33rd Annual INCOSE international symposium

hybrid event

Honolulu HI USA

JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

Risa Gorospe (risa.gorospe@jhuapl.edu)
Shannon Dubicki (shannon.dubicki@jhuapl.edu)

Architecting Digital Engineering Requirements for Risk Management & Systems Architecting

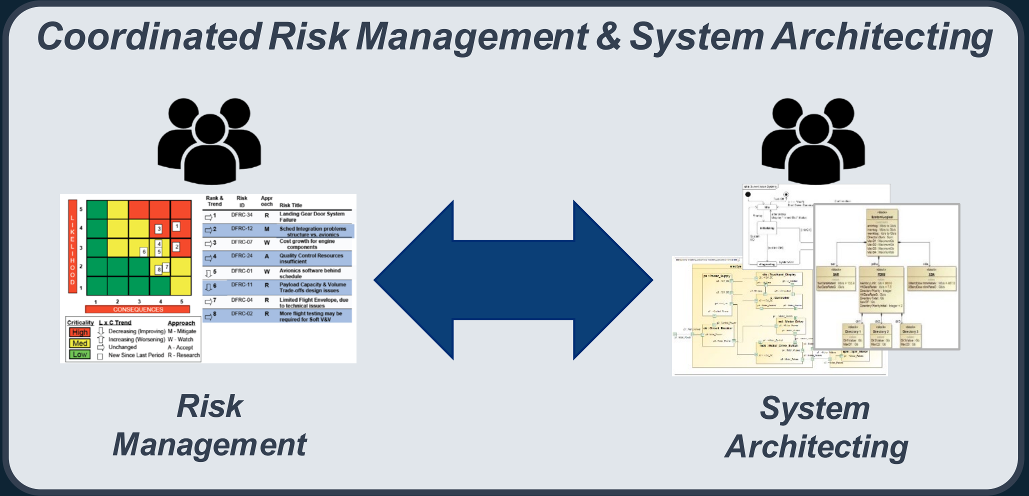
Session Agenda

- Research Background
- Technical Approach Overview
- Methodology Execution
- Other Execution Details
- Research Paper Conclusions
- Quick Overview of Research Prototype
- Lessons Learned

Research Background

Digital Thread Overview

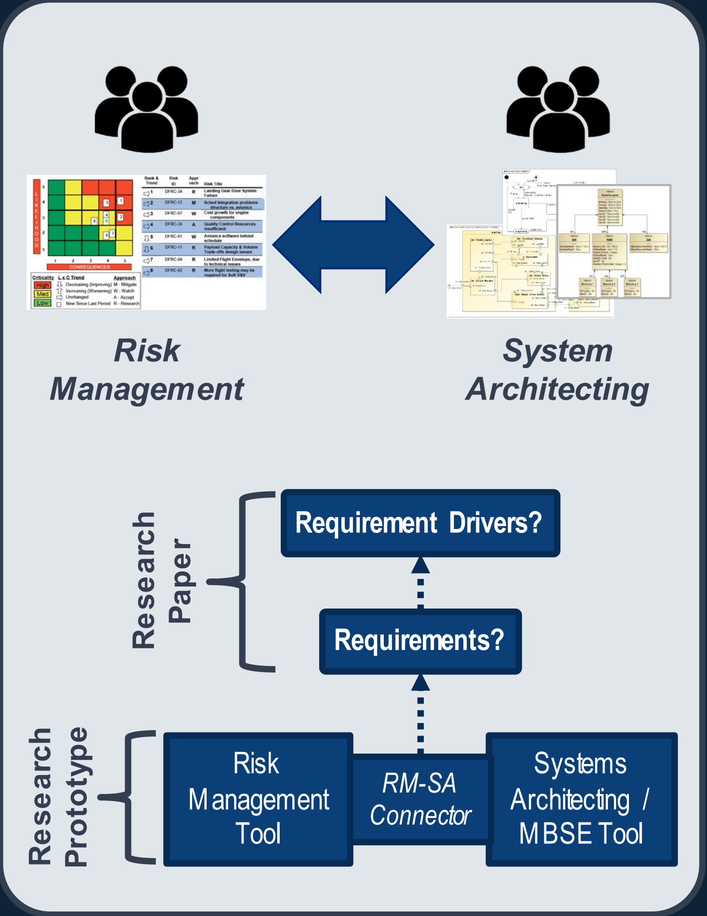
- A digital thread is
  - *"an extensible, configurable and component enterprise-level analytical framework that seamlessly expedites the controlled interplay of authoritative … information … by providing the capability to access, integrate and transform disparate data into actionable information"* (Defense Acquisition University, 2022)
- A digital thread can be understood as:
  - The coordination between domains …
    - ("*interplay of authoritative information*")
  - … executed in a seamless way …
    - ("*seamlessly expedites*")
  - … to take informed action upon.
    - ("*transform disparate data into actionable information*")



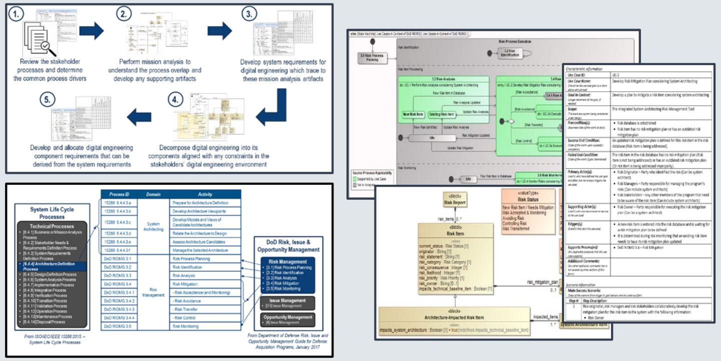
Research Background

Risk Management – System Architecting Digital Thread

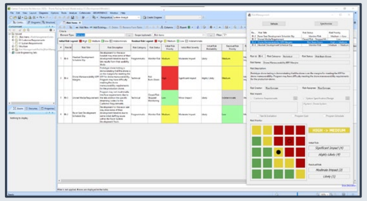
- Our research explores the digital thread between risk management and system architecting
- Research Paper:
  - Developed a methodology that:
    - Extracts the common domain needs from risk management and system architecting processes
    - Uses those needs to develop the implementable requirements for the tools to coordinate with each other
- Research Prototype:
  - Demonstrates an example of this digital thread concept using Cameo, Jira and other custom software

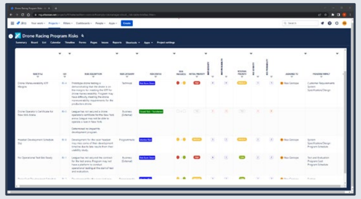


Research Paper (Methodology):
"Architecting Digital Engineering Requirements for Risk Management & Systems Architecting"
(2023 INCOSE International Symposium)



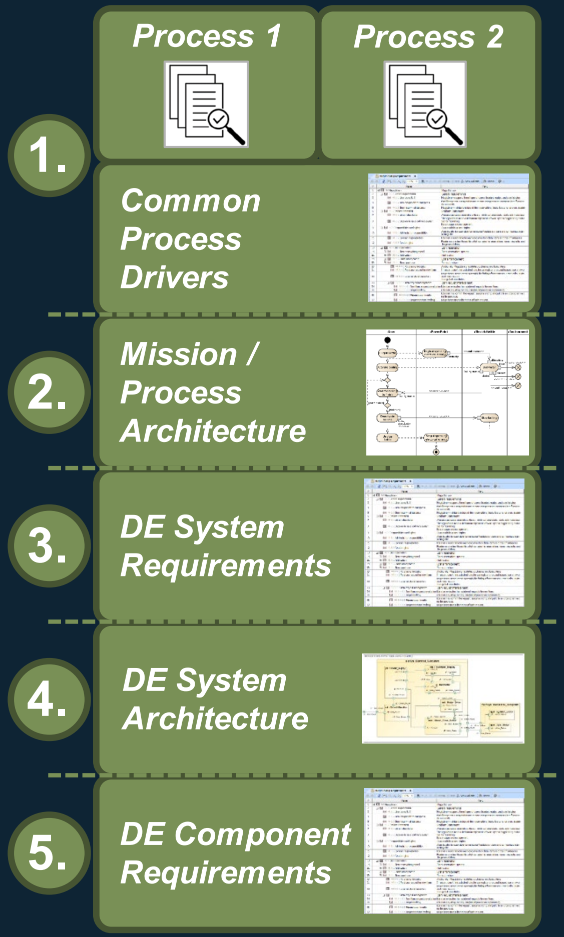
Research Prototype (Example Implementation):
"Understanding the Digital Thread between MBSE and Program Risk Management" (Vitech Integrate23)





Technical Approach Overview

- We developed a systems engineering methodology that takes process definitions and methodically develops digital engineering component requirements:
  1. Review the processes of the domains of interest to the stakeholders and determine the common process drivers
  2. Perform mission analysis to understand the technical overlap between the processes and develop any supporting artifacts that specify this overlap (e.g. use cases, black-box models, data models, etc.)
  3. Develop system requirements for digital engineering which trace to these mission analysis artifacts
  4. Decompose digital engineering into its logical components aligned with any constraints in the stakeholders' digital engineering environment
  5. Develop and allocate digital engineering component requirements that can be derived from the system requirements
- We applied this methodology on the two domains of interest (system architecting and risk management), developed the appropriate artifacts, and documented our observations



Process 1

Process 2

1. Common Process Drivers

2. Mission / Process Architecture

3. DE System Requirements

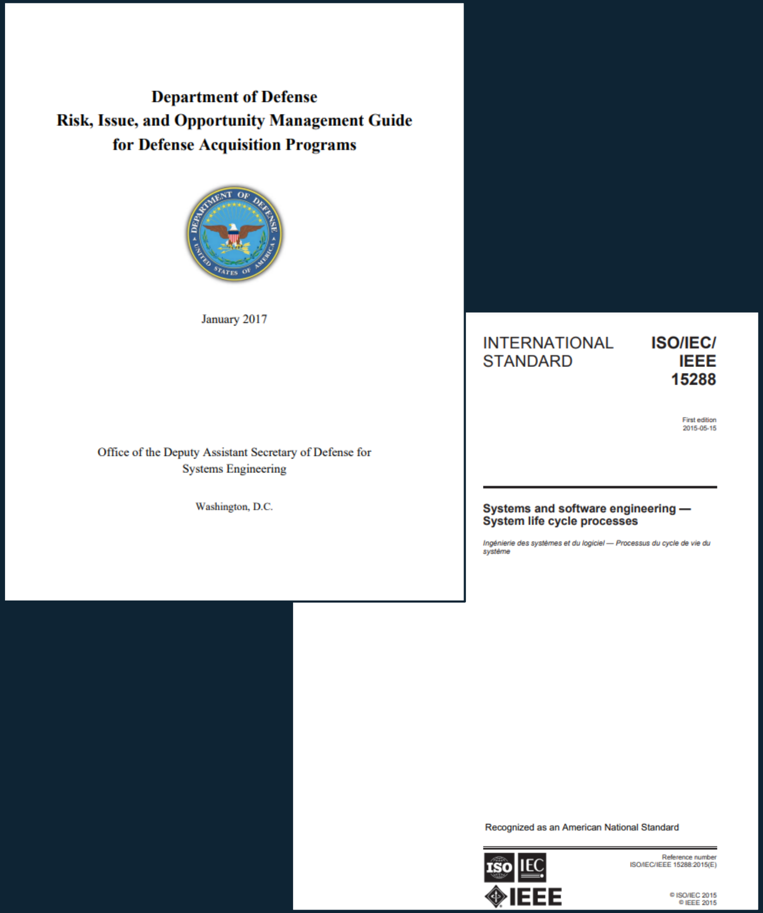
4. DE System Architecture

5. DE Component Requirements

1. Understand Processes and Determine Process Drivers

Execution Overview

- We researched the relevant processes for risk management and system engineering
- Risk Management Process: United States Department of Defense Risk, Issue, and Opportunity Management Guide for Defense Acquisition Programs January 2017 (DoD RIOMG)
  - Contains thorough process details for risk management
  - Holds a special interest with our stakeholders
- Systems Engineering Process: ISO/IEC/IEEE 15288:2015
  - Broad description of the systems engineering functions
  - Not project or program-specific
- We identified process activities from section 3 of the DoD RIOMG and section 6.4.4 of the 15288 to focus on risk management and system architecting



Department of Defense
Risk, Issue, and Opportunity Management Guide
for Defense Acquisition Programs

January 2017

Office of the Deputy Assistant Secretary of Defense for
Systems Engineering

Washington, D.C.

