

Agile Governance at Scale

Project Management Institute New York City

April 29, 2020

Craig K. Strong, CTO,
Savant Financial Technologies d/b/a Ariel Partners
1501 Broadway, 12th Floor, NY, NY 10036
May 27, 2020

1. About Me

2. Context

- 2.1 Genesis of the effort
- 2.2 Governance and Oversight
- 2.3 Agile Governance and Oversight
- 2.4 Traditional vs Agile Governance

3. Agile Audit Framework Large- Scale Agile

- 3.1 Sources
- 3.2 Project Scorecard
- 3.3 SSA OIG
- 3.4 Control Areas
- 3.5 Control Activities

4. Governance for Large-Scale Agile

- 4.1 Measure the Right Things
- 4.2 Avoid Common Failure Modes
- 4.3 Adopt Better Practices
- 4.4 Keep Track

5. Key Takeaways

6. Trainings

7. About Ariel

1. About Me

Craeg Strong



CTO, Ariel Partners
AKT, KCP, KMP, CSM, CSP, CSPO,
ITILv3, PMI-ACP, PMP, LeSS, SAFe

www.arielpartners.com
cstrong@arielpartners.com
[@ckstrong1](https://twitter.com/ckstrong1)

- Software Development since 1988
- Large Commercial & Government Projects
- Kanban Coach / DevOps Engineer
- Kanban Trainer / SpecFlow Trainer
- Performance & Scalability Architect
- Certified Ethical Hacker
- New York & Washington DC Area



Department of
Social Services



FannieMae®

1. What We've Done



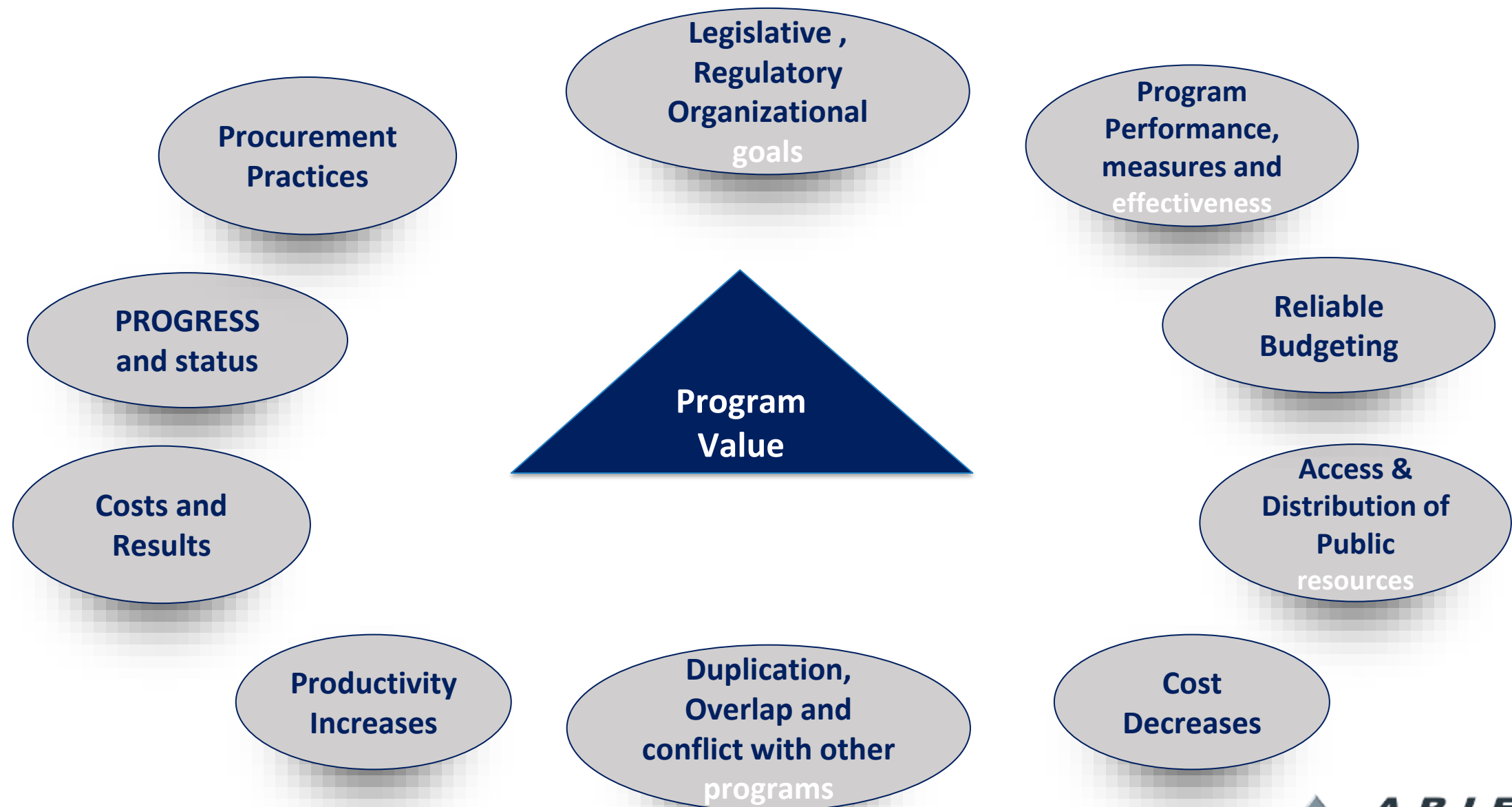
2. Context

2.1 Genesis of this Effort



- Creating Agile Auditing Framework for US Agency
- Agency Context
 - ❑ Lots of Oversight, including House Ways and Means
 - ❑ Large \$100M+ Agile Efforts
 - ❑ Relatively New to Agile
 - ❑ Large, Diverse Group of Stakeholders
 - ❑ All 50 states
 - ❑ Significant Legacy Component

2.2 Governance and Oversight: Audit Effectiveness



2.3 Agile Governance and Oversight

Why is oversight of an Agile projects more difficult?



