Foundations for Successful Data Projects

Strata Data Conference, San Francisco 2019
Ted Malaska | @ted_malaska
Jonathan Seidman | @jseidman
About the presenters

Ted Malaska

- **Capital One**: Director of Enterprise Architecture
- **Blizzard Ent**: Director of Engineering of Global Insights
- **Cloudera**: Principal Solution Architect
- **FINRA**: Lead Architect
- **Contributor**: Apache Spark, Hadoop, Hive, Sqoop, Yarn, Flume, others
About the presenters

Jonathan Seidman

- Software Engineer at Cloudera
- Previously Technical Lead on the big data team at Orbitz
- Co-founder of the Chicago Hadoop User Group and Chicago Big Data
Foundations of Successful Data Projects

Understand the problem
Select software
Manage risk
Build effective teams
Build maintainable architectures
Agenda

- Understanding the key data project types
- Selecting data management solutions
- Building effective teams
- Managing risk in projects
- Ensuring data integrity
- Metadata management
- Using abstractions
Understanding the Key Data Project Types
Understanding the key data project types

- Major Data Project Types
- Primary Considerations & Risk Management
- Team Makeup
Major Data Project Types

- Data Pipelines and Data Staging
- Data Processing and Analysis
- Application Development
Data Pipelines and Data Staging

- Sourcing Data
- Transmitting Data
- Staging Data
- Accessibility Options
- Discovery
Data Processing and Analysis

- Curating Data
- Cultivating Ideas
- Data Product Generation
  - Reports, Models, Insight, Charts, ...
Application Development

- Traditional or Model Serving
- Inner Loop
- Outer Loop
Primary Considerations

▪ Data Pipelines and Data Staging
▪ Data Processing and Analysis
▪ Application Development
Primary Considerations

Data Pipelines and Data Staging
Data Pipelines and Data Staging – Considerations

- On boarding paths for Data Suppliers
  - Files
  - Embedded code
  - APIs (Rest, WebSocket, GRPC, Syslog, ...)
  - Agents
Data Pipelines and Data Staging – Considerations

- Transmission
  - At Least Once, Duplication, Latency, and Ordering

- Tokenization & Auditing & Governance
  - GDPR, CA Protection Laws, Misuse, Data Breach

- Quality
  - Schema Validation, Rules Validation, Carnality Verance

- Access
  - Security, Matching the use case to the storage system
Data Pipelines and Data Staging – Considerations

- Meta Management
  - New and mutated Datasets
  - Security

- Access
  - Matching the use case to the storage system
  - SQL is King
  - No one tool
  - Trade Offs
    - Cost vs Time to Value vs Value of Data
Primary Considerations

Data Processing and Analysis
Data Processing and Analysis – Considerations

- Curating Data
  - Working with Producers
  - Joining
  - Time series
  - CDC

- Undering Quality of Data
  - SLAs
  - Correctness of the Data
  - Stability of the Data
  - Coupling
Data Processing and Analysis – Considerations

- Cultivating Ideas
  - Defining Real Goals
  - Evaluating ROI

- Productionization of Pipelines
  - Service Reliability Engineering

- Culture
  - ML vs AI vs Engineer
Data Processing and Analysis – Considerations

- Understanding
  - Explainable Outcomes
  - Defendable Solutions

- Promotion Paths
  - Deploying Products
  - Historical Evaluation
  - Up to Date Auditing
Primary Considerations

Application Development
Application Development – Considerations

▪ Availability and Failure
  - How will it fail
  - How will failure impact customers
  - What level of fail should be tested for
  - Levels of failure design

▪ State Locality and Consistency
  - What are the requirements
  - Speed, cost, or truth
  - Transactions and Locking
Application Development

- Latency and Throughput
  - Expectations and Throughput
  - Is it really big data?
  - Inner and Outer Looping

- Granularity of Deployments
  - Monolith single deployment
  - Monolith microservices

- Culture
  - Development Towers
  - Over the wall
  - Development Granularity
Team Makeup
Team Makeup

- Data Pipelines and Data Staging
- Data Processing and Analysis
- Application Development
Data Pipelines and Data Staging – Team Makeup

- Data Engineers
- Site Reliability Engineers (SRE)
- Application Engineers
- Data Architects
- Governance
- Solution Engineers/Architects
Data Processing and Analysis – Team Makeup

- Visionaries
- The Brains
- Problem Seekers
- Engineers
- Duct Tapers
- Tech Debt Payers
- Site Reliability Engineers (SRE)
Application Development – Team Makeup

