THE PRAGMATIC SYSTEMS ENGINEER

Hard Earned Lessons on Systems Engineering From the Ages, for the Ages

Brian L. Shaub, CSEP, PMP, APMP-F, CSM

CP 571 236 1959, Brian.Shaub@Bluepiedmont.com

PRB 13 Jan 2021 by RUTHTM

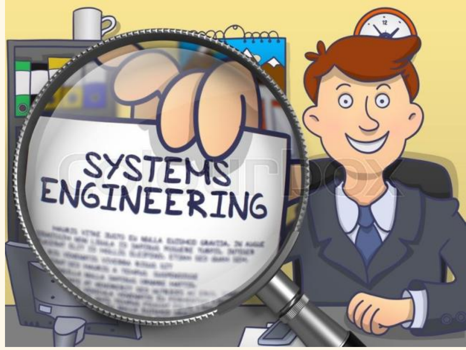
INTRODUCTION

- Facility Overview
  - Food, Restrooms, Emergency Exists
- Who are We?
  - Around the Room – 30 Seconds Each
- Objective
  - Know Systems Engineering When You See It
  - Converse with Systems Engineers
  - Apply Basic Systems Engineering
- What to Expect Today
  - Some Talk and Presentation
  - Group Case Studies
  - Verification Checks
- No Proprietary Information Will be Disclosed
- The Handbook and Training Slides Are Almost Identical – Training will have Section from Handbook

WHAT IS DIFFERENT ABOUT THIS TRAINING

- Starts over 5K Years Ago
- The Thread of Reductionist and Holistic effecting us today
- Importance of 'soft skills'
- Special Emphasis on 'Pain Points'
- Use of Modeling and Simulation for Schedules and Budgets
- Blending of SE Models, Methods, Lifecycles
- People, Process, Tool, and Inputs
- How to Use Systems Engineering in Business
- Something Big is Happening - 'Era of Enlightenment' 'The Fifth Era' 'The Forth Industrial Revolution'
- Three Generations of Systems Engineering

OUTLINE



Introduction
Section 0 – Before the Beginning …..
Section 1 – the **<u>Fun</u>**damentals – doing push ups
Section 2 – The Systems Lifecycle – and Development Models and Methods
Section 3 – Processes
Section 4 – Say What You Mean, Mean What You Say – The Language of Systems Engineers
Section 5 – Supporting Skills – Tricks of the Trade
Section 6 – What's Next in System Engineering – I take some liberties here
Appendix – SE Checklist, Case Studies, Glossary, Bibliography

SECTION 0 BEFORE THE BEGINNING

THE GENESIS OF SYSTEMS ENGINEERING

IMPROVING OUR CONDITION

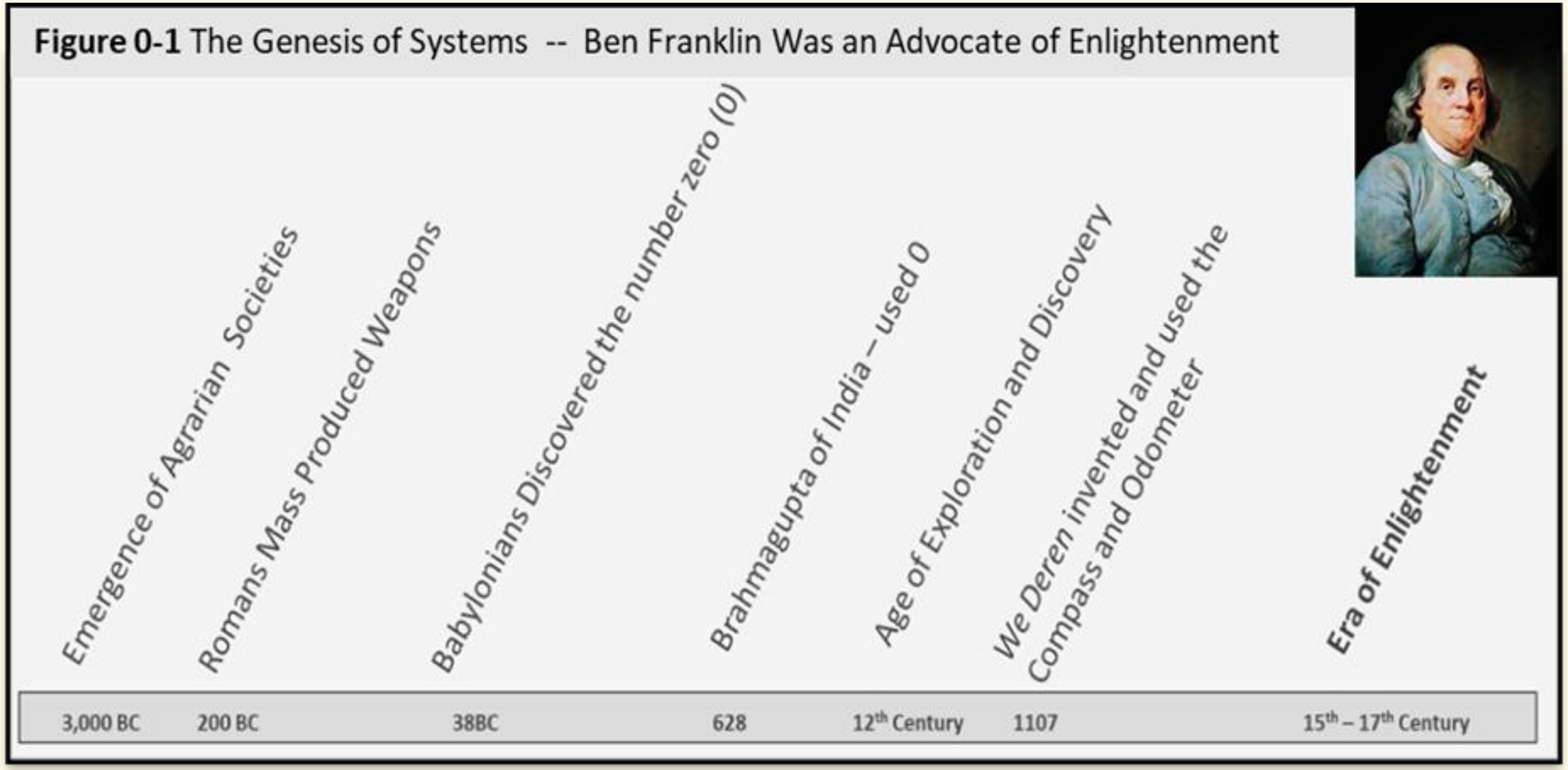
SOME AREAS THAT NEED IMPROVEMENT

WHAT CHANGED TO BRING ABOUT MODERN ENGINEERING?

- Era of Enlightenment
  - Self Governance versus a Monarchy
  - People can determine and improve their condition
  - Rectifying Religion and Science – empirical evidence vs anecdotal
  - Reductionist Theory – Top Down
    - An entity can be broken down into smaller pieces – a hierarchy
    - From which the larger piece draws resources from the smaller pieces
    - The Larger pieces has 'Emergent Behavior' which is greater or different from the smaller pieces
  - Holistic Approach – as in an eco-system - systems that evolve and change as needed, also composed of parts. Bottoms Up

For Systems Engineering - The Sum is Greater than its parts – whether reductionist or holistic.

THE GENESIS OF SYSTEMS ENGINEERING



Brahmi		—	=	≡	+	𑁖	𑁗	𑁘	𑁙	𑁚
Hindu	०	१	२	३	४	५	६	७	८	९
Arabic	٠	١	٢	٣	٤	٥	٦	٧	٨	٩
Medieval	o	I	2	3	[illegible]	ς	6	ʌ	8	9
Modern	0	1	2	3	4	5	6	7	8	9

Roman: I II III IV V VI VII VIII IX X

AND MORE CHANGE – SOMETHING BIG IS HAPPENING !!

- The Fourth Industrial Revolution by the World Economic Forum
  - Convergence of Artificial Intelligence
  - Autonomy
  - Internet of Things
  - Physical Life and Mechanical Control
- The Fifth Era
  - Digital Revolution – easy access to data, analytics, computing power
  - Quick communication, predictive analysis, data and analysis
  - Modeling and Simulation
  - Convergence of capabilities to affect previous unaffected industries – Transportation, Agriculture, and esp Biotechnology

ITS NOT ALL ABOUT COUNTING, MEASURING AND SYSTEMS – PEOPLE ARE MOST IMPORTANT

- The Organization –
  - Have a Purpose, a Vision, Goals
  - Structure, roles and responsibilities
  - Best Practices – a Routine
  - Publish It!
- Leaders, Managers, Administrators/Project Controllers - which one are you?
  - Leaders – motivate others to take a course of action
    - Situational – direct vs coaching vs, laid back
  - Managers – efficiently apply resources to achieve a desired outcome
    - Planning, assigning, adjusting, communicating
  - Administrators/Project Controllers – to keep order of documents, schedules, find improvements
- All Important and you can be all three, two of three or only one

‘WHAT WE HAVE HERE IS A FAILURE TO COMMUNICATE!’

WHAT WE HAVE HERE IS A FAILURE TO COMMUNICATE - Sheriff Buford T ...

- Leader sets the culture and the environment
- Common Goal, Vision, Ops Rhythm to all stakeholders (including team)
- Formal
  - The Organizational Artifacts, All Hands, Posters, etc
  - Project and Systems – the Requirements, Architectures, Minutes and Records, Test Plans and Results
- Informal
  - Informal to match formal – keep it consistent
  - Note non-verbal comms – 80 percent of comms is voice inflection, body language. Never trust email
  - Do ‘Walk Arounds’ – face to face
- Everyone counts – including the interns, the quiet people – who may have the best ideas

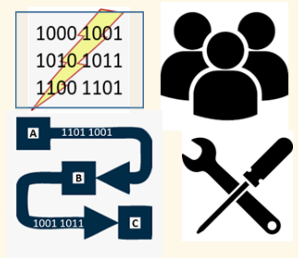
DO YOU HAVE STYLE ? – A STYLE GUIDE (1)

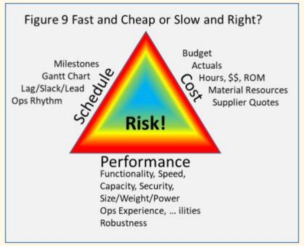
- Colors –
  - **Red is Bad** - High Risk, Out of Tolerance, Over Budget, etc
  - Yellow - getting bad, but can be fixed
  - Green is OK – on target, on budget, going as planned with minor repairable glitches
  - Blue – virtually perfect (but don’t get complacent)
- Fonts
  - *Italics – references to external papers, books, journals*
  - **Bold – for emphasis – an important point**
  - Underline – a defined word or expression
  - URLs – ease of navigation within document/slides, or to external sites
  - To the Book – gray usually top right
  - The ‘Call Out Box’ bottom of slide

Why is This Important? So those creating it have a common language and those reading it can understand and see it. Define a style at the start of a project to prevent rework

DO YOU HAVE STYLE – A STYLE GUIDE (2)

0.6 Style Guide
0.6.3 Terms
0.6.4 ICONS





- Terms
  - Will use ‘Systems Engineering’ but will apply to Project Management unless otherwise stated
  - Should – its optional, recommended but if not done will add risk
  - Must – do or it will cause a problem, may be required by regulation/policy
  - Will – a desire or intent, non-binding
  - Provider vs Contractor vs Vendor – those who provide the product – one of the same
- Icons and Notations
  - People, Process, Tools, Energy Input
  - Cost, Schedule, Performance, Risk
  - Decision Gate – A Gate Review, Milestone Review, Way Point – Thumbs Up or Thumbs Down
  - Model – a representation of a thing, structure, design, often simplified for understanding
  - Method or Methodology – a uniform set of activities for engineering, science - an empirical process to acquire knowledge, careful observation;
- More Terms and Icons as we go along

METAPHORS,
AND ANALOGIES & SIMILES

- Great way to express a complex idea but can cause confusion or oversimplification

  Boots on the Ground

  Shovel Ready

  I Smell a Rat

  Other Shoe Drops

  Writing on the Wall

  Got Our Arms Around It

  Cannot See the Forest for the Trees

  Brass Ring

  Baked In

  Needle in The Haystack

  Connect the Dots

  Her vocabulary was as bad as, like, whatever

- Misuse perpetuates confusion
- I am careful in the use of metaphors

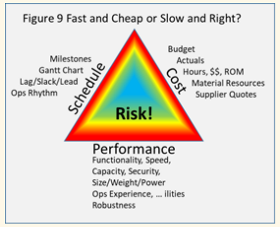
APPROACH TO THIS TRAINING

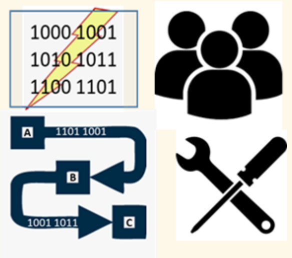
- Several Passes Over Systems Engineering – and level of training
  - Started with the Prerequisites
  - First Pass - Ten Thousand Foot Level – Knowledge
    - Summary and few details
    - History, What, Why
  - Second Pass – Five Thousand Foot and Some Details – Understand, Recognize
    - Methodologies, Lifecycles
  - Final Pass – On the Ground – Details – Critically Think, Improve
    - Processes, Activities, Language
    - Related Specialties – work as a team
- Not Just Academics but my experiences and suggestions along the way

It May Seem I'm Redundant but each time I go into more detail – the nature of systems

SUMMARY

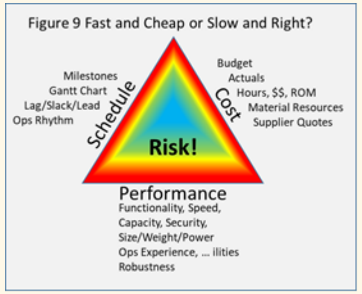
- Background of Engineering and Technology
  - A Language of numbers and values
  - Mass Production of Equipment
- Era of Enlightenment (16th & 17th Century)
- Reductionist – Top Down – systems draw resources on smaller parts
- Holistic – Bottom Up – finding inputs, outputs, connections
- Build your program, project and system on ….
  - A strong organization – leaders, managers, roles and responsibilities, vision
  - Common Terms and Vocabulary

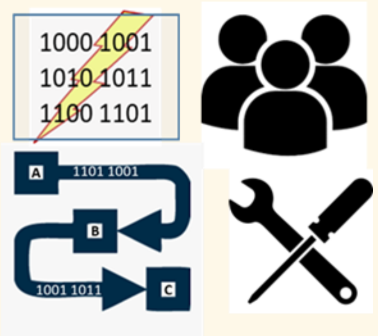




SECTION 0 KNOWLEDGE VERIFICATION

1. What Era was considered the seeds of Systems Engineering (at least in my opinion)
2. If we are only working in a systems environment, why is leadership, or management important ? – we are a bunch of techies.
3. Why are a common set of notations or language important?
4. Name one good and one bad metaphor.
5. What do these mean? - below





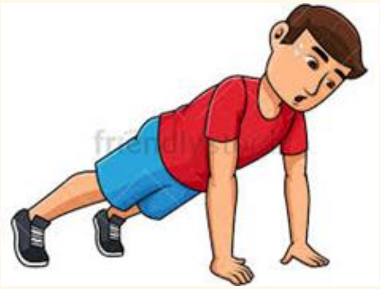
Responsibility is a unique concept... You may share it with others, but your portion is not diminished. You may delegate it, but it is still with you... If responsibility is rightfully yours, no evasion, or ignorance or passing the blame can shift the burden to someone else. Unless you can point your finger at the man who is responsible when something goes wrong, then you have never had anyone really responsible.

