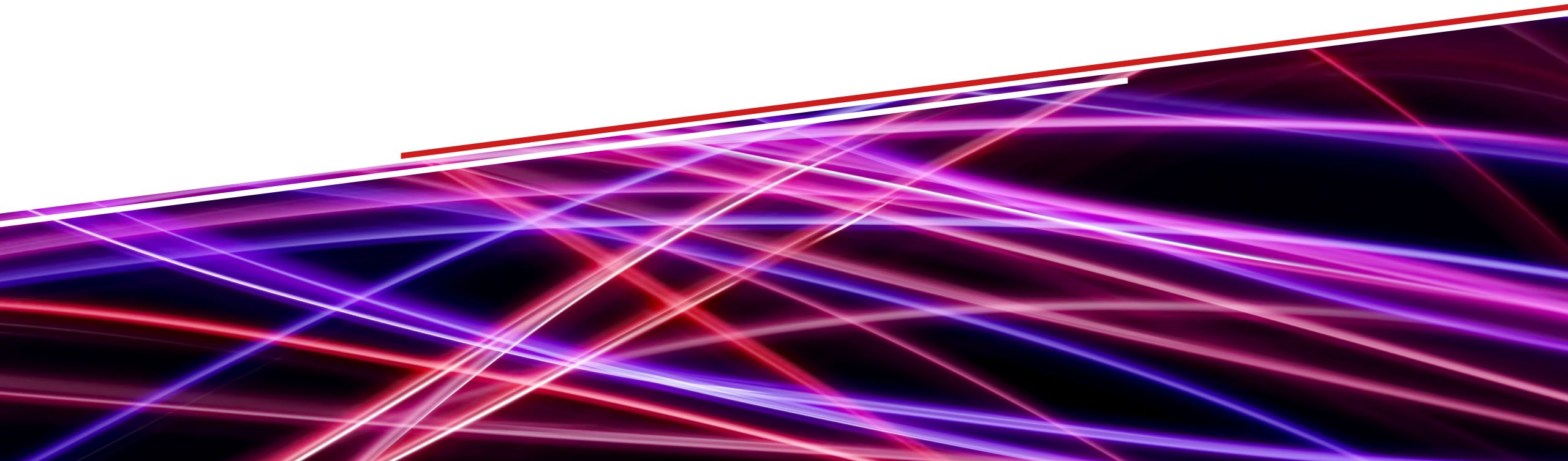


ACE 2024

Building the Digital Engineering Backbone

Jake Donovan and Chad Grubb

March 5th, 2024



ReadyOne

BUILDING THE DIGITAL ENGINEERING BACKBONE

Jake Donovan
Digital Engineer

Chad Grubb
Principal Digital Engineer

Engineering Innovation Factory

SAIC



"THIS DOCUMENT IS THE PROPERTY OF SCIENCE APPLICATIONS INTERNATIONAL CORPORATION. IT MAY BE USED BY RECIPIENT ONLY FOR THE PURPOSE FOR WHICH IT WAS TRANSMITTED AND WILL BE RETURNED UPON REQUEST OR WHEN NO LONGER NEEDED BY RECIPIENT. IT MAY NOT BE COPIED OR COMMUNICATED WITHOUT THE ADVANCE WRITTEN CONSENT OF SAIC. IN ADDITION, THIS DOCUMENT COULD CONTAIN TECHNICAL DATA, THE EXPORT OF WHICH IS RESTRICTED BY THE U.S. INTERNATIONAL TRAFFIC IN ARMS REGULATIONS (ITAR) OR THE U.S. EXPORT ADMINISTRATION REGULATIONS (EAR)."

Table of Contents

1. Intro
2. Data Model
3. Requirements
4. Project Access
5. Release Management
6. Conclusion
7. Questions



Abstract Overview

- ▶ In this presentation we will be giving a high-level overview on how our Aras data model handles the digital engineering lifecycle. Specifically, we will be focusing on requirements as a top-level item and how data moves down the thread from there. We will then be stopping to dive into the use of our Project Access solution, and how it allows for database consolidation and separation. Finally, we will explore release management and the ways it helps us succeed.
- ▶ Successful DE needs good data management and configuration management
- ▶ ReadyOne uses Aras Innovator to maintain a strong DE backbone
 - Configured and customized to fit the digital engineering world in DoD
- ▶ SAIC utilizes Aras Innovator to connect each source of truth and keep a continuous digital thread throughout the digital engineering lifecycle



Data Model

Aras Data Model



Platform Features

Modeling Engine

Platform Services

Succeeds At:

- ▶ Great at PDM
- ▶ Basic Digital Engineering
- ▶ Maintain a single source of truth

Needs Improvement:

- ▶ Full Digital Thread connectivity
- ▶ Maintaining several different sources of truth
- ▶ Supporting a Digital Engineering Ecosystem



SAIC Data Model

SAIC Data Model

Custom Items, properties, connectors, business rules, etc.



