

Проекты компании Acronis

6 курс

Contact: eabatalov89@gmail.com

Кеш для WAN оптимизации бекапа данных

Проект

О проекте (1)

- Множество клиентов отправляет свои данные в облако для бекапа
- Медленная скорость бекапа из-за низкой пропускной способности сети и большого объема данных
- Отправляемые данные содержат дубликаты (например, файлы Mac OS)

О проекте (2)

- Вероятность появления дубликатов зависит от внешних событий (например, обновления Windows)
- Для ускорения бекапа хочется не передавать дубликаты по медленной глобальной сети

О проекте (3)

- Данные каждого клиента хранятся в облаке без дедубликации (по экономическим соображениям)
- Необходимо, однако, иметь в облаке кеш данных, уже отправленных клиентами, чтобы не передавать дубликаты от других клиентов

Цель проекта

- Разработать оптимальный алгоритм кеширования клиентских данных на сервере для исключения передачи дубликатов по сети
 - Оптимальность - высокая вероятность попадания в кеш, минимальный размер кеша, высокая скорость работы кеша

Языки - C++, Python

Задачи проекта

- Реализовать модель поведения клиентов и сервера, в т.ч. алгоритмы работы кеша, модели железа
- Добавить сбор множества статистик в модель
- Научиться оценивать эффективность различных подходов к кешированию по собранной статистике
- Модификация алгоритмов и их проверка на модели
- Возможность добавить машинное обучение

Компрессия данных в distributed COW FS

Проект

О проекте (1)

- COW FS пишет данные append only потоком
- Есть код COW FS
- Данные надо разбить на правильные блоки для записи на диски
- При наличии компрессии конечный размер блока заранее неизвестен

О проекте (2)

- Пока не сжат первый блок, не ясно по какому смещению окажется второй и т.д.
- N-ый блок не может “ждать” сжатия N-1-х (скорее всего это будет долго)
- Хотелось бы добавить адаптивную компрессию - не сжимать то, что не сожмется и наоборот

Цель проекта

- Разработать оптимальный алгоритм online/offline сжатия данных в распределенной COW FS
 - Оптимальность - throughput/latency COW FS, репликации, экономия места на дисках

Языки - C++, Python

Задачи проекта

- В существующий код ФС добавить модель георепликации, код сжатия данных
- Добавить сбор множества статистик
- Научиться оценивать эффективность различных подходов к сжатию по собранной статистике
- Модификация алгоритмов и их проверка на модели

I/O гарантии в распределенном хранилище данных

Проект

О проекте (1) пример

Есть мощная VM в хостинге. Ее виртуальные диски лежат в распределенном хранилище на разных нодах. У VM есть SLA на IO. Надо SLA выполнить.

О проекте (2)

- Ко всем storage нодам обращается множество I/O client'ов
- Как понять, кому сколько давать I/O?
- Как понять что наша VM получила гарантированные ей X IOPS'ов? Не суммировать же каждую секунду IOPS'ы со всех хостов?

О проекте (3)

- Если VM дали например $X/2$, что это означает? А может она больше и не хотела?
- В каком месте планировать, приоритизировать IO запросы?

Цель проекта

- Разработать планировщик распределенных IO запросов с приоритизацией и лимитами

Языки - C++, Python

Задачи проекта

- Реализовать модель описанного use case'a и др.
- Реализовать множество вариантов планирования
- Добавить сбор множества статистик
- Научиться оценивать эффективность различных подходов к планированию по собранной статистике
- Модификация алгоритмов планирования и их проверка на модели

Кросс-платформенный userspace драйвер для XFS

Проект

О проекте (1)

- XFS - высокопроизводительная ФС, используемая в серверах
- Предоставляет множество классных функций
- <http://en.wikipedia.org/wiki/XFS>
- Хочется уметь с ней работать не зависимо от наличия, типа, конфигурации ОС (для восстановления, обслуживания ФС)

Цель проекта

- Реализовать read only кросс-платформенный userspace драйвер для XFS
- Поддержать восстановление XFS раздела на разделы большего и меньшего размера

Язык - C++

Условия работы (1)

- Проекты из Москвы. Работа полностью удаленная
- Каждые 1-2 недели созвоны в skype. Встречи с Евгением в БЦ Таймс, код ревью и т.п. при необходимости
- Нет четких обязательств, но желательно работать по 20 часов в неделю

Условия работы (2)

- +15000-20000р. к стипендии (начиная с июля) при наличии видимых результатов
- Можно совмещать с работой в весеннем семестре 6 курса