Hyman G. Rickover, Admiral,

Father of the Nuclear Navy

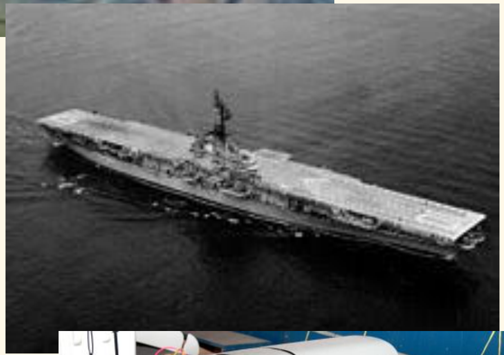
SECTION 1 SYSTEMS ENGINEERING FUNDAMENTALS

What, Why, How



20TH CENTURY SYSTEMS ENGINEERING

- Mentioned The 'Era of Enlightenment'
  - Systems were implied, not thought of in 'systems'
- 20th Century
  - Large Engineering Efforts that Required Various Engineering Domains
  - World War II – Aircraft Carriers, Aircraft,
  - Integrated capabilities of Air – Land – Sea and maintaining control
- Systems Approach continued beyond WWII
  - Space Systems
  - Transportation
  - Logistics
  - Telecommunications
  - Computer Systems
- Standards and Guidance
  - Military and Government
  - Commercial
  - Best Practices, Processes
- Primarily the Reductionist – Top Down



A SYSTEM IS ……

- A set of interrelated parts, each performing a unique function that delivers something of value
- May consider government or a nature as a system
- **Human made systems of material, energy, data that delivers a function that serve people**
- Systems are Complex
- Systems Deliver 'Emergent Behavior'
- Ubiquitous, interconnected
- System is bounded, scoped – cannot solve epic problems

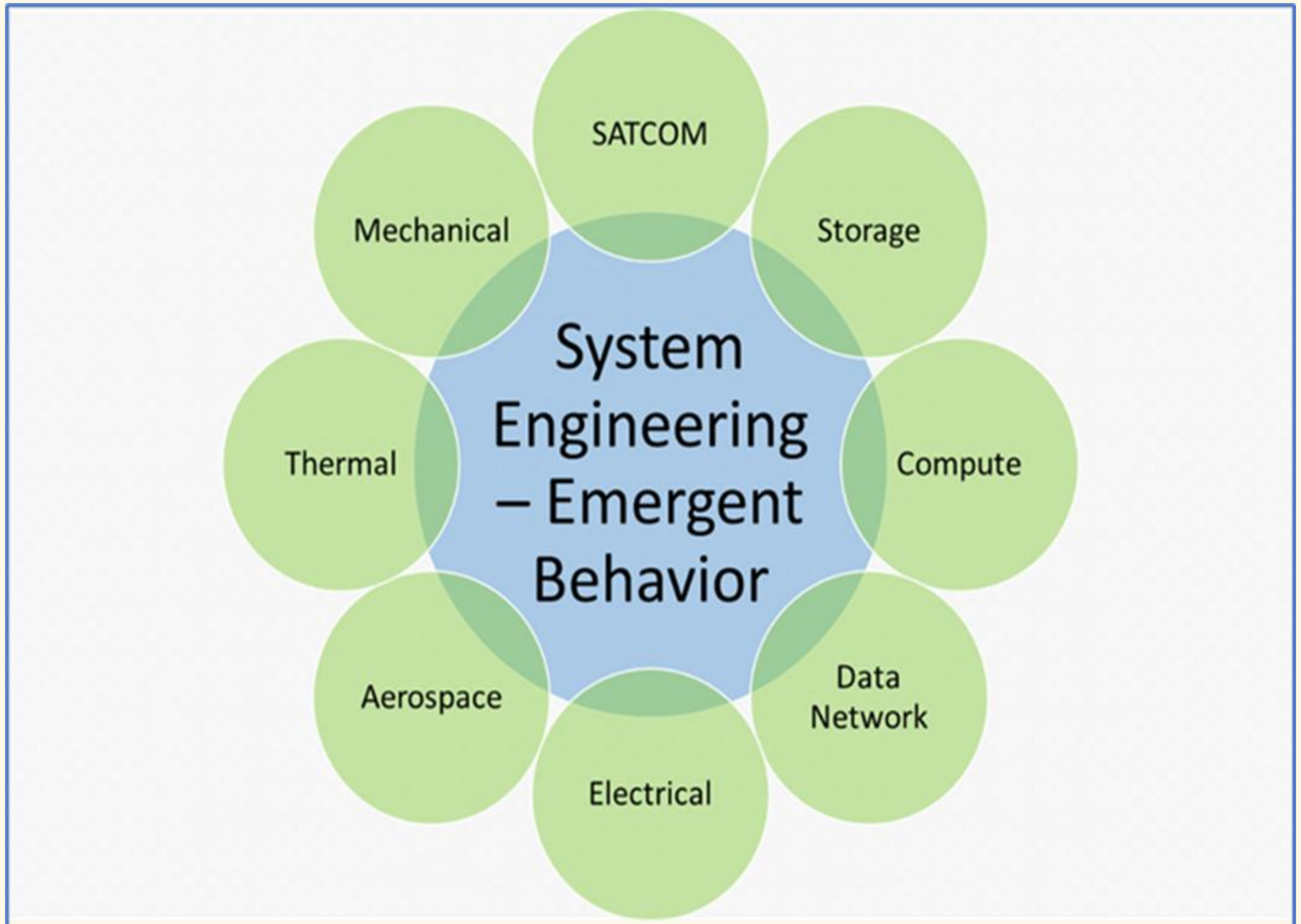
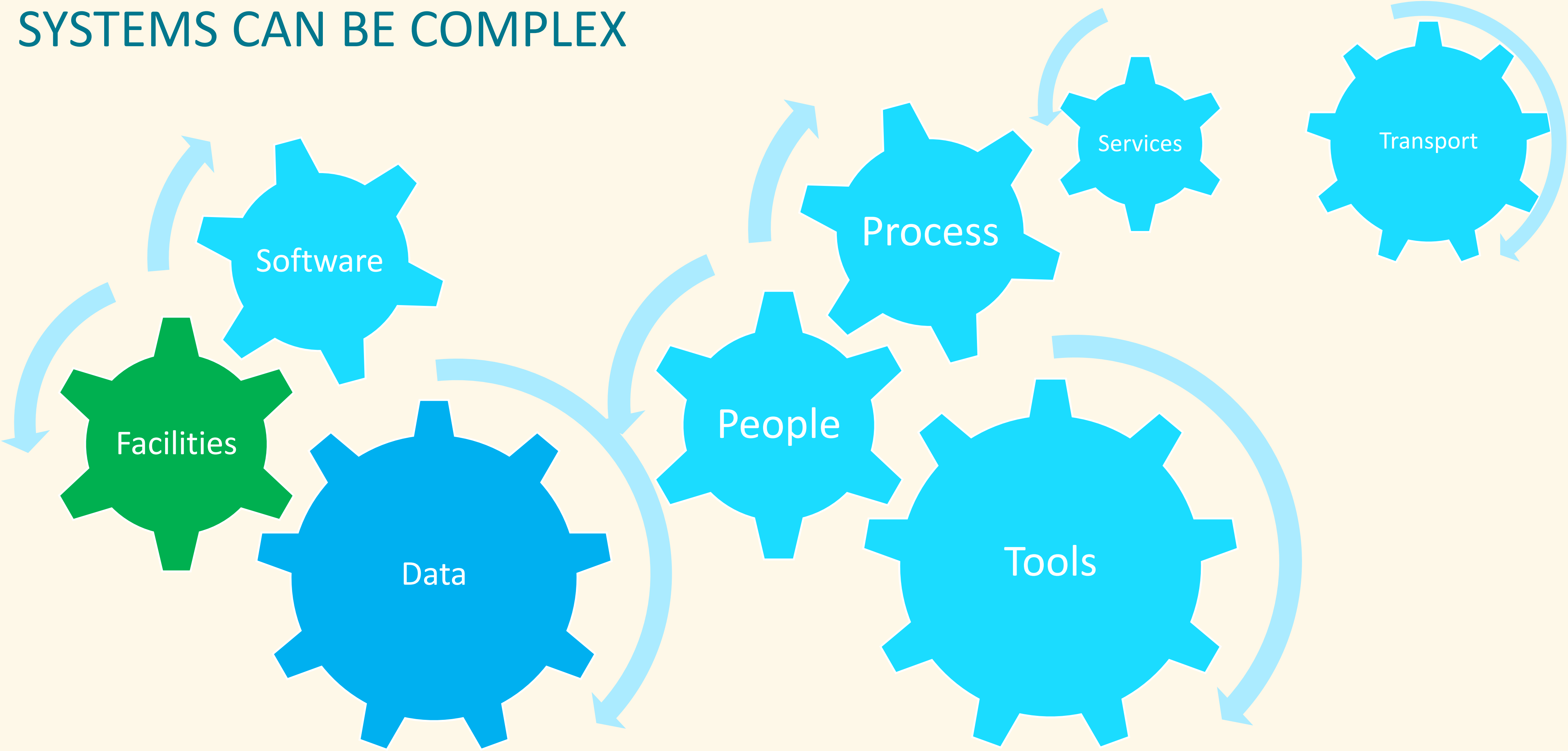


Figure 4 Systems Connect Various Engineering Domains to Realize 'Emergent Behavior'

Human Made to Serve Humans

SYSTEMS CAN BE COMPLEX



'The whole is more than the sum of its parts; the part if more than a fraction of the whole' Aristotle

Pandemic Ontology

Military C3 System

Aircraft Carrier

Insurance System

Russia MIR

Appointment System

'A Medical Records System'
Define Problem and System
Is bounded and scoped
** This is Your System ***

Aerospace

SpaceX

SCADA

Transportation System

Telephony

Data Center

Auto Manufacturing

Natural Gas Distribution

WHAT IS SYSTEM ENGINEERING

- An interdisciplinary approach and means to enable the realization of successful system. (INCOSE, 2004)
- Solving a complex problem by discovering and learning of the problem, that is composed of various entities and disciplines by **decomposing** into smaller pieces, planning different solutions, then solving he problem by constructing the system, modeling the behavior, while ensuring it will function and perform as planned.
  - Discovering and Learning of the Problem
  - Complex Problem
  - Various Entities and Disciplines
  - Breaking It Down or **Decomposing**
  - Planning Your Architecture, Design, even the schedule and Budget
  - Modeling the Emergent Behavior
  - Ensuring it will function and Perform – testing
  - Different Solutions (alternatives)
  - Constructing the System

What Does All That Mean?

What is System Engineering

- Discovering and Learning – communicating with users/operators, observing behavior, eliciting details of the problem – this is requirements collection.
- Complex Problem – problems or challenges involving more than one domain – mechanical, electrical, compute and finance, business, contracts
- Various Entities and Disciplines – each domain thinks and often live in their own world, not thinking too much about how they will relate to others – until later in their careers
- Breaking It Down or Decomposing – Reduce the problems to understandable levels and parse out to each domain.
- Planning Your Architecture, Design, even the schedule and budget - function is designed and planned but how much will it cost, how long will it take – what are the risks?
- Emergent Behavior – the system provides a service greater or different then its parts, usually predictable but sometimes an anomaly
- Modeling the Emergent Behavior – a structure to anticipate the emergent behavior.
- Ensuring it will function and Perform – test as you go along – how does the actual behavior relate to the expected behavior?
- Different Solutions (alternatives) – to allow for most suitable solution – fast and expensive, or slow and cheap
- Constructing the System – building, assemble, integrate – as designed

Modeling and Simulation

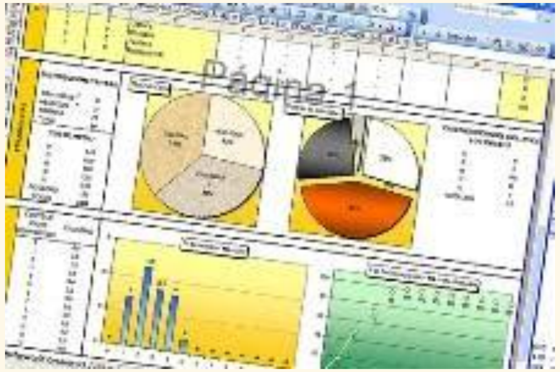
- Planning Your Architecture, Design, Resources Expenditure - even the schedule and budget
- Modeling the Emergent Behavior - model and simulate, model and simulate, model and simulate

Winds Tunnels are Models & Simulation

$A=1+2$ $\quad ax^2 + bx + c = 0 \quad$ $P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad$ $\lim_{x \to p} f(x) = L,$

'All Models are Wrong but Some Are Useful' – George Box, 20th Century Philosopher (Smart Guy) said this. What did he mean? I believe that we should keep a skeptical eye on a model – a syntax that represents and abstract reality. We tend to oversimplify or lose context in the model, or the model is mis-used. Make a good, useful model but always ask questions – 'what and why, and so what?' (Georgiev, The Start Up, 2019)

https://medium.com/swlh/all-models-are-wrong-does-not-mean-what-you-think-it-means-610390c40c9c



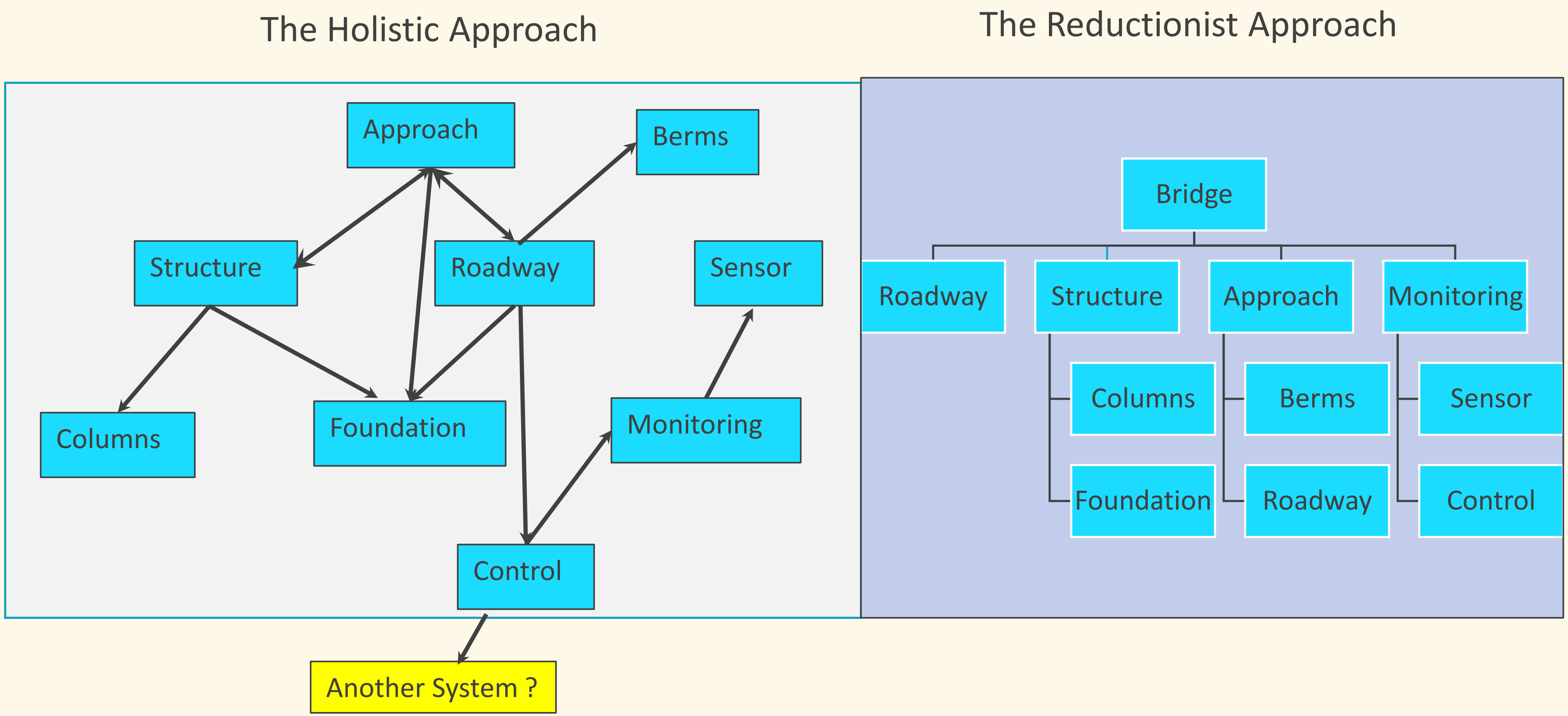
'YOU ARE A SYSTEMS ENGINEER!'

Table 1: Systems Engineers are usually a blend of Thinkers or Perspectives but people naturally lean one way or the other	
Synthesis ← Some of Each → Analysis	
<u>Systems Thinkers</u> Holistic, **bottom up** approach of **synthesizbottom-up**behavior of the small parts, how it contributes to the larger. The ability to understand the cause and effect, cascading events, interconnections of actions and an ontology.	<u>Systems Perspective</u> Reductionist and hierarchy approach, **top-down analysis**, inputs – process – output and the results; abstractions and models, learn and understand the system that includes parts, what is required for each part, and anticipating the emergent behavior.
<u>Systems Professionals</u> Built on Thinkers and Perspective – Certifications and University Degrees – the custodian of the Systems Engineering Profession. Systems Professional can determine the best way to deliver a solution, to include the processes, documentation, risks – even if it's a hybrid or new method.	
Systems Engineer Development – 70% experience/OJT, 20% mentoring, 10% training	

https://www.youtube.com/redirect?redir_token=LQrhsnRWMzWH86J_z1rOAtILZx18MTU4ODA5NDYxMUAxNTg4MDA4MjEx&v=QrocQLqqIKQ&q=https%3A%2F%2Fwww.amazon.com%2FEngineering-Design-Perspective-William-Fortney%2Fdp%2F1987483413%2Fref%3Dsr_1_8%3Fie%3DUTF8%26qid%3D1534361776%26sr%3D8-8%26keywords%3Dsystems%2Bengineering%2Bdesign&event=video_description

HOLISTIC AND REDUCTIONIST OF **A BRIDGE**

HOLISTIC DISCOVER THE SYSTEM THROUGH OBSERVATION AND TRACING
REDUCTIONIST PERCEIVES IT AS A STRUCTURE AND HIERARCHY



TYPICAL WORK OF A SYSTEMS ENGINEER

- Work with Users to define what they want to do, and what they need
- Document the Requirements based on what users say, your observations, and regulatory, law, policy
- Create (or work with Architects/Designers) to create potential solutions
- Lead the team in selecting the most suitable alternative
- Aid in selecting providers (sub-contractors), establishing agreements and parsing work.
- Lead and manage the testing process to capture the results
- Set up and perform modeling and simulation
- Set Up and Manage the Decision Gates – (way points)
- Work with the project manager, domain engineers for integration of processes, systems, manage cost and schedule
- Monitor and maintain the system, improve it and modify it
- Retire and dispose of the system when finished – ensure transition is complete
- **The SE is NOT** – System Administrator, a Software Programmer, Windows Engineer, a Project Manager or Administrator, Project Controller or just a good person to have around to solve technical problems

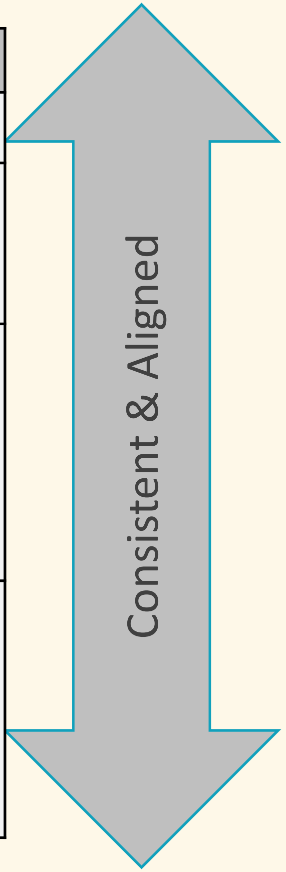
1.6 Common Stakeholders and their Perspective?