Scheduling and Budgeting Challenges

- Lack of Detailed Plans Up-Front

Quality Assurance Challenges

- Reduced Emphasis on Documentation

Measurement Challenges

- Lack of Traditional Metrics such as Earned Value
- Unfamiliar, Subjective Metrics such as “Story Points”

Management Challenges

- Plethora of Agile methods and practices
- Diversity of Approaches
- Conflicting advice
- Rapidly evolving ecosystem
- Traditional sources such as PMBOK have not kept pace

2.4 Traditional Versus Agile Governance

What makes agile so different?

| | Traditional | Agile |
|----|---|--|
| 1 | Planning with Accuracy and Precision | Planning with Accuracy and Adaptability |
| 2 | Predictive: Forecasting via Estimation | Empirical: Forecasting via Statistics and Probability |
| 3 | Adherence to Plan | Flexibility: Welcome changes/clarifications |
| 4 | Up-Front Requirements Gathering, Baselining | Direct, Continuous Customer Involvement |
| 5 | Documentation First | Automation First |
| 6 | Measure Major Milestones | Measure Continuous Flow of Value |
| 7 | Handoffs Between Defined Roles and Duties | Collaboration within and across teams, Cross-Training |
| 8 | Post-Mortem Lessons Learned | Inspection and Adaptation via Continual Retrospectives |
| 9 | Comprehensive Analysis, Design, Documentation | Lean Analysis, Design, Documentation |
| 10 | If something is risky and difficult, measure twice, cut once. Make sure to get it right the first time! | If something is risky and difficult, then do it constantly. Constant integration, constant refactoring, etc. |

3. Audit Framework for Large-Scale Agile

3.1 Sources



- ☐ Digital Services Playbook
- ☐ Management and Oversight of Federal Information Technology (FITARA)



- ☐ TechFAR Handbook



- ☐ Biannual FITARA Scorecard



- ☐ Government Auditing Standards ("Yellow Book")
- ☐ Effective Practices and Federal Challenges in Applying Agile Methods
- ☐ Technology Assessment Design Handbook
- ☐ Organizational Transformation: A Framework for Assessing and Improving Enterprise Architecture
- ☐ A Framework for Assessing and Improving Process Maturity
- ☐ 10+ Congressional Reports

3.1 Sources



- ❑ PMBOK 6th Edition Agile Practice Guide



- ❑ CMMI v2.0 Model At-A-Glance
- ❑ How to Truly Scale Agile Development In the Enterprise – with CMMI
- ❑ Using Agile with Scrum and CMMI



- ❑ DIB Guide: Detecting Agile BS
- ❑ DIB Ten Commandments of Software
- ❑ DIB Metrics for Software Development



- ❑ Agile Development & Delivery for Information Technology Instruction Manual

3.2 Project Scorecard

What is our level of risk?

1. Project Integration Management
2. Project Scope Management
3. Project Schedule Management
4. Project Cost Management
5. Project Quality Management
6. Project Resource Management
7. Project Communications Management
8. Project Risk Management
9. Project Procurement Management
10. Project Stakeholder Management

