Disciplined Agile

THE AGILE DATABASE TECHNIQUES STACK: FUNDAMENTAL SKILLS FOR DATABASE DEVELOPERS

Scott Ambler

Vice President & Chief Scientist, Disciplined Agile

Project Management Institute

@scottwambler

Ownership Notices

Disciplined Agile™

Disciplined Agile ® (DA) is owned by Project Management Institute (PMI) PMI.org/disciplined-agile

Agile Data

Agile Data is owned by Ambysoft Inc. AgileData.org

Scott Ambler

Co-creator of PMI’s Disciplined Agile tool kit
VP & Chief Scientist, Disciplined Agile at PMI

Thought Leader, Agile Data and Agile Modeling

Twitter: @scottwambler
linkedin.com/in/sambler/

Agenda

- Our data quality challenge: Data technical debt
- Ways of thinking (WoT) for agile data development
- Ways of working (WoW): Agile database techniques stack
- Adoption strategies
- Learn more

Our Data Quality Challenge: Data Technical Debt

What is Data Technical Debt?

Technical debt is the accumulation of defects, quality issues (such as difficult to read code or low data quality), poor architecture, and poor design in existing solutions

Data technical debt refers to quality challenges associated with legacy data sources, including both mission-critical sources of record as well as "big data" sources of insight.

Source: http://agiledata.org/essays/dataTechnicalDebt.html

Types of Data Technical Debt

1. **Structural**. The design of a table, column or view.
2. **Data quality**. The consistency or usage of data values.
3. **Referential integrity**. Does a referenced row exist within another table? Is a row which is no longer needed is (soft) deleted appropriately?
4. **Architectural**. How external programs interact with a data source.
5. **Documentation**. Quality issue with any supporting documents, including models.
6. **Method/functional**. The execution aspects within a data source, such as stored procedures, stored functions, and triggers.

Source: AgileData.org/essays/dataTechnicalDebt.html

Agile
Data

PMI 2022 Data Quality Survey

April 11 to May 22, 2022

66 Responses

95% believe data is a corporate asset

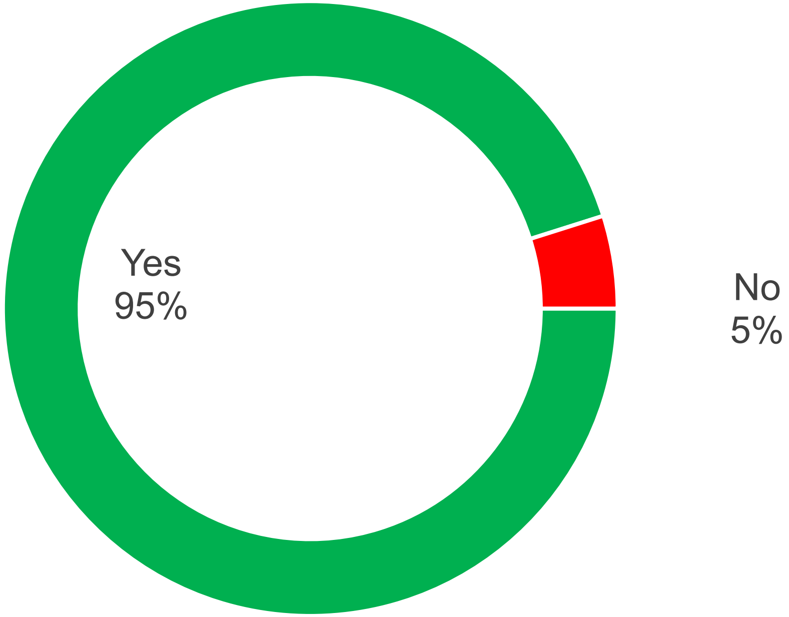
54% of organizations measure data quality

42% believe that the most recent data source they used was high quality

Source: ProjectManagement.com/blog-post/72688/data-technical-debt--2022-data-quality-survey-results

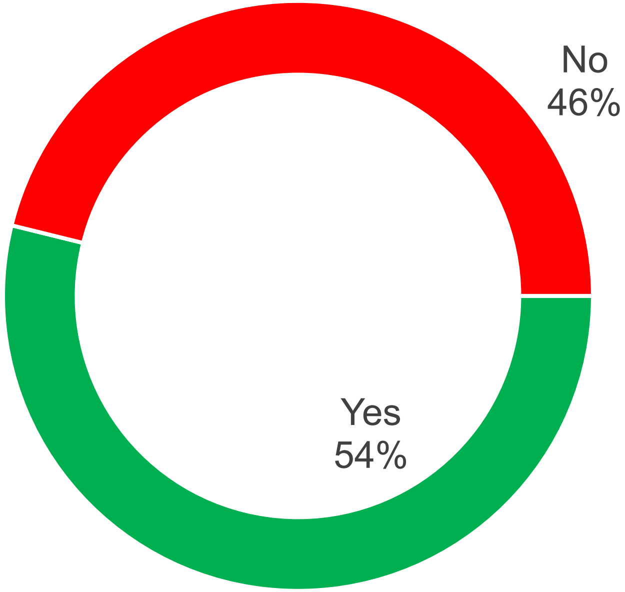
How important is data to your organization?

Does your organization consider data to be an important asset?



BUT…

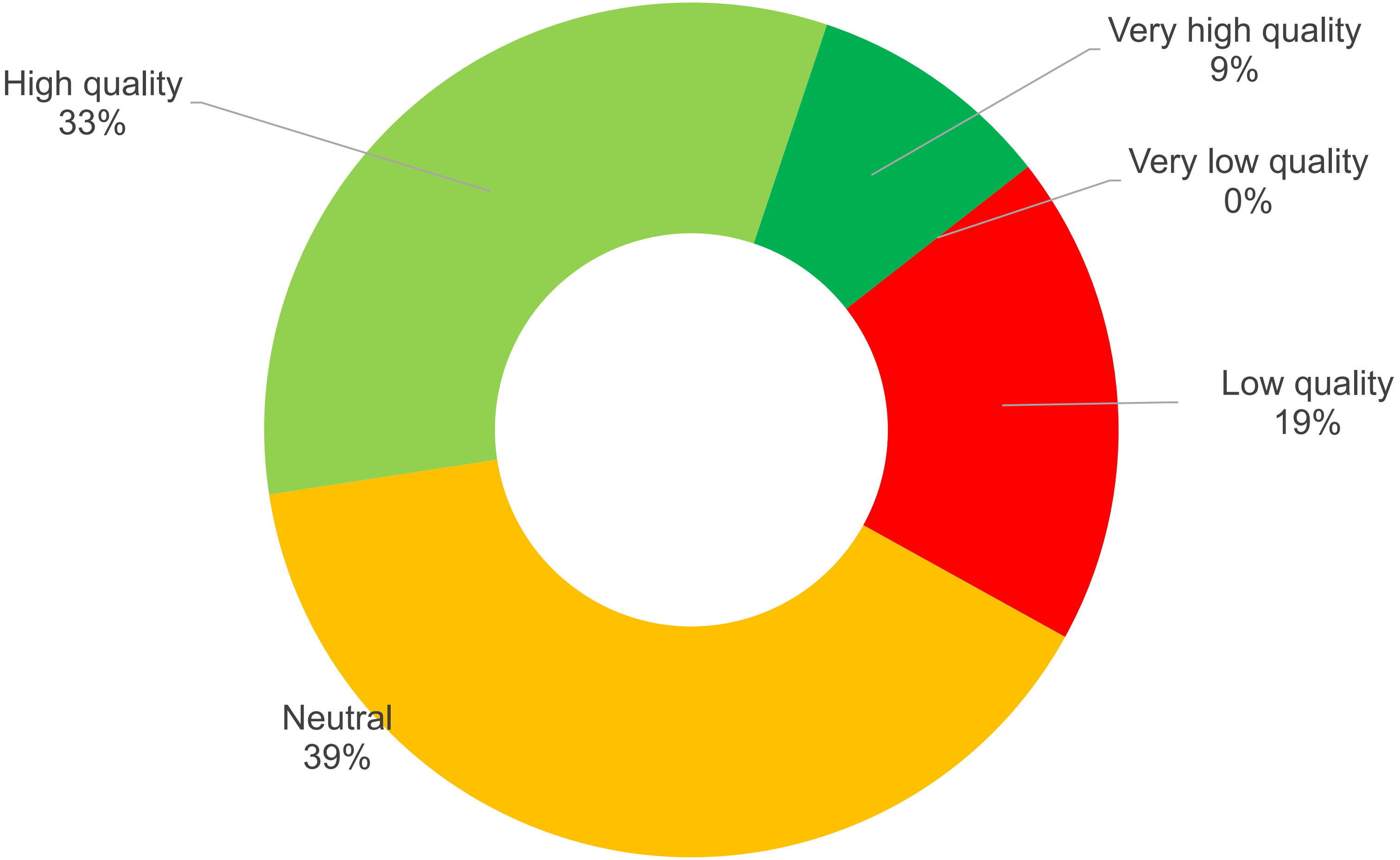
Does your organization measure the quality of your data sources?