	Systems Engineers	Program or Project Managers	Contracts / COTR	User and Operator
Are my people in the right positions on the same playbook?	Are my engineers working the right tasks?	Are people charging to the right charge number?	Are CLINS overrunning the budget?	I'm a paying too much for this system?
Is our timing synchronized?	Will we create our design on time?	Where is the critical path in the schedule?	Will vendors comply with the scheduled?	Will it be ready and available when I need it?
Will we cross the goal line?	Have I traced my requirements?	Will the requirements be met?	Will they satisfy the SOW Requirements?	Will it perform and operate as long as I need it?
Do I need a more robust team?	I need someone who can write embedded 'C' Code!	Do I need to hire another experienced developer?	Is there a position for another full time equivalent (FTE)?	Will I need someone to perform maintenance on it?
What will take me off the game plan?	What may go wrong in My System?	Where is Murphy on My Project?	Have they delivered the Risk Management Plan for My Contract?	I don't what to think about this system too much, I have my job to be concerned about.
Table 1 Same Objective, Differing Perspectives				

Success is defined differently based on one's perspective – more in Enterprise Architecture

DO YOUR JOB !

Table 2 What is My Job?			
Org. Level	**Purpose**	**Directive/ Guidance**	**Major Participants**
Corporate, Division, Business Area	Business, Mission – Profit/Loss, Make $$	Policy and Guidance, Strategy, Federal Acquisition Authority	Owners, Board, CEO, CTO, Sr Mgt Team
Program	– Delivering Product or Service, Then Sustaining It –	FAR, Program/ Engineering Policy, Cost/Schedule/ Performance and Risk Metrics, Gate Reviews, Enterprise Process, Best Practices, PM Plan, **SE Manager Plan (SEMP)**	PM, Lead Engineer, SEIT, SMEs, Operations, Contracts, Finance,
Project	Temporary Endeavor -Distinct Start / End, Unique Purpose	Program/Engineering Plans, Cost/Schedule/ Performance/Risk, Metrics, Architectures, Best Practices, Processes, Gate Reviews, Checklists	SE, Project, Task Managers, Lead Engineer, Developers, V&V, SMEs



Note Importance of the **Systems Engineering Management Plan (SEMP)**
Systems Engineering Integration and Test (SEIT) Senior or Lead Systems Engineer at a program or division level. Responsible for all processes, people, tools but may these people may be matrixed into the program

Must Work as One Seamless and Focused Team

THE SYSTEMS ENGINEER VS THE PROJECT MANAGER

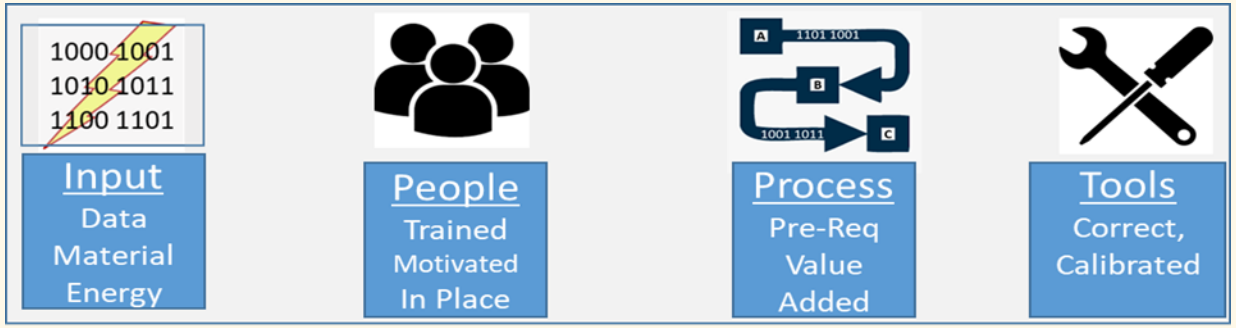
- When assessing a system or project – ask these questions and look for these things – too many – aahs??, or dahs?? – and they have problems

- **Murphy** is on Every Project, in every system

Sgt Murphy – a fictitious WW2 Solder who could NOT get it right – so all missions had to be Murphy Proof

Figure 9 Fast and Cheap or Slow and Right?

The Project Manager



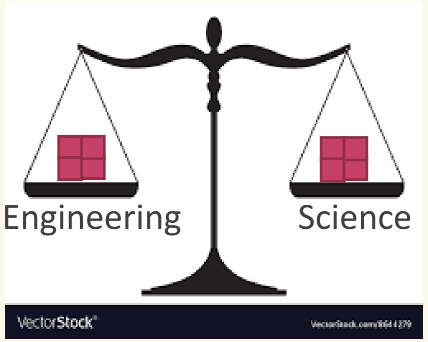
The Systems Engineer

Systems Engineers should use BOTH Perspectives

SCIENCE VS ENGINEERING – DIFFERENT BUT COMPLIMENTARY DISCIPLINES (WIKIPEDIA)

- Engineering - use of scientific principles to design and build machines, structures, and other items, including bridges, tunnels, roads, vehicles, and buildings.
- Science - (from the Latin word *scientia*, meaning "knowledge")[1] is a systematic enterprise that builds and organizes knowledge in the form of testable explanations and predictions about the universe.

New Technologies – Examples - are these ready to use in a system or not!		
Product	**Potential Capability**	**Status**
Metal Foam	Heat Resistance, durable and strong – but is toxic to people, and my corrode with connected parts.	Prototype
Quantum Dot	Semiconductor – few nanometers in size that illuminate when charged by UV light in different colors. Restrictions on by-product heavy metal	Not safe for consumers - Theories and Lab Experiments
Conductive Polymer	Organic polymers that conduct electricity, but disperse the electricity	Lab Experiments



Balance between Science and Engineering is Key

TECHNOLOGY READINESS LEVEL (TRL)

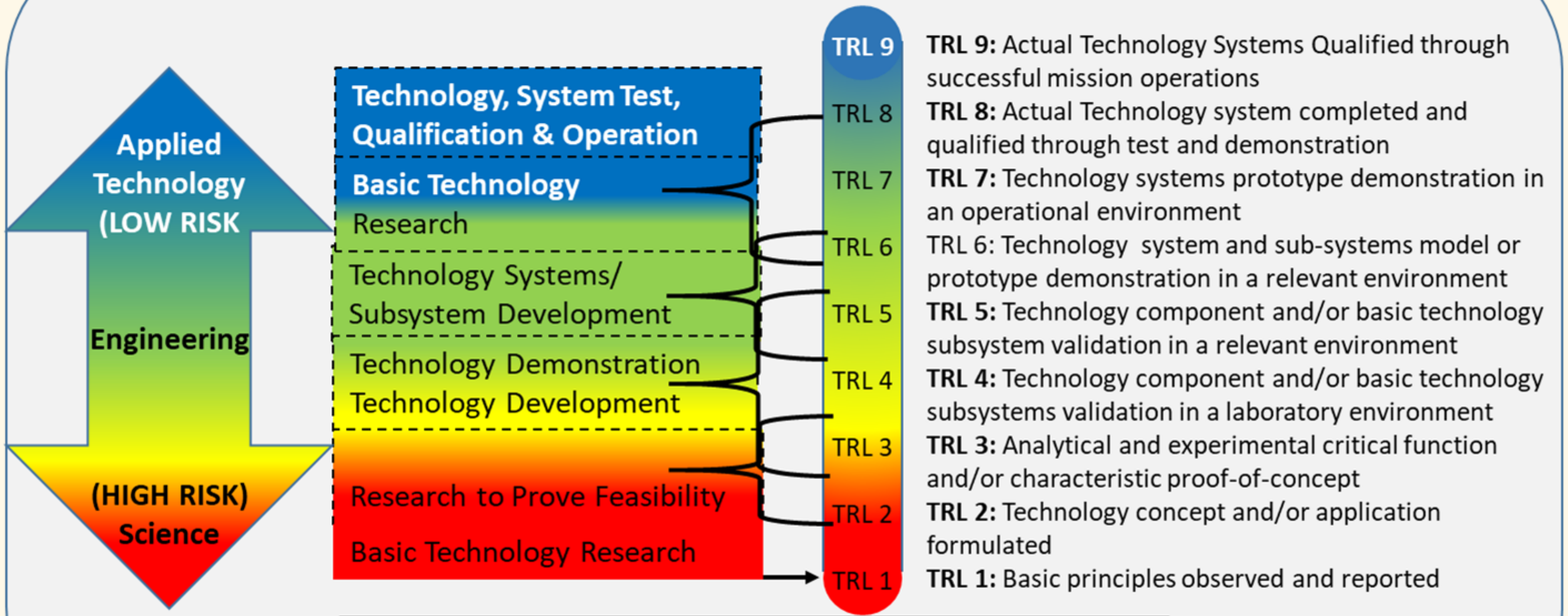


Figure 10 Use the TRL to Assess the Readiness of Technology
Source: ASD/R&E, TRA Guidance, April 2011

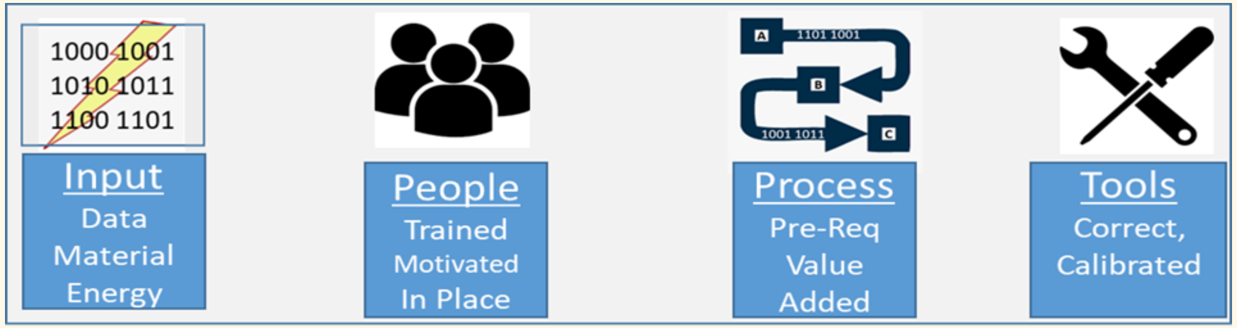
- Your organization probably doesn't use TRLs but you can still apply to assess
- Ask the provider if the product is used outside a lab, or if there is a 'White Paper'
- Ask for documented evidence
- Or do your own assessment

Leading edge is OK – Bleeding Edge is NOT OK

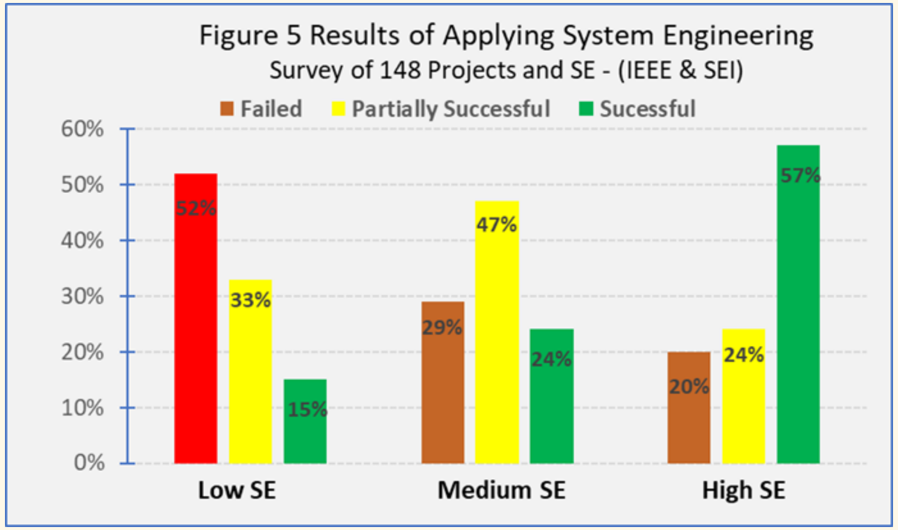
WHAT HAVE WE DISCUSSED SO FAR?

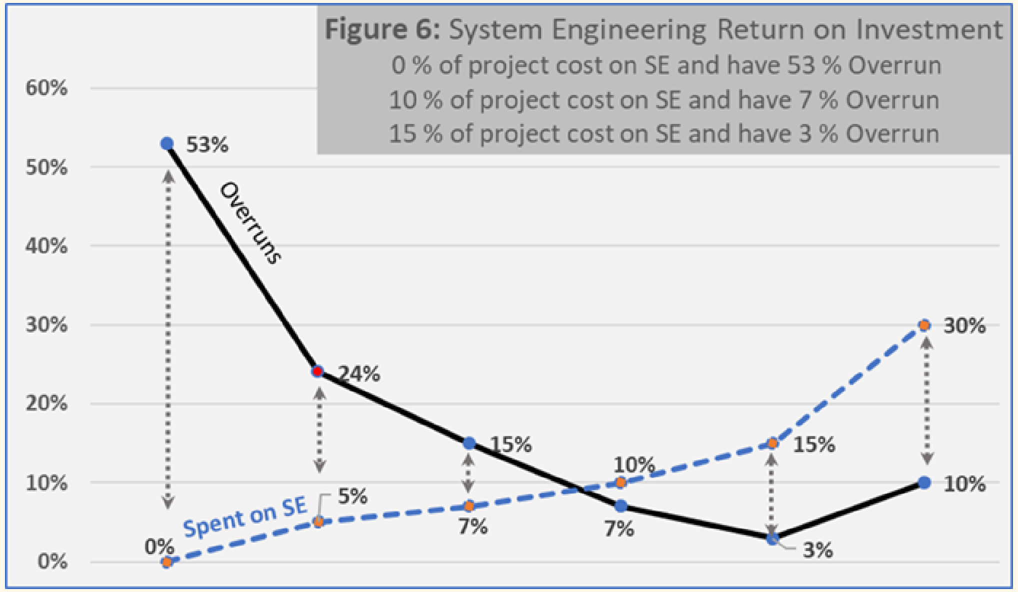
- What – solving complex problems, varies entities and disciplines, decomposing, modeling and simulation, Testing, Integrate
- Remember ‘Era of Enlightenment’
- Smaller elements provide resources to larger element which provides **Emergent Behavior**.
- How – decompose, **model** and **simulate**, select most suitable alternative
- Who – Systems Thinker, Systems Perspective, Systems Professional
- Why – Saves Money and Time, improves likelihood of success
- Cost, Schedule, Performance and Risk
- People, Process, Tools and Input
- Way Points are Decision Gates

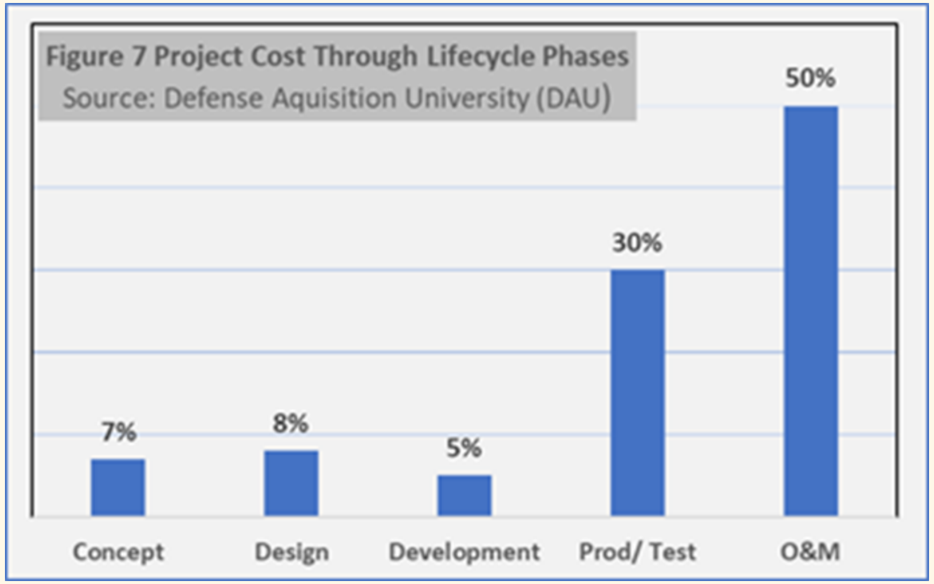
Figure 9 Fast and Cheap or Slow and Right?