Goals:

- ▶ Upgrade existing Aras Innovator capabilities
- ▶ Build new solutions on top of the out of the box data model
- ▶ Support a Digital Engineering Environment

Key Factors:

- ▶ Custom
- ▶ Aligned with DoD Digital Engineering strategy
- ▶ Controllable
- ▶ Agile

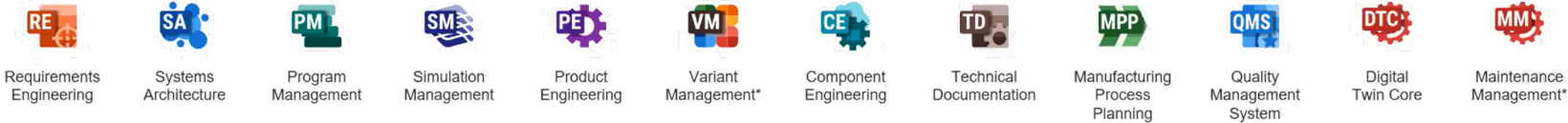


Layers of Data Model



SAIC Data Model

Custom Items, properties, connectors, business rules, etc.



Platform Features

Modeling Engine

Platform Services

ReadyOne Ecosystem

=

SAIC Data Model

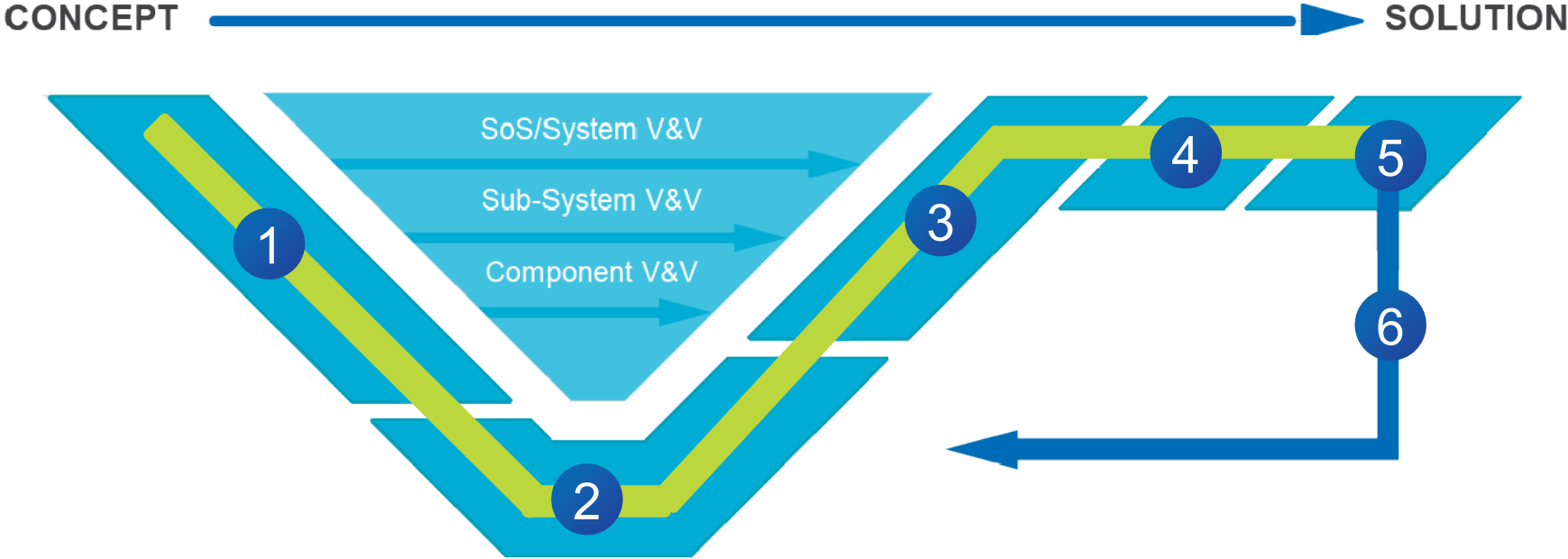
+

Aras Data Model



Requirements

Data Model Application

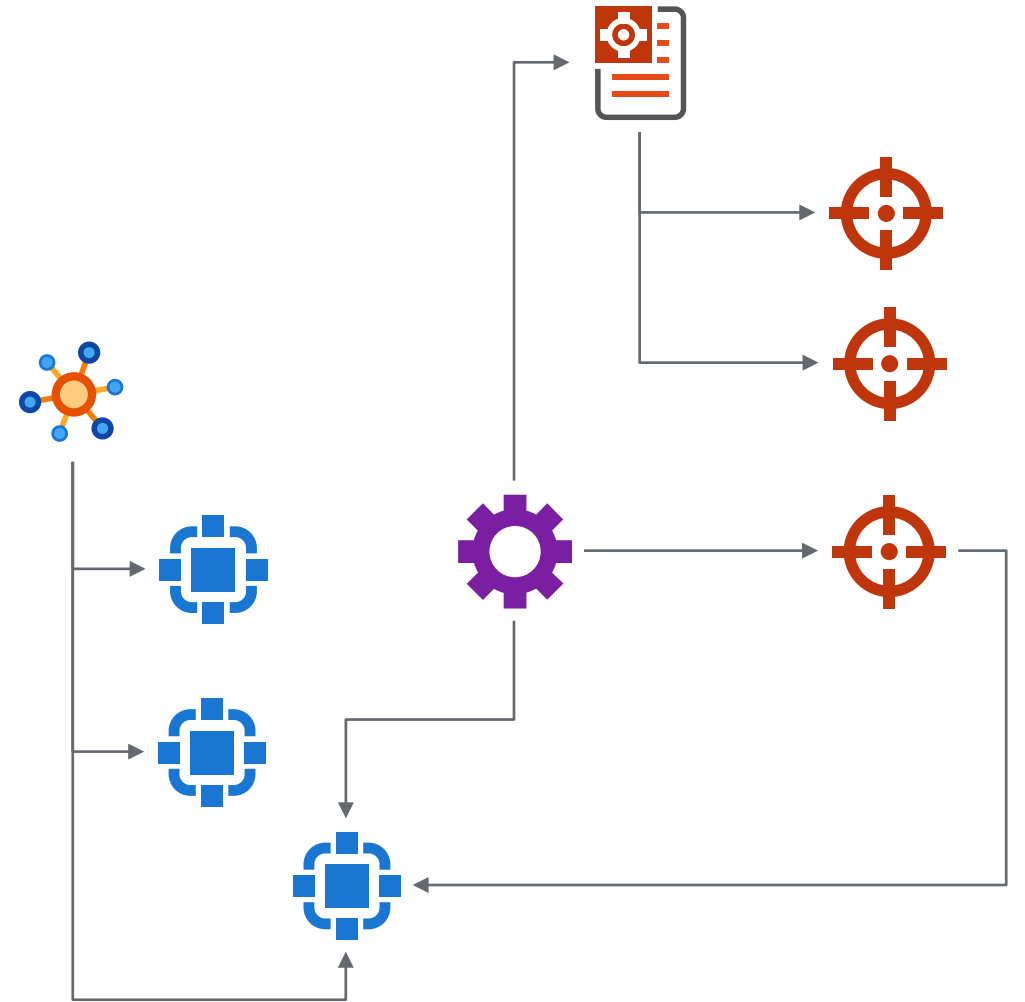


- Digital Thread Platform: Aras Innovator
- Rapidly Deployable Digital Engineering Ecosystem: ReadyOne
- 1 Digital Architectures and Requirements Engineering (DARE)
- 2 Engineering Design and Development
- 3 Modeling, Simulation, Analysis, and Prototyping
- 4 Production and Manufacturing
- 5 Model-Based Product Support
- 6 Digital Twin