PMBOK Knowledge Areas

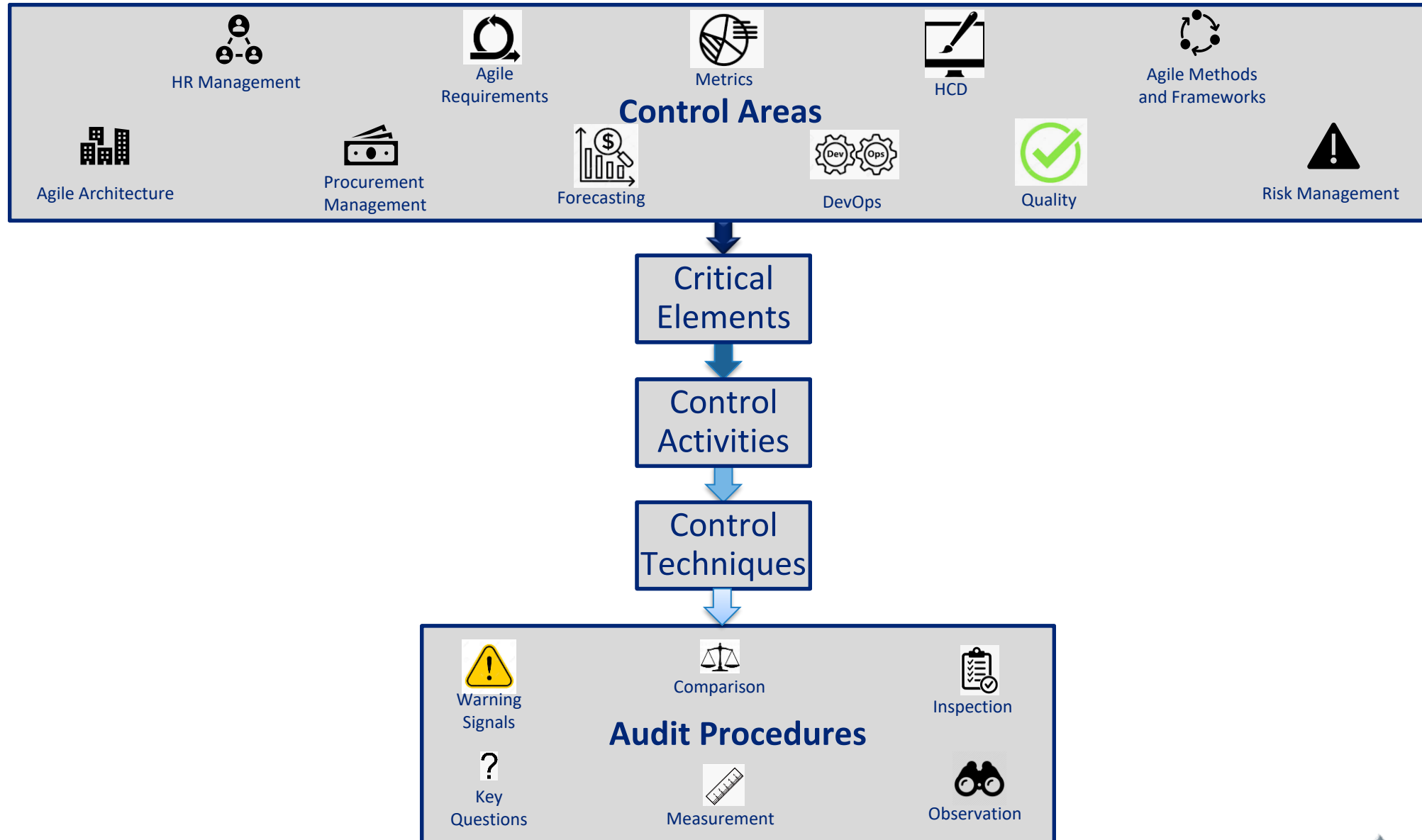
FITARA 9.0

| Agency | Nov 2015 Grade | May 2016 Grade | Dec 2016 Grade |
|--------|----------------------|----------------------|----------------------|
| USDA | D | C | C- |
| DOC | B | B | B+ |
| DOD | D | D | D+ |
| Ed. | F | D | C+ |
| Energy | F | C | C- |
| HHS | D | D | D- |
| DHS | C | C | B- |
| HUD | D | D | C- |
| DOI | C | C | B+ |
| DOJ | D | C | B- |
| DOL | D | C | C- |
| State | D | D | D- |
| DOT | D | D | F+ |
| Treas. | D | D | C- |
| VA | C | C | B+ |
| EPA | C | C | B+ |
| GSA | B | C | B+ |
| NASA | F | F | C+ |
| NSF | D | D | C- |
| NRC | C | C | C- |
| OPM | D | C | C+ |
| SBA | D | D | D- |
| SSA | D | C | B+ |
| USAID | D | D | D- |



| | Project DCAPS |
|--|------------------|
| 1 Agile Methods & Frameworks (AMF) | D |
| 2 Agile Requirements (AR) | F |
| 3 Forecasting, Scheduling & Planning (FSP) | C |
| 4 Metrics & Tracking (MT) | F |
| 5 Risk Management (RM) | D |
| 6 HR Management / Staffing (HR) | A |
| 7 Human Centered Design (HCD) | C |
| 8 Quality and Test Automation (QA) | D |
| 9 DevOps and ALM (DO) | C |
| 10 Agile Architecture (AA) | D |
| 11 Procurement Management (P) | B |
| Overall | C- |

3.3 SSA OIG: Performance Audit Framework



3.4 Control Areas & Critical Elements

■ Agile Methods and Frameworks (AMF) Controls

- Critical Element AMF-1. Ensure Project Team uses an Appropriate Team-Level Agile Method
- Critical Element AMF-2. Implement Effective Team-Level Agile Controls
- Critical Element AMF-3. Ensure Project Team Uses an Appropriate Scaled Agile Method
- Critical Element AMF-4. Implement Effective Scaled Agile Controls

■ Agile Requirements (AR) Controls

- Critical Element AR-1. Ensure Project Team Has a Documented Vision and Overall Strategy
- Critical Element AR-2. Implement Effective High-Level Scope Controls
- Critical Element AR-3. Implement Effective Roadmap Controls
- Critical Element AR-4. Implement Effective Requirements Elaboration & Maintenance Controls
- Critical Element AR-5. Implement Effective Scope Management & Reduction Controls

■ Forecasting, Scheduling, & Planning (FSP) Controls

- Critical Element FSP-1. Ensure Project Team Establishes Initial Project Forecast
- Critical Element FSP-2. Implement Effective Progress Tracking Controls
- ...