INTERNATIONAL STANDARD

ISO/IEC/
IEEE
15288

First edition
2015-05-15

Systems and software engineering —
System life cycle processes

Ingénierie des systèmes et du logiciel — Processus du cycle de vie du système

Recognized as an American National Standard

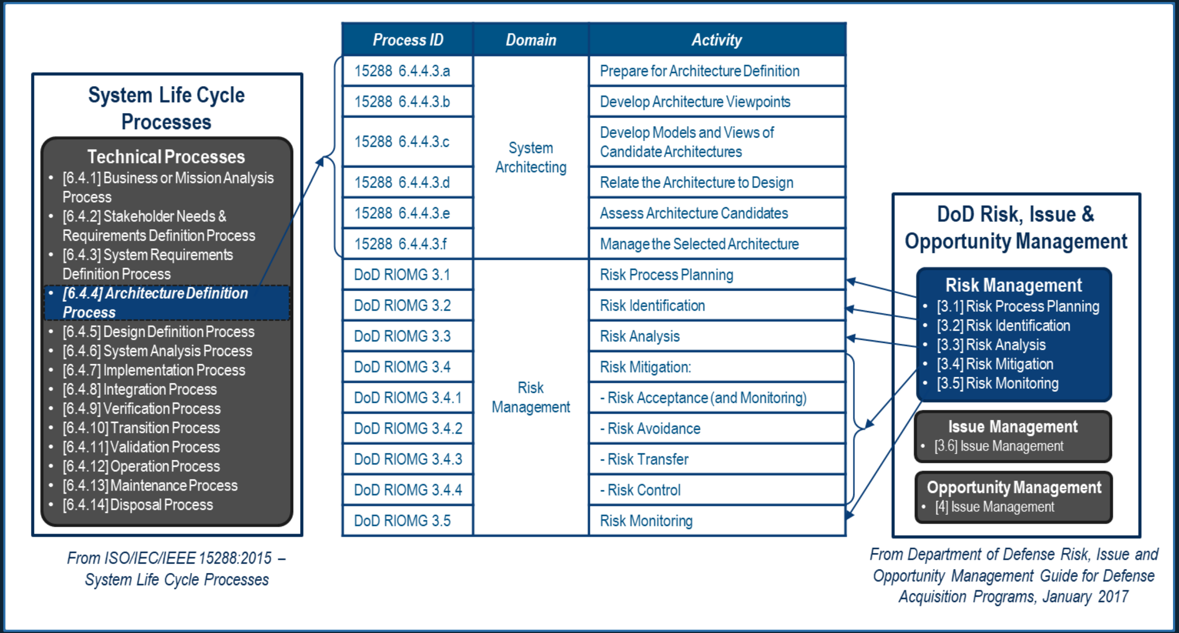
Reference number
ISO/IEC/IEEE 15288:2015(E)

© ISO/IEC 2015
© IEEE 2015

1. Understand Processes and Determine Process Drivers

Determining Processes of Interest and Common Process Drivers

- We developed some guidelines to consider when identifying the source processes:
  - Process Applicability
  - Consideration of Tool-Specific Influences to the Process
  - Stakeholder Interest/Need
- We identified process activities from the following:
  - Risk Management: Section 3 ("Risk Management") of the United States Department of Defense Risk, Issue, and Opportunity Management Guide for Defense Acquisition Programs January 2017 (DoD RIOMG)
  - System Architecting: Section 6.4.4 ("Architecture Definition Process") of the ISO/IEC/IEEE 15288:2015
- These process activities became our main requirement drivers



System Life Cycle Processes

Technical Processes

- [6.4.1] Business or Mission Analysis Process
- [6.4.2] Stakeholder Needs & Requirements Definition Process
- [6.4.3] System Requirements Definition Process
- ***[6.4.4] Architecture Definition Process***
- [6.4.5] Design Definition Process
- [6.4.6] System Analysis Process
- [6.4.7] Implementation Process
- [6.4.8] Integration Process
- [6.4.9] Verification Process
- [6.4.10] Transition Process
- [6.4.11] Validation Process
- [6.4.12] Operation Process
- [6.4.13] Maintenance Process
- [6.4.14] Disposal Process

From ISO/IEC/IEEE 15288:2015 – System Life Cycle Processes

Process ID	*Domain*	*Activity*
15288 6.4.4.3.a	System Architecting	Prepare for Architecture Definition
15288 6.4.4.3.b		Develop Architecture Viewpoints
15288 6.4.4.3.c		Develop Models and Views of Candidate Architectures
15288 6.4.4.3.d		Relate the Architecture to Design
15288 6.4.4.3.e		Assess Architecture Candidates
15288 6.4.4.3.f		Manage the Selected Architecture
DoD RIOMG 3.1	Risk Management	Risk Process Planning
DoD RIOMG 3.2		Risk Identification
DoD RIOMG 3.3		Risk Analysis
DoD RIOMG 3.4		Risk Mitigation:
DoD RIOMG 3.4.1		- Risk Acceptance (and Monitoring)
DoD RIOMG 3.4.2		- Risk Avoidance
DoD RIOMG 3.4.3		- Risk Transfer
DoD RIOMG 3.4.4		- Risk Control
DoD RIOMG 3.5		Risk Monitoring

DoD Risk, Issue & Opportunity Management

Risk Management

- [3.1] Risk Process Planning
- [3.2] Risk Identification
- [3.3] Risk Analysis
- [3.4] Risk Mitigation
- [3.5] Risk Monitoring

Issue Management

- [3.6] Issue Management

Opportunity Management

- [4] Issue Management

From Department of Defense Risk, Issue and Opportunity Management Guide for Defense Acquisition Programs, January 2017

2. Perform Mission Analysis to Understand Process Overlap

Execution Overview

- We performed use case analysis on the identified process activities and developed common use cases that support those process activities
- We assessed the use cases to determine behavioral interdependencies and common information
- We identified a system scope and applicable user roles

2. Perform Mission Analysis to Understand Process Overlap

Use Cases

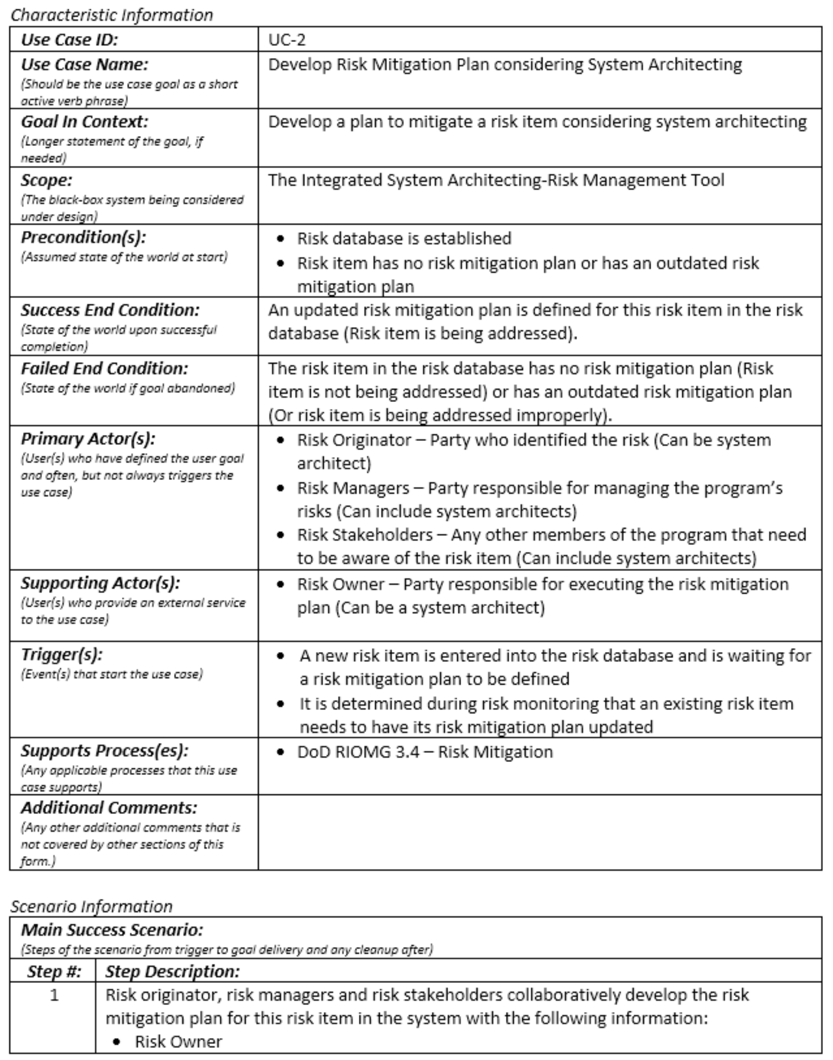
- We identified six use cases from the process activities
- A valid use case supported one of two main goals for the digital thread being designed:
  - The risk manager wants the system engineer to account for risk designing a system
  - The system engineer wants the risk manager to account for system architecture concerns when managing risk
- Many of the use cases are existing process activities with modifications to support the other domain
- We did not develop use cases for process activities that can be executed by one domain on their own
  - e.g. Risk management does not need system architecting to execute the Risk Acceptance Activity

Use Case ID	*Name*	*Description*	*Supports Process Activity*
UC-1	Perform Risk Analysis considering System Architecting	Perform risk analysis on a risk item considering system architecting	DoD RIOMG 3.3
UC-2	Develop Risk Mitigation Plan considering System Architecting	Develop a plan to mitigate a risk item considering system architecting	DoD RIOMG 3.4
UC-3a	Execute Risk Control Plan considering System Architecting	Execute the risk control plan on a risk item considering system architecting	DoD RIOMG 3.4.4
UC-3b	Execute Risk Avoidance Plan considering System Architecting	Execute the risk avoidance plan on a risk item considering system architecting	DoD RIOMG 3.4.2
UC-4	Monitor Risks considering System Architecting	Monitor the risk database with considering system architecting	DoD RIOMG 3.5
UC-5	Perform System Architecting with Risk Management	Designing the system architecture with considering risk management	15288 6.4.4.3.c 15288 6.4.4.3.d 15288 6.4.4.3.e

2. Perform Mission Analysis to Understand Process Overlap

Use Case Descriptions

- For each of the six use cases, we developed detailed use case descriptions
  - Attributes are mostly from Alistair Cockburn's "Writing Effective Use Cases" with some modifications to support this specific methodology:
    - Use Case ID
    - Use Case Name
    - Goal in Context
    - Scope
    - Precondition(s)
    - Success End Condition
    - Failed End Condition
    - Primary Actor(s)
    - Supporting Actor(s)
    - Trigger(s)
    - Supports Process(es)
    - Additional Comments
    - Main Success Scenario with Steps and Step Description
    - Extensions with Affected Steps, Conditions and Branching Actions
    - Subvariations with Affected Steps and Branching Actions



