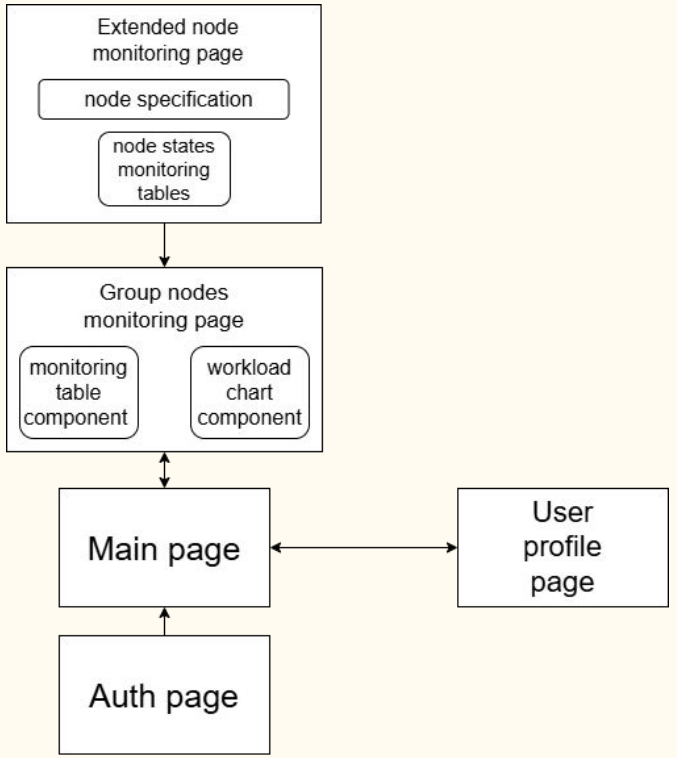
Data visualization processing scheme



Functionality

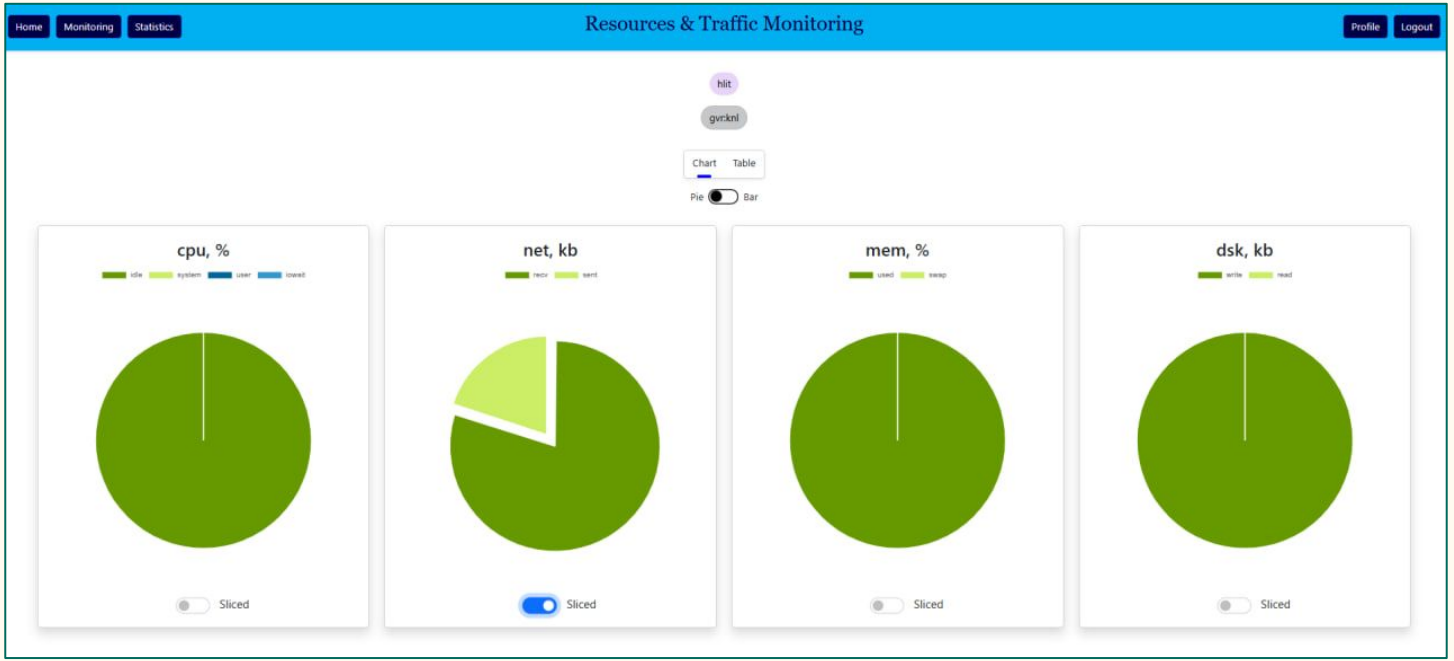
Pages:

- Auth page
- User profile page
- Main page
- Group nodes monitoring page
- Extended node monitoring page



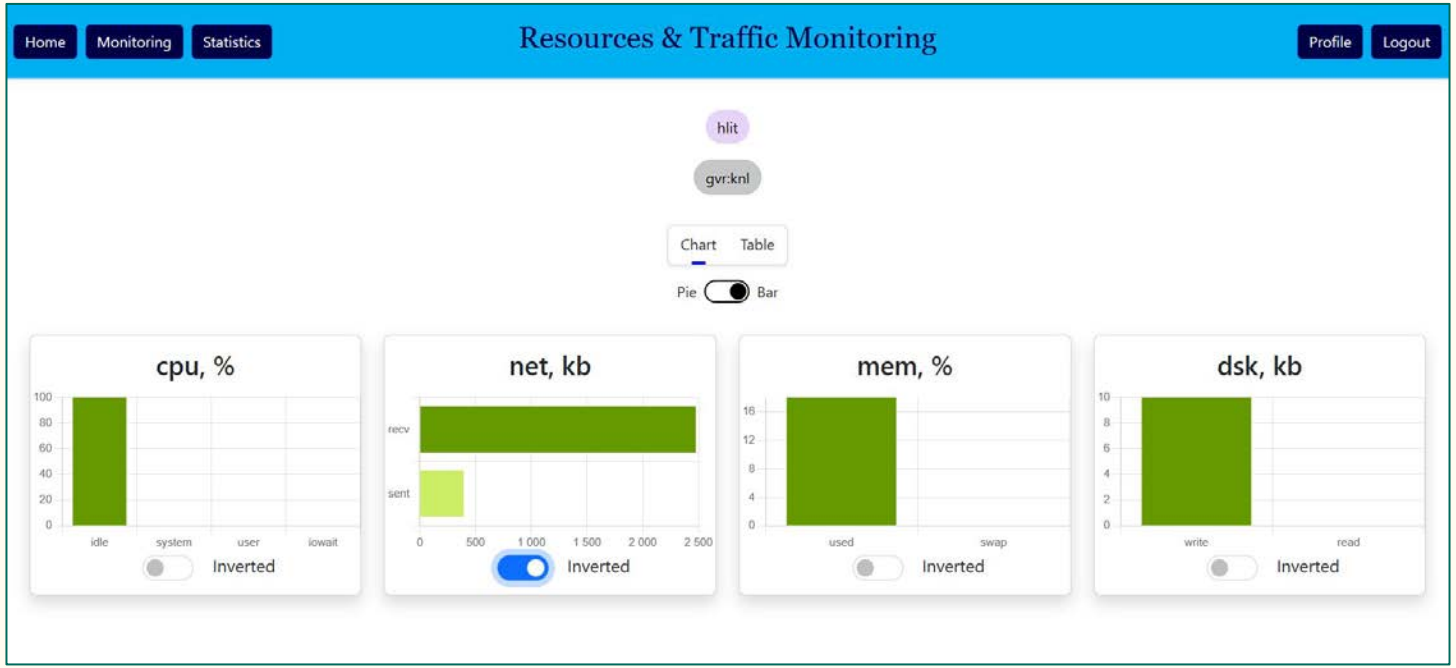
Functionality

Charts page



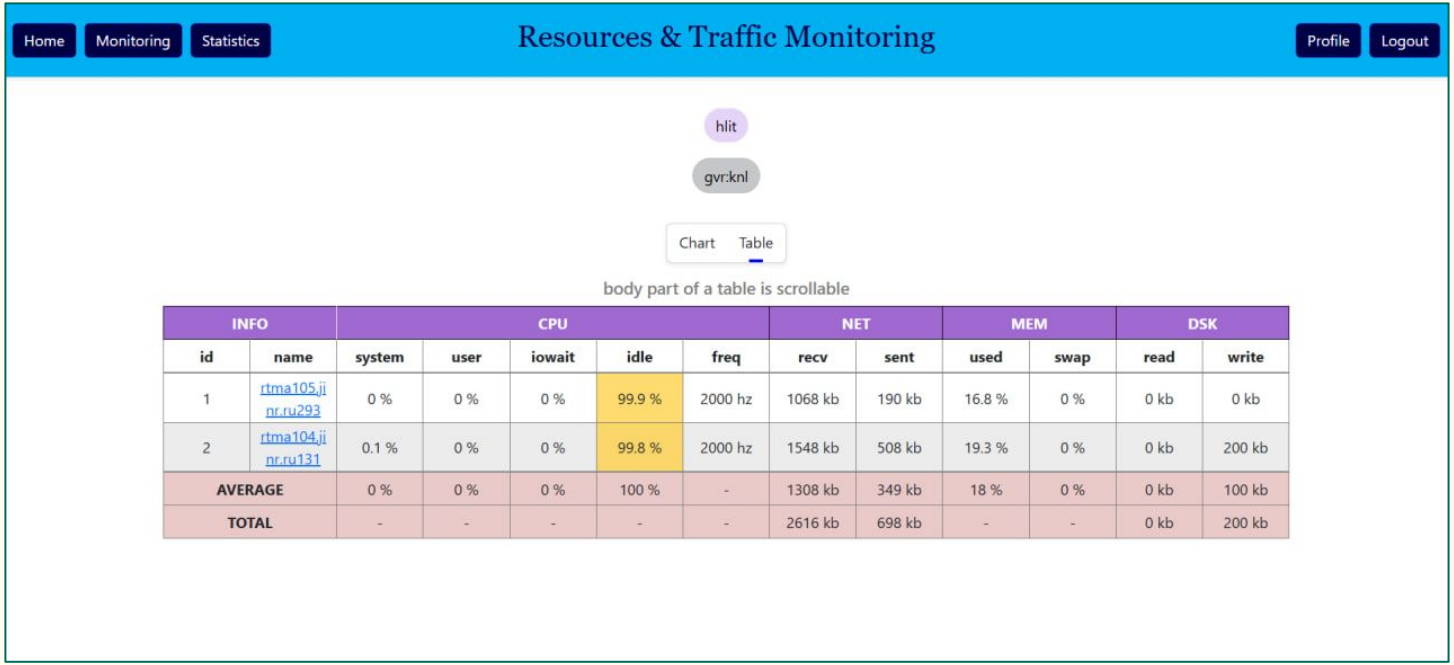
Functionality