Why Perform Systems Engineering



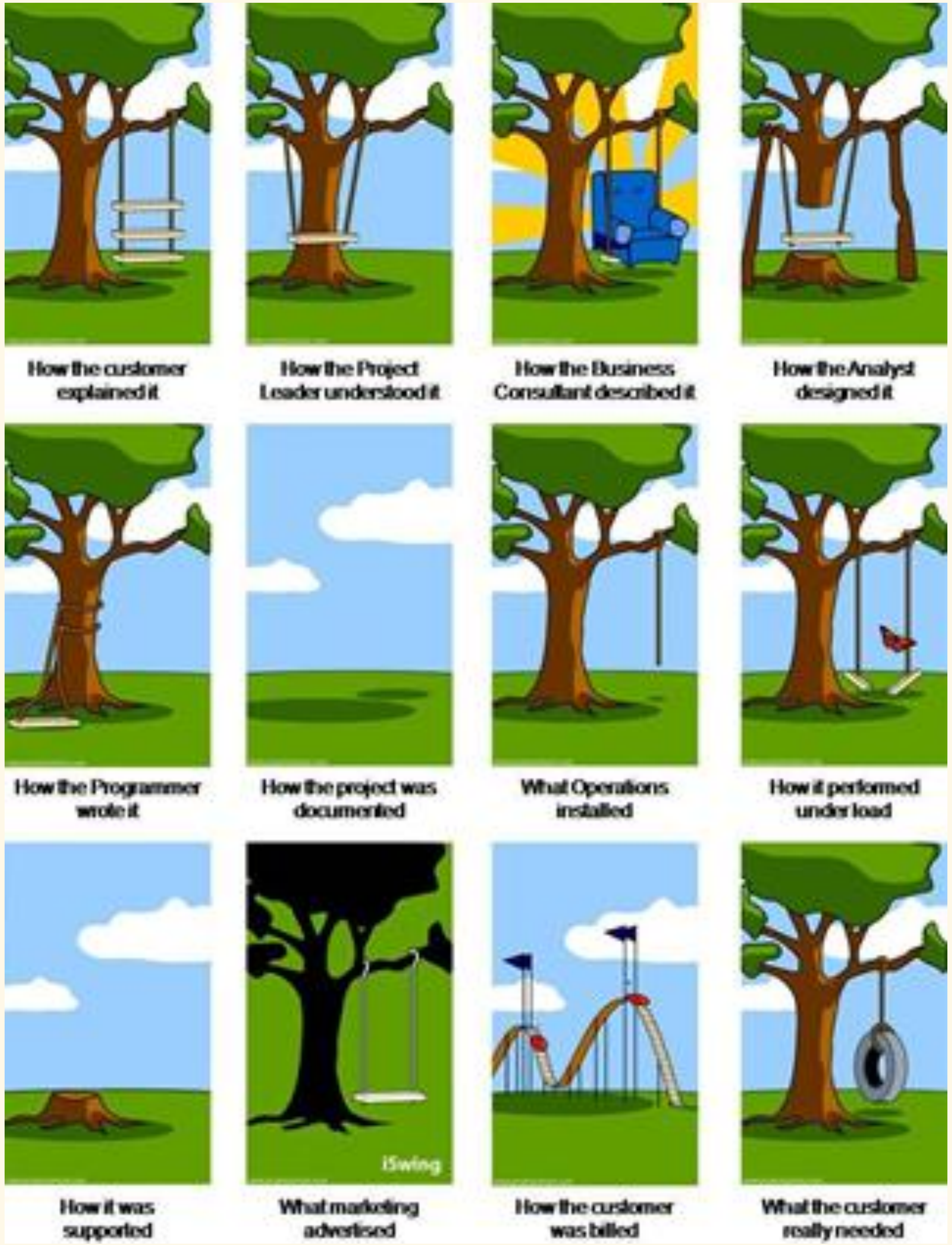




There is no business case in haphazardly building a system.

SECTION 1 KNOWLEDGE VERIFICATION

1. What is Systems Engineering?
2. Why Use Systems Engineering?
3. Who are Systems Engineers – what three ‘types’ are there?
4. Which type are you?
5. What is Emergent Behavior?
6. What is the ‘mental model’ for the Project Manager?
7. What is the ‘mental model’ for the Systems Engineer?
8. How can you break a problem down to its basic understanding (what is that called)?
9. How are Science and Engineering related?
10. Name a system that you encountered so far, today.


How the customer explained it
How the Project Leader understood it
How the Business Consultant described it
How the Analyst designed it
How the Programmer wrote it
How the project was documented
What Operations installed
How it performed under load
How it was supported
What marketing advertised
iSwing
How the customer was billed
What the customer really needed

SECTION 2 THE SYSTEMS LIFECYCLE

Second Pass – at Five Thousand Feet
We can go high speed fast version, or we can go slower – long version

APPROACHES TO A LIFECYCLE METHODOLOGY, MODEL AND FRAMEWORK

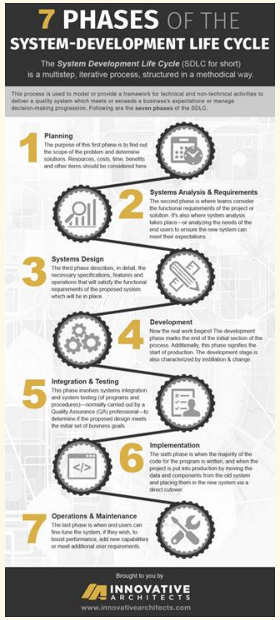
- Methodology- Practices, techniques, procedures, and rules used by those who work in a discipline. A methodology may offer opportunities for innovation or change but work within the guidelines of the defined methodology.
  - Process and action oriented
  - For brevity, the term Method will also be used.

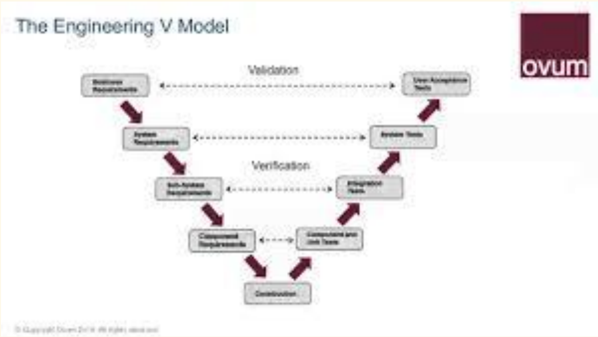
- Model – to form or plan - the practiced activities to realize a system as visualized are a Lifecycle Model. Models are stable, are reused for management of projects or to determine the emergent behavior of a system. Recall that modeling is the structure, and inserting data and material into the model is simulation.
  - Goes hand in hand with simulation
- Framework – the loosely defined guidelines, offering boundaries, on how a set of activities are performed.

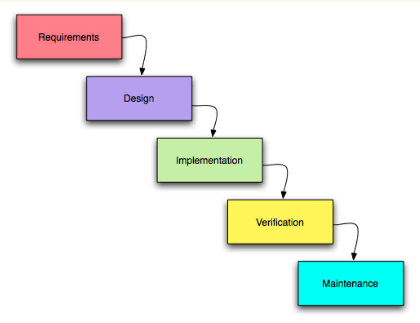
These terms overlap, often misused, and the use of a method may drive the model. So I use the terms almost interchangeably here. But remember the difference.

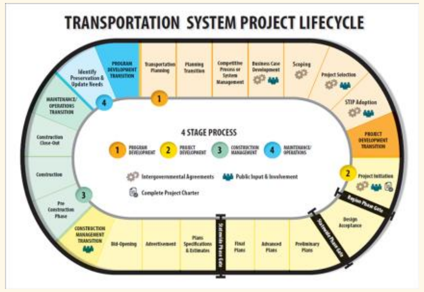
WHAT IS A LIFECYCLE

- All systems (including natural systems) – don't live forever into perpetuity
- They are all 'born', they have a 'life' that offers the needed service, they must be 'cared for', but they someday are no longer needed
- Select or create a lifecycle model that allows the team to understand and visualize the plan
- Important for the System Engineer to understand the pros and cons – select one or a combination
- Not mutually exclusive – may use one or a combination
- Establish a 'Dialog' with your system – (Alistair Cockburn, 'Agile Software Development – A Cooperative Game')
  - Recursive – Digging into the details to learn more, solve the details
  - Iteration - stepping back, repeating steps as you learn more, and adding detail



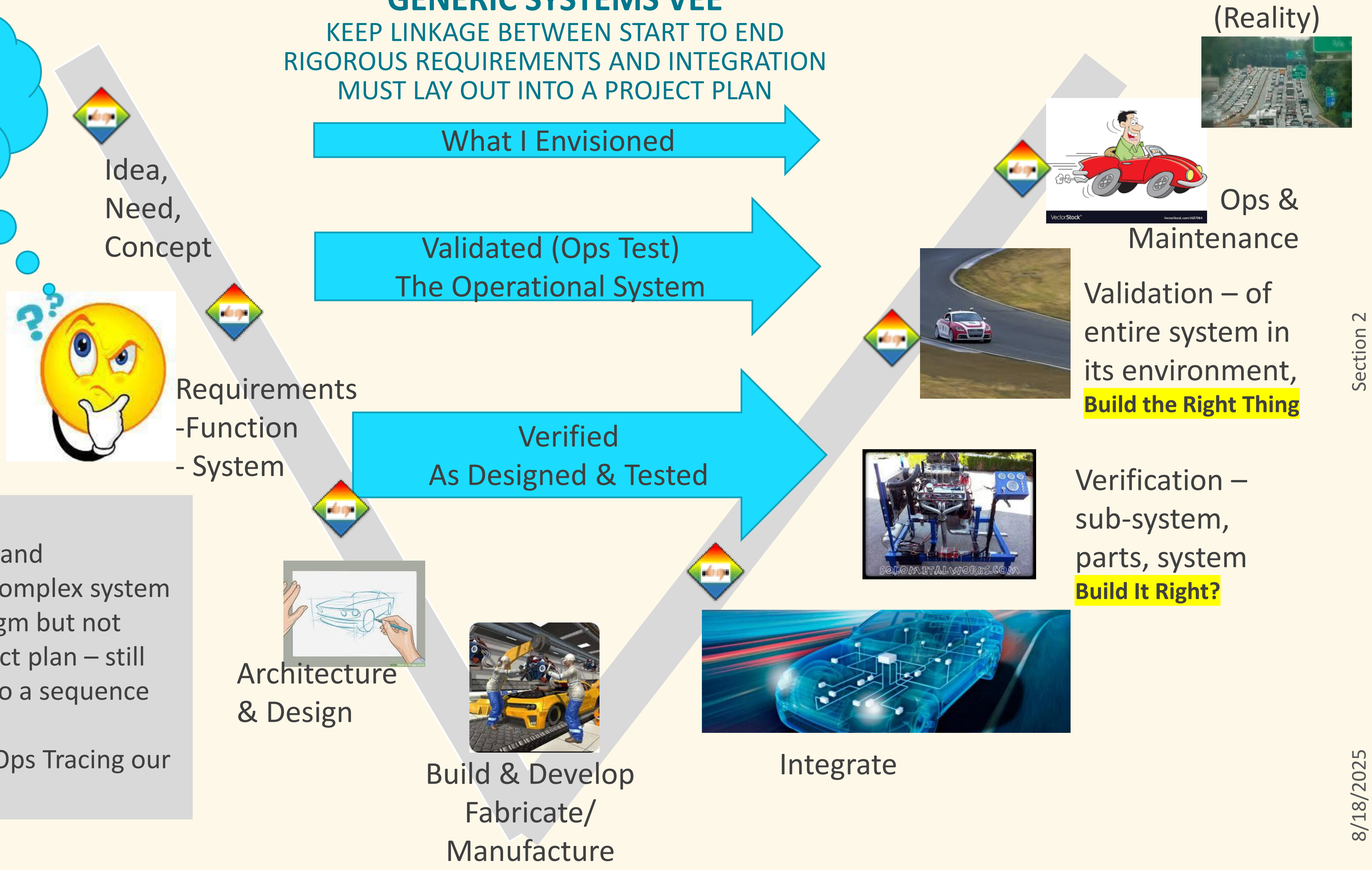






GENERIC SYSTEMS VEE

KEEP LINKAGE BETWEEN START TO END
RIGOROUS REQUIREMENTS AND INTEGRATION
MUST LAY OUT INTO A PROJECT PLAN



Traits

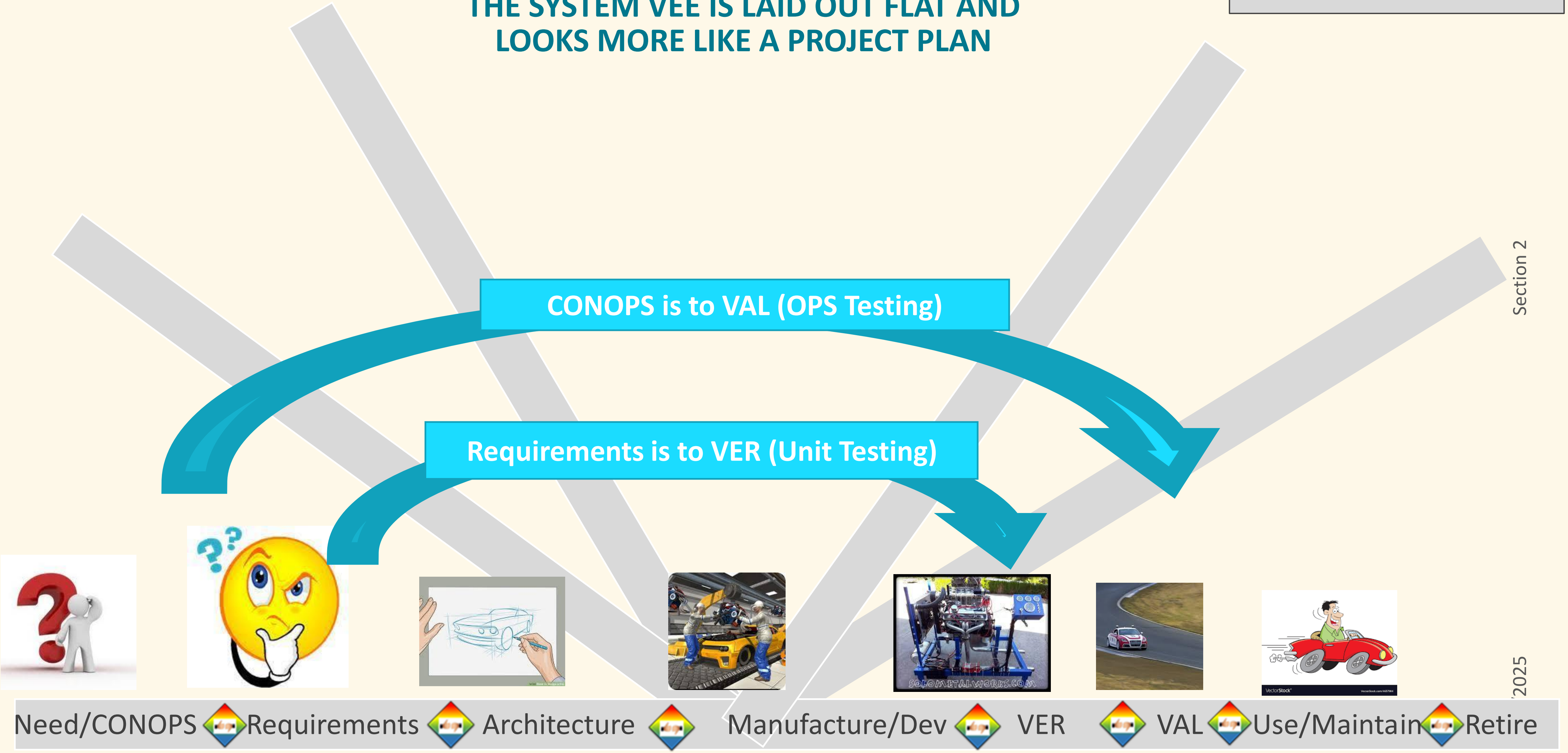
- Aligns the interests and requirements of a complex system
- An excellent paradigm but not effective for a project plan – still must be laid out into a sequence for scheduling
- Requirements and Ops Tracing our inherent