3.5 Control Activities, Control Techniques, Audit Procedures

| Control Activities | Control Techniques | Audit Procedures |
|--|--|--|
| AR-1.1 Project has a documented vision | AR-1.1.1 Overall Objectives and Goals for the Project Have Been Documented | <ul style="list-style-type: none">▪ Project Vision artifact exists▪ Appropriate level of detail▪ All team members are aware▪ Evidence of being actively maintained |
| | AR-1.1.2 Business Drivers for Project Have Been Documented | <ul style="list-style-type: none">▪ Vision describes business drivers, goals, and objectives▪ Goals include reasoning and justification▪ Goals are ordered▪ Target dates and cost, any budgetary or time considerations▪ Evidence of trade-off decisions informed by drivers |
| | AR-1.1.3 High Level Functions for Project Have Been Documented | <ul style="list-style-type: none">▪ Vision includes high-level business functions with context▪ Functions out of scope are listed▪ Reasonable cardinality (dozens / hundreds, not thousands) |
| | AR-1.1.4 Technical & Business Constraints Have Been Documented | <ul style="list-style-type: none">▪ Major Integrations, platform requirements, standards listed▪ Business constraints are listed (e.g. data center, place of performance) |

4. Governance for Large-Scale Agile

4. Highlights

1. Measure the Right Things

- Broken Windows Strategy
- Balanced Metrics
- Human Centered Design
- Testable Architecture

2. Avoid Common Failure Modes

- Agile Methodology Failures
- Using Velocity for Long-Range Forecasting

3. Adopt Better Practices

- Kanban Flight Levels for Dependency Coordination
- Monte Carlo Simulation Based Forecasts

4. Keep Track

- We Need a High-Quality Repository of Reference Data

Agile Governance

4.1 Measure the Right Things

4.1 Broken Windows Strategy

Sweat the Small Stuff



| | | |
|-------------------------------------|---|--|
| Code Does Not Meet Style Guide | ➡ | Linting / Code Static Analysis |
| Huge Product Backlog | ➡ | Query: Find Old & Untouched Stories |
| Orphan / Unlinked Items in ALM Tool | ➡ | Query: Find Badguys (“Lint” for ALM) |
| ALM Has Poor Usability / Friction | ➡ | Count # Clicks, # Defaultable Items without Default Value |
| Rubberstamp Reviews | ➡ | Count Short / Blank Peer Reviews |
| Duplicative / Copy Paste Code | ➡ | Detect Duplicates, Watch For Trend of Code Size to Flatten |

4.1 Balanced Metrics

Balance Armor & Mobility

Balance Armor & Mobility



Sprint Velocity &
Sprint Story Delivery Count

+

Undelivered Story Points
Undelivered Story Count

Average Throughput

+

Average Lead Time

Number of Unit Tests

+

Build Time

Code Coverage

+

Functional Coverage

Deployment Frequency

+

Failed Deployment
Down-Time

4.1 What are some key DevOps Metrics?

Google DORA

Change Failure Rate (CFR)

- Number of changes that result in failure
- An indicator of how well we are doing manual and automated testing

Mean Time to Recover (MTTR)

- Downtime divided by number of incidents
- Decreases may indicate our DevOps pipeline is better able to deploy things more quickly and safely

Mean Lead Time for Changes (MLT)

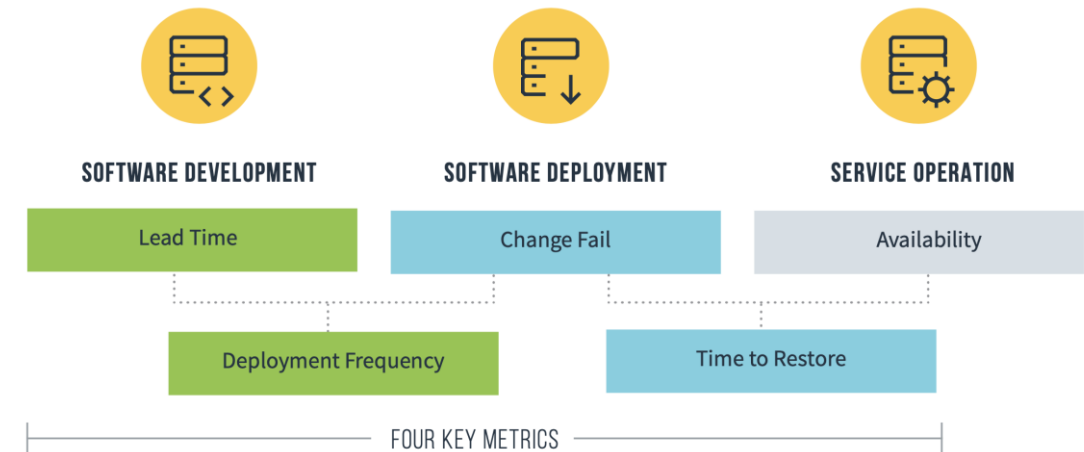
- From Code commit to running successfully in prod—how long on average does it take?

Deployment Frequency (DF)

- Number of deployments per day

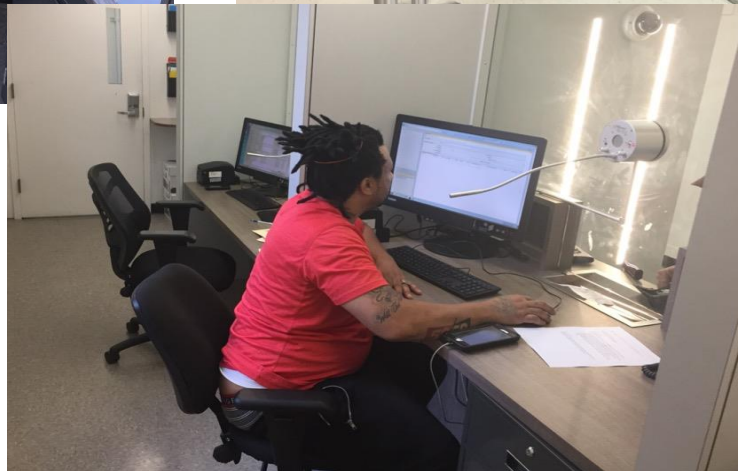


PERFORMANCE METRICS



4.1 Human Centered Design

Know Your Customer

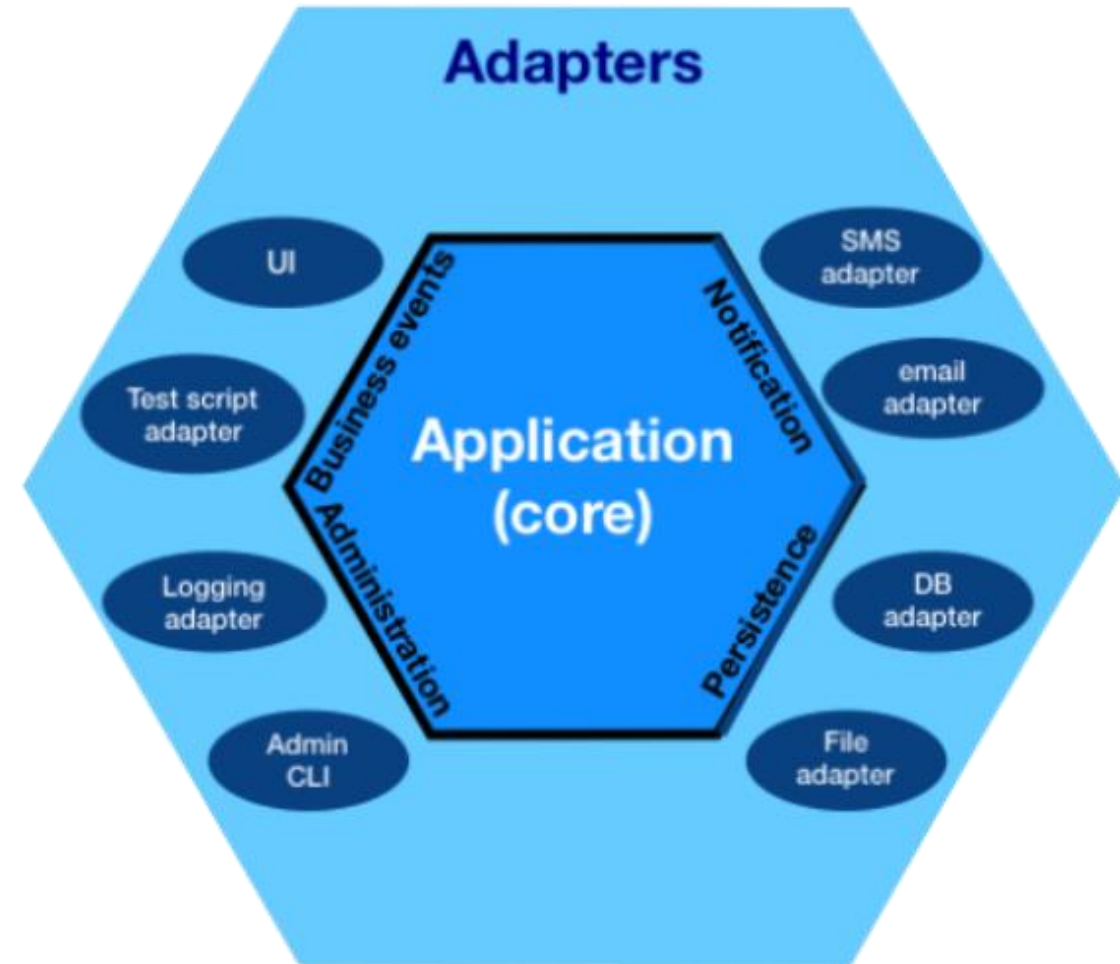


- How Many Team Members Have Participated in In-Context Immersion?

4.1 Testable Architecture: APIs

The More Testable an Architecture Is, The Better It Is

- Testability Brings
 - Instrumentation
 - Scalability
 - Pluggability
 - Performance Tunability
 - Tests-as-Documentation