Source: PMI's 2022 Data Quality Survey

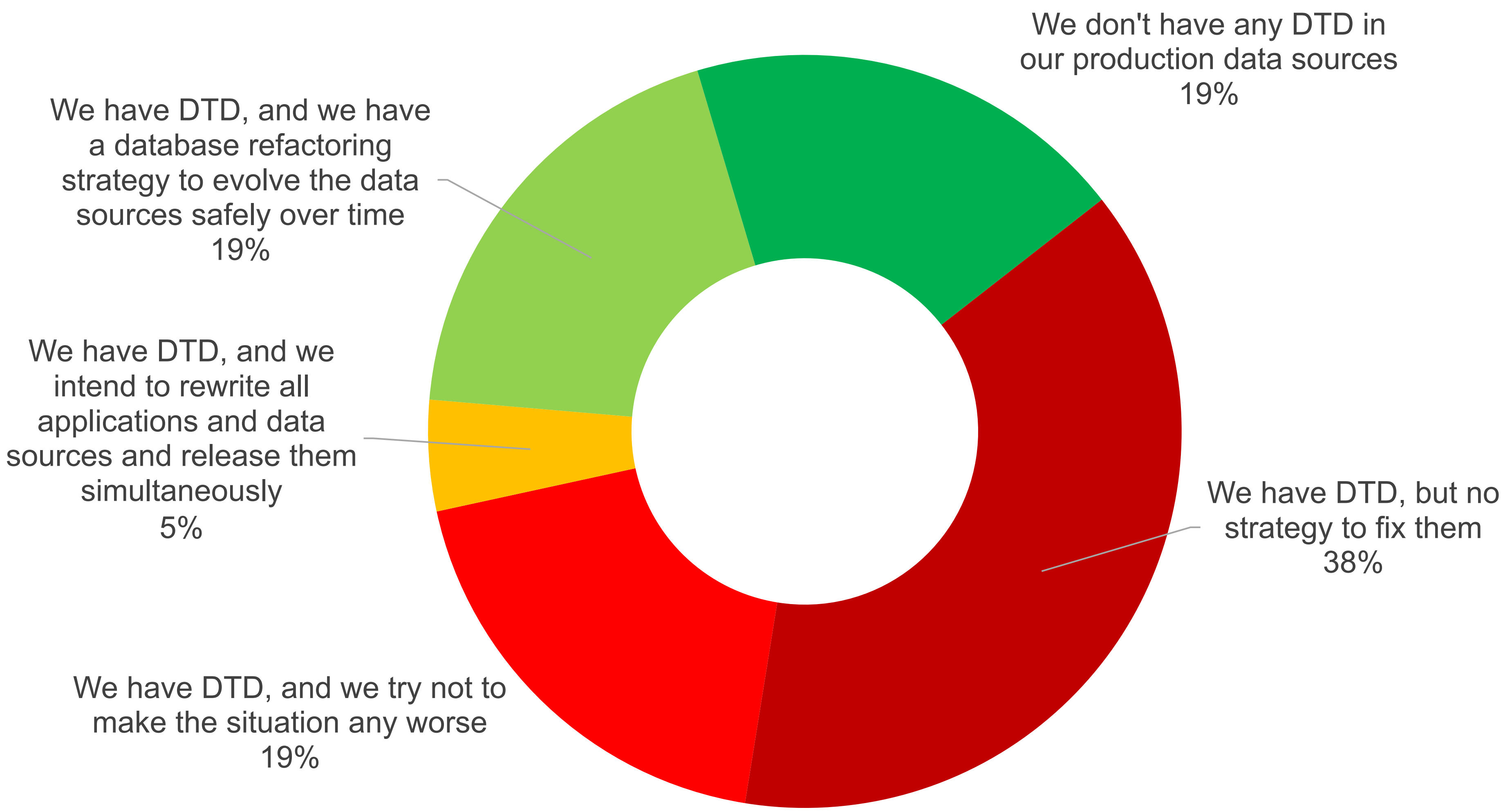
Disciplined Agile

Consider the most recent production data source (such as a database or data file), that is owned by your organization, that you have worked with. How would you rate the general quality of the data within that data source?



Source: PMI’s 2022 Data Quality Survey

What strategy, if any, does your organization have for addressing any existing data technical debt (DTD) with your production data sources?



Source: PMI's 2022 Data Quality Survey

Disciplined Agile

Ways of Thinking (WoT) for Agile Data Development

The Philosophies of the Agile Data (AD) Method

The mindset of the AD method is captured as a collection of six philosophies:

1. Beyond data
2. Enterprise awareness
3. Agile enterprise groups
4. Unique context
5. Teamwork
6. Sweet spot

Source: AgileData.org/essays/philosophies.html

Agile Data

The Disciplined Agile Mindset

The Way of Thinking for Enterprise Agility

Source: PMI.org/disciplined-agile/mindset

Principles

We believe in these principles:

- Delight customers
- Be awesome
- Context counts
- Be pragmatic
- Choice is good
- Optimize flow
- Organize around products/services
- Enterprise awareness

Promises

So, we promise to:

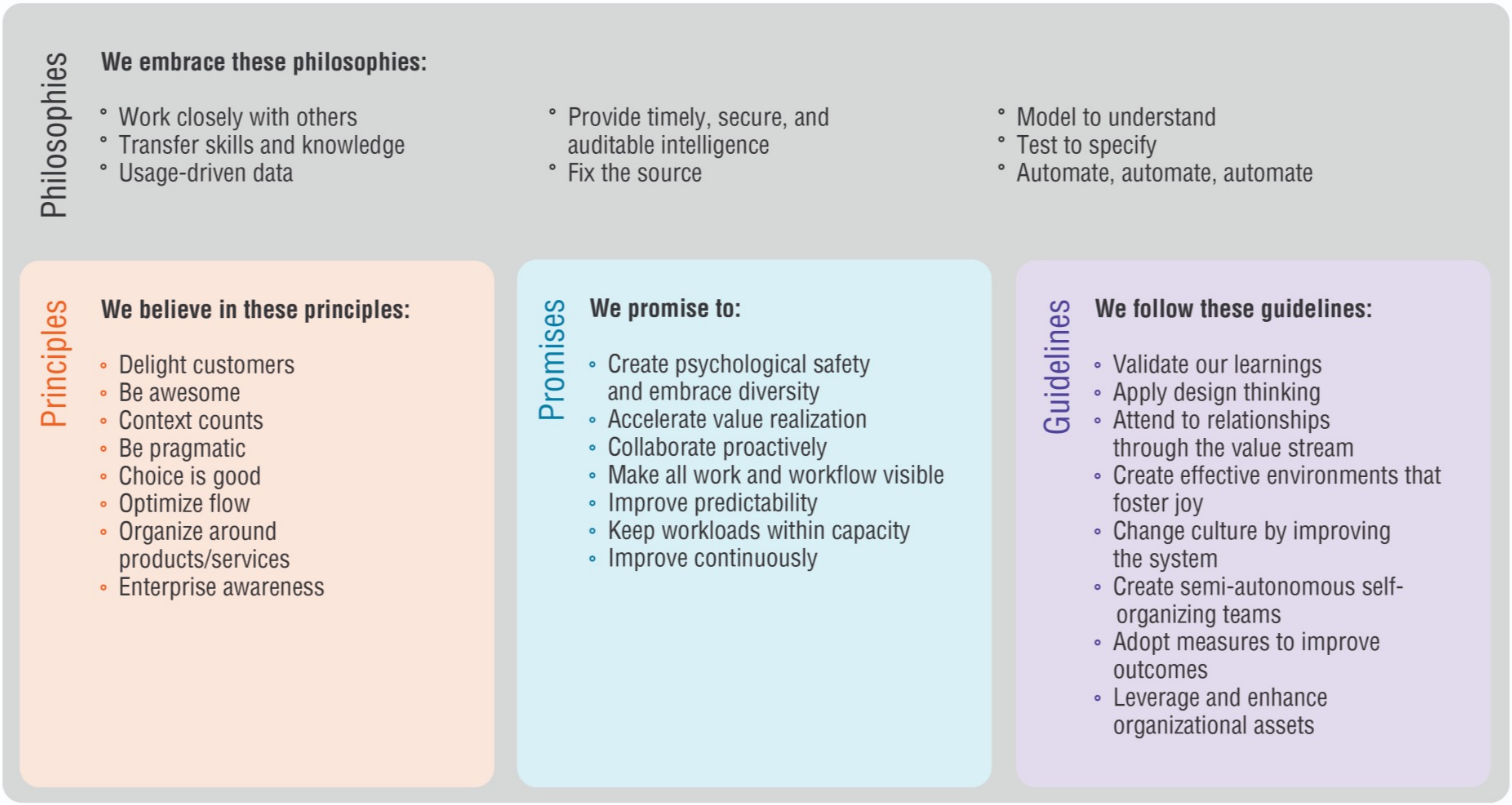
- Create psychological safety and embrace diversity
- Accelerate value realization
- Collaborate proactively
- Make all work and workflow visible
- Improve predictability
- Keep workloads within capacity
- Improve continuously

Guidelines

And follow these guidelines:

- Validate our learning
- Apply design thinking
- Attend to relationships throughout the value stream
- Create effective environments that foster joy
- Change culture by improving the system
- Create semi-autonomous, self-organizing teams
- Adopt measures to improve outcomes
- Leverage and enhance organizational assets

A Disciplined Agile Mindset for Data Management



Philosophies

We embrace these philosophies:

- Work closely with others
- Transfer skills and knowledge
- Usage-driven data
- Provide timely, secure, and auditable intelligence
- Fix the source
- Model to understand
- Test to specify
- Automate, automate, automate

Principles

We believe in these principles:

- Delight customers
- Be awesome
- Context counts
- Be pragmatic
- Choice is good
- Optimize flow
- Organize around products/services
- Enterprise awareness

Promises

We promise to:

- Create psychological safety and embrace diversity
- Accelerate value realization
- Collaborate proactively
- Make all work and workflow visible
- Improve predictability
- Keep workloads within capacity
- Improve continuously

Guidelines

We follow these guidelines:

- Validate our learnings
- Apply design thinking
- Attend to relationships through the value stream
- Create effective environments that foster joy
- Change culture by improving the system
- Create semi-autonomous self-organizing teams
- Adopt measures to improve outcomes
- Leverage and enhance organizational assets

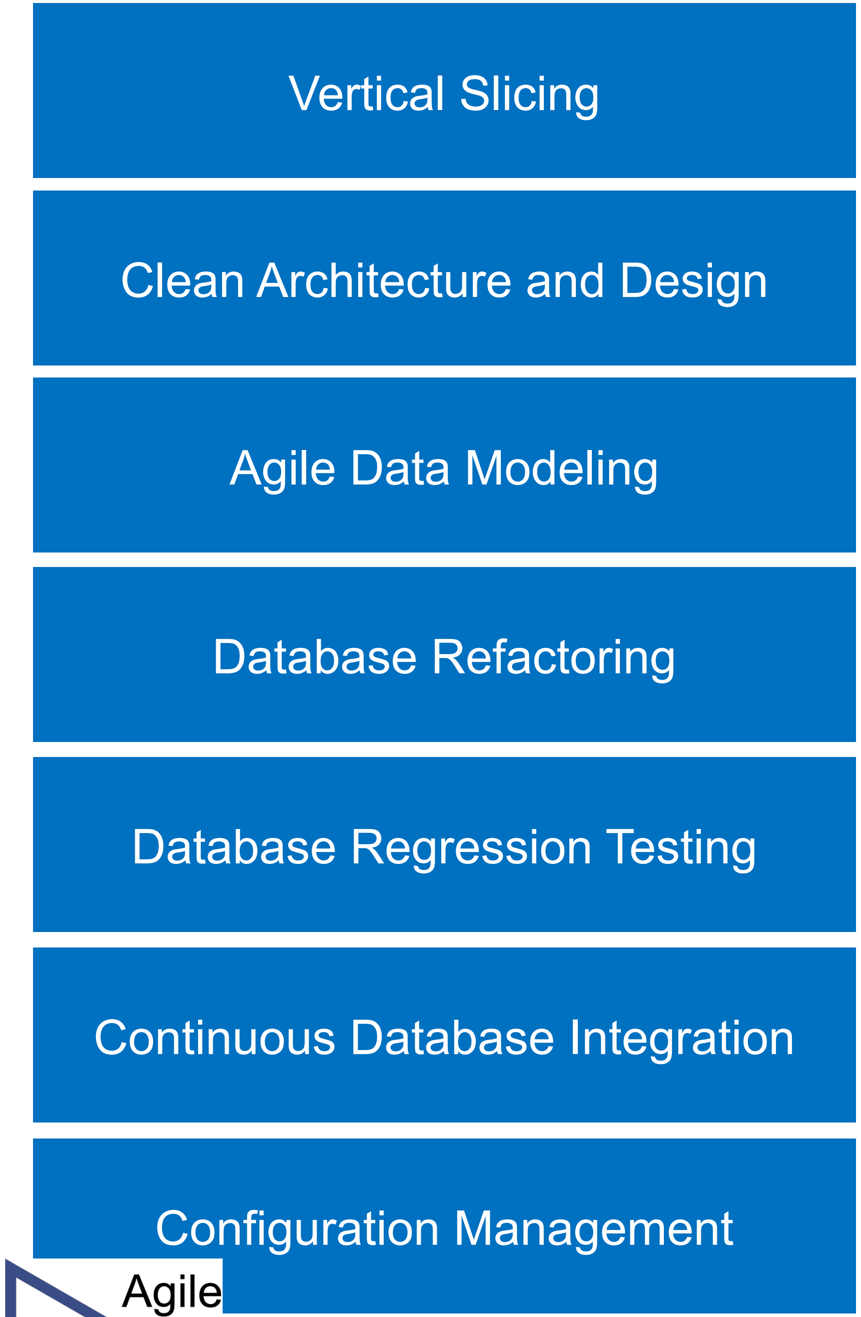
Source: PMI.org/disciplined-agile/process/data-management/data-management-mindset

Audience Discussion

How close to this is the current way of thinking (WoT) within your organization?