Charts page



Functionality

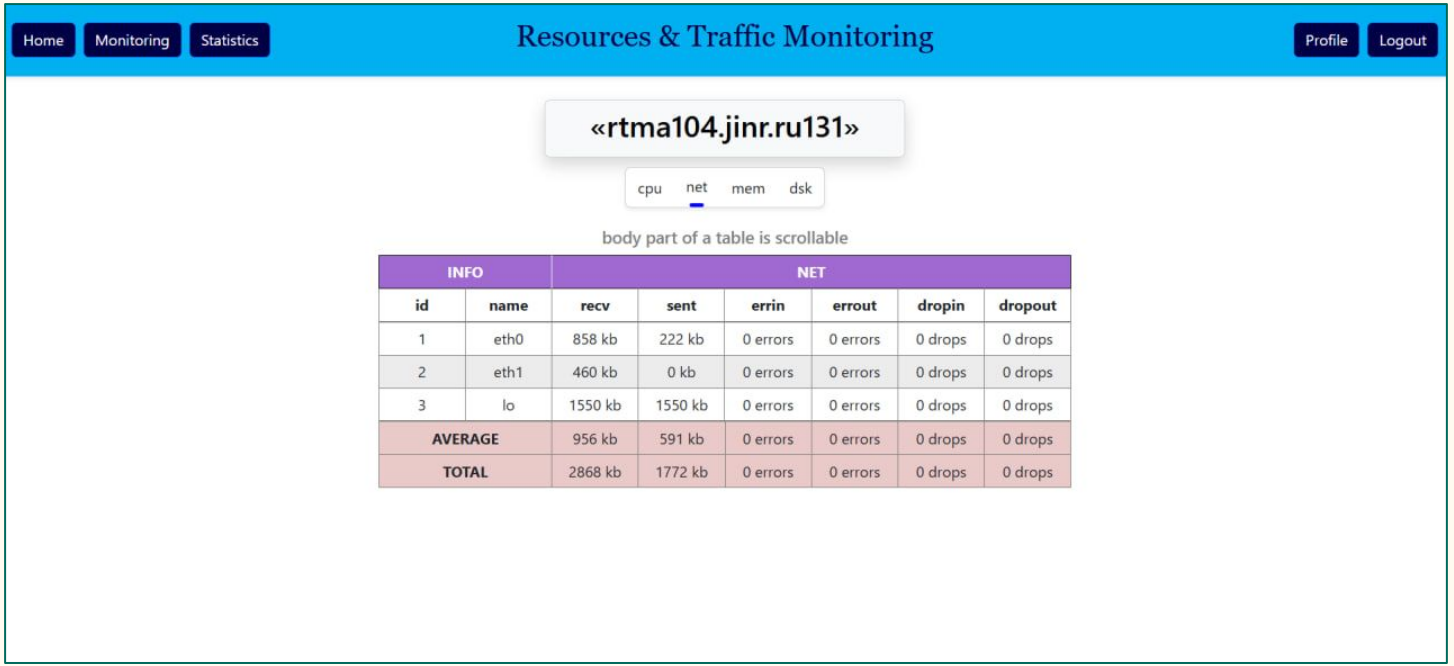
Group nodes monitoring page



INFO		CPU					NET		MEM		DSK	
id	name	system	user	iowait	idle	freq	recv	sent	used	swap	read	write
1	rtma105.jinr.ru293	0 %	0 %	0 %	99.9 %	2000 hz	1068 kb	190 kb	16.8 %	0 %	0 kb	0 kb
2	rtma104.jinr.ru131	0.1 %	0 %	0 %	99.8 %	2000 hz	1548 kb	508 kb	19.3 %	0 %	0 kb	200 kb
AVERAGE		0 %	0 %	0 %	100 %	-	1308 kb	349 kb	18 %	0 %	0 kb	100 kb
TOTAL		-	-	-	-	-	2616 kb	698 kb	-	-	0 kb	200 kb