Agile Governance

4.2 Avoid Common Failure Modes

4.2 Common Failure Modes

Mixing and Matching Disciplines



Skipping Inconvenient or Difficult Practices

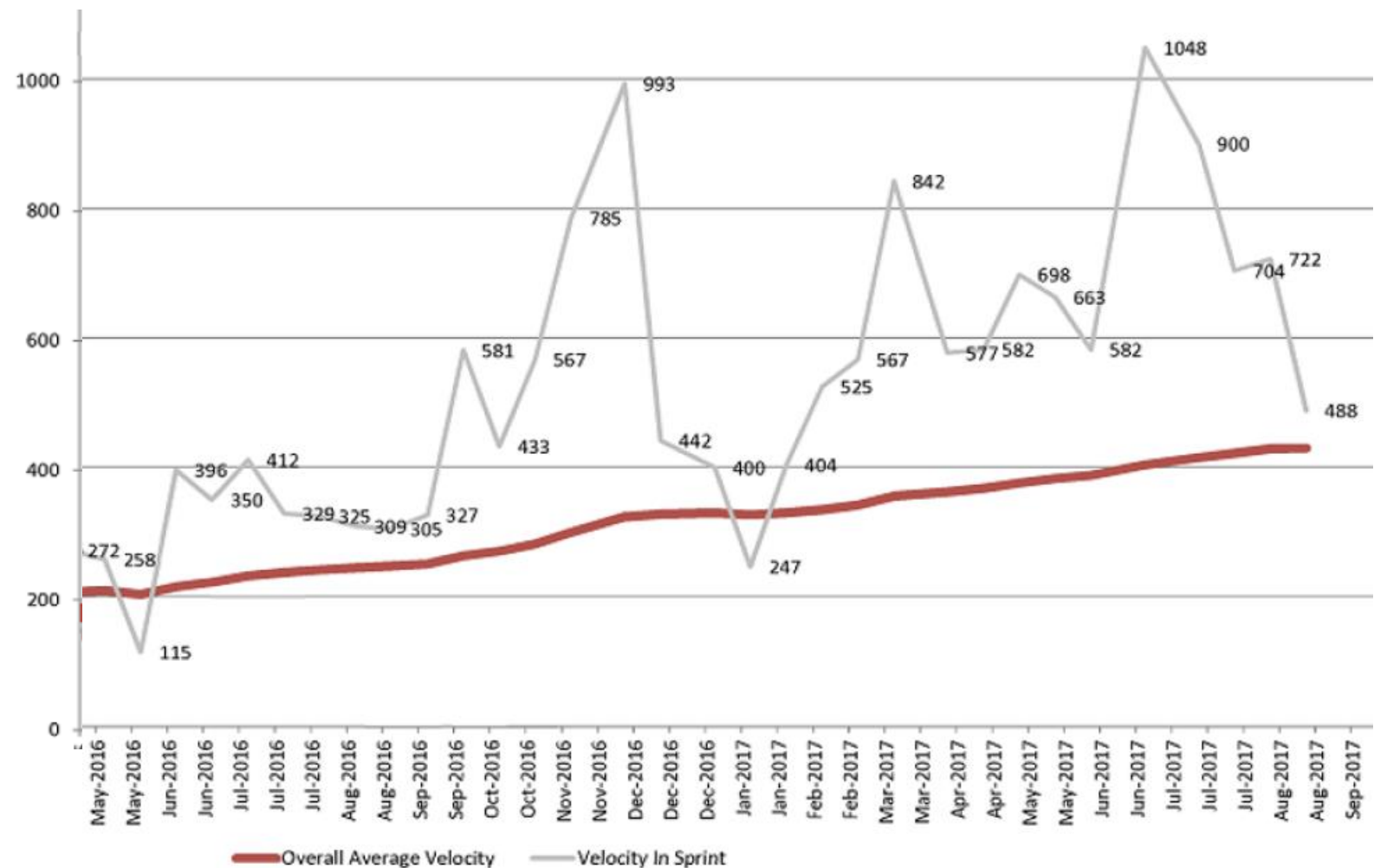


Confidential

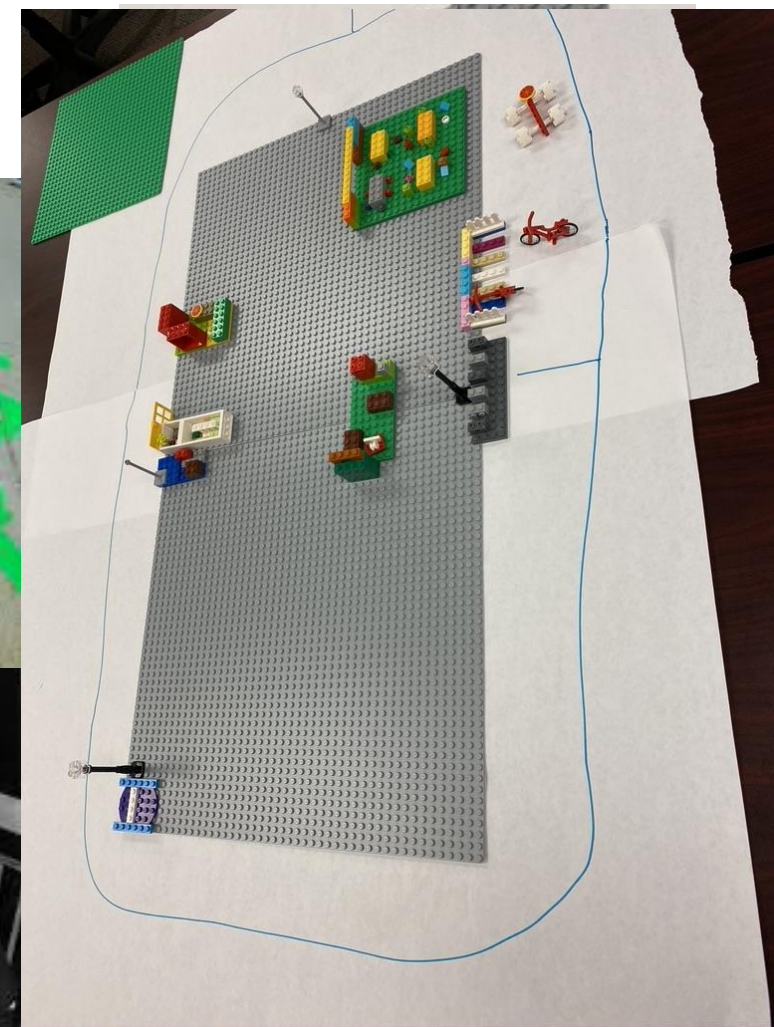
Mandating Advanced Practices w/o Adequate Prep