Requirements Data Model

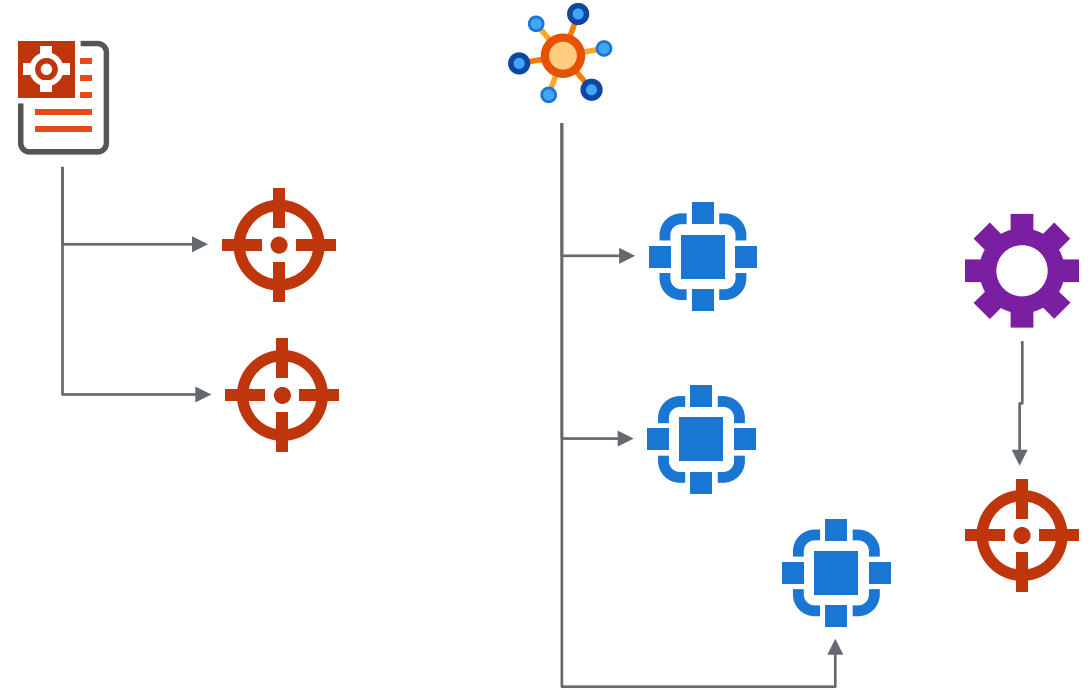
- ▶ Requirements and requirement documents exist as children of parts
- ▶ System elements and models fit awkwardly into the digital thread
- ▶ Multi level relationships exist
 - Parts can be related to both requirement documents and requirements



