33rd Annual INCOSE international symposium

hybrid event

Honolulu HI USA

A Systematic and Traceable MOSA Evaluation Process for Systems Architectures: A Digital Engineering Tool

15-20 July - 2023

www.incose.org/symp2023 #INCOSEIS

Georgia Tech Research Institute

Dr. Awele Anyanhun, CSEP
Senior Research Engineer
MBSE Research & Application Branch
Awele.Anyanhun@gtri.gatech.edu

Ms. Clarissa Fleming, CSEP
Senior Research Engineer
Applied Open Architecture Branch (formerly)
CFlemRuss@gmail.com

Mr. Whit Matteson
Branch Head, AOAB
Applied Open Architecture Branch
Whit.Matteson@gtri.gatech.edu

This work was sponsored by
Program Executive Office - Ground Combat Systems, US Army
under contract # W31P4Q18D0002-W31P4Q20F0394

Digital engineering is a key enabler for highly effective development of MOSA-based architectures.

- This MBSE process provides confidence that MOSA-based architecture development will result in cost and schedule savings *as required.*

…Provides confidence for realization of more highly effective implementation

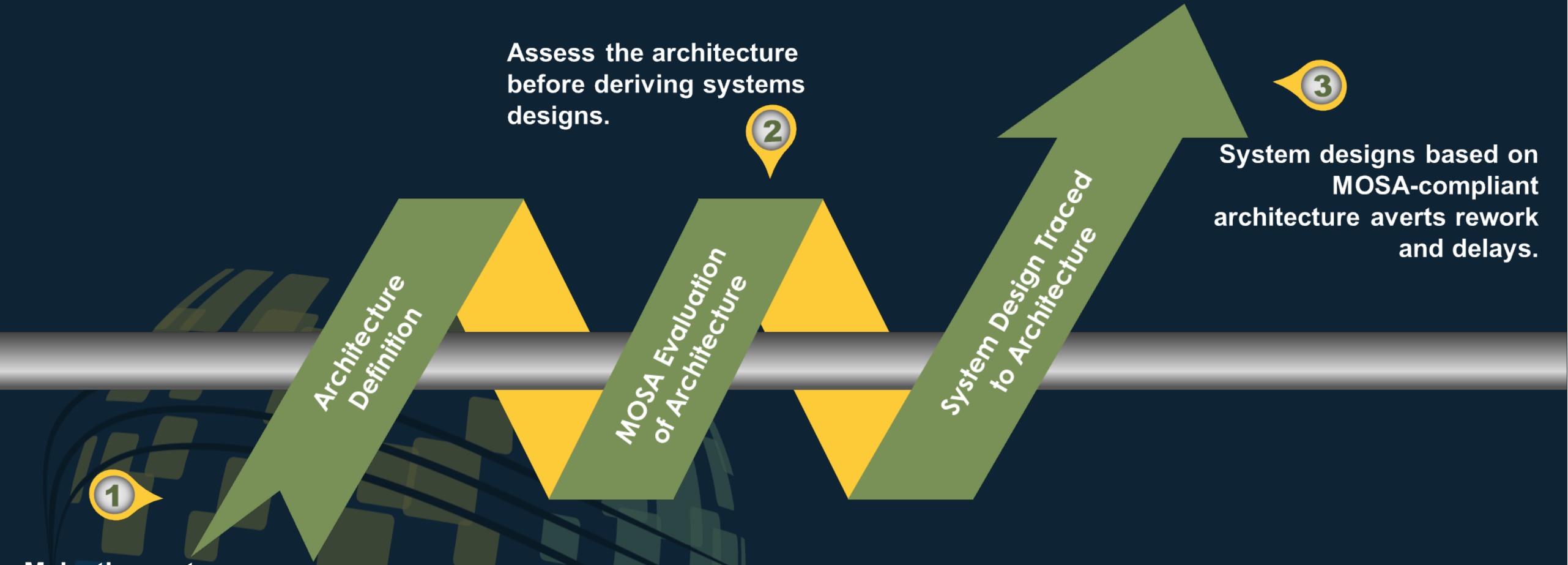
…Increases probability of efficient systems engineering practice

…Establishes traceable relationships between SE artifacts in one location

…Easily sustains integrated project data integrity during changes

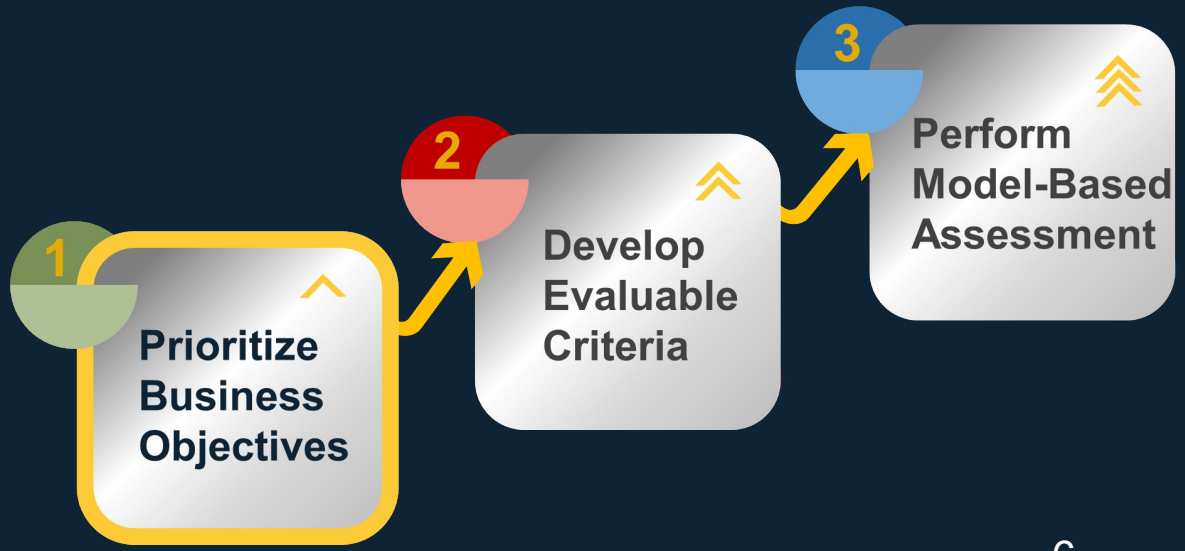
A digital engineering approach…

Validation of MOSA-related architecture decisions prior to the design phase reduces the risk of program failure.



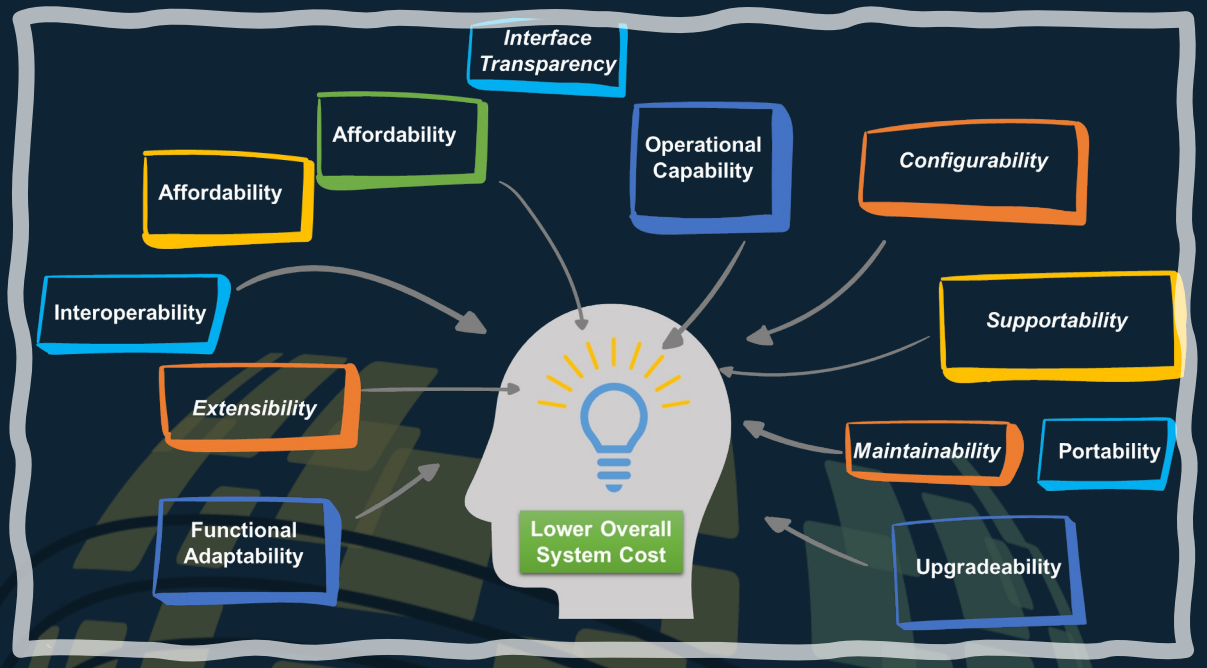
Our approach uses MOSA QA criteria to evaluate architecture requirements prior to system design.

Prioritize Specified Business Objectives

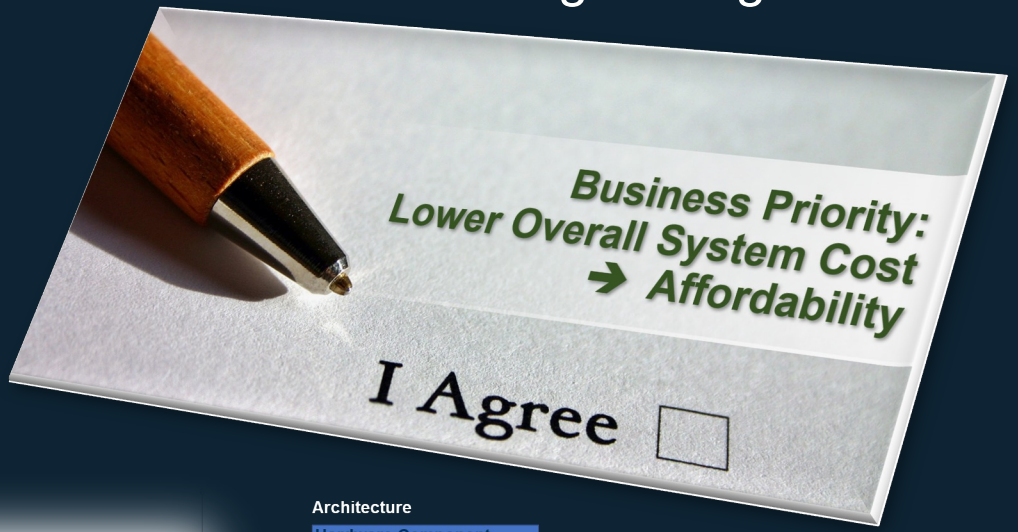


Define applicable MOSA Quality Attributes (QAs).

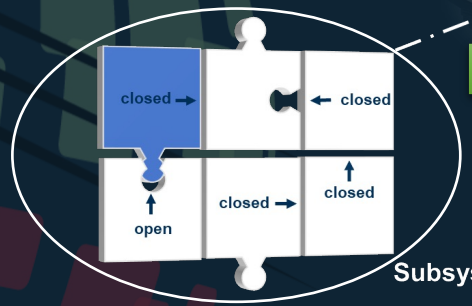
Identify and understand the business and technical priorities for MOSA integration.



Ensure mutual stakeholder understanding and agreement.



Understand how MOSA is being integrated into the architectural design.

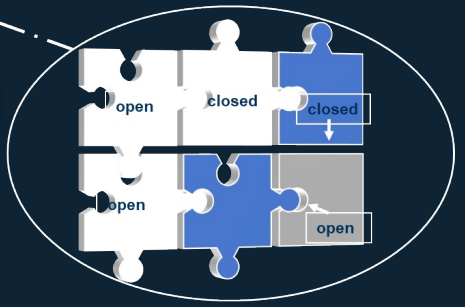


Subsystem A

Lower Overall System Cost

Subsystem B

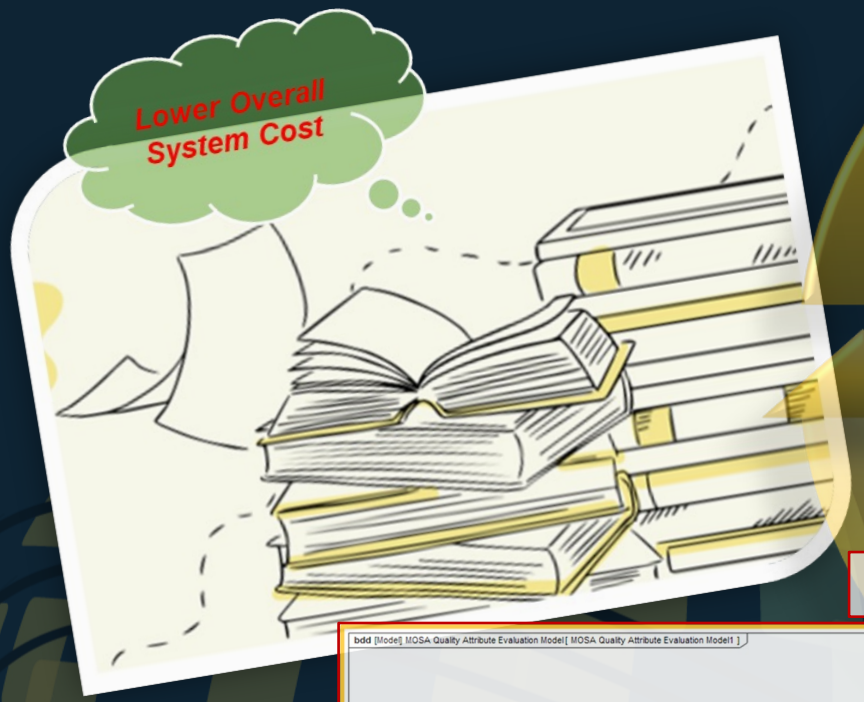
Architecture
Hardware Component
Software Component
Human

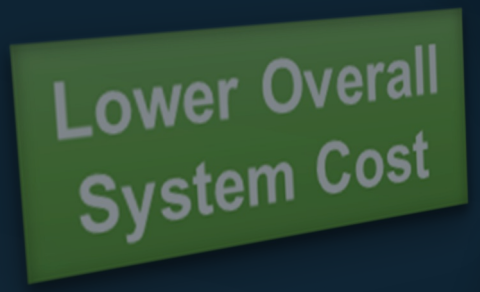


1 Prioritize Specified Business Objectives

Ensure sufficient coverage of business drivers.

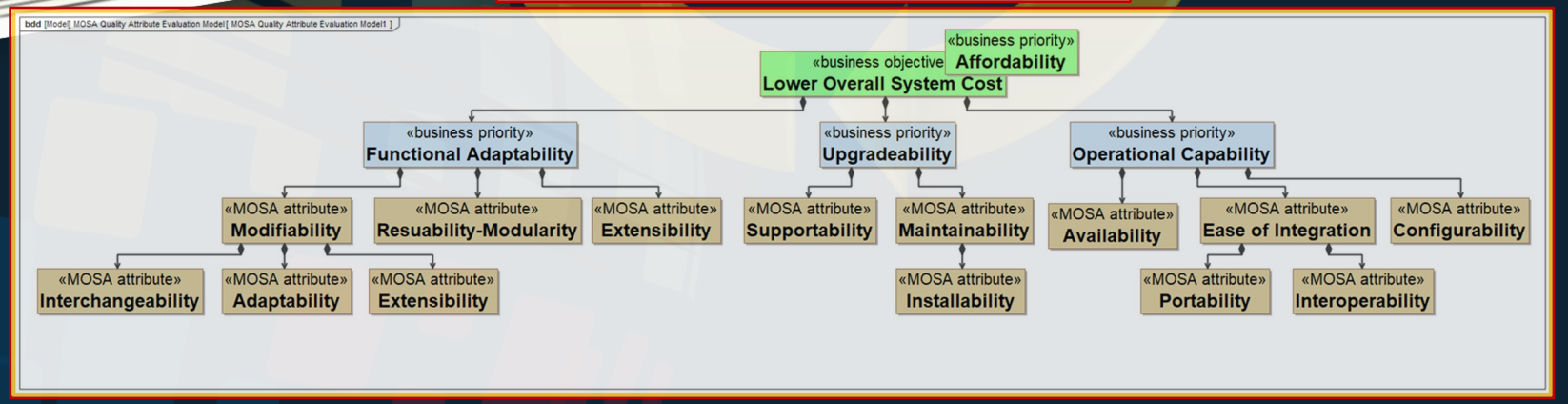
Research, analyze, down-select an applicable body of MOSA QAs.





Is the “Business Objectives” view sufficiently, clearly indicated?

MOSA Business Objectives View

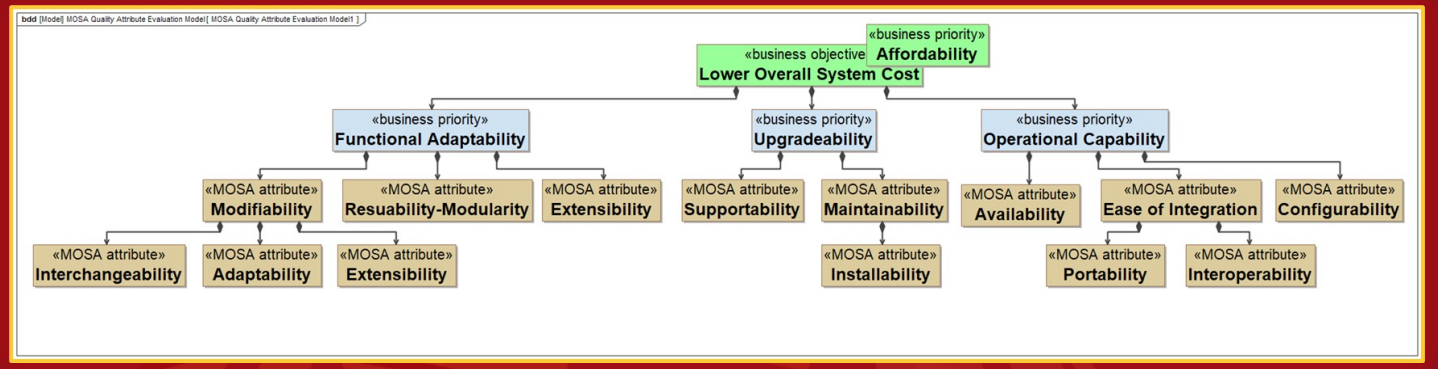


1 Prioritize Specified Business Objectives

Deconflict MOSA QA redundancy.

Are the program priorities for MOSA completely and distinctly specified by the selected MOSA QAs?

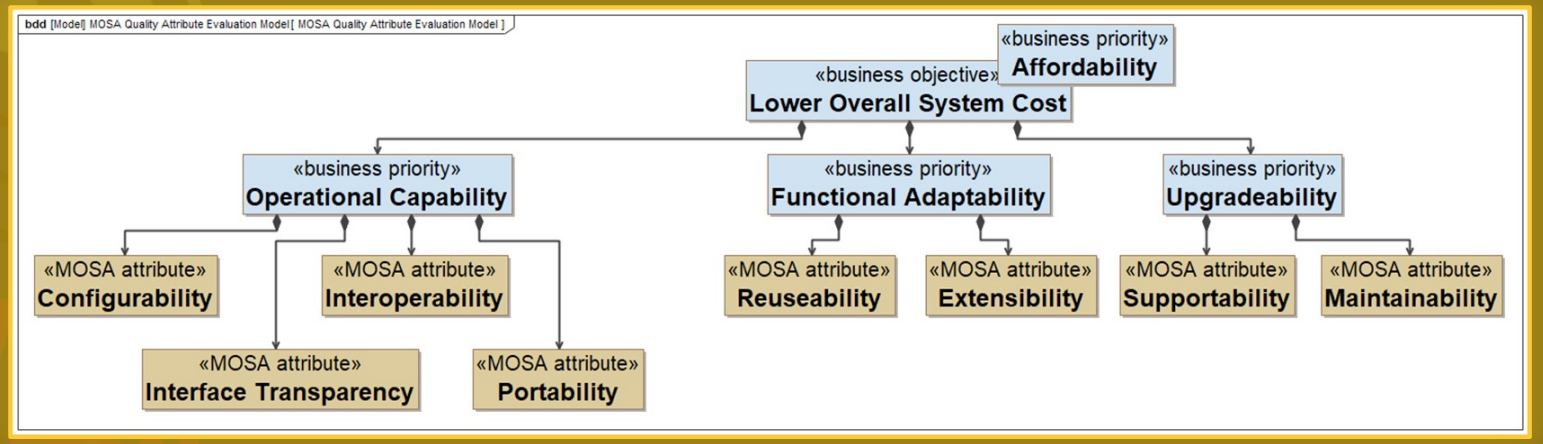
MOSA Business Objectives View



Is the “Business Objectives” view clearly and distinctively specified?

Is the “Evaluation View” (technical) clearly specified?

MOSA Compliance Evaluation View



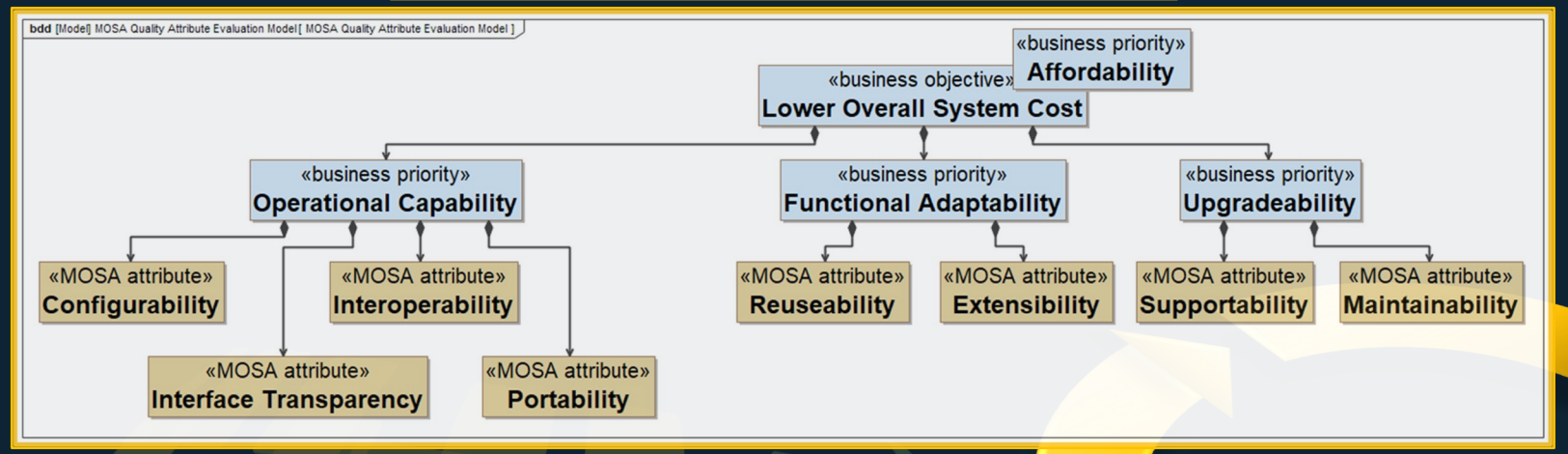
Focus on coverage of concepts, not semantics.

1 Prioritize Specified Business Objectives

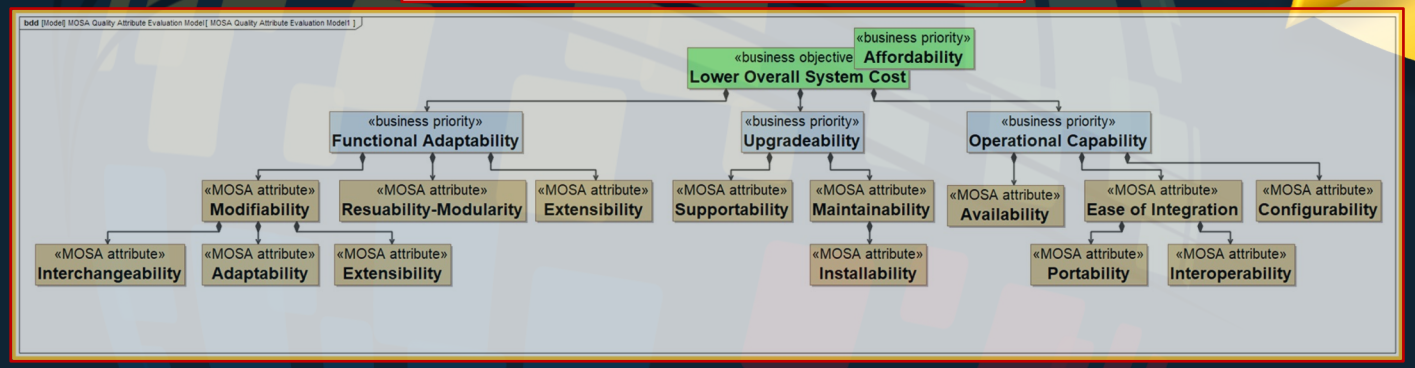
In summary, first step is to firmly establish the basis and focus for MOSA application.

