**Описание программного обеспечения для организации распределенной системы хранения данных «МГТУ-РСХД»**

**.**

1. **Назначение и описание системы**

Распределенная система хранения данных (РСХД) предназначена для организации доступа к данным, сохраняемым на дисках системы, по файловым и объектным протоколам.

РСХД относится к классу горизонтально-масштабируемых систем (scale-out), характеризуемых наличием большого числа серверов, объединенных в кластер, на которых распределяется блочный и файловый доступ для клиента.

Кластер РСХД, состоит из узлов стандартной архитектуры двух типов:

1. модуль управления, включающий в себя процессоры, кэш-память, в том числе и энергонезависимая, Ethernet-порты средств подключения. Модули управления РСХД образуют отказоустойчивый кластер, резервируя друг друга попарно или по четыре, в зависимости от настройки конфигурации системы. Их задача подготавливать данные для записи и организовывать чтение. Все поступающие от клиентов данные записываются в кэш-память контроллеров, и клиенты немедленно получают ответ об успешной записи. В зависимости от требований к надежности хранения данных, блоки данных могут записываться двумя и более копиями на различные модули хранения. Количество копий задается конфигурацией системы.

Модуль управления предоставляет клиентам доступ к хранимой в системе информации по протоколам доступа: S3, NFS, iSCSI.

1. модуль хранения, включающий в себя систему дисков и управляющий ими процессор. Модули хранения РСХД - образуют рой, асинхронно и параллельно взаимодействующий с контроллерами. Контроллер подготавливает данные для записи в кэш-память, а модули хранения в свою очередь параллельно забирают данные для записи. За счет массового параллелизма обеспечивается пропускная способность системы.

Соединение узлов РСХД осуществляется через внутреннюю высокоскоростную коммуникационную шину.

Помимо этого, модули управления соединены попарно прямым сетевым соединением (интерконнектом). Добавление модулей управления к кластеру РСХД также возможно только парами.

На масштабирование по модулям хранения никаких ограничений не накладывается, к контроллерной паре могут быть подключены десятки и даже сотни модулей хранения.

Информационный поток данных между РСХД и клиентскими рабочими станциями равномерно распределяется между модулями управления кластера, в которых выполняется прямое и обратное преобразование потока данных к совокупности блоков данных, хранимых на жестких дисках, и дальнейший информационный обмен с устройствами хранения данных.

Входной поток данных в РСХД распределяется по конечным устройствам хранения данных с обеспечением равномерной нагрузки на все доступные конечные устройства хранения данных в кластере.

Взаимодействие компонентов кластера спроектировано для обеспечения максимальной производительности процессов и требуемой надежности хранения.

1. **Принцип работы РСХД**

Система состоит из следующих основных блоков:

1. Модуль управления
2. Модуль хранения.

## **Модуль управления**

Пара модулей управления РСХД образуют отказоустойчивый кластер, резервируя друг друга попарно. Их задача подготавливать данные для записи и организовывать чтение. Все поступающие от клиентов данные, записываются в кэш память контроллеров и клиенты немедленно получают ответ об успешной записи.

В свою очередь, модуль управления (контроллер) состоит из следующих структурных компонентов:

**API модуля управления.**

Перечень команд управления ресурсами системы хранения данных (API-функции) является неотъемлемой частью данного блока “API модуля управления”.