Characteristic Information

Use Case ID:	UC-2
Use Case Name: *(Should be the use case goal as a short active verb phrase)*	Develop Risk Mitigation Plan considering System Architecting
Goal In Context: *(Longer statement of the goal, if needed)*	Develop a plan to mitigate a risk item considering system architecting
Scope: *(The black-box system being considered under design)*	The Integrated System Architecting-Risk Management Tool
Precondition(s): *(Assumed state of the world at start)*	• Risk database is established • Risk item has no risk mitigation plan or has an outdated risk mitigation plan
Success End Condition: *(State of the world upon successful completion)*	An updated risk mitigation plan is defined for this risk item in the risk database (Risk item is being addressed).
Failed End Condition: *(State of the world if goal abandoned)*	The risk item in the risk database has no risk mitigation plan (Risk item is not being addressed) or has an outdated risk mitigation plan (Or risk item is being addressed improperly).
Primary Actor(s): *(User(s) who have defined the user goal and often, but not always triggers the use case)*	• Risk Originator – Party who identified the risk (Can be system architect) • Risk Managers – Party responsible for managing the program's risks (Can include system architects) • Risk Stakeholders – Any other members of the program that need to be aware of the risk item (Can include system architects)
Supporting Actor(s): *(User(s) who provide an external service to the use case)*	• Risk Owner – Party responsible for executing the risk mitigation plan (Can be a system architect)
Trigger(s): *(Event(s) that start the use case)*	• A new risk item is entered into the risk database and is waiting for a risk mitigation plan to be defined • It is determined during risk monitoring that an existing risk item needs to have its risk mitigation plan updated
Supports Process(es): *(Any applicable processes that this use case supports)*	• DoD RIOMG 3.4 – Risk Mitigation
Additional Comments: *(Any other additional comments that is not covered by other sections of this form.)*	

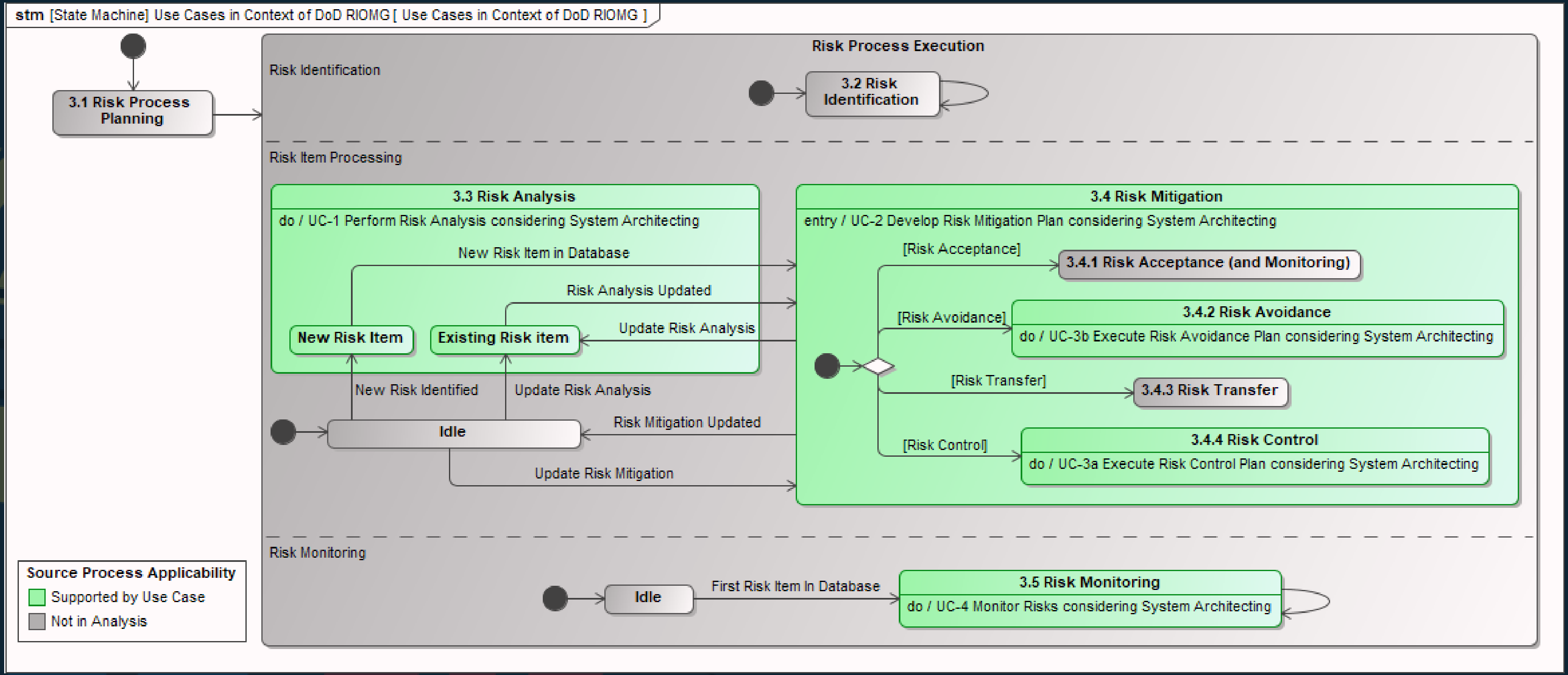
Scenario Information

Main Success Scenario: *(Steps of the scenario from trigger to goal delivery and any cleanup after)*	
Step #:	***Step Description:***
1	Risk originator, risk managers and risk stakeholders collaboratively develop the risk mitigation plan for this risk item in the system with the following information: • Risk Owner

2. Perform Mission Analysis to Understand Process Overlap

Use Case Functional Interdependencies and State Machine

- The following State Machine helped us understand some interdependencies
  - Most of the use cases are driven from risk management (DoD RIOMG)
  - The use cases are functionally interrelated to each other

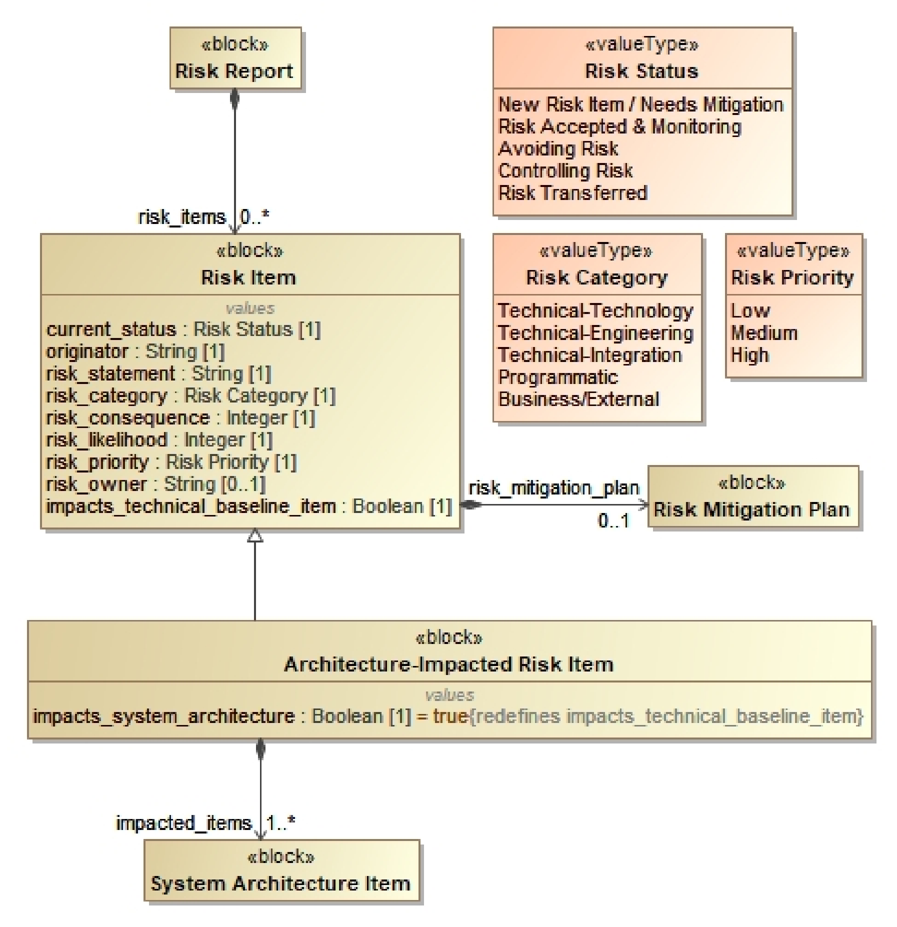


stm [State Machine] Use Cases in Context of DoD RIOMG [Use Cases in Context of DoD RIOMG]
3.1 Risk Process Planning
Risk Process Execution
Risk Identification
3.2 Risk Identification
Risk Item Processing
3.3 Risk Analysis
do / UC-1 Perform Risk Analysis considering System Architecting
New Risk Item in Database
Risk Analysis Updated
Update Risk Analysis
New Risk Item
Existing Risk item
New Risk Identified
Update Risk Analysis
Idle
Risk Mitigation Updated
Update Risk Mitigation
3.4 Risk Mitigation
entry / UC-2 Develop Risk Mitigation Plan considering System Architecting
[Risk Acceptance]
3.4.1 Risk Acceptance (and Monitoring)
[Risk Avoidance]
3.4.2 Risk Avoidance
do / UC-3b Execute Risk Avoidance Plan considering System Architecting
[Risk Transfer]
3.4.3 Risk Transfer
[Risk Control]
3.4.4 Risk Control
do / UC-3a Execute Risk Control Plan considering System Architecting
Risk Monitoring
Idle
First Risk Item In Database
3.5 Risk Monitoring
do / UC-4 Monitor Risks considering System Architecting
Source Process Applicability
🟩 Supported by Use Case
▨ Not in Analysis

2. Perform Mission Analysis to Understand Process Overlap

Information Model and Architecture-Impacted Risk Item

- The other major place where there is process overlap is in the information model
- Two main pieces of information where the two domains meet:
  - Risk Item – Main information element in Risk Management
  - System Architecture Item – An information element in System Architecting
- Developed a new information element, Architecture-Impacted Risk Item:
  - Specific type of risk item, includes all of the attributes of a regular risk item
  - Also includes two additional attributes:
    - Impacts System Architecture – Indicator that the risk item has some impact to the system architecture
    - Impacted Items – System architecture items that the risk item impacts

