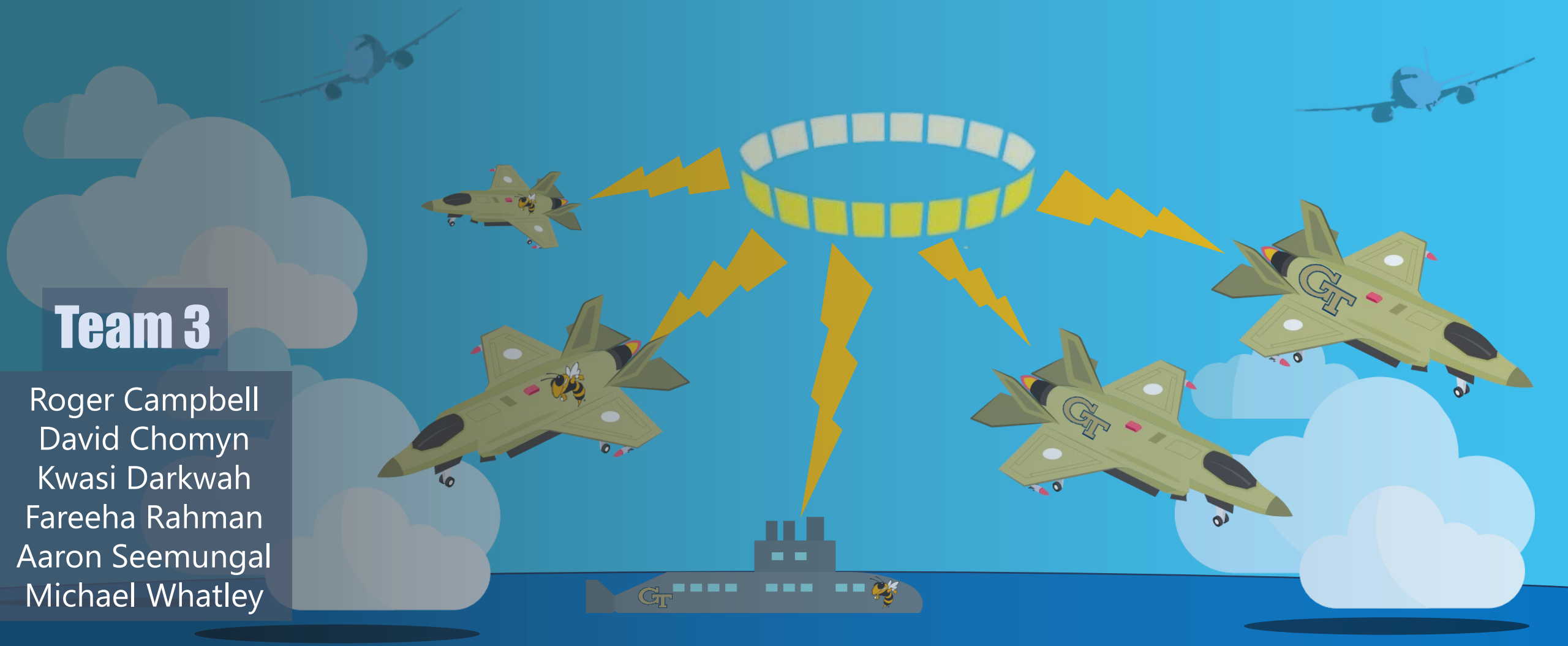
ASE6104 Capstone Final Presentation

Multi-Tactical Data Link Architecture (MTA) Analysis Tool



Team 3

Roger Campbell
David Chomyn
Kwasi Darkwah
Fareeha Rahman
Aaron Seemungal
Michael Whatley

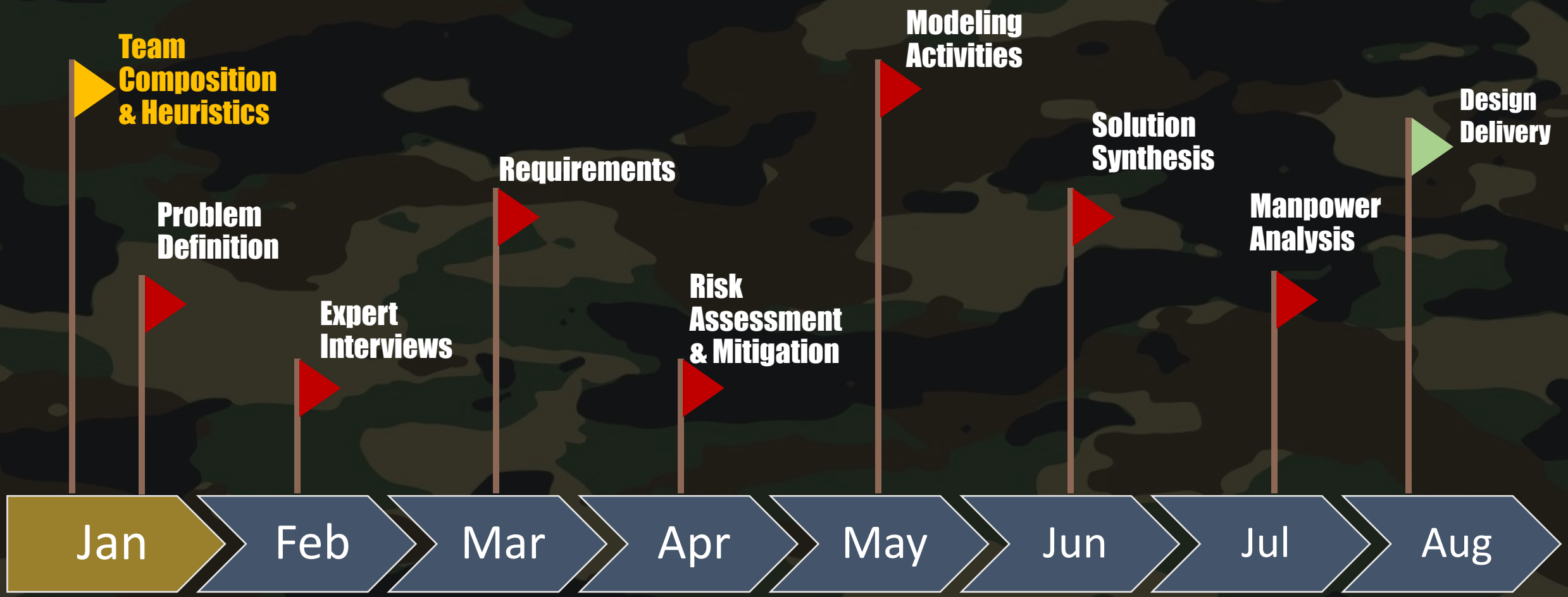
Team 3 Composition

Georgia Tech Capstone DC

Michael Whatley	Fareeha Rahman	David Chomyn	Roger Campbell	Aaron Seemungal	Kwasi Darkwah
					
TDL Domain Engineer & SME	**Technology Product Manager**	**Software Architect**	**MBSE Engineer**	**Communications, Analytics & Visualization**	**Principal Research & Communications Engineer**

1. Team Composition & Heuristics

Georgia Tech Capstone DC



1.1 Personal Heuristics

Georgia Tech Capstone DC

Ask questions to illuminate any gaps in proposed solutions.

Surround yourself with a world-class team you can trust.

1.2 Know your Sponsor

Georgia Tech Research Institute (GTRI)

Sensors & Electromagnetic Applications Laboratory **(SEAL)**

Portfolio $150 million
of sponsored research

Research Areas

Intelligence, surveillance, & reconnaissance (ISR)	Air & missile defense	Foreign material exploitation and electromagnetic systems	Electronic attack/electronic protection (EA/ EP).

SEAL develops advanced signal and data processing methods for :

1. Acoustic sensors
2. Multi-sensor intelligence exploitation architectures and algorithms covering all wavebands.

C3D Division
75 + persons

Communication is the glue that makes everything else operate effectively.

You can't command and control something if you have not established contact!

DC

1.3 Capstone Project Background

DC

“The DoD needs the ability to conduct trade analysis for increased and accurate situational awareness “

“Create a reference framework and MTA Model Based System Engineering artifacts“

“Reference MTA framework to physics-level modeling of current and future sensors and communications architectures. ”

“An estimated **300-400** hours would be sufficient”

“The MTA tool should be testable“

1.4 Initial Conversation 1/23/2023

DC

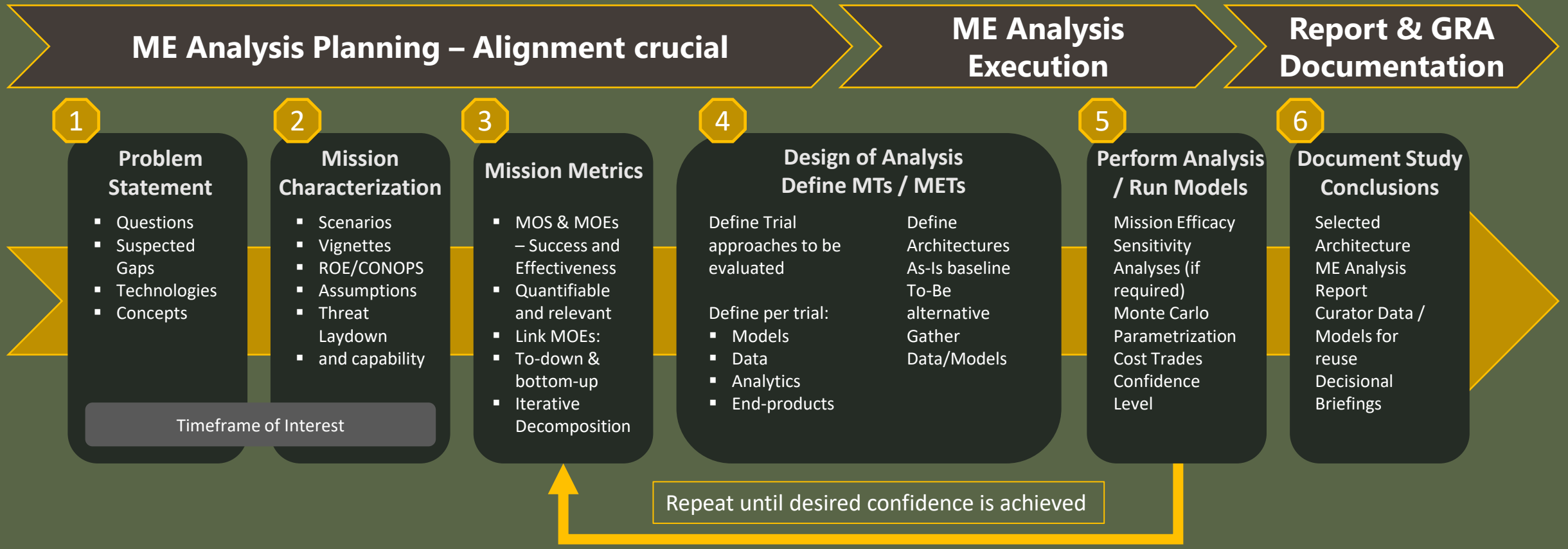
How do we provide assurance that the emerging technologies can be incorporated onto legacy platforms, and how do we know they will scale effectively?

Clarified Scope of Project between Framework Design and Software Design

Diagrams and relationships may have been traced by hand, which is inefficient and prone to error

1.5 DoD Mission Engineering Process

DC

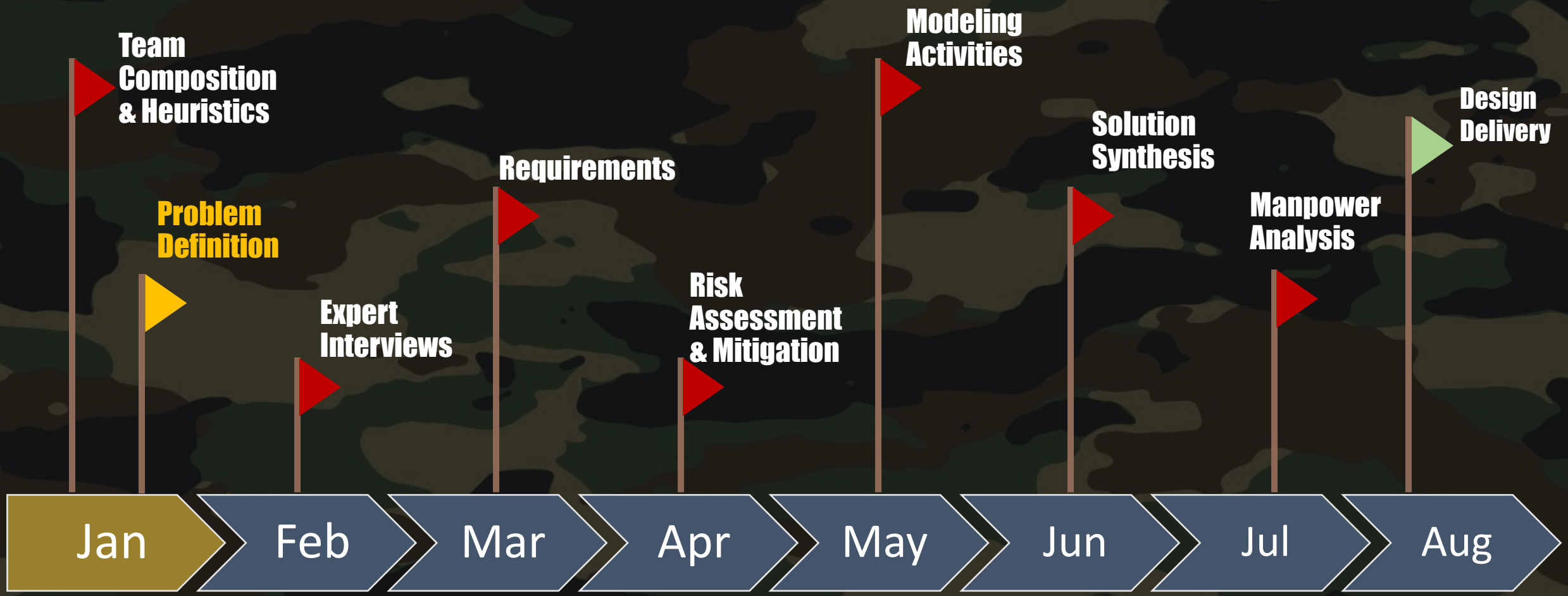


ROE - Rules of Engagement MOS – Measures of Success MOE – Measure of Effectiveness MET – Mission Engineering Thread MT - Mission Thread

2. Problem Definition

Georgia Tech Capstone

MW



2.1 Purpose – How did it arise?

MW

Was the Purpose at Onset or an Evolution?

Started at Onset, but can evolve into many other areas of analysis such as satellite, internet and HF pathways.

To reduce time spent building Modeling Viewpoints and understanding communication Parametrics

Who's this for?

The Stakeholders: Sponsor, Client = Stakeholders. GTRI – SEAL/C3 Division

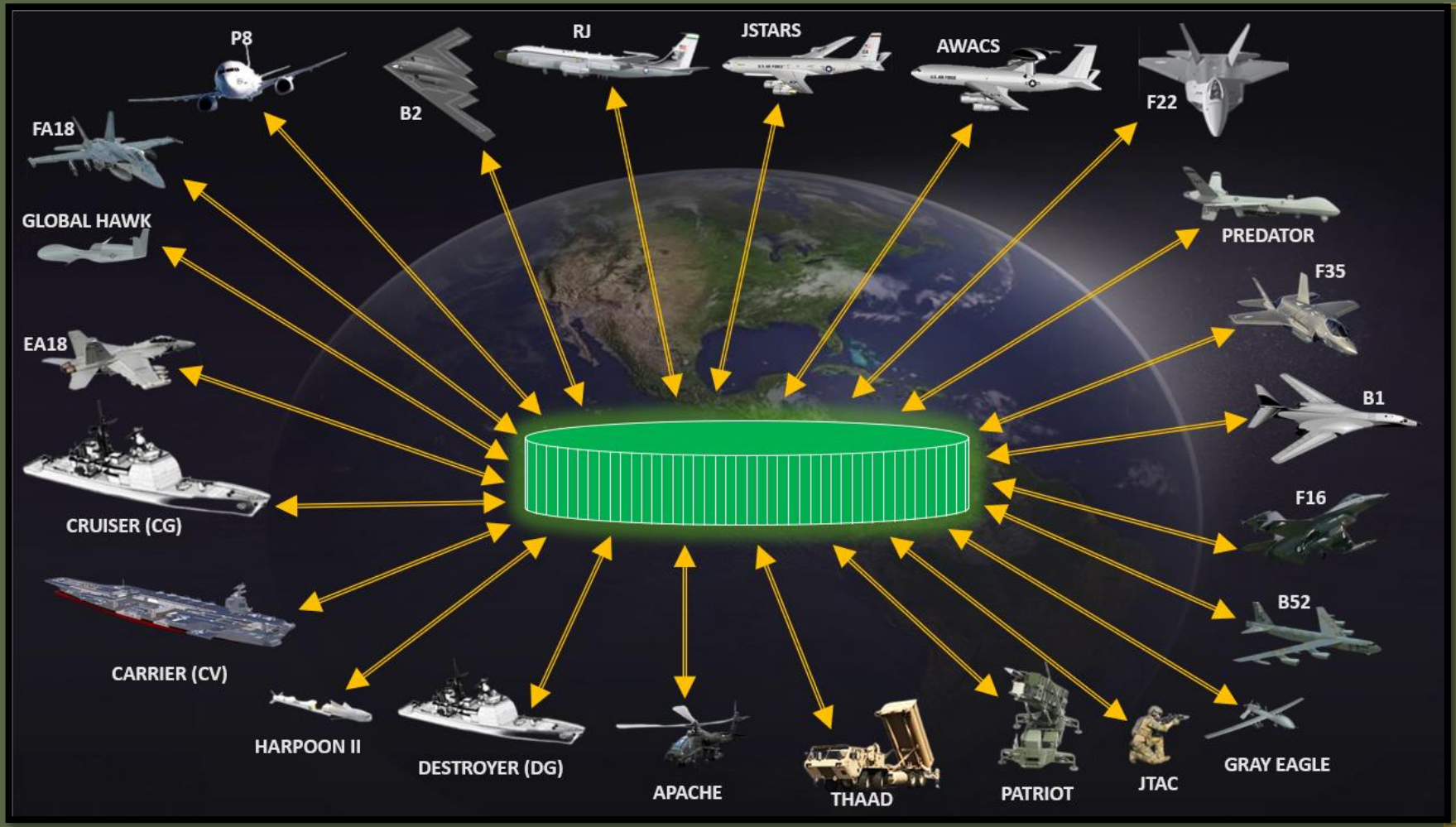
<u>Ultimate</u> End Users: For C3 System operators and military decision-makers

Is this unprecedented?

No, *Cyber Assassin* is a tool by Systems Planning & Analysis, Incorporated. Sponsor stated this tool is too costly for licensing needs.

2.2 Topic Area Summary

MW



A TDL uses a data link standard in order to provide communication via radio waves, cable or satellites used by United States, Allies & Partnered nations.

Military weapons systems use TDLs.

Tactical Information

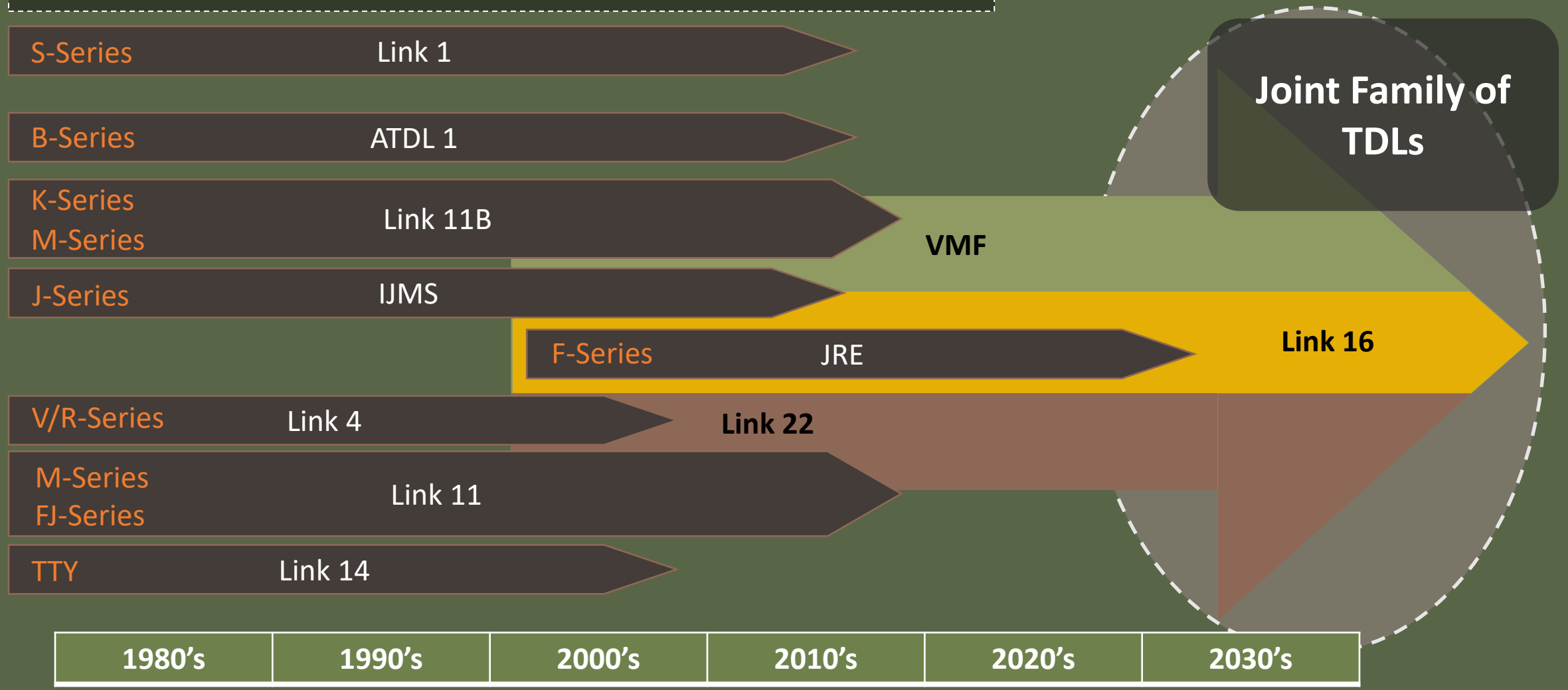
TRANSMIT	RELAY	RECEIVE

Link-16 & Link-22 are TDL architectures that use Standards such as MIL-STD-6016 and STANAG-5522.