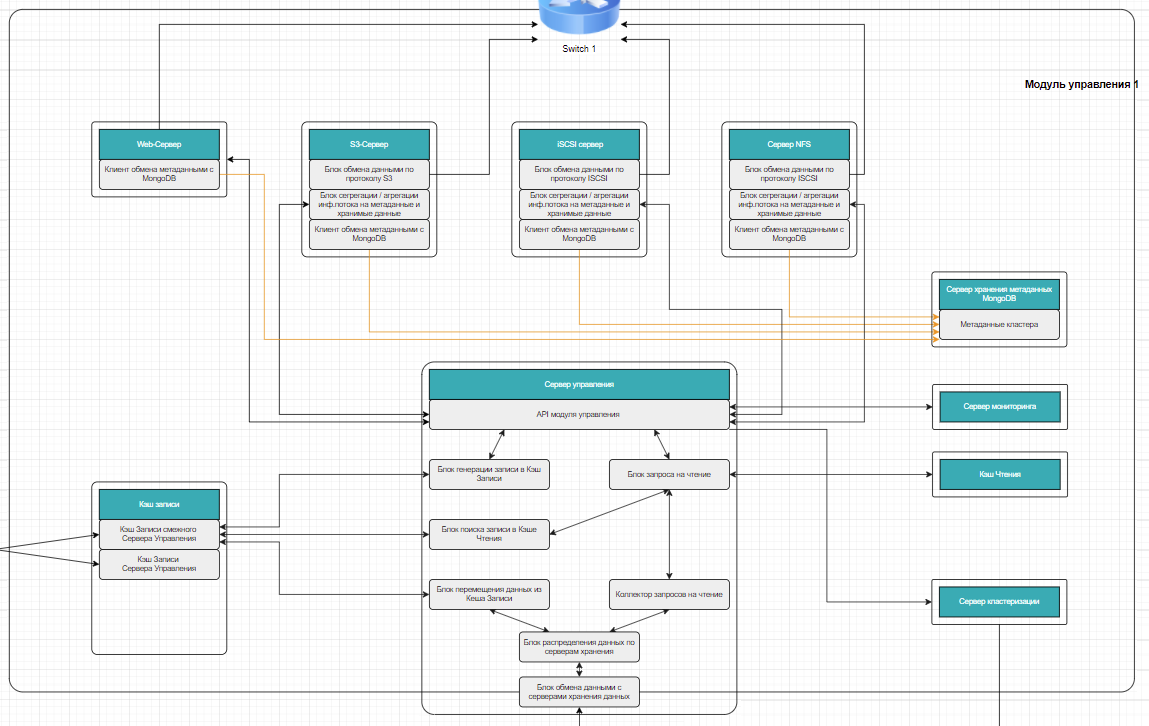


Рисунок . Структура модуля управления

Соответственно, согласно схеме приведенной на Рисунок 2, обработка любого обращения к модулю управления осуществляется в этом блоке.

Через API модуля управления осуществляется взаимодействие модуля управления со всеми серверами протокола доступа, сервером мониторинга, сервером кластеризации, взаимодействие с АРМ Администратора.

**Блок генерации записи в Кэш Записи**

Создание регистрационной записи и последующее копирование “сырого” пользовательского буфера переменного размера в Кэш Записи модуля управления.

**Блок запроса на чтение**

Запрос на чтение состоит из идентификационного номера объекта и интервала данных, кратного размеру блока хранимых данных.

Осуществляет поиск данных в Кэше Чтения, соответствующих требованиям запроса.

Формирует запрос на поиск требуемых данных в Кэше Записи. Если вернулись все данные запроса, то формируется ответ на поступивший запрос. Если остались в наличии отсутствующие блоки, то формируется запрос на поиск отсутствующих данных в модулях хранения.

При поступлении всех данных формируется ответ по полученному запросу.

**Блок поиска записи в Кэше Записи**

Осуществляет поиск данных, отвечающих запросу на чтение. В случае удачного завершения поиска, найденные по запросу данные копирует из Кэша Записи в Кэш Чтения.

**Блок перемещения данных из Кэша Записи**

Срабатывает по одному из триггеров: “область Кэша Записи заполнена”, таймаут операции.

Вызывает блокировку области, которая должна подвергнуться процедуре перемещения данных и запускает алгоритм перемещения данных, записанных в области Кэша Записи, на конечные устройства хранения.

**Коллектор запросов на чтение**

В данном блоке осуществляется накопление запросов на чтение перед дальнейшей их обработкой.

**Блок распределения данных по серверам хранения**

Проводит однозначное сопоставление идентификатора объекта и интервала данных кратных размеру блока хранимых данных с идентификатором конечного хранилища (жесткого диска).

Также используется для получения идентификатора конечного хранилища искомых блоков данных объекта в процессе чтения объекта из РСХД.

Поддерживает операцию “зеркалирования” для хранения резервной копии блоков хранимых данных.

**Блок обмена данными с серверами хранения данных**

Осуществляет обмен данными между модулем управления и модулями хранения по внутреннему протоколу обмена данными module-protocol.

**Кэш Записи**

Представляет собой область памяти в энергозащищенной оперативной памяти модуля управления.

В свою очередь, сам Кэш Записи разбивается на ряд независимых областей.

Область I. Область Кэша Записи, отведенная для текущей оперативной деятельности модуля управления.

Область II. Область Кэша Записи, отведенная для репликации текущей оперативной деятельности смежного (или парного) модуля управления.