- Web Developers
- Front end Developers
- Data Engineers (DBAs)
- Performance Focused Engineers
- SOA / Queue Engineers
- Site Reliability Engineers (SRE)
Evaluating and Selecting Data Solutions
Evaluating and Selecting Data Solutions

- Solution Life Cycles
- Tipping Point Considerations
- Considerations for Technology Selection
Solution Life Cycles

- Private Incubation Stage
- Release Stage
- “Curing Cancer” Stage
- Broken Promises Stage
- Hardening Stage
- Enterprise Stage
- Decline and Slow Death Stage
Private Incubation Stage

- Technology Trigger
- Vision
Release Stage

▪ Changes
  - Inviting People In
  - Documentation
  - Marketing

▪ Reasons for Releasing
  - Money
  - Hiring
  - Culture
  - Future Building

▪ Big Promises
“Curing Cancer” Stage

- Big Promise
- Maybe outside area of expertise
  - Promise to push internally
  - Promises to gain influence
  - Promises to get attritions
- Promises can be good and bad
Broken Promises Stage

- Cracks in the Dream
  - Scale
  - Usability
  - Use Case
  - Security
  - Practicality
  - Skill Requirements
  - Auditability
  - Maintainability
  - Integration
  - Quality
  - Lies
Hardening

- Balance Features
- Technical Debt
- Partnering
- Corp Partnerships
- Leadership Stories
- Easy Success Paths
Enterprise Stage

- Stable
- Predictable
- Easy to hire for
- Supportable / Maintainable
- Pragmatists outnumber innovators
- No longer cool, but still very lucrative
Slow Decline Stage

- Not Worth Retiring
- Not worth Investing In
- Good Enough
Tipping Point Considerations

- Mavericks
- Connectors
- Salesman
- Stickiness
- Context
Mavericks

- Passion Driven
- Helpful
- Bottom Up Power
- They see the future or may see shadows
Connectors

- High triangles
- Trusted weak ties
- Gateways for pain, needs, and opportunities
- Considering the towered companies
Sales Man

- Make the Deal Happen
- Right or wrong doesn’t matter as much as action
- Momentum starters
Stickiness

- Think about gravity
  - Data
  - Code
  - User’s Favor
  - Results
Context

▪ Where is the company
▪ Looking for Opportunities
  - Holding down the fort
  - Lower cost
  - Play around
▪ The Swing Pendulum Effort
  - Where is the ball now and where is it heading
Tipping Point Considerations

- Mavericks
- Connectors
- Salesman
- Stickiness
- Context
Considerations for Technology Selection

- Demand
- Fit
- Visibility
- Risks
Evaluating the Demand

- Business Needs
- Internal Demand
- Desire to live on the edge
Evaluating the Fit

- Primary Capability
- Skill Sets
- Level of Commitments
- Level of Alignment
Evaluating the Visibility

- Benchmarks
  - Hidden biases, Motivated Biases, Unfair Comparisons
- Fundamentals
  - There is no magic
- Leaders Success
- Market Trends
Reviewing Fundamentals

- Relative Location of Data to Readers
- Compression formats and rates
- Data Structures
- Partitioning, Replication, and Failure
- API and Interfaces
- Resource Allocations and Tuning
Reviewing Market Trends

Google Trends
Reviewing Market Trends

Github activity
Reviewing Market Trends

Jira Counts and Charts

Created vs Resolved Issues Report

This chart shows the number of issues created vs the number of issues resolved in the last 30 days.

Pie Chart Report

Data Table

<table>
<thead>
<tr>
<th>Issues</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unassigned</td>
<td>391</td>
</tr>
<tr>
<td>Wes McKinney</td>
<td>156</td>
</tr>
<tr>
<td>Lisa I. Kern</td>
<td>125</td>
</tr>
<tr>
<td>Ryan Blu</td>
<td>97</td>
</tr>
<tr>
<td>Denise Napel</td>
<td>81</td>
</tr>
<tr>
<td>Alex Laverton</td>
<td>30</td>
</tr>
<tr>
<td>Julien La Dam</td>
<td>27</td>
</tr>
</tbody>
</table>
Reviewing Market Trends

Conferences and meetups
Reviewing Market Trends

- Also:
  - Community Interest
  - Email Lists and Forums
  - Contributors
  - Follow the Money $$$
Evaluating the Risks

- Risk Tolerance
- Stress Tolerance
- Leader vs follower
Future Proofing

- Assume Change
- Interface Design
- Producer & Consumer Experience
Assume Change

- Remember the Logic and Physical
- Think Logical and Implementation
Interface Design

- Standards
- SQL
- DataFrames / DataSets
- REST, GRPC
- AVRO, Parquet, Protobuf, Thrift, JSON, CSV
Building Successful Teams

Lessons Learned Building Big Data Teams
Build well rounded teams

- Sysadmins
- Developers
- Analysts
- Data Scientists

Other roles:
- Data Protection Officer
- Product Managers
- Network/Systems Engineers
- SRE
How to find people?