SUCCESSIVE AND EMBEDDED SYSTEM VEE

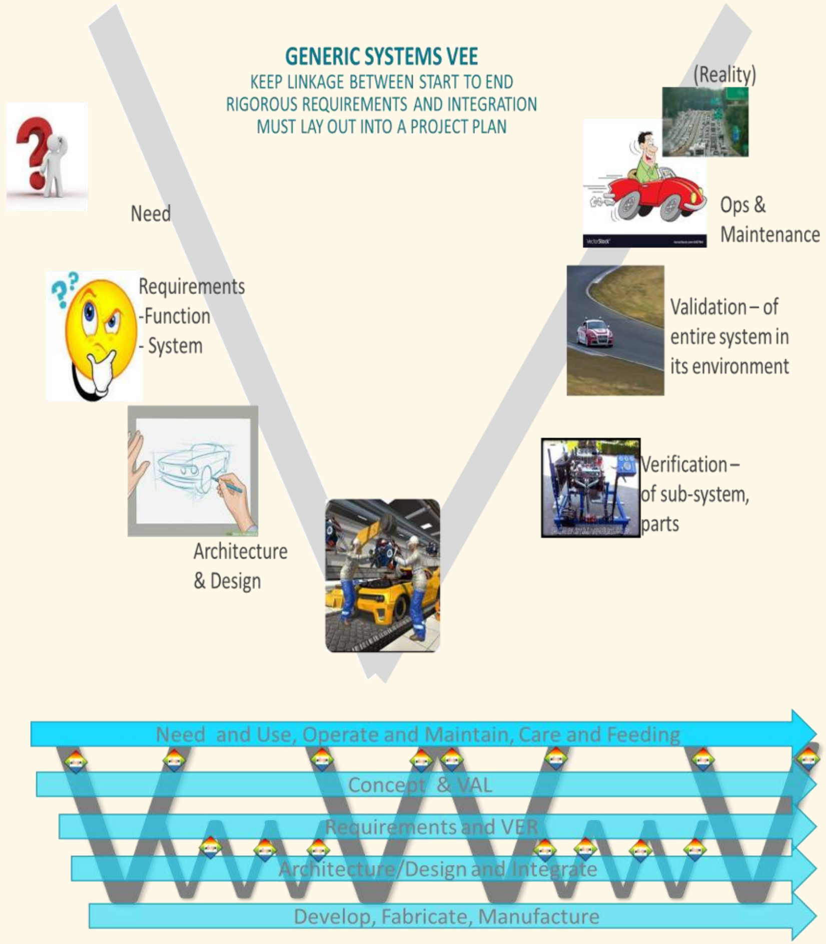
<u>Traits</u>

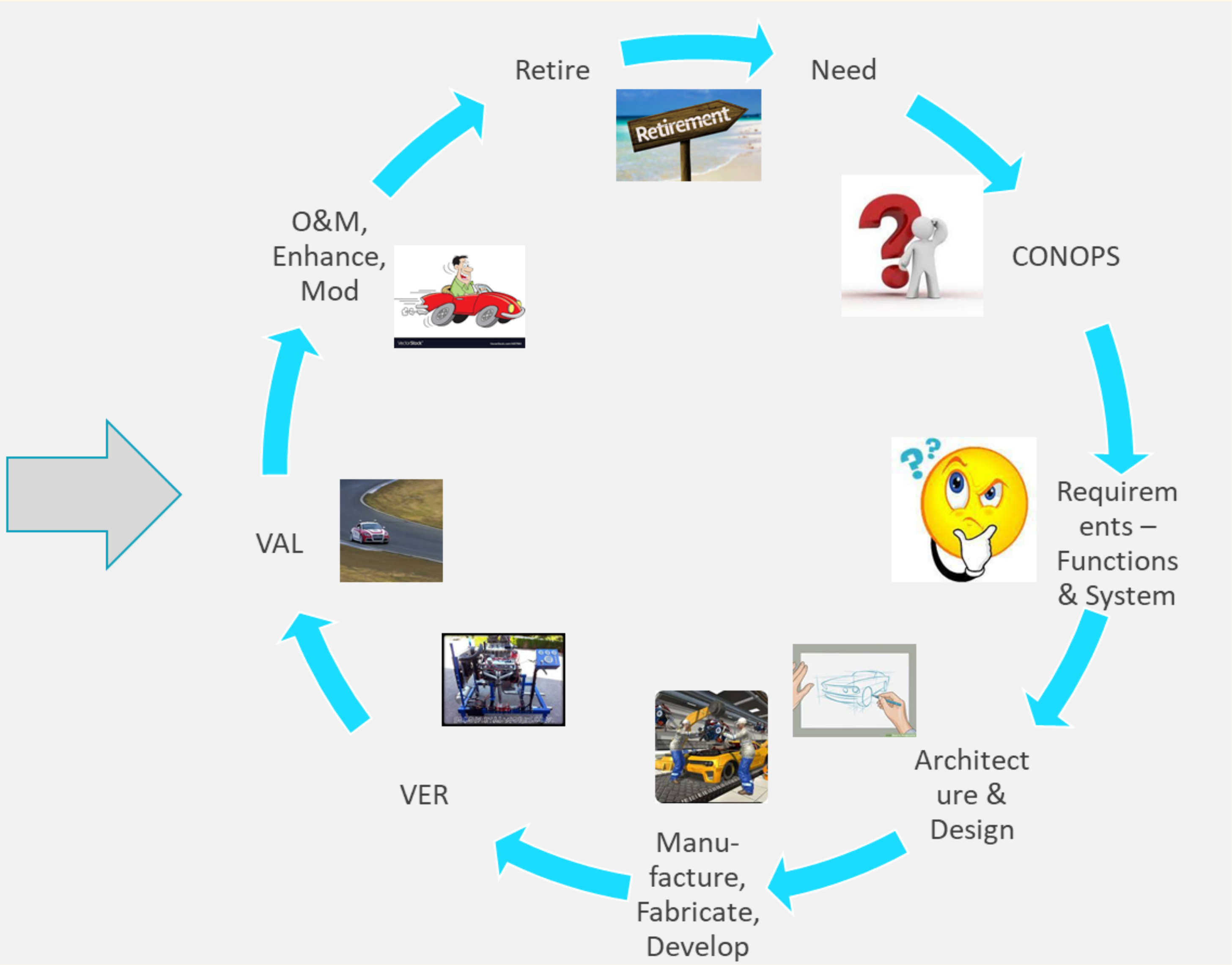
- Same Idea as Regular Systems Vee
- Note the Smaller System Vees for each Version leading to Integration, and then VAL (Ops Test)
- Good Conceptual Understanding of How a System Is Cared and Maintained over time
- Collect More Requirements, Correct Defects but ensure a **deliberate and controlled** method before using
- Note the well-placed way points – the Decision Gates
- Slow and Deliberate and 'feels' like agile due to continual versions
- Has Robust Recursion and Iteration
- Example: Software based Weapons and Aerospace Systems

THE SYSTEM VEE IS LAID OUT FLAT AND LOOKS MORE LIKE A PROJECT PLAN



THE SYSTEMS VEE BECOMES A CYCLE OF EVER IMPROVING CHANGES

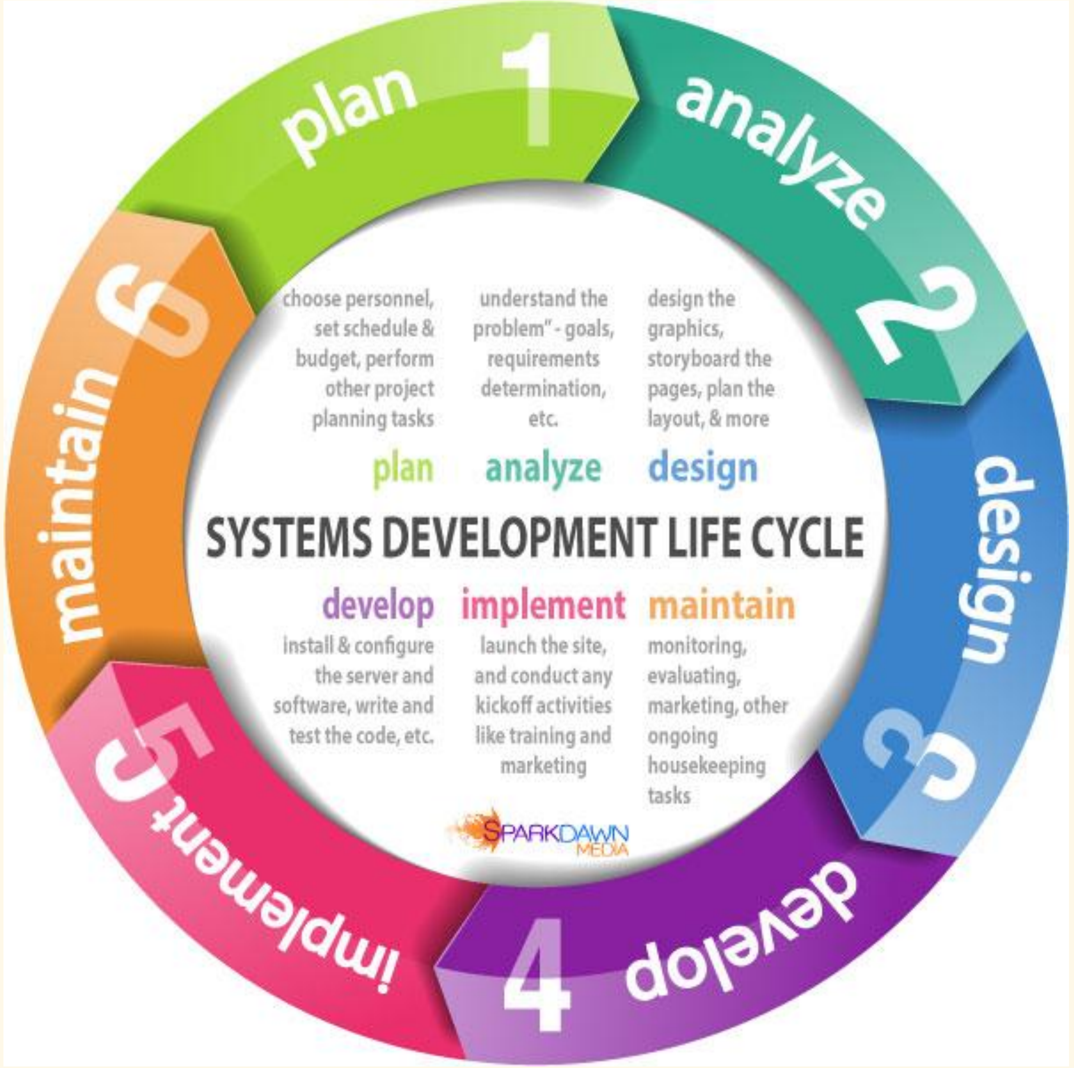




CIRCULAR MODEL

<u>Traits</u>

- Existing systems to build on to, and augment
- Continual Improvement
- Recursion – obtaining and working on details is OK
- Iterative – Repeating previous step as more is learned is OK
- Rapid insert new requirements
- Simple and Nice Visually – allows failure early
- But doesn't lend itself to a plan with a schedule, easier to lose requirements trace
- Start with a linear Model but once initial system is delivered can use a Circular
- Example: Applications and Services (with few stockholders – private equity)



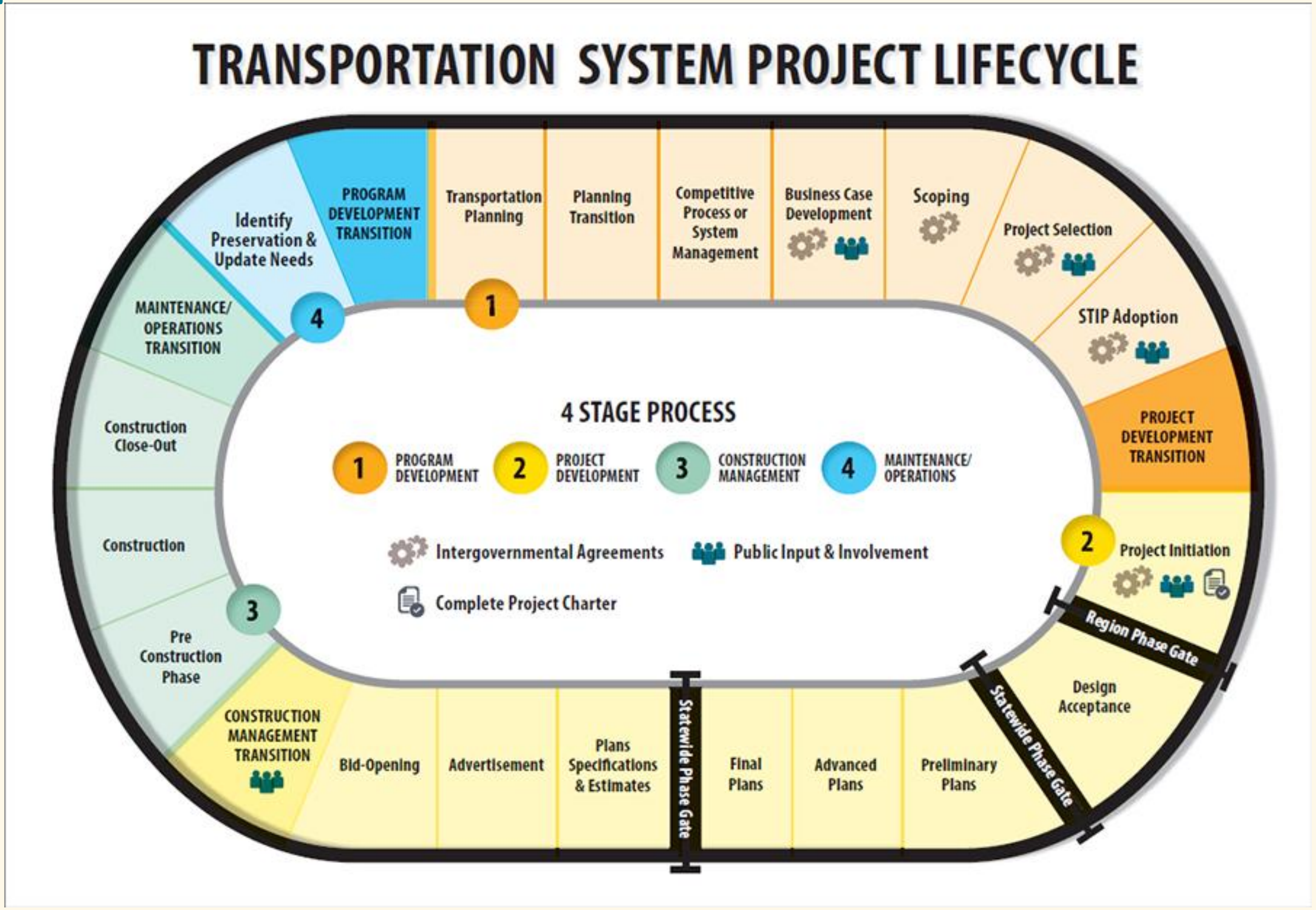
OVAL MODEL

<u>Traits</u>

Similar to Circular …

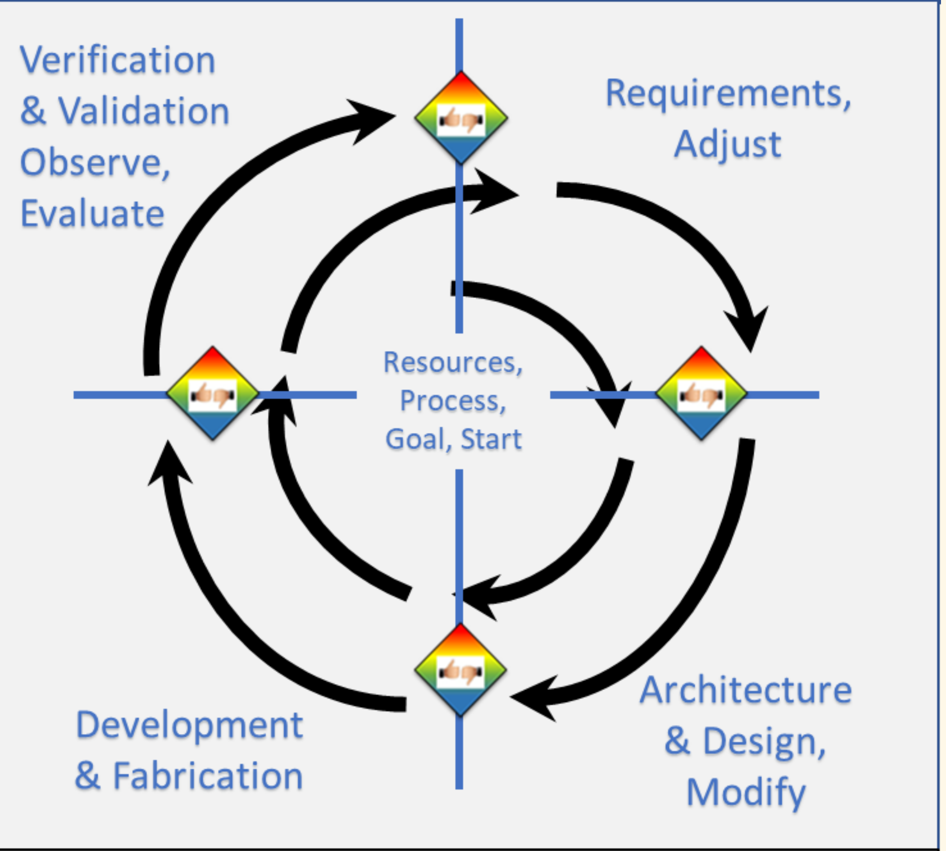
- Existing systems to augment
- Continual Improvement
- Rapid insert new requirements
- Easy to Understand – Nice Visually
- Recursion and Iterative is OK

- Plenty of Gates to Control Risk
- Lots of Up Front Work before executing
- Good for public works and costly projects/systems
- But doesn't lend itself to a plan with a schedule
- Easier to lose Requirements Trace
- Example: Roads, Bridges, Tunnels, Railroads, Airports



FORM OF CIRCULAR DEVELOPMENT MODEL - SPIRAL

Figure 2-7 Spiral – Develop, Assess, Adjust



Each Spiral is an Ever Maturing Capability with Continuous Risk Assessment, Control, Adjustments

<u>Traits</u>

- 'I'll know it what I need when I see it!'
- Early and progressive deliver of some capability
- User Says 'Now Try This!'

