

# Система мониторинга узлов вычислительного кластера NICA для эксперимента MPD/NICA

**Иван Слепов**

ЛФВЭ, ОИЯИ

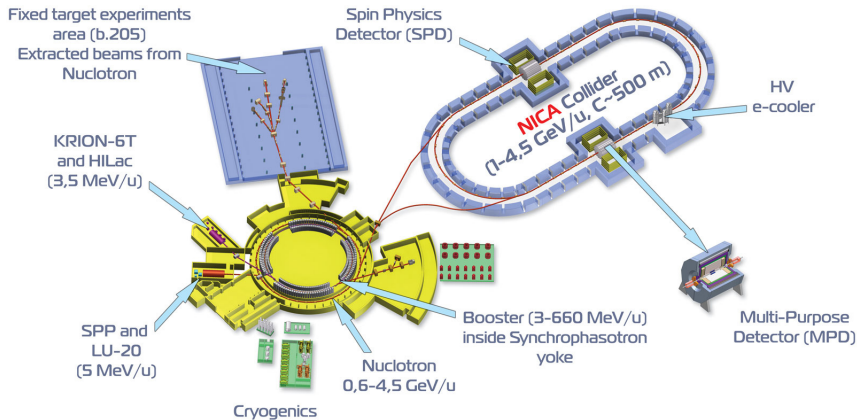
*III школа-конференция молодых ученых и специалистов*

Алушта, 2014

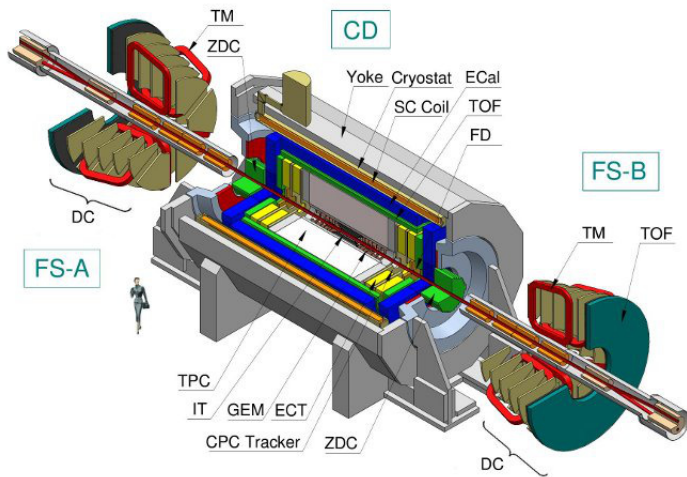


# Ускоритель Нуклотрон + коллайдер NICA

## Superconducting accelerator complex **NICA** (**N**uclotron based **I**on **C**ollider **f**Acility)



# Многофункциональный детектор – MPD для исследования столкновений тяжелых ионов



# Программное обеспечение для MPD - MpdRoot

Компоненты MpdRoot фреймворка:

ROOT + FairRoot (FairBase + FairSoft набор пакетов)

MpdRoot используется для:

1. Моделирования детекторов
2. Реконструкции данных
3. Анализа событий

MpdRoot для разработчиков

доступен на нашем сайте в виде SVN репозитория:

<http://mpd.jinr.ru/trac/browser/mpdroot>

## Компоненты MpdRoot фреймворка:

**ROOT + FairRoot** (FairBase + FairSoft набор пакетов)

## MpdRoot используется для:

1. Моделирования детекторов
2. Реконструкции данных
3. Анализа событий

## MpdRoot для разработчиков

доступен на нашем сайте в виде SVN репозитория:

<http://mpd.jinr.ru/trac/browser/mpdroot>

## Компоненты MpdRoot фреймворка:

**ROOT** + **FairRoot** (FairBase + FairSoft набор пакетов)

## MpdRoot используется для:

1. Моделирования детекторов
2. Реконструкции данных
3. Анализа событий

## MpdRoot для разработчиков

доступен на нашем сайте в виде SVN репозитория:

<http://mpd.jinr.ru/trac/browser/mpdroot>

## Компоненты MpdRoot фреймворка:

**ROOT + FairRoot** (FairBase + FairSoft набор пакетов)

## MpdRoot используется для:

1. Моделирования детекторов
2. Реконструкции данных
3. Анализа событий

## MpdRoot для разработчиков

доступен на нашем сайте в виде SVN репозитория:

<http://mpd.jinr.ru/trac/browser/mpdroot>

## Вычислительный кластер NICA (ЛФВЭ)



**сейчас CPU: 128 XEON ядер | GPU: 1500 TESLA ядер**  
**планируется 10.000 XEON ядер**

## Другие ресурсы

1. Вычислительный кластер в ЛИТ
2. Компьютеры общего пользования
3. Компьютеры разработчиков ПО MpdRoot



## Вычислительный кластер NICA (ЛФВЭ)



сейчас CPU: 128 XEON ядер | GPU: 1500 TESLA ядер  
планируется 10.000 XEON ядер

## Другие ресурсы

1. Вычислительный кластер в ЛИТ
2. Компьютеры общего пользования
3. Компьютеры разработчиков ПО MpdRoot

# Мотивация для разработки системы мониторинга кластера NICA

- Контроль загрузки узлов кластера (cpu, memory, free space, etc.)
- Отслеживание пользовательских задач (batch + interactive)
- Информация об установленных пакетах ПО для MpdRoot
- Информация о работе пакетов кластера (SGE, xrootd, proof, etc.)

## Для чего это нужно?

1. Для выявления ошибок, сбоев и оптимизации работы программ
2. Для контроля работы кластера со стороны как администратора, так и пользователя

## Почему не использовать Nagios или Ganglia?

Необходимо создать алгоритм сбора и анализа данных для дальнейшего развития системы **обработки данных для MPD**

# Мотивация для разработки системы мониторинга кластера NICA

- Контроль загрузки узлов кластера (cpu, memory, free space, etc.)
- Отслеживание пользовательских задач (batch + interactive)
- Информация об установленных пакетах ПО для MpdRoot
- Информация о работе пакетов кластера (SGE, xrootd, proof, etc.)

## Для чего это нужно?

1. Для выявления ошибок, сбоев и оптимизации работы программ
2. Для контроля работы кластера со стороны как администратора, так и пользователя

## Почему не использовать Nagios или Ganglia?

Необходимо создать алгоритм сбора и анализа данных для дальнейшего развития системы **обработки данных для MPD**

# Мотивация для разработки системы мониторинга кластера NICA

- Контроль загрузки узлов кластера (cpu, memory, free space, etc.)
- Отслеживание пользовательских задач (batch + interactive)
- Информация об установленных пакетах ПО для MpdRoot
  - Информация о работе пакетов кластера (SGE, xrootd, proof, etc.)

## Для чего это нужно?

1. Для выявления ошибок, сбоев и оптимизации работы программ
2. Для контроля работы кластера со стороны как администратора, так и пользователя

## Почему не использовать Nagios или Ganglia?

Необходимо создать алгоритм сбора и анализа данных для дальнейшего развития системы **обработки данных для MPD**

# Мотивация для разработки системы мониторинга кластера NICA

- Контроль загрузки узлов кластера (cpu, memory, free space, etc.)
- Отслеживание пользовательских задач (batch + interactive)
- Информация об установленных пакетах ПО для MpdRoot
- Информация о работе пакетов кластера (SGE, xrootd, proof, etc.)

## Для чего это нужно?

1. Для выявления ошибок, сбоев и оптимизации работы программ
2. Для контроля работы кластера со стороны как администратора, так и пользователя

## Почему не использовать Nagios или Ganglia?

Необходимо создать алгоритм сбора и анализа данных для дальнейшего развития системы обработки данных для MPD

# Мотивация для разработки системы мониторинга кластера NICA

- Контроль загрузки узлов кластера (cpu, memory, free space, etc.)
- Отслеживание пользовательских задач (batch + interactive)
- Информация об установленных пакетах ПО для MpdRoot
- Информация о работе пакетов кластера (SGE, xrootd, proof, etc.)

## Для чего это нужно?

1. Для выявления ошибок, сбоев и оптимизации работы программ
2. Для контроля работы кластера со стороны как администратора, так и пользователя

## Почему не использовать Nagios или Ganglia?

Необходимо создать алгоритм сбора и анализа данных для дальнейшего развития системы обработки данных для MPD

# Мотивация для разработки системы мониторинга кластера NICA

- Контроль загрузки узлов кластера (cpu, memory, free space, etc.)
- Отслеживание пользовательских задач (batch + interactive)
- Информация об установленных пакетах ПО для MpdRoot
- Информация о работе пакетов кластера (SGE, xrootd, proof, etc.)

## Для чего это нужно?

1. Для выявления ошибок, сбоев и оптимизации работы программ
2. Для контроля работы кластера со стороны как администратора, так и пользователя

## Почему не использовать Nagios или Ganglia?

Необходимо создать алгоритм сбора и анализа данных для дальнейшего развития **системы обработки данных для MPD**

# Мониторинг, состояние 5 минут

Выборка по некоторым узлам кластера

## Farm | Workers

### Users | Data | MPD Storage

| MACHINE<br>Fairssoft<br>version      | OS<br>Kernel<br>CPU Info   | Nproc | Free<br>Mem<br>MB | Load                 | Activity               | Free<br>Disk<br>GB | Status      | nCPU | Updated                                  |
|--------------------------------------|--|-------|-------------------|----------------------|------------------------|--------------------|-------------|------|--|
| 159.93.63.22<br>nc2.jinr.ru<br>apr13 | Scientific Linux<br>CERN SLC release<br>6.5 (Carbon)<br>2.6.32-<br>358.23.2.el6.x86_64<br>Intel(R) Xeon(R)<br>CPU E5310 @<br>1.60GHz | 272   | 901               | 0.20<br>0.10<br>0.03 | boliana(1)             | 538                | Interactive | 8    | Wed Feb<br>26<br>12:10:01<br>MSK<br>2014 |
| 159.93.63.23<br>nc3.jinr.ru<br>apr13 | Scientific Linux<br>CERN SLC release<br>6.5 (Carbon)<br>2.6.32-<br>358.23.2.el6.x86_64<br>Intel(R) Xeon(R)<br>CPU E5310 @<br>1.60GHz | 270   | 1351              | 0.06<br>0.08<br>0.08 | yordan(1)              | 597                | Interactive | 8    | Wed Feb<br>26<br>12:01:01<br>MSK<br>2014 |
| 159.93.63.28<br>nc8.jinr.ru<br>apr13 | Scientific Linux<br>CERN SLC release<br>6.5 (Carbon)<br>2.6.32-<br>431.3.1.el6.x86_64<br>Intel(R) Xeon(R)<br>CPU E5630 @<br>2.53GHz  | 418   | 27138             | 2.90<br>2.97<br>2.98 | strmaxim(6)            | 584                | Interactive | 16   | Wed Feb<br>26<br>12:14:40<br>MSK<br>2014 |
| 159.93.63.29<br>nc9.jinr.ru<br>apr13 | Scientific Linux<br>CERN SLC release<br>6.4 (Carbon)<br>2.6.32-<br>358.18.1.el6.x86_64<br>Intel(R) Xeon(R)<br>CPU E5630 @<br>2.53GHz | 469   | 27338             | 0.00<br>0.00<br>0.00 | zinchenk(7)<br>sovd(1) | 1311               | Interactive | 16   | Wed Feb<br>26<br>12:05:17<br>MSK<br>2014 |



# Мониторинг, текущее состояние

Процессы пользователя + фильтрация

| PID   | Processes of strmaxim(6) - 159.93.63.28                              | Mem | CPU | PS | Time     |
|-------|--|-----|-----|----|----------|
| 13756 | emacs MpdCellAutomat.cxx   | 0.0 | 0.1 | S  | 00:03:40 |
| 13784 | emacs MpdCellTrack.cxx   | 0.0 | 0.1 | S  | 00:03:37 |
| 24489 | root.exe   | 0.2 | 0.0 | T  | 00:00:05 |
| 27250 | emacs /nica/mpd5/strmaxim/mpdroot-211112/lhetrack/MpdCellAutomat.cxx | 0.1 | 0.1 | S  | 00:15:14 |
| 31612 | emacs runMC.C  | 0.0 | 0.1 | S  | 00:09:52 |
| 31801 | root.exe   | 0.1 | 0.0 | S  | 00:00:02 |

**PID** - process identifier

**Mem** - % of memory use

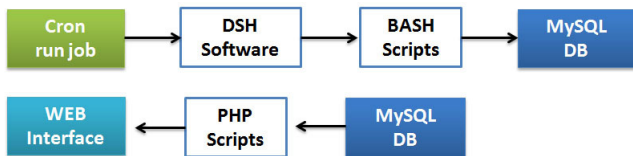
**CPU** - % of CPU use

**PS** - user process status:

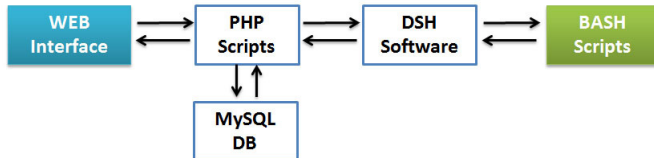
- D - Uninterruptible sleep (usually IO)
- R - Running or runnable (on run queue)
- S - Interruptible sleep (waiting for an event to complete)
- T - Stopped, either by a job control signal or because it is being traced.
- Z - Defunct ("zombie") process, terminated but not reaped by its parent.

# Как это работает?

**Scheme 1** – for collect general information



**Scheme 2** – for collect information about user tasks and provide data management



# Основные идеи реализации системы

Как организовать одновременный запуск скрипта на множестве узлов?

- Собственный конвейер на BASH + фоновые процессы
- Или с помощью пакета DSH или PDSH

Как собрать результат выполнения скрипта со всех узлов?

- Записать результат со всех узлов в одну БД MySQL
- Безопасность открытого порта MySQL контролируется iptables

Как обрабатывать и выводить собранную информацию?

- Веб-интерфейс + обработка вывода клиентом или сервером
- Таблицы: используя jqGrid + jQuery + PHP + MySQL
- Графики: jQuery + Flot + PHP + MySQL или MRTG, RRDtool

# Основные идеи реализации системы

Как организовать одновременный запуск скрипта на множестве узлов?

- Собственный конвеер на BASH + фоновые процессы
- Или с помощью пакета DSH или PDSH

Как собрать результат выполнения скрипта со всех узлов?

- Записать результат со всех узлов в одну БД MySQL
- Безопасность открытого порта MySQL контролируется iptables

Как обрабатывать и выводить собранную информацию?

- Веб-интерфейс + обработка вывода клиентом или сервером
- Таблицы: используя jqGrid + jQuery + PHP + MySQL
- Графики: jQuery + Flot + PHP + MySQL или MRTG, RRDtool

# Основные идеи реализации системы

Как организовать одновременный запуск скрипта на множестве узлов?

- Собственный конвеер на BASH + фоновые процессы
- Или с помощью пакета DSH или PDSH

Как собрать результат выполнения скрипта со всех узлов?

- Записать результат со всех узлов в одну БД MySQL
- Безопасность открытого порта MySQL контролируется iptables

Как обрабатывать и выводить собранную информацию?

- Веб-интерфейс + обработка вывода клиентом или сервером
- Таблицы: используя jqGrid + jQuery + PHP + MySQL
- Графики: jQuery + Flot + PHP + MySQL или MRTG, RRDtool

# Основные идеи реализации системы

Как организовать одновременный запуск скрипта на множестве узлов?

- Собственный конвейер на BASH + фоновые процессы
- Или с помощью пакета DSH или PDSH

Как собрать результат выполнения скрипта со всех узлов?

- Записать результат со всех узлов в одну БД MySQL
- Безопасность открытого порта MySQL контролируется iptables

Как обрабатывать и выводить собранную информацию?

- Веб-интерфейс + обработка вывода клиентом или сервером
- Таблицы: используя jqGrid + jQuery + PHP + MySQL
- Графики: jQuery + Flot + PHP + MySQL или MRTG, RRDtool

# Основные идеи реализации системы

Как организовать одновременный запуск скрипта на множестве узлов?

- Собственный конвеер на BASH + фоновые процессы
- Или с помощью пакета DSH или PDSH

Как собрать результат выполнения скрипта со всех узлов?

- Записать результат со всех узлов в одну БД MySQL
- Безопасность открытого порта MySQL контролируется iptables

Как обрабатывать и выводить собранную информацию?

- Веб-интерфейс + обработка вывода клиентом или сервером
- Таблицы: используя jqGrid + jQuery + PHP + MySQL
- Графики: jQuery + Flot + PHP + MySQL или MRTG, RRDtool

# Основные идеи реализации системы

Как организовать одновременный запуск скрипта на множестве узлов?