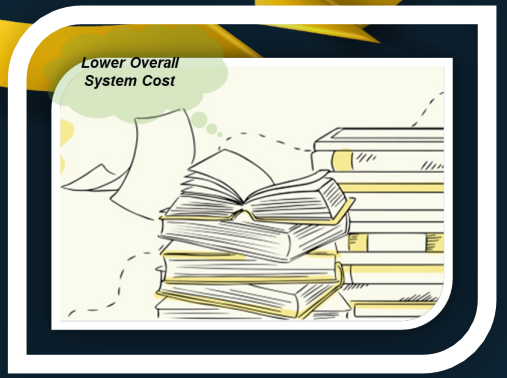
- What is the business motivation for MOSA?
- Are the selected MOSA QAs sufficient?
- Is the trace from MOSA QAs to priority clear or at least characterized?

MOSA Compliance Evaluation View



MOSA Business Objectives View

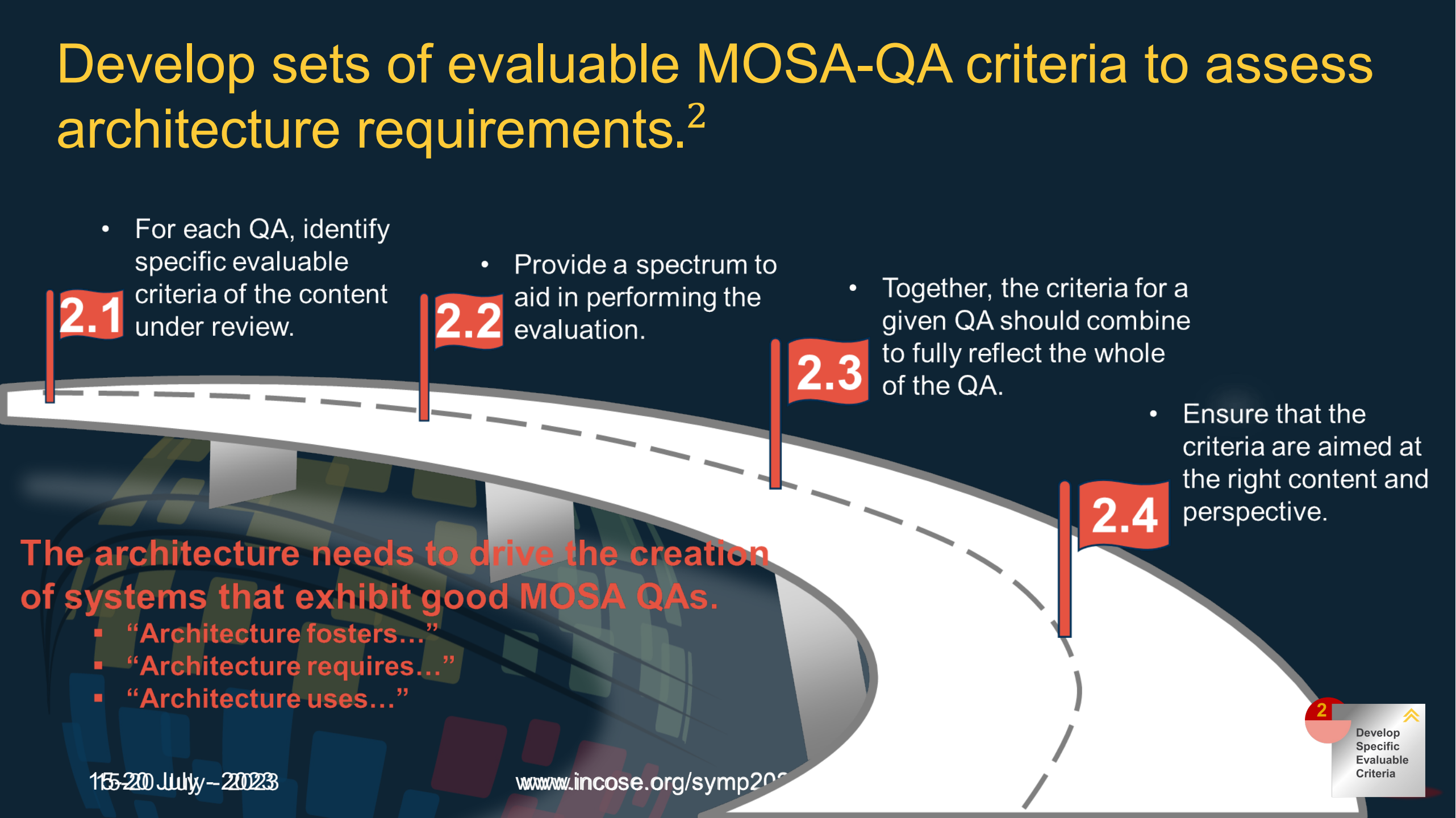




1 Prioritize Specified Business Objectives

Develop Sets of Evaluable Criteria

Develop sets of evaluable MOSA-QA criteria to assess architecture requirements.[2]



The architecture needs to drive the creation of systems that exhibit good MOSA QAs.

- "Architecture fosters…"
- "Architecture requires…"
- "Architecture uses…"

2 Develop Specific Evaluable Criteria

Develop sets of evaluable MOSA-QA criteria to assess architecture requirements.[1]

MQA-Configurability

E.g. Criterion: Architectures requires ability to adjust parameters of software systems without recompilation.

MQA-Extensibility

E.g. Criterion: Architecture fosters addition of new functionality without requiring changes to the fundamental baseline architecture.

MQA-Interface Transparency

E.g. Criterion: Architecture requires disclosure of component interface behaviour

MQA-Interoperability

E.g. Criterion: Architecture fosters a service-oriented paradigm for component interaction.

MQA-Maintainability

E.g. Criterion: Architecture fosters the use of explicit interfaces

MQA-Portability

E.g. Criterion: Architecture fosters clear definition of interfaces for potentially-replaceable components

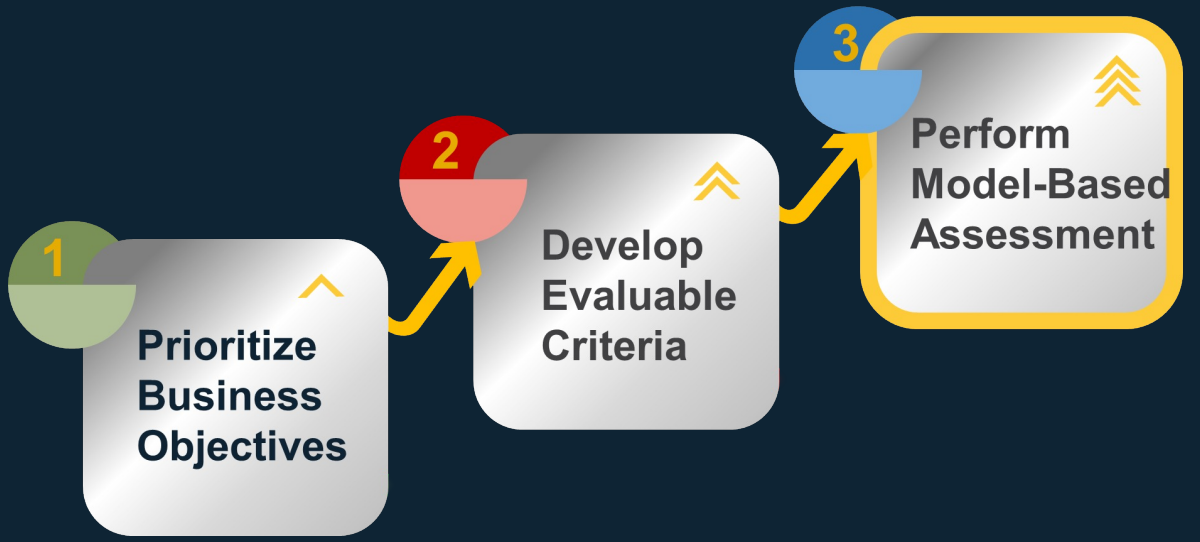
MQA-Reusability

E.g. Criterion: Architecture fosters reuse of products in other platforms without modification

MQA-Supportability

E.g. Criterion: Architecture fosters development of built-in-mechanisms for ensuring quality control

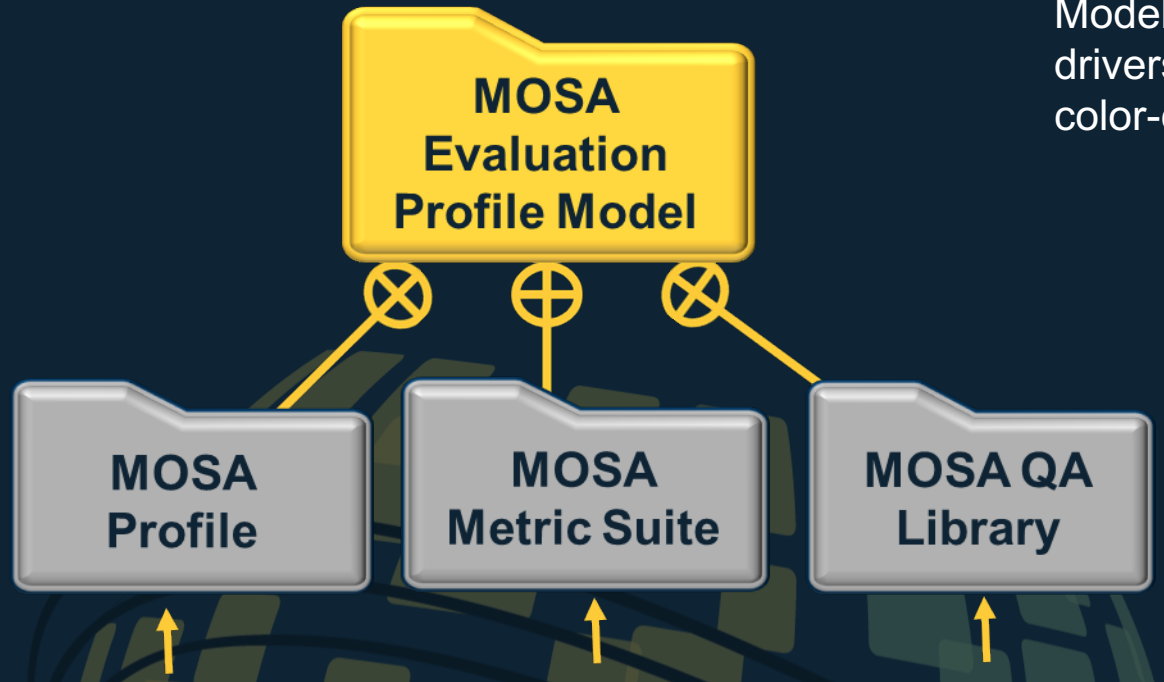
Perform Model-Based Assessment & Evaluation



STEP 3.1

Develop MOSA Evaluation Profile Model

The MOSA evaluation profile model serves as the kernel of the model-based approach.

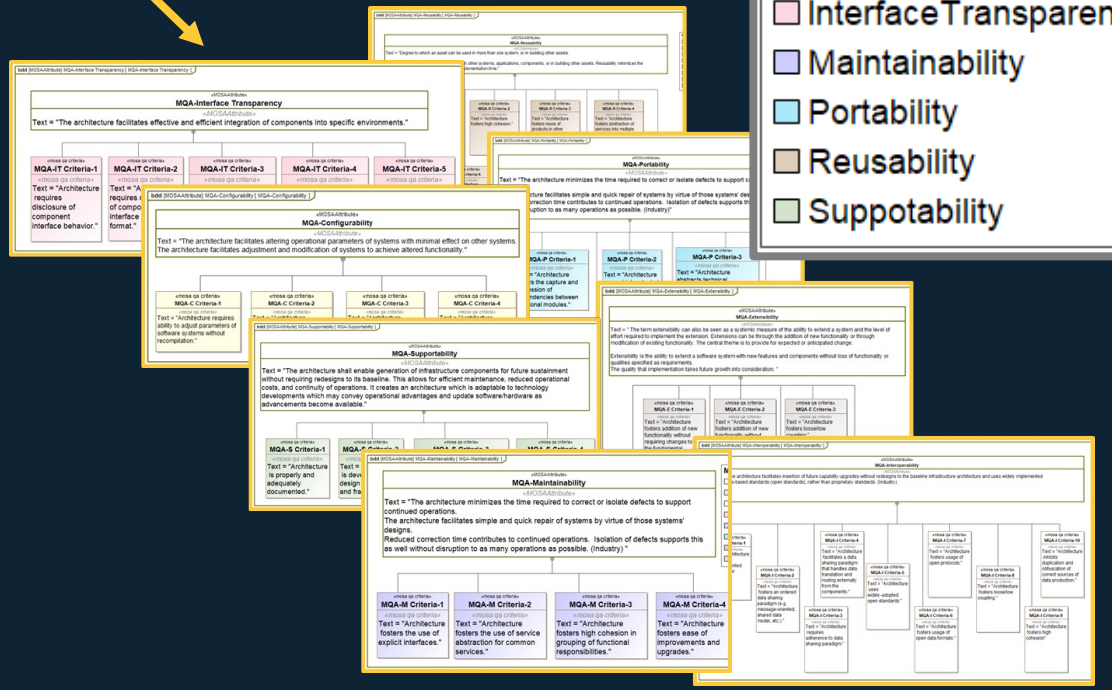


Contains MOSA stereotypes and customizations.

Contains MOSA metrics elements.

Contains MOSA QAs, MOSA-QA-criteria elements, and MOSA QA Evaluation tools.

Model representations of business drivers, quality attributes and criteria are color-coded for ease of identification.



STEP 3.1.1

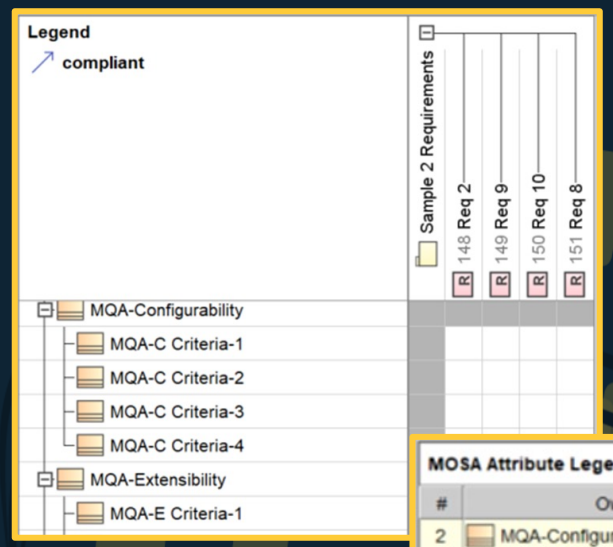
Develop Model-Based Assessment Tools

Create evaluation tools to facilitate mapping and verification of evaluation results.[1]

Mapping Tools create clear-cut dependencies between requirements and MOSA QAs.

R-C: Requirements-to-Criteria
C-R: Criteria-to-Requirements

1. QA Evaluation Matrix Tool



Requirements are loaded by setting the scope of R-C Evaluation Table.

2. R-C Evaluation Tool

#	Id	Name	Text	MOSA QA Criteria	MOSA QA Criteria Text	MOSA Quality Attribute
1	148	R Req 2	This is requirement 2			
2	149	R Req 9	This is requirement 9			
3	150	R Req 10	This is requirement 10.			
4	151	R Req 8	This is requirement 8			

C-R Evaluation Table comes preloaded with MQA-criteria.

3. C-R Evaluation Tool

MOSA Attribute Legend: Configurability Extensibility Interoperability InterfaceTransparency Maintainability Portability Reusability Suppotability

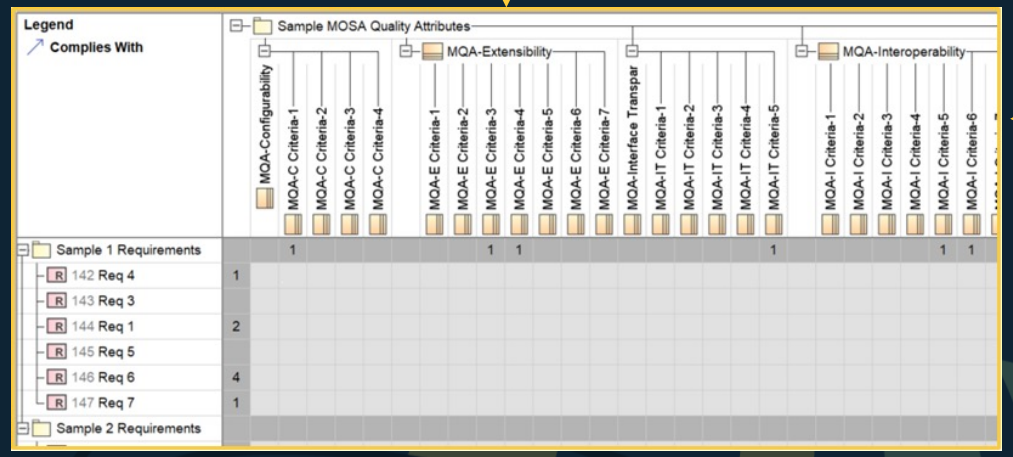
#	Owner	Name	Text	Compliant Requirement	Requirement Text	Requirement Id
2	MQA-Configurability	MQA-C Criteria-2	Architecture requires ability to adjust systems during runtime.			
3	MQA-Configurability	MQA-C Criteria-3	Architecture requires ability to perform physical adjustment of			
4	MQA-Configurability	MQA-C Criteria-4	Architecture requires multiple physical configurations of hardware			
5	MQA-Extensibility	MQA-E Criteria-1	Architecture fosters addition of new functionality without requiring			
6	MQA-Extensibility	MQA-E Criteria-2	Architecture fosters addition of new functionality without requiring			
7	MQA-Extensibility	MQA-E Criteria-3	Architecture fosters loose/low coupling.			
8	MQA-Extensibility	MQA-E Criteria-4	Architecture fosters ability to configure services at run-time.			
9	MQA-Extensibility	MQA-E Criteria-5	Architecture use widely-adopted open standard.			

3 Perform Model-Based Assessment

Create evaluation tools to facilitate mapping and verification of evaluation results .[2]

- Verification Tools provide comprehensive analysis results of an evaluation.

4. Compliance Matrix Tool



Compliance Matrix Tool comes preloaded with MQA-Criteria.

Executive Summary Tool comes preloaded with the MQAs.

6. Executive Summary Tool

#	Name	Executive Summary
1	MQA-Interface Transparency	
2	MQA-Maintainability	
3	MQA-Reusability	
4	MQA-Configurability	
5	MQA-Extensibility	
6	MQA-Interoperability	
7	MQA-Portability	
8	MQA-Supportability	

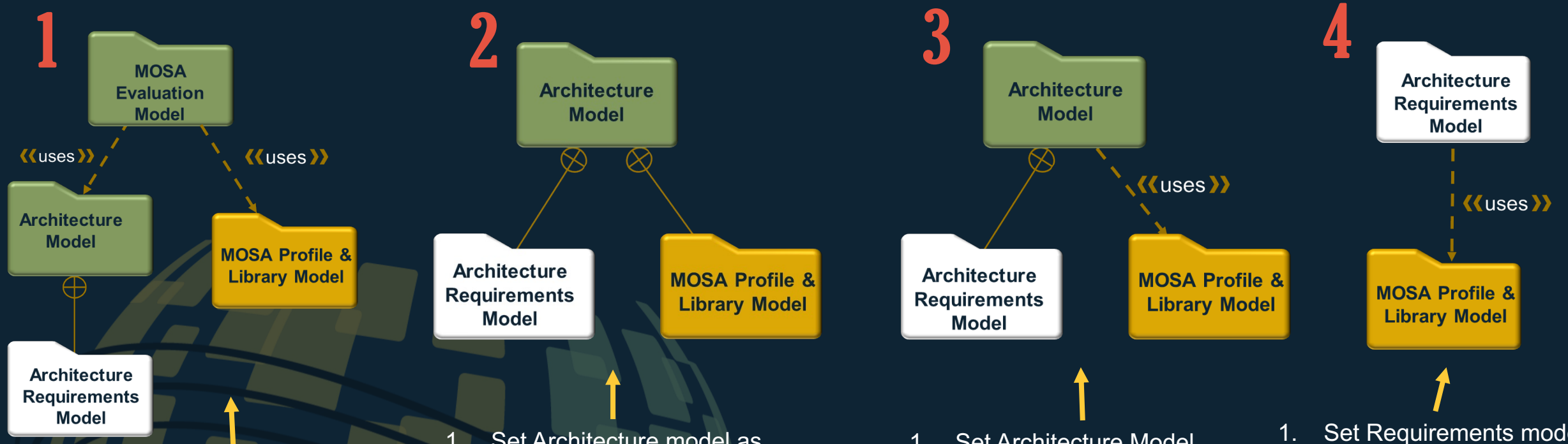
5. QA Evaluation Metric Suite

#	Date	Scope	Evaluated Requirements	Applicable MOSA Q As	Applicable QA Criteria	Traced Requireme	Configurability Criteria	Extensibility Criteria	Interface Transparency Criteria	Interoperability Criteria	Maintainability Criteria	Portability Criteria	Reuseability Criteria	Supportability Criteria
1	2022.09.29 11	Sample 1 Requirements												
2	2022.09.29 11	Sample 2 Requirements												

STEPS 3.2 & 3.3

Perform Model-Based Evaluation

Set up the evaluation model using any one of four model configuration patterns.



1. Create a new model, i.e. MOSA Evaluation Model
2. Bring in Architecture model and MOSA Profile Model as project usages.

1. Set Architecture model as MOSA Evaluation Model.
2. Import Requirements and MOSA Profile models.

1. Set Architecture Model (containing requirements) as MOSA Evaluation Model.
2. Bring in MOSA Profile model as a project usage.

1. Set Requirements model as MOSA Evaluation Model.
2. Bring in MOSA Profile model as project usage.

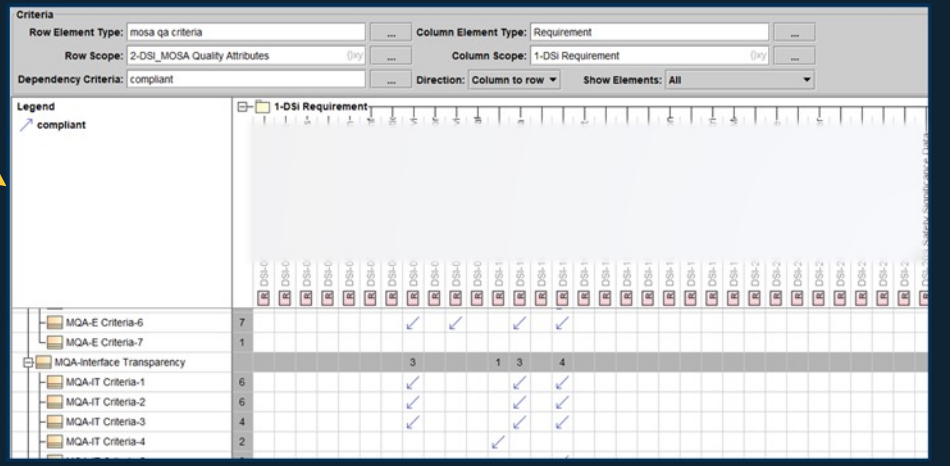
3 Perform Model-Based Assessment

Perform the evaluation:[1] create mappings between QA-criteria and requirements using the mapping tools.

A drop down menu from which to pick a *compliant Requirement* facilitates rapid mapping.

#	Owner	Name	Text	Compliant Requirement	Requirement Text	Requirement Id
1	MQA-Configurability	MQA-C Criteria-1	Architecture requires ability to adjust parameters of software systems			
2	MQA-Configurability	MQA-C Criteria-2	Architecture requires ability to adjust systems during runtime.	R 135 Req 6		135
3	MQA-Configurability	MQA-C Criteria-3	Architecture requires ability to perform physical adjustment of hardware			
4	MQA-Configurability	MQA-C Criteria-4	Architecture requires multiple physical configurations of hardware			
5	MQA-Extensibility	MQA-E Criteria-1	Architecture fosters addition of new functionality without requiring			
6	MQA-Extensibility	MQA-E Criteria-2	Architecture fosters addition of new functionality without requiring			
7	MQA-Extensibility	MQA-E Criteria-3	Architecture fosters loose/low coupling.			
8	MQA-Extensibility	MQA-E Criteria-4	Architecture fosters ability to configure services at run-time.			
9	MQA-Extensibility	MQA-E Criteria-5	Architecture use widely-adopted open standard.	R 131 Req 4	This is requirement 4	131
10	MQA-Extensibility	MQA-E Criteria-6	Baseline architecture interface is knowable/open.			
11	MQA-Extensibility	MQA-E Criteria-7	Architecture abstracts technical responsibilities into layers.			
12	MQA-Interface Transparency	MQA-IT Criteria-1	Architecture requires disclosure of component interface behavior.			
13	MQA-Interface Transparency	MQA-IT Criteria-2	Architecture requires disclosure of component interface data format.			
14	MQA-Interface Transparency	MQA-IT Criteria-3	Architecture requires disclosure of component interface data meaning /	R 134 Req 5		134
15	MQA-Interface Transparency	MQA-IT Criteria-4	Architecture does not allow for exceptions to required disclosure of			
16	MQA-Interface Transparency	MQA-IT Criteria-5	Architecture fosters the capture and expression of dependencies			
17	MQA-Interoperability	MQA-I Criteria-1	Architecture fosters a service-oriented paradigm for component			
18	MQA-Interoperability	MQA-I Criteria-2	Architecture fosters an ordered data sharing paradigm (e.g.	R 141 Req 2	This is requirement 2	141
19	MQA-Interoperability	MQA-I Criteria-3	Architecture requires adherence to data sharing paradigm.			
20	MQA-Interoperability	MQA-I Criteria-4	Architecture facilitates a data sharing paradigm that handles data			

Matrix comes preloaded with QA-Criteria.



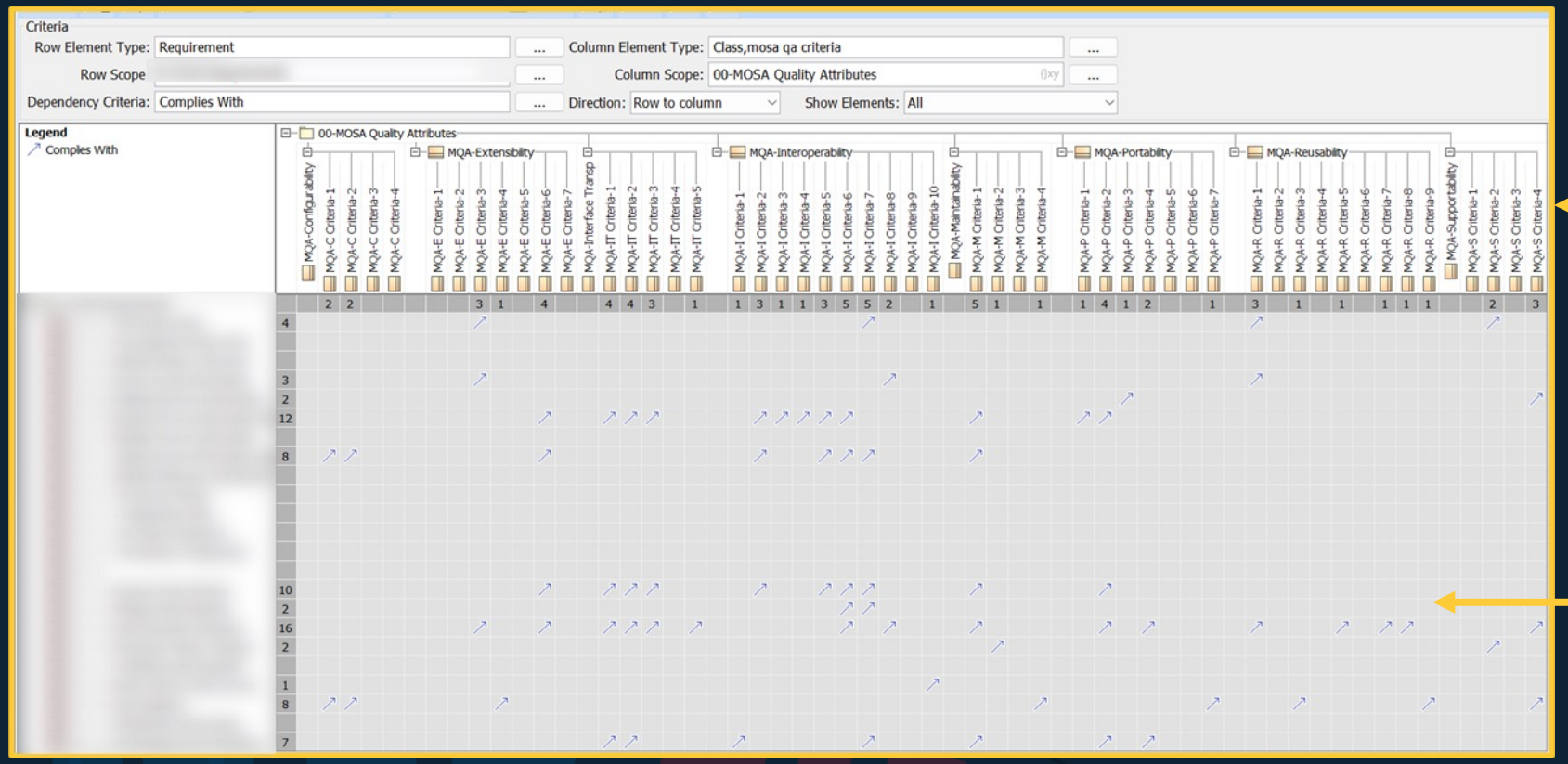
Criteria Text **and** ***Quality Attribute*** **columns automatically populate once a QA-Criteria is selected from the drop down menu.**

#	Id	Name	Text	MOSA QA Criteria	MOSA QA Criteria Text	MOSA Quality Attribute
1	R-11	R Req 11	The Communication Manager shall provide an open standards based communication interface for communication with devices..	MQA-M Criteria-1 MQA-M Criteria-3 MQA-I Criteria-5 MQA-R Criteria-6 MQA-IT Criteria-1 MQA-IT Criteria-2 MQA-IT Criteria-3	Architecture fosters the use of explicit interfaces. Architecture fosters high cohesion in grouping of functional responsibilities. Architecture uses widely-adopted open standards. Architecture fosters development of infrastructure or framework functionality Architecture requires disclosure of component interface behavior. Architecture requires disclosure of component interface data format. Architecture requires disclosure of component interface data meaning / context (e.g. via data mo	MQA-Maintainability MQA-Interoperability MQA-Reusability MQA-Interface Transparenc

3 Perform Model-Based Assessment

Perform the evaluation:[2] use the *MOSA QA* Compliance Verification tool to expedite verification of results.

The *read-only* tool facilitates independent analysis of results without the risk of creating or deleting traceability relationships.



The compliance matrix is preloaded with QA criteria.

The compliance matrix auto-populates in real-time with the results of an ongoing mapping activity.

Perform the evaluation:^{3a} compute the level of QA compliance using the QA Evaluation Metric table.

Scope field is set to compile metrics for single specification

Total number of QA-criteria requirements map to.

Total number of applicable QAs requirements map to.

#	Date	Scope	Evaluated Requirements	Applicable QA Criteria	Applicable MOSA Q As	Configurability Criteria
1	2022.03.04 08.37	Specification	34	17	6	1
2	2022.03.04 08.56	Specification	34	17	6	1

Total number of requirements evaluated

Total number of criteria per QA to which requirements trace to.

ty	Extensibility Criteria	Interface Transparency Criteria	Interoperability Criteria	Maintainability Criteria	Portability Criteria	Reuseability Criteria	Supportability Criteria
	4	5	0	1	1	6	0
	4	4	0	1	1	6	0

3 Perform Model-Based Assessment

Perform the evaluation:^{3b} compute the level of QA compliance using the QA Evaluation Metric table.

Scope field is set to compile metrics for multiple specification

Total number of evaluated requirements.

Actual number of requirements traced to a QA.