<u>PROS</u>

- Volatile/Dynamic Baselines
- Well Managed Risk
- User Participation (or a proxy)
- User has Ownership

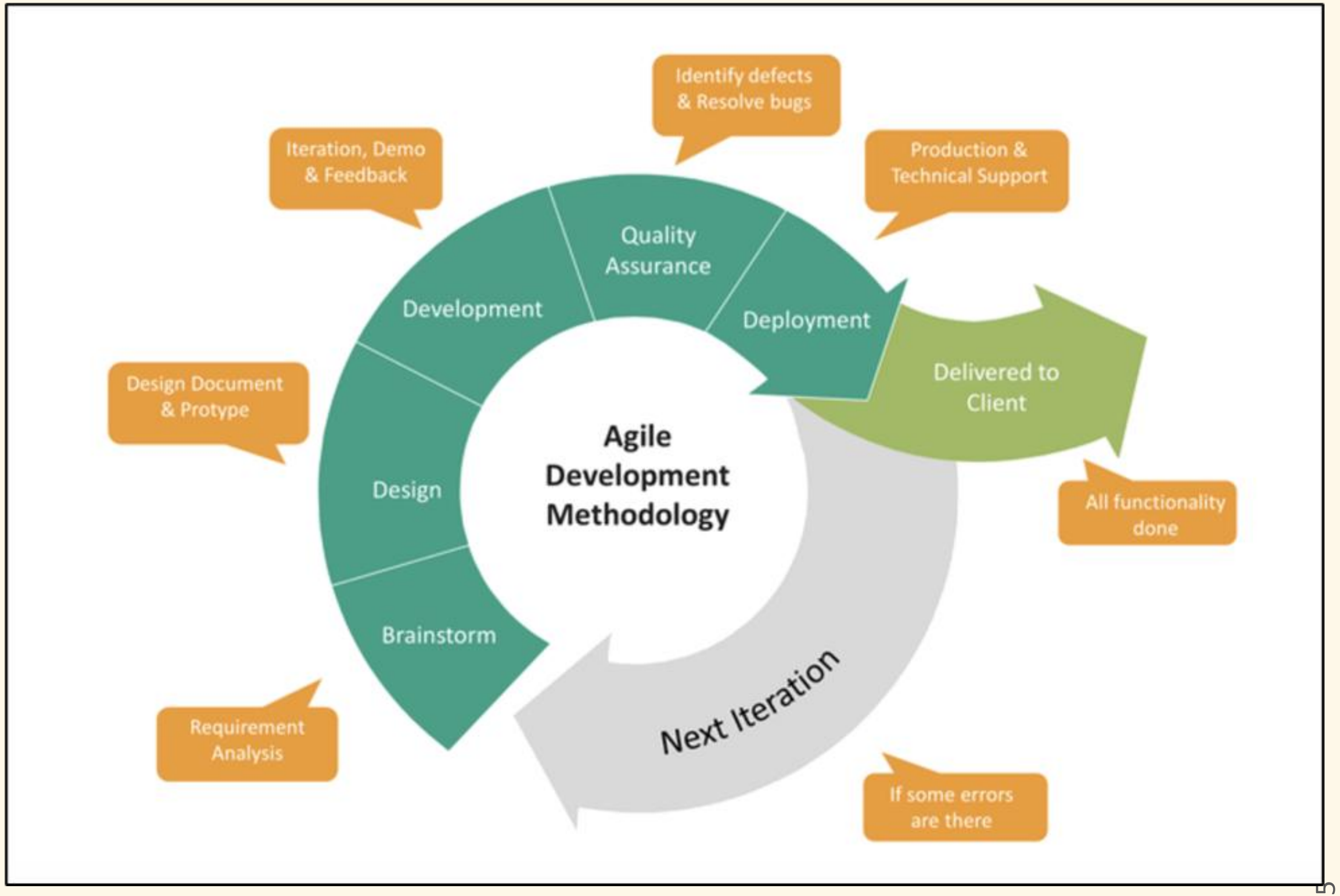
<u>CONS</u>

- PM – Less Visibility into Cost, Schedule
- Continues Subject Matter Expert
- Many steps and prone to delay
- When Finished?

FORM OF CIRCULAR DEVELOPMENT MODEL - AGILE

<u>Traits</u>

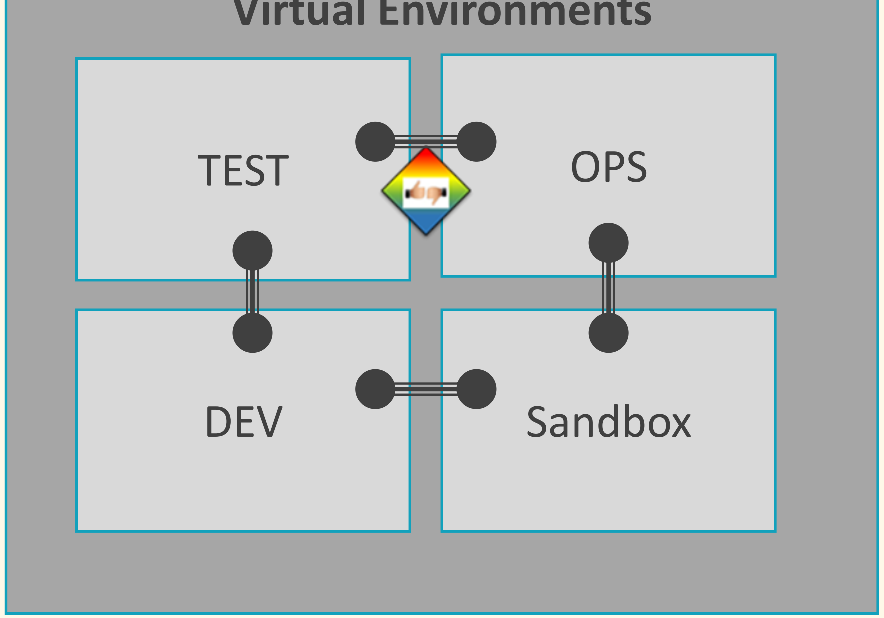
- Hundreds of Flavors and Types of Agile
- Considers the Triad plus Risk – but Schedule is 'Time Boxed'
- Requirements are 'User Stories' – 'As A User I want to ….'
- Sprints & Increments - backlog of User Stories
- Iteration often realized in subsequent Sprint
- Recursion – expected and needed but may be the reason to defer to subsequent Increment
- Weighted Points given to User Stories
- Completion of Points indicates Velocity
- Roles:
  - Scrum Master
  - Product Owner
  - Developers
- Agile – not suitable for every project or system – still need a plan, requirements, an Architecture, VER & VAL, Risk Management, to Manage Configuration



DEV – OPS (**DEV**ELOPMENT – **OP**ERATION**S**) VIRTUALIZATION OF ENVIRONMENTS – GOES HAND IN HAND WITH AGILE

- **OPS** – strict controlled access, only approved changes, business/ mission environment
- **TEST** – controlled access, simulate OPS, approved changes made
- **DEV** – control is less, dynamic environment, document to allow rebuild/replicate
- **Sandbox** (Experimental) – can be called anything but virtually no control – try out new ideas
- Once Approved Build is ‘**Promoted’** to next environment
- Then start all over again with next set of requirements/user stories

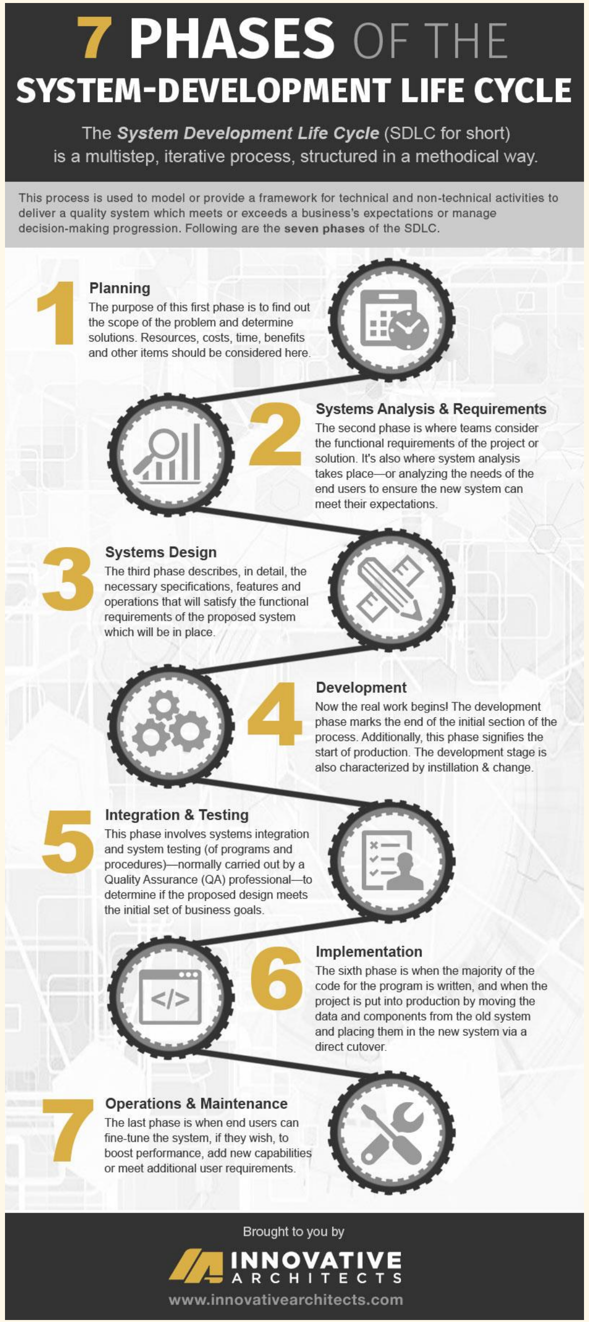
Figure 19



SOFTWARE WITH PHASES

<u>Traits</u>

- Linear and Sequential
- Seven (7) Distinct Phases with Decision Gates
- When Delivered start over again with new system
- Higher Risk, Defined Static Requirements
- But Still need a schedule, with phases, approvals
- Example: Cloud Applications and Services, Electronic Medical Records, Mil C3

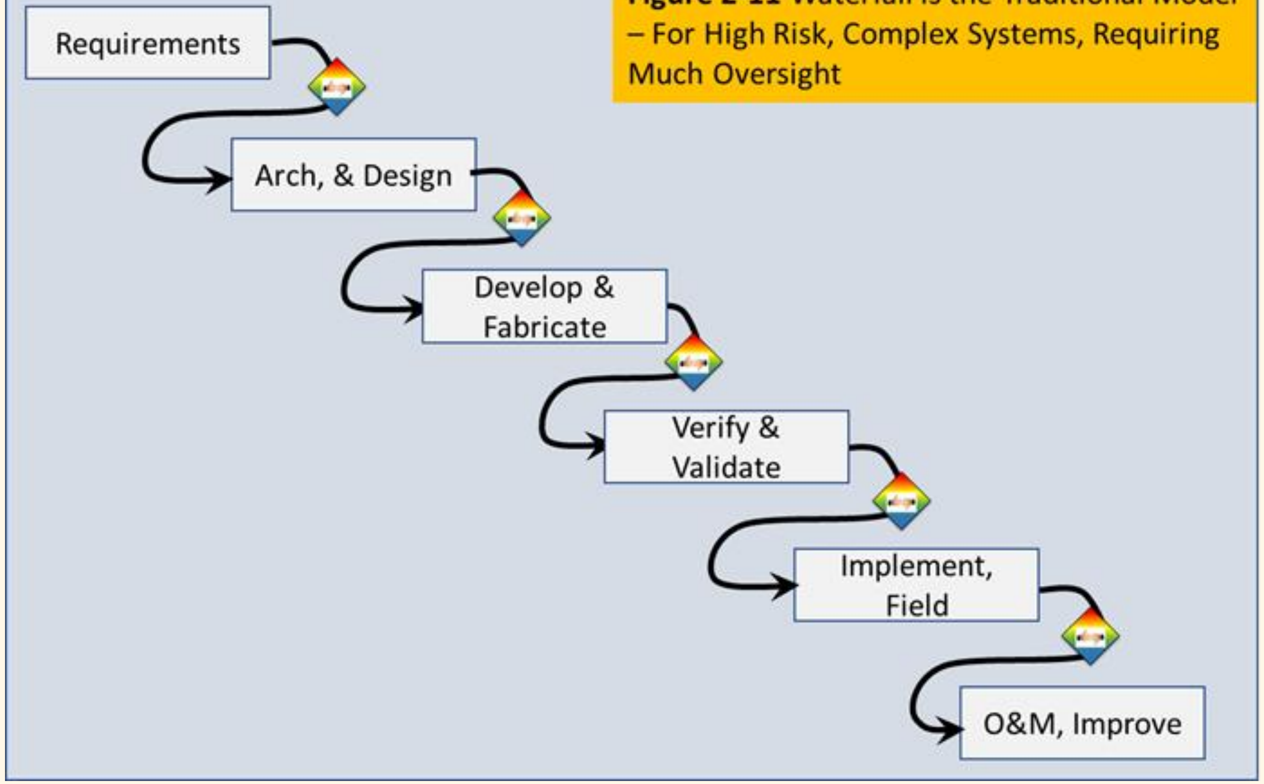


WATERFALLS MODEL

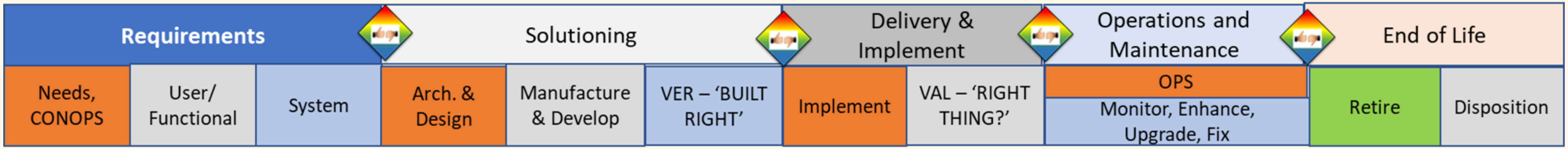
Traits

- Traditional and maybe the original model
- Phases and Incremental -
    - perform work/pause/check/approve ~
- Simple Visual but can become complex
- For Systems and Projects Prone to High Risk
- Well managed but fairly static requirements
- Prone to Delays due to sequential and rigid Decision Gates
- Starting to Look Like a Plan, Phases, Schedule
- Examples: Weapon Systems, Aerospace

Figure 2-11 Waterfall is the Traditional Model – For High Risk, Complex Systems, Requiring Much Oversight



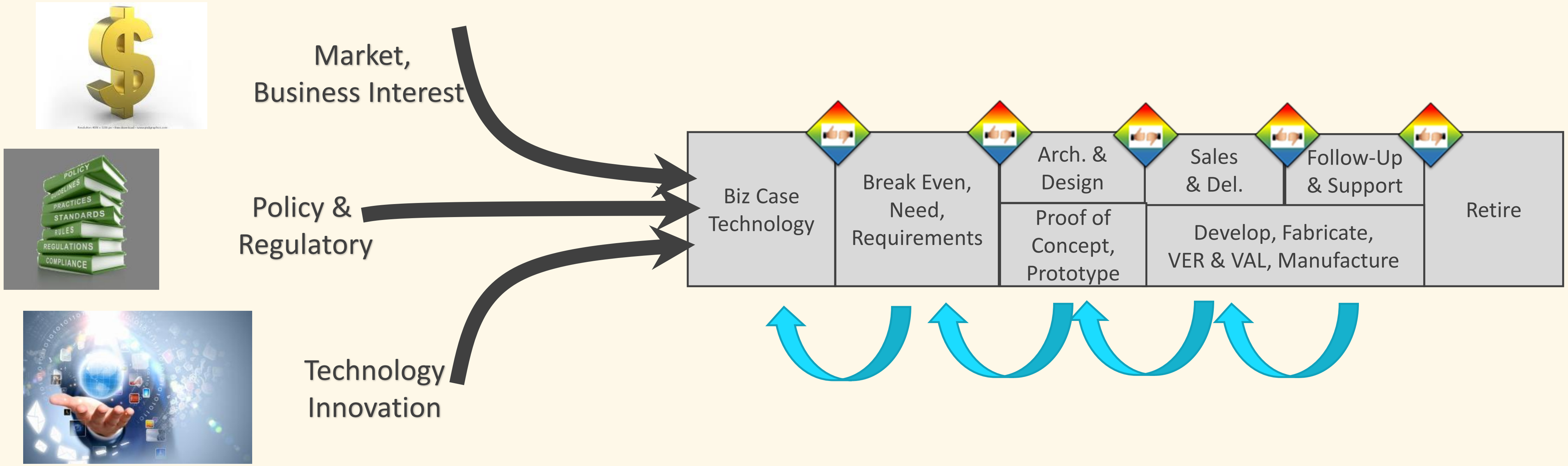
LINEAR



Traits

- linear and Sequential – Traditional – Easy to transition from a Systems Vee
- Five (5) Distinct Phases with Decision Gates
- Breaking Phases into Smaller and Manageable Pieces
- Higher Risk, Defined Static Requirements
- Allow for Recursion (Detail) and Iteration (Repeating a Process)
- Once initial systems is delivered may transition into a Cycle
- Example: C3 Systems, Aerospace, New and Large IT