Joint Tactical Radio System (MIDS-JTRS) is a software defined radio for Link-16.

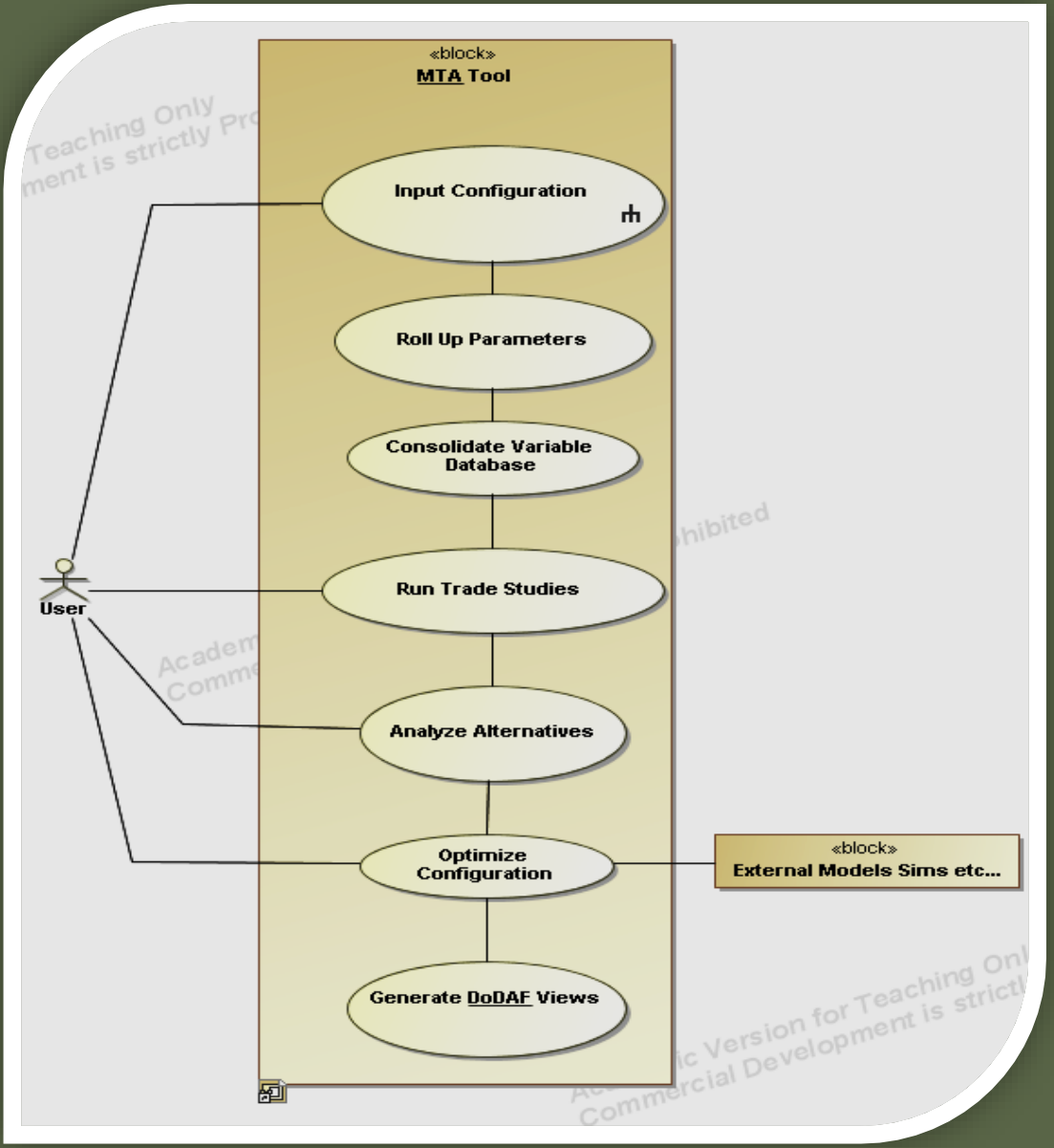
TDL - Tactical Data Link

2.3 Multi Link Environment



MW

2.4 The Larger SoS



- ✓ A **platform** that provides **exchange** of voice, video, and data **communications** between military entities- ground vessels, aircrafts, & navy vessels in real-time
- ✓ Made to operate in challenging environments
- ✓ Uses a **combination** of line-of-sight communication links and satellite
- ✓ Uses **standardized** data formats & protocols to ensure interoperability between links
- ✓ **Integrates** with other mission systems to bring awareness to commanders

2.5 What needs are being addressed?

Georgia Tech Capstone

MW

Purpose of Design:
Develop an organic Link-16 & Link-22 communications analysis tool to produce DoDAF views:

OV-1	OV-2	OV-3	OV-5a/b	OV-6c	SV-1	SV-2	SV-5a/b	SV-6	SV-10c	StdV-1	StdV-2

Problems

Mismatched Link-16 & Link-22 systems that do not exchange information.

Software Defined (e.g., MIDS-JTRS) and Legacy Radios miscommunications

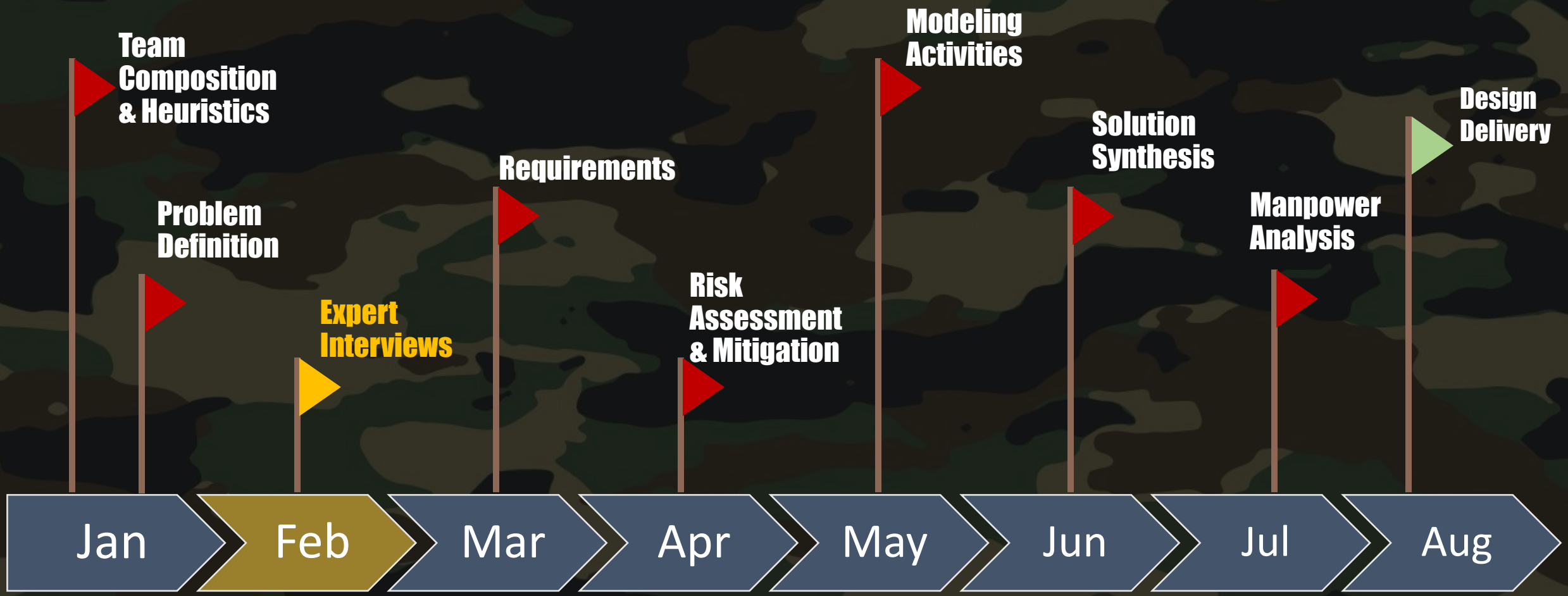
Not all platforms have both Link-16 or Link-22 systems

No organic, low-cost tool to conduct communication analysis for C3 research projects for DoD.

*DoDAF stands for **DoD Architecture Framework***

3. Expert Interviews

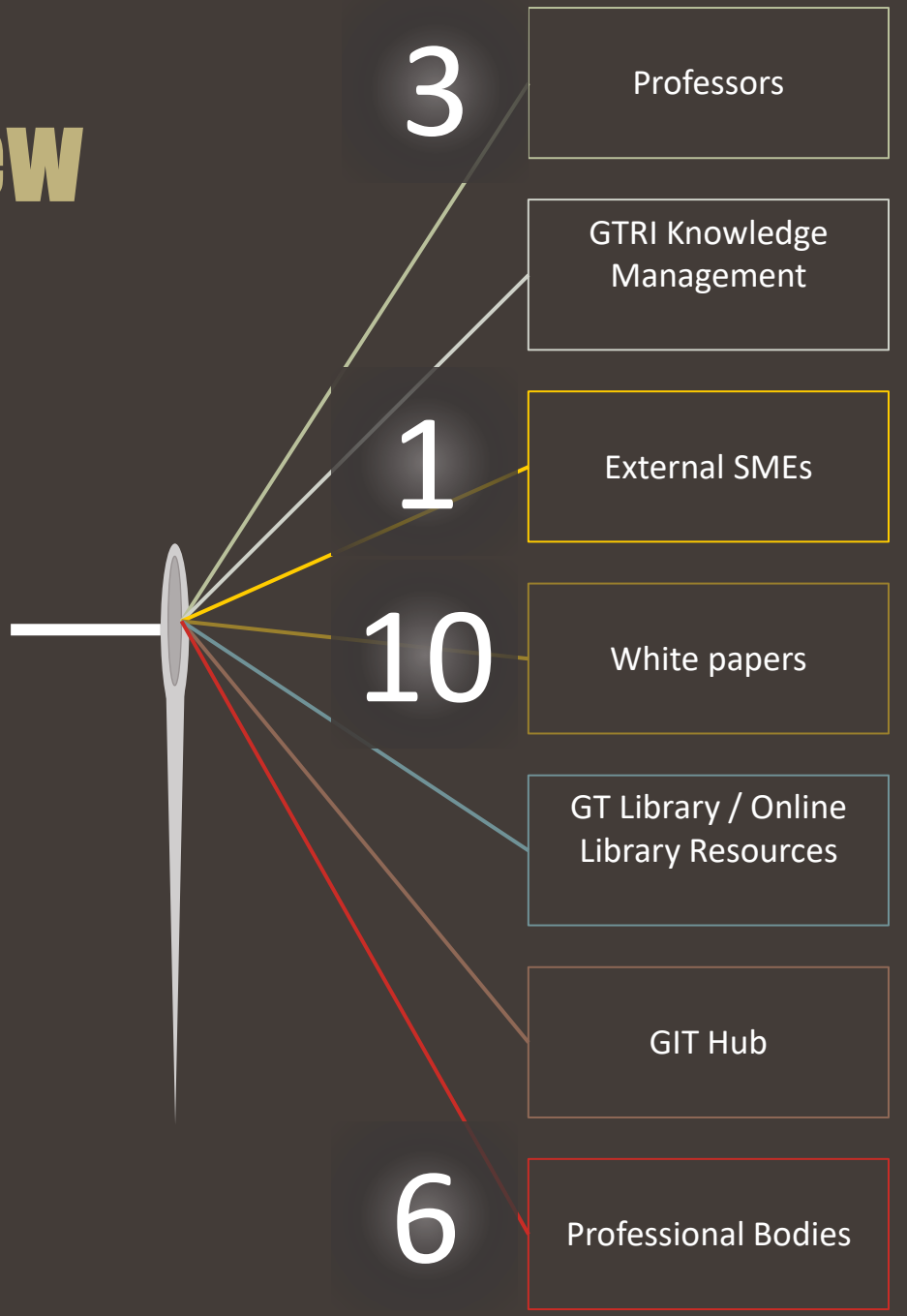
Georgia Tech Capstone
RC



3.1 Literature Review

How did we plan and execute the research?

- Examined what we could from available industry products
- Identified options to research
- Divided and conquered identified approaches



RC

3.2 Operational View – 1 : AoA

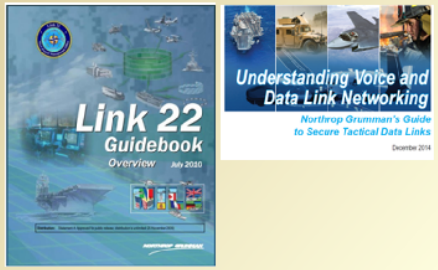
bdd [Package] MTA Tool OV-1[1 OV1 Concept View]

00 Start Here - Grand Menu

TDL Military Specifications

parts

link 16 : Link 16
link 22 : Link 22



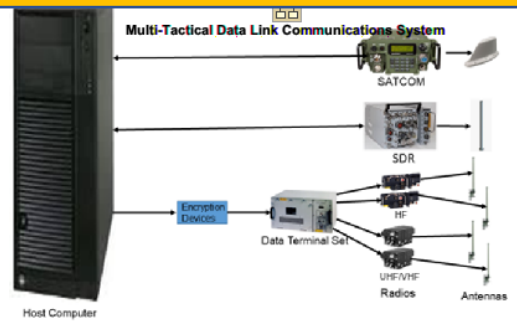
Tactical Data Link Specification Data

Tactical Data Link Specification Data

MTA Tool

MTA Analysis Tool

Multi Tactical Data Link Communication System



View Information

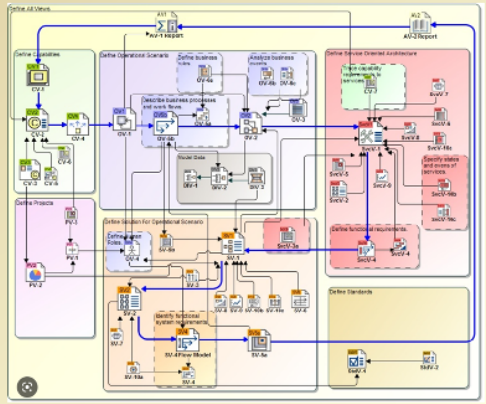
DoDAF Views

parts

OV-1 : OV-1

references

OV : OV
SV : SV
StdV : StdV
AV : AV

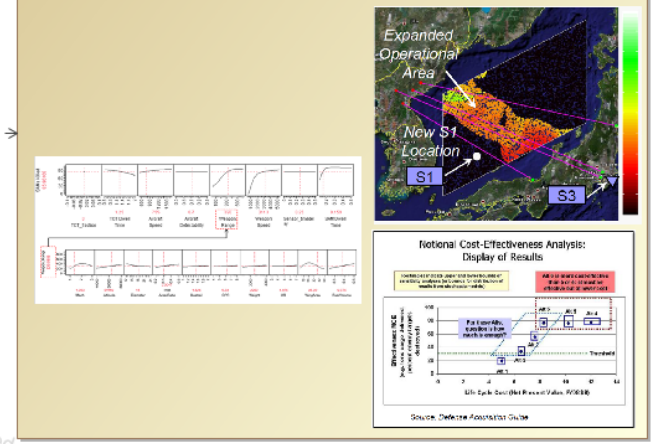


Analysis Results

Analysis Results File

Analysis of Alternatives

Analyses of Alternatives Tool

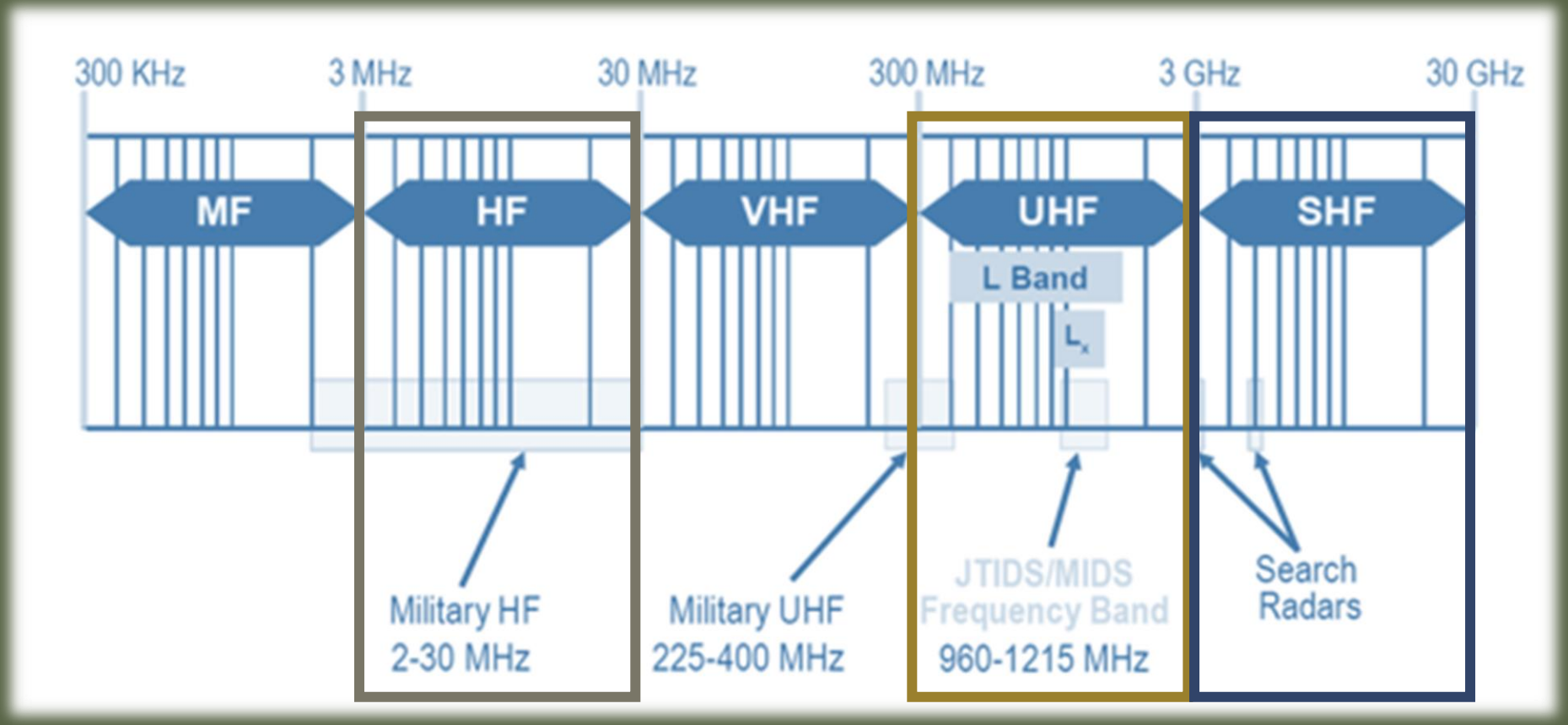


System Analysis

Systems Analyses

3.3 TDL Frequency Spectrum Bands

RC

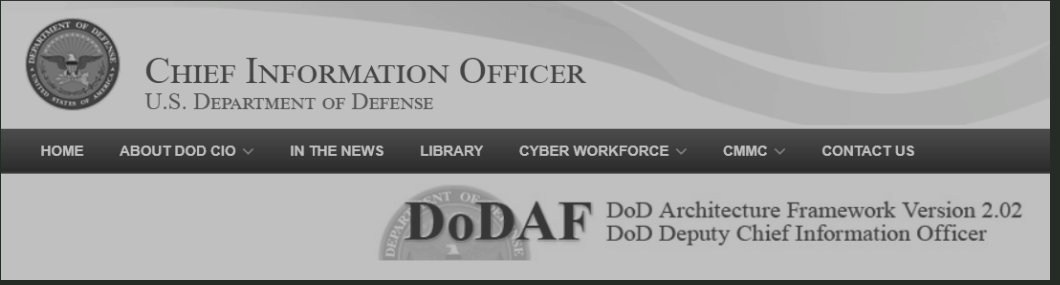


Satellites operate in SHF band

High Frequency

Ultra High Frequency

Super High Frequency



3.4 Creating DoDAF

Georgia Tech Capstone

RC

- OV-1
- OV-2
- OV-3
- OV-5a/b
- OV-6c
- SV-1
- SV-2
- SV-5a/b
- SV-6
- SV-10c
- StdV-1
- StdV-2

Initial views requested by Project Sponsor (x14)

De-Scoped to Top 3 Views

View Product
AV-1
OV-1
TV-1
STD-1

View Product
SV-2
SV-4
SV-6

Resource Flow Specification (Ports?)