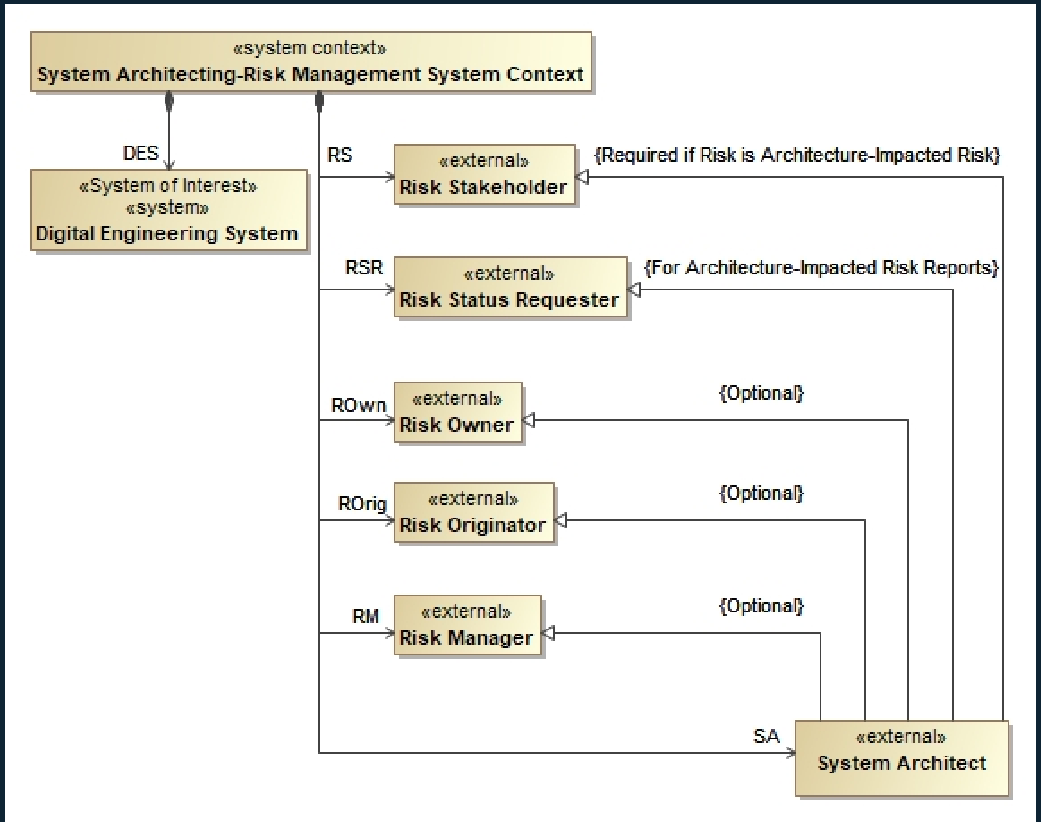


«block»
Risk Report
«valueType»
Risk Status
New Risk Item / Needs Mitigation
Risk Accepted & Monitoring
Avoiding Risk
Controlling Risk
Risk Transferred
risk_items
0..*
«block»
Risk Item
values
current_status : Risk Status [1]
originator : String [1]
risk_statement : String [1]
risk_category : Risk Category [1]
risk_consequence : Integer [1]
risk_likelihood : Integer [1]
risk_priority : Risk Priority [1]
risk_owner : String [0..1]
impacts_technical_baseline_item : Boolean [1]
«valueType»
Risk Category
Technical-Technology
Technical-Engineering
Technical-Integration
Programmatic
Business/External
«valueType»
Risk Priority
Low
Medium
High
risk_mitigation_plan
0..1
«block»
Risk Mitigation Plan
«block»
Architecture-Impacted Risk Item
values
impacts_system_architecture : Boolean [1] = true{redefines impacts_technical_baseline_item}
impacted_items
1..*
«block»
System Architecture Item

2. Perform Mission Analysis to Understand Process Overlap

Black-Box System Context & User Roles

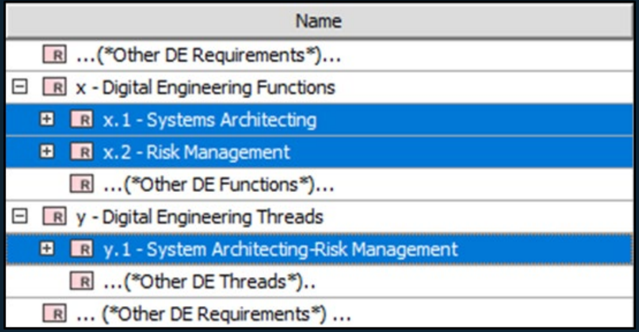
- Developed a black-box system context to understand the high-level system scope
- For the system context, we explicitly called out *Digital Engineering System* as the system-of-interest for two reasons:
  1. Enables us to explicitly call out *“The Digital Engineering System shall…”* for the system requirements later in the methodology
  2. Wanted to define the system-part of the Digital Engineering Ecosystem
- User roles:
  - Risk Management Domain: Risk Stakeholder, Risk Status Requester, Risk Owner, Risk Originator and Risk Manager
  - System Architecting Domain: System Architect
    - The System Architect can also serve the roles within the risk management domain depending on the use case



3. Develop Digital Engineering System Requirements

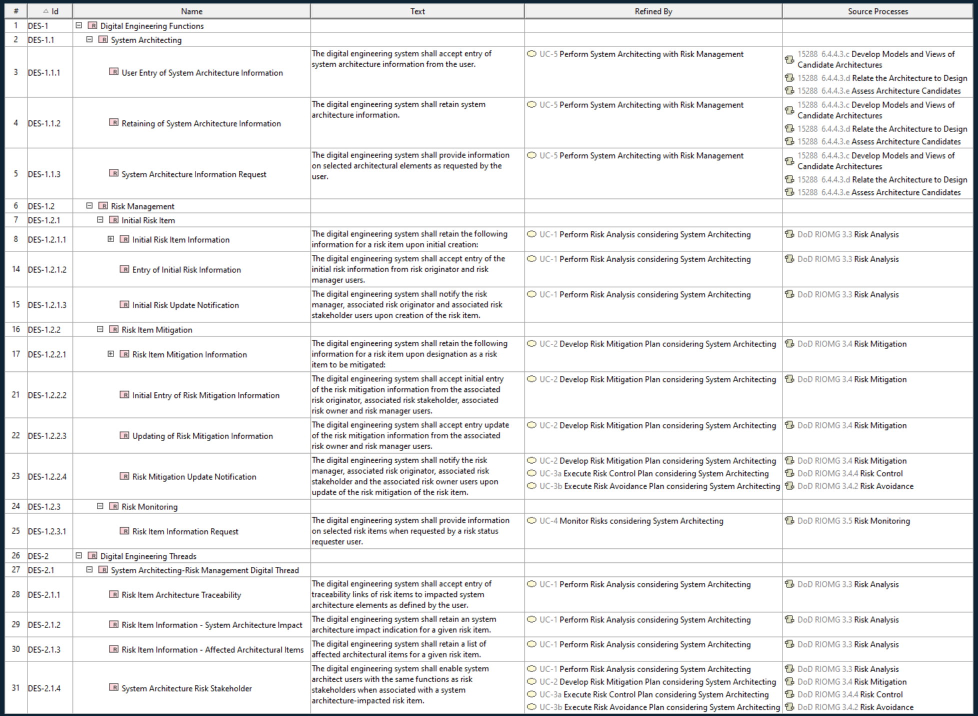
Execution Overview & System Requirement Structure

- Execution Overview:
  - We developed a set of system requirements for Digital Engineering that supports these use cases
  - We wrote these requirements as "The Digital Engineering System shall …"
- Structured our system requirements into the following groups:
  - Digital Engineering Functions
    - System Architecting
    - Risk Management
  - Digital Engineering Threads
    - System Architecting-Risk Management
- Structured the requirements this way so that we could easily integrate these requirements with other Digital Engineering system requirements



Name

- R …(*Other DE Requirements*)…
- ⊟ R x - Digital Engineering Functions
  - ⊞ R x.1 - Systems Architecting
  - ⊞ R x.2 - Risk Management
  - R …(*Other DE Functions*)…
- ⊟ R y - Digital Engineering Threads
  - ⊞ R y.1 - System Architecting-Risk Management
  - R …(*Other DE Threads*)..
- R … (*Other DE Requirements*) …

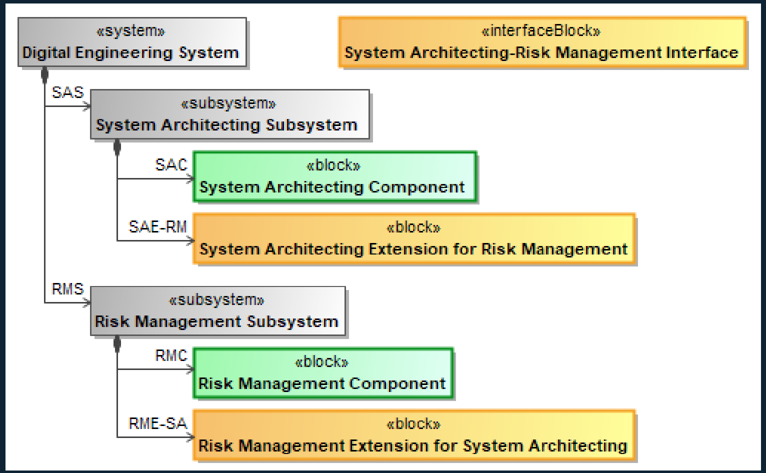


#	Id	Name	Text	Refined By	Source Processes
1	DES-1	⊟ R Digital Engineering Functions			
2	DES-1.1	⊟ R System Architecting			
3	DES-1.1.1	R User Entry of System Architecture Information	The digital engineering system shall accept entry of system architecture information from the user.	○ UC-5 Perform System Architecting with Risk Management	15288 6.4.4.3.c Develop Models and Views of Candidate Architectures 15288 6.4.4.3.d Relate the Architecture to Design 15288 6.4.4.3.e Assess Architecture Candidates
4	DES-1.1.2	R Retaining of System Architecture Information	The digital engineering system shall retain system architecture information.	○ UC-5 Perform System Architecting with Risk Management	15288 6.4.4.3.c Develop Models and Views of Candidate Architectures 15288 6.4.4.3.d Relate the Architecture to Design 15288 6.4.4.3.e Assess Architecture Candidates
5	DES-1.1.3	R System Architecture Information Request	The digital engineering system shall provide information on selected architectural elements as requested by the user.	○ UC-5 Perform System Architecting with Risk Management	15288 6.4.4.3.c Develop Models and Views of Candidate Architectures 15288 6.4.4.3.d Relate the Architecture to Design 15288 6.4.4.3.e Assess Architecture Candidates
6	DES-1.2	⊟ R Risk Management			
7	DES-1.2.1	⊟ R Initial Risk Item			
8	DES-1.2.1.1	⊞ R Initial Risk Item Information	The digital engineering system shall retain the following information for a risk item upon initial creation:	○ UC-1 Perform Risk Analysis considering System Architecting	DoD RIOMG 3.3 Risk Analysis
14	DES-1.2.1.2	R Entry of Initial Risk Information	The digital engineering system shall accept entry of the initial risk information from risk originator and risk manager users.	○ UC-1 Perform Risk Analysis considering System Architecting	DoD RIOMG 3.3 Risk Analysis
15	DES-1.2.1.3	R Initial Risk Update Notification	The digital engineering system shall notify the risk manager, associated risk originator and associated risk stakeholder users upon creation of the risk item.	○ UC-1 Perform Risk Analysis considering System Architecting	DoD RIOMG 3.3 Risk Analysis
16	DES-1.2.2	⊟ R Risk Item Mitigation			
17	DES-1.2.2.1	⊞ R Risk Item Mitigation Information	The digital engineering system shall retain the following information for a risk item upon designation as a risk item to be mitigated:	○ UC-2 Develop Risk Mitigation Plan considering System Architecting	DoD RIOMG 3.4 Risk Mitigation
21	DES-1.2.2.2	R Initial Entry of Risk Mitigation Information	The digital engineering system shall accept initial entry of the risk mitigation information from the associated risk originator, associated risk stakeholder, associated risk owner and risk manager users.	○ UC-2 Develop Risk Mitigation Plan considering System Architecting	DoD RIOMG 3.4 Risk Mitigation
22	DES-1.2.2.3	R Updating of Risk Mitigation Information	The digital engineering system shall accept entry update of the risk mitigation information from the associated risk owner and risk manager users.	○ UC-2 Develop Risk Mitigation Plan considering System Architecting	DoD RIOMG 3.4 Risk Mitigation
23	DES-1.2.2.4	R Risk Mitigation Update Notification	The digital engineering system shall notify the risk manager, associated risk originator, associated risk stakeholder and the associated risk owner users upon update of the risk mitigation of the risk item.	○ UC-2 Develop Risk Mitigation Plan considering System Architecting ○ UC-3a Execute Risk Control Plan considering System Architecting ○ UC-3b Execute Risk Avoidance Plan considering System Architecting	DoD RIOMG 3.4 Risk Mitigation DoD RIOMG 3.4.4 Risk Control DoD RIOMG 3.4.2 Risk Avoidance
24	DES-1.2.3	⊟ R Risk Monitoring			
25	DES-1.2.3.1	R Risk Item Information Request	The digital engineering system shall provide information on selected risk items when requested by a risk status requester user.	○ UC-4 Monitor Risks considering System Architecting	DoD RIOMG 3.5 Risk Monitoring
26	DES-2	⊟ R Digital Engineering Threads			
27	DES-2.1	⊟ R System Architecting-Risk Management Digital Thread			
28	DES-2.1.1	R Risk Item Architecture Traceability	The digital engineering system shall accept entry of traceability links of risk items to impacted system architecture elements as defined by the user.	○ UC-1 Perform Risk Analysis considering System Architecting	DoD RIOMG 3.3 Risk Analysis
29	DES-2.1.2	R Risk Item Information - System Architecture Impact	The digital engineering system shall retain an system architecture impact indication for a given risk item.	○ UC-1 Perform Risk Analysis considering System Architecting	DoD RIOMG 3.3 Risk Analysis
30	DES-2.1.3	R Risk Item Information - Affected Architectural Items	The digital engineering system shall retain a list of affected architectural items for a given risk item.	○ UC-1 Perform Risk Analysis considering System Architecting	DoD RIOMG 3.3 Risk Analysis
31	DES-2.1.4	R System Architecture Risk Stakeholder	The digital engineering system shall enable system architect users with the same functions as risk stakeholders when associated with a system architecture-impacted risk item.	○ UC-1 Perform Risk Analysis considering System Architecting ○ UC-2 Develop Risk Mitigation Plan considering System Architecting ○ UC-3a Execute Risk Control Plan considering System Architecting ○ UC-3b Execute Risk Avoidance Plan considering System Architecting	DoD RIOMG 3.3 Risk Analysis DoD RIOMG 3.4 Risk Mitigation DoD RIOMG 3.4.4 Risk Control DoD RIOMG 3.4.2 Risk Avoidance