BUSINESS TECHNOLOGY MODEL



<u>Traits</u>

- Business – Technology Model – accounts for the Market Demands (will they buy it?), Policy (is it permitted?), and Technology (is it possible?)
- For large, years long systems and capabilities
- Multiple integrated disciplines
- Keep your eye on the business case – if the business case is no longer viable – change the system or kill it
- Easy to transform to a schedule
- Examples: Commercial Aircraft, 5G, Data Centers

DEVELOPMENT METHOD – PARALLEL OR CONCURRENT – MULTIPLE AND CONCURRENT SERIAL EVENTS

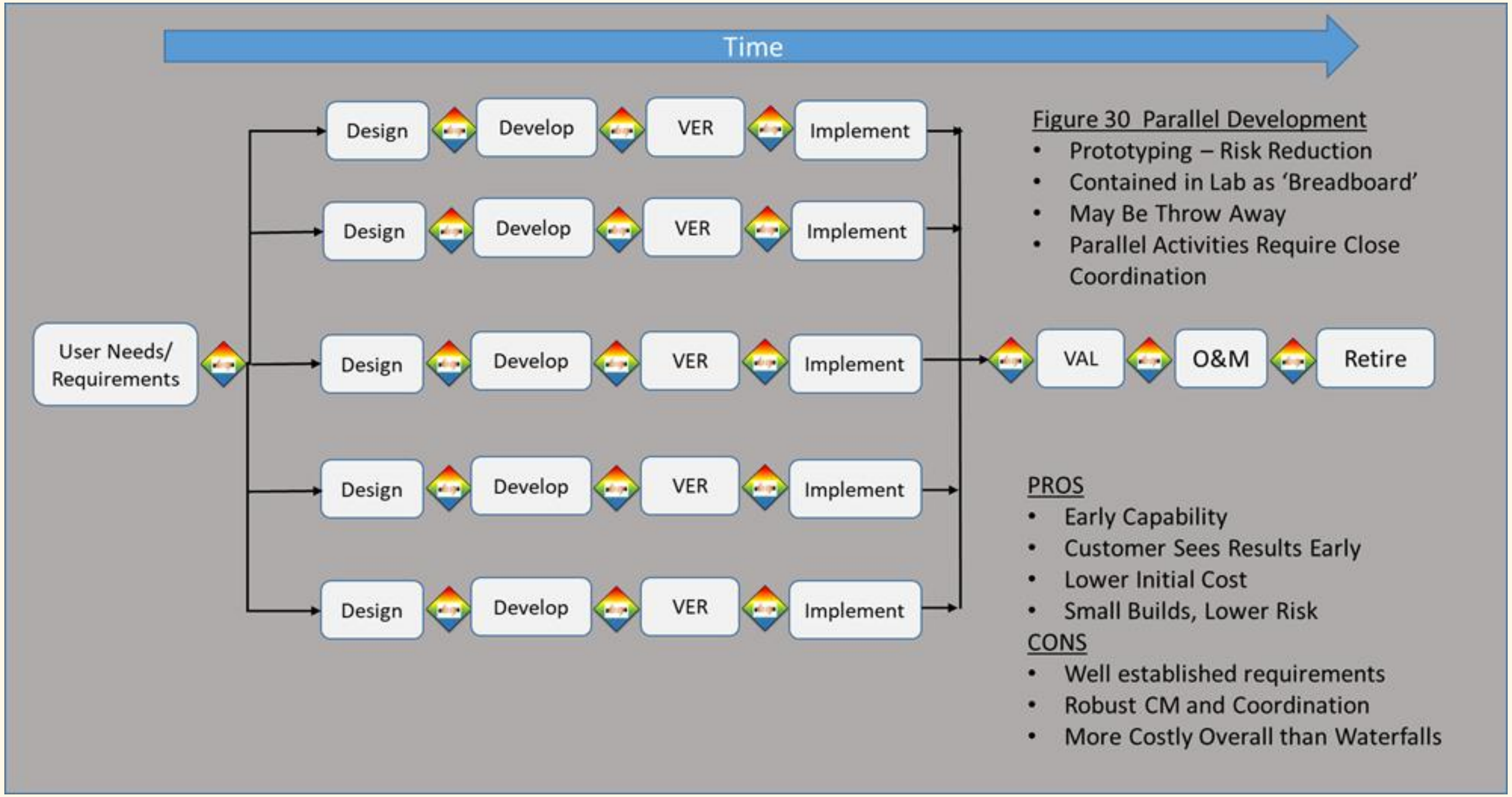


Figure 30 Parallel Development

- Prototyping – Risk Reduction
- Contained in Lab as 'Breadboard'
- May Be Throw Away
- Parallel Activities Require Close Coordination

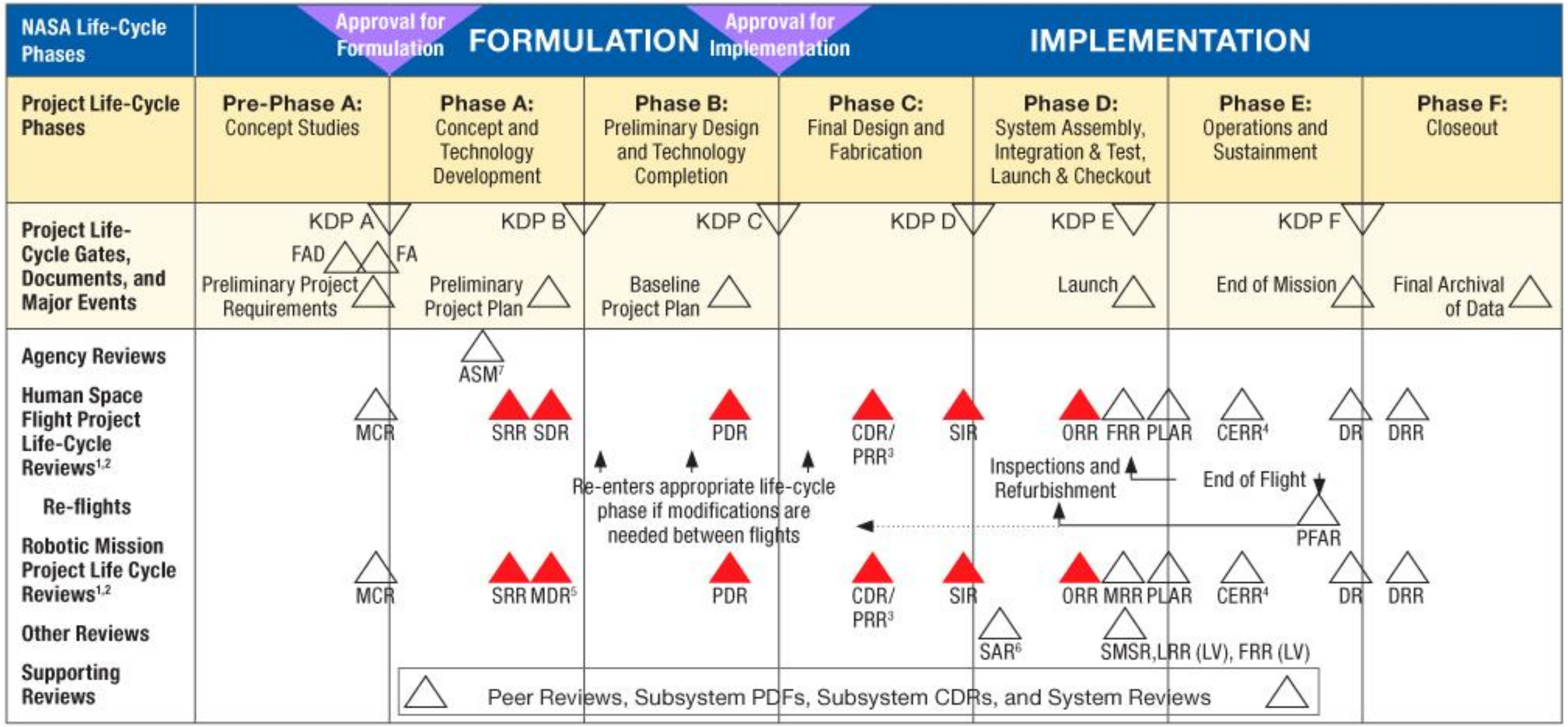
PROS

- Early Capability
- Customer Sees Results Early
- Lower Initial Cost
- Small Builds, Lower Risk

CONS

- Well established requirements
- Robust CM and Coordination
- More Costly Overall than Waterfalls

LIFECYCLE (3) – **NASA** – RISK, INTEGRATION, GOVERNANCE, APPROVALS - FRAMEWORK



FOOTNOTES

1. Flexibility is allowed as to the timing, number, and content of reviews as long as the equivalent information is provided at each KDP and the approach is fully documented in the Project Plan.
2. Life-cycle review objectives and expected maturity states for these reviews and the attendant KDPs are contained in Table 2-5 and Appendix D Table D-3 of this handbook
3. PRR is needed only when there are multiple copies of systems. It does not require an SRB. Timing is notional.
4. CERRs are established at the discretion of program .
5. For robotic missions, the SRR and the MDR may be combined.
6. SAR generally applies to human space flight.
7. Timing of the ASM is determined by the MDAA. It may take place at any time during Phase A.

▲ Red triangles represent life-cycle reviews that require SRBs. The Decision Authority, Administrator, MDAA, or Center Director may request the SRB to conduct other reviews.

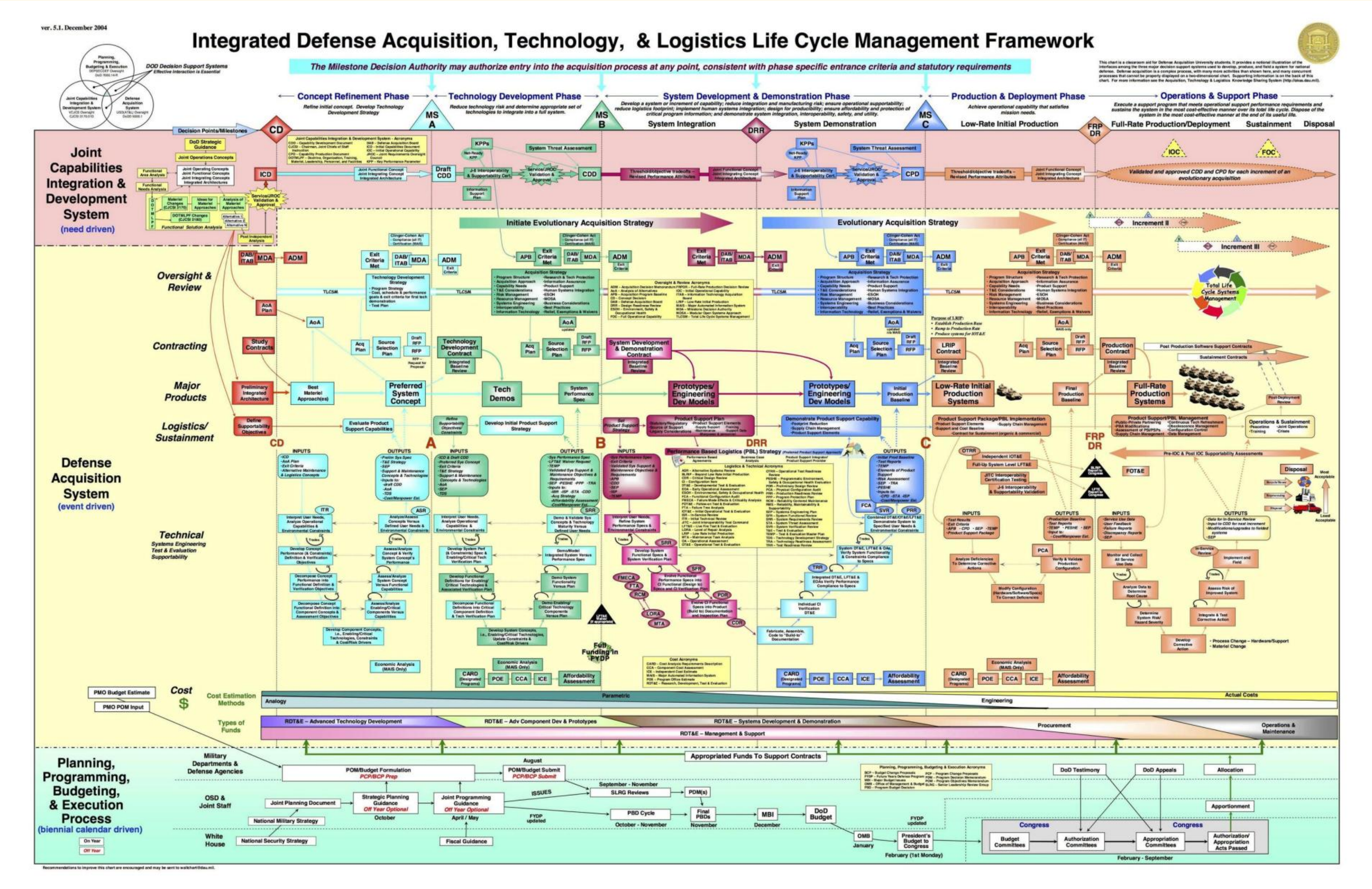
ACRONYMS

ASM – Acquisition Strategy Meeting
CDR – Critical Design Review
CERR – Critical Events Readiness Review
DR – Decommissioning Review
DRR – Disposal Readiness Review
FA – Formulation Agreement
FAD – Formulation Authorization Document
FRR – Flight Readiness Review
KDP – Key Decision Point
LRR – Launch Readiness Review
LV – Launch Vehicle
MCR – Mission Concept Review
MDR – Mission Definition Review
MRR – Mission Readiness Review
ORR – Operational Readiness Review
PDR – Preliminary Design Review
PFAR – Post-Flight Assessment Review
PLAR – Post-Launch Assessment Review
PRR – Production Readiness Review
SAR – System Acceptance Review
SDR – System Definition Review
SIR – System Integration Review
SMSR – Safety and Mission Success Review
SRB – Standing Review Board
SRR – System Requirements Review

FIGURE 3.0-1 NASA Space Flight Project Life Cycle from NPR 7120.5E

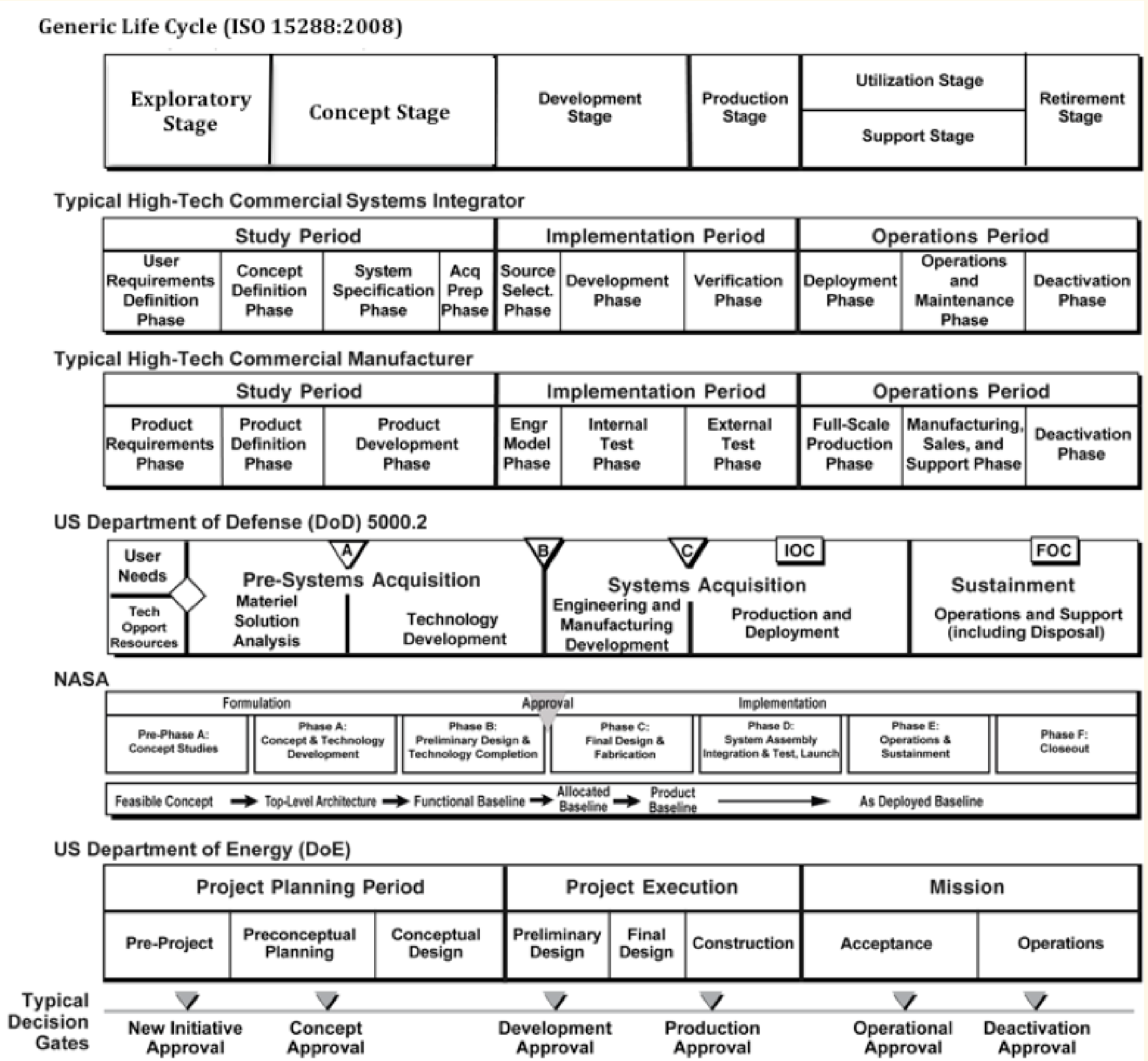
LIFECYCLE (4) – DEPARTMENT OF DEFENSE – FRAMEWORK

MUCH RISK, GREAT COST, PROCESS AND SYSTEMS INTEGRATION, OVERSIGHT – SELECT THE APPLICABLE PROCESSES



https://www.dau.edu/training/career-development/logistics/blog/Updated-DoD-Acquisition-Life-Cycle-Wall-Chart

MISCELLANEOUS LIFECYCLES



SECTION 2 SUMMARY

- ‘Unconventional’ Approach – blend of Lifecycle Model and Development Model
- The two have blended and are hard to separate
- Lifecyle – from concept, design and realization, supporting to retirement like in real life
- Use or create one that works, that people understand
- System Vee – Conceptual Model liking the SE Activities over time
- Circular
- Linear
- Business Lifecycle
- DOD and NASA – very complex

SECTION 2 LIFECYCLE – KNOWLEDGE CHECK

- Define Methodology
- Define Model
- What is a Framework?
- What are the traits of the System Vee?
- What Methodology has a set of repeatable activities, and continual improvement?
- What Methodology is most suitable for high risk systems, where there are continual milestones before proceeding
- Why is it important to use a Lifecycle that people understand?

