

Cassandra Summit 2016

株式会社 INTHEFOREST

村岡 志保

Keynote

- ▶ Cassandra New Storage Engine
- ▶ Materialized views
- ▶ Tick-tock releases
- ▶ SASI
- ▶ 3.5 driver
- ▶ Read path
- ▶ TWCS
- ▶ GROUP BY



Cassandra New Storage Engine(3.0)

- ▶ Cassandra 3.0では、ストレージエンジンのコードが
リファクタリングされてCQLに最適化し効率がよくなった
- ▶ パフォーマンスの最適化
- ▶ データ一貫性の操作を改善
- ▶ データ保存領域の50%の削減

Materialized views(3.0)

- マテリアライズド・ビューは別のテーブルのデータを使用して、新しいプライマリーキーを指定して構築したテーブル
- 高いカーディナリティのデータに適している

マテリアライズド・ビューの作成要件

- ▶ 元のテーブルのプライマリ・キーのカラムは、マテリアライズド・ビューのプライマリ・キーの一部である必要があります。
- ▶ マテリアライズド・ビューのプライマリ・キーには1つのみの新しいカラムを追加できます。

Materialized views

```
CREATE TABLE songs (  
  id uuid PRIMARY KEY,  
  title text,  
  album text,  
  artist text  
);
```

```
CREATE MATERIALIZED VIEW song_by_album AS  
SELECT * FROM songs  
WHERE album IS NOT NULL  
PRIMARY KEY (album, id);
```

```
SELECT * FROM songs_by_album  
WHERE album = 'Tres Hombres';
```



Tick-tock releases

3.0 Engine MV

3.2 Hints compaction

3.4 SASI

3.6 Read path

3.8 Clustering in WHERE clause TWCS

3.10 Slow query log

3.1 Fixes

3.3 Fixes

3.5 Fixes

3.7 Fixes

3.9 Fixes



SASI :

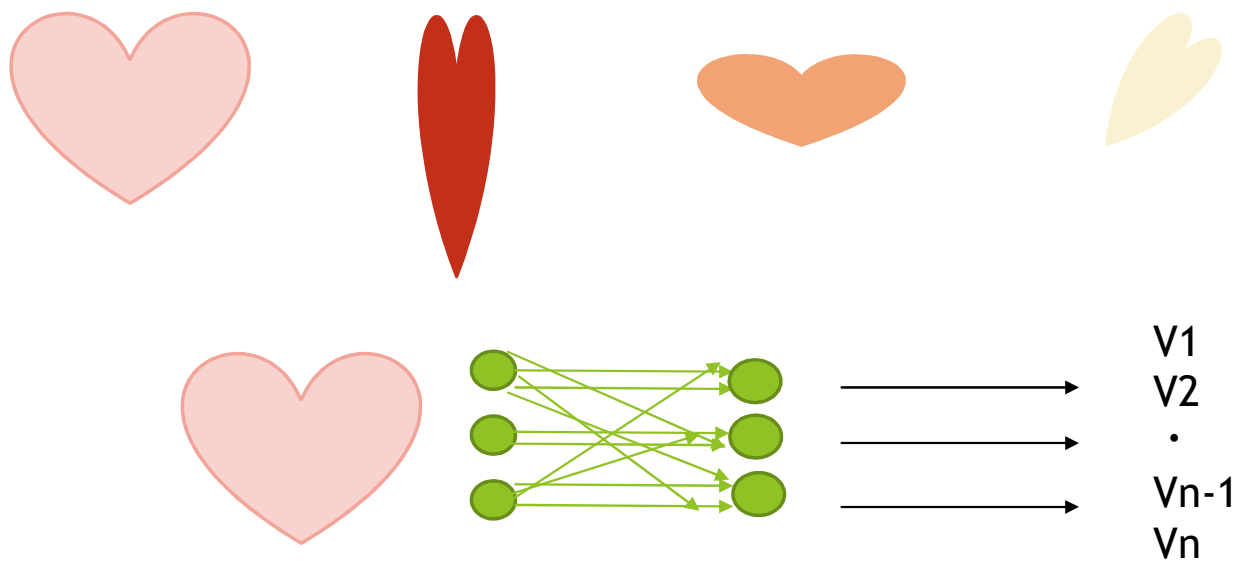
SSTable-Attached Secondary Indexes

- ▶ 大幅にセカンダリインデックスのパフォーマンスを改善
- ▶ CQLを使用し、テーブルに定義された非コレクションカラムを作成することができる
- ▶ SASIは3種類のインデックスがあります
PREFIX, CONTAINS, SPARSE