4. Decompose Digital Engineering into its Components

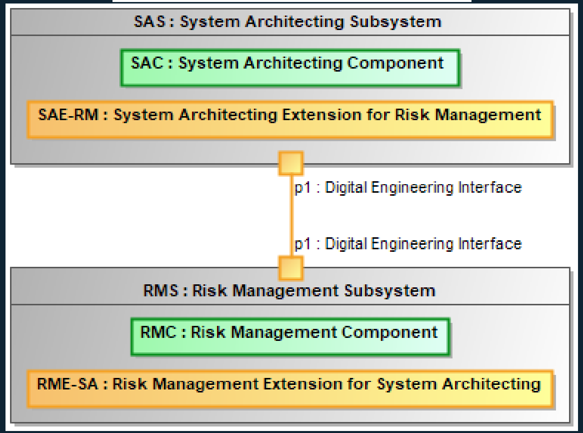
Execution Overview & Digital Engineering Logical Architecture

- Execution Overview
  - Developed a digital engineering logical architecture and decomposed the logical architecture into its components
  - We considered any alternative digital engineering logical architectures
- We developed a digital engineering logical architecture with the following components:
  - System Architecting Subsystem
    - System Architecting Component
    - System Architecting Extension for Risk Management
  - Risk Management Subsystem
    - Risk Management Component
    - Risk Management Extension for System Architecting
  - Subsystem Interfaces
    - System Architecting-Risk Management Interface
- A few notes on this architecture
  - This architecture implies using existing software tools to perform the base components (system architecting and risk management components)
  - This architecture implies developing custom software for the extensions and the interface


«system»
Digital Engineering System
«interfaceBlock»
System Architecting-Risk Management Interface
SAS
«subsystem»
System Architecting Subsystem
SAC
«block»
System Architecting Component
SAE-RM
«block»
System Architecting Extension for Risk Management
RMS
«subsystem»
Risk Management Subsystem
RMC
«block»
Risk Management Component
RME-SA
«block»
Risk Management Extension for System Architecting

Expected Component Implementation

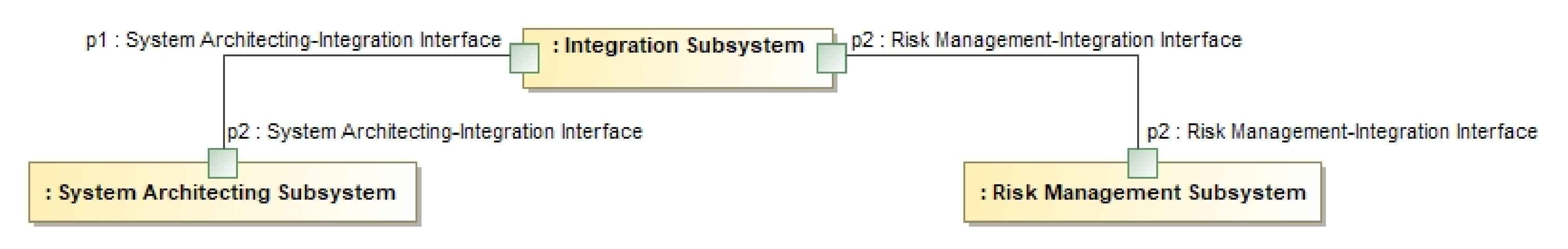
- 🟩 Purchase or Reuse Existing Software Tool
- 🟧 Develop Custom Software
- ⬜ Not Component / Not Applicable


SAS : System Architecting Subsystem
SAC : System Architecting Component
SAE-RM : System Architecting Extension for Risk Management
p1 : Digital Engineering Interface
p1 : Digital Engineering Interface
RMS : Risk Management Subsystem
RMC : Risk Management Component
RME-SA : Risk Management Extension for System Architecting

4. Decompose Digital Engineering into its Components

Alternative Logical Architectures

- There are a few main considerations with our chosen architecture:
  - Wanted existing tools that our stakeholders have if available
  - The stakeholders do not mind the point-to-point nature of the architecture right now as long as we got a working capability
- There can be alternative architectures depending on the stakeholders' requirements and environment limitations
  - Could use an integration platform with some custom modifications to get a functional capability
  - Will revisit this alternative architecture considering other tools

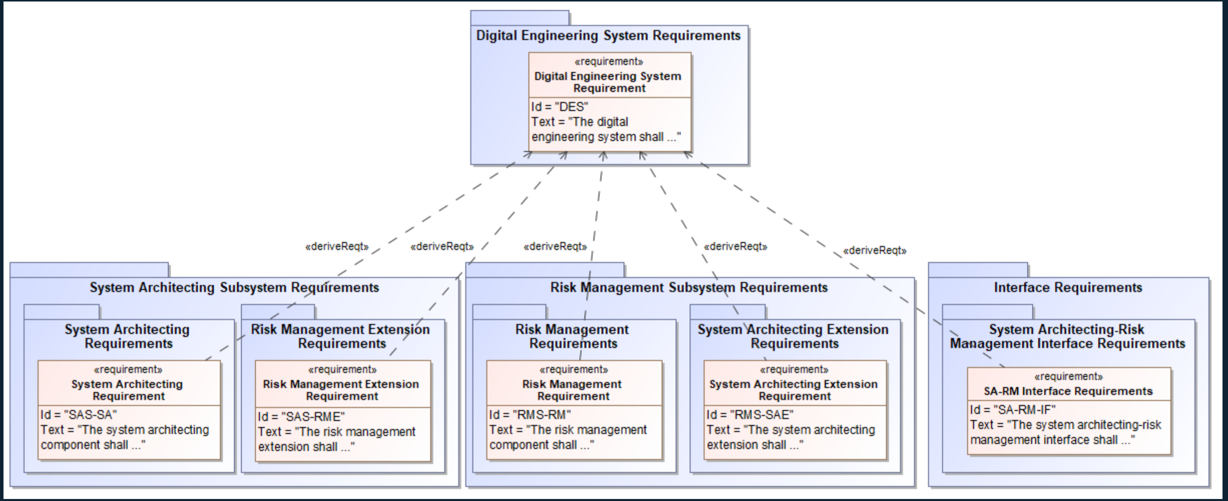

p1 : System Architecting-Integration Interface
: Integration Subsystem
p2 : Risk Management-Integration Interface
p2 : System Architecting-Integration Interface
p2 : Risk Management-Integration Interface
: System Architecting Subsystem
: Risk Management Subsystem

5. Develop Digital Engineering Component Requirements

Execution Overview & Component Requirement Sets

- Developed sets of component requirements that are allocated to the digital engineering components based upon the structure of our logical architecture
  - Wrote these requirements as *“The <*Digital Engineering Component*> shall …”*
    - e.g. *“The System Architecting Component shall …”*
- Structured component requirements into requirement sets based upon the structure of our digital engineering logical architecture

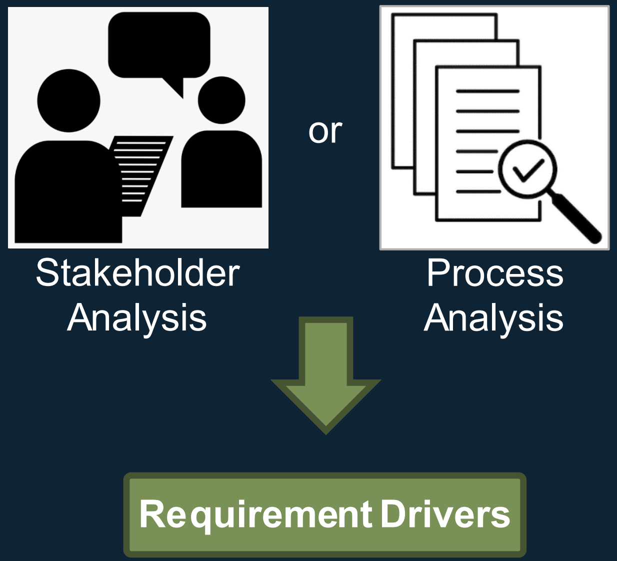
Requirement Set	*Requirement Set Type*	*Purpose*
System Architecting Component Requirements	Component Requirements	To assess an existing software tool for purchase/reuse
System Architecting Extension for Risk Management Component Requirements	Component Requirements	To develop new custom software
Risk Management Component Requirements	Component Requirements	To assess an existing software tool for purchase/reuse
Risk Management Extension for System Architecting Component Requirements	Component Requirements	To develop new custom software
System Architecting-Risk Management Interface	Interface Requirements	To develop new custom software


Digital Engineering System Requirements
«requirement»
Digital Engineering System Requirement
Id = "DES"
Text = "The digital engineering system shall ..."
«deriveReqt»
«deriveReqt»
«deriveReqt»
«deriveReqt»
«deriveReqt»
System Architecting Subsystem Requirements
System Architecting Requirements
«requirement»
System Architecting Requirement
Id = "SAS-SA"
Text = "The system architecting component shall ..."
Risk Management Extension Requirements
«requirement»
Risk Management Extension Requirement
Id = "SAS-RME"
Text = "The risk management extension shall ..."
Risk Management Subsystem Requirements
Risk Management Requirements
«requirement»
Risk Management Requirement
Id = "RMS-RM"
Text = "The risk management component shall ..."
System Architecting Extension Requirements
«requirement»
System Architecting Extension Requirement
Id = "RMS-SAE"
Text = "The system architecting extension shall ..."
Interface Requirements
System Architecting-Risk Management Interface Requirements
«requirement»
SA-RM Interface Requirements
Id = "SA-RM-IF"
Text = "The system architecting-risk management interface shall ..."

