



Сравнение NoSQL баз данных

А П А С Х Е
HBASE



mongo**DB**




cassandra



Тип	Column-oriented (Java) Колоночная	Document-based (C++) Документо-ориентированная Сохраняет весь документ в BSON (binary JSON) формате. INSERT/UPDATE/FETCH на документ целиком.	Column-oriented (Java) Колоночная
Блокировки	Нет блокировок	Блокировка записи и множественного чтения	Нет блокировок
Вторичные индексы	Нет встроенной поддержки	Поддерживает	Поддерживает
Модель распределения данных	Репликация HDFS Одна точка отказа – namemode (HDFS)	Шардинг	Peer-to-peer no-single-point-of-failure
Механизм хранения	HDFS	B-деревья	Механизм хранения только добавляет обновленные данные; Поддержка SSD & mixed SSD и HDD
Аналитика	Использует инфраструктуру Hadoop	Встроенный Map-Reduce Framework	CFS (HDFS совместимая Cassandra File System) + есть интеграция с Hadoop

«Распределенная, децентрализованная, эластично масштабируемая, высокодоступная, отказоустойчивая база данных с открытым исходным кодом, настраиваемой согласованностью и ориентацией на столбцы. Проект распределенной структуры основан на Amazon Dynamo, а модель данных — на Google Bigtable»


(источник: «Cassandra: The Definitive Guide», O'Reilly Media, 2010, p. 14).

DATASTAX  Разработка программных продуктов для Apache Cassandra.

Некоторые из преимуществ Cassandra:

- ✓ высокая масштабируемость и надежность (**no-single-point-of-failure architecture**);
- ✓ реализация семейства NoSQL Column;
- ✓ SQL-подобный язык запросов (начиная с версии 0.8 - **CQL**) и поддержка поиска посредством вторичных индексов;
- ✓ настраиваемая согласованность и поддержка репликации;
- ✓ гибкая схема данных;
- ✓ механизмы фоновой оптимизации данных (**compaction**);
- ✓ поддержка Map-Reduce через Hadoop, Spark (MapReduce).
Поддержка Apache Pig () , Apache Hive () .

Особенности:

 нормализация для уменьшения избыточности данных

NO SQL нет поддержки Foreign Key, нельзя сделать JOIN для удовлетворения запросу

Моделирование таблиц в зависимости от запросов.



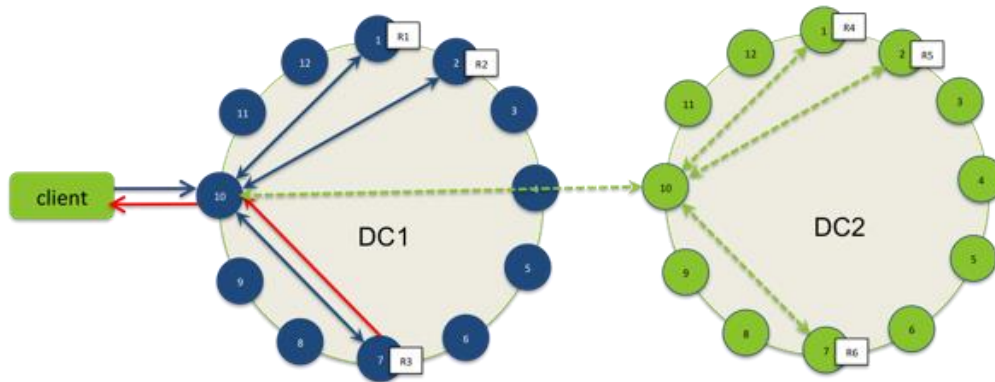
Cassandra предназначена для хранения больших объемов данных (сотен терабайт) на множестве машин, соединенных в кольцо.

Реализация 

Поддержка 

Разработка 

Архитектура Cassandra



Настраиваемая согласованность данных:

Replication factor (RF) – количество копий данных в кластере.

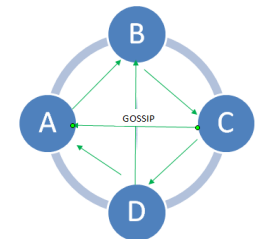
R – число узлов, к которым обращается клиент при чтении данных,
W – число узлов, от которых ожидается подтверждение успешной записи

Node (узел) – основной компонент инфраструктуры. Хранение данных.

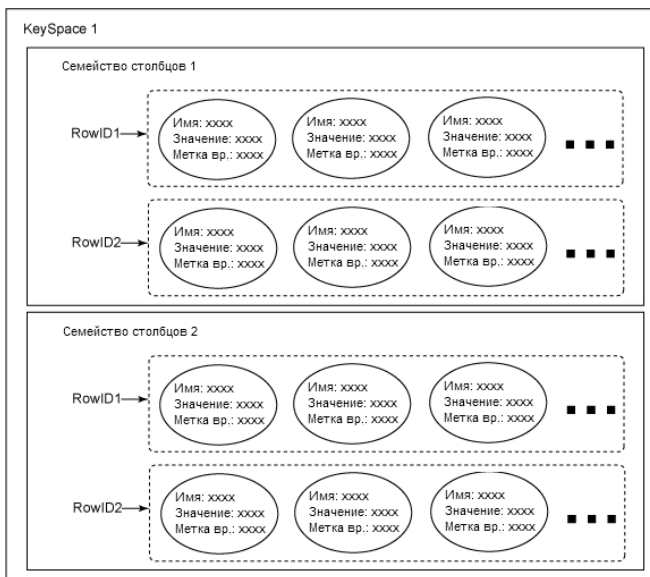
Data Center – объединение узлов для решения определенных задач.

Cluster содержит один или несколько Data Center.

Gossip – протокол пиринговой коммуникации между узлами Cassandra.



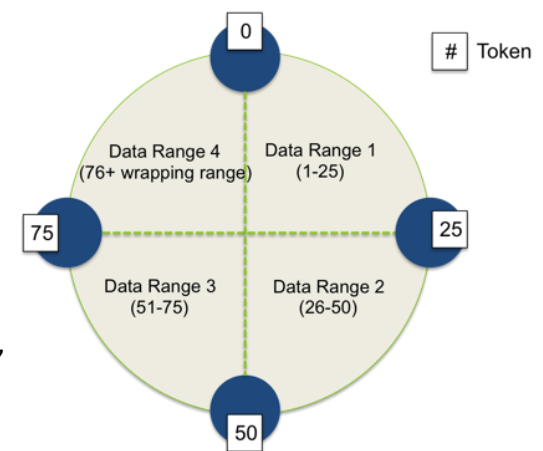
PEER TO PEER DISTRIBUTION
MODEL OF CASSANDRA



Модель данных

- ✓ **Column (Столбец)** – содержит имя, значение и метку времени.
- ✓ **Row (Строка)** – именованная коллекция столбцов.
- ✓ **Column Family / Table (Семейство столбцов)** – именованная коллекция строк.
- ✓ **KeySpace (Пространство ключей)** – группа из многих семейств столбцов, собранных вместе.

Partitioner – определяет методы распределения данных между узлами кластера.

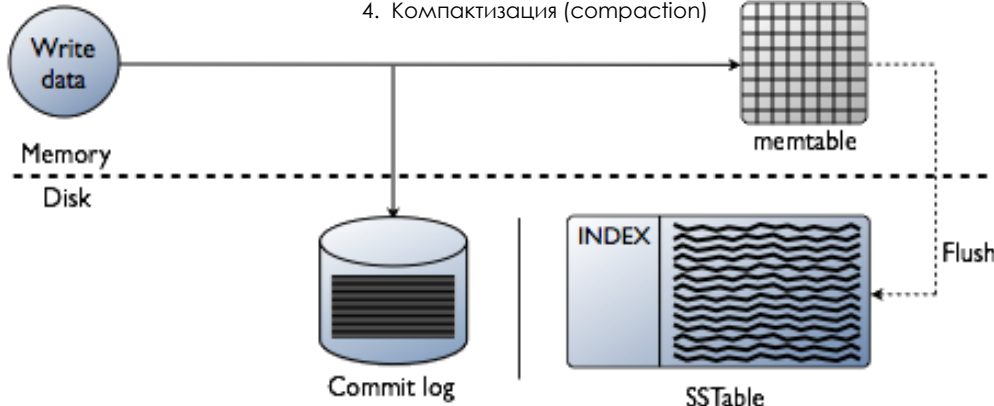


Чтение / запись в Cassandra

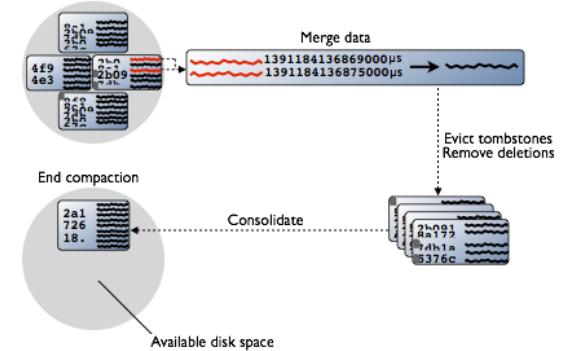


Запись

1. Запись данных в commit log
2. Запись в memtable
3. memtable → SSTables
4. Компактизация (compaction)

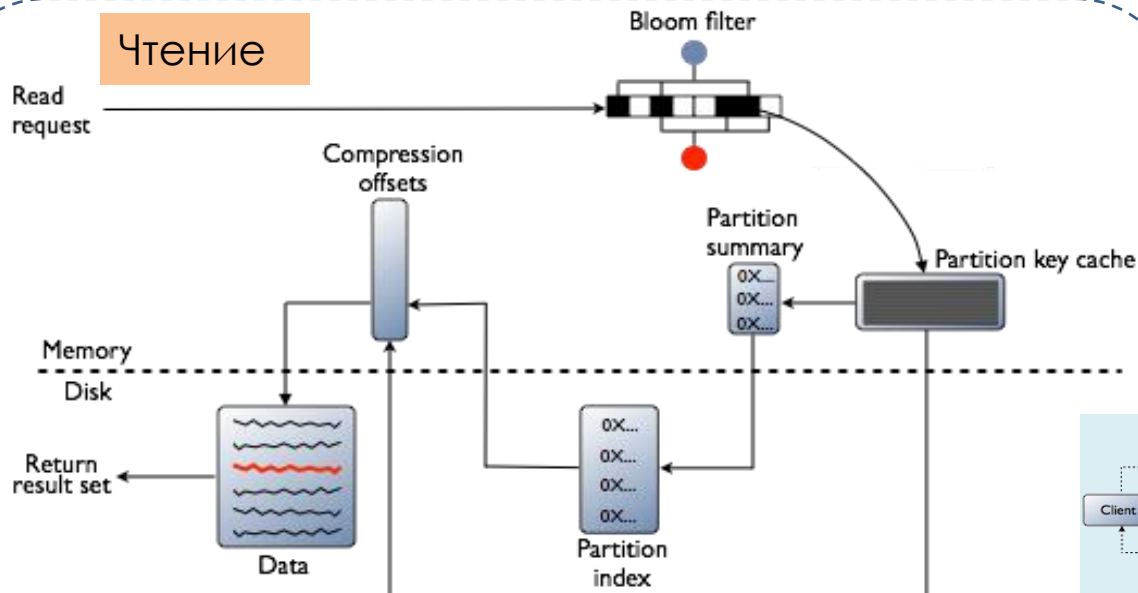


Start compaction



Компактизация

Чтение



Периодическая компактизация необходима для поддержания БД в «здоровом» состоянии, так как в Cassandra процессы записи/обновления/удаления не выполняются «in place».

Ограничения:

- ✓ Значение колонки ≤ 2GB;
- ✓ Значение коллекции ≤ 64KB.
- ✓ Макс. к-во колонок в строке = 2 млрд.

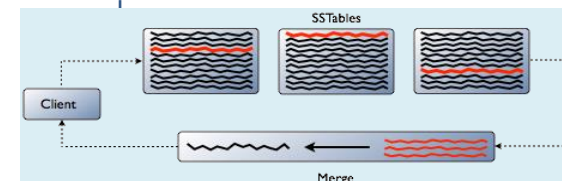




Таблица диапазонов хэш-значений ключей

Узел	Начало диапазона	Конец диапазона	Первичный ключ	Значение хэш-функции
A	-9223372036854775	-46116860184273	3372	-672337285403678
B	-4611686018427387	-14894768874875	4976	-224546267672322

COMPOUND PRIMARY KEY

[Partition Key + Clustering Key + Unique Key]

```
CREATE TABLE task (
    taskID int,
    modificationTime timestamp,
    pandaID bigint,
    PRIMARY KEY (taskID,
                modificationTime,
                jobID);
```

SQL Таблица «TASK»

ROW [3372]	2014-08-11, 2233974170	2014-01-02, 2034481717	2014-01-02, 2034481720	2014-01-02, 2034481727	...
ROW [4976]	2014-01-04, 20356778401	2014-01-06, 20364186511	-	-	...

```
CREATE TABLE source_status (
    source varchar,
    jobStatus varchar,
    modificationTime timestamp,
    pandaID bigint,
    PRIMARY KEY ((source, jobStatus),
                modificationTime,
                pandaID));
```

SQL Таблица
«SOURCE_STATUS»

ROW [user : cancelled]	2014-08-11, 2233964401	2014-08-11, 2233974170	...
ROW [managed : failed]	2014-08-10, 2236775011	2014-08-10, 2236840547	...
ROW [managed : finished]