Clustering in WHERE clause

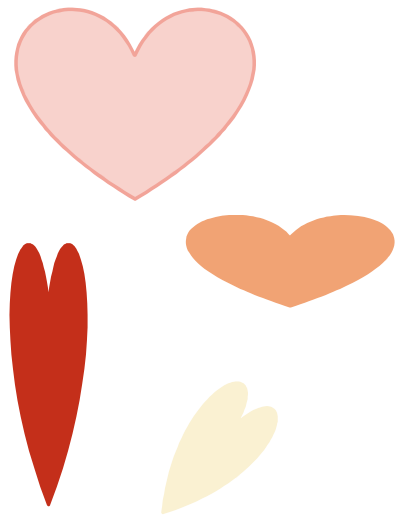
- ▶ カサンドラ3.6以降では、クラスタリングカラムはセカンダリインデックスが作成されていない場合でもWHERE句で定義することでALLOW FILTERINGを使用できます。

Semantic Image Search

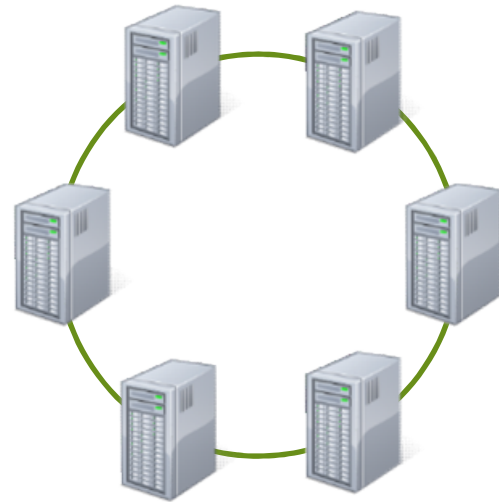


- ▶ 1イメージ 4000次元ベクトル
- ▶ 1TBイメージ 300GBの特徴

Feature Extraction



Cassandra



テーブル

▶ CREATE TABLE features

listind_id uuid,

image_id uuid,

future_vector blob,

PRIMARY KEY ((listing_id),image_id)