- Собственный конвеер на BASH + фоновые процессы
- Или с помощью пакета DSH или PDSH

Как собрать результат выполнения скрипта со всех узлов?

- Записать результат со всех узлов в одну БД MySQL
- Безопасность открытого порта MySQL контролируется iptables

Как обрабатывать и выводить собранную информацию?

- Веб-интерфейс + обработка вывода клиентом или сервером
- Таблицы: используя jqGrid + jQuery + PHP + MySQL
- Графики: jQuery + Flot + PHP + MySQL или MRTG, RRDtool



# Основные идеи реализации системы

Как организовать одновременный запуск скрипта на множестве узлов?

- Собственный конвейер на BASH + фоновые процессы
- Или с помощью пакета DSH или PDSH

Как собрать результат выполнения скрипта со всех узлов?

- Записать результат со всех узлов в одну БД MySQL
- Безопасность открытого порта MySQL контролируется iptables

Как обрабатывать и выводить собранную информацию?

- Веб-интерфейс + обработка вывода клиентом или сервером
- Таблицы: используя jqGrid + jQuery + PHP + MySQL
- Графики: jQuery + Flot + PHP + MySQL или MRTG, RRDtool

# Основные идеи реализации системы

Как организовать одновременный запуск скрипта на множестве узлов?

- Собственный конвейер на BASH + фоновые процессы
- Или с помощью пакета DSH или PDSH

Как собрать результат выполнения скрипта со всех узлов?

- Записать результат со всех узлов в одну БД MySQL
- Безопасность открытого порта MySQL контролируется iptables

Как обрабатывать и выводить собранную информацию?

- Веб-интерфейс + обработка вывода клиентом или сервером
- Таблицы: используя jqGrid + jQuery + PHP + MySQL
- Графики: jQuery + Flot + PHP + MySQL или MRTG, RRDtool

# Динамически обновляемые данные

Сортировка, поиск по параметрам + выделение файлов и передача их значений

| Data simulation DB for MPD - EDITABLE |      |           |        |         |  |        |
|---------------------------------------|------|-----------|--------|---------|--|--------|
| Type                                  | Gen  | Collision | Energy | Trigger | Path   | Descr. |
|                                       | qgs  |           | 7      |         |  |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb                   |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_100.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_101.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_103.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_106.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_109.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_10.r12  |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_113.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_117.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_11.r12  |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_120.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_121.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_125.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_128.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_129.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_130.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_132.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_133.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_134.r12 |        |
| MC                                    | QGSM | AuAu      | 7      | mb      | /nica/data4mpd1/QGSM/AuAuss7mb/AuAuss7mb_135.r12 |        |

+ Новый | Изменить | Удалить | Поиск | Обновить | Стр. 1 из 4 | Просмотр 1 - 20 из 65

# Результаты и заключение

На данном этапе мониторинг позволяет

- Контролировать загрузку узлов кластера (cpu, memory, free space, etc.)
- Отслеживать пользовательские задачи (batch + interactive)
- Получать информацию об установленных пакетах ПО MpdRoot
- Получать иную информацию о кластере и записывать её в БД

## Заключение

1. Некоторыми компонентами системы уже активно пользуются
2. Ведется дальнейшее развитие и доработка системы
3. Весь код переписывается на python, из-за проблем bash+mysql

# Результаты и заключение

На данном этапе мониторинг позволяет

- Контролировать загрузку узлов кластера (cpu, memory, free space, etc.)
- Отслеживать пользовательские задачи (batch + interactive)
- Получать информацию об установленных пакетах ПО MpdRoot
- Получать иную информацию о кластере и записывать её в БД

## Заключение

1. Некоторыми компонентами системы уже активно пользуются
2. Ведется дальнейшее развитие и доработка системы
3. Весь код переписывается на python, из-за проблем bash+mysql

# Результаты и заключение

На данном этапе мониторинг позволяет

- Контролировать загрузку узлов кластера (cpu, memory, free space, etc.)
- Отслеживать пользовательские задачи (batch + interactive)
- Получать информацию об установленных пакетах ПО MpdRoot
- Получать иную информацию о кластере и записывать её в БД