SECTION 2 LIFECYCLE – KNOWLEDGE CHECK

- Define Methodology – Practices, techniques, procedures, and rules – opportunies for innovation or change.
- Define Model – to form or plan the activities to realize the system as envisioned. Simulation goes hand in hand with Modeling
- What is a Framework? – loosely defined guidelines, with boundaries, on how a set of activities are performed
- What are the traits of the System Vee? – a Vee where the beginning of the lifecycle links that desired end stage, or interim activities. Allows continual linkage to the CONOPS and Requirements
- What Methodology has a set of repeatable activates, and continual improvement? Agile, circular, Spiral
- What Methodology is most suitable for high risk systems, where there are continual milestones before proceeding. Waterfalls, Spiral with Prototyping, even Agile if done with robust Risk Management
- Why is it important to use a Lifecycle that people understand? It's a tool used to communicate – actual activities completed and planned activities and identifies roles and a system/project road map. Will give confidence to stakeholders

SECTION 2 KNOWLEDGE VERIFICATION – MATCH COLUMN ON LEFT TO COLUMN ON RIGHT

- A. Systems Requirements
- B. Emergent Behavior
- C. Gate Reviews
- D. Development
- E. Did We Build It Right?
- F. Did We Build the Right Thing?
- G. Functional Requirements

- 1. Verification
- 2. Greater than the Sum of its Parts
- 3. For the Architect / Developer
- 4. Validation
- 5. ‘Bend Metal’
- 6. Way Points
- 7. From the Operator/User

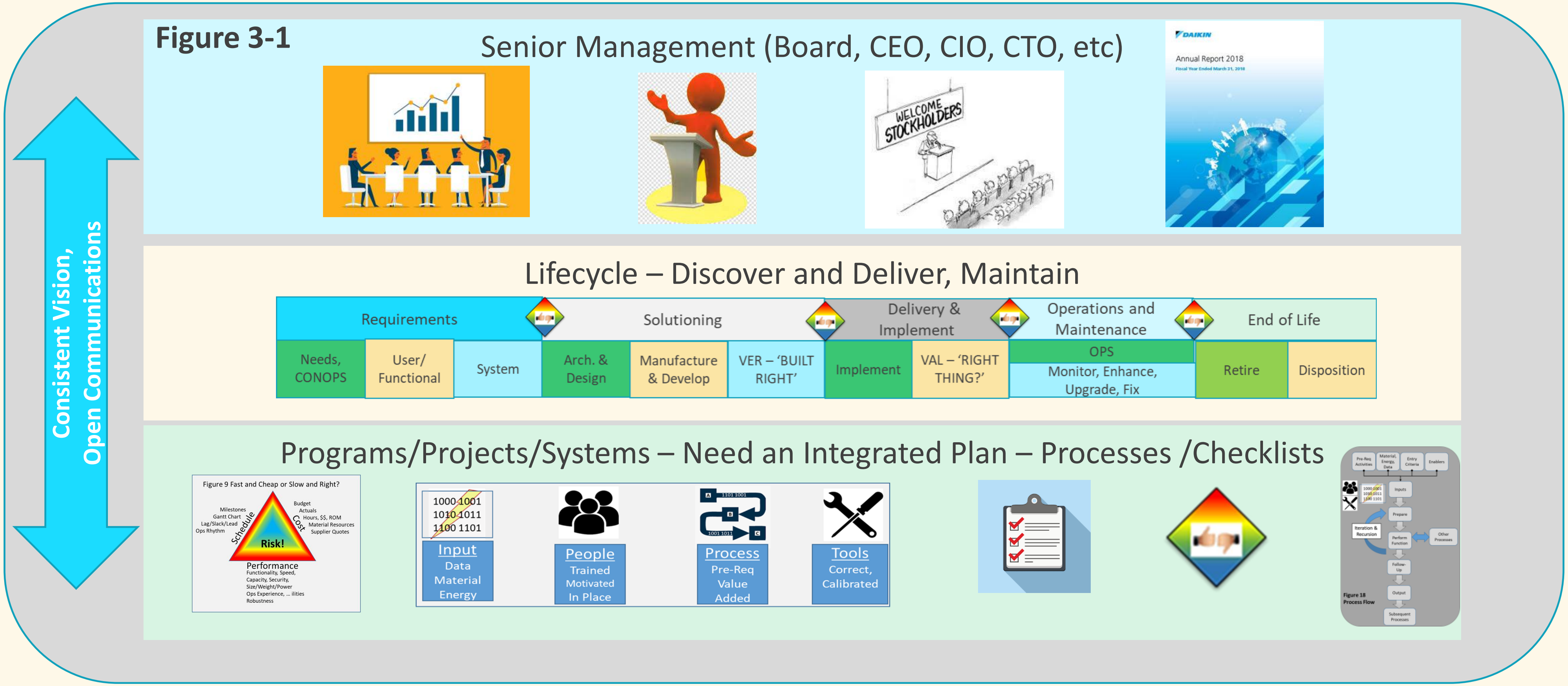
- A. Systems Requirements – 3
- B. Emergent Behavior -2
- C. Gate Reviews - 6
- D. Development 5.
- E. Did We Build It Right? - 1.
- F. Did We Build the Right Thing? - 4
- G. Functional Requirements - 7.

RECOMMENED CASE STUDIES

SECTION 3 PROCESSES

- Recap –
  - Section 0 The Foundation
  - Section 1 What, Who, Why
  - Section 2 Lifecycle – Methodologies, Models, Frameworks
- Section 3 Processes and Decision Gates
  - What is a Process and a Decision Gate - and Some Examples
  - Elaboration of some processes
    - **Section 3.0 Introduction to Processes and Gate Approvals**
    - Section 3.1 Decision Gates
    - Section 3.1 Process Descriptions to end of Requirements
    - Section 3.2 CM/DM to Fitting Processes Together – to end

LIFECYCLE AND PROCESS



PROCESSES AND DECISION GATES

- Why Processes – Creatures of Habit, Focus less on routine and more on problem solving
- Synchronize and Coordinate plans and activities, so your Output is to the expectation of the recipients Input
- Consistent to Roles and Responsibilities – to discover improvements, gaps, overlaps, improvements
- Systems Engineers and Project Managers – perform only those that add value.
- Typical and Most Important Processes

Manage Project	Manage Schedule
Manage Risk	Manage Requirements
Architecting	VER & VAL
Technical Exchange Meeting (TEM)	Conduct Trade Study
Manage Defects	Project Management Review (PMR)
Integration	Transitions
Configuration Management/Document Management	Agreements (MOA, Service Levels)

Thank You INCOSE!

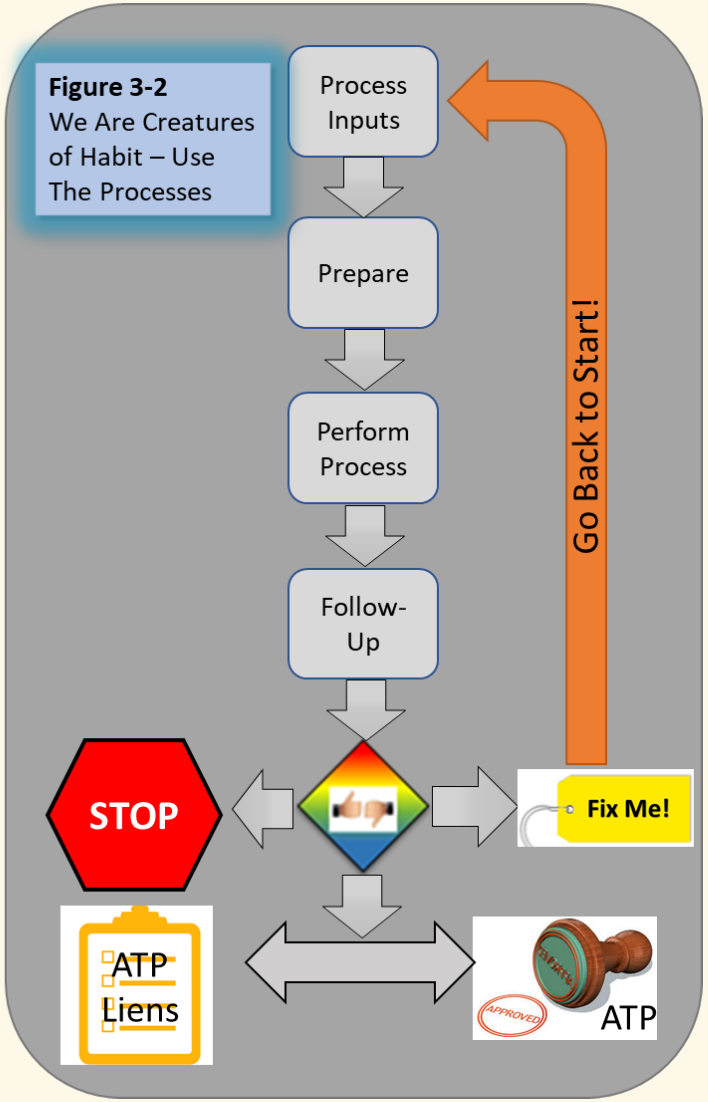
PROCESSES

- Usually more lengthy and detailed, specific than what I've described
- Based on business model, type system, criticality
- Most Processes are Company Proprietary – a company asset
    - Due to the years of experience and knowledge –
    - Don't need just anyone looking at their process and unjustly criticizing – if they know nothing about the business
    - Process are based on company policy, which in turn is based on Law and Regulatory
    - 

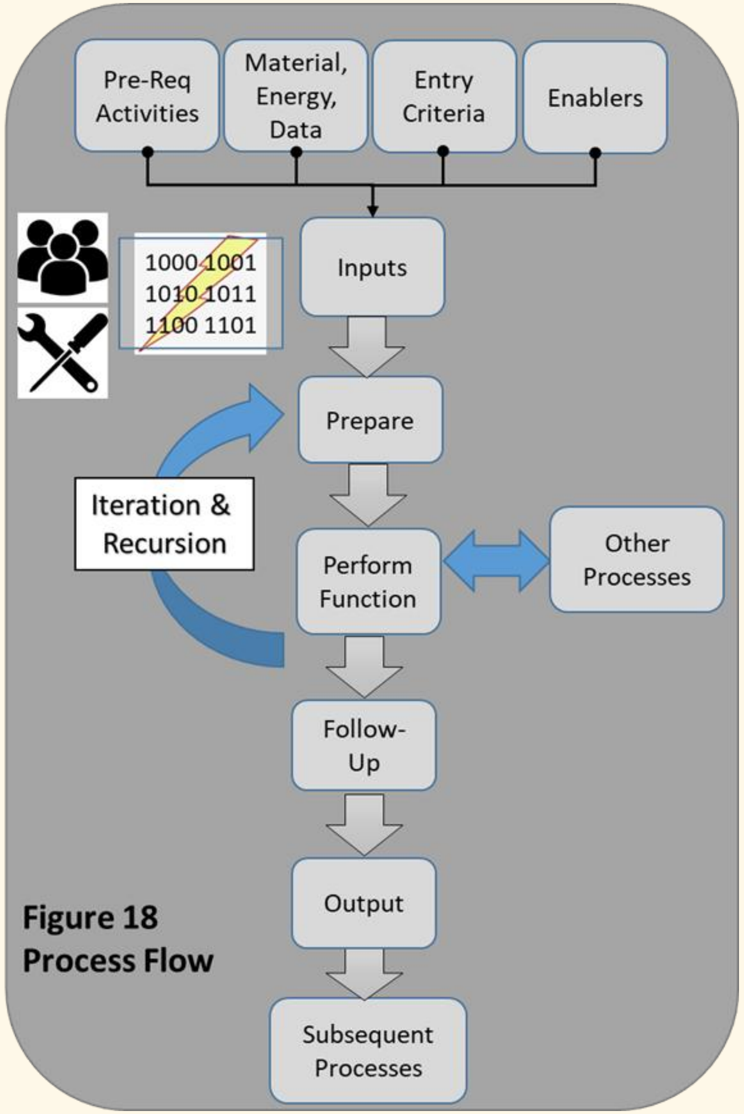
<u>If a person 'follows the process' they'll likely NOT be held liable for anything that went wrong</u>
- Processes can be, and are audited by key customers
- Processes should be based on Industry Best Practices, and have been audited by a third party – to ensure compliance to ISO or CMMi-
- ISO Certified vs ISO Compliant – no such criteria as ISO Compliant

A PROCESS
TO COMPLETE AN ACTIVITY OR TO OBTAIN APPROVAL

=



Gate Review - Gain Approval



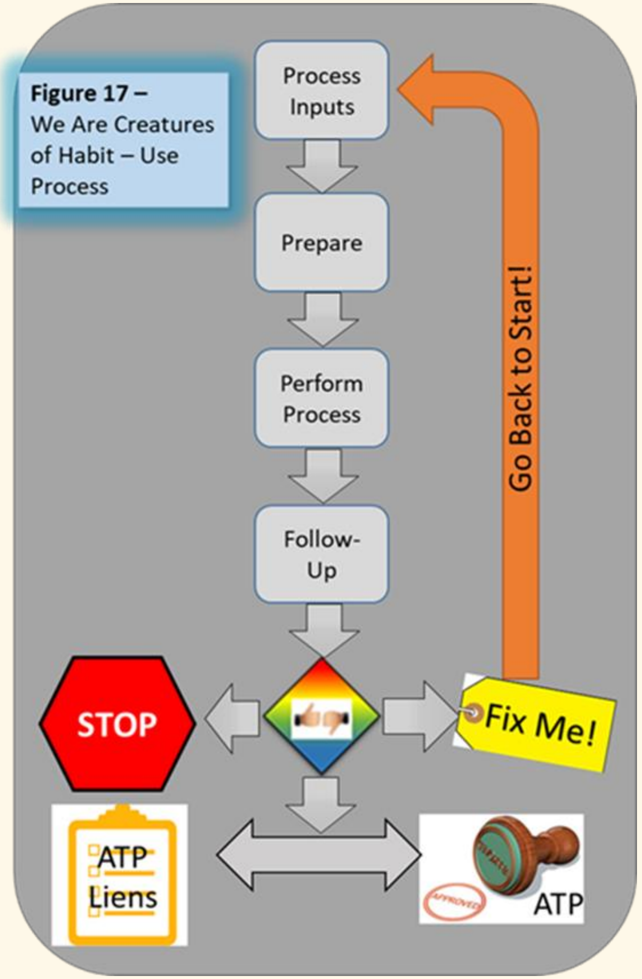
A Generic Process – Complete an Activity

SECTION 3 PROCESSES

- Section 3.0 Introduction to Processes and Gate Approvals
- **Section 3.1.1 Decision Gates**
- Section 3.1.2 A Process – Use a Common Game Plan
- Section 3.2 CM/DM to Fitting Processes Together – to end

DECISION GATE

- Prepare early – send any documents out in advance to pre-read, ask questions – and visit people face to face before
- **Grease It** if needed – learn any issues the approver may have
- Test Projector, Computers, VTC, any other equipment in advance
- Have an Agenda, Record Results to include Actions, Risks
- Four Outcomes 1) **STOP** the Program (business case changed, risk is too great) 2) Has Some problems that need fix – not approved but rework required – return later 3) Approval to Proceed – have actions, issues, liens to fix but conditional approval is granted 4) ATP – with no liens
- **The worst outcome is to approve a system that is not ready – resulting later, in catastrophic failure**



Use a Gate Review to **Help Strengthen Consensus**

IMPORTANT DECISION GATES

- Approval to Proceed
  - Initiate Project – approval to start a project, