

Introducing Intuitive MBSE for the Ground Segment Starting with the Euclid Mission

Marcus Wallum, Mehran Sarkarati

Ground System Architectures Workshop 2019
26/02/2019

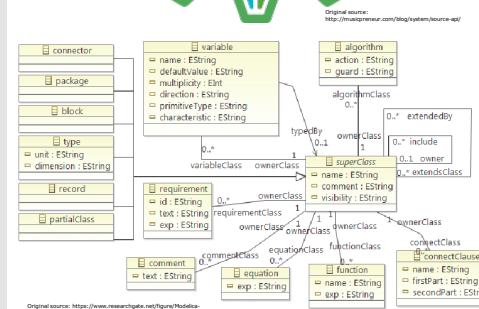
© 2019 by the European Space Agency
Published by The Aerospace Corporation with permission

ESA UNCLASSIFIED - Releasable to the Public



Why?

- Complexity
- Flow of Information through mission phases
- Scattered Knowledge
- Domain-specific tools
- Paper Centric



What?

- Manage Complexity
- Collaborative
- Formal: engineering data with underlying data model and precise semantics
- Informal: model annotations, discussions
- **Model Centric**



Difficulty?

- **Change** to BAU
- **Risk**: perceived as unnecessary risk
- “**Big Bang**” **approach** has failed in the past
- **Formal** Notations: SysML, BPMN, UPDM, etc
- **Over-modelling** Tendency
- **Lack of guided methodology**
- Must maintain formality and strong semantics
- **Difficult**: Extreme learning-curve
- **Funding**: New Tools, New Experts, ...

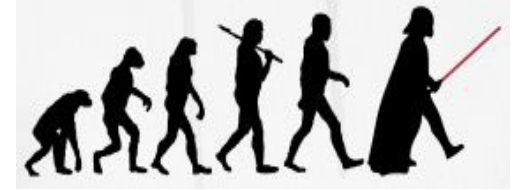


Original source: <https://www.brightidea.com/blog/overcoming-innovation-program-anxiety-program-sponsors-creed-risk-responsibilities-2/>



Solution?

- **Avoid Big Bang**
- Target low-hanging fruits
- **User-driven** development
- Pair modelling experts with end users
- Real use case: Euclid Mission
- Easy-to-use tool tailored to your needs



Our Tool: Paperless Ground Segment Engineering



- **ECSS standards:** systems, operations & software engineering, quality assurance
- **Configuration control:** support to multiple design baselines and review milestones
- **Document generation:** support transition phase, generated directly from the model
- **Integrated:** Interfaces with existing Software Development Environment tools



Data Model (under iteration)



- engineeringmodel
 - base
 - Dataltem
 - EngineeringDataItem -> DataItem
 - description
 - name : EString
 - description : EString
 - aliases : Alias
 - satisfies : Satisfy
 - Alias
 - EngineeringModelType
 - Component = 0
 - Requirement = 1
 - Test = 2
 - Process = 3
 - Interface = 4
 - Diagram = 5
- requirements
 - TraceReq -> ReferenceItem
 - CopyReq -> TraceReq
 - DeriveReq -> TraceReq
 - Verify -> ReferenceItem
 - Satisfy -> ReferenceItem
 - Requirement -> DataItem
 - GenModel
 - type : ReqKind
 - reqIdentifier : EString
 - title : EString
 - text : EString
 - version : EString
 - force : ReqForceKind
 - verification : ReqVerificationMethodKind
 - comment : EString
 - justification : EString
 - children : Requirement
 - parent : Requirement
 - satisfiedBy : Satisfy
 - verifiedBy : Verify
 - ReqSatisfyKind
 - ReqVerificationMethodKind
 - ReqForceKind
 - ReqKind
- Test -> EngineeringDataItem
 - GenModel
 - steps : TestStep
 - preTests : Test
 - verifiedRequirements : Verify
 - testedComponents : ComponentType
 - prevalidatedInterfaces : Interface
 - start : EDate
 - endDate : EDate
 - responsible : Actor
 - conductor : Actor
 - involvedSites : EString
 - functionalPrerequisites : EString
 - testPrerequisites : EString
 - requiredTestData : EString
 - requiredAnalysisTools : EString
 - outputs : EString
 - generalResult : EString
 - testReadinessReview : EString
 - trrDate : EDate
 - trrConfig : EString
 - trrDocuments : EString
- qudv
 - QUDDVData
 - name : EString
 - compositePropertyElement : CompositePropertyElement
 - valuePropertyCube : ValuePropertyCube
 - valueProperty : ValueProperty
 - scale : Scale
 - dimension : ScaleValueDefinition
 - quantityKindFactor : QuantityKindFactor
 - unitFactor : UnitFactor
 - prefix : Prefix
 - dimension : Dimension
 - factor : DimensionFactor
 - quantityKind : QuantityKind
 - unit : Unit
 - systemOfQuantities : SystemOfQuantities
 - systemOfUnits : SystemOfUnits
 - Integer [int]
 - Float [double]
 - Long [long]
 - Double [double]
 - String [java.lang.String]
 - Boolean [boolean]
 - blocks
 - conceptualmodel
 - valueproperties

