## **Enterprise Engineering Analytics Dashboards**

Julie Fant, Jodene Sasine, Donna Nystrom, Jennifer Starcher, and Vivian Sau Software Engineering Subdivision

February, 2022

Approved for public release. OTR 2022-00324.

#### Overview

- This presentation will discuss the Engineering Analytics Dashboards (EAD) framework including:
  - The specialized key performance indicators (KPIs) and metrics selected to support the framework
  - A demonstration of the EAD framework's current capabilities
  - A summary of the framework's future direction
  - Insights on the challenges and lessons learned to adopting

### Goals

- Engineering Analytics Dashboards will provide architects, engineers, and managers with the right knowledge at the right time to support effective, integrated decision-making
- Dashboards will be built on a flexible framework of reusable templates and patterns that can be customized to an Agile enterprise's processes, roles, and distributed data sources
- KPIs, metrics and visualizations will harness the power of distributed MBSE and Agile enterprise data to
  - Create holistic, Agile enterprise knowledge
  - Inform effective, integrated decision-making
  - Highlight issues and reduce risk



# Engineering Analytics Dashboards transform distributed data to inform effective, integrated decision-making

## The Relationship between MBSE and Agile





=

#### FY21 Dashboards Scope: Levels of Enterprise Scope

- Within an enterprise, there are multiple levels of scope where MBSE and Agile can be applied
- Each level of scope has its own roles and use cases with differing perspectives and concerns
- Engineering Analytics Dashboards have the potential to be applied at any level, but must be tailored to that level's roles and use cases
- Initial scope address proof-of-concept System Level Dashboards





## **Agile Focused Dashboard**

### **Agile Process Assumptions**



- Select Agile Tracking Tool (e.g, Jira)
- Identify Program Increment (PI), Minimal Viable Product (MVP), and Technical Debt in tool to enable metric generation
  - Determine authoritative source of truth for calculation purposes

#### Agile Considerations: Process Assumptions



- Select Agile requirement hierarchy for program
  - Defines metrics to be visualized

#### Use Case Ideas for Agile



Use Case	Description
Monitor MVP Progress	Dashboard used by Materiel Lead to monitor overall program progress with respect to prioritized MVPs and to understand risks associated with key events and/or dependencies (e.g., hardware). Also used by Product Manager to monitor progress and quality of the MVP work effort and to understand risks associated with achieving MVP for timely user feedback.
Track Program Increment (PI) Status	Dashboard used by Product Manager to monitor progress and quality of the PI for the release train and to understand risks associated with achieving increment goals, team goals and resourcing bottlenecks. Also used by Release Train Engineer to monitor progress towards PI goals, as agreed to by stakeholders at PI planning, and to understand risks associated with achieving product functionality and quality based on PI goals.
Monitor Agile Development	Dashboard used by Release Train Engineer to monitor Agile processes and health of the release train and to understand risks associated with resourcing, team velocity, and quality.

## Use Case: Monitor MVP Progress

Stakeholders: Materiel Lead, Product Manager



- Requires assumptions on Agile requirements hierarchy
- Requires identification of MVP, Technical Debt, and Blockers

#### Dashboard monitors overall program progress, effort, and quality of prioritized MVPs



## **MBSE Focused Dashboard**

# **MBSE Process Assumptions**



- MBSE is used to develop the architectural runway and to inform the Agile Increment Planning Process
  - Epics and Features will be identified in the model and traced to model elements
  - Subset of system model will be used as design / reference architecture for the Epic (and maybe Features too)
- System implementation changes will need to be flowed back up
- Some Requirements Management tools and processes are responsible for maintaining and reporting on external requirements

=

# Use Case Ideas for MBSE



Use Case	Description
Monitor MBSE Model Development	Dashboard used to track the overall model development and how well the PI Epics and Features are satisfied by the design
Track Requirement Satisfaction in Design	A lower-level dashboard that tracks the requirement satisfaction of external and Agile requirements in design and Verification and Validation information in the model

## Use Case: Monitor MBSE Model Development

Stakeholders: Lead System Architect



- Requires assumptions/customizations for how Agile requirements are traced into design
- Baseline data assumes some frequency of model baseline

# Dashboard monitors overall model development and how well the PI Epics and Features are satisfied by the design

#### **Metric Customization**

- There can be multiple ways to model and chapter information within MBSE models
- For example, there is no standard approach for capturing Epics and Features in models AND there is no standard way to capture the traceability
  - The figure at right shows examples of two equally valid traceability approaches
  - The model query required to calculate requirement satisfaction for Example1 is quite different than the query required for Example 2
- Our proof-of-concept tool will utilize one approach, but customization is needed when applied to a specific program









