Optimizing the data warehouse

Justin Erickson // Sr Director, Products @ Cloudera
Agenda

- Motivation
- Process and tips
- Demo example
Trends within Data Warehousing

Expand to Do More
Nearly 40% want greater capacity for growing data, users, reports, analyses, etc¹

Improve Responsiveness
67% of users are requesting to do more BI/analytics on their own²

Balance Business Critical & Exploration
42% looking to augment EDW with modern platform¹

¹Data Warehouse Modernization In the Age of Big Data Analytics, TDWI, 2016
²Achieving Greater Agility with Business Intelligence, TDWI, 2014
Limitations of Existing Data Warehouse Landscape

- Not able to take on more reports, use cases, users, etc.
- Constrained exploration to prevent risking critical SLAs

- Need to contain costs for existing workloads
- Difficult to justify budget for expansion
- Struggle to do more with less

- Proliferation of data silos to address non-SQL and exploratory workloads
- Inefficiencies for storage, processing, and people

- Designed for curated reports, not iterative, self-service analytics
- Not built for self-service cloud flexibility
Existing DW Landscape

- Data Sources
- ETL/Staging
- EDW
- Archive
- Canned Reports
- Self-Service BI/Ad Hoc
- Dashboards/Analytic Applications
- Data Marts
- Non-SQL Workloads
Modern Data Warehouse Landscape

Data Sources
- Non-SQL Workloads
- Self-Service BI/Ad Hoc
- Dashboards/Analytic Applications
- Flexible Reporting

Modern Data Platform
- Data Science & Engineering
- Analytic Database
- Operational Database
- Shared Data Experience

EDW

Fixed Reports
Process and tips
But where do you start?

Query volume can be huge

Queries can be very complex

- How do you determine what workloads to run on Cloudera’s platform?
- Will the queries run efficiently?
- What does it take to migrate?
- How do you prioritize?

Numerous databases, thousands of tables, many users and applications
EDW Optimization Process

Evaluate
- Evaluate the need for offload
  - Clarify Use Cases
  - Set Objectives
  - Identify Suitable Workloads
  - Initial POC

Plan
- Impact analysis, prioritized plan
  - Impact Analysis
  - Estimate Effort
  - Prioritized Plan
  - Capacity Planning

Offload
- Offload each workload
  - Risk Analysis
  - Schema Design
  - Offload Actions
  - Test & Validate

Optimize
- Optimize performance
  - Fine Tuning Data Model on Hadoop
  - Optimize Queries for Performance
  - Validate ROI, Cost
Before you start...

• Clarify and prioritize motivations
  • Cost savings
  • New use cases and/or capabilities

• Understand current state
  • Drivers for costs, usage, and needs
  • Current data models and optimizations
  • Current common usage patterns and requirements

Cost savings
Use case flexibility
More / converged data
Beyond SQL
Cloud migration
Start with ETL or BI?

Offload ETL
- Often >40% of EDW consumption
- Least impactful to end-users

Helps most with goals of:
- Reducing costs
- Relieving resource pressures
- Faster, more scalable, more flexible ETL

Offload BI
- More end-user benefits
- Directly impacts end-users

Helps most with goals of:
- New use cases and capabilities
- Self-service and exploration
- Speeding up use case delivery
- Scaling analysis to more data
- Converging data silos/marts
ETL offload tips

• Most typical starting point for new customers given Hadoop’s origins
• Mostly manual effort convert jobs to Spark and/or Hive

• Consider migrating data pipelines with:
  • High ROI in freeing up EDW resources and costs
  • Sizable opportunities for transformational speed-ups from scale-out
  • Use for new use cases for value beyond cost savings
BI offload tips

• Segment and prioritize workloads
  • Review common workload patterns to determine conversion costs
  • Look for highest ROI starting from simpler workloads
  • Consider which have most direct needs for your re-platforming objectives

• Move data model as-is then optimize
  • Hadoop is the ideal ETL engine regardless
  • Optimize for time-to-value
  • Focus on known need and minimize impact
  • Still follow best practices (eg file sizes, partitioning, Parquet etc)

• Optimize based on need and most common usage patterns
  • Optimizing for common patterns provides scalable gains
Leverage data zones and governance

Legend:
- **Raw / Prep**
- **Explore**
- **Analysis**
- **Real-time**

Data Sources

Landing zones

Data Prep and ETL zones

Curated data zones

Real-time data streaming & serving zones

Self-Service/Ad Hoc

Dashboards/Analytic Applications

Reporting

Data Exploration zones

© Cloudera, Inc. All rights reserved.
Demo

Navigator Optimizer
EDW Optimization Process

Evaluate
- Evaluate the need for offload
  - Clarify Use Cases
  - Set Objectives
  - Identify Suitable Workloads
  - Initial POC

Plan
- Impact analysis, prioritized plan
  - Impact Analysis
  - Estimate Effort
  - Prioritized Plan
  - Capacity Planning

Offload
- Offload each workload
  - Risk Analysis
  - Schema Design
  - Offload Actions
  - Test & Validate

Optimize
- Optimize performance
  - Fine Tuning Data Model on Hadoop
  - Optimize Queries for Performance
  - Validate ROI, Cost
Identify Suitable Workloads
Get insights into what’s happening today

Understand Usage
- Top tables, top columns
- Usage-based ER diagram
- Unused tables & columns

Understand Suitability
- Query duplication
- Query complexity
- Query compatibility

Initial POC
- Identify simple, initial workload for PoC
- Get partitioning key suggestions
Impact Analysis & Prioritized Plan
Understand what takes to offload

Impact Analysis
• Focus efforts by identifying duplications
• Risk assessment based on complexity and best practice
• Query or schema changes required for offload

Prioritized Plan
• Estimate effort
• Identify easiest places to start for fast success
• Prioritize workloads based on effort
Predictable Offload
Remove the guesswork

Understand offload requirements
• Workload breakdown by risk level & assessment by offload unit
• Query syntax compatibility
• Develop data-/usage-driven offload strategy

Actionable recommendations
• Recommendations for each offload unit
  • Schema design
  • DDL generation
  • Query optimization
Optimizing in Hadoop

Maintain peak performance

Understand usage and keep up with data needs
- Understand most common usage patterns
- Identify optimization opportunities
- Proactively adjust data models

Performance optimizations
- Best practice guidance for Hive and Impala
- Query performance optimization
- Increase platform adoption
Thank you