4.2 Common Failure Modes: Using Velocity for Long-Range Forecasting



4.2 Why Not Use Story Point Estimates for Long-Range Forecasting?



Agile Governance

4.3 Adopt Better Practices

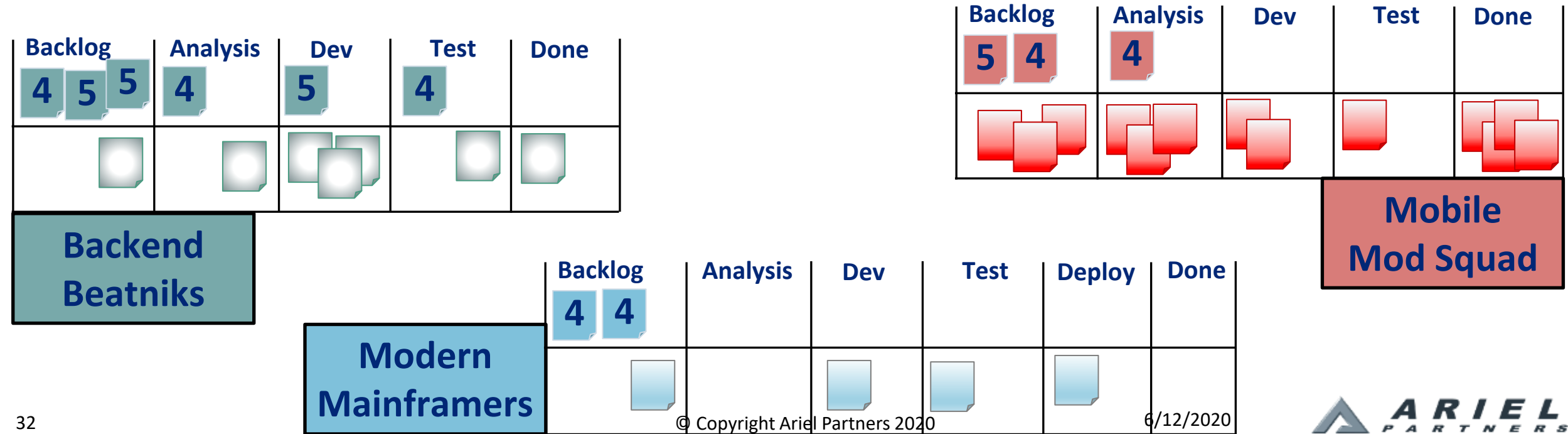
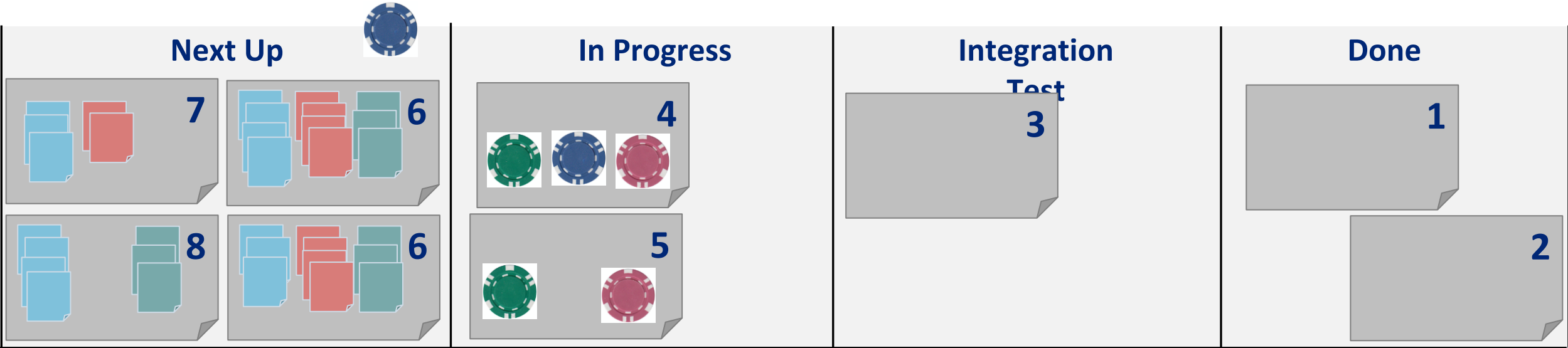
4.3 Dependency Handling The Hard Way

Method One: Giant Up-Front Meeting



- Opportunity Cost of Large Meeting
- Tough to Detect all Dependencies Up Front
- Significant Planning & Management Overhead

4.3 Easier Dependency Handling: Kanban Coordination Board



4.3 Portfolio Board Example



- Ties strategy to execution
- Simplifies trade-off analysis
- Facilitates Hypothesis-Driven Development (HDD)
- Enables Business Agility

4.3 Forecasting

Modern Methods Provide Improved Predictive Power

| Feature | Total Stories | Freeze Date | 85% Completion Date | Completion Likelihood | Stories Remaining |
|-------------------------------|---------------|-------------|---------------------|-----------------------|-------------------|
| Carryover from previous PI | 32 | 1/31/17 | 1/25/17 | 37.42% | 12 |
| Child turning 18 | 20 | 1/31/17 | 5/12/17 | 24.56% | 5 |
| Age 18 redetermination | 1 | 1/31/17 | 1/10/17 | 99.99% | 1 |
| Auto cas assignment | 35 | 1/31/17 | 1/3/17 | 99.99% | 35 |
| FiscalSearch | 4 | 1/31/17 | 1/3/17 | 99.90% | 4 |
| Hourly invoice/pay per action | 24 | 1/31/17 | 2/22/17 | 28.46% | 8 |
| Link Family cases | 27 | 1/31/17 | 1/17/17 | 99.94% | 27 |
| 3rd party child contacts | 6 | 1/31/17 | 2/3/17 | 80.58% | 5 |
| Multi-language insert | 17 | 1/31/17 | 12/21/16 | 99.99% | 17 |
| Enhanced Referrals | 188 | 1/31/17 | 3/24/17 | 25.06% | 47 |
| Assistance Scope | 5 | 1/31/17 | 12/21/16 | 99.99% | 5 |
| Workload sharing | 12 | 1/31/17 | 1/17/17 | 99.99% | 12 |
| MER - Auto match vendors | 10 | 1/31/17 | 12/21/17 | 0.74% | 10 |
| Resend correspondence | 63 | 1/31/17 | 9/12/17 | 0.01% | 63 |
| Re-openings | 6 | 1/31/17 | 2/8/17 | 99.90% | 6 |