Other Methodology Execution Details

Alternative Methodology Steps – Stakeholder Analysis

- An alternative to the first methodology step is to perform stakeholder analysis to elicit the requirement drivers
- Focused on process analysis for a few different reasons:
  - The process documents were readily available and we were able to extract requirement drivers quickly
    - In-person stakeholder analysis takes time to interview and gather the relevant information
    - Stakeholder analysis would have needed to assess two sets of stakeholders (system architects and risk managers) requiring more time
  - There was the possibility of the stakeholders injecting their own biases and uncommon/non-standard approaches
- Plan to include stakeholder analysis to supplement and verify our analysis



Stakeholder Analysis

or

Process Analysis

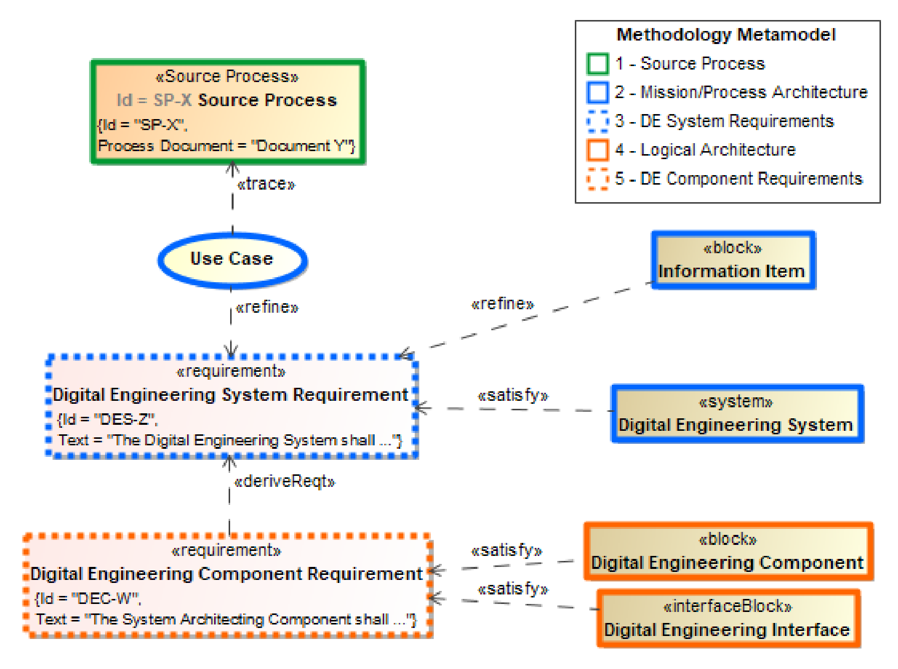
Requirement Drivers

Other Methodology Execution Details

Architecture Development and Artifact Traceability

- Defined many of the artifacts using SysML and Cameo Systems Modeler
  - Applied principles of using MBSE to design the Digital Engineering system
  - Our methodology does not require using MBSE principles in order to develop the artifacts

- Traced the artifacts with a structured metamodel
  - We can trace any component requirement through the artifacts all the way back to the source processes
  - We took an MBSE approach to developing the artifacts to be able to do the traceability



Methodology Metamodel
🟩 1 - Source Process
🟦 2 - Mission/Process Architecture
⬚ 3 - DE System Requirements
🟧 4 - Logical Architecture
⬚ 5 - DE Component Requirements

«Source Process»
Id = SP-X Source Process
{Id = "SP-X",
Process Document = "Document Y"}

«trace»

Use Case

«refine»

«refine»

«block»
Information Item

«requirement»
Digital Engineering System Requirement
{Id = "DES-Z",
Text = "The Digital Engineering System shall ..."}

«satisfy»

«system»
Digital Engineering System

«deriveReqt»

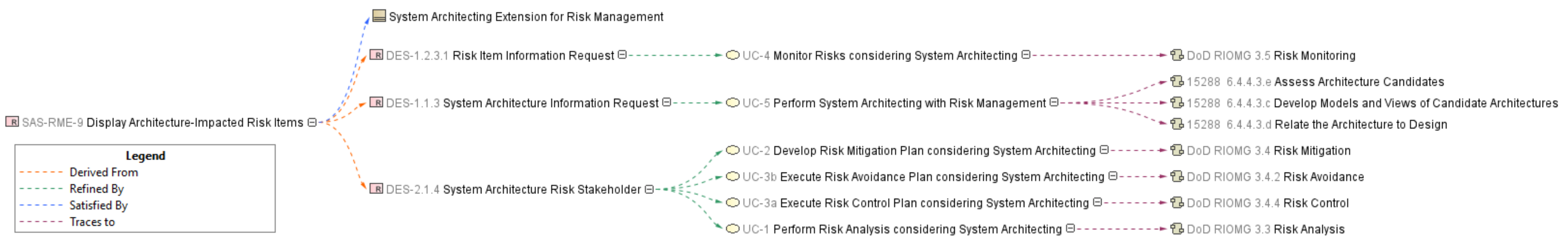
«requirement»
Digital Engineering Component Requirement
{Id = "DEC-W",
Text = "The System Architecting Component shall ..."}

«satisfy»

«block»
Digital Engineering Component

«satisfy»

«interfaceBlock»
Digital Engineering Interface



System Architecting Extension for Risk Management

SAS-RME-9 Display Architecture-Impacted Risk Items

DES-1.2.3.1 Risk Item Information Request

DES-1.1.3 System Architecture Information Request

DES-2.1.4 System Architecture Risk Stakeholder

UC-4 Monitor Risks considering System Architecting

UC-5 Perform System Architecting with Risk Management

UC-2 Develop Risk Mitigation Plan considering System Architecting

UC-3b Execute Risk Avoidance Plan considering System Architecting

UC-3a Execute Risk Control Plan considering System Architecting

UC-1 Perform Risk Analysis considering System Architecting

DoD RIOMG 3.5 Risk Monitoring

15288 6.4.4.3.e Assess Architecture Candidates

15288 6.4.4.3.c Develop Models and Views of Candidate Architectures

15288 6.4.4.3.d Relate the Architecture to Design

DoD RIOMG 3.4 Risk Mitigation

DoD RIOMG 3.4.2 Risk Avoidance

DoD RIOMG 3.4.4 Risk Control

DoD RIOMG 3.3 Risk Analysis

Legend
Derived From
Refined By
Satisfied By
Traces to

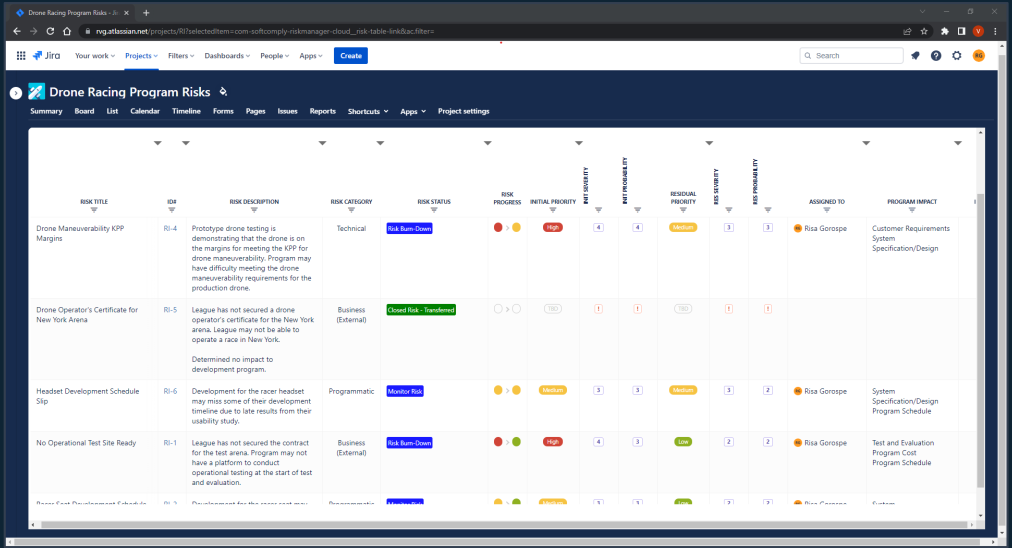
Research Paper Conclusions

- We demonstrated that we can develop digital engineering requirements from process documentation in a methodical way
  - We were able to develop digital engineering component requirements to assess existing digital engineering software tools or if developing digital engineering custom software is required
  - We accomplished this through a systems engineering approach that's traced to the requirement drivers from the source documentation
- This methodology has flexibility and can be tailored in a few different ways to meet the needs of the stakeholders
  - We could choose different domains for a digital thread to apply this methodology
  - Within the two selected domains, we could choose different process documents to develop requirement drivers
  - Within the selected process documents, the technical overlap between the two domains varies which effects the downstream architecture and requirements
  - Depending on the stakeholder constraints on the digital engineering environment, we could have developed different logical architecture structures

Quick Overview of Research Prototype

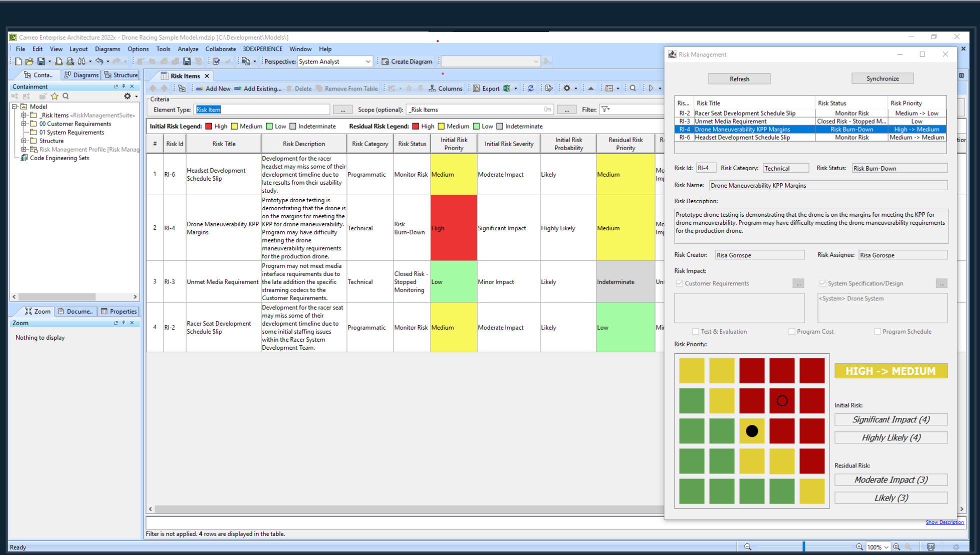
Research Prototype

- The research prototype is an example implementation of the system architecting-risk management digital thread concept
- For the risk management tool, the prototype uses Atlassian Jira with the SoftComply Risk Manager plugin and other customizations
- For the system architecture tool, the prototype uses Dassault Systemes' Cameo Systems Modeler with some profile and plugin customizations
- In this presentation, we will focus on the prototype example and the scenario
  - For a deep dive, review the Vitech Integrate23 presentation: *"Understanding the Digital Thread between MBSE and Program Risk Management"*