Requirement Upgrade

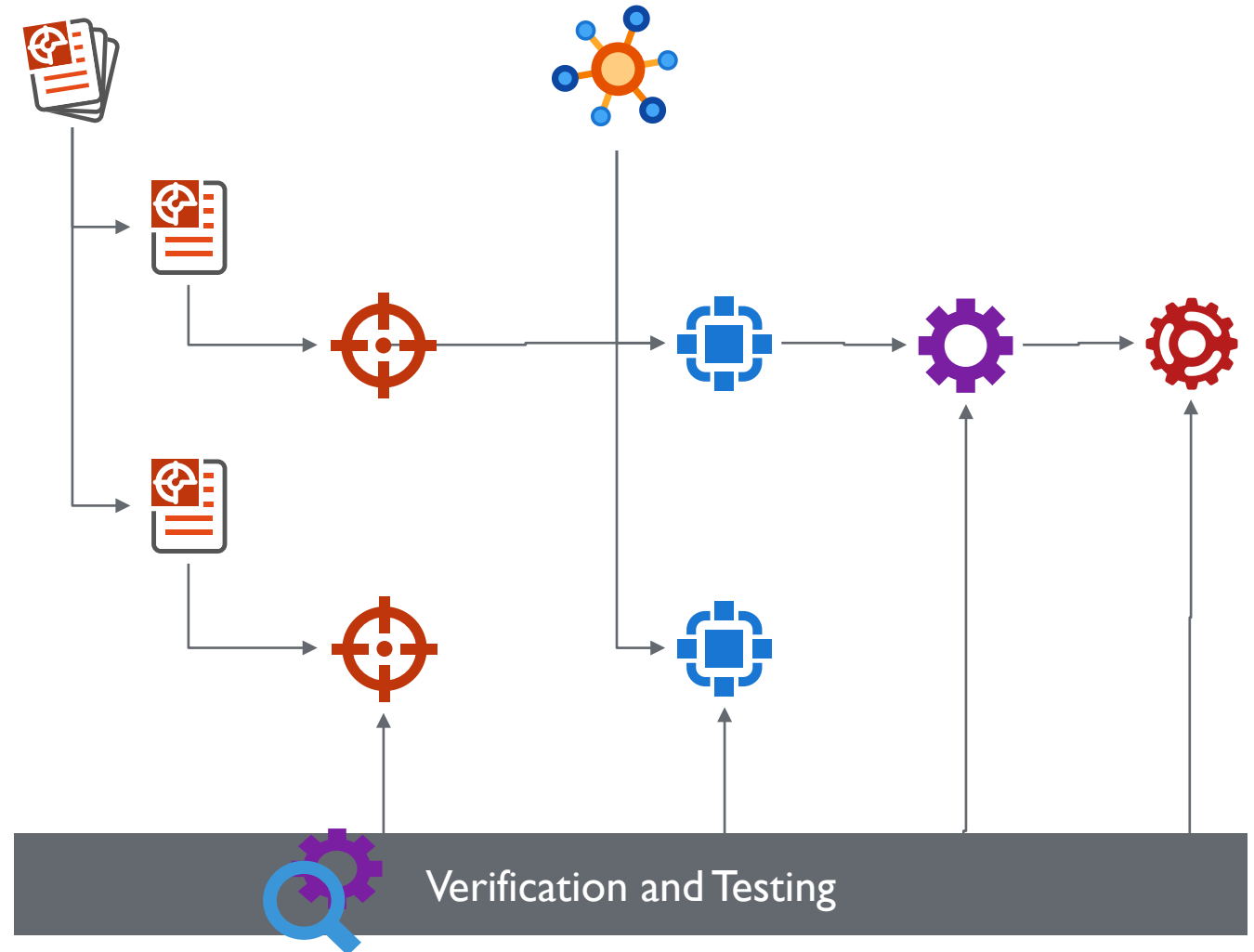
GOALS

- ▶ Defined from our Ontology
- ▶ Requirements based data model
- ▶ Logical structure
- ▶ Simplified forms
- ▶ Custom attributes
- ▶ Well thought out relationships



Requirements Data Model

- ▶ Requirement's structure is updated
 - Project owns modules
 - Modules own requirements
 - Requirements are satisfied by parts or system elements
- ▶ Traceable digital thread
- ▶ Data flows in one direction



Requirement Projects

- ▶ Container for requirement modules
 - Allows you to associate modules together for better understanding.
 - Mirrors DNG to help transition and connect to the software with minimal mapping changes
- ▶ Defines the class types allowed for the requirements within the project
- ▶ Lists the possible attributes that will need to be associated with the requirements

Modules Class Types Attributes Documents

Class Types

Name	Description	Class Role	Icon [...]	Workflow [...]
Test				
System				
Design Constraint				
Functional				
Requirement				

< Prev Next > Page: 1 of 1 | 5 Results | 25

Modules Class Types Attributes Documents

Attributes

Name	Description	Classification
Risk		String
Priority		Integer
Status		String
Start Date		Date

< Prev Next > Page: 1 of 1 | 4 Results |



Requirement Modules

RM-2 JRD Requirement Module ☆



Module

Module Number	RM-2	Rev	A	Status	Preliminary	Managed By	Jake Donovan	Related Source	RP-2 JRD Requirem	Created By:	Jake Donovan
Name	JRD Requirement Module			Owned By	Jake Donovan		Created On:	2/19/2024	Modified By:	Jake Donovan	
Description	<div style="border: 1px solid #ccc; height: 40px;"></div>										
								Modified On:	2/19/2024	Locked By:	
								Release Date:		Effective Date:	
								Generation:	1		

Content | Attributes | Documents

Requirements

Hidden

Chapter	Classification	Number ↑	Text [...]	Cam
	Requirement	RQ-11	This is a demo requirement with no data	
	Test	RQ-12	Another test requirement	
	Requirement	RQ-13	This is a child requirement	

< Prev Next > Page: 1 of 1 | 3 Results

- ▶ Container for requirements specific to that project and module
- ▶ Allows for more descriptive breakdown
- ▶ EX:
 - Project: Car
 - Module: Brakes



Individual Requirements

- ▶ Simplified form with only the crucial data for user understanding
- ▶ Only essential relationships included

RQ-11 ☆

Edit Refresh Share

Requirement Number	Rev	Status	Type	Assigned Owner	Designated User	Related Source
RQ-11	A	Preliminary	Requirement	Jake Donovan		RM-2 JRD Requirement Module

Format: None | Font: | Size: small

This is a demo requirement with no data

Constraints | **Attributes** | Outgoing Links | External Links | Satisfied By | Verified By | Documents | Requirement Diagrams

Attributes ☆

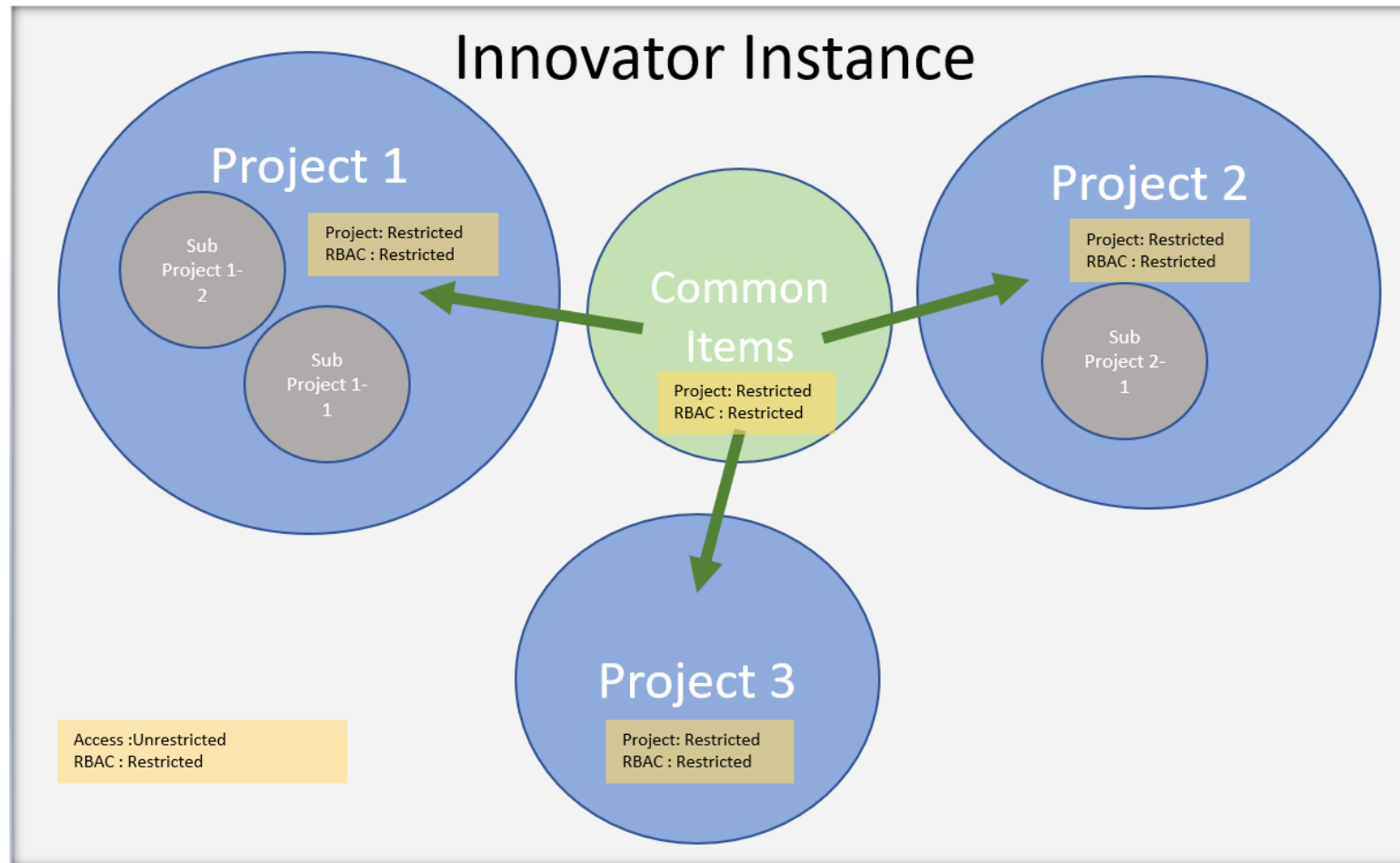
Attribute [...]	Value
Risk	
Priority	3
Status	
Start Date	2/19/2024

< Prev | Next > | Page: 1 of 1 | 4 Results | 25

Project Access

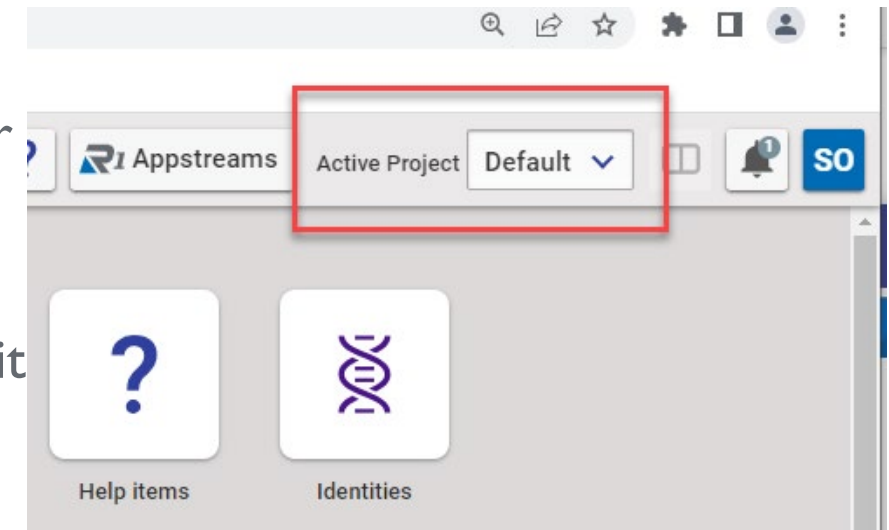
Project Access Purpose

- ▶ Enable the ability to isolate program-specific data from within a common Innovator instance so that users can only access data in which they have been given access to.



Summary of Basic Functionality

- ▶ No controlled items can be created if the User's selected "Active Project" is set to "Default"
- ▶ New, controlled items, will be automatically assigned to the User's selected "Active Project"
- ▶ If an item is controlled under another project that is not in your "Active Project" dropdown, you will be unable to see it or access it.
- ▶ All Users can see items in the "Common Items", but cannot edit them unless they are part of the project
- ▶ All users are assigned to a "Default" project upon creation so that they can receive minimal project access rights.
- ▶ Users with Administrator privileges are exempt from the Project Access rules



Features

- ▶ All users should be able to be assigned to any identified (controlled) item type and access should be controlled to members of said project.
- ▶ Users will have access to Get/Discover any item assigned as a "common item"
- ▶ All Users should be able to be assigned to a project
- ▶ Access rights should be maintained and controlled through MACPolicy implementation
- ▶ Enable the ability to assign specific ItemTypes to project access assignments
- ▶ Instances of a ItemTypes that are being controlled by Project Access must be controlled and identifiable



Release Management

Release Management

- ▶ Release management is a request platform to increase transparency active projects on a database
 - Request is created to add in a project, upgrade, etc.
 - This would allow us to track and create user stories in ADO and link it
 - Keep discussion history with customers and other people within release candidate
- ▶ Allows a release candidate to quickly be approved and deployed after testing is complete
 - Release candidate can be a hotfix for quick release or go through the appropriate offering manager approval steps
 - Creator of the release candidate can stage after offering approvals
 - Quick release given a small group of offering managers and engineers and allows for notification and signoffs at each activity
- ▶ The release candidate can be a plugin, Aras software, a connector, an update, etc.
- ▶ Provide a list of all possible upgrades to database



Form Layout

The screenshot shows a software interface for a 'Release Candidate'. The form includes the following fields and sections:

- Number:** A text input field containing 'Server Assigned'.
- Name:** An empty text input field.
- URL Link:** A text input field containing 'link'.
- Description:** A large text area for detailed information.
- Created_on:** A date selection field.
- Release Date:** A date selection field.

Below the form is a 'Files' section with a 'SignOffs' tab. It features a toolbar with icons for adding, deleting, searching, and hiding files, along with a table with columns: Sequence, File [...], File Type [...], Comments, and Indexed On [...].