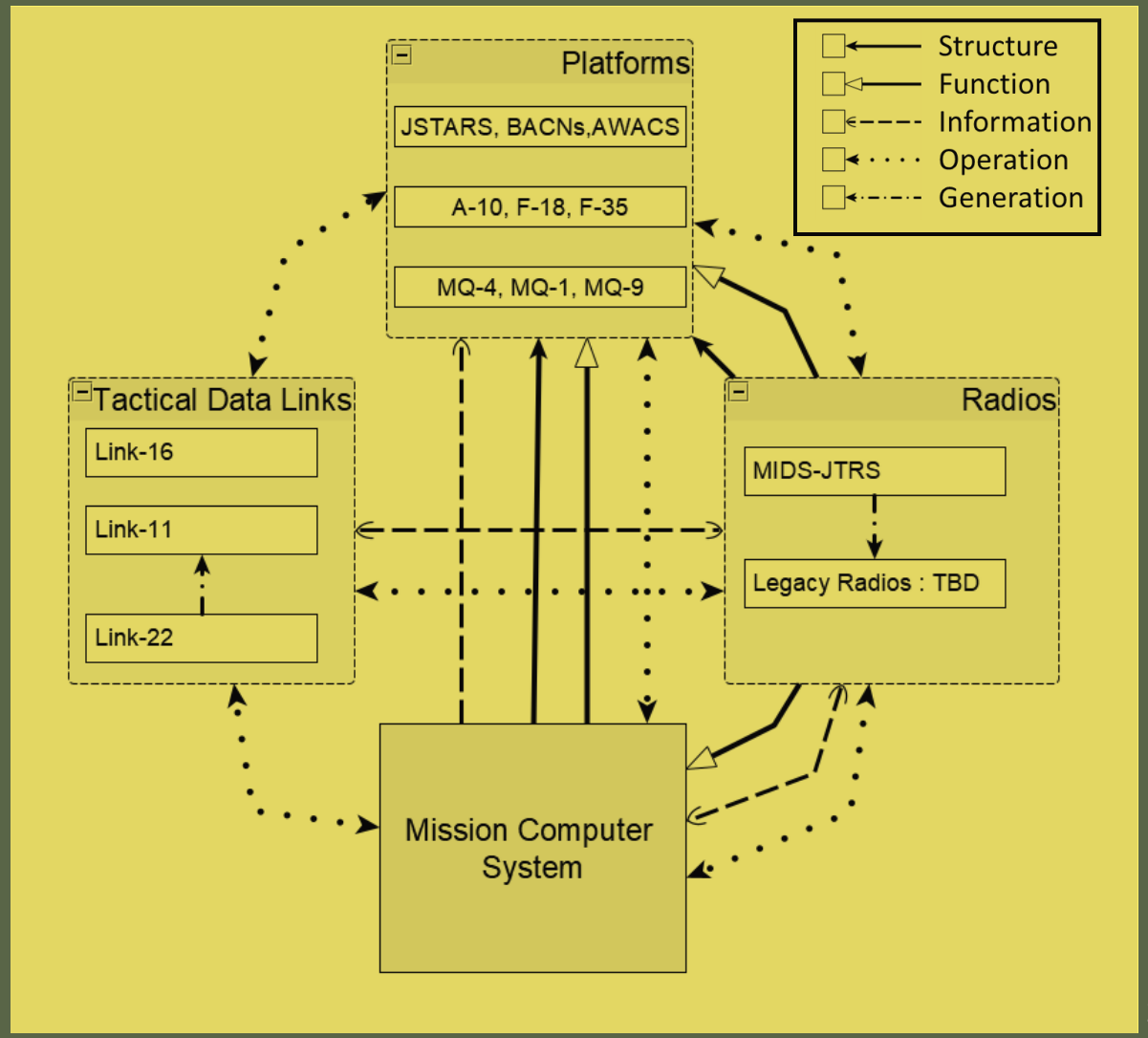
System Functionality Description (Function?)

System Resource Flow Matrix (Exchanges?)

3.5 Constituent System Relations

RC

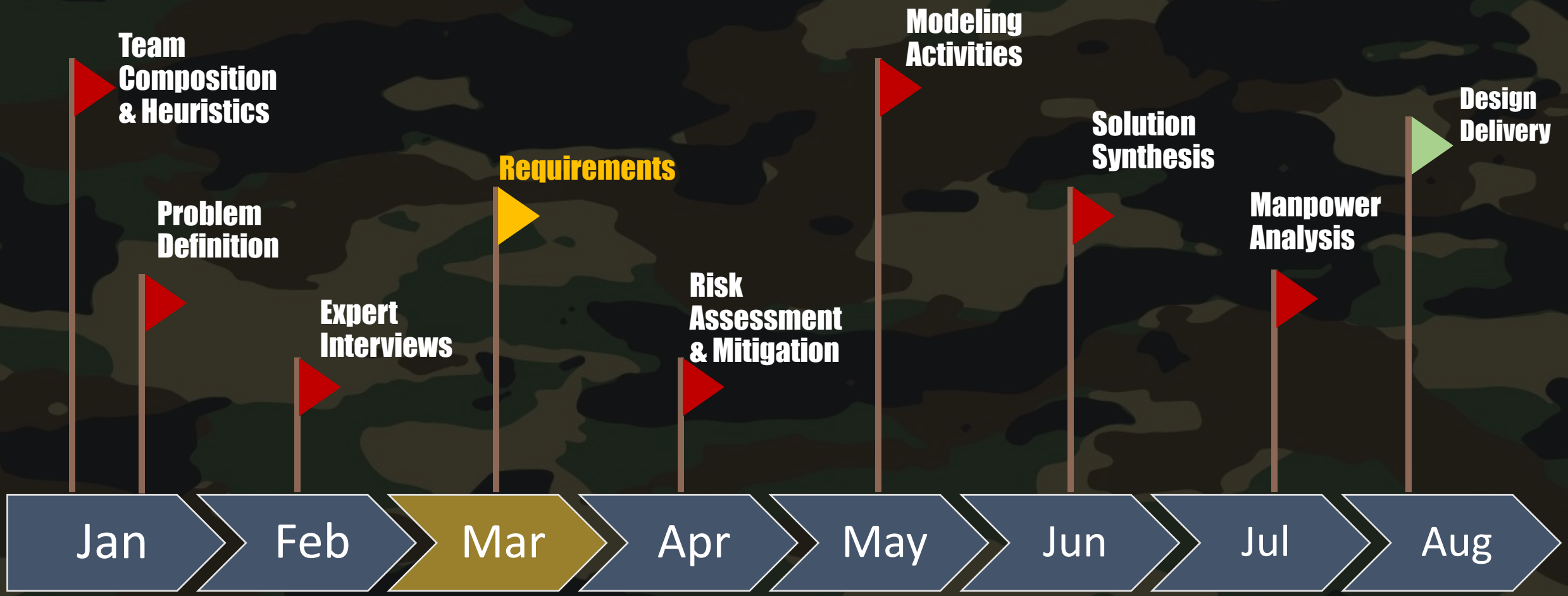
- **Structure:**
  - If one is a component or basis of the other.
- **Function:**
  - If one system requires certain functions or services by another system to perform its own function.
- **Information:**
  - If requirements or information is exchanged between the two.
- **Operation:**
  - If they are both used in an operation scenario to jointly fulfill a mission.
- **Generation:**
  - If one system will be a replacement of the other.



4. Requirements

Georgia Tech Capstone

AS



4.1 Requirements

S - Specific (simple, sensible, significant)
M - Measurable (meaningful, motivating)
A - Achievable (agreed, attainable)
R - Relevant (reasonable, realistic)
T - Time bound (time-based, time limited)

TEC	General Technical Requirements
BUS	Business Requirements
ERR	Error Handling Requirements
DBM	Database/Data Management Requirements
PTT	Production, Test and Training Requirements
PER	System Performance Requirements
SEC	Security
REL	Reliability and Availability
REG	Regulatory

MoSCoW Method Requirements Prioritization

Mo	**MUST HAVE - Things You can't live without**
S	**SHOULD HAVE - Things considered important, not vital**
Co	**COULD HAVE - Good to have**
W	**WON'T HAVE - Little to no value**

Requirement Taxonomy

DBM_1

Requirement Category

Requirement Number

AS

4.2 Requirements Verification

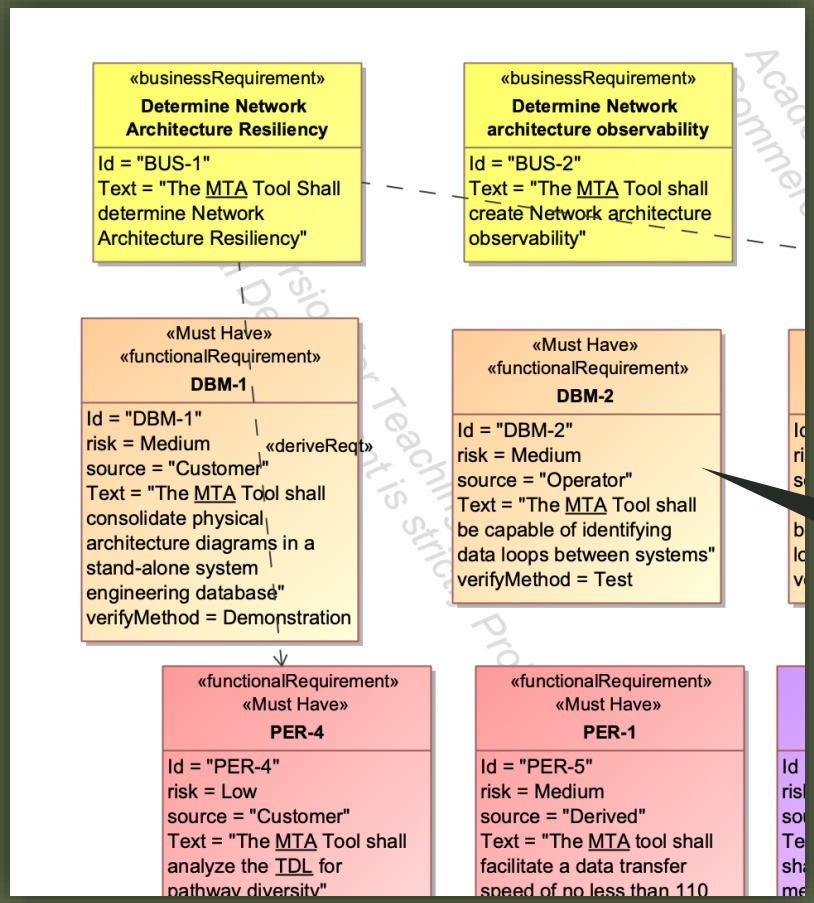
AS

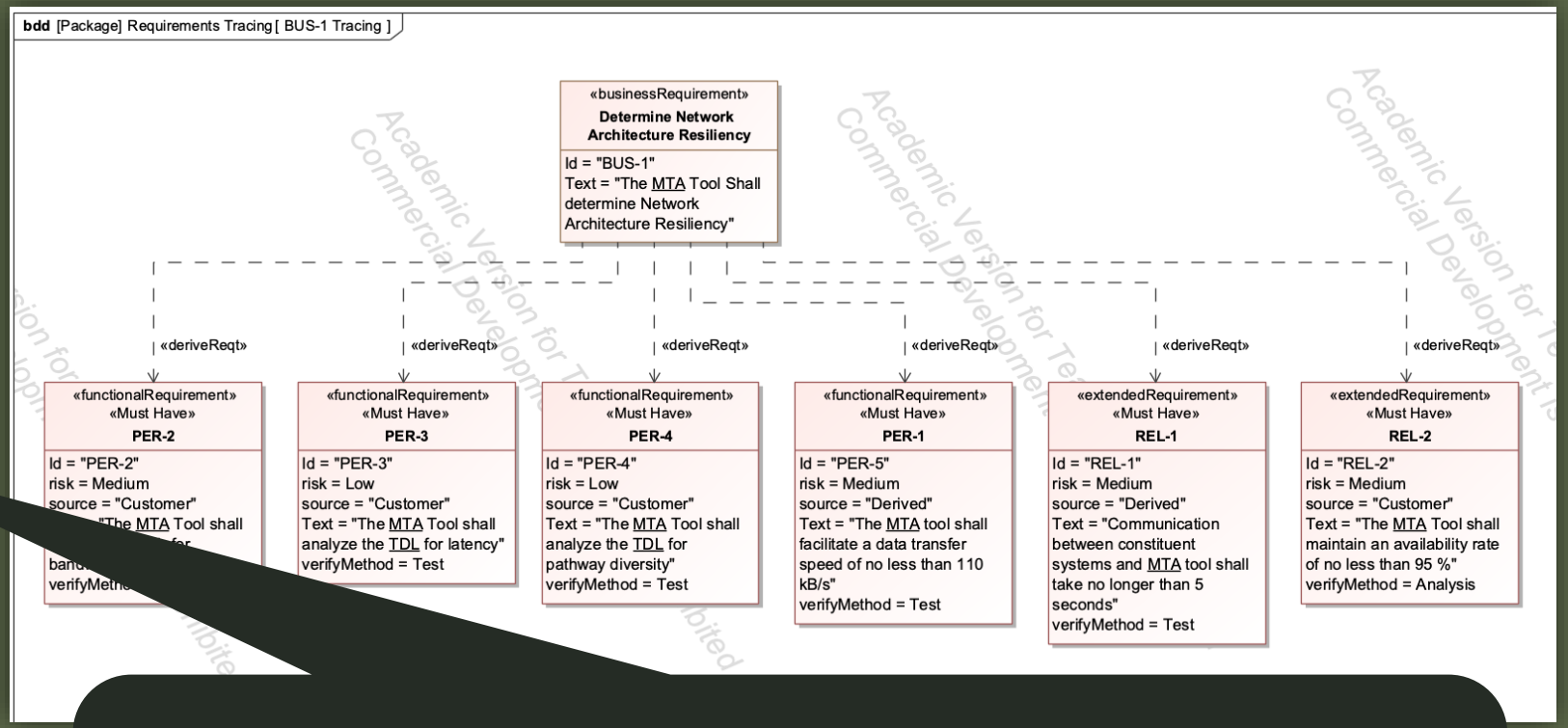
#	Id	Text	Source	Verify Method
1	DBM-4	The MTA Tool shall be compatible with C++ codex	Customer	Test
2	PTT-3	The MTA Tool shall allow for the IPERF tool analysis	Customer	Test
3	PTT-4	The MTA Tool shall allow for the IPERF tool analysis	Derived	Analysis
			omer	Inspection
			ved	Inspection
			omer	Test
			omer	Inspection
			omer	Test
			omer	Inspection
			omer	Inspection

Inspection	**The non-destructive examination of a product or system using one or more of the five senses (visual, auditory, olfactory, tactile, taste).**
Demonstration	**The manipulation of the product or system as it is intended to be used to verify that the results are as planned or expected.**
Analysis	**The verification of a product or system using models, calculations and testing equipment.**
Test	**A series of inputs, data, or stimuli to ensure that the product or system will produce a very specific and predefined output as specified by the requirements.**

4.3 Requirements tracing

AS





Lower-Level Requirements roll up into 6 business objectives

4.4 Prioritization : Analytical Criteria Method

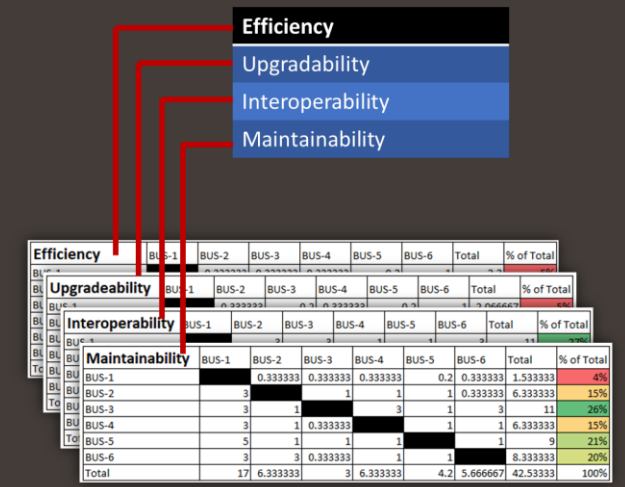
AS

Step 1: Ranking the Criteria

Accuracy	Recoverability	Installability	Efficiency	Maintainability	Scalability	Standards compliance
Debuggability	Reliability	Interoperability	Fidelity	Modularity	Simplicity	Survivability
Effectiveness	Reusability	Learnability	Usability	Testability	Upgradability	Tailorability

	Accuracy	Debuggability	Effectiveness	Efficiency	Fidelity	Installability	Interoperability	Learnability	Maintainability	Modularity	Recoverability	Reliability	Reusability	Scalability	Simplicity	Standards compliance	Survivability	Tailorability	Testability	Upgradability	Usability	Total	% of Total
Accuracy		3	1	3	3	5	1	3	1	3	1	1	3	3	3	3	3	3	1	1	3	48	8.007%
Debuggabili	0.33333		0.333333333	0.333333	0.3333	3	1	3	1	1	0.333333333	0.333333	1	1	3	3	1	3	1	1	1	26	4.337%
Effectivenes	1	3		3	3	5	1	3	1	3	1	1	3	3	3	3	3	3	1	1	3	48	8.007%
Efficiency	0.33333	3	0.333333333		1	3	1	3	1	3	1	1	3	3	3	3	3	3	1	1	3	40.66667	6.784%
Fidelity	0.33333	3	0.333333333	1		3	1	3	1	3	0.333333333	0.333333	3	1	3	3	3	3	0.3333333	1	3	36.66667	6.117%
Installability	0.2	0.333333333	0.2	0.333333	0.3333		0.333333333	0.333333	0.333333333	0.333333	0.2	0.2	0.333333	0.333333	0.333333	0.333333333	0.333333	1	0.3333333	0.333333333	0.33333	6.8	1.134%
Interoperabi	1	1	1	1	1	3		3	1	3	1	1	1	3	3	1	3	3	1	1	3	36	6.005%
Learnability	0.33333	0.333333333	0.333333333	0.333333	0.3333	3	0.333333333		0.333333333	0.333333	0.333333333	0.333333	1	0.333333	1	0.333333333	1	1	0.3333333	0.333333333	1	12.66667	2.113%
Maintainabi	1	1	1	1	1	3	1	3		3	1	1	3	3	3	3	3	3	1	1	3	40	6.673%
Modularity	0.33333	1	0.333333333	0.333333	0.3333	3	0.333333333	3	0.333333333		0.333333333	0.333333	1	1	1	1	1	1	0.3333333	0.333333333	1	17.33333	2.891%
Recoverabil	1	3	1	1	3	5	1	3	1	3		1	3	3	3	3	3	3	1	1	3	42	7.006%
Reliability	1	3	1	1	3	5	1	3	1	3	1		3	3	3	3	3	3	1	1	3	46	7.673%
Reusability	0.33333	1	0.333333333	0.333333	0.3333	3	1	1	0.333333333	1	1	0.333333		1	1	1	1	1	0.3333333	0.333333333	1	16.66667	2.780%
Scalability	0.33333	1	0.333333333	0.333333	1	3	0.333333333	3	0.333333333	1	0.333333333	0.333333	1		3	1	3	1	0.3333333	0.333333333	1	22	3.670%
Simplicity	0.33333	0.333333333	0.333333333	0.333333	0.3333	3	0.333333333	1	0.333333333	1	0.333333333	0.333333	1	0.333333		0.333333333	1	1	0.3333333	0.333333333	1	13.33333	2.224%
Standards co	0.33333	0.333333333	0.333333333	0.333333	0.3333	3	1	3	0.333333333	1	0.333333333	0.333333	1	1	3		3	3	0.3333333	0.333333333	1	23.33333	3.892%
Survivability	0.33333	1	0.333333333	0.333333	0.3333	3	0.333333333	1	0.333333333	1	0.333333333	0.333333	1	0.333333	1	0.333333333		1	0.3333333	0.333333333	1	14.66667	2.447%
Tailorability	0.33333	0.333333333	0.333333333	0.333333	0.3333	1	0.333333333	1	0.333333333	1	0.333333333	0.333333	1	1	1	0.333333333	1		0.3333333	0.333333333	1	12	2.002%
Testability	1	1	1	1	3	3	1	3	1	3	1	1	3	3	3	3	3	3		1	3	42	7.006%
Upgradabili	1	1	1	1	3	3	1	3	1	3	1	1	3	3	3	3	3	3	1		3	40	6.673%
Usability	0.33333	1	0.333333333	0.333333	0.3333	3	0.333333333	1	0.333333333	1	0.333333333	0.333333	1	1	1	0.333333333	1	1	0.3333333	0.333333333		15.33333	2.558%
Total	11.2	28.66666667	11.2	16.66667	23.333	66	14.66666667	47.3333333	13.33333333	38.66667	13.2	11.86667	35.33333	35.33333	45.33333	36.66666667	41.3333333	44	12.66667	13.33333333	39.3333	599.4667	100.000%

Step 2: Ranking Options by Criteria

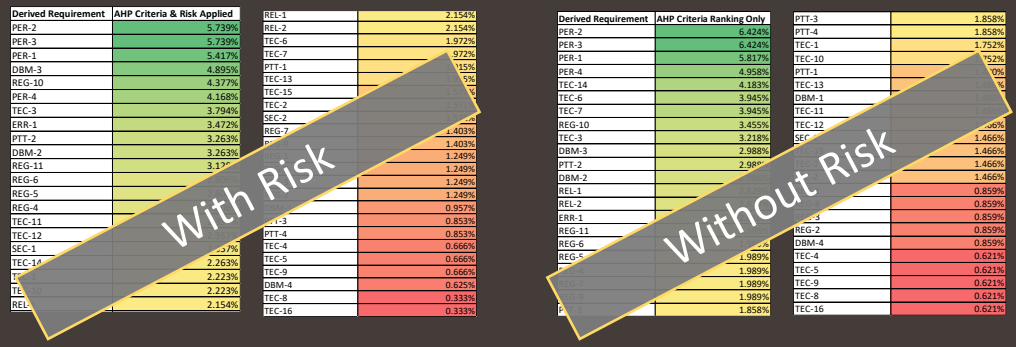


Maintainability	BUS-1	BUS-2	BUS-3	BUS-4	BUS-5	BUS-6	Total	% of Total
BUS-1		0.333333	0.333333	0.333333	0.2	0.333333	1.533333	4%
BUS-2	3		1	1	1	0.333333	6.333333	15%
BUS-3	3	1		3	1	3	11	26%
BUS-4	3	1	0.333333		1	1	6.333333	15%
BUS-5	5	1	1	1		1	9	21%
BUS-6	3	3	0.333333	1	1		8.333333	20%
Total	17	6.333333	3	6.333333	4.2	5.666667	42.53333	100%

Step 3: Ranking options by all Criteria

	Effectiveness	Accuracy	Reliability	Testability	Recoverability	Maintainability	Upgradeability	Interoperability	Efficiency	Total Across Col (%)
Architecture Resiliency	0.01	0	0.02	0.01	0	0	0	0.02	0	12%
Architecture Observability	0	0	0	0.01	0	0.01	0.01	0.01	0.01	9%
Data Limitations	0.02	0.03	0.01	0.02	0.02	0.02	0.02	0	0.01	22%
Network Capabilities	0.02	0.02	0.01	0.02	0.02	0.01	0.01	0.01	0.02	21%
System Interoperability	0.02	0.02	0.01	0.02	0.02	0.01	0.02	0.01	0.02	20%
Architecture Reliability	0.01	0.01	0.03	0.01	0.01	0.01	0	0.01	0.01	16%
Col Total	0.08	0.08	0.08	0.07	0.07	0.07	0.07	0.06	0.07	0.638