По достижении определенной величины заполненности текущей области или по достижении величины таймаута, текущая область блокируется на запись и вызывается процедура роспуска записей, накопленных в этой области, по модулям хранения. По окончании процедуры роспуска текущая область разблокируется для приема записей.

При роспуске области осуществляется формовка содержимого записей к совокупности блоков хранимых данных, см. Рисунок 3.

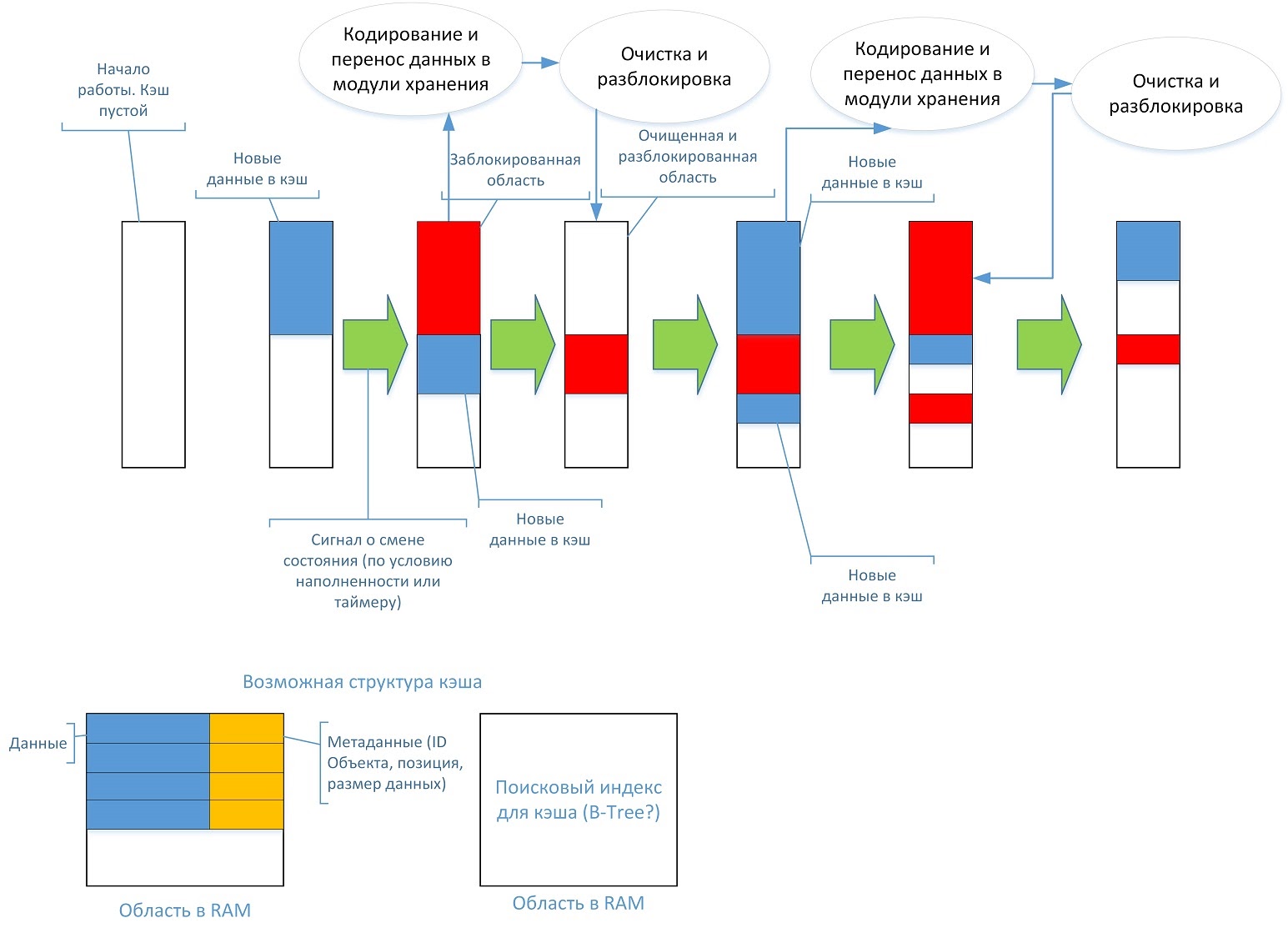


Рисунок . Работа кэша записи

**Кэш Чтения**

Кэш Чтения является буфером в области оперативной памяти модуля управления, аккумулирующей в себе поступающие блоки данных на операции чтения.

Данные могут попасть в Кэш Чтения в результате 2-х операций:

а) требуемые блоки данных обнаружены в Кэше Записи и регистрируются с копированием в Кэш Чтения

б) требуемые блоки данных поступили из модулей хранения и регистрируются в Кэше Чтения.

Взаимодействие с Кэшем Записи осуществляется через “Блок запроса на чтение” модуля управления.

**Web-сервер**

Через Web-сервер осуществляется конфигурирование кластера системы хранения данных.

Web-сервер предоставляет графический интерфейс пользователя, посредством которого формируются команды управления установленного вида (текущее рабочее обозначение - API-функции).

Поступившая команда управления передается в rpc-сервер, через который осуществляется информационный обмен между всеми узлами и подсистемами кластера СХД.

Также в rpc-сервере отображается результат исполнения команды управления либо возвращается сообщение с кодом ошибки.

**S3-сервер**

Осуществляет информационный обмен между клиентской рабочей станцией и кластером схд по протоколу обмена данными S3.

Программное обеспечение состоит из клиентской части, устанавливаемой на рабочей станции клиента и серверной части, являющейся неотъемлемой частью программного обеспечения модуля хранения СХД.

В задачи S3-сервера входит:

для процесса записи:

* прием пакета данных от клиентской рабочей станции
* осуществление разделения принимаемого потока данных на собственно данные и метаданные
* регистрация метаданных в сервере хранения метаданных MongoDB
* передача собственно данных в сервер управления с использованием средств блока “API модуля управления”

для процесса чтения:

* формирование запроса на сервер хранения метаданных MongoDB на получение метаданных, соответствующих запрошенному объекту
* передача запроса, содержащего идентификационный номер объекта и интервала данных, в сервер управления с использованием средств блока “API модуля управления”
* прием от сервера управления поступивших по запросу данных
* формирование и передача клиентской рабочей станции пакета данных, содержащем в себе как поступившие данные от сервера управления так и связанные с ними метаданные, поступившие от сервера хранения метаданных MongoDB.

**Сервер хранения метаданных MongoDB**

Сервер хранения метаданных MongoDB является документоориентированной системой управления базами данных (СУБД) с открытым исходным кодом, не требующей описания схемы таблиц. Классифицируется как NoSQL, использует JSON-подобные документы и схему базы данных.

Масштабируется горизонтально, используя технику сегментирования (англ. sharding) объектов баз данных — распределение их частей по различным узлам кластера.

MongoDB является сторонней разработкой.

Используется для хранения метаданных хранимых объектов и параметров конфигурации СХД.

## **Модуль хранения**

Модули хранения РСХД - образуют рой, асинхронно и параллельно взаимодействующий с контроллерами. Контроллер подготавливает данные для записи в кэш памяти, а модули хранения в свою очередь параллельно забирают данные для записи. За счет массового параллелизма обеспечивается высокая пропускная способность системы.

Структурная схема модуля хранения показана на Рисунок 4.

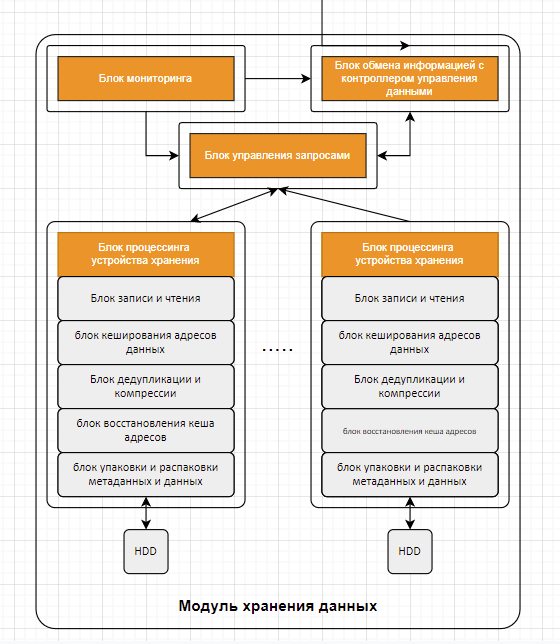


Рисунок . Структура модуля хранения

**Блок мониторинга**

Является неотъемлемой частью подсистемы мониторинга и диагностики СХД.

В задачи блока входит сбор данных о состоянии оборудования модуля хранения по протоколу snmp и взаимодействие с сервером мониторинга кластера посредством обмена запросами через rpc-сервер.

**Блок обмена информацией с контроллером управления данными**

Осуществляет информационный обмен с модулем управления по специально разработанному внутреннему протоколу.

**Блок управления запросами**

Осуществляет передачу поступивших запросов по блокам процессинга жестких дисков.

**Блок записи и чтения**

Осуществляет информационный обмен с жестким диском

**Блок кэширования адресов данных**

Регистрирует в кэше адресов жесткого диска информацию о блоке хранимых данных и физическом адресе на жестком диске места хранения блока данных. Кэшом является контейнер, предоставляющий быстрый поиск адреса по мета информации.

**Блок дедупликации и компрессии**

Дедупликация данных - это технология обнаружения и последующего исключения избыточных данных в дисковом хранилище. В программном обеспечении РСХД механизм дедупликации построен на использовании хеш таблицы, которая содержит уникальные (по хешу) пары: хеш блока и адрес блока в устройстве хранения. Она хранится в мета-данных каждого модуля хранения с таблицей статистики дедупликации. Для достижения максимальной эффективности таблица дедупликации располагается в оперативной памяти, что предъявляет требования к количеству оперативной памяти.

Это связано с тем, что каждый дедуплицированный блок данных файловой системы может иметь запись в таблице дедупликации. Таблица дедупликации растет линейно в зависимости от количества уникальных блоков в пуле. Поиск значения в таблице дедупликации выполняется по значению SHA256-хеша. Иллюстрация принципа работы механизма дедупликации показана на Рисунок 5.

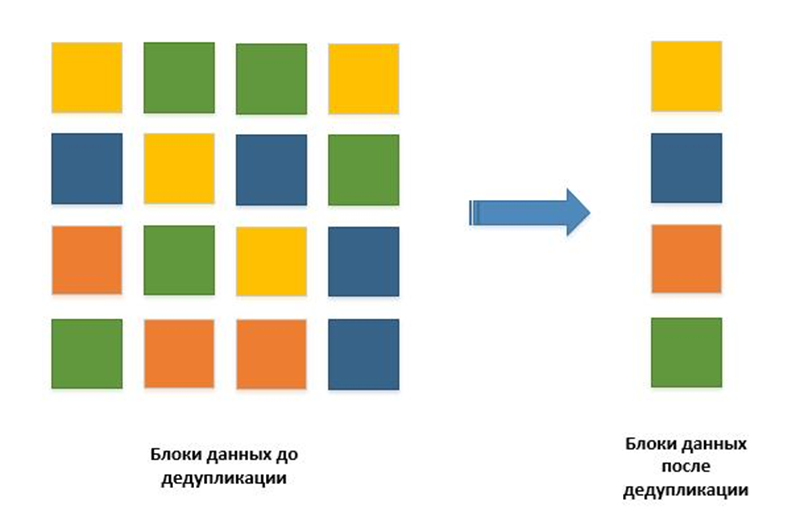


Рисунок . Принцип работы алгоритма дедупликации данных

**Блок восстановления кэша адресов**

При запуске модуля хранения осуществляется посекторный обход всего доступного пространства диска. Осуществляется считывание информации о метаданных, хранимых в секторе и регистрация считанной информации в кэше адресов.

**Блок упаковки и распаковки метаданных и данных**

Для процесса записи: осуществляет разделение данных, хранимых в кэше адресов от собственно блока хранимых данных

Для процесса чтения: осуществляет формирование пакета данных, содержащего блок хранимой информации и связанных с ним метаданных, для передачи в модуль управления.

## **Кластеризация**

Сервис кластеризации предназначен для контроля доступности модулей управления в составе кластера и обеспечения автоматической миграции разделяемых ресурсов (IP адрес управления, виртуальные сетевые интерфейсы и др.) в пределах кластера, а также контроля работы сервисов в пределах модуля управления. Сервис кластеризации учитывает наличие в кластере контроллерных пар с взаимной репликацией кэша записи. Схема кластеризации показана на Рисунок 6.

Компоненты сетевых интерфейсов указаны на схеме. Атрибуты IP адреса – IP адрес, маска подсети, шлюз подсети. Кластерный сервис работает на выделенном интерфейсе (внутренняя сеть). Между парами контроллеров также имеется выделенный интерфейс репликации кэша записи (infiniband). Группа физических интерфейсов позволяет логическому интерфейсу перемещаться между физическими интерфейсами в группе.