Name

Name of release candidate

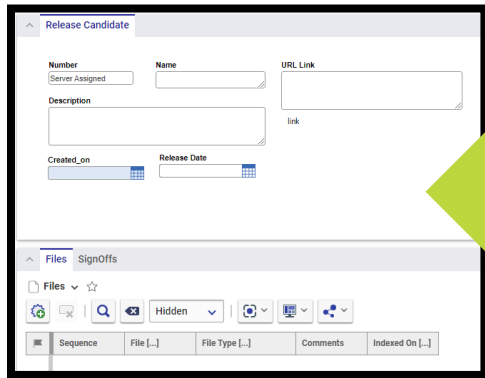
URL Link

Active link to version control

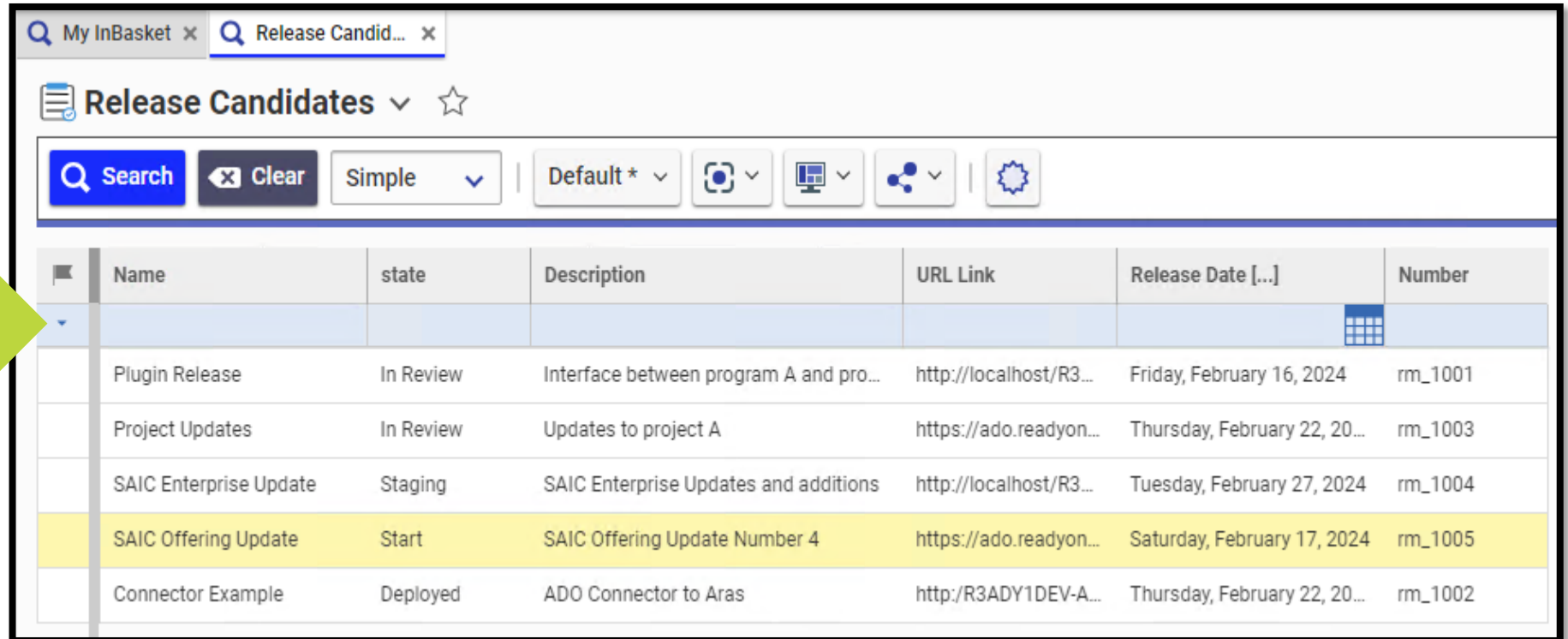
Description

Necessary information to describe what is in the package and what offerings are affected

List of Release Candidates



A screenshot of a 'Release Candidate' form. It includes fields for 'Number' (Server Assigned), 'Name', 'URL Link', 'Description', 'Created_on', and 'Release Date'. Below the form are tabs for 'Files' and 'SignOffs', and a table with columns: Sequence, File [...], File Type [...], Comments, and Indexed On [...].



A screenshot of a web application showing a list of 'Release Candidates'. The interface includes a search bar, a 'Clear' button, and several filter icons. The table below lists various release candidates with columns for Name, state, Description, URL Link, Release Date, and Number. The row for 'SAIC Offering Update' is highlighted in yellow.

Name	state	Description	URL Link	Release Date [...]	Number
Plugin Release	In Review	Interface between program A and pro...	http://localhost/R3...	Friday, February 16, 2024	rm_1001
Project Updates	In Review	Updates to project A	https://ado.readyon...	Thursday, February 22, 20...	rm_1003
SAIC Enterprise Update	Staging	SAIC Enterprise Updates and additions	http://localhost/R3...	Tuesday, February 27, 2024	rm_1004
SAIC Offering Update	Start	SAIC Offering Update Number 4	https://ado.readyon...	Saturday, February 17, 2024	rm_1005
Connector Example	Deployed	ADO Connector to Aras	http://R3ADY1DEV-A...	Thursday, February 22, 20...	rm_1002



Release Management Advantages

ADVANTAGES

A Release Candidate package can be deployed within the same day

A list of all release candidates will be recorded within Aras and can be viewed anytime

A small group of people is needed for approvals and deployment

FUTURE TASKS

Automated final build and deploy

Different levels of releases



DATABASE CONSOLIDATION AND SEPARATION

Allows for project to coexist in one database

Minimizes data leaks

STANDALONE SOLUTIONS

Utilizes out of the box Aras Data Model and builds on top of the existing solutions

Modular design for easy deployment

DOD FOCUSED DATA MODEL

OPEN model Based SOA with pre-configured integrations for modeling tools from all domains

Always available data, No proprietary formats

ReadyOne™

Award winning, easy to deploy, cost effective, and powerful integrated environment that delivers Digital Transformation

INTERNAL DATABASE RELEASE MANAGEMENT

Transparency of deployment within the database

Streamline deployment process

REQUIREMENT BASED

Realizes the true Digital Engineering lifecycle

Generates a traceable Digital Thread

LIFECYCLE INTEROPERABILITY

Brings in the Engineering Vee diagram to the digital engineering environment

Digital Thread from conops to sustainment



Thank you!

ACE 2024