#### Dashboards will need to be customized based on modeling approach



# Challenges and Lessons Learned

#### **Challenges and Lessons Learned**

#### Agile Process

- Requirements hierarchy differs across multiple programs
- Processes need to be in place and consistent to calculate MVP, Technical Debt, and Blocker metrics
- Cameo Plugin and API limitations
  - JavaDocs documenting the OpenAPI are hard to understand
    - Some OpenAPI methods are deprecated and replaced with other methods, but it is hard to match them up
    - Cameo documentation briefly talks about the OpenAPI and how to use it
  - Accessing branch and version data in Teamwork Cloud is difficult
    - OpenAPI cannot access this data
    - Teamwork Cloud REST API requires elevated privileges
  - Navigating and querying model data is challenging because models can be large, complex, and varying in organization
    - Requires iterating through the model to find the data using the OpenAPI. We are exploring alternative approaches that may be easier such as querying the data within Cameo and accessing the result from a single generic table

#### **Challenges and Lessons Learned**

#### • JIRA Data

- Jira REST API every Jira project can have different custom field attributes (e.g., custom\_field1234 = story points)
- May not have full access to the metrics needed to create dashboards

#### Tableau

- Data must be clean for Tableau to understand it
- With small data sets the views you can create in Tableau are limited

# MBSE Considerations: Accessing Data in MBSE Tools



- Many MBSE tool environments offer capabilities to access model data through either the modeling tool itself or the model configuration management tool
- Some MBSE metrics can be universally applied without the need for customization, e.g., number of model elements, number of Activity Diagrams
- Other MBSE metrics are dependent on the model implementation and require customization

# Accessing metric data from MBSE tools is not straightforward and will require development of tool and metrics customizations

### Summary and Future Work

- This presentation covered the Engineering Analytics Dashboards (EAD) framework including:
  - The specialized KPIs and metrics selected to support the framework
  - A demonstration of the EAD framework's current capabilities
  - Insights on the challenges and lessons learned to adopting
- Next steps
  - Continue to refine and pilot the prototype implementation of Agile and MBSE dashboards at the System level
  - Provide guidance and support of adopting System level dashboards
  - Research KPIs and metrics selected at the System of Systems level and develop dashboards

#### References

- 1. SAFe for Lean Enterprises, scaledagileframework.com
- 2. Lean Advancement Initiative, Systems Engineering Advancement Research Initiative, Systems Engineering Leading Indicators Guide, Version 2.0, January 29, 2010.
- 3. Cloutier, Robert. Compilation of SysML RFI- Final Report, OMG, 2010. https://www.nomagic.com/mbse/images/whitepapers/omg\_rfi\_final\_report\_02\_20\_2010-1.pdf
- 4. Fant, Julie, McShane, Karen, Velmurugan, Ronald, Pettit, Robert. OTR 2021-00256 Overview of Model Assurance Levels (MALs) for Systems and Software Models, February 2020
- 5. ISO/IEC/IEEE 15288:2015 Systems and software engineering System life cycle processes
- 6. DOD Agile Metrics Guide, Strategy Considerations and Sample Metrics for Agile Development Solutions, Version 1.1 23, September 2019
- 7. Agile Metrics, Agile Acquisition Webinar #6, The Aerospace Corporation, 2020
- 8. Agile Acquisition Webinars -#7 Agile Risks and Technical Debt Management , The Aerospace Corporation, 2020
- 9. Agile Alliance, https://www.agilealliance.org/
- 10. Practical Software and Systems Measurement Continuous Iterative Development Measurement Framework Part 2: Measurement Specifications: Automated Test Coverage Version 2.1, April 15, 2021.

#### FY21 Dashboards Scope: User Roles

User Role	Description
Materiel Lead	Responsible for cost and schedule to deliver the product and accountable to their command leader at the O6 level.
Product Manager	Responsible for defining and supporting the building of desirable, feasible, viable, and sustainable products that meet customer needs over the product-market lifecycle. © Scaled Agile, Inc.
Release Train Engineer (RTE)	Servant leader and coach for the Agile Release Train (ART). The RTE's major responsibilities are to facilitate the ART events and processes and assist the teams in delivering value. RTEs communicate with stakeholders, escalate impediments, help manage risk, and drive relentless improvement. © Scaled Agile, Inc.
Lead System Architect	Responsible for defining and communicating a shared technical and architectural vision for an Agile Release Train (ART) to help ensure the system or Solution under development is fit for its intended purpose. © Scaled Agile, Inc.
Program Office Liaison to External Requirement Stakeholders	Responsible for establishing and maintaining communication with external individuals or organizations having a right, share, claim, or interest in a system or in its possession of characteristics that meet their needs and expectations. ISO/IEC/IEEE 2015

#### FY21 User Roles address Program Office roles for System Level Use Cases