Start with people you already have, but make sure you invest in training...

- Linux, network, DBAs → sysadmins
- Developers → developers
  - Easy if you’re at a company like Orbitz, otherwise maybe not so much
- Analysts → analysts
- It’s not an easy path though
  - Set goals instead of micro-managing development
  - Be prepared to iterate, don’t be afraid to fail
Also don’t forget other teams

Communication is key

DBAs

Other Project Teams
Hi Jonathan,

If you are interested in the following Hadoop Data Scientist/Administrator position in downtown Chicago, please email your resume and salary requirements to me.

- Develop and extend in-house data toolkits based in Python and Java.
- Consult and educate internal users on Hadoop technologies.
- Improve the performance of financial analytics platforms built around the Hadoop ecosystem.

**Qualifications:**

- 3+ years of experience working with Hadoop 2 (YARN), cluster management experience preferable
- 3+ year of experience with Hadoop SQL interfaces including Hive and Impala
- 2+ years of experience developing solutions using Spark
- Strong systems background, preferably including Linux administration
- Unix scripting experience (bash, tcsh, zsh, python, etc)
- Experience with DevOps tools such as SALT and Puppet as part of a CI/CD development and deployment process.
- Demonstrated ability to troubleshoot and conduct root-cause analysis
Data Scientists

Admins
Differing Skill Sets

- Detail-Oriented
- Experimental
- The Communicator
- ...
Think beyond just skills

- Also look for complementary personalities
- And avoid toxic personalities
  - But what if they’re really talented?
  - See above.
Customer Engagement

- Your teams should work closely with your customers, whether they’re external or internal
Managing Project Risk
Managing Risk

1 in 11.5 million

1 in 4292

Shark photo: http://www.travelbag.co.uk/
Managing Risk – Risk Categories

Technology Risk

Team Risk

Requirements Risk
Managing Risk – Categorizing Risk
Managing Risk – Categorizing Risk
Managing Risk – Categorizing Risk
Risk Weighting

- Technology Risk
  - How much experience do we have with this technology?
  - Do we have production experience with the technology?
  - We know SQL, but what about Cassandra CQL?
  - ...

Risk Weighting

- Team Risk
  - Experience level of team members
  - Team skill sets
  - Size of team
  - ...

- Don’t forget about other teams
  - System dependencies
  - ...

Risk Weighting

- Requirements Risk
  - Vaguely defined requirements
  - Novel requirements (e.g. stringent latency requirements)
Managing Risk – Categorizing Risk

- Cassandra
  - Limited technical experience (team risk)
  - Need to validate data model (reqs risk)
  - Stringent uptime requirements (tech risk)
Mitigating Risk

- Requirements Risk
  - Ensure good functional requirements
  - Break requirements up – don’t boil the ocean
  - Share requirements and get buy-in from all stakeholders
  - Get agreement on scope
Mitigating Risk

- Technology Risk
  - Tackle important/complex components first
  - Use external resources to help fill knowledge gaps
  - Consider replacing riskier technologies with more familiar ones
Mitigating Risk

- Technology Risk
  - Use proofs of concept
  - Than throw them away
Mitigating Risk

- Technology Risk
  - Use abstractions to minimize dependencies
  - Ensure repeatable build, deployment, monitoring processes
Mitigating Risk

- Technology Risk
  - Start building early
Mitigating Risk

- Team Risk
  - Build well rounded teams
  - Ensure communication with other teams
    - But work to reduce coupling
Communicating Risk

- Make sure stakeholders are aware of risks
  - But remember there can be risks to overstating risk
- Collaborate and get buy-in
- Share risk
- Risk can be a negotiation tool
Ensuring Data Integrity
Ensuring Data Integrity

- Pre-defined vs Derived via Discovery
- Path of Fidelity
- Validation of Quality
Pre-Defined vs Derived via Discovery

- Producer - Productivity vs Audit
- Consumer - Consistency
Producer - Productivity vs Audit

- Red Tape Predefined
- Flexible/Limited Predefined
- After the Fact Discovery

Easy to Audit > Short - Term Productivity
Predefined Traps

- Centralized Reviewing Org
- High bar to on board
- Unclear schema evolution paths
Discovery Traps