- Use Reference Class Forecasting
- Address Outliers, Increase Predictability
- Use Monte-Carlo Simulations to Predict Likelihood of Hitting Target
- Proactive: Few Surprises!



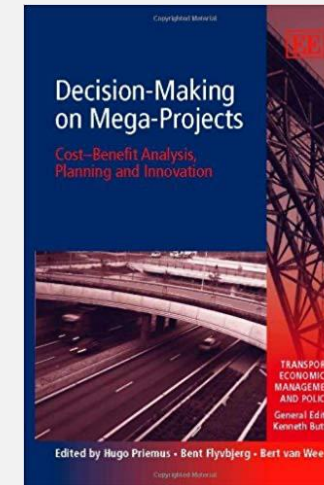
MCS-powered release dashboards expose areas of higher risk early in a release, enabling them to be addressed proactively

Agile Governance

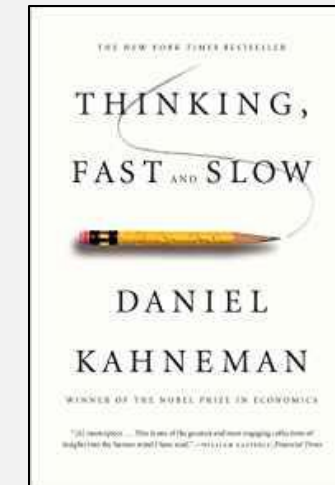
4.4 Keep Track

4.4 Keeping Track: Reference Class Forecasting

- Degree of Complexity: Stakeholders, Interfaces, Legacy
- Agile Methods & Practices Used
- Retrospective Notes
- Rate of “Dark Matter” Expansion
- Cumulative Flow Diagram
- Test Coverage Curve
- Other Metrics



2008



2013

5. Key Takeaways

- Current State of Large-Scale Agile Governance is Woefully Inadequate
- Fixing This Requires A New Approach
 - Measure the Right Things
 - Avoid Common Failure Modes
 - Adopt Better Practices
 - Keep Track
- Benefits: Lower Risk for Large-Scale Software Development

6. Training Offerings

| # | Class | Description |
|----|---|---|
| 1 | Fundamentals of Agile | (2-day) Overview of Scrum, Kanban, scaling, HCD, DevOps |
| 2 | Human Centered Design | (2-day) Design thinking, journey maps, personae, How Might We (HMW), stakeholder mapping, empathy mapping, behavioral economics |
| 3 | Professional Scrum | (1-day) Team-level Scrum |
| 4 | AKT | (1-day) Team-level Kanban |
| 5 | KMP I and II | (4-day) Certification course for Kanban Management Professional (KMP) |
| 6 | Scaled Scrum | (1-day) Overview of LeSS, SAFe, Nexus, Scrum@Scale |
| 7 | Scaling with Kanban | (1-day) Overview of Flight Levels, Portfolio Kanban, managing dependencies with coordination boards, Kanban Maturity Model |
| 8 | Agile Requirements | (1/2-day) User Story workshops, Story Splitting |
| 9 | Agile Estimation, Metrics and Forecasting | (1/2-day) Forecasting using Monte Carlo Simulation, Rightsizing, NoEstimates, T-Shirt, Story points |
| 10 | Agile Architectures | (1/2-day) Design patterns for testability, manageability and legacy migration including Strangler and Hexagonal patterns. |
| 11 | Application Lifecycle Management with VersionOne and JIRA | (1/2-day) Power user's guide to using advanced features, reporting, querying, import/export. Variants for both VersionOne and Atlassian JIRA. |
| 12 | Agile Automated Testing | (1/2-day) Test-driven, behavior-driven, acceptance-test-driven, hypothesis-driven development, traceability, reporting |
| 13 | DevOps Foundations | (1/2-day) Overview of DevOps and DevSecOps |
| 14 | Agile for Leaders and Executives | (1 day) Seminar for leaders and executives leading digital transformation, reviewing mindset and behaviors for cultural shift and connecting strategy to execution. |

7. About Ariel

www.arielpartners.com
sales@arielpartners.com

Training Offerings

- Fundamentals of Agile
- Agile for Leaders & Executives
- Kanban Management Professional
- Professional Scrum
- Human Centered Design
- BDD With Cucumber Acceptance Testing
- Agile Estimation, Forecasting, & Metrics
- Agile Requirements

Other Offerings

- Digital Transformation
- Cloud Native App Development
- Agile / Kanban Coaching
- DevOps Jumpstart
- Compliance As Code
- Test Automation Jumpstart
- Legacy Modernization
- JIRA Jumpstart

**Thank
You!**

1501 Broadway, 12th Floor,
NY, NY 10036
<https://www.arielpartners.com>
646-467-7394