Drone Racing Program Risks

RISK TITLE	ID#	RISK DESCRIPTION	RISK CATEGORY	RISK STATUS	RISK PROGRESS	INITIAL PRIORITY	INIT SEVERITY	INIT PROBABILITY	RESIDUAL PRIORITY	RES SEVERITY	RES PROBABILITY	ASSIGNED TO	PROGRAM IMPACT
Drone Maneuverability KPP Margins	RI-4	Prototype drone testing is demonstrating that the drone is on the margins for meeting the KPP for drone maneuverability. Program may have difficulty meeting the drone maneuverability requirements for the production drone.	Technical	Risk Burn-Down	● > ●	High	4	4	Medium	3	3	Risa Gorospe	Customer Requirements System Specification/Design
Drone Operator's Certificate for New York Arena	RI-5	League has not secured a drone operator's certificate for the New York arena. League may not be able to operate a race in New York. Determined no impact to development program.	Business (External)	Closed Risk - Transferred	○ > ○	TBD	!	!	TBD	!	!		
Headset Development Schedule Slip	RI-6	Development for the racer headset may miss some of their development timeline due to late results from their usability study.	Programmatic	Monitor Risk	● > ●	Medium	3	3	Medium	3	2	Risa Gorospe	System Specification/Design Program Schedule
No Operational Test Site Ready	RI-1	League has not secured the contract for the test arena. Program may not have a platform to conduct operational testing at the start of test and evaluation.	Business (External)	Risk Burn-Down	● > ●	High	4	3	Low	2	2	Risa Gorospe	Test and Evaluation Program Cost Program Schedule



Criteria

Element Type: Risk Item

Scope (optional): _Risk Items

Filter:

Initial Risk Legend: 🟥 High 🟨 Medium 🟩 Low ⬜ Indeterminate **Residual Risk Legend:** 🟥 High 🟨 Medium 🟩 Low ⬜ Indeterminate

#	Risk Id	Risk Title	Risk Description	Risk Category	Risk Status	Initial Risk Priority	Initial Risk Severity	Initial Risk Probability	Residual Risk Priority
1	RI-6	Headset Development Schedule Slip	Development for the racer headset may miss some of their development timeline due to late results from their usability study.	Programmatic	Monitor Risk	Medium	Moderate Impact	Likely	Medium
2	RI-4	Drone Maneuverability KPP Margins	Prototype drone testing is demonstrating that the drone is on the margins for meeting the KPP for drone maneuverability. Program may have difficulty meeting the drone maneuverability requirements for the production drone.	Technical	Risk Burn-Down	High	Significant Impact	Highly Likely	Medium
3	RI-3	Unmet Media Requirement	Program may not meet media interface requirements due to the late addition the specific streaming codecs to the Customer Requirements.	Technical	Closed Risk - Stopped Monitoring	Low	Minor Impact	Likely	Indeterminate
4	RI-2	Racer Seat Development Schedule Slip	Development for the racer seat may miss some of their development timeline due to some initial staffing issues within the Racer System Development Team.	Programmatic	Monitor Risk	Medium	Moderate Impact	Likely	Low

Risk Management

Ris...	Risk Title	Risk Status	Risk Priority
RI-2	Racer Seat Development Schedule Slip	Monitor Risk	Medium -> Low
RI-3	Unmet Media Requirement	Closed Risk - Stopped M...	Low
RI-4	Drone Maneuverability KPP Margins	Risk Burn-Down	High -> Medium
RI-6	Headset Development Schedule Slip	Monitor Risk	Medium -> Medium

Risk Id: RI-4 Risk Category: Technical Risk Status: Risk Burn-Down

Risk Name: Drone Maneuverability KPP Margins

Risk Description:

Prototype drone testing is demonstrating that the drone is on the margins for meeting the KPP for drone maneuverability. Program may have difficulty meeting the drone maneuverability requirements for the production drone.

Risk Creator: Risa Gorospe Risk Assignee: Risa Gorospe

Risk Impact:

☑ Customer Requirements ☑ System Specification/Design

<System> Drone System

□ Test & Evaluation □ Program Cost □ Program Schedule

Risk Priority:

HIGH -> MEDIUM

Initial Risk:

Significant Impact (4)

Highly Likely (4)

Residual Risk:

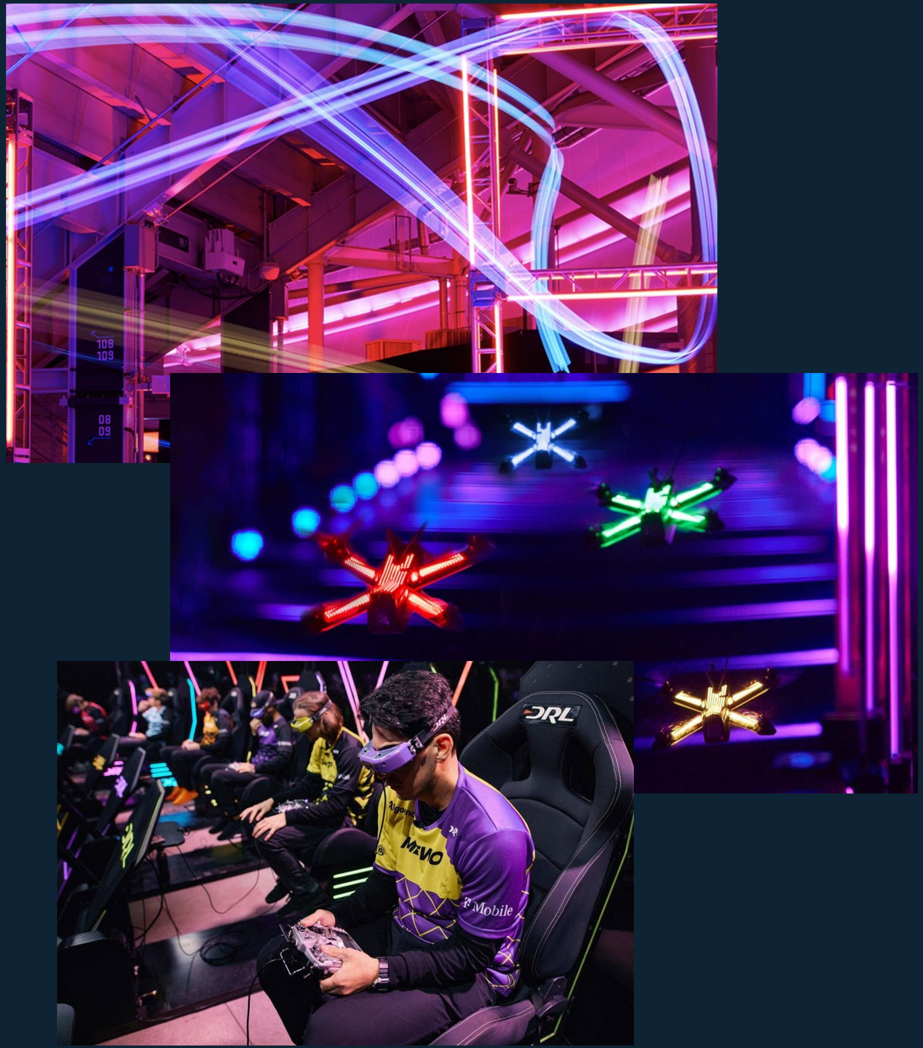
Moderate Impact (3)

Likely (3)

Quick Overview of Research Prototype

Research Prototype Example & Scenario

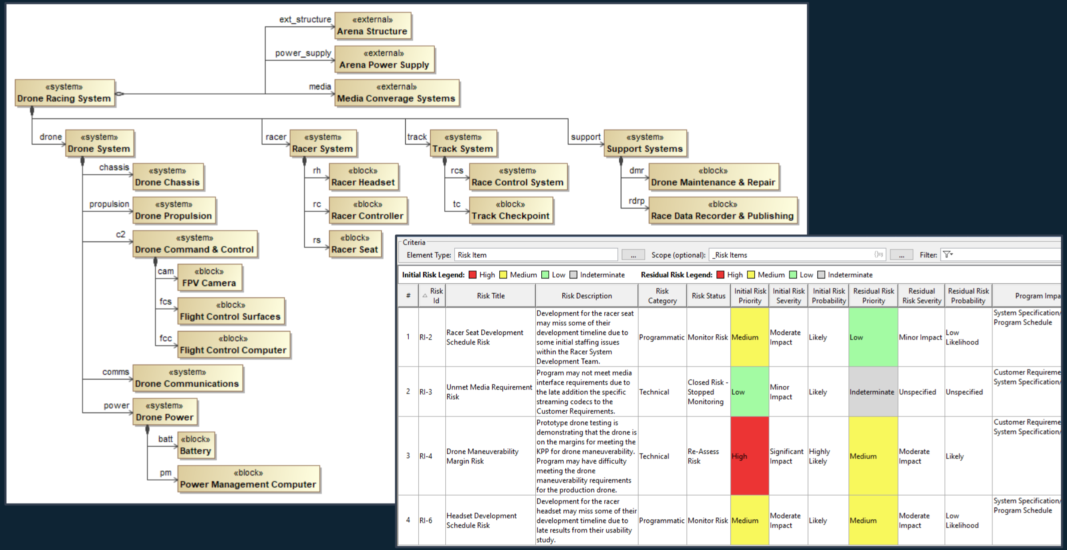
- The prototype example is a fictional scenario of developing a drone racing system:
  - A sports league who wants to start a first-person view (FPV) drone racing series
  - The sports league wants to create this as a spec-racing series:
    - The racers cannot use their own drones
    - Racers must purchase approved race drones from the league
  - We are the technical engineering firm contracted to develop this racing system
  - The league has contracted your firm to:
    - Develop both the drones and any supporting equipment (e.g. FPV headsets, race controller, drone charging stations, etc.)
    - Design, build and integrate from customer requirements to a complete delivered system



Quick Overview of Research Prototype

Research Prototype Example & Scenario

- We developed sample data for the prototype to execute the fictional scenario
  - We developed a Drone Racing System model in Cameo to define requirements and system architecture
  - We developed a database of program risk items in Jira
- When we applied this prototype to the fictional scenario, we realized that there is a program sweet spot for the system architecting-risk management digital thread
  - The program must be far enough along that the system architecture is initially defined with enough detail
  - But the program must be not too far along in detailed design where we may decide to trace to specific design documentation rather than the architecture itself
    - e.g. trace to the electrical CAD of a battery rather than the system architecture element of the battery
- For this example, we are somewhere in between System Requirements Review (SRR) and Preliminary Design Review (PDR) in a traditional DoD program lifecycle