- Uncommon output
- Data quality standards
- Uncommon SLAs
- The balloon problem
Consumers Point of View

- Consistency is Key
- Access to Powerful Tools
- Multiple Landing Areas is Key
  - Long Term
  - Indexed
  - Lucene Indexed
  - Streams
- Future Proofing
Path of Fidelity

- What is Fidelity
- What can we mutate
What is Full Fidelity

- The cells and their values are preserved
- Field names and definitions are preserved
- No matter where or how you access the data
- No Filtering
- No Irreversible Mutations
What can we mutate

- Tokenization
- Underlining files
- Storage system
- Access Path
Validate Quality

- Validation of Fidelity
- Validation of Quality
Validation of Fidelity

- Row Counts
- Check Sums
- Reversible byte by byte check
Validation of Quality

- Column level rules
- Null counts
- Field carnality
- Record counts
Metadata Management

▪ What do we mean?
  - Understanding what data you have
  - Knowing what the data is
  - Knowing where the data is

▪ This is complex
  - Large number of data sources, storage systems, processing…
  - Ease of data access and creation of new data sets
  - Start planning at the beginning of your project!
Why Do We Care?

- Visibility – know what data you collect and how to access it
  - Faster time to market
  - Avoid duplication of work
  - Derive more value from data
  - Identify gaps
Why Do We Care?

- Relationships
Why Do We Care?

Regulations

GDPR, etc.
Types of Metadata

- Data at rest
- Data in motion
- Source data
- Data processing
- Reports, dashboards, etc.
Data At Rest

- Files, database tables, Lucene indexes, etc.
Data At Rest – Database Table Example

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>User_id</td>
<td>Long</td>
</tr>
<tr>
<td>Receipt_num</td>
<td>Long</td>
</tr>
<tr>
<td>Item_purchased_id</td>
<td>Long</td>
</tr>
<tr>
<td>Amount</td>
<td>Decimal(7,2)</td>
</tr>
<tr>
<td>Timestamp</td>
<td>Timestamp</td>
</tr>
<tr>
<td>Method</td>
<td>String</td>
</tr>
<tr>
<td>Card_id</td>
<td>Long</td>
</tr>
<tr>
<td>Purchased_port</td>
<td>String</td>
</tr>
</tbody>
</table>
Data At Rest – Other Metadata Types

Audit Logs
Data At Rest – Other Metadata Types

Comments
Data At Rest – Other Metadata Types

Tags
Data At Rest – Other Metadata Types

Lineage

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>User_id</td>
<td>Long</td>
</tr>
<tr>
<td>Receipt_num</td>
<td>Long</td>
</tr>
<tr>
<td>Item_purchased_id</td>
<td>Long</td>
</tr>
<tr>
<td>Amount</td>
<td>Decimal(7,2)</td>
</tr>
<tr>
<td>Timestamp</td>
<td>Timestamp</td>
</tr>
<tr>
<td>Method</td>
<td>String</td>
</tr>
<tr>
<td>Card_id</td>
<td>Long</td>
</tr>
<tr>
<td>Purchased_port</td>
<td>String</td>
</tr>
</tbody>
</table>
Data In Motion

- This is data that’s moving through the system
  - Batch or streaming ingestion
  - Data processing
  - Derived data
Data in Motion – What to Capture

- Paths
- Sources
- Transformations
- Destinations
- Reports/Dashboards
Data in Motion – Paths

- How does the data move through the system?
  - Source systems
  - Data collection systems
  - Routing
  - Transformations
  - Etc.
Source Data

- External systems
- Internal systems
Data In Motion – Transformations

- Data format changes, for example JSON to protocol buffers
- Data fidelity – is the data filtered or changed?
- Metadata about processing – job names, technologies, inputs, outputs, etc.
Data Processing – Machine Learning

- More complex algorithms can require special considerations
  - Purpose of a model
  - Technologies, algorithms, etc.
  - Features
  - Datasets – training, test, etc.
  - Goals of the model
  - Who owns the model?
Reports and Dashboards

- Data sources
- Any data transformations
- Information on the report’s creator
- Log of modifications
- Purpose of report
- Tags
Approaches to Metadata Collection

▪ Declarative
  - Require and enable metadata to be created as data is added to the system
▪ Discovery
  - After the fact cataloging of data
How?

- Create your own solution
- Use tools provided by your vendor
- Use third party tools
Thank you!

Ted Malaska | @ted_malaska
Jonathan Seidman | @jseidman