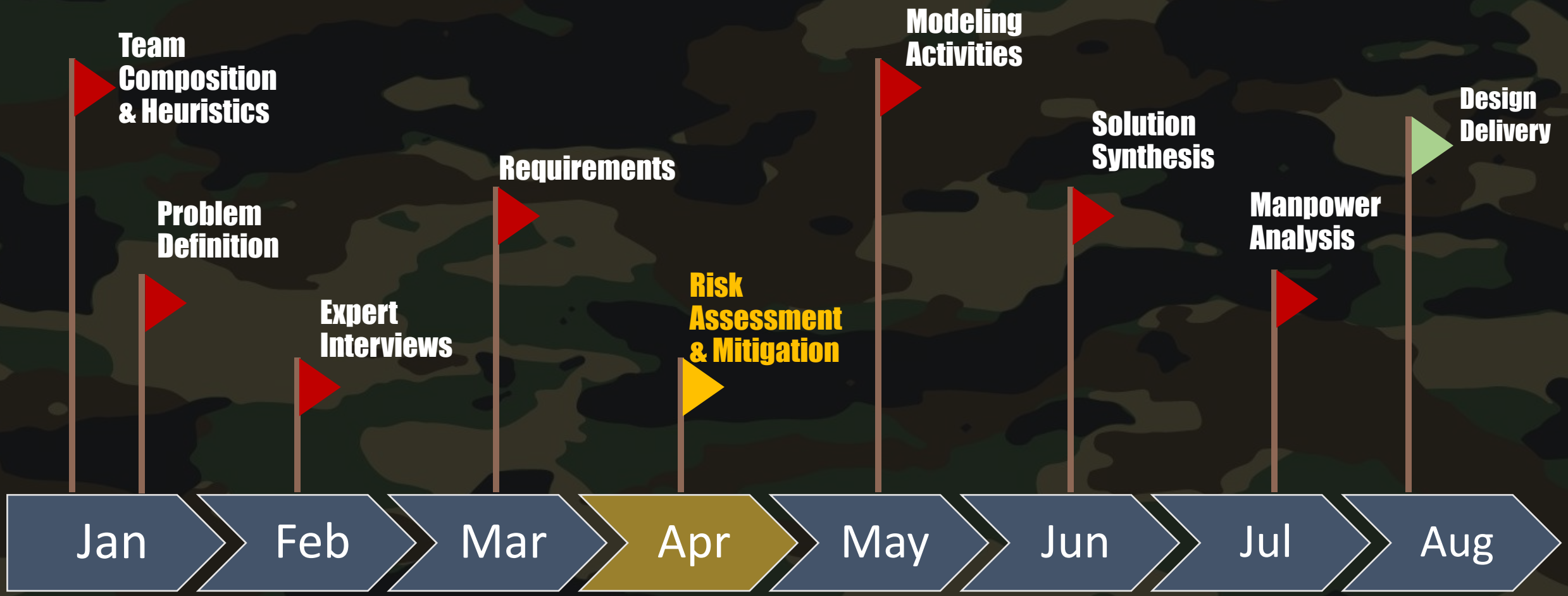
Step 4: Final Requirement Rankings



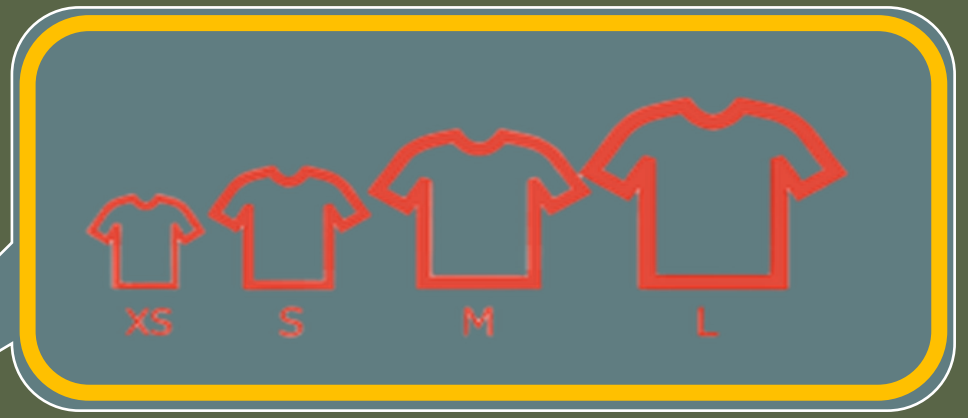
5. Risk Assessment & Mitigation

Georgia Tech Capstone

KD



5.1 T-Shirt Sizing

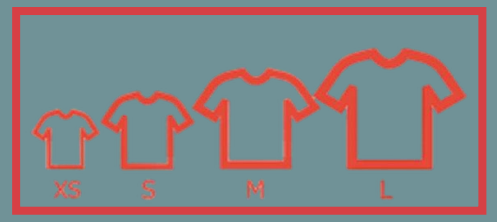


#	Name	Risk	Documentation	Text
1	DBM-1	Medium	Demonstrate by querying the system engineering database for the diagrams	The MTA Tool shall consolidate physical architecture diagrams in a stand-alone system engineering database
2	DBM-2	Medium	Test communication paths between two systems or nodes by sending data from one system to another and quickly returning same data to the sender.	The MTA Tool shall be capable of identifying data loops between systems
3	DBM-3	High	Test data loops between two systems or nodes by sending data from one system to another and verify if the data is being repeated or stops after its nullified.	The MTA Tool shall be capable of nullifying data loops between systems
4	DBM-4	Low	Test the system with C++ configuration script	The MTA Tool shall be compatible with C++ codex
5	ERR-1	High	Demonstrate the behavior of the MTA analysis tool to ensure that it conforms to functional requirements	The MTA Tool output shall produce error messages for "Format", "Drop Messages", "Non-conforming attribute" as part of its

KD

5.2 Risk Sizing

KD

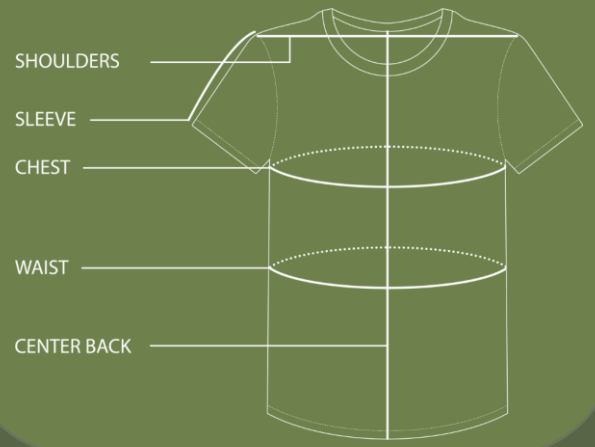


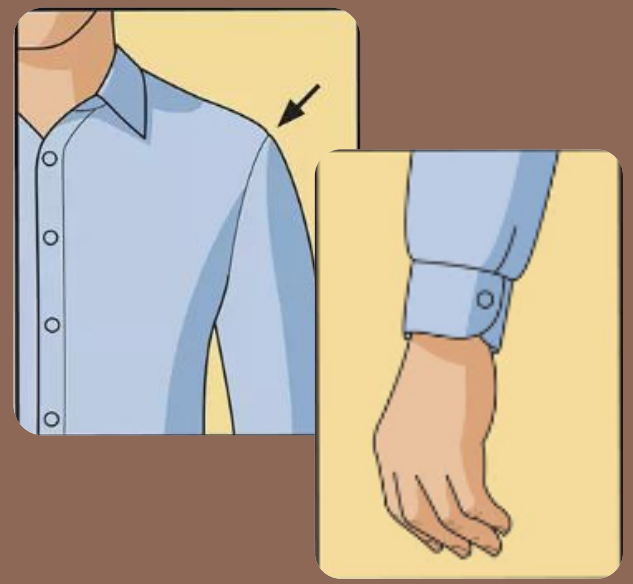
Initial Risk Assessment

NASA Risk Prioritization Matrix

Likelihood					
7	16	20	23	25	
6	13	18	22	24	
4	10	15	19	21	
2	8	11	14	17	
1	3	5	9	12	

Consequence

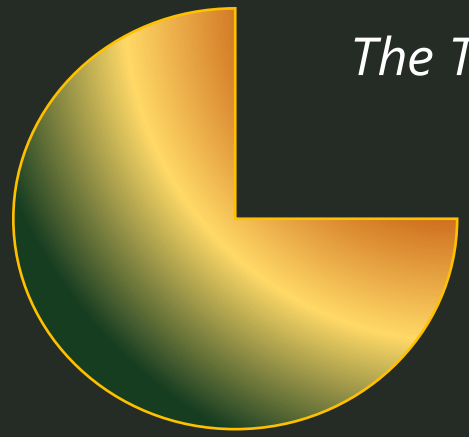




Revised Risk Assessment

5.3 High Risk Concern & Mitigation

KD

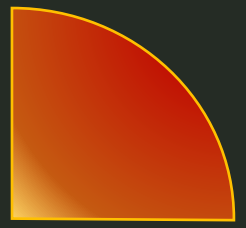


The Team maintains a top-level understanding of the outputs expected from the tool.

Technical analyses:

1) Analysis of Alternatives
2) Discrete system evaluation

Implementing the simulations and integrating them to lower risk



Biggest Risk encountered from scoring:

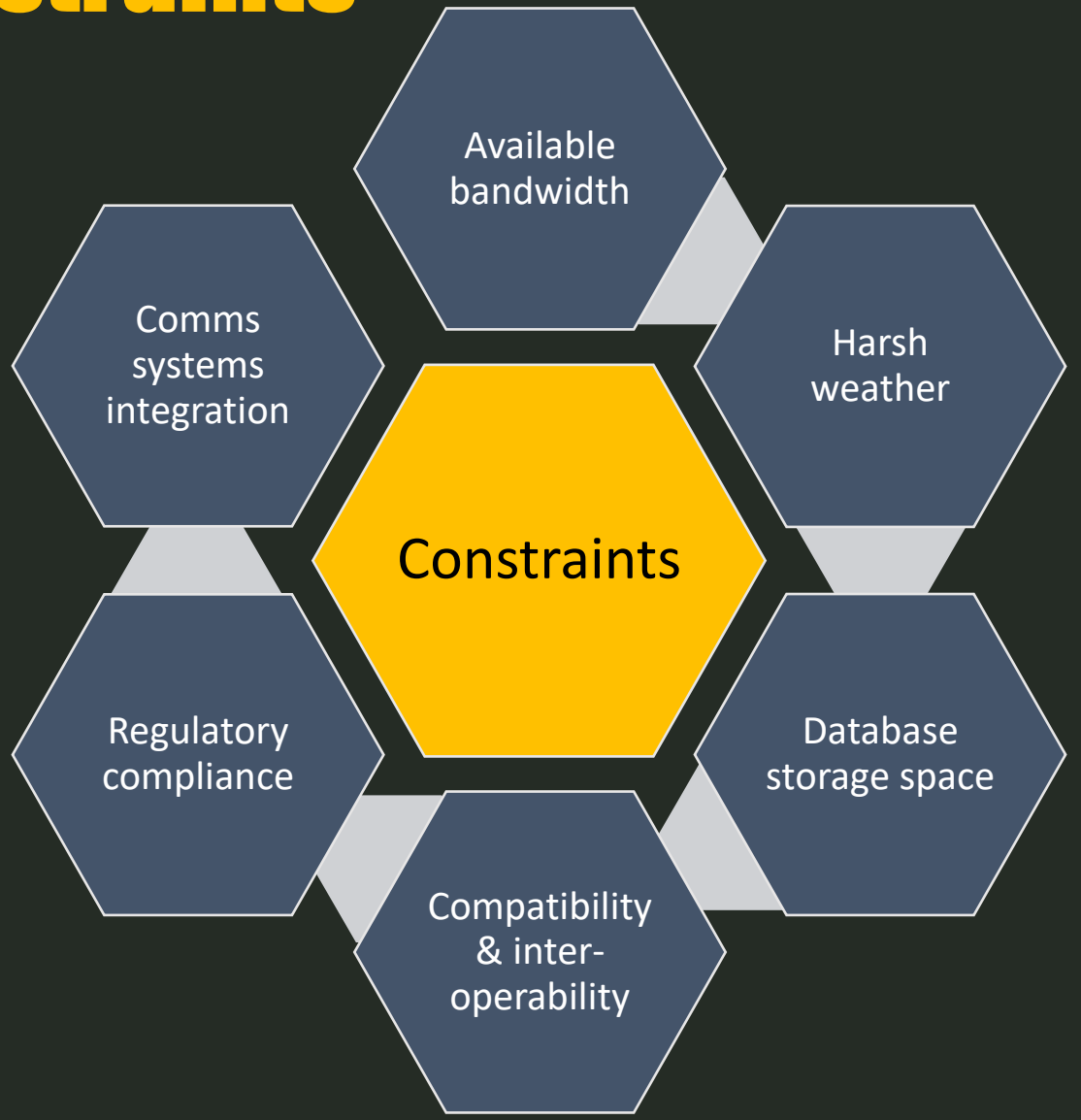
Method of **generating DoDAF views** was researched and clarified to develop the architecture and frame the approach

5.4 Assumptions & Constraints

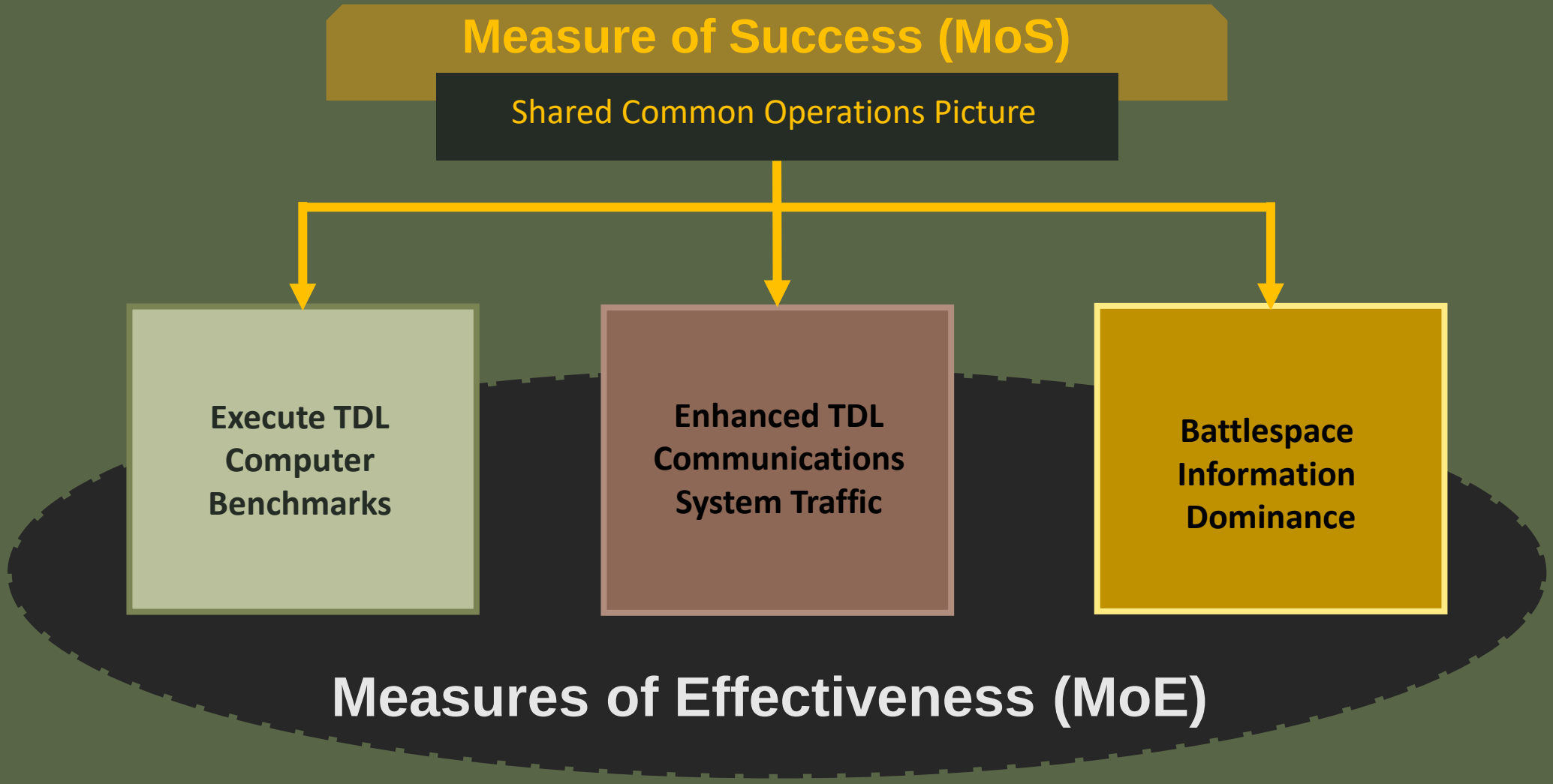
KD

- Treated as an MVP
- Flexible & adaptable
- Scalable
- Seamless synchronization
- All nodes will be identical
- Security will not be compromised
- Limited bandwidth
- Network congestion

Key Assumptions



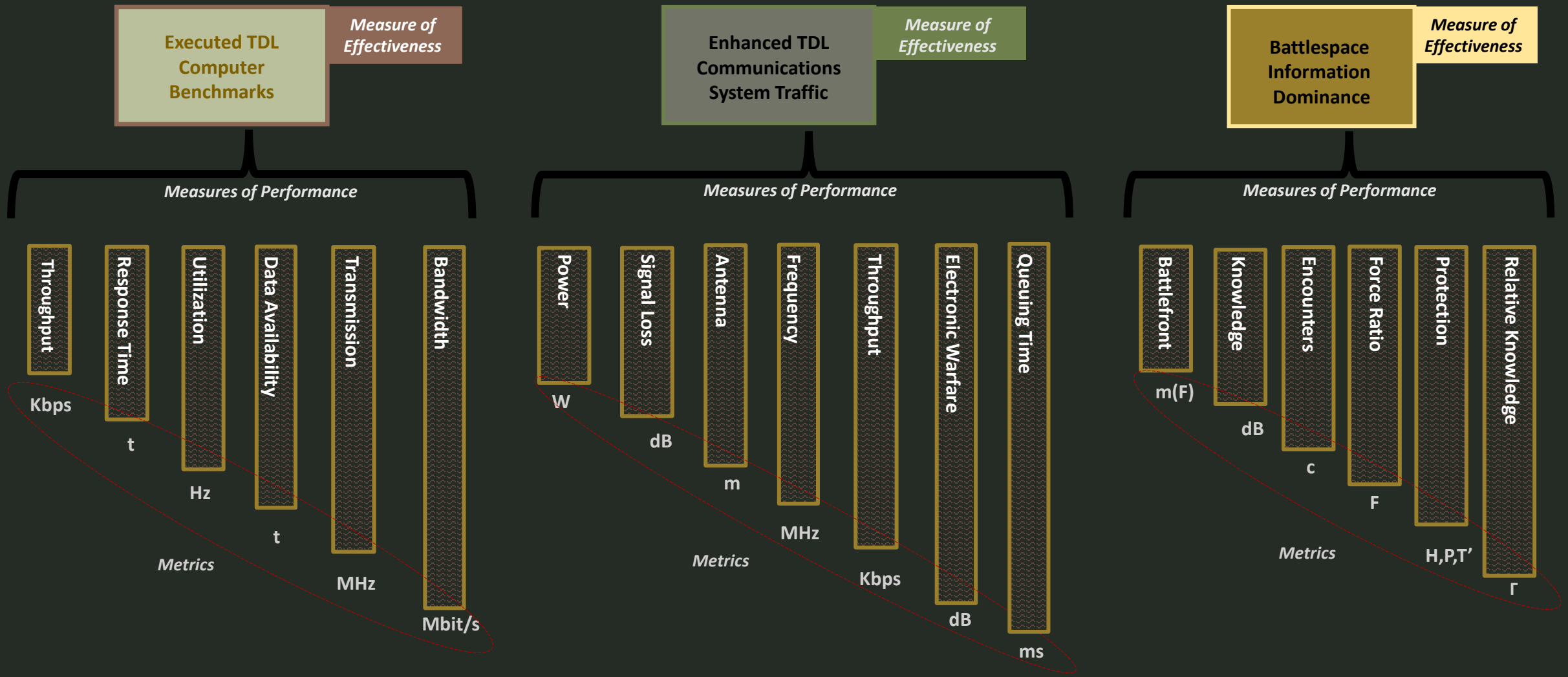
5.5.1 MoE & MoS Overview



5.5.2 Hardware MoE

FR

We combine these MOE metrics into a single “score” using z-score



5.6 Certification (V&V)

FR

Verification ensures an objective set of standards is met and that no bias can interfere with assessment.

Robustness & Reliability

System operates without crashes.
System properly handles invalid data.

Performance:

How many messages can the analysis tool successfully process in a specified time interval?
Are we able to operate in Real Time?

Accuracy:

Meets stated level of model fidelity.
Test result comparisons with real world data.
Verify all message content meets MIL Standards.

Security:

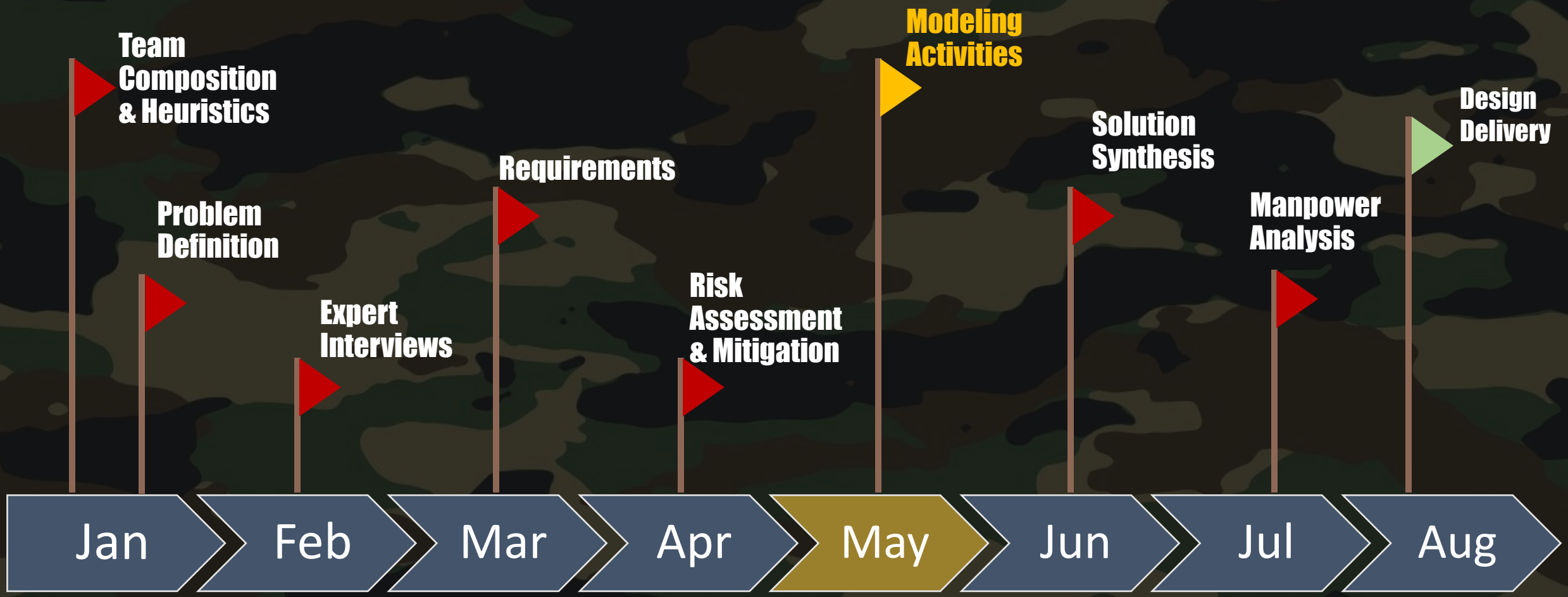
(May not be taking this into consideration initially).
System maintains security & confidentiality of data

Acceptable tests for certification of a Multi-Tactical Data Link system are identified by the client/sponsor.

The tests are conducted in an unbiased manner, so no conflict of interest between systems team and testing center.

6. Modeling Activities

Georgia Tech Capstone DC

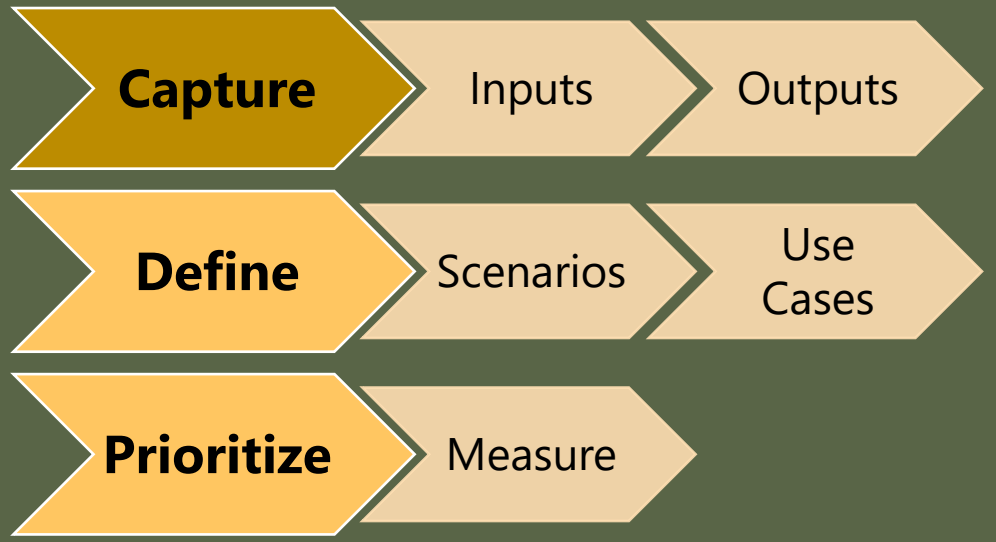


6.1 Capturing Behavior

DC

Attributes the artifact must produce as part of its behavior:

Health Status
Bandwidth Availability
Statistics for messages sent/received
Errors: Format, Drop Messages, Non-conforming, Periodicity, Latency



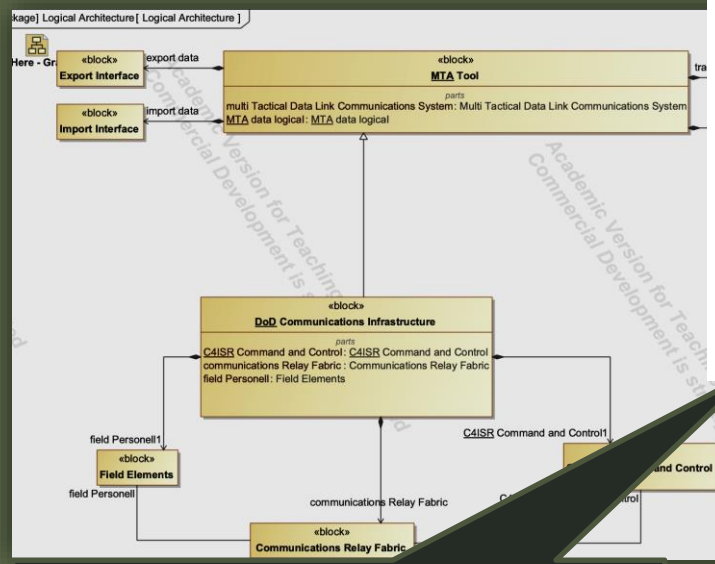
Standards the behavior must follow:

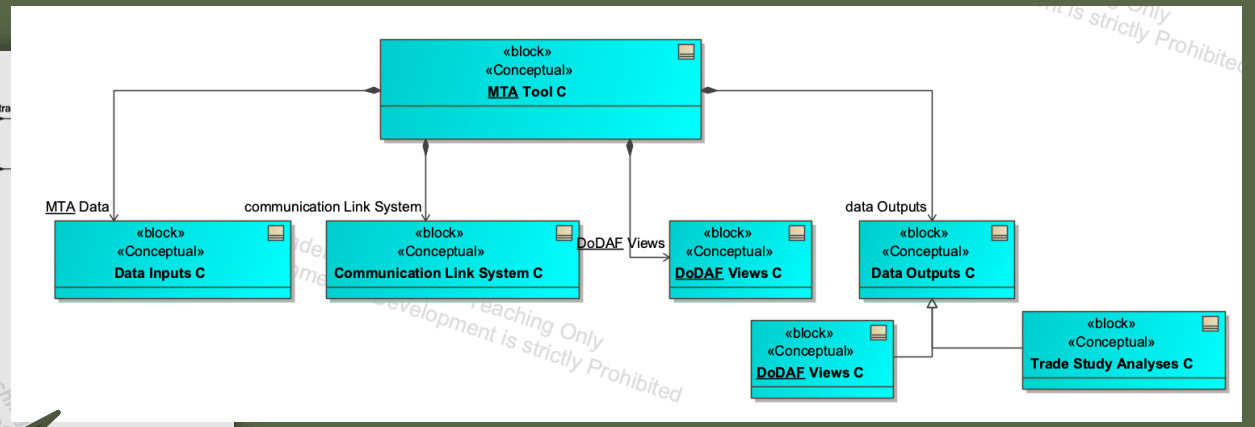
- UI/UX Standards for IPV6, RS232, R449, X.25, RJ19, 1553B, USB
- J-Series & F-Series protocols
- Regulatory requirements: IPERF and JPERF

Bandwidth availability is the most important attribute for scalability as noted by the client team.

Next step here is to develop data and event flow networks to measure transfer and message bandwidth consumption from events.

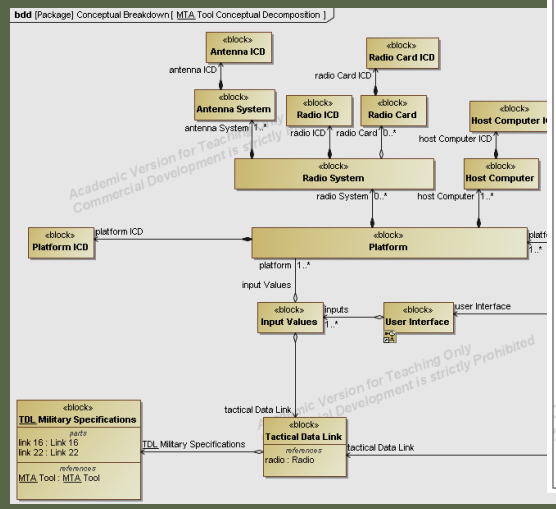
6.2 Conceptual – Logical Architecture

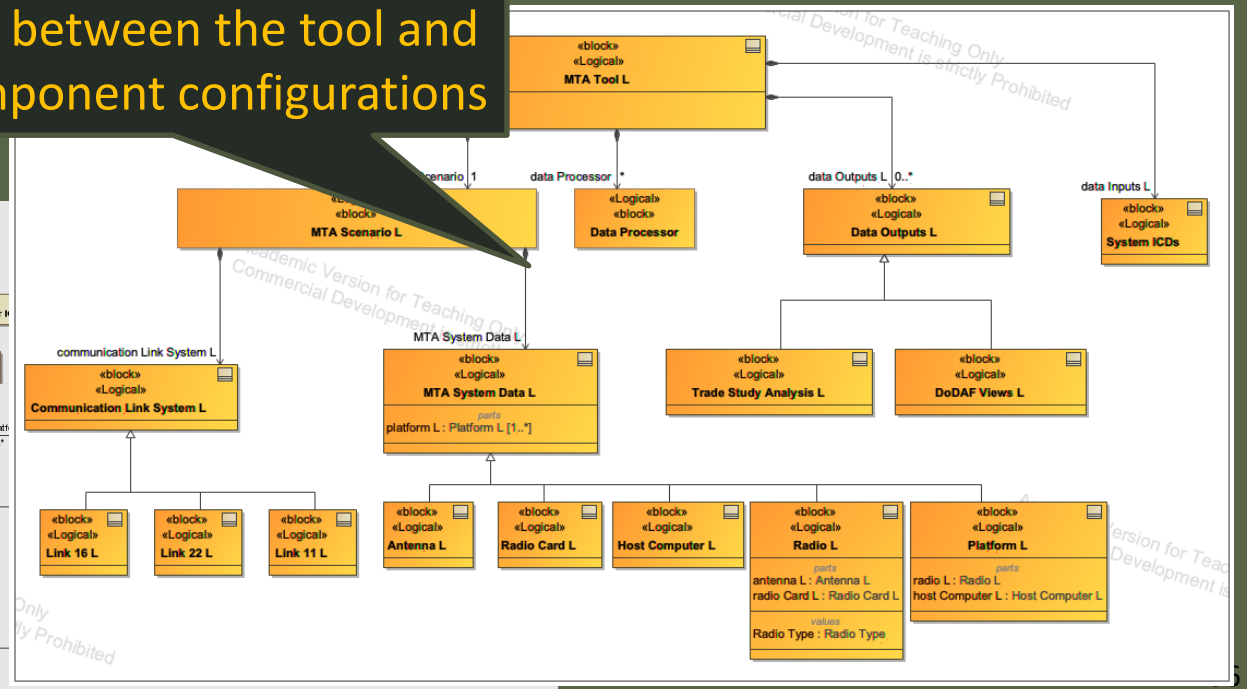




Iteration of the conceptual architecture to display the interconnections between the high-level components of the MTA tool domain

Decomposition of the logical architecture between the tool and external component configurations





6.3 Defining behavior

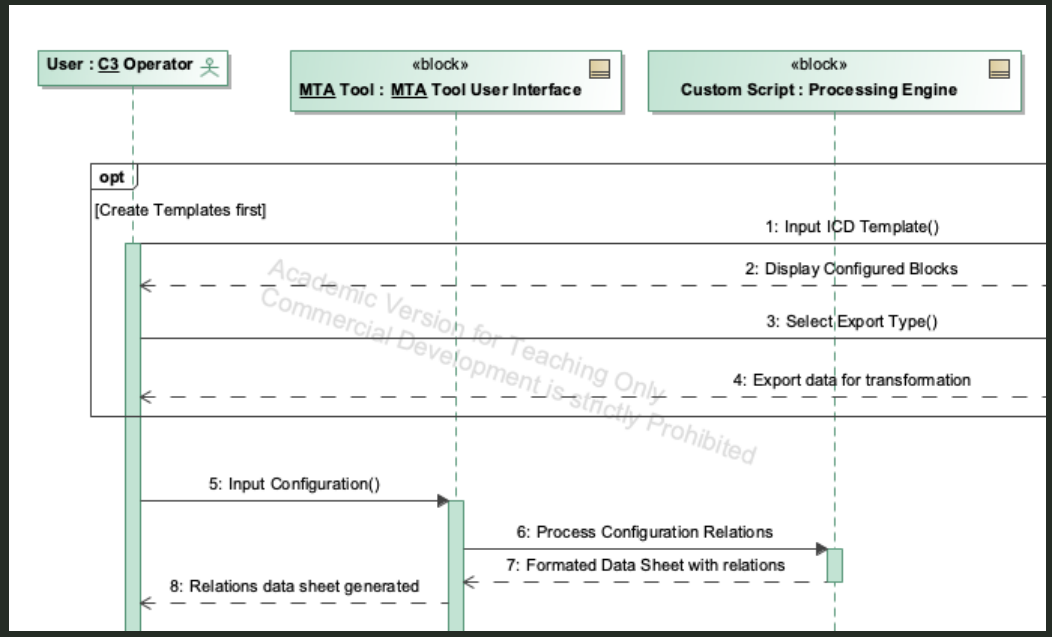
Georgia Tech Capstone DC

Activities and Ownership

- Whom or what system is responsible for each activity
- What information is passed between these activities

Sequencing

- Interactions between systems and processes
- Interaction between the operator and the Tool

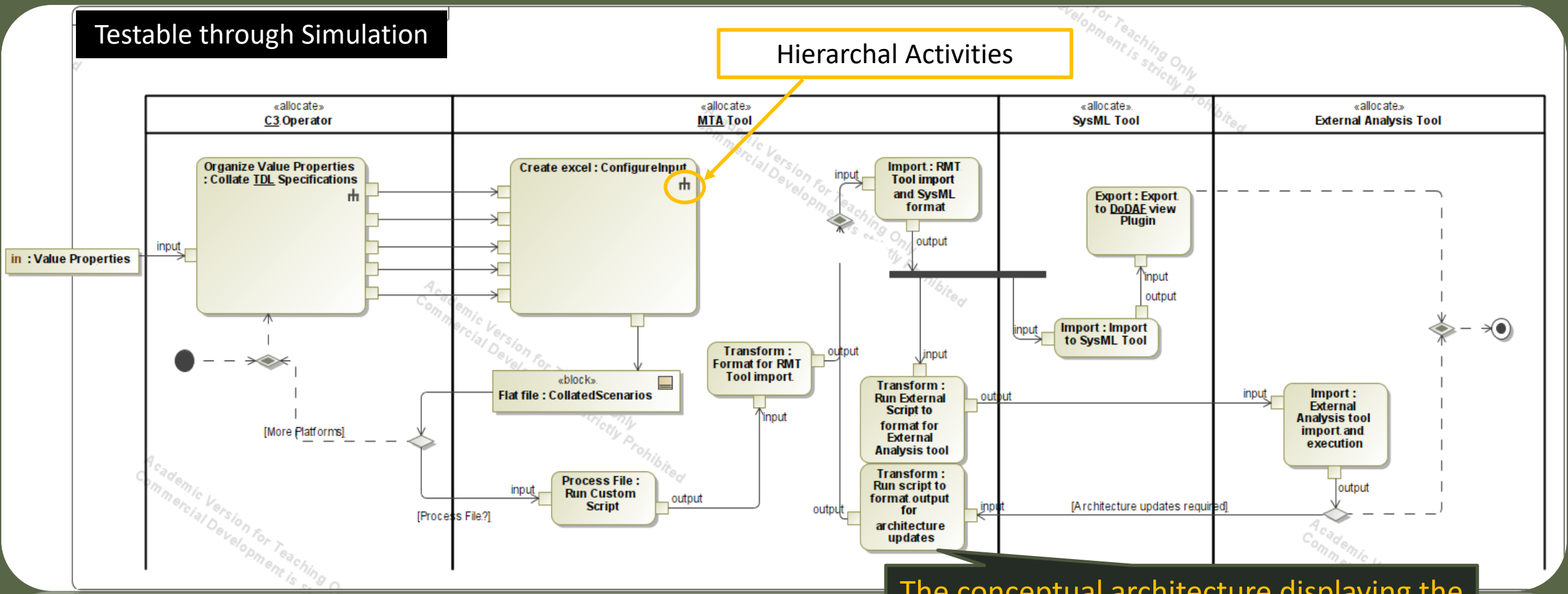


(Information Flow) →

Owner A Step 1 → Owner B Step 2 → Owner C Step 3 → Owner A Step 4 → Output

6.4 End-End Activities

Testable through Simulation



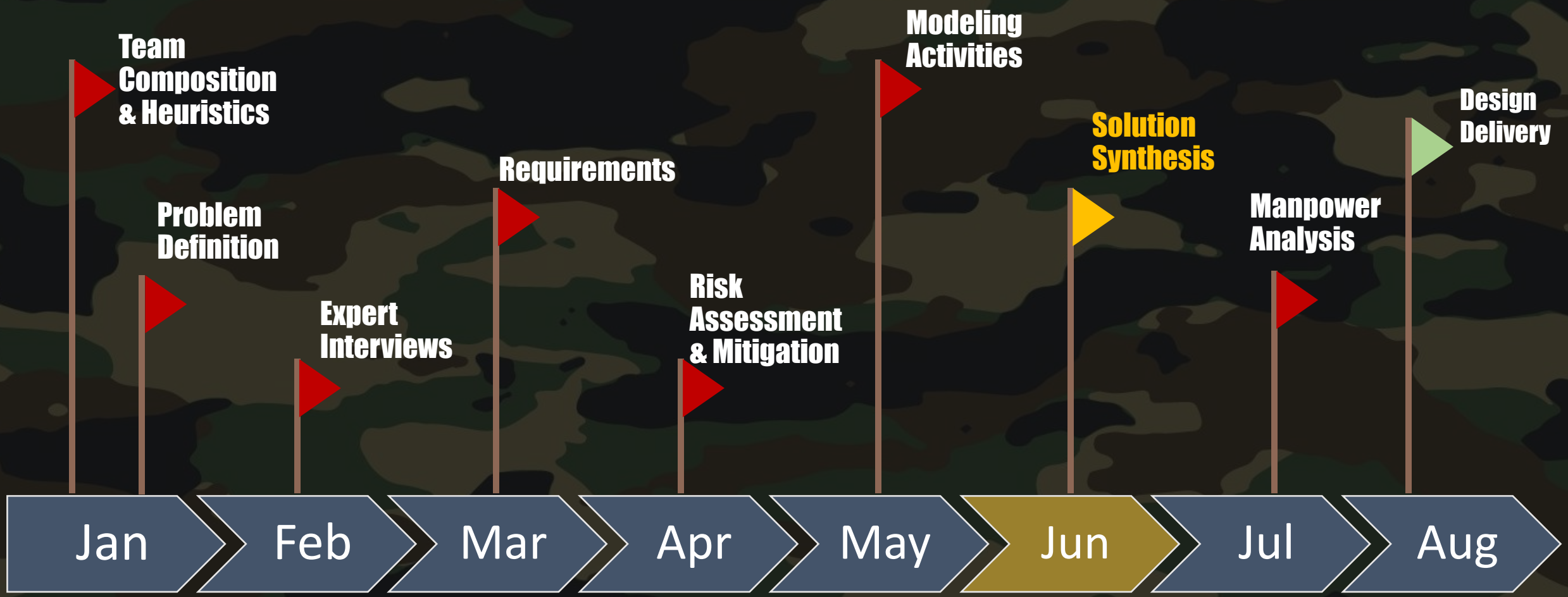
The conceptual architecture displaying the interconnections between the high-level components of the MTA tool domain

DC

7. Solution Synthesis

Georgia Tech Capstone

AS



7.1.1 Market Survey | DoDAF Viewpoints

Georgia Tech Capstone

AS

 Python with Plug-ins / Macros	 PURE Connector	 MagicDraw OPEN / JAVA API	 Artificial Intelligence
- Adds new components to a host program	- A description of the transformation in XML is generated	- Synthesizes DoDAF Views to directly work with the model data	- Use Generative AI and machine learning algorithms to create new images
- Extends its capabilities beyond its original design.	- XML is then used for the creation process.	- Leverages the API functions instead of exporting and parsing the XMI file.	- Based on a set of input parameters or conditions.

7.1.2 Sponsor decisions

Georgia Tech Capstone

AS

Alternatives:

- Python with Plug-ins / Macros
- PURE Connector
- MagicDraw OPEN/JAVA API
- Artificial Intelligence

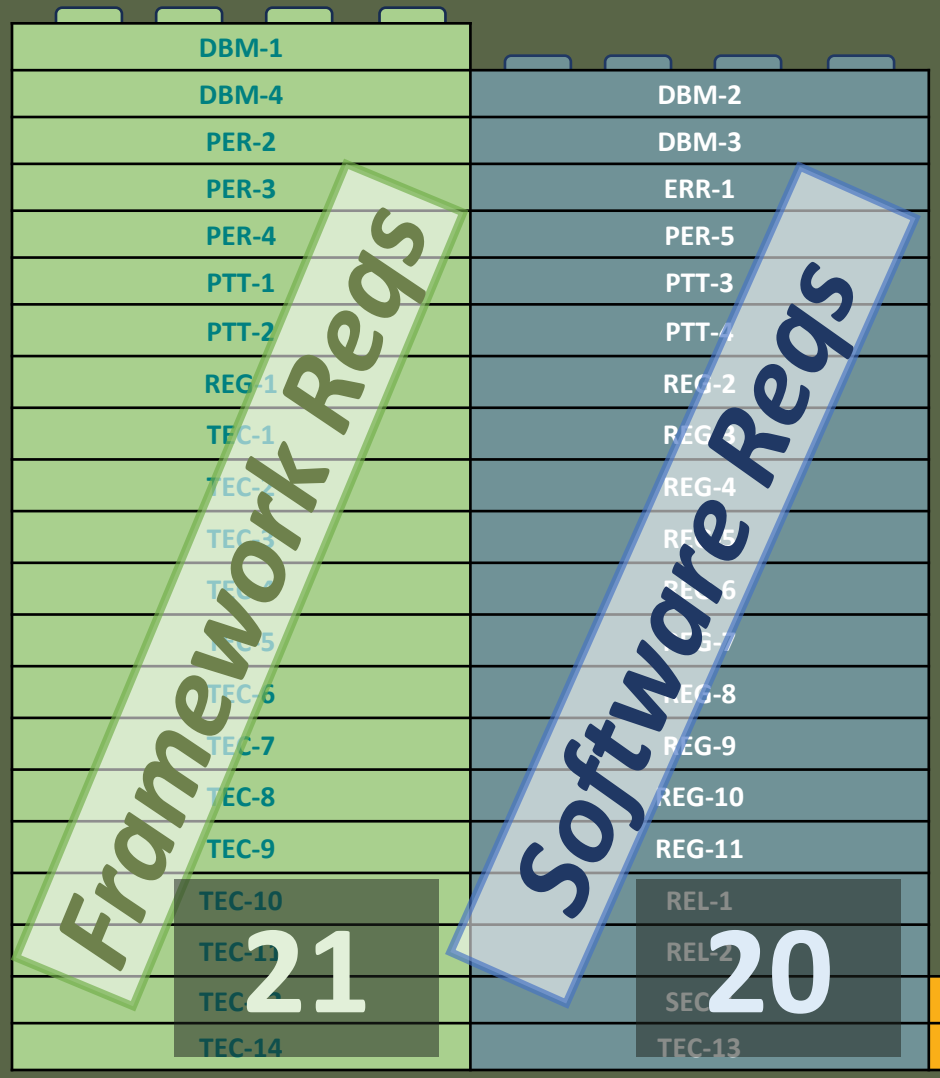
Discussion of risks and opportunities for each option

Python with Plug-ins / Macros

Flexibility & Maintainability

7.2.1 Final Requirements Assessment

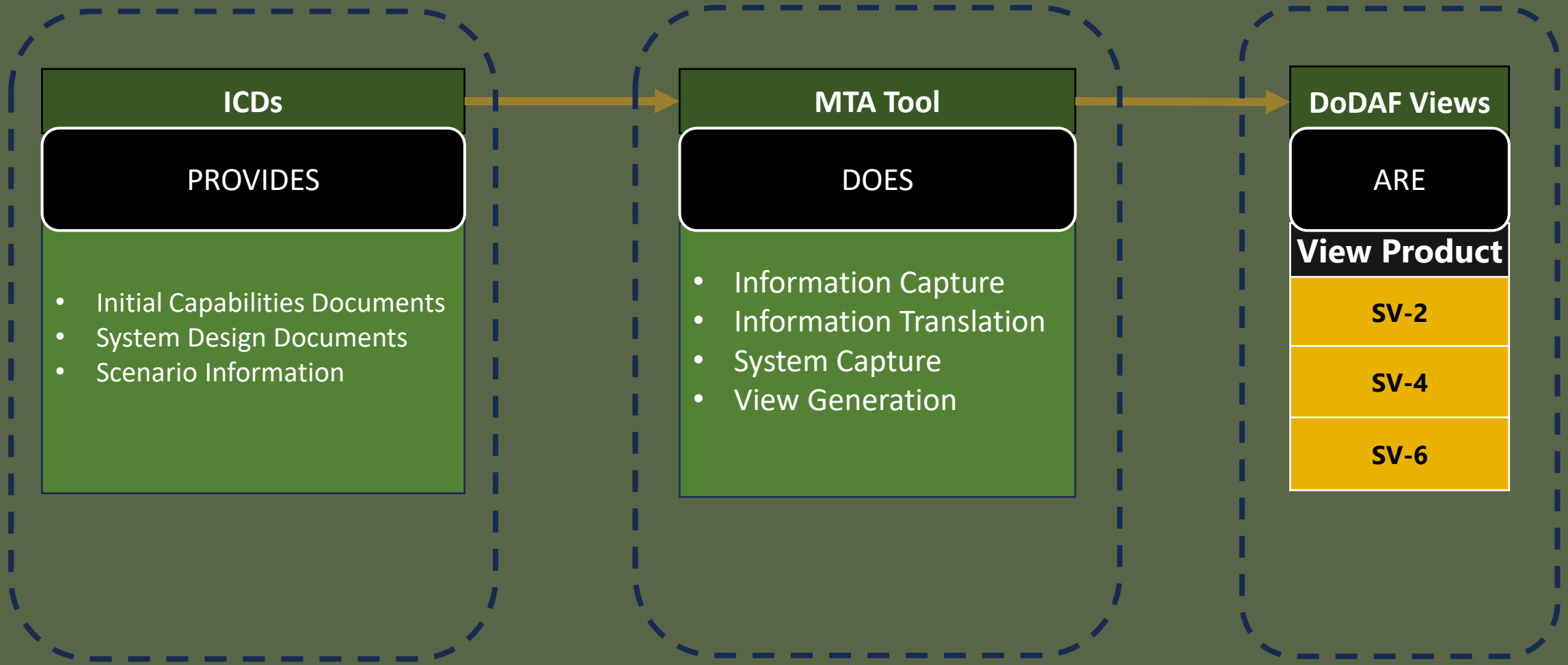
RC



21	MTA Analysis Tool Framework requirements met
20	MTA Analysis Tool Software Design requirements
2	MTA Analysis Tool Software Design requirements under further consideration
1	Out of Scope – Geo data
1	Additional requirement captured

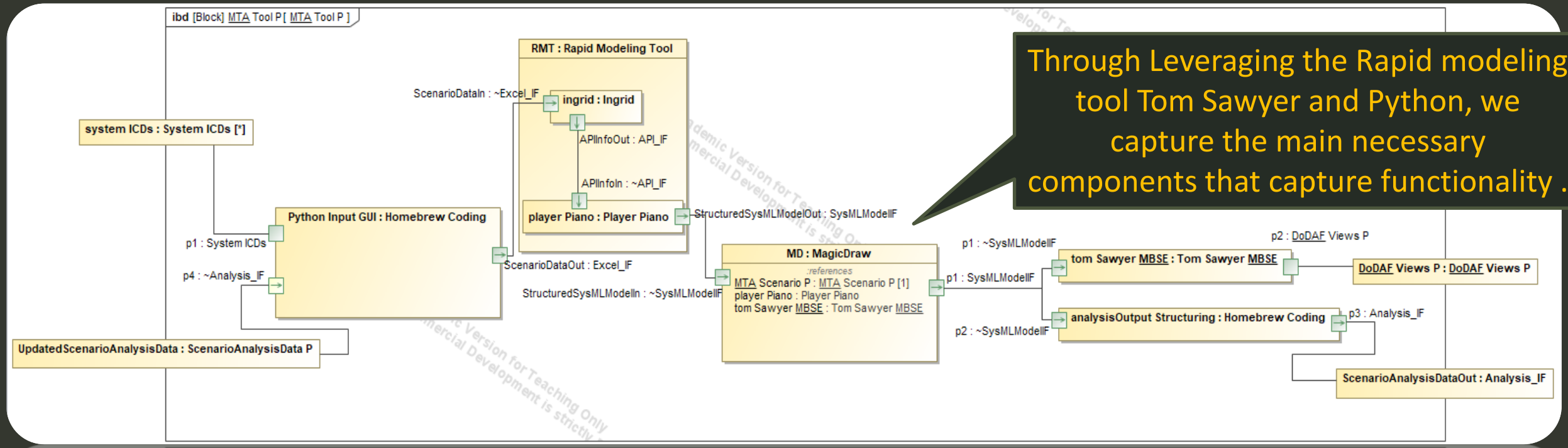
7.2.2 DoDAF Viewpoint Solution Synthesis Overview

RC



7.2.3 MTA Tool Internal Block Diagram

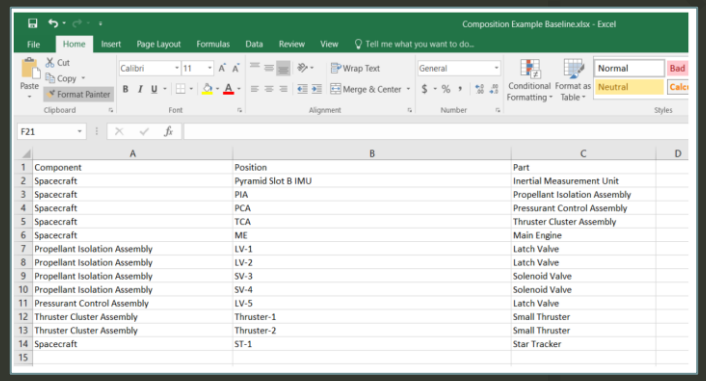
RC

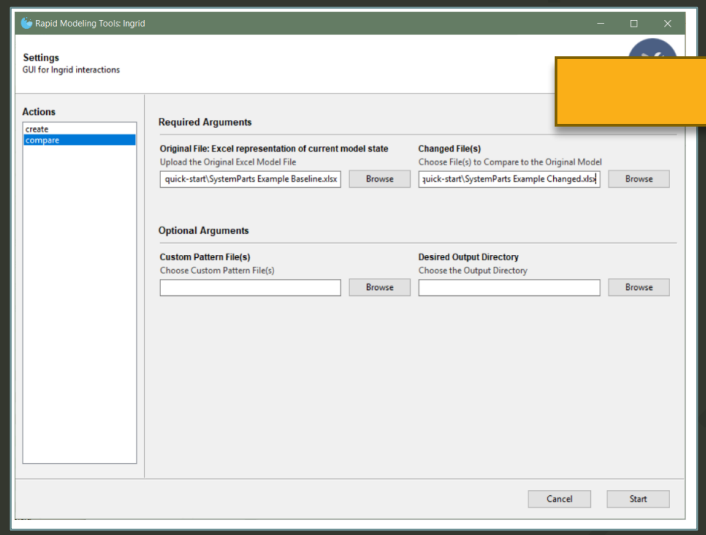


7.2.4 Rapid Modeling Tool

1

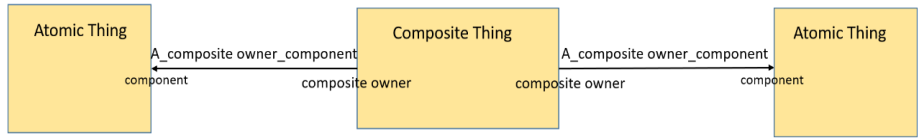
INGRID



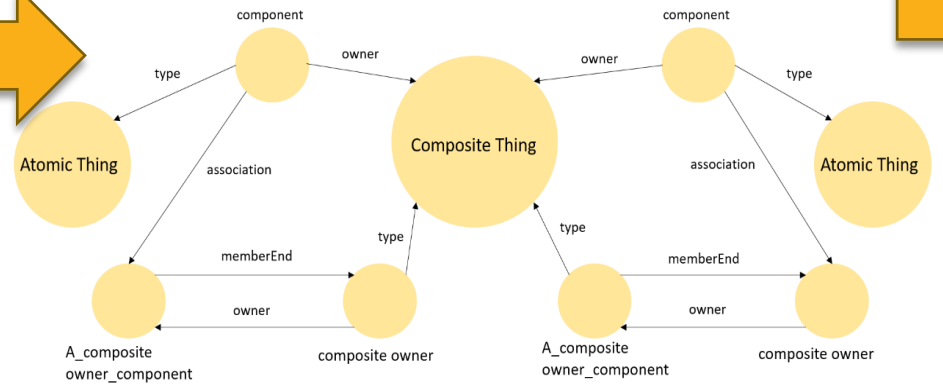


2

SysML Representation

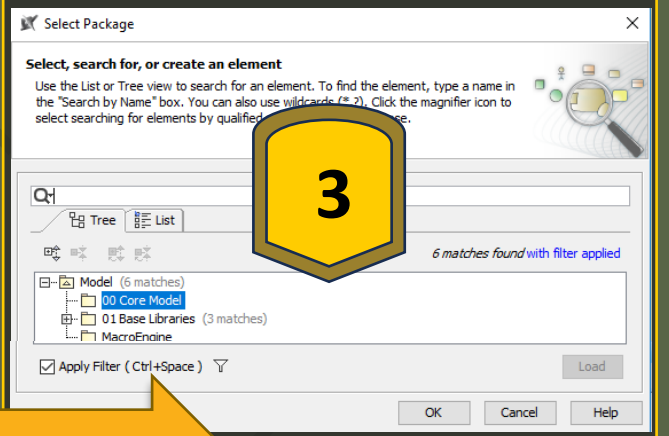


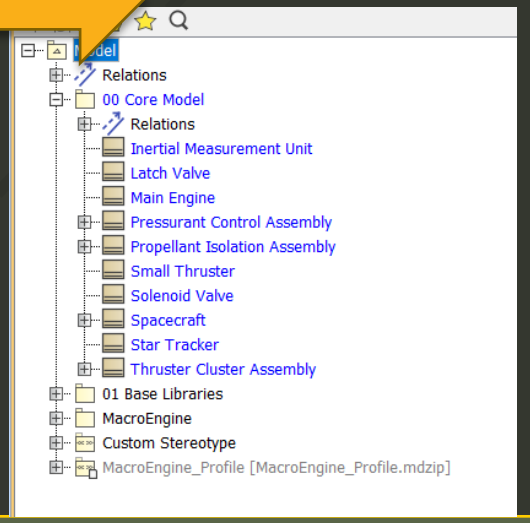
Pattern Representation



PLAYER PIANO

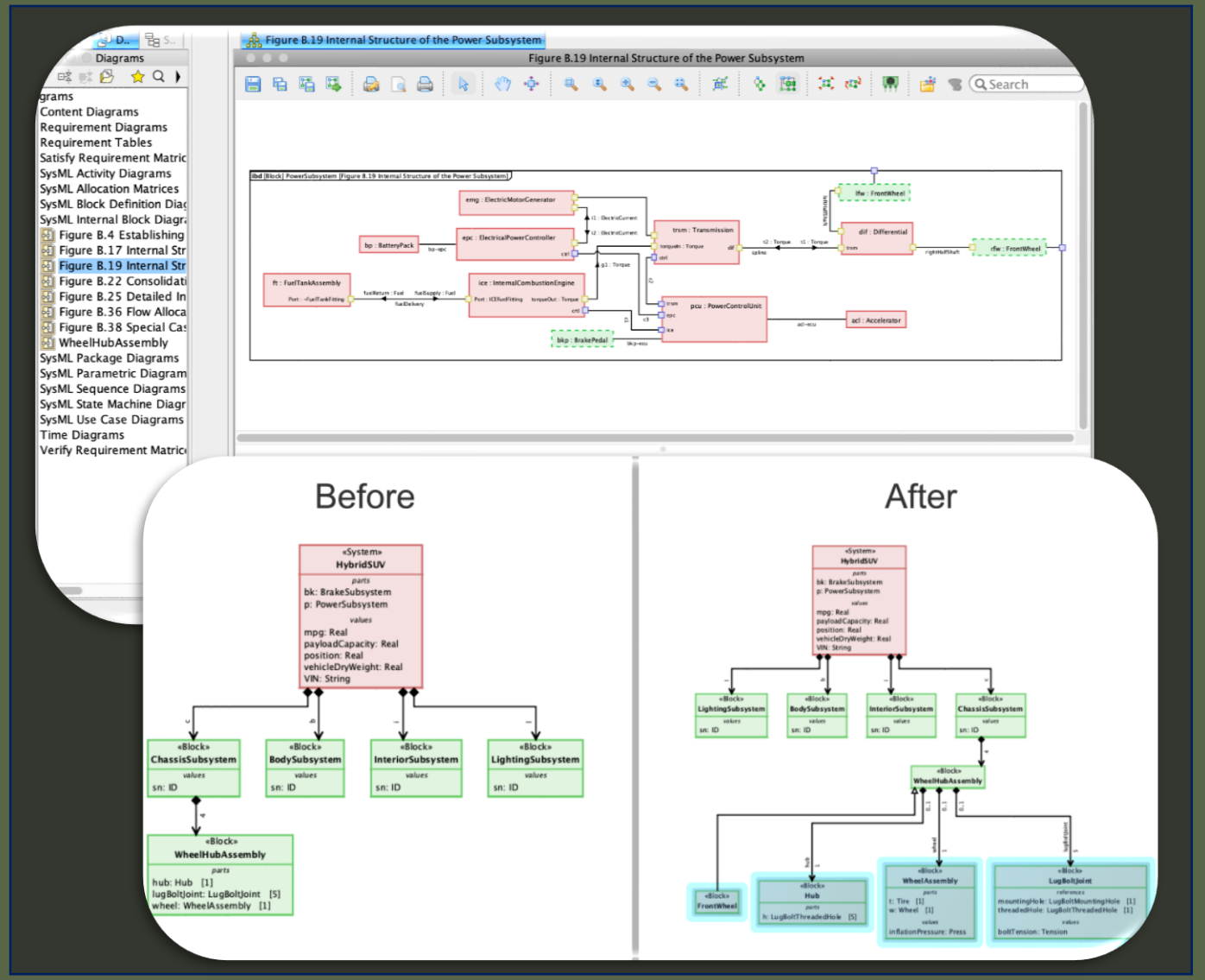
3





7.2.5 Tom Sawyer

- Plugin that allows for automatic laying out and rendering of multiple SysML diagram types.
- Tom Sawyer Perspectives MBSE Tool provides the end user with interactive features such as

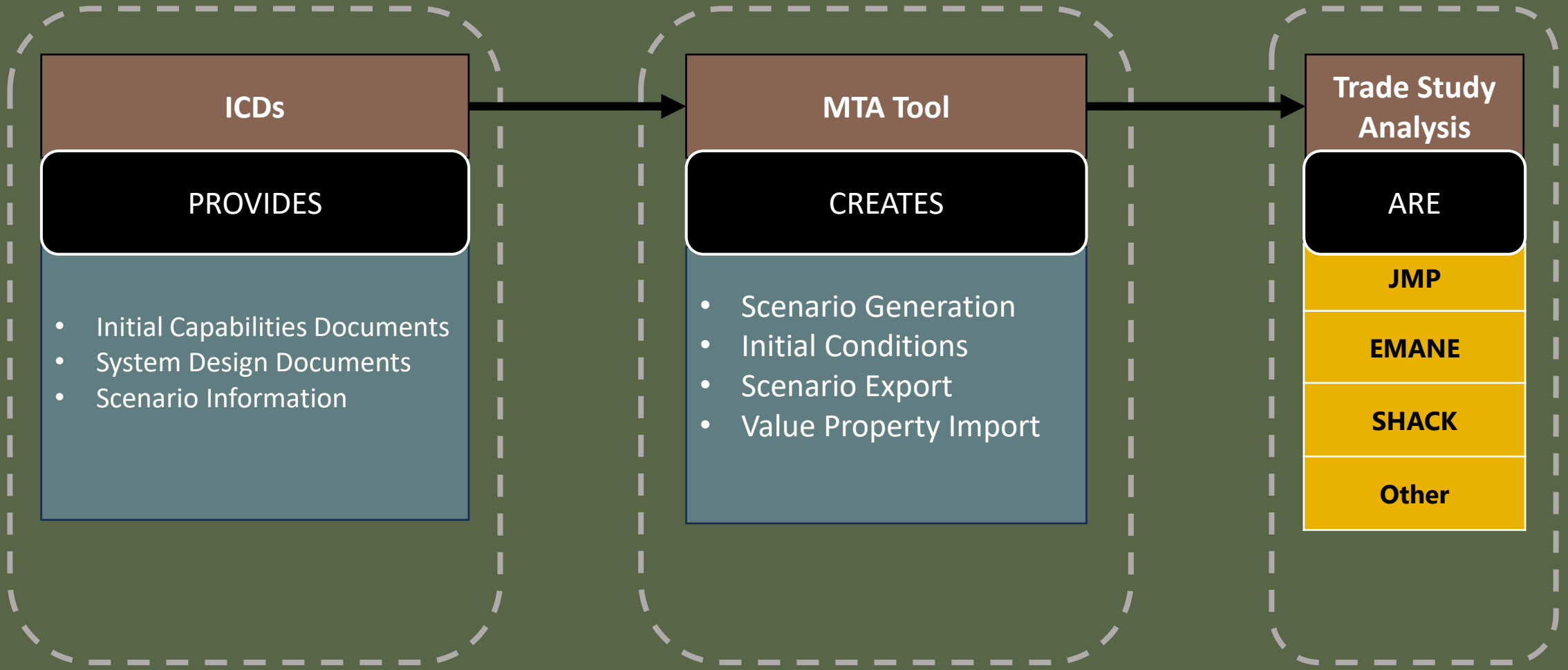


- Zoom
- Scroll
- Search
- Drill-down
- Drill-up
- Expand
- Collapse

- Automatically generate SysML diagrams
- Ability to load neighbors
- Element centric diagrams
- Ability to filter model elements

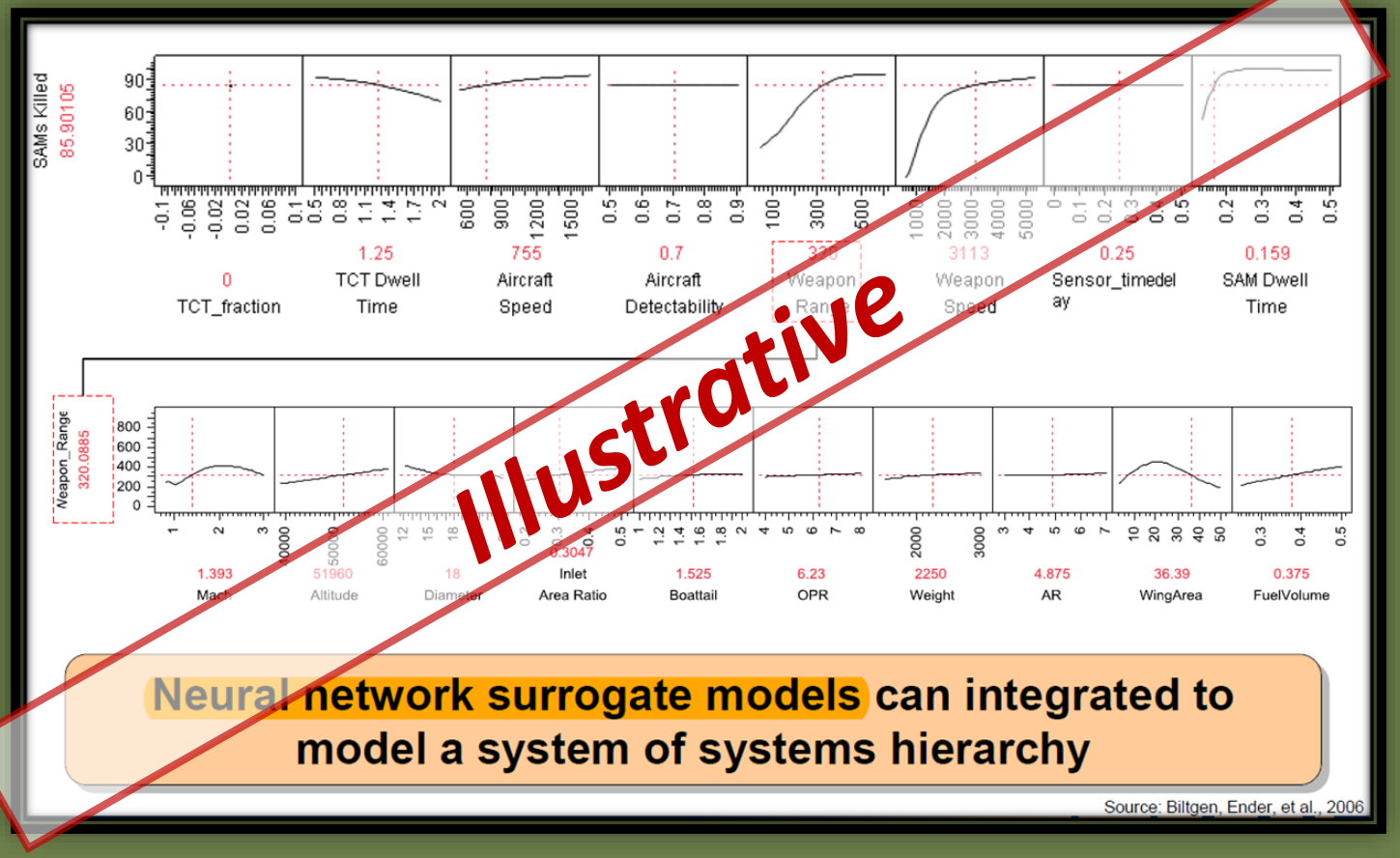
7.2.6 MTA Analysis Solution Synthesis Overview

RC



7.3.1 Surrogate Models & Neural Networks

MW



MTA Analysis Tool Metrics

- General
- Computer
- Radio
- Picture

- X-Y Variables in JMP

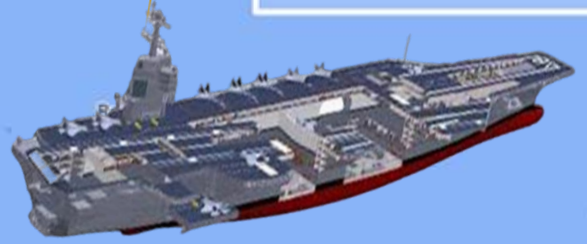
Dr. Ender's Top-down, Hierarchical System-of-Systems Design of Air Defense Weapon Thesis, 2006

7.3.2 Scenario

AWACS

STRYKER

SATCOM



CARRIER

Hawaii

HQ INDO PACOM

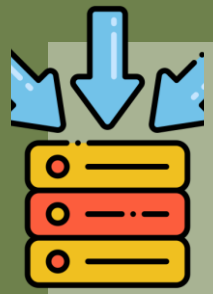
North Pacific Ocean

Midway Atoll

Bikenibeu

7.3.3 How do the TDL Assets Communicate with each other ?

MW

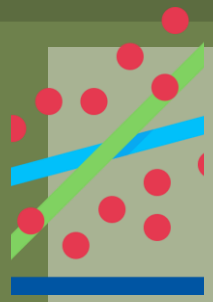


1. Inputs Parameters

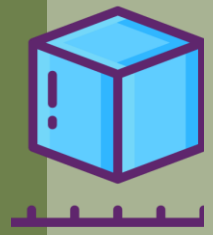
2. Output Parameters

3. Calculate Output Metrics

4. DoE via Model Center



5. Calculate responses for each DoE run

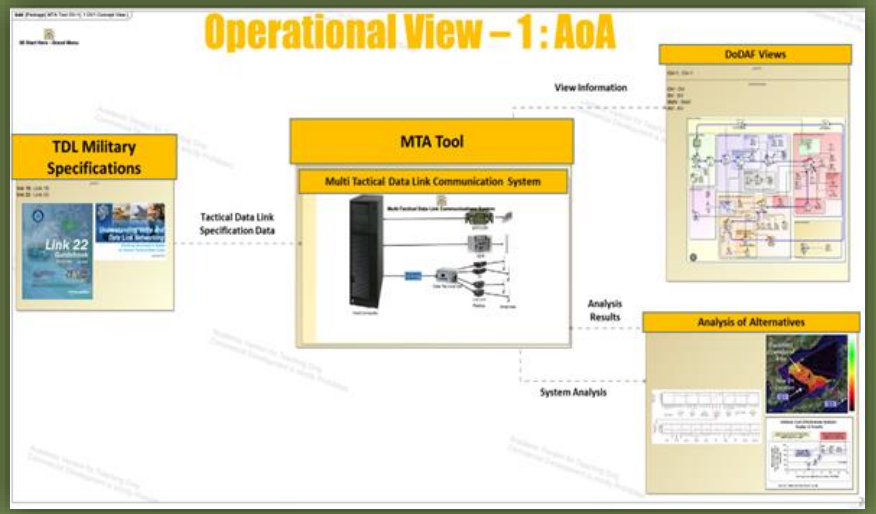


6. Generate JMP Regression model

New Throughput Equation

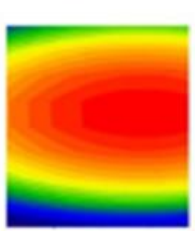
7.3.4 Throughput Regression Model Fit

MW

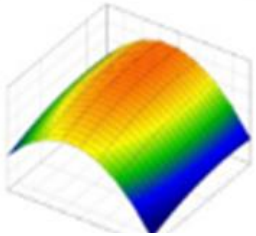


Response Surface Methodology

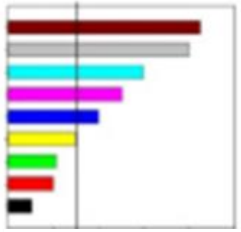
$$y_j = \beta_0 + \sum_{i=1} \beta_i x_i + \sum_{i=1} \beta_{ii} x_i^2 + \sum \sum_{i<j=1} \beta_{ij} x_i x_j$$



Contour Plot

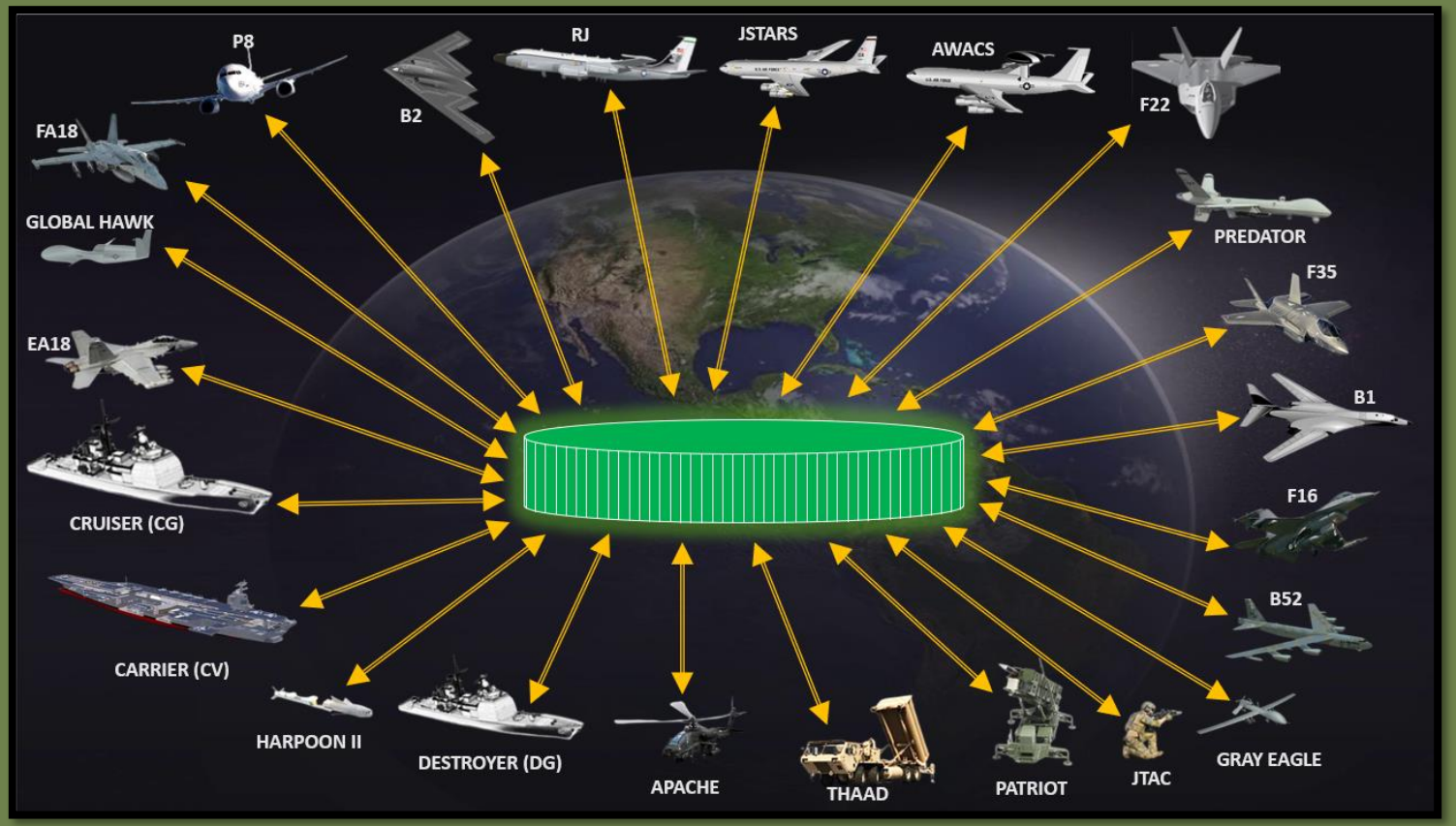


Response Surface Plot



Pareto Chart

Analysis of Variance to check the fitness of model

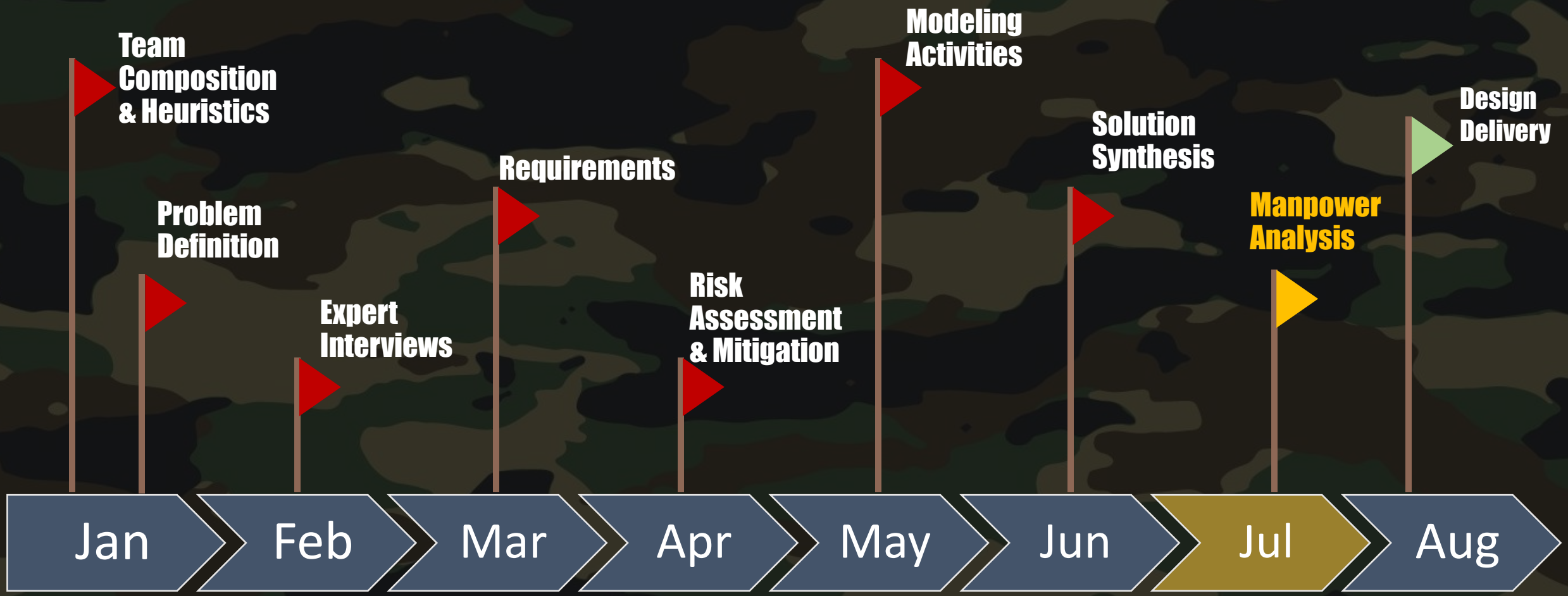


- Define Independent Variables and Output Responses
- Selection of the proper design
- Running and measure of the response of each experiment
- Verification of Model

8. Manpower Analysis

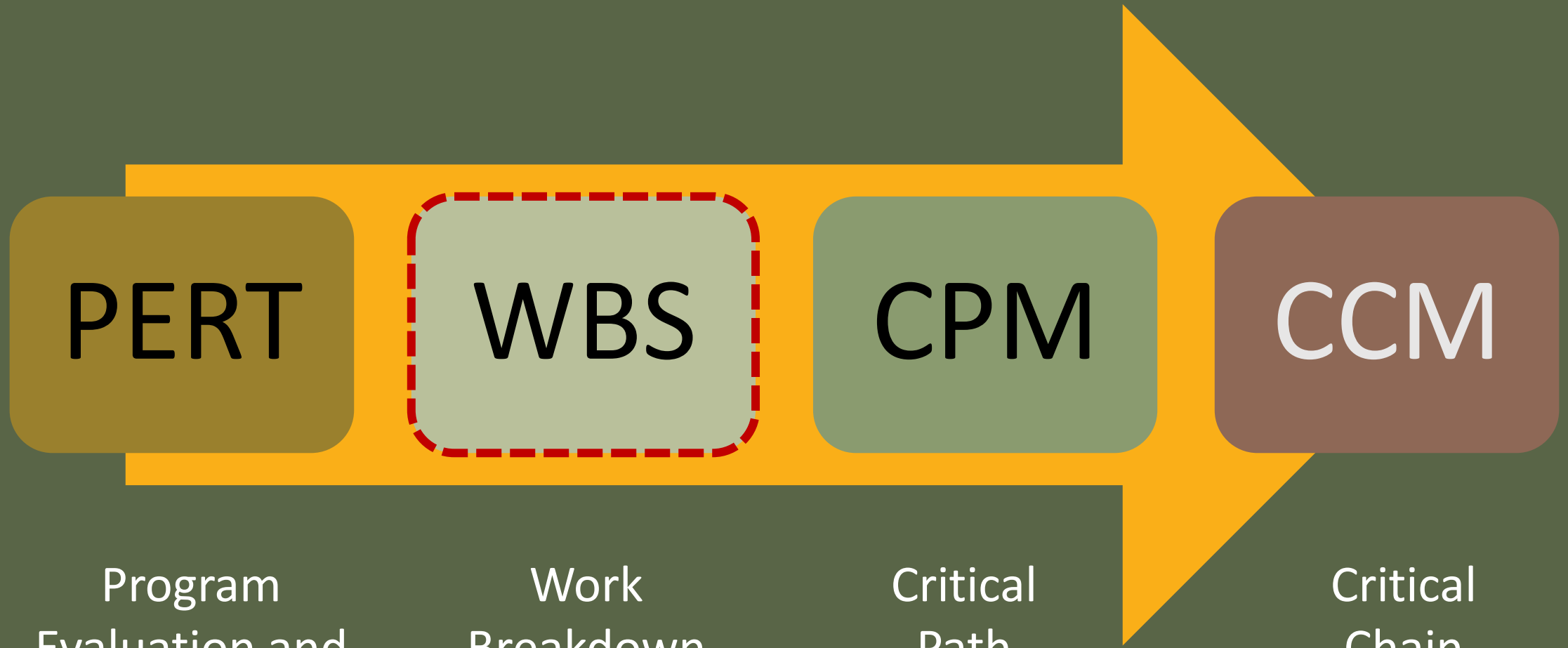
Georgia Tech Capstone

FR



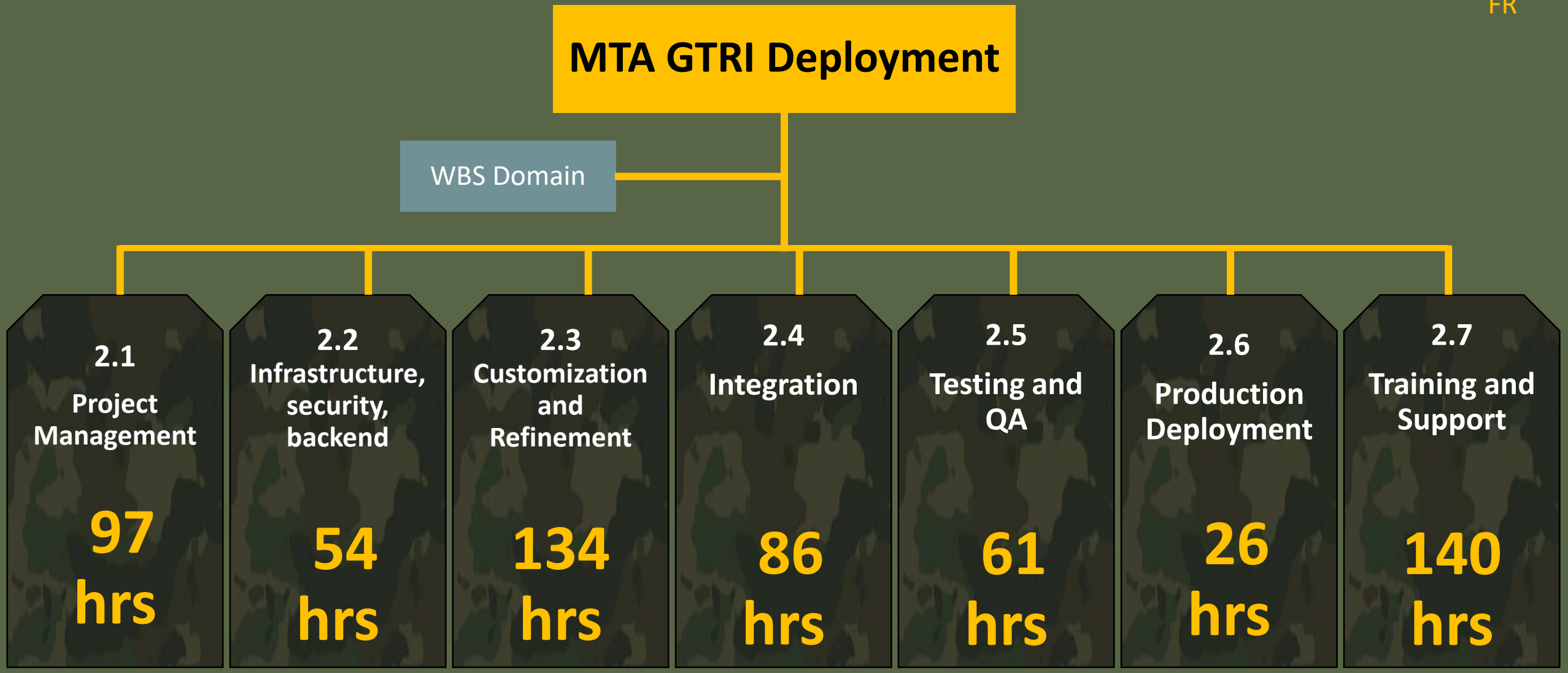
9.1 Manpower Analysis – Subsequent Build

FR



9.2 WBS – GTRI Software Build

FR



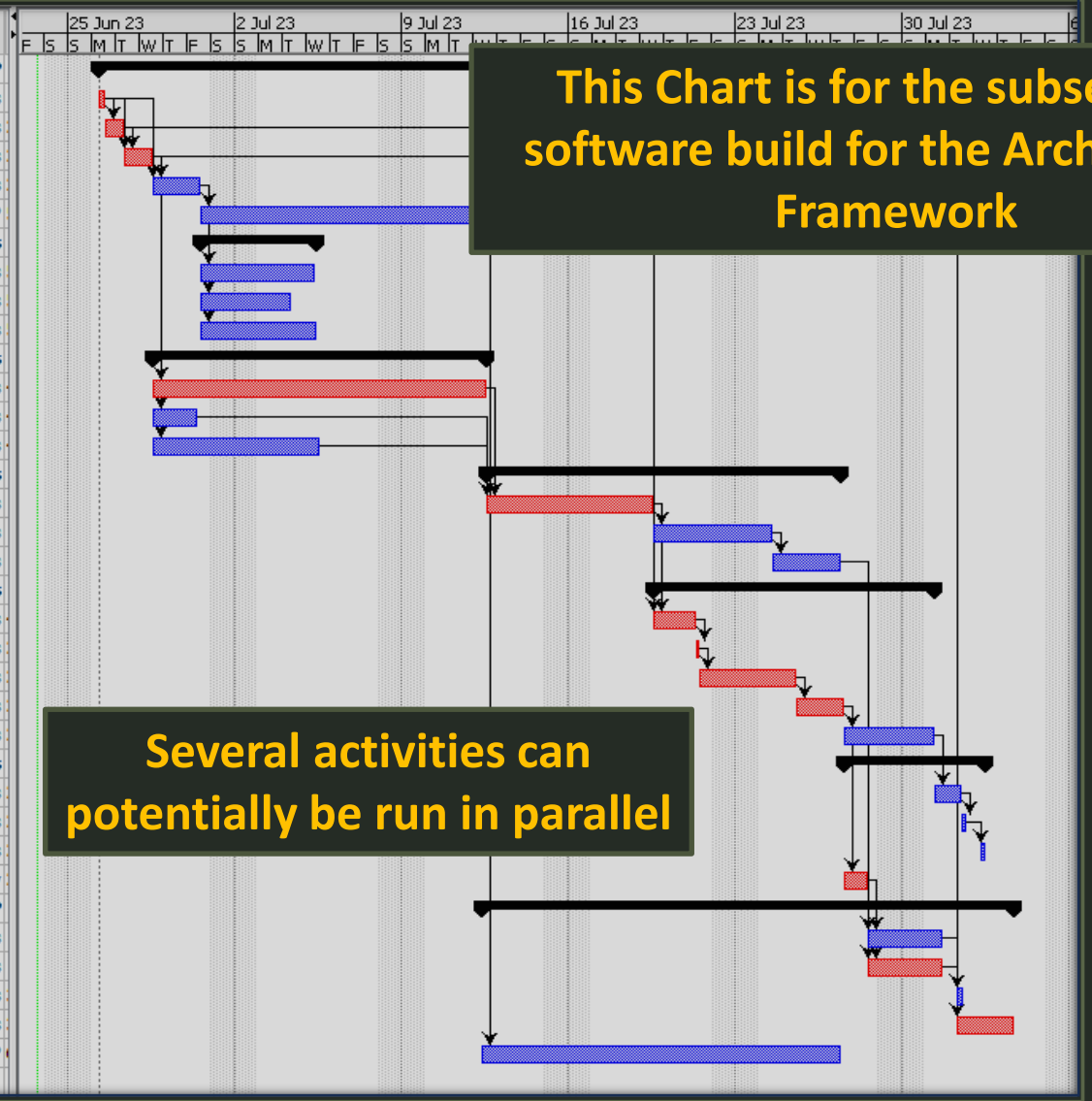
WBS – Work Breakdown Structure

9.3 Gantt Chart – GTRI Software Build

Georgia Tech Capstone

FR

	Name	Duration
1	**2.1 Project Management**	**12.125 days?**
2	2.1.1 Conduct initial assessment of the system delivered	0.75 days
3	2.1.3. Identify stakeholders and resources	0.375 days
4	2.1.3. Develop requirements for remaining work	1.5 days
5	2.1.4 Develop project management plan (including retirement plan)	2 days
6	2.1.5 Project Management Activities	7.5 days?
7	**2.2 Infrastructure, security, backend**	**2.75 days**
8	2.2.1. Infrastructure specifications	2.5 days
9	2.2.2. Incorporate security and compliance	1.5 days
10	2.2.3. Database specification and creation	2.75 days
11	**2.3 Customization and Refinement**	**10 days**
12	2.3.1. Develop code as per requirements	10 days
13	2.3.2. Design data encryption and security measures	1.75 days
14	2.3.3. Develop code for interface requirements	5 days
15	**2.4 Integration**	**10.75 days**
16	2.4.1. Perform integration of developed code	5 days
17	2.4.2. Perform system performance optimization	3 days
18	2.4.3. Create detailed system technical documentation	2.75 days
19	**2.5 Testing and QA**	**7.625 days**
20	2.5.1. Develop test plans and test cases	1.5 days
21	2.5.2. Deploy code to test environment	0.375 days
22	2.5.3. Perform unit testing of individual components	2.25 days
23	2.5.4. Execute functional and non-functional testing conduct (validation & verification)	2 days
24	2.5.5. Conduct user acceptance testing	1.5 days
25	**2.6 Production Deployment**	**3.75 days**
26	2.6.1. Prepare Production deployment plan	1.25 days
27	2.6.2. Prepare Production environment	0.5 days
28	2.6.3. Deploy code to Production environment	0.5 days
29	2.6.4. Document lessons learned	1 day
30	**2.7 Training and Support**	**16.875 days?**
31	2.7.1. Create support plan	1.25 days
32	2.7.2. Create training plan	1.25 days
33	2.7.3. Form support team	0.75 days
34	2.7.4. Conduct training for users	3 days
35	2.7.5. Code enhancement and maintenance (first 6 months)	11.25 days?



This Chart is for the subsequent software build for the Architecture Framework

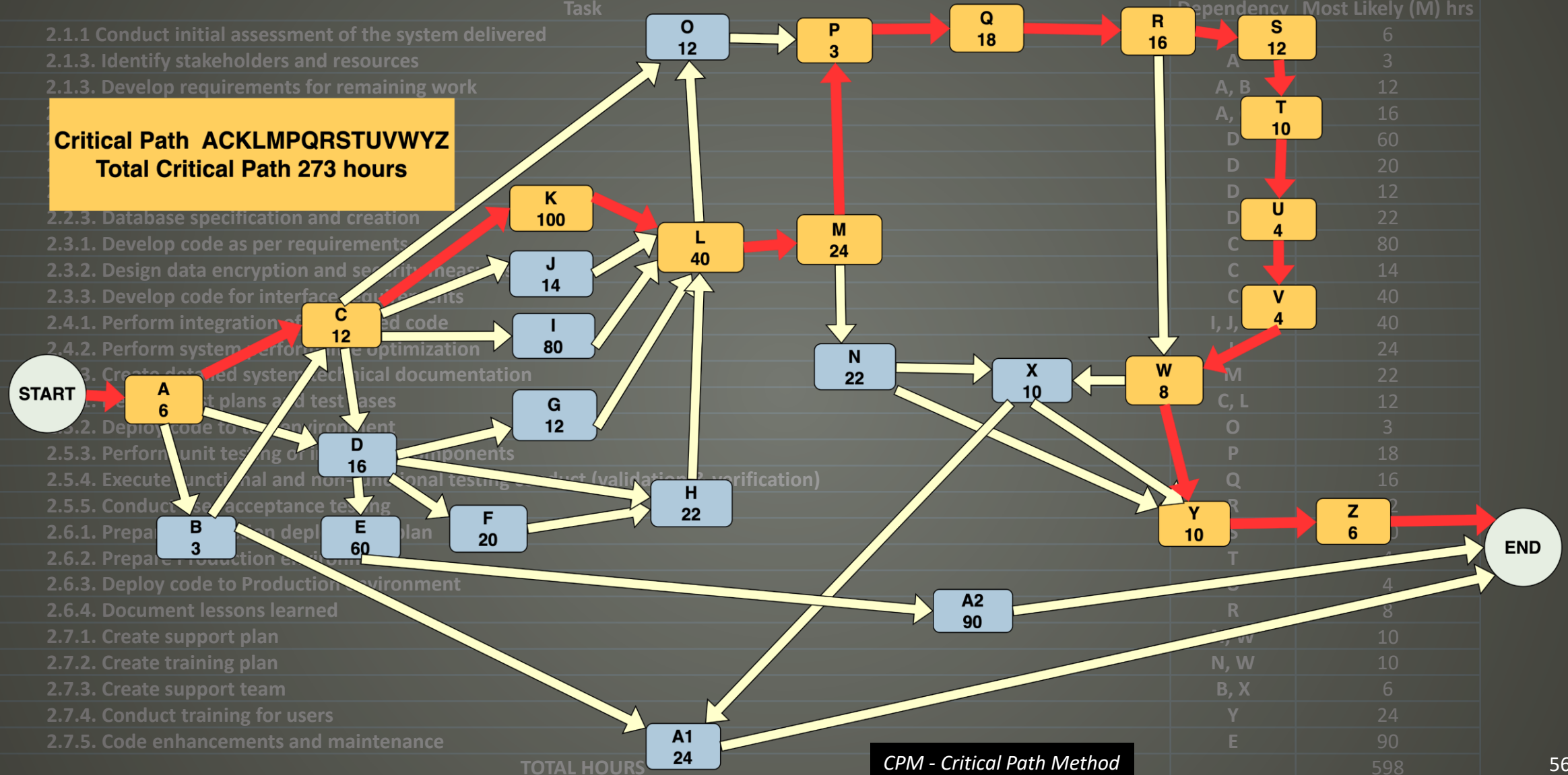
Several activities can potentially be run in parallel

9.4 CPM – GTRI Software Build

FR

Task ID	Task	Dependency	Most Likely (M) hrs
A	2.1.1 Conduct initial assessment of the system delivered		6
B	2.1.3. Identify stakeholders and resources	A	3
C	2.1.3. Develop requirements for remaining work	A, B	12
D		A,	16
E		D	60
F		D	20
G		D	12
H	2.2.3. Database specification and creation	D	22
I	2.3.1. Develop code as per requirements	C	80
J	2.3.2. Design data encryption and security measures	C	14
K	2.3.3. Develop code for interface requirements	C	40
L	2.4.1. Perform integration of developed code	I, J,	40
M	2.4.2. Perform system performance optimization		24
N	2.4.3. Create detailed system technical documentation	M	22
O	2.5.1. Develop test plans and test cases	C, L	12
P	2.5.2. Deploy code to test environment	O	3
Q	2.5.3. Perform unit testing of components	P	18
R	2.5.4. Execute functional and non-functional testing (validation & verification)	Q	16
S	2.5.5. Conduct user acceptance testing	R	12
T	2.6.1. Prepare implementation deployment plan	S	10
U	2.6.2. Prepare Production environment	T	4
V	2.6.3. Deploy code to Production environment	U	4
W	2.6.4. Document lessons learned	R	8
X	2.7.1. Create support plan		10
Y	2.7.2. Create training plan	N, W	10
Z	2.7.3. Create support team	B, X	6
A1	2.7.4. Conduct training for users	Y	24
A2	2.7.5. Code enhancements and maintenance	E	90
	TOTAL HOURS		598

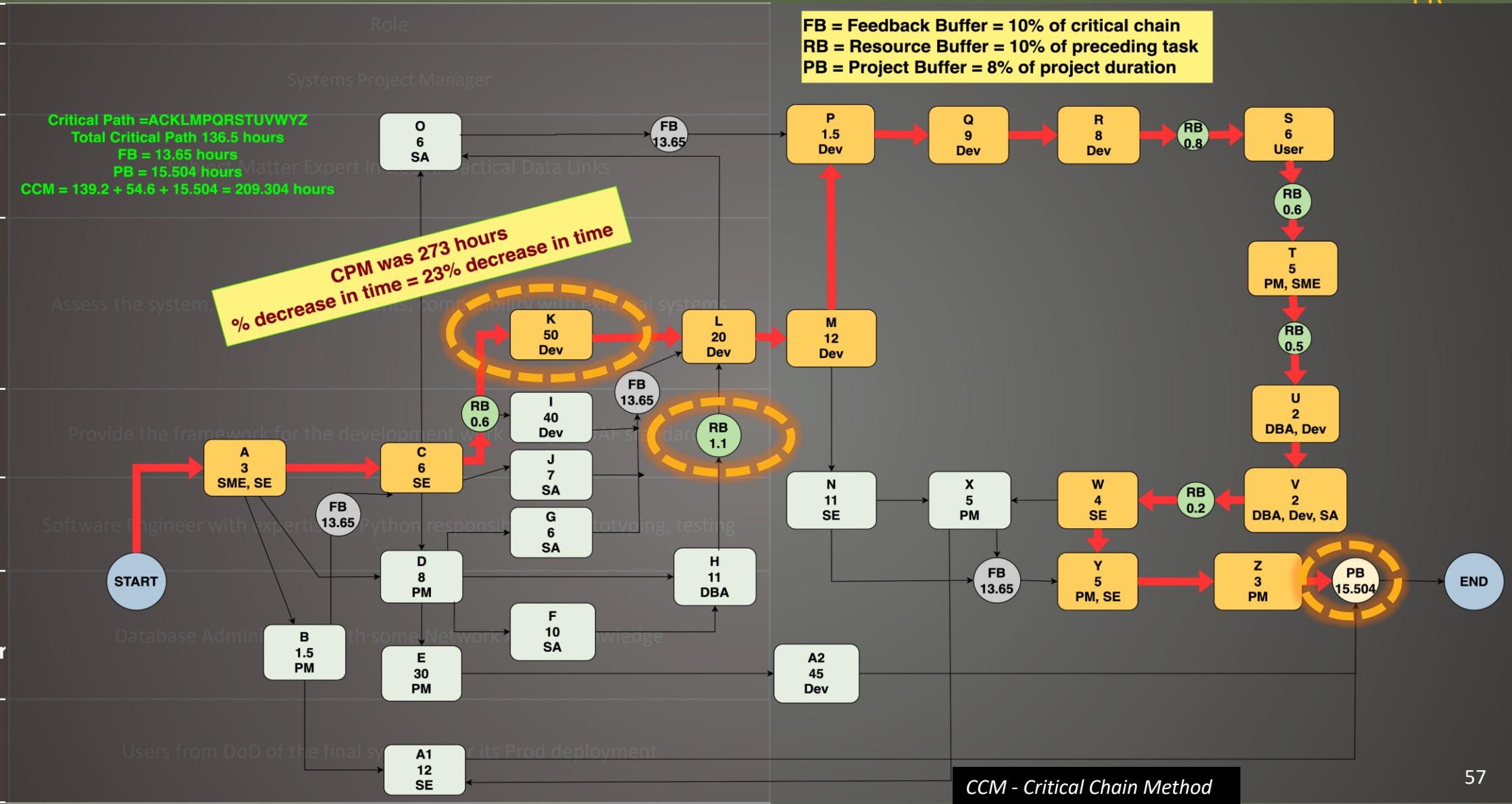
Critical Path ACKLMPQRSTUVWYZ
Total Critical Path 273 hours



CPM - Critical Path Method

9.5 CCM – GTRI Software Build

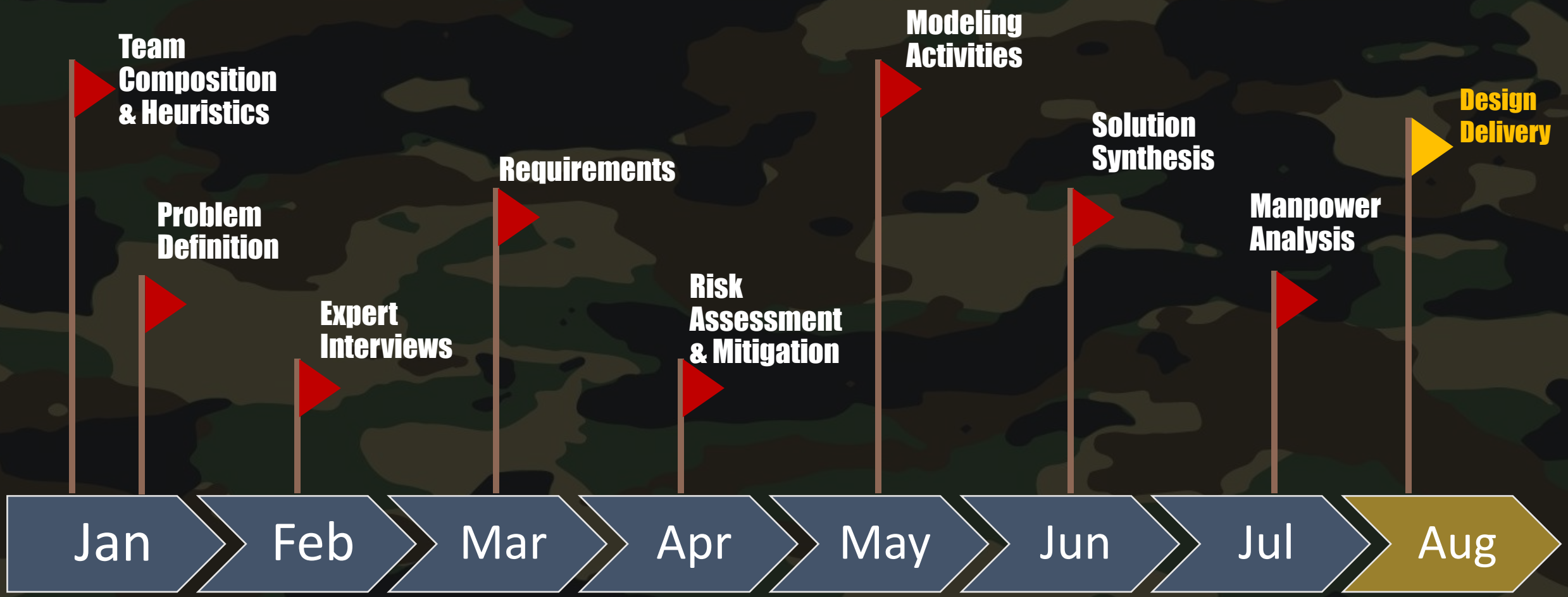
ID	Title
PM	PM
SME	SME
SE	Systems Engineer / Systems Analyst
SA	Software Architect
Dev	Developer / Tester
DBA	Database Administrator
User	MTA Tool User



CCM - Critical Chain Method

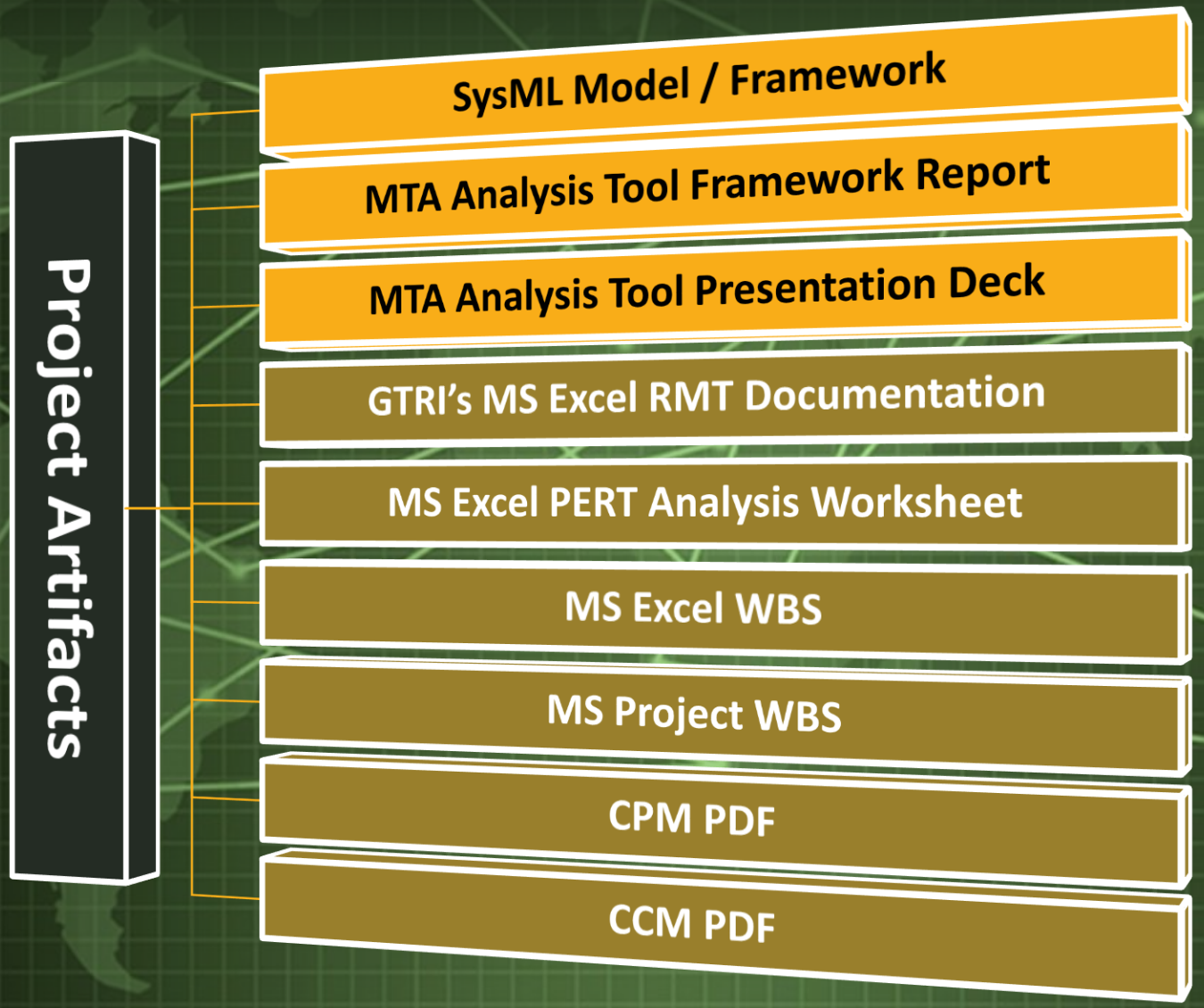
9. Design Delivery

Georgia Tech Capstone DC



12.1 Recommendations

DC



Project Objectives Met

Exceeded Expectations

Thank you

PMASE Professors & Support
SMEs / Mentors

Sponsor Team

GTRI

Graduate PMASE Student Families

Graduate PMASE Student Employers

Follow the Rabbit