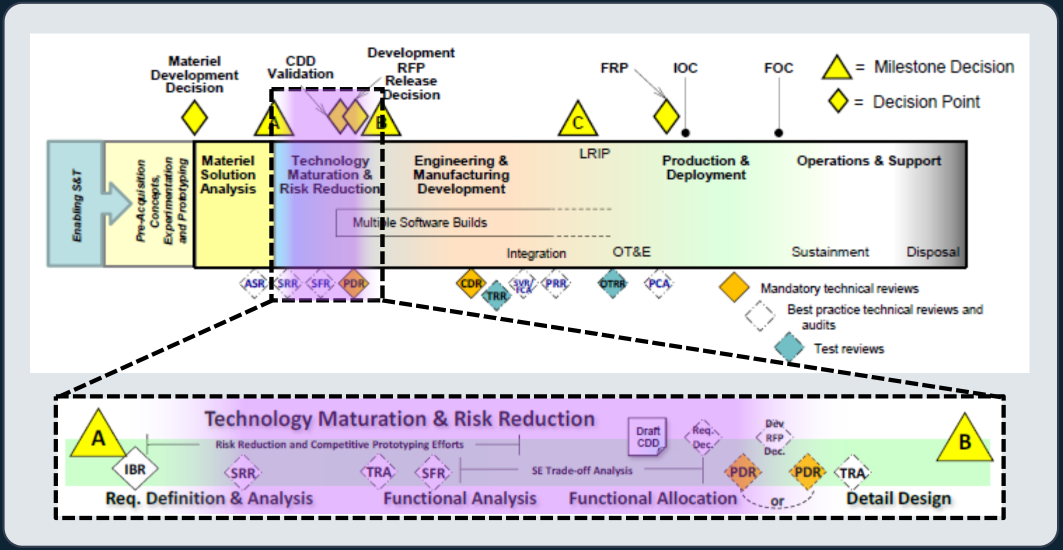
«system» Drone Racing System
ext_structure
«external» Arena Structure
power_supply
«external» Arena Power Supply
media
«external» Media Converage Systems
drone
«system» Drone System
chassis
«system» Drone Chassis
propulsion
«system» Drone Propulsion
c2
«system» Drone Command & Control
cam
«block» FPV Camera
fcs
«block» Flight Control Surfaces
fcc
«block» Flight Control Computer
comms
«system» Drone Communications
power
«system» Drone Power
batt
«block» Battery
pm
«block» Power Management Computer
racer
«system» Racer System
rh
«block» Racer Headset
rc
«block» Racer Controller
rs
«block» Racer Seat
track
«system» Track System
rcs
«system» Race Control System
tc
«block» Track Checkpoint
support
«system» Support Systems
dmr
«block» Drone Maintenance & Repair
rdrp
«block» Race Data Recorder & Publishing

Criteria

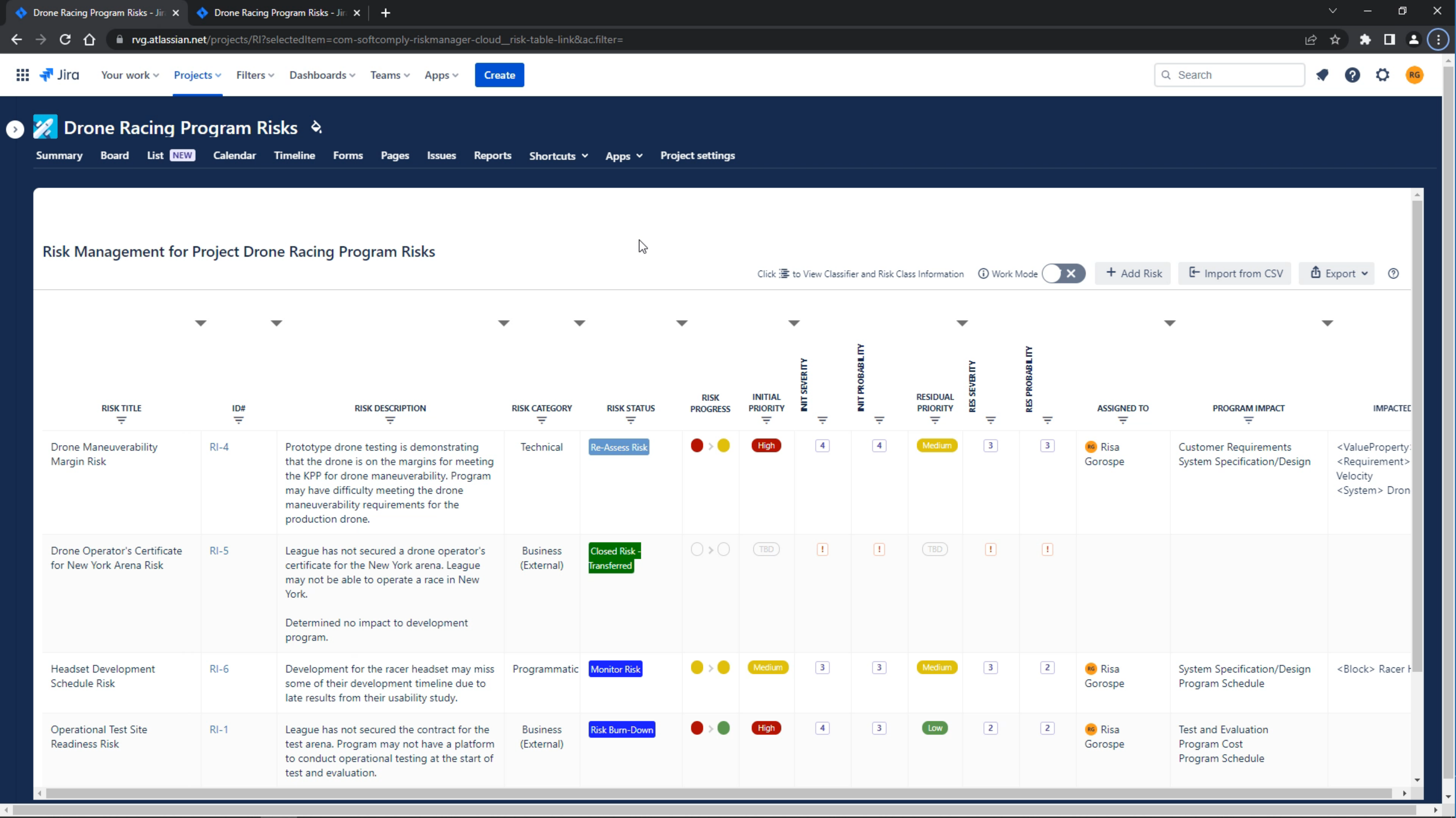
Element Type: Risk Item ... Scope (optional): _Risk Items ... Filter:

Initial Risk Legend: 🟥 High 🟨 Medium 🟩 Low ⬜ Indeterminate **Residual Risk Legend:** 🟥 High 🟨 Medium 🟩 Low ⬜ Indeterminate

#	Risk Id	Risk Title	Risk Description	Risk Category	Risk Status	Initial Risk Priority	Initial Risk Severity	Initial Risk Probability	Residual Risk Priority	Residual Risk Severity	Residual Risk Probability	Program Impa
1	RI-2	Racer Seat Development Schedule Risk	Development for the racer seat may miss some of their development timeline due to some initial staffing issues within the Racer System Development Team.	Programmatic	Monitor Risk	Medium	Moderate Impact	Likely	Low	Minor Impact	Low Likelihood	System Specification/ Program Schedule
2	RI-3	Unmet Media Requirement Risk	Program may not meet media interface requirements due to the late addition the specific streaming codecs to the Customer Requirements.	Technical	Closed Risk - Stopped Monitoring	Low	Minor Impact	Likely	Indeterminate	Unspecified	Unspecified	Customer Requireme System Specification/
3	RI-4	Drone Maneuverability Margin Risk	Prototype drone testing is demonstrating that the drone is on the margins for meeting the KPP for drone maneuverability. Program may have difficulty meeting the drone maneuverability requirements for the production drone.	Technical	Re-Assess Risk	High	Significant Impact	Highly Likely	Medium	Moderate Impact	Likely	Customer Requireme System Specification/
4	RI-6	Headset Development Schedule Risk	Development for the racer headset may miss some of their development timeline due to late results from their usability study.	Programmatic	Monitor Risk	Medium	Moderate Impact	Likely	Medium	Moderate Impact	Low Likelihood	System Specification/ Program Schedule



Materiel Development Decision
CDD Validation
Development RFP Release Decision
FRP
IOC
FOC
= Milestone Decision
= Decision Point
A
B
C
Enabling S&T
Pre-Acquisition Concepts, Experimentation and Prototyping
Materiel Solution Analysis
Technology Maturation & Risk Reduction
Engineering & Manufacturing Development
LRIP
Production & Deployment
Operations & Support
Multiple Software Builds
Integration
OT&E
Sustainment
Disposal
ASR
SRR
SFR
PDR
CDR
TRR
SVR/FCA
PRR
OTRR
PCA
Mandatory technical reviews
Best practice technical reviews and audits
Test reviews
Technology Maturation & Risk Reduction
Risk Reduction and Competitive Prototyping Efforts
Draft CDD
Req. Dec.
Dev RFP Dec.
IBR
SRR
TRA
SFR
SE Trade-off Analysis
PDR
PDR
TRA
or
Req. Definition & Analysis
Functional Analysis
Functional Allocation
Detail Design


Drone Racing Program Risks - Jir ✕
Drone Racing Program Risks - Jir ✕
+
rvg.atlassian.net/projects/RI?selectedItem=com-softcomply-riskmanager-cloud__risk-table-link&ac.filter=
Jira
Your work
Projects
Filters
Dashboards
Teams
Apps
Create
Search
RG
Drone Racing Program Risks
Summary
Board
List NEW
Calendar
Timeline
Forms
Pages
Issues
Reports
Shortcuts
Apps
Project settings

Risk Management for Project Drone Racing Program Risks

Click ☰ to View Classifier and Risk Class Information
ⓘ Work Mode
+ Add Risk
Import from CSV
Export

RISK TITLE	ID#	RISK DESCRIPTION	RISK CATEGORY	RISK STATUS	RISK PROGRESS	INITIAL PRIORITY	INIT SEVERITY	INIT PROBABILITY	RESIDUAL PRIORITY	RES SEVERITY	RES PROBABILITY	ASSIGNED TO	PROGRAM IMPACT	IMPACTED
Drone Maneuverability Margin Risk	RI-4	Prototype drone testing is demonstrating that the drone is on the margins for meeting the KPP for drone maneuverability. Program may have difficulty meeting the drone maneuverability requirements for the production drone.	Technical	Re-Assess Risk	● > ●	High	4	4	Medium	3	3	RG Risa Gorospe	Customer Requirements System Specification/Design	<ValueProperty: <Requirement> Velocity <System> Dron
Drone Operator's Certificate for New York Arena Risk	RI-5	League has not secured a drone operator's certificate for the New York arena. League may not be able to operate a race in New York. Determined no impact to development program.	Business (External)	Closed Risk - Transferred	○ > ○	TBD	!	!	TBD	!	!			
Headset Development Schedule Risk	RI-6	Development for the racer headset may miss some of their development timeline due to late results from their usability study.	Programmatic	Monitor Risk	● > ●	Medium	3	3	Medium	3	2	RG Risa Gorospe	System Specification/Design Program Schedule	<Block> Racer H
Operational Test Site Readiness Risk	RI-1	League has not secured the contract for the test arena. Program may not have a platform to conduct operational testing at the start of test and evaluation.	Business (External)	Risk Burn-Down	● > ●	High	4	3	Low	2	2	RG Risa Gorospe	Test and Evaluation Program Cost Program Schedule	

Lessons Learned

- Organizations can apply system engineering practices to digital engineering if we treat digital engineering like a system
  - Organizations can conduct stakeholder analysis, understand the use cases and write requirements like any other system
  - By understanding the use cases and writing requirements, organizations can be more focused in developing digital engineering capability or assessing potential solutions from vendors
- A complete digital engineering capability supports the needs and execution of the stakeholders in the digital thread
  - More than integrating different software tools together or dumping data into a model/database
  - We demonstrated that we can have a thorough capability by sharing only the necessary data between tools
    - A mature digital engineering system should expose the right information at the right time to its users to do their jobs
- Organizations can learn from digital threads that include non-technical domains (e.g. risk management, program management, etc.)
  - Some of these non-technical domains have functions that span over multiple phases of a program's lifecycle
  - These non-technical domains can drive organizations to consider how the digital engineering environments span the program lifecycle

Questions?

Risa Gorospe (risa.gorospe@jhuapl.edu)
Shannon Dubicki (shannon.dubicki@jhuapl.edu)

JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

33rd Annual **INCOSE**
international symposium

hybrid event

Honolulu HI USA

www.incose.org/symp2023