#	Date	Scope	Evaluated Requirements	Applicable QA Criteria	Traced Requirements	Applicable MOSA Q As	Configurability Criteria
1	2023.07.03 21.16	Framework Specification	6	11	6	6	0
2	2023.07.03 21.17	Manager Specification	5	7	5	4	0
3	2023.07.03 21.19	Resource Specification	3	5	3	3	0

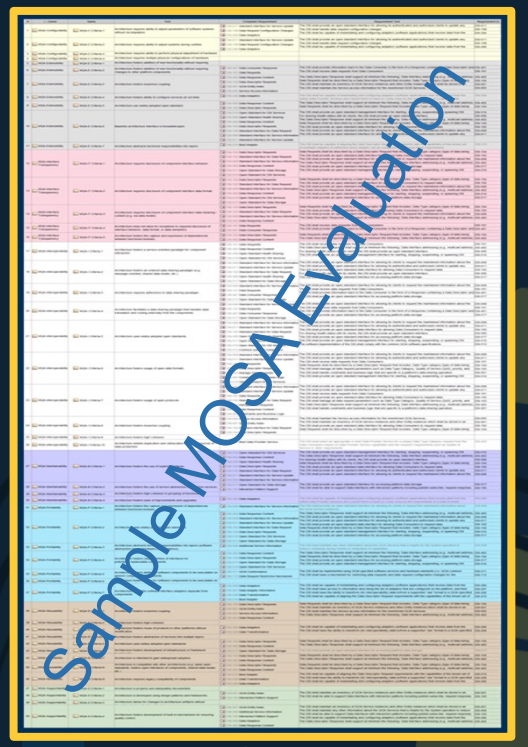
Total number of criteria per QA to which requirements trace to.

ıbility ıa	Extensibility Criteria	Interface Transparency Criteria	Interoperability Criteria	Maintainability Criteria	Portability Criteria	Reuseability Criteria	Supportability Criteria
	0	1	1	1	3	3	2
	0	3	1	2	0	1	0
	0	3	0	1	1	0	0

3 Systematic & Traceable MOSA Process

Perform the evaluation:[4] Interpret, summarize, and capture findings in the model.

- Use systems engineering judgement to make an overall assessment of the architecture against each quality attribute.
- An Executive Summary Tool is prepared for each architecture specification.



HCC C-R MOSA Evaluation

- Determine whether a high, medium, or low threshold was met by each QA's criteria
  - Or indicate that it was non-indicative

E.g.

HCC Spec. fosters a moderate degree of: Configurability

HCC Spec. fosters a high degree of: Reusability

HCC Spec. is non indicative for Extensibility

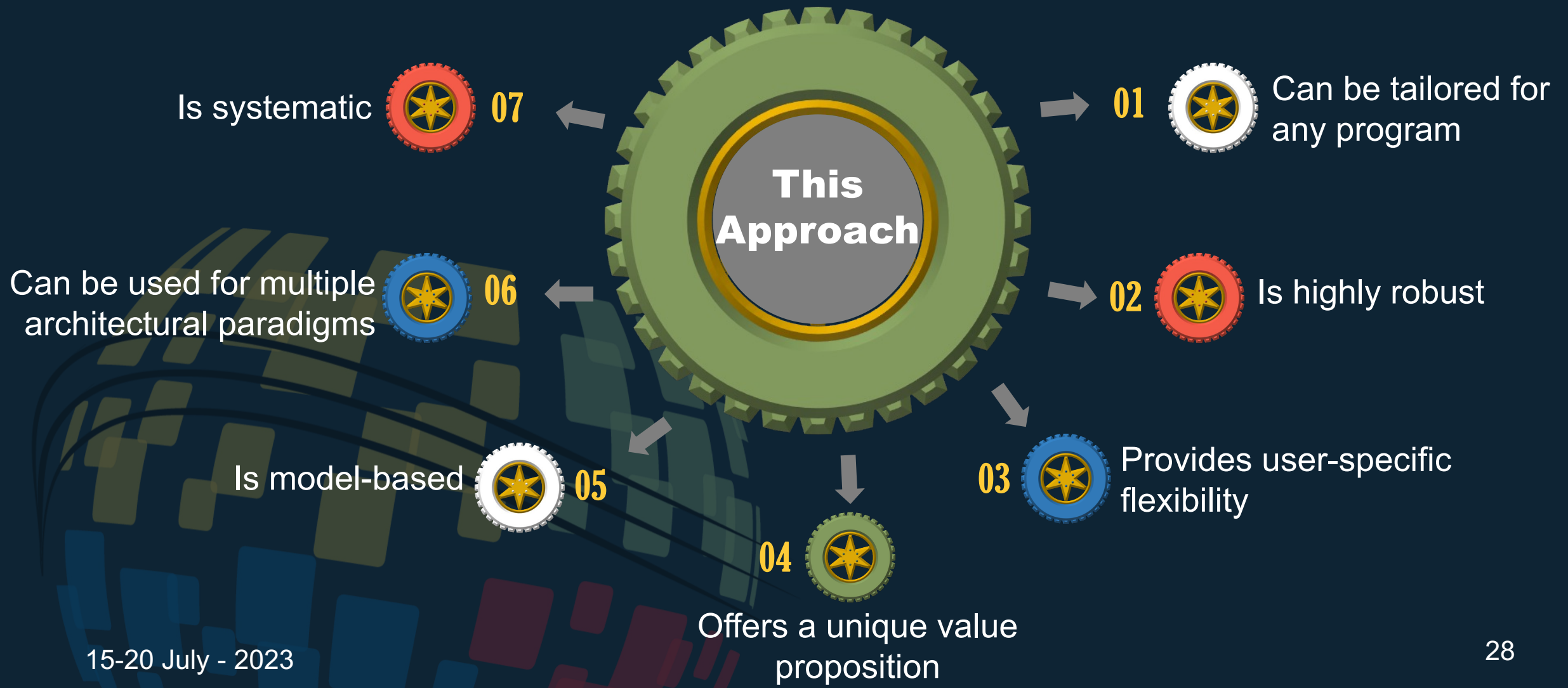
#	Name	△ Executive Summary
1	HCC-MQA-Interface Transparency	HCC fosters a high degree of interface transparency. Disclosure of HCC interface behavior, data format, and data meaning is specified implicitly for HCC connectors via required compliance to applicable open standards.
2	HCC-MQA-Maintainability	**HCC fosters a high degree of maintainability.** HCC speicifies ease of improvements and upgrades via its requirement for SBC or OpenVPX compliance to widely-adopted VITA open standards and chassis hosting processing cards.
3	HCC -MQA-Reusability	**HCC fosters a high degree of reusability / modularity.** This assessment is based on the assumption that the HW components and standards supported by the HCC are in fact widely-adopted open standards and therefore meet or exceed the indicated Reusability criteria. Also the specifications call for discrete changes to components, so as not to affect other systems
4	HCC-MQA-Configurability	**HCC fosters a moderate degree of configurability.** HCC implicitly requires ability to perform physical adjustment of HCC components and allows greater than one HCC component physical instantiations. The first two configurability MQA criteria are inapplicable to HCC since they are software-related
5	HCC-MQA-Extensibility	**HCC fosters a moderate degree of extensibility.** This assessment is based on the assumption that the HW components and standards supported by the HCC are in fact widely-adopted open standards and therefore meet or exceed the indicated Reusability criteria. Coupling is called for to be limited, however the specificity of the degree and method to this could be expanded upon
6	HCC-MQA-Interoperability	**HCC fosters a moderate degree of interoperability.** For HCC, the usage of military standards, data formats, and widely-used protocols means that it is more likely that HCC components will be naturally compatible with each other, and not require significant adaptation. This assessment is based on the assumption that the standards supported by the HCC are in fact widely-adopted open standards and therefore meet or exceed most of the listed Interoperability criteria. If this is not the case, then the HCC in GCIA-compliant platforms will be significantly less interoperable because their interface behavior, format, and semantics will require more effort to understand and integrate.
7	HCC-MQA-Portability	**HCC fosters a moderate degree of portability.** HCC requires use of standard interfaces and connector specifications. Mandating considerations for lowering coupling between interfaces increases likelihood of usability in other systems. However, the specificity of the degree or method for limiting this could be expanded upon to increase the real-world realization of lower coupling. Usage of common computing fosters execution of components on generic computing environments. Connectors are also standardized.
8	HCC-MQA-Supportability	**HCC fosters a moderate degree of supportability.** Considerations for low coupling is mandated, and use of standardized connector specifications provide baseline documentation of interfaces, while still allowing for flexibility of pinouts.

Executive Summary Tool for a Hardware Common Compute (HCC) Specification

3 Perform Model-Based Assessment

Conclusion

In conclusion, this digital engineering tool provides early confidence that an architecture fosters MOSA principles.



Questions

A Systematic and Traceable MOSA Evaluation Process for Systems Architectures: A Digital Engineering Tool

15-20 July - 2023

Georgia Tech Research Institute

33rd Annual INCOSE international symposium

hybrid event

Honolulu HI USA

www.incose.org/symp2023

#INCOSEIS