## Заключение

1. Некоторыми компонентами системы уже активно пользуются
2. Ведется дальнейшее развитие и доработка системы
3. Весь код переписывается на python, из-за проблем bash+mysql

# Результаты и заключение

На данном этапе мониторинг позволяет

- Контролировать загрузку узлов кластера (cpu, memory, free space, etc.)
- Отслеживать пользовательские задачи (batch + interactive)
- Получать информацию об установленных пакетах ПО MpdRoot
- Получать иную информацию о кластере и записывать её в БД

## Заключение

1. Некоторыми компонентами системы уже активно пользуются
2. Ведется дальнейшее развитие и доработка системы
3. Весь код переписывается на python, из-за проблем bash+mysql

# Результаты и заключение

На данном этапе мониторинг позволяет

- Контролировать загрузку узлов кластера (cpu, memory, free space, etc.)
- Отслеживать пользовательские задачи (batch + interactive)
- Получать информацию об установленных пакетах ПО MpdRoot
- Получать иную информацию о кластере и записывать её в БД

## Заключение

1. Некоторыми компонентами системы уже активно пользуются
2. Ведется дальнейшее развитие и доработка системы
3. Весь код переписывается на python, из-за проблем bash+mysql



# Результаты и заключение

На данном этапе мониторинг позволяет

- Контролировать загрузку узлов кластера (cpu, memory, free space, etc.)
- Отслеживать пользовательские задачи (batch + interactive)
- Получать информацию об установленных пакетах ПО MpdRoot
- Получать иную информацию о кластере и записывать её в БД

## Заключение

1. Некоторыми компонентами системы уже активно пользуются
2. Ведется дальнейшее развитие и доработка системы
3. Весь код переписывается на python, из-за проблем bash+mysql

# Результаты и заключение

На данном этапе мониторинг позволяет

- Контролировать загрузку узлов кластера (cpu, memory, free space, etc.)
- Отслеживать пользовательские задачи (batch + interactive)
- Получать информацию об установленных пакетах ПО MpdRoot
- Получать иную информацию о кластере и записывать её в БД

## Заключение

1. Некоторыми компонентами системы уже активно пользуются
2. Ведется дальнейшее развитие и доработка системы
3. Весь код переписывается на python, из-за проблем bash+mysql

The screenshot shows the MpdRoot website interface. At the top, there is a search bar and language options (rus, eng) and a login link. Below the header is a navigation menu with categories: General, Documents, Computing, References, and Forum. The 'Computing' menu is expanded, showing options like MPDRoot, BMNRoot, NICA cluster, Monitoring system, MPO databases, and HowTo. A sub-menu for 'NICA cluster' is also visible, listing 'Structure', 'How to work', 'Batch processing', and 'PROOF parallelization'. The main content area features a 'News' section with a link to 'New location of Data4mpd'. Below the menu, there is a diagram of the NICA cluster structure, showing 'Interactive.aad' servers (nc2-nc9) and 'Batch servers' (nc10-nc15) connected via 'Ethernet 1 Gb/s'. A 'Batch.aad' server is also shown in 'building 215'. The diagram includes detailed specifications for each server type.

**Interactive.aad:**

- nc2: 8 cores, 16 GB RAM
- nc3: 8 cores, 16 GB RAM
- nc4: 16 cores, 32 GB RAM
- nc5: 16 cores, 32 GB RAM
- total: 48 cores, 96 GB RAM

**Batch servers:**

- nc10: 24 logic cores, 32 GB RAM (30 CPU Scheduler, 10 CPU PROOF)
- nc11: 24 logic cores, 32 GB RAM (30 CPU Scheduler, 10 CPU PROOF)
- nc12: 32 logic cores, 32 GB RAM (14 CPU Scheduler, 14 CPU PROOF)
- nc13: 32 logic cores, 64 GB RAM (14 CPU Scheduler, 14 CPU PROOF)
- nc14: 32 logic cores, 64 GB RAM (14 CPU Scheduler, 14 CPU PROOF)
- nc15: 144 cores, 224 GB RAM

**Batch.aad:**

- total: 144 cores, 224 GB RAM

Figure 1. The current structure of the NICA cluster

If you know how to work with Sun Grid Engine system (user guide) you can use `osub` command on LHEP farm



Спасибо за внимание!