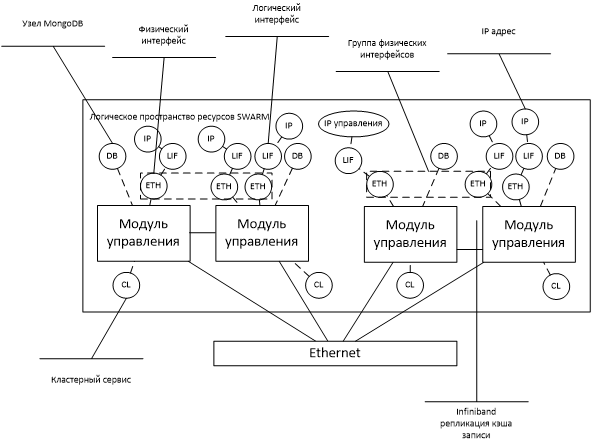


Рисунок . Схема кластеризации

1. **Структура файлов**

Структура файлов одинакова на всех модулях РСХД. В зависимости от функциональных требований запускается конфигурация для модуля управления или для модуля хранения.

Файлы программы находятся в каталоге: /home/bi/swarm2/build.

## **Исполняемые файлы**

* swarm\_cluster - исполняемый файл сервиса кластеризации;
* swarm\_controller\_app - исполняемый файл модуля управления;
* swarm\_iscsi\_config - исполняемый файл сервиса конфигурации iscsi;
* swarm\_net - исполняемый файл сервиса управления сетью;
* swarm\_nfs\_config - исполняемый файл сервиса настройки nfs;
* swarm\_rpc\_router - исполняемый файл сервиса rpc;
* swarm\_rpc\_ws - вспомогательный сервис-адаптер для сервиса rpc;
* swarm\_module\_app - исполняемый файл модуля хранения;
* swarm\_module\_stress\_app - исполняемый файл стресс теста модуля хранения;
* swarm\_controller\_stress\_app - исполняемый файл стресс теста модуля управления.

## **Библиотеки**

* ibcontroller.so - библиотека контроллера;
* libs3\_srv\_lib.so - библиотека для бекенда s3 сервера;
* libtcmu-runner.so - библиотека для iSCSI бэкстора;
* libswarmfs.so - библиотека для fuse;
* libstopper.so - библиотека для остановки приложений различными способами, в т.ч. через сигналы системы;
* libswarm\_rpc.so - библиотека для RPC клиента;
* libshare.so - библиотека с утилитами, такими как обслуживание ID объекта;
* libkitten\_logger.so - библиотека для логирования;
* libkittens\_trace.so - библиотека для профилировки;
* libwrite\_cache.so - библиотека для кэша записи;
* libmodule\_server.so - библиотека с сервером для модулей хранения;
* libdistributor.so - библиотека для распределения через CRUSH;
* libtransport.so - TCP клиент-сервер;
* libcontroller\_replicator.so - библиотека, обслуживающая RDMA;
* libread\_cache.so - библиотека, обслуживающая кэш чтения;
* libmodule\_protocol.so - протокол работы с модулем хранения;
* libmongo.so - обертка вокруг драйвера mongoDB;
* libcrush.so - реализация алгоритма CRUSH;
* libsettings.so - утилитарная библиотека для работы с файлом настроек;
* liberror.so - утилитарная библиотека по процессингу ошибок;
* libjson.so - утилитарная библиотека для работы с JSON форматом;
* libswarm\_db.so - утилитарная библиотека для работы с mongoDB;
* libsock\_transfer.so - утилитарная библиотека для RPC сервиса.

1. **Технические характеристики РСХД**

|  |  |  |
| --- | --- | --- |
| № | Параметр | Значение |
| 1 | Максимальный объем хранимых данных, Пбайт | 2 |
| 2 | Максимальная скорость чтения/записи данных (при использовании флеш накопителей), МБ/с | 500 |
| 3 | Количество операций чтения/записи (при использовании флеш накопителей), IOPS | 100000 |
| 4 | Средние задержки на чтение и запись (при использовании флеш накопителей), мс | 5 |
| 5 | Целевой коэффициент дедупликации и компрессии данных, % | 15 |
| 6 | Минимальное поддерживаемое количество узлов кластера в распределенной архитектуре, узлов | 3 |
| 7 | Максимальное поддерживаемое количество узлов кластера в распределенной архитектуре, узлов | 64 |
| 8 | Поддерживаемое количество кластеров в геораспределенной катастрофоустойчивой архитектуре | 2 |