Ways of Working (WoW): The Agile Database Techniques Stack

The increasing pace of change, increasing complexity, and increasing volume of data demands nothing less than complete agility.



Vertical Slicing

Clean Architecture and Design

Agile Data Modeling

Database Refactoring

Database Regression Testing

Continuous Database Integration

Configuration Management

Configuration Management

- All work products should be stored in a versioned repository
- Maintains the integrity of the system and all supporting work products as it evolves
- Facilitates change in a controlled fashion
- All revisions kept and who made what changes to all work products
- Potential benefits include the ability to:
  - Manage versions of systems across environments
  - Rollback to previous versions
  - Modify work products in parallel and then merge
  - Find source of defects injected

Source: AgileData.org/essays/configurationManagement.html

- Part of building the system is building the database (if it changed)

- Challenge: Tests SHOULD put the database back into a known state, but sometimes don’t
  - You will want to rebuild the (non-production) database from scratch every so often

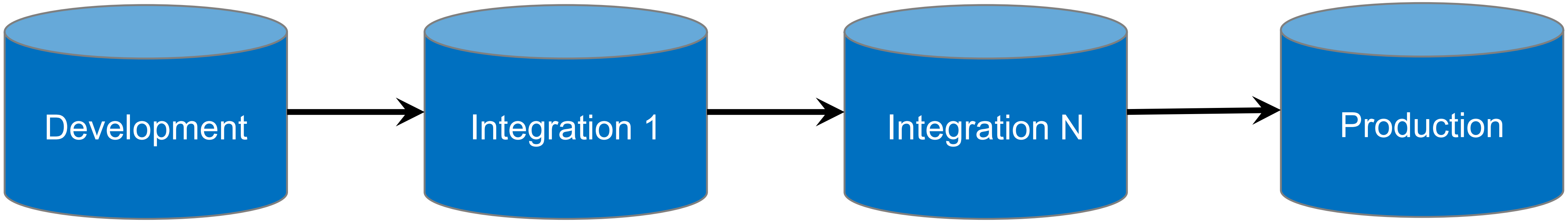
- Challenge: Database accesses take time
  - Some test suites will test against DB mocks
  - You still need to test the actual database occasionally

- Source: AgileData.org/essays/continuousIntegration.html

Continuous Database Integration

Configuration Management

Continuous Deployment between Sandboxes



- Each integration DB must know it's version, so as to support database refactoring
- CDI differs between sandboxes
  - Development sandboxes often implement database mocks
  - Integration sandboxes should test against the database
  - Production has limited tests and checks for performance reasons
- Continuous database deployment should arguably be called out as part of CDI (expect an update later this year)

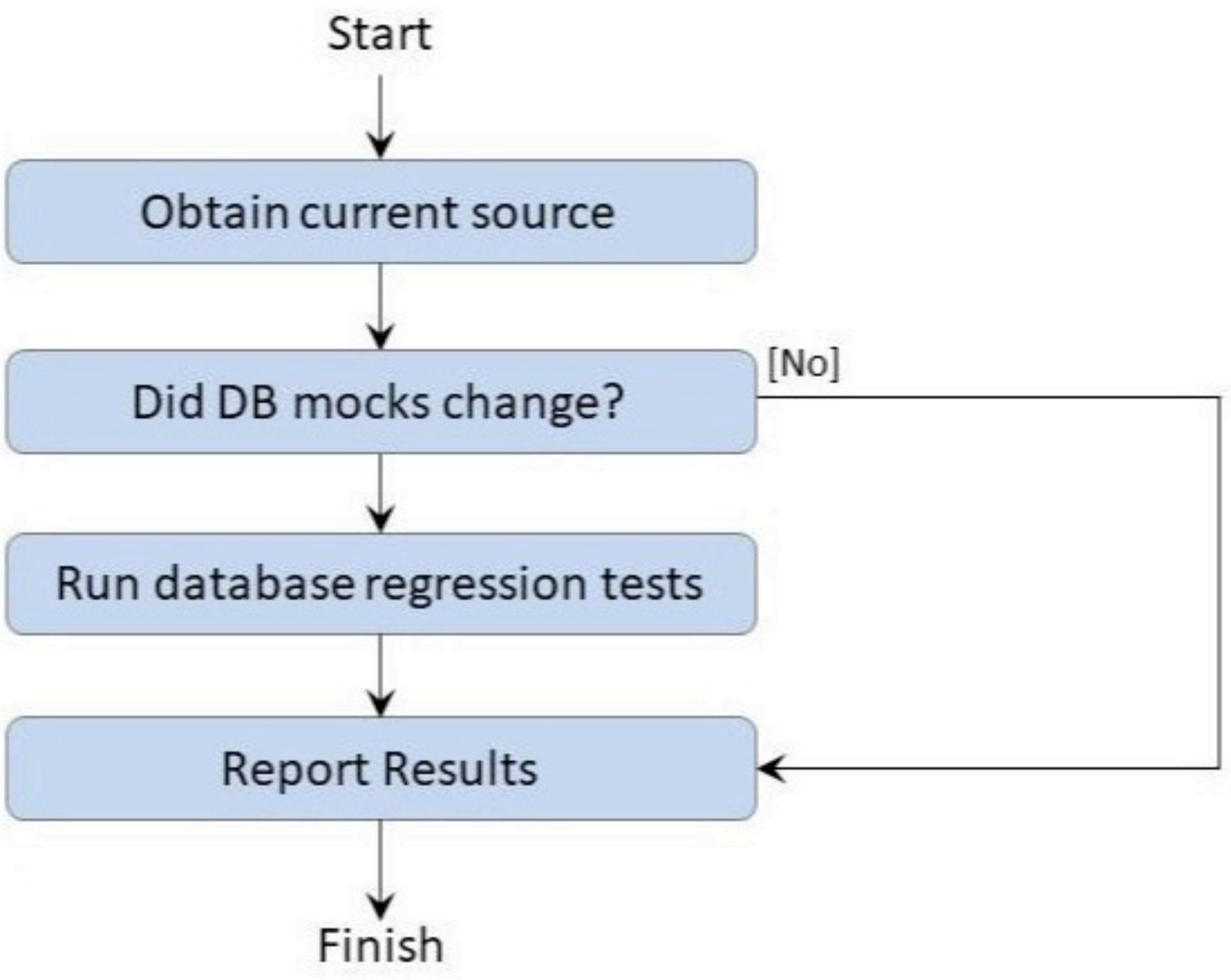
Continuous Database Integration

Configuration Management

Source: AgileData.org/essays/sandboxes.html

Agile Data

CDI in Development Sandboxes



Start

Obtain current source

Did DB mocks change?

[No]