Sparkの力

Sparkとは

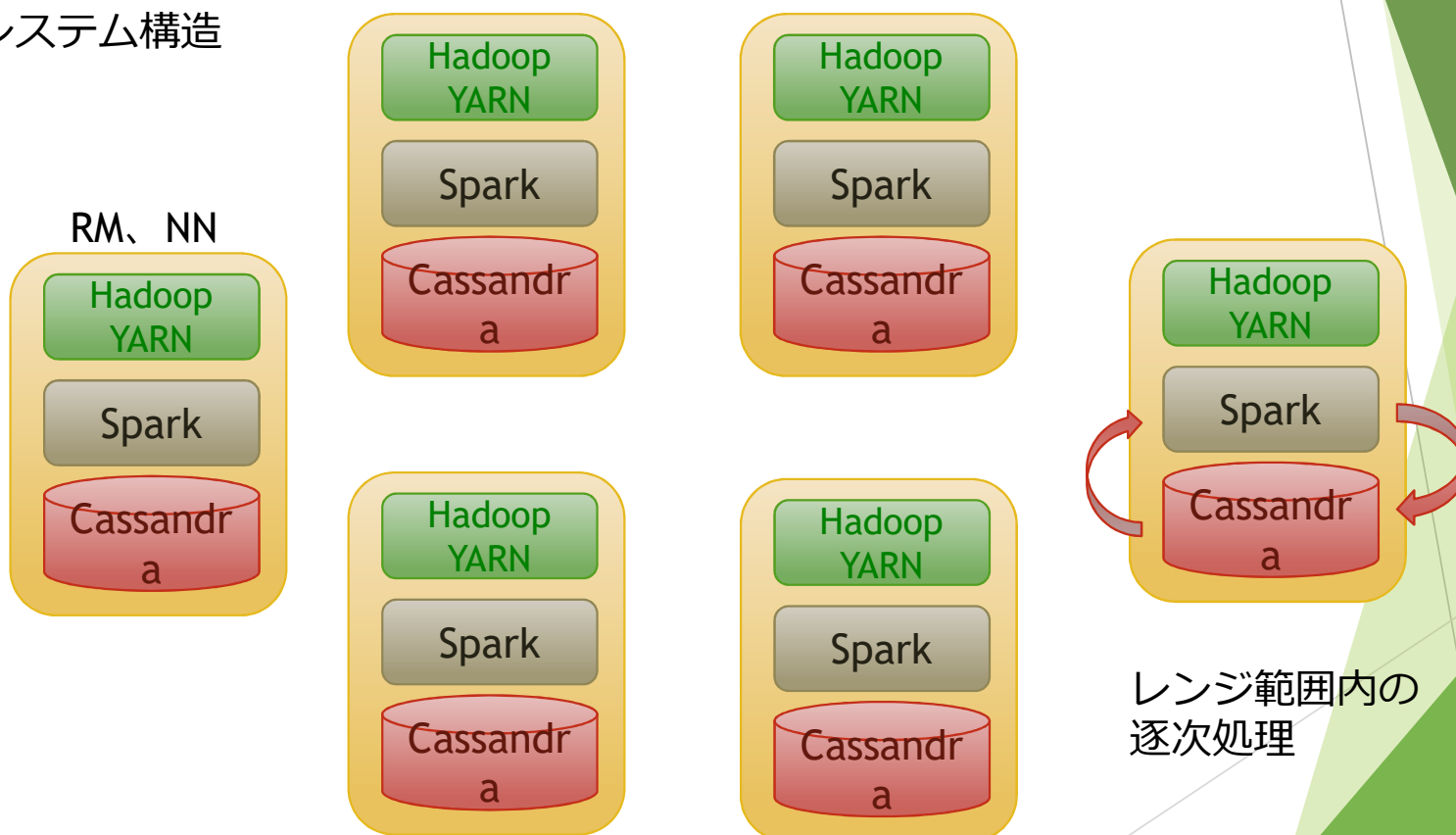
高速なデータ分析のための新たな手段

Sparkの特徴

- ◇ RDD (Resilient Distributed Dataset・弾性分散データセット)
 - ◇ 不変 (イミュータブル)
 - ◇ 分割・分散配置
 - ◇ インメモリー
 - ◇ 遅延評価
- ◇ Hadoop連携
 - ◇ HDFS自動連携
 - ◇ YARN連携

Sparkの力

システム構造



レンジ範囲内の
逐次処理

Spark Similarity Search

- ▶ 1. 300GB の特徴をスパークにキャッシュ
- ▶ 2. 検索イメージに全ての特徴ベクトルスコアをもつ
- ▶ 3. 各リスト内の画像のための集計スコア
- ▶ 4. (optional) Join on 'live' C* data
- ▶ 5. 書き込み結果をカサンドラに戻す

- ▶ CREATE TABLE features
 - listind_id uuid,
 - image_id uuid,
 - future_vector blob,
 - PRIMARY KEY ((listing_id),image_id)
- ▶ CREATE TABLE stock_level
 - listind_id uuid,
 - inventory int,
 - etag uuid,
 - PRIMARY KEY (listing_id)
- ▶ CREATE TABLE query_results
 - query_id uuid,
 - listing_id uuid,
 - score float,
 - PRIMARY KEY ((query_id),score)

まとめ

- ▶ 新しいカサンドラの機能はニーズにより様々な変化をしている
- ▶ 画像検索や動画検索の場面でカサンドラ・スパークの利用が広がっている
- ▶ トラブルシューティングの情報が増えている