Functionality

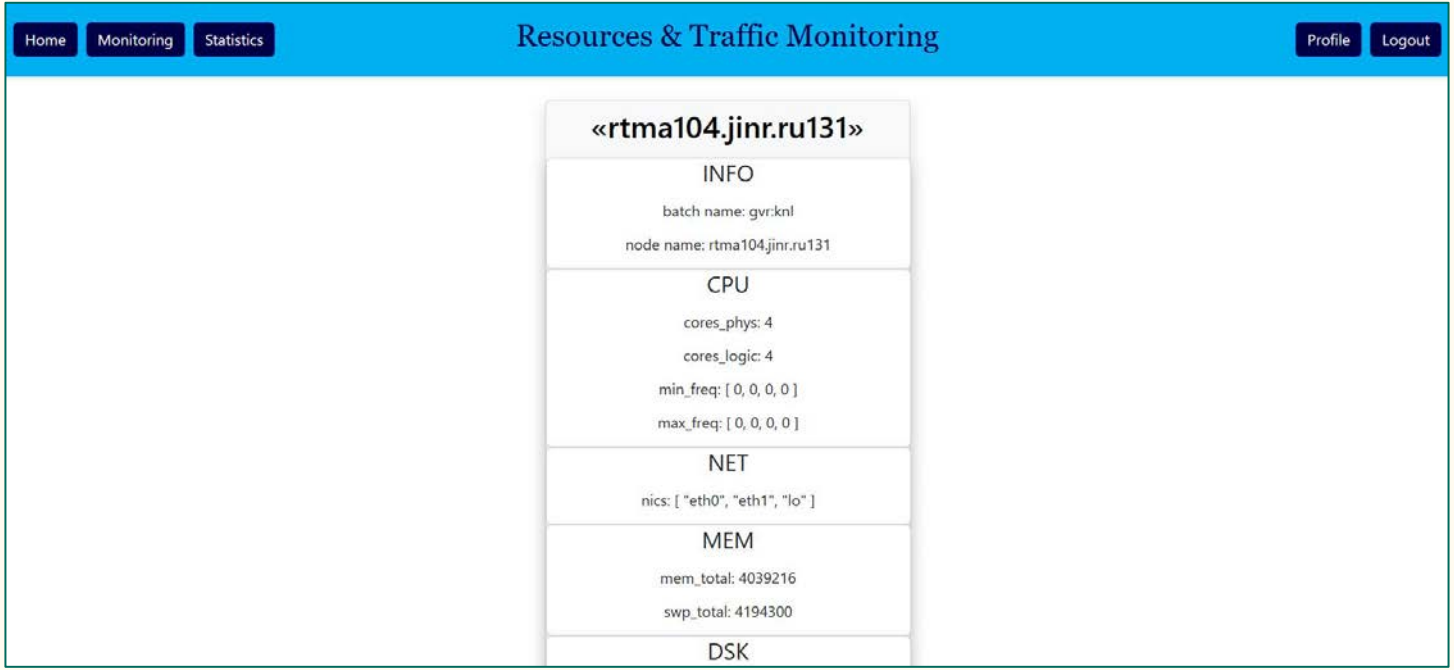
Tables of the state of the components of the computing node



INFO		NET					
id	name	recv	sent	errin	errout	dropin	dropout
1	eth0	858 kb	222 kb	0 errors	0 errors	0 drops	0 drops
2	eth1	460 kb	0 kb	0 errors	0 errors	0 drops	0 drops
3	lo	1550 kb	1550 kb	0 errors	0 errors	0 drops	0 drops
AVERAGE		956 kb	591 kb	0 errors	0 errors	0 drops	0 drops
TOTAL		2868 kb	1772 kb	0 errors	0 errors	0 drops	0 drops

Functionality

Computing node specification component



Conclusion. Current results

- The functionality of the monitoring system was defined;
- The architecture of the sensor, server, and client has been developed;
- The design of the web application was developed;
- Data exchange protocols between sensors, clients and the server have been developed;
- Auth functionality based on FreeIPA was implemented;
- Additional monitoring functions have been developed;
- The system for monitoring was successfully deployed on resources of Heterogeneous platform HybriLIT.

Conclusion

Future plans

- Add user roles that provides a different level of functionality within the web application;
- Add a monitoring of network traffic sources and destinations;
- Add methods to collect system status and usage statistics;
- Prepare a manual for deploying sensors and web application.

Thank you for your attention!

Acknowledgements:
Far Eastern Federal University, Vladivostok;
Meshcheryakov Laboratory of Information Technologies, JINR;
University centre, JINR.