Run database regression tests

Report Results

Finish

- Source: AgileData.org/essays/continuousIntegration.html

Continuous Database Integration

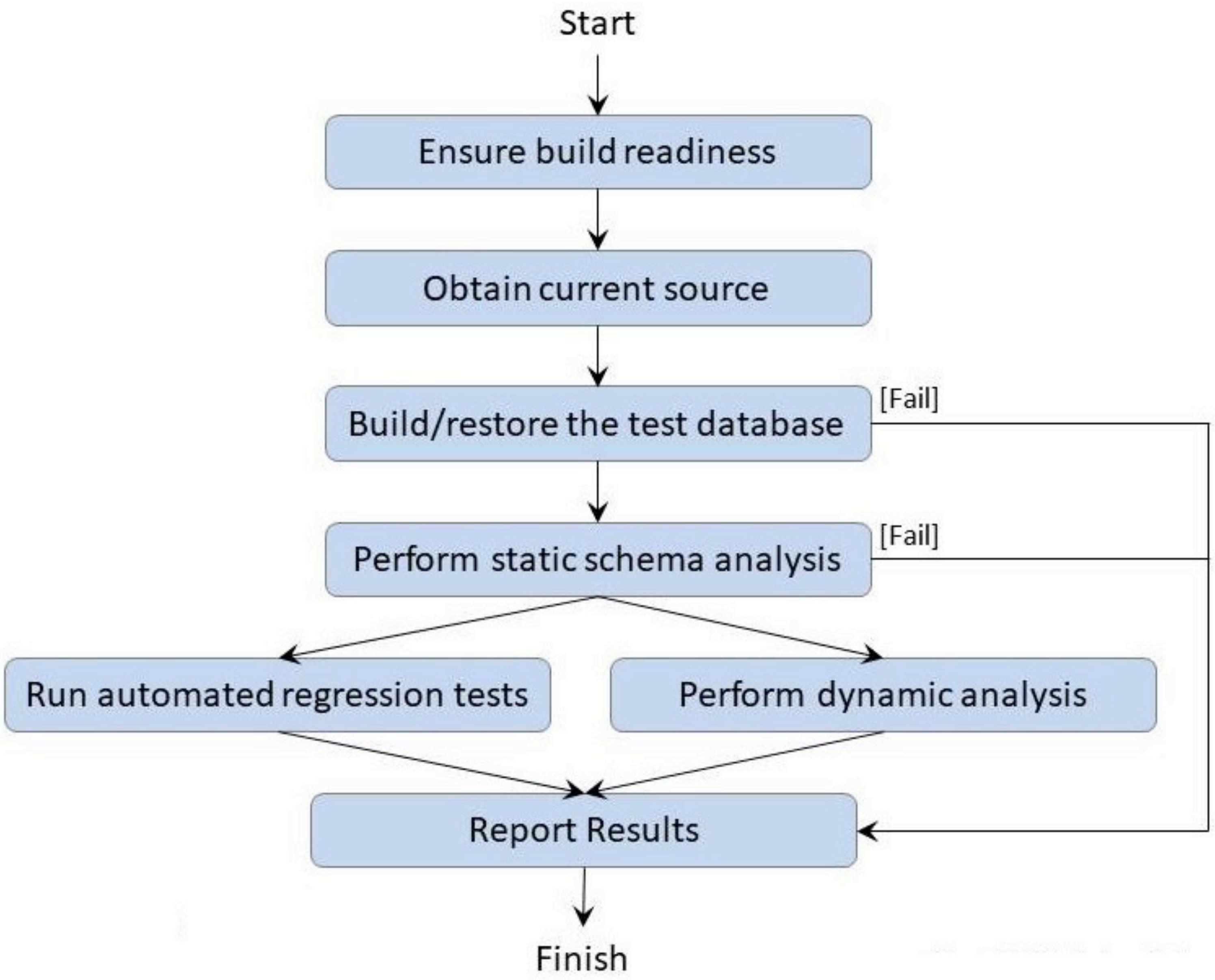
Configuration Management

Agile
Data

CDI in Integration Sandboxes

Continuous Database Integration

Configuration Management



- Source: AgileData.org/essays/continuousIntegration.html

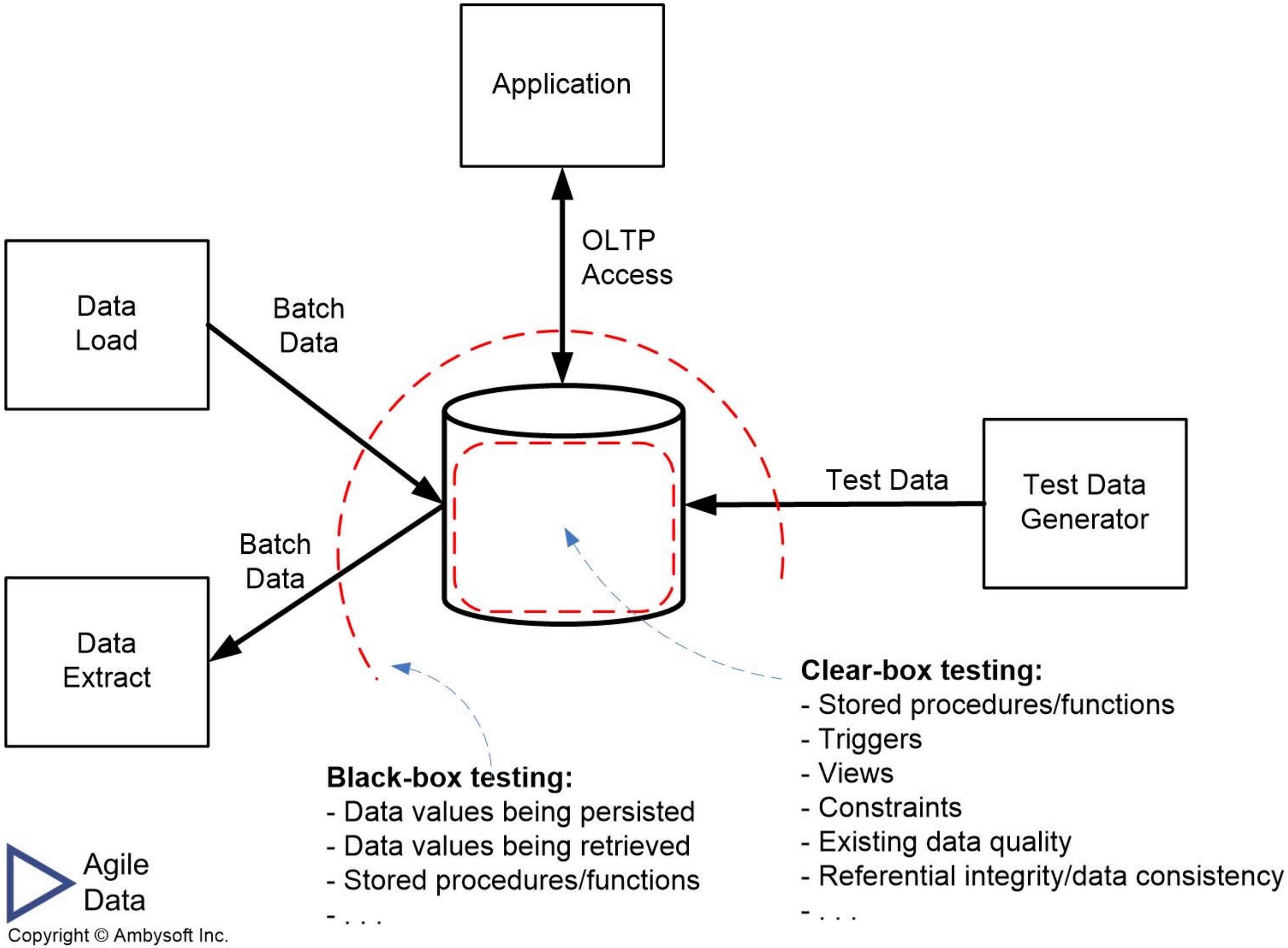
Agile Data

Database Regression Testing

Continuous Database Integration

Configuration Management

Agile Data



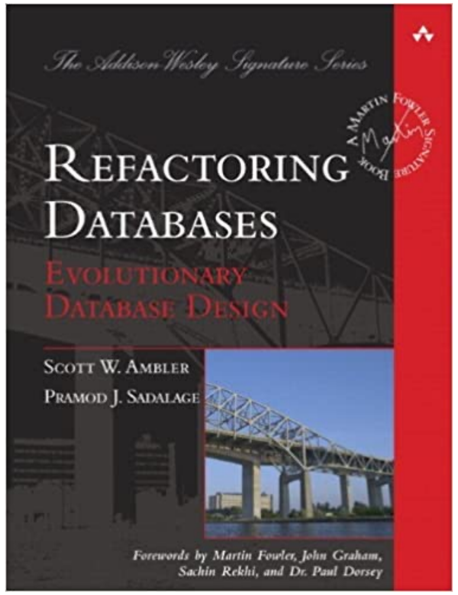
Black-box testing:
- Data values being persisted
- Data values being retrieved
- Stored procedures/functions
- . . .

Clear-box testing:
- Stored procedures/functions
- Triggers
- Views
- Constraints
- Existing data quality
- Referential integrity/data consistency
- . . .

Source: AgileData.org/essays/databaseTesting.html

A database refactoring is a simple change to a database schema that improves its design while retaining both its *behavioral and informational semantics*

A database schema includes structural aspects such as table and view definitions; functional aspects such as stored procedures and triggers; and informational aspects such as the data itself


The Addison Wesley Signature Series
A Martin Fowler Signature Book
Refactoring Databases
Evolutionary Database Design
Scott W. Ambler
Pramod J. Sadalage
Forewords by Martin Fowler, John Graham, Sachin Rekhi, and Dr. Paul Dorsey

Source: AgileData.org/essays/databaseRefactoring.html

Database Refactoring

Database Regression Testing

Continuous Database Integration

Configuration Management

Agile Data

Audience Discussion

What challenges does your organization currently face regarding database refactoring?

Why Evolving Databases is Thought to be Hard

Database Refactoring

Database Regression Testing

Continuous Database Integration

Configuration Management

Production databases are often highly coupled to other systems, services, data sources, …

Source: AgileData.org/essays/databaseRefactoring.html

Agile
Data

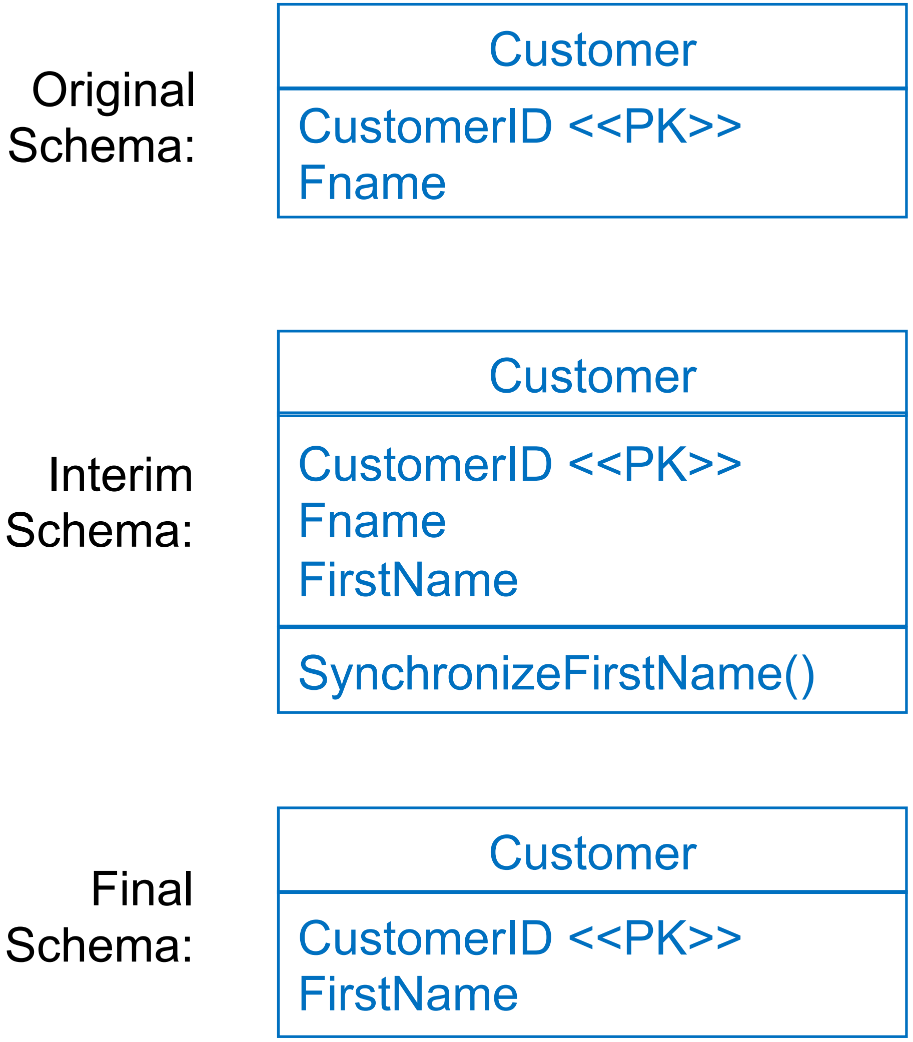
Implementing a Database Refactoring: Rename Column

Database Refactoring

Database Regression Testing

Continuous Database Integration

Configuration Management



Original Schema:

Customer

CustomerID <<PK>>
Fname

Interim Schema:

Customer

CustomerID <<PK>>
Fname
FirstName

SynchronizeFirstName()

Final Schema:

Customer

CustomerID <<PK>>
FirstName

Source: AgileData.org/essays/databaseRefactoring.html

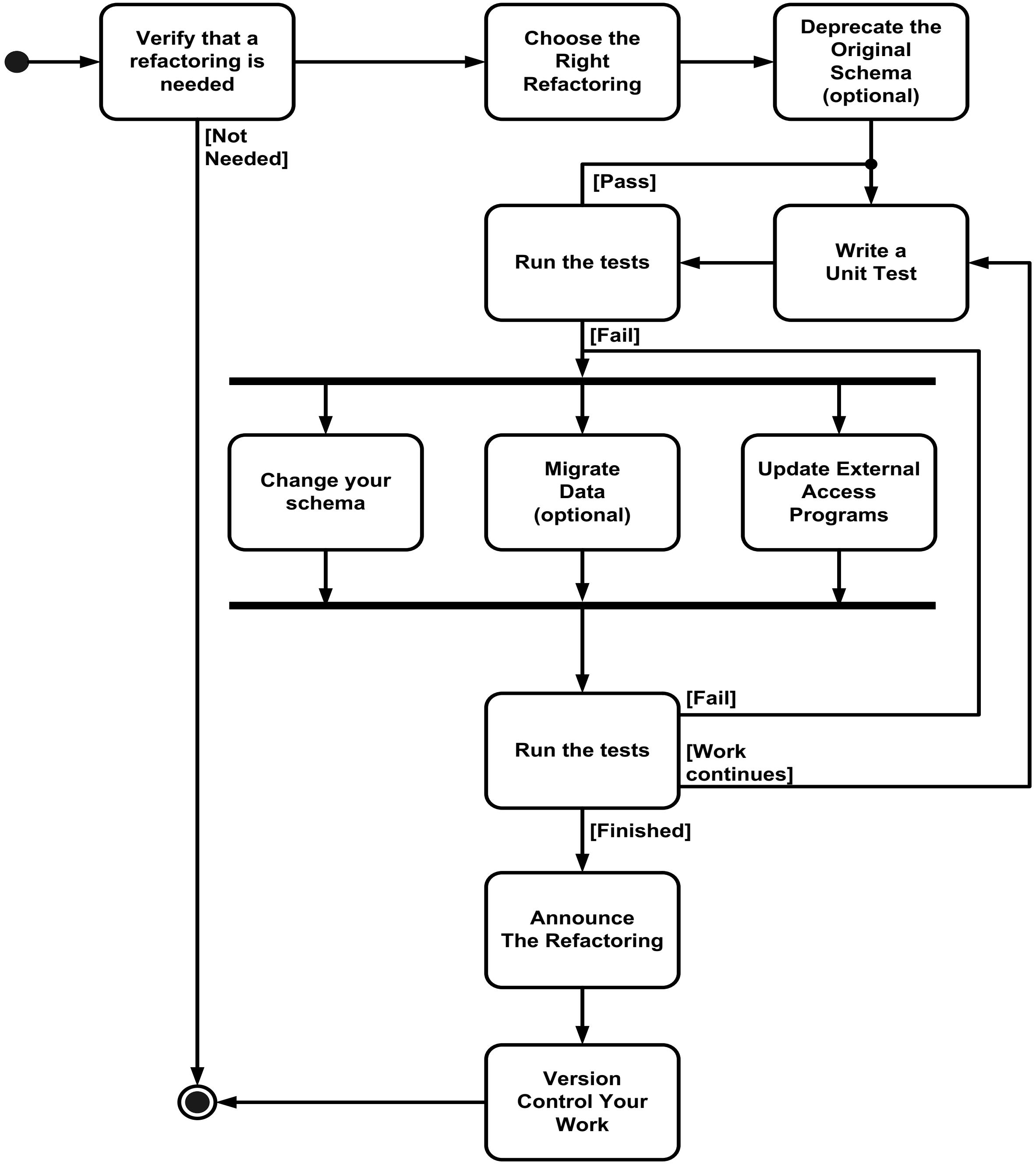
The Process of Database Refactoring

Database Refactoring

Database Regression Testing

Continuous Database Integration

Configuration Management



Source: AgileData.org/essays/databaseRefactoring.html

Agile Data

Agile Data Modeling

Database Refactoring

Database Regression Testing

Continuous Database Integration

Configuration Management

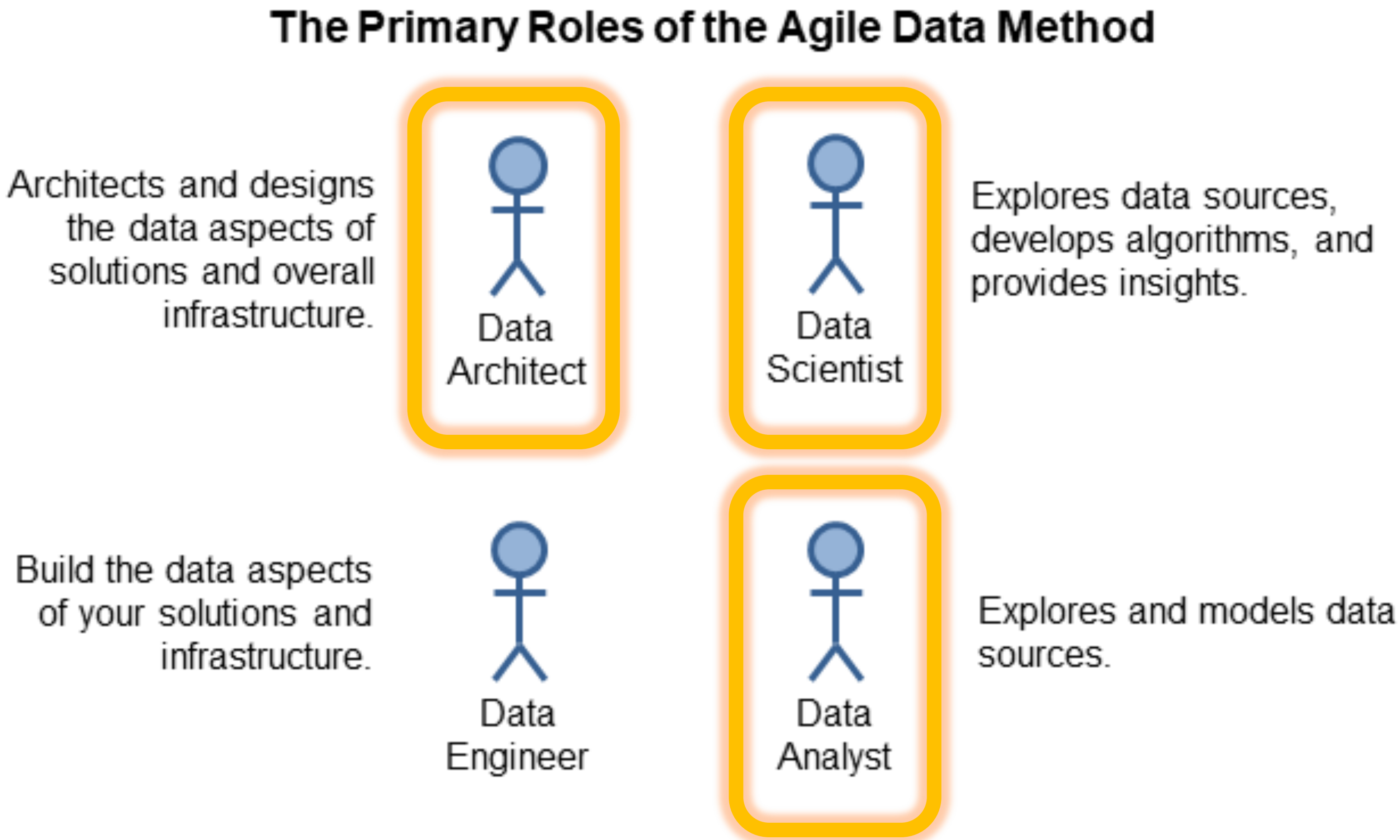
Data modeling is the act of exploring data-oriented structures

Evolutionary data modeling is data modeling performed in an iterative and incremental manner

Agile data modeling is evolutionary data modeling done in a collaborative manner

Source: AgileData.org/essays/agileDataModeling.html

- Agile Data Modeling
- Database Refactoring
- Database Regression Testing
- Continuous Database Integration
- Configuration Management



The Primary Roles of the Agile Data Method

Data Architect: Architects and designs the data aspects of solutions and overall infrastructure.

Data Scientist: Explores data sources, develops algorithms, and provides insights.

Data Engineer: Build the data aspects of your solutions and infrastructure.

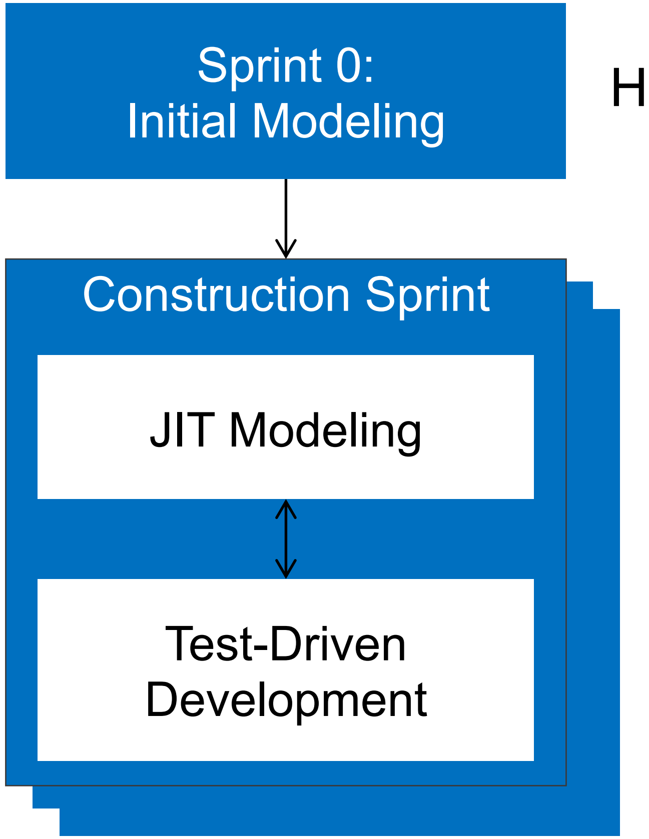
Data Analyst: Explores and models data sources.

Source: AgileData.org/essays/roles.html

Agile Data

Agile Data Modeling Throughout the Lifecycle

- Agile Data Modeling
- Database Refactoring
- Database Regression Testing
- Continuous Database Integration
- Configuration Management



High-level conceptual modeling

Detailed physical data modeling (to generate DDL)