Core Engineering Data Items:
- Components
- Requirements
- Interfaces
- Processes
- Tests



Intuitive graphical data exploration and editing

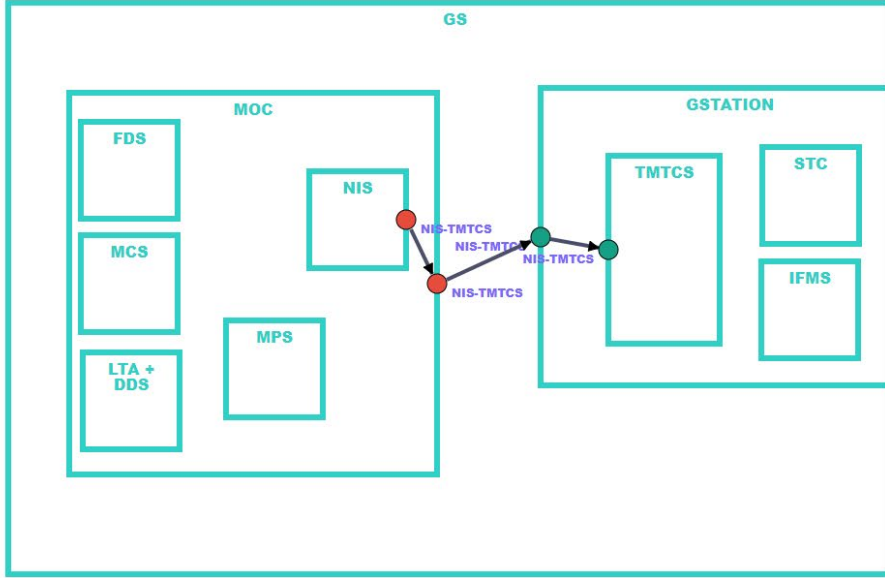
Projects:

- File
- test2 [master]
 - Requirements
 - Components
 - GS
 - GStation
 - nis-tmtcs
 - IFMS
 - STC
 - TMTCS
 - MOC
 - nis-tmtcs
 - FDS
 - MCS
 - LTA + DDS
 - MPS
 - NIS
 - Interfaces
 - NIS-TMTCS
 - MCS-FDS
 - MPS-FDS

Details Diagram Discussions Branches Changes Validation

Diagram: [Icons] 100% Element: [Remove] [References]

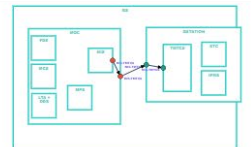
ACTION:



Toolbox

- Select Item
- Requirements
 - Requirement
 - Satisfy
- Components
 - Component
 - Port
 - Interface
- Processes
- Tests

Outline



Integrated Requirements Management, Import, Export



Details

ID:

GS-01

Name:

MOC

Children

Requirements

Interfaces

Tests

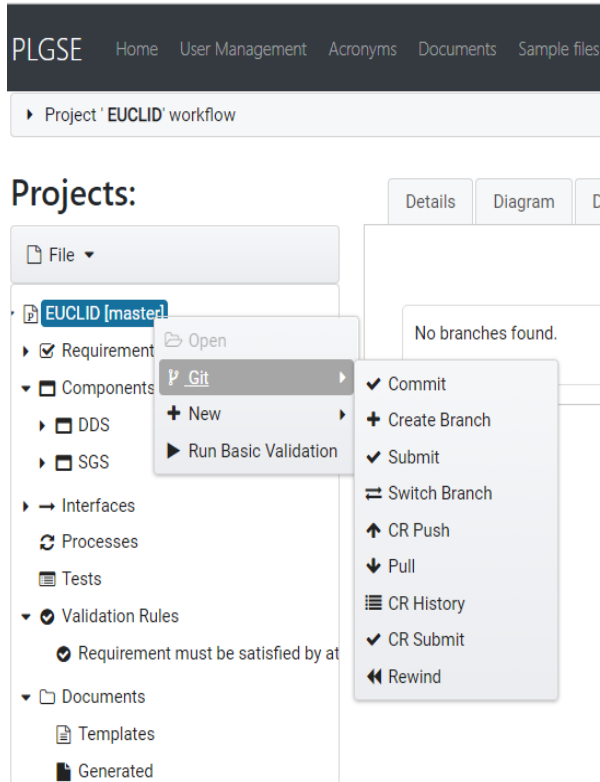
Select requirement(s)

Name	Identifier	Text	
MIRD-F-100	MIRD-F-100	The Euclid Mission Operation Centre (MOC) shall be responsible for mission opera... more	
MIRD-F-110	MIRD-F-110	The NEW1 MOC shall perform the scheduling, monitoring and control of the spacecr... more	
MIRD-F-120	MIRD-F-120	The MOC shall produce the data products for platform and instruments health che... more	
MIRD-F-130	MIRD-F-130	MOC shall prepare and validate of the Flight Operations Plan and operations proc... more	
MIRD-F-140	MIRD-F-140	MOC shall prepare and validate the ground segment operations procedures necessar... more	
MIRD-F-150	MIRD-F-150	MOC shall perform the mission analysis, which output shall be reflected in the C... more	
MIRD-F-160	MIRD-F-160	MOC shall be responsible for the definition of all the necessary interfaces to M... more	

Description

B *I* U x_2 x^2 Normal

Mission Operations Center



- Git enables support for multiple baselines and dedicated branches
- Access to commit history and branch information
- Enables Concurrent Engineering (push/pull changes)

CDR

Created 3 minutes ago by Admin. Updated 3 minutes ago.

Description

This is a branch, dedicated to the Critical Design Review (CDR)

Changes

Change for baseline changes
00 · Created a minute ago by admin

Under review

Branch: Change for baseline changes

Discussions

Authors: All ▾

Assignees: All ▾

Search or filter

No discussions resol...

Last updated ▾

Discrepancies to agreed interface updates
Opened a few seconds ago by admin

master-review-2

updated a few seconds ago

New discussion



Title of new discussion

B **I** U    x_2 x^2 

Enter a new discussion...

Global status: **Under review**

Review Panel

No review panel


Add review panel

Notifications

No subscriptions

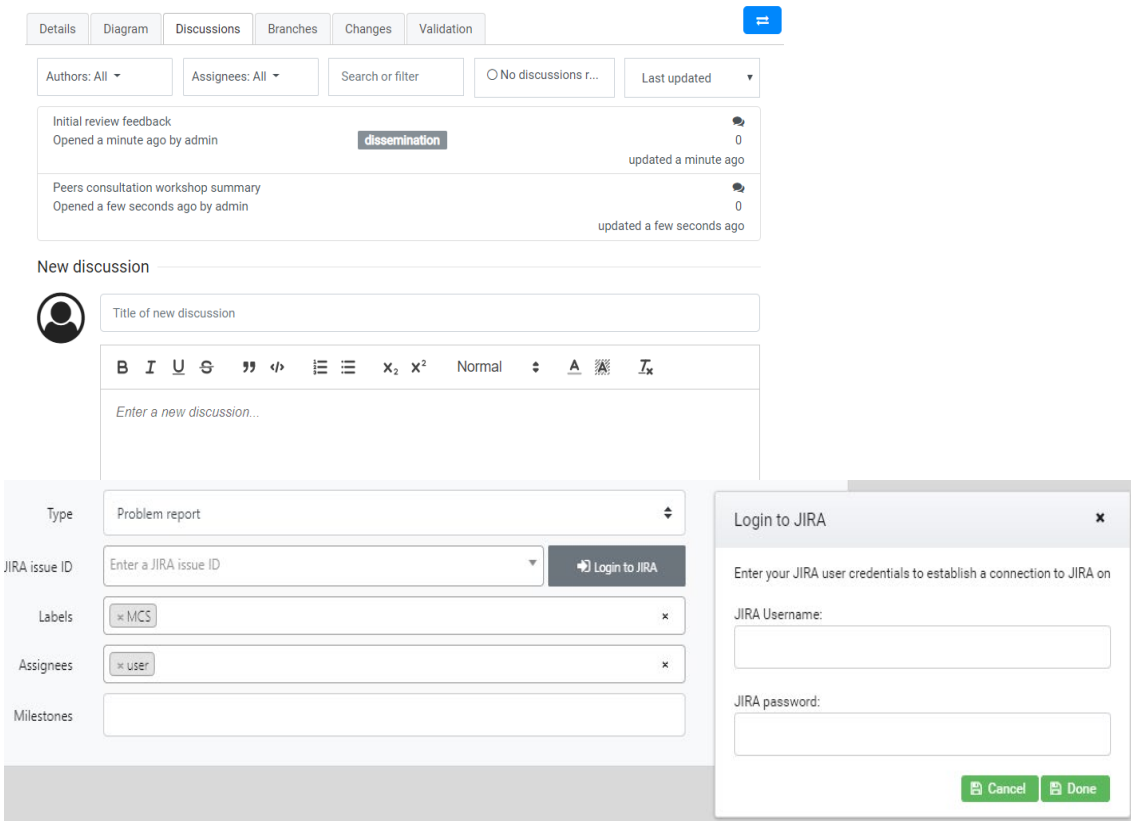
Subscribe

Labels

CDR 

Add label

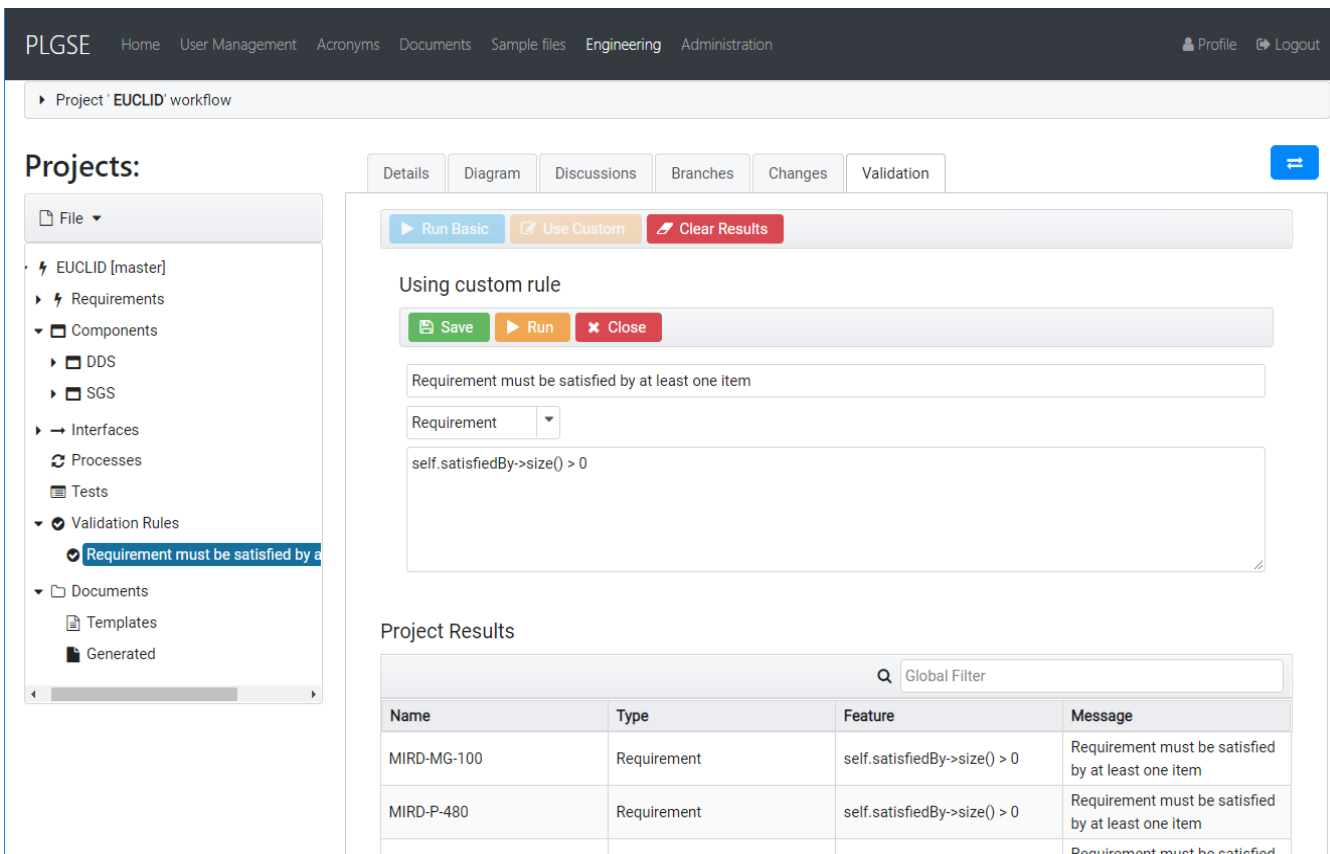
- Collaborative approval workflows: changes and milestone reviews, delegation, subscription, push notifications
- Changes to the model can be submitted and reviewed in dedicated Review Mode, prior to undergoing approval and merge to master branch



The screenshot displays the 'Collaboration' interface with several components:

- Navigation:** Tabs for 'Details', 'Diagram', 'Discussions', 'Branches', 'Changes', and 'Validation'. A blue menu icon is on the right.
- Filters:** 'Authors: All', 'Assignees: All', 'Search or filter', 'No discussions r...', and 'Last updated'.
- Discussion List:**
 - Initial review feedback: Opened a minute ago by admin. Tag: dissemination. 0 comments. updated a minute ago.
 - Peers consultation workshop summary: Opened a few seconds ago by admin. 0 comments. updated a few seconds ago.
- New discussion form:** Includes a profile icon, a title input field, and a rich text editor with options for bold, italic, underline, link, quote, list, link, x₂, x², normal, text color, background color, and link.
- Discussion Creation Form:** Fields for 'Type' (set to 'Problem report'), 'JIRA issue ID', 'Labels' (MCS), 'Assignees' (user), and 'Milestones'. A 'Login to JIRA' button is present.
- Login to JIRA Modal:** A dialog box with the title 'Login to JIRA' and a close button. It contains the text 'Enter your JIRA user credentials to establish a connection to JIRA on' and input fields for 'JIRA Username:' and 'JIRA password:'. 'Cancel' and 'Done' buttons are at the bottom.

- Support for ad-hoc discussions, with tags linked to engineering data
- Discussions with types, including Problem Reports linked to JIRA
- Delegations, subscriptions, notifications support



The screenshot shows the PLGSE web interface. At the top, there is a navigation bar with 'PLGSE' and various menu items like 'Home', 'User Management', 'Acronyms', 'Documents', 'Sample files', 'Engineering', and 'Administration'. Below this, a breadcrumb trail shows 'Project 'EUCLID' workflow'. The main content area is titled 'Projects:' and features a sidebar with a tree view of the project structure, including 'EUCLID [master]', 'Requirements', 'Components', 'Interfaces', 'Processes', 'Tests', 'Validation Rules', and 'Documents'. The 'Validation Rules' section is expanded, showing a rule named 'Requirement must be satisfied by a...'. The main panel displays the configuration for this rule, with buttons for 'Run Basic', 'Use Custom', and 'Clear Results'. The 'Using custom rule' section shows a 'Save' button, a 'Run' button, and a 'Close' button. Below this, there is a text input field containing the requirement 'Requirement must be satisfied by at least one item' and a dropdown menu set to 'Requirement'. A text area contains the OCL constraint: `self.satisfiedBy->size() > 0`. The 'Project Results' section shows a table with a search filter and the following data:

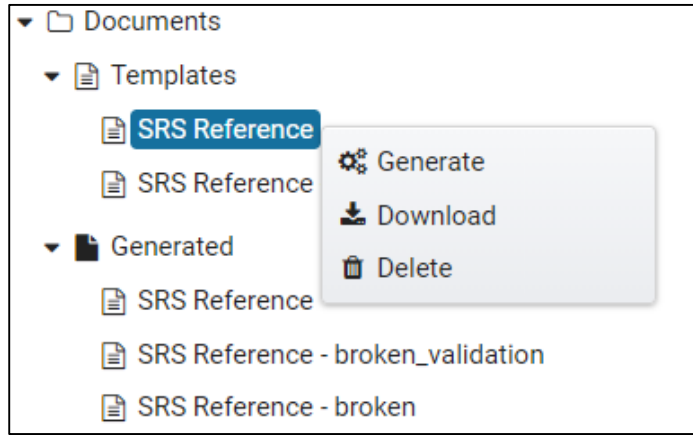
Name	Type	Feature	Message
MIRD-MG-100	Requirement	self.satisfiedBy->size() > 0	Requirement must be satisfied by at least one item
MIRD-P-480	Requirement	self.satisfiedBy->size() > 0	Requirement must be satisfied by at least one item

- Basic formal validation: Ecore meta-model

- Custom validation: user-defined OCL constraints

-> Ensures rigour and consistency

-> Answers questions



- Based on '.docx' templates, M2Doc
- Ensures consistent deliverables
- Templates and documents saved in the database

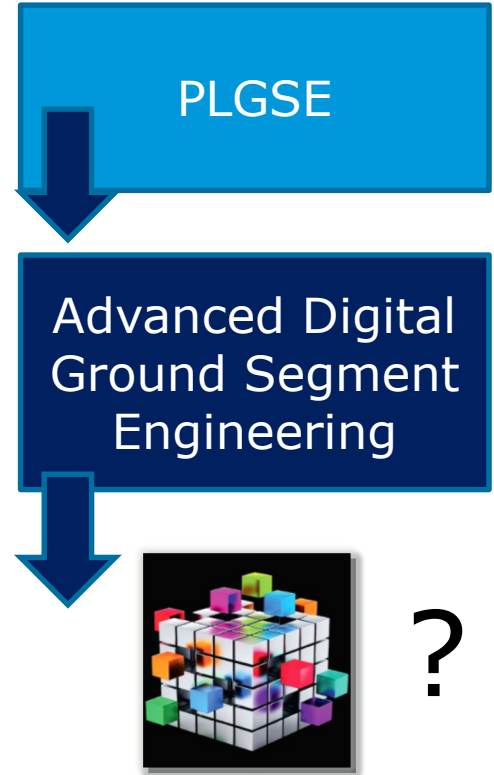
ID	Type	Title	Text	Version
<code>{ m:for requirement allRequirements }</code>				
<code>{ m:requirement.reqIdentifier }</code>	<code>{ m:requirement.type }</code>	<code>{ m:requirement.title }</code>	<code>{ m:requirement.text }</code>	<code>{ m:requirement.version }</code>
<code>{ m:endfor }</code>				



MIRD-F-100	GENERIC	MIRD-F-100	The Euclid Mission Operation Centre (MOC) shall be responsible for mission operations and shall be the only interface with the spacecraft through the ground stations network.	1.0
------------	---------	------------	--	-----

PLGSE: Next steps and future

- Add parametrics (QUDV) to front-end
- Validate: **shadow-engineering of Euclid Mission**
- More tools integration
- Reference libraries and architectures
- Expand existing tool interfaces
- Space segment inter-model mappings
- Executable models and links to analysis tools
- Integration with wider ESA/ESOC context:
- Model-based X engineering?
 - Security, Reliability, Cost, Operations



Thank you for your attention!

Q&A and Discussion