Detailed specification

Source: AgileData.org/essays/agileDataModeling.html

High-Level
Conceptual Model

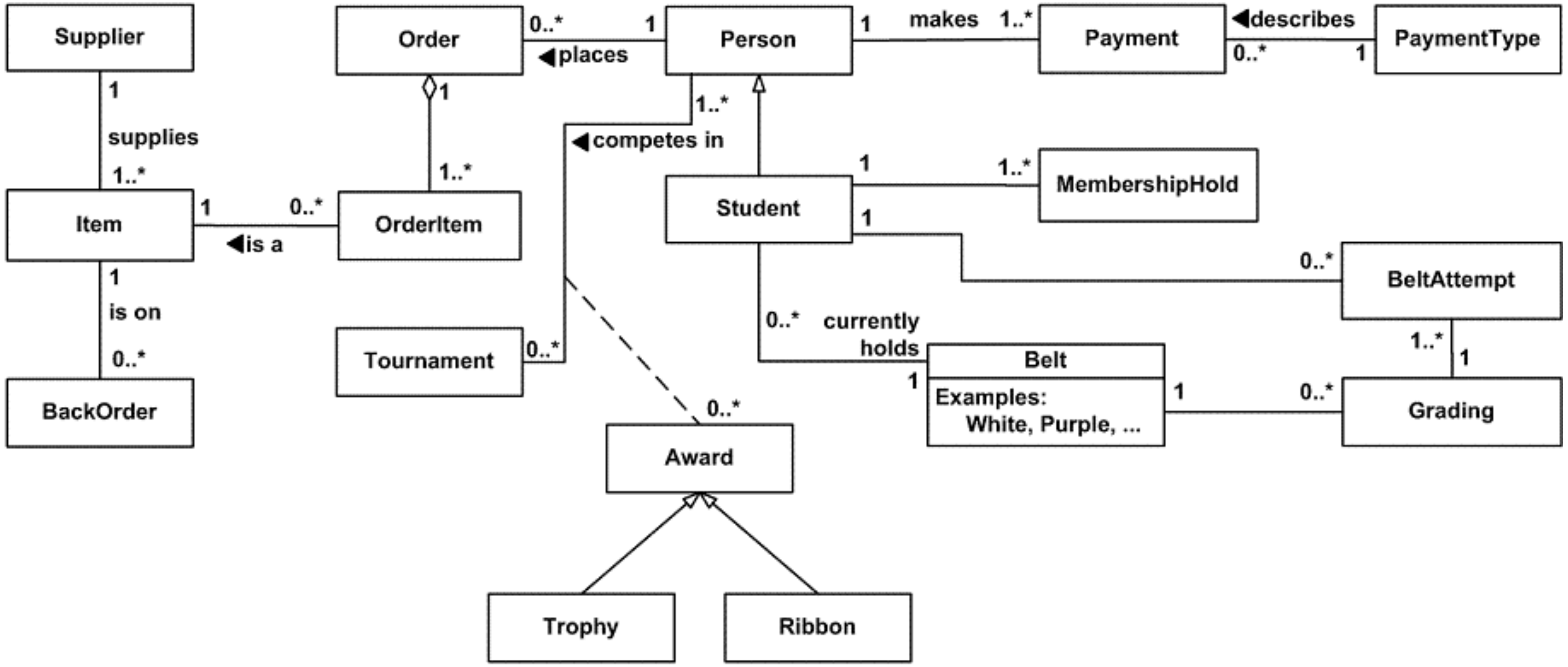
Agile Data Modeling

Database Refactoring

Database Regression Testing

Continuous Database Integration

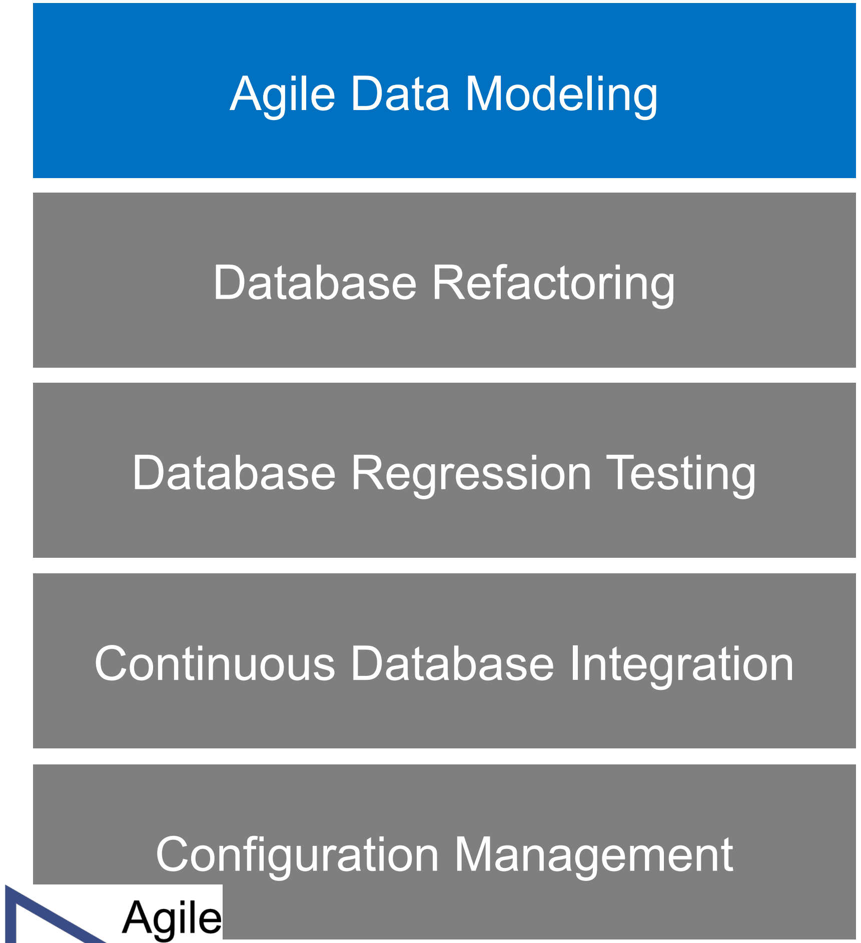
Configuration Management



Source: AgileData.org/essays/agileDataModeling.html

Agile
Data

Question Stories



A question story is a specialized user story specific to data-oriented requirements.

Ex: As a sales manager I want to know the level of sales by my team by the end of each day so that I know where we stand.

Formats:

As a [Role] I want to know [Something] by [Timeframe] because [Reason]

- OR -

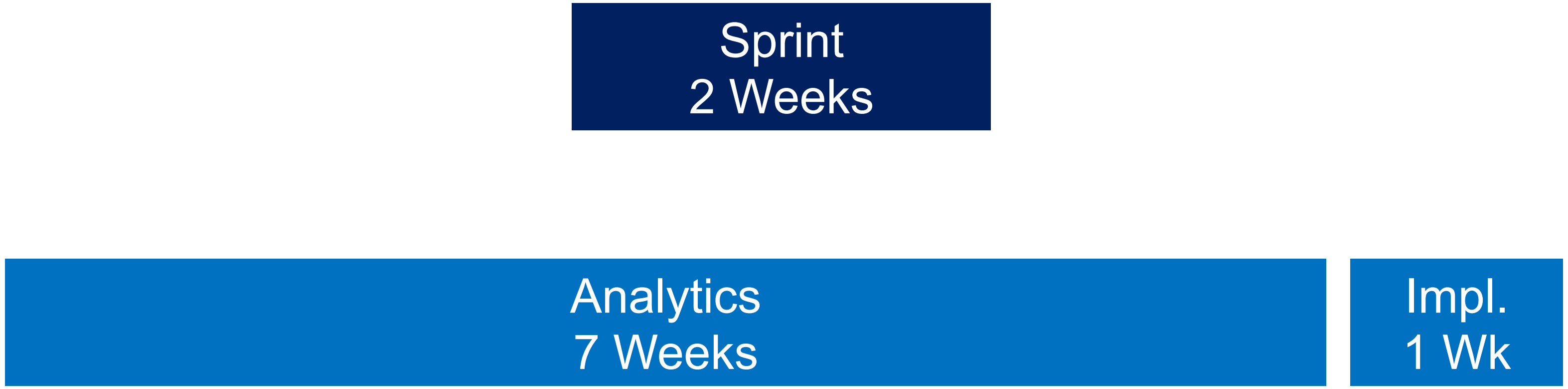
As a [Role] I want to know [Something] because [Reason]

Source: AgileData.org/essays/questionStories.html

Audience Discussion: What Do You Do?

Scenario:

- You’re on an agile team, and Sprints are two weeks long.
- It takes 8 weeks in total to fully develop a question story.
- 7 weeks of analytics work and 1 week of implementation work.

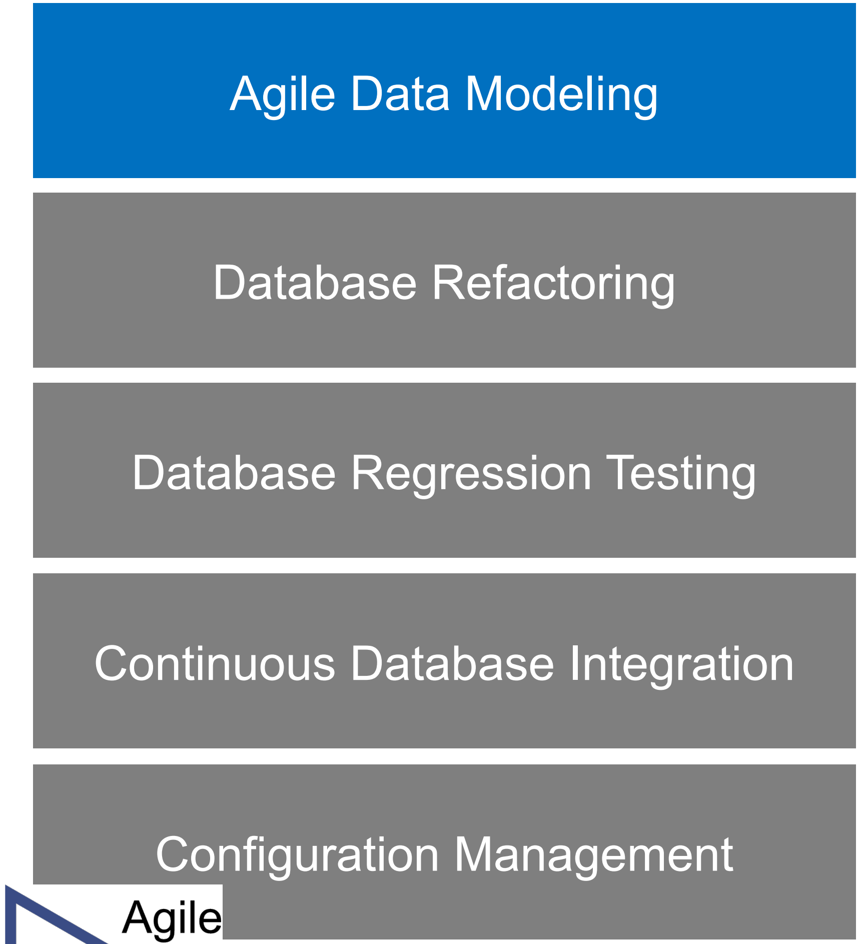


Sprint
2 Weeks

Analytics
7 Weeks

Impl.
1 Wk

Look-Ahead
Data Analysis



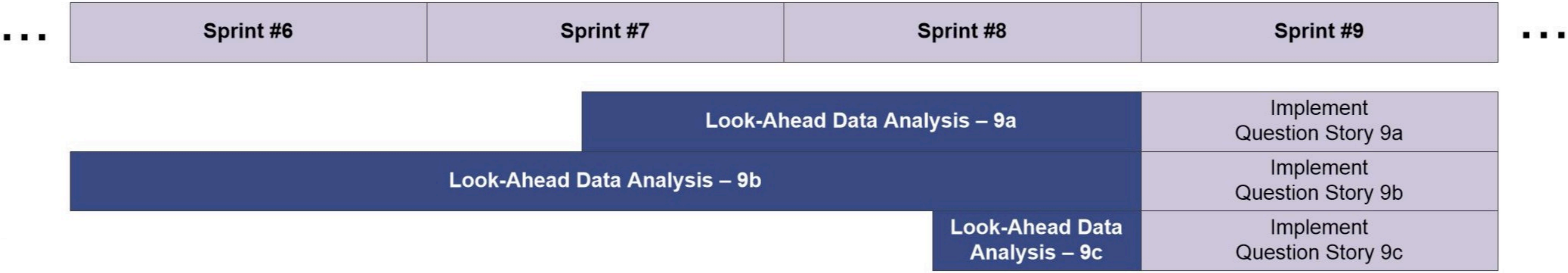
Sometimes it takes several days, even weeks, to perform data analytics before you can implement a question story.

There are several factors that will determine how far ahead you need to perform look-ahead data analysis:

- The complexity of the data source(s).
- Your ability to gain access to the data source(s).
- The difficulty of the question being asked.
- The skill, experience, and knowledge of the data analyst(s).
- The availability of the data analyst(s).
- Your data profiling tools.

Source: AgileData.org/essays/lookAheadDataAnalysis.html

Look-Ahead Data Analysis: Agile



… Sprint #6 | Sprint #7 | Sprint #8 | Sprint #9 …

Look-Ahead Data Analysis – 9a

Implement Question Story 9a

Look-Ahead Data Analysis – 9b

Implement Question Story 9b

Look-Ahead Data Analysis – 9c

Implement Question Story 9c

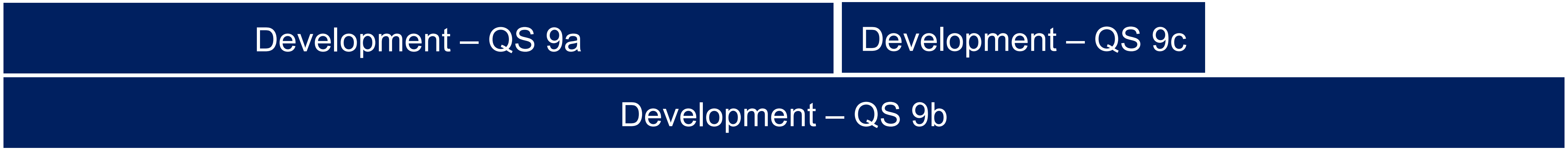
Scenario: You want to implement three question stories in sprint #9

You need to:

- Have a definition of ready (DoR) indicating the amount of data analysis work required
- Guesstimate the amount of data analysis required for each one, and then perform the analysis sufficiently before sprint #9
- Have sufficient capacity to perform look-ahead data analysis
- Interleave data analysis for other sprints into the work of the people performing it

Note: Staffing your team with specialists will exacerbate work scheduling challenges. Consider generalizing specialists (AgileModeling.com/essays/generalizingSpecialists.htm) instead.

Look-Ahead Data Analysis: Continuous Delivery



Development – QS 9a

Development – QS 9c

Development – QS 9b

Scenario: You want to implement the same three question stories

- You are not constrained by organizing the work into sprints
- Development = Data analysis + other implementation work
- Work can be brought into the team as capacity permits
- Value is delivered when it is available
- Average cycle time to deliver stories is shorter
- Staffing your team with generalizing specialists may enable you to swarm around data analysis work and shorten the development time of a question story

Clean Architecture and Design

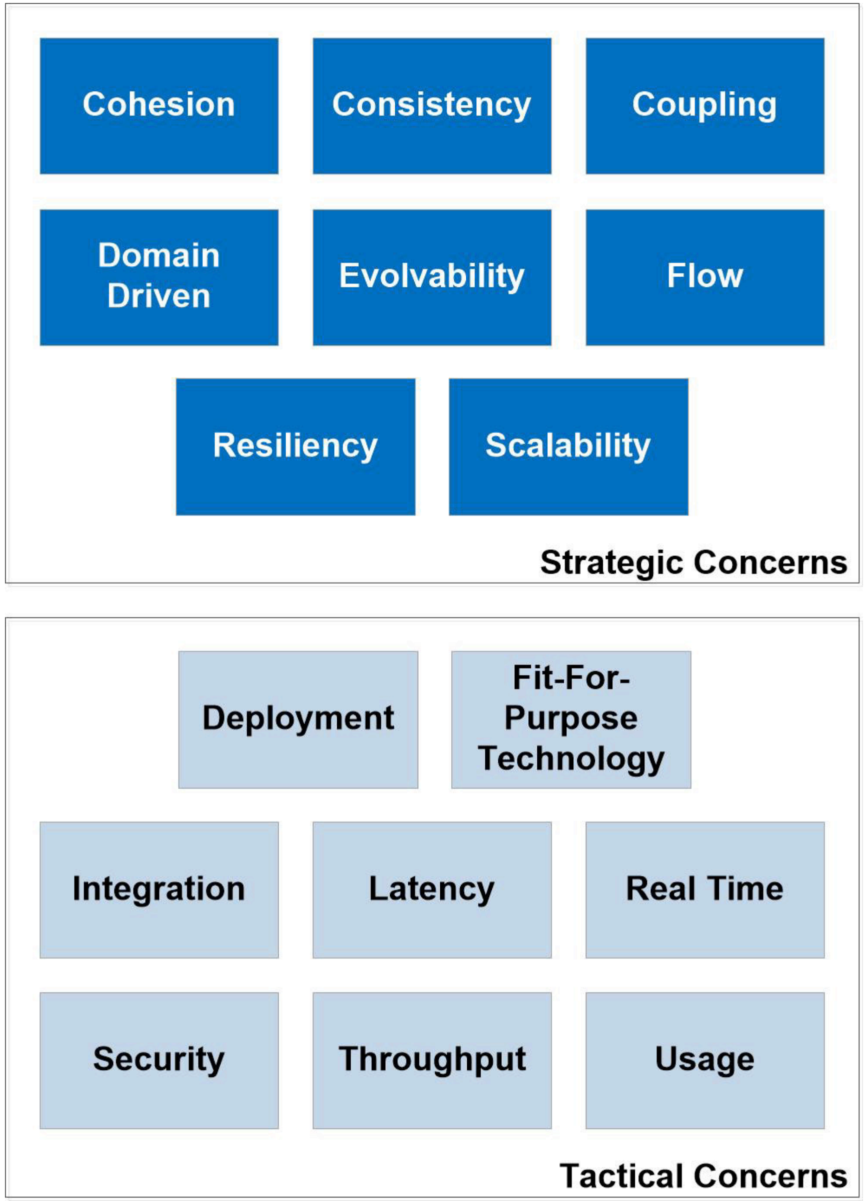
Agile Data Modeling

Database Refactoring

Database Regression Testing

Continuous Database Integration

Configuration Management



Cohesion

Consistency

Coupling

Domain Driven

Evolvability

Flow

Resiliency

Scalability

Strategic Concerns

Deployment

Fit-For-Purpose Technology

Integration

Latency

Real Time

Security

Throughput

Usage

Tactical Concerns

Source: AgileData.org/essays/dataArchitectureConcerns.html

- Clean Architecture and **Design**
- Agile Data Modeling
- Database Refactoring
- Database Regression Testing
- Continuous Database Integration
- Configuration Management

Normalization

- Are you storing data in one and one only place?
- Tables and columns are cohesive
- Tables are loosely coupled

Design for the database type

- OLTP (transactional ➔ normalized design)
- OLAP (analytical ➔ denormalized design)

Technical future proofing

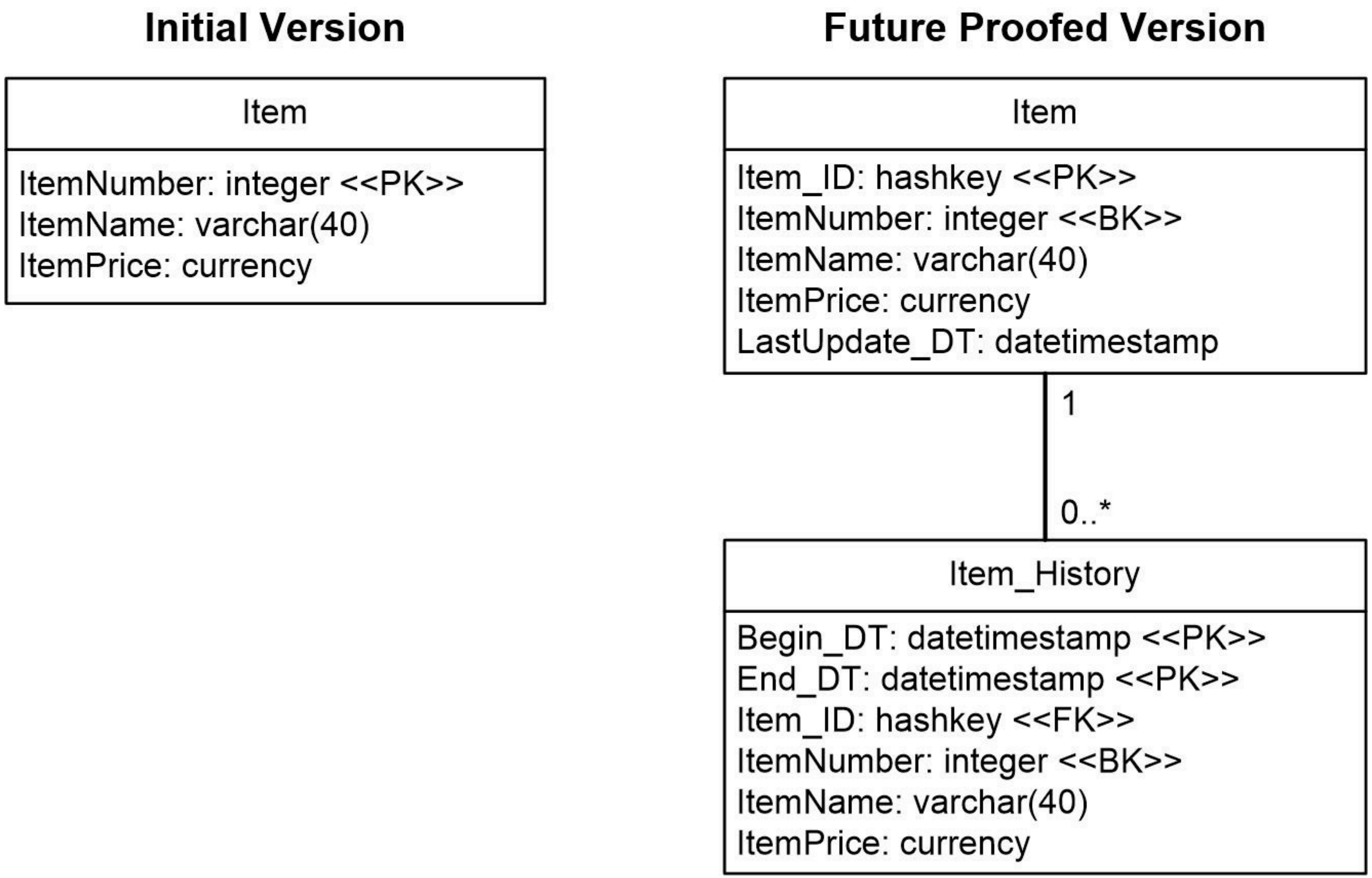
- Maintain historical data values
- Implement soft deletes
- Adopt truly unique surrogate keys
- Set and follow common conventions

Source: AgileData.org/essays/databaseDesign.html

Agile
Data

Technical
Future Proofing

- **Clean Architecture and Design**
- Agile Data Modeling
- Database Refactoring
- Database Regression Testing
- Continuous Database Integration
- Configuration Management



Initial Version

Item

ItemNumber: integer <<PK>>
ItemName: varchar(40)
ItemPrice: currency

Future Proofed Version

Item

Item_ID: hashkey <<PK>>
ItemNumber: integer <<BK>>
ItemName: varchar(40)
ItemPrice: currency
LastUpdate_DT: datetimestamp

Item_History

Begin_DT: datetimestamp <<PK>>
End_DT: datetimestamp <<PK>>
Item_ID: hashkey <<FK>>
ItemNumber: integer <<BK>>
ItemName: varchar(40)
ItemPrice: currency

Source: AgileData.org/essays/databaseDesign.html

- **Vertical Slicing**
- Clean Architecture and Design
- Agile Data Modeling
- Database Refactoring
- Database Regression Testing
- Continuous Database Integration
- Configuration Management

Every sprint a disciplined agile team produces a working solution

Functionality is added in “vertical slices”

Example slicing strategies for DW/BI:

- One new data element from a single data source
- One new data element from several sources
- A change to an existing report
- A new report
- A new reporting view
- A new data mart table

Source: AgileData.org/essays/verticalSlicing.html

Adopting the Agile Database Techniques Stack

Common Adoption Challenges

1. Traditional data culture
2. Minimal automated testing
3. Executive apathy
4. Overwhelming backlog of data work
5. Little willingness to experiment

Parting Thoughts

The Agile Data Site
AgileData.org

Key articles:

- Data Technical Debt: How to Address Quality Problems in Data Sources
- Database Techniques Stack
- The Agile Data Architect

Agile
Data

The Disciplined Agile Site
PMI.org/disciplined-agile

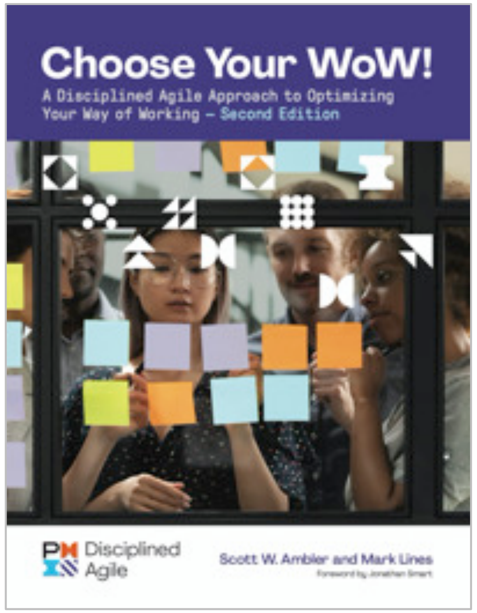
Key articles:

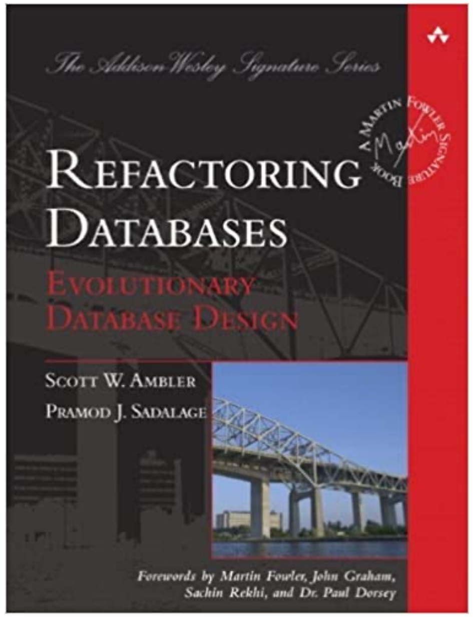
- The Disciplined Agile Mindset
- Disciplined Agile Data Management
- Disciplined Agile Enterprise Architecture

Thank You!

- Twitter: @scottwambler
- linkedin.com/in/sambler/

- scott@scottambler.com

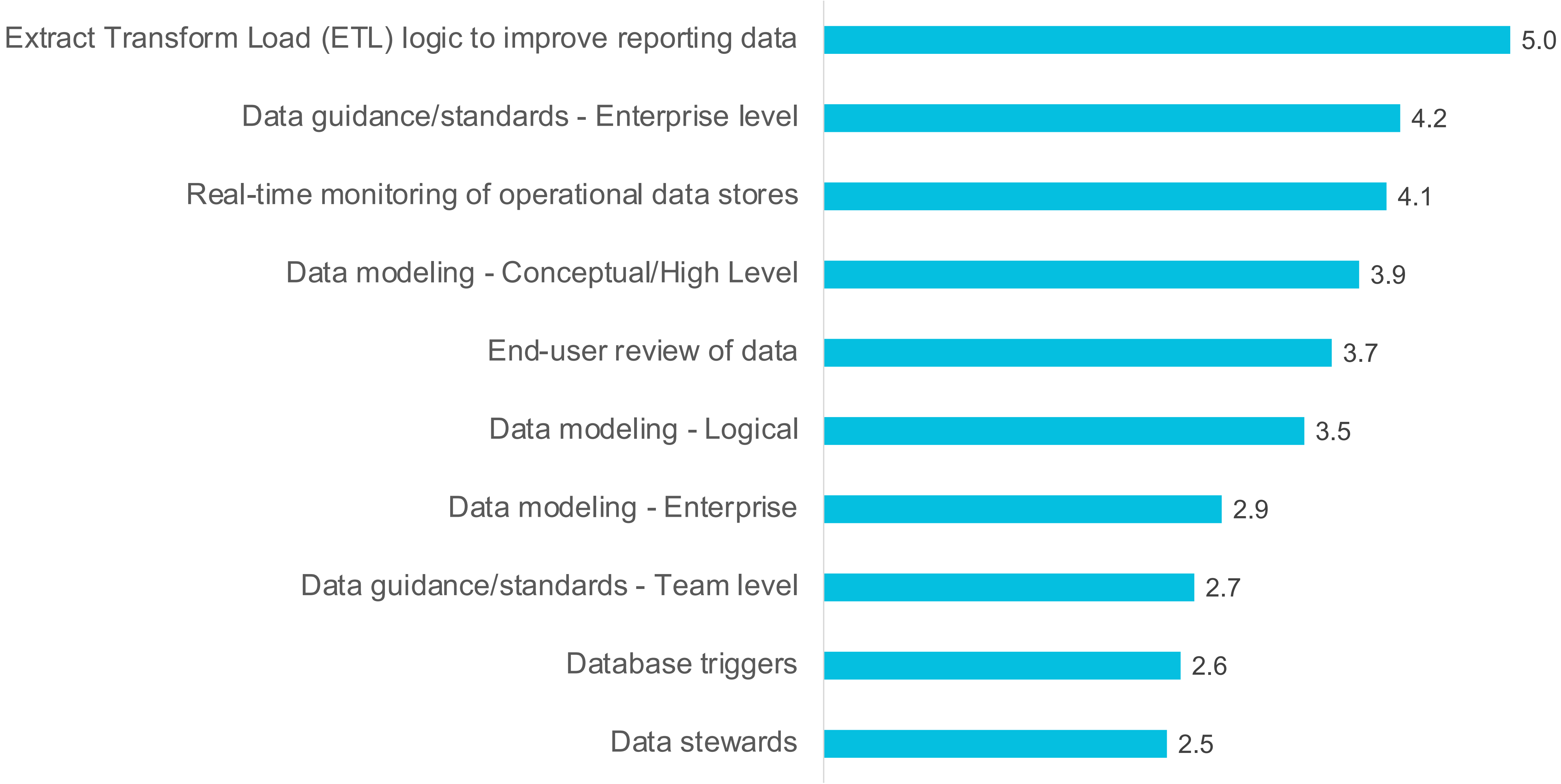

Choose Your WoW!
A Disciplined Agile Approach to Optimizing Your Way of Working – Second Edition
PMI Disciplined Agile
Scott W. Ambler and Mark Lines
Foreword by Jonathan Smart


The Addison Wesley Signature Series
A Martin Fowler Signature Book
Refactoring Databases
Evolutionary Database Design
Scott W. Ambler
Pramod J. Sadalage
Forewords by Martin Fowler, John Graham, Sachin Rekhi, and Dr. Paul Dorsey

BACKUP SLIDES

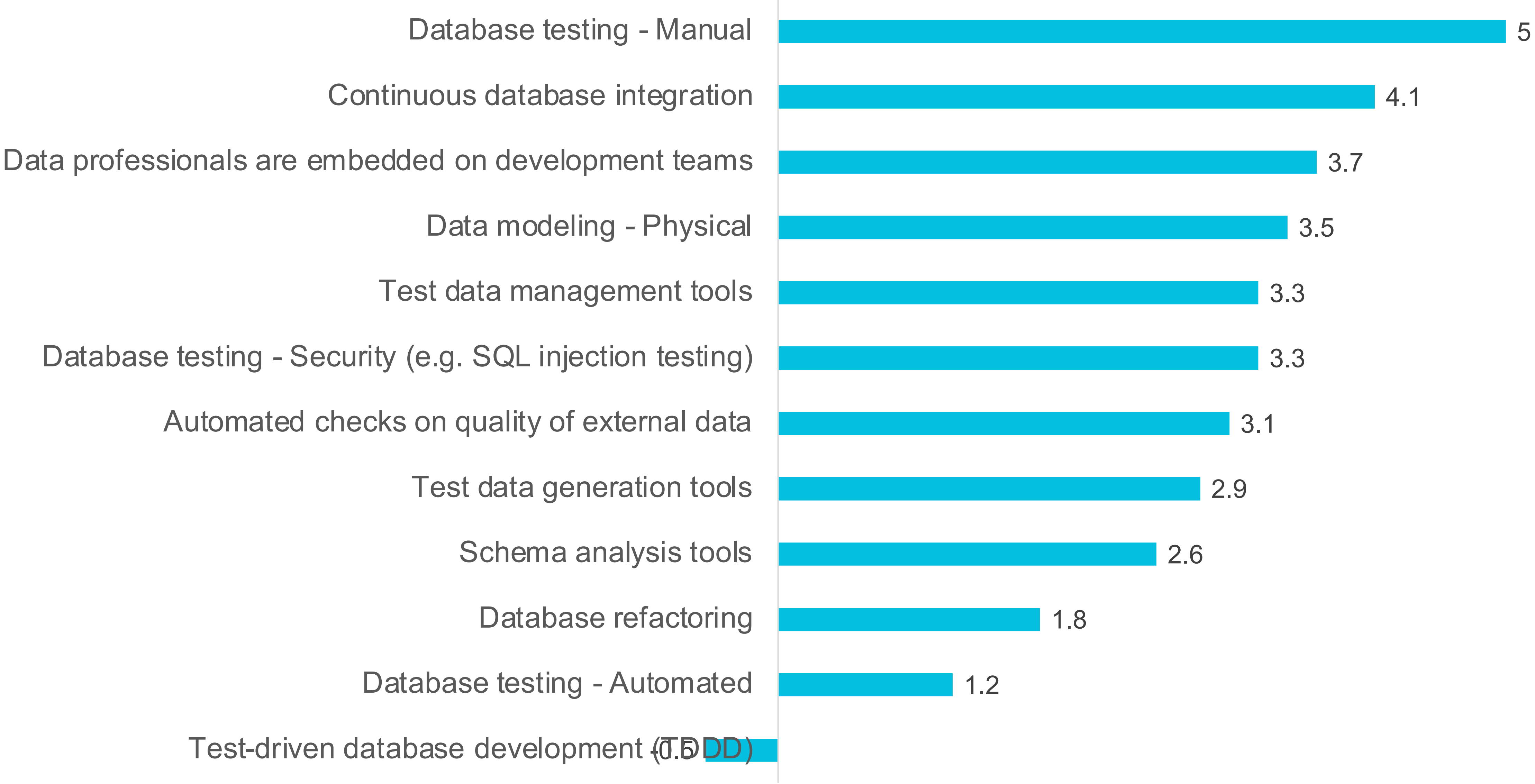
FOR MORE INFORMATION:
WWW.PMI.ORG/DISCIPLINED-AGILE

Adoption Rates of Traditional Data Quality Techniques
Score: +10 to -10



Source: PMI's 2022 Data Quality Survey

Adoption Rates of Agile Data Quality Techniques
Score: +10 to -10



Source: PMI's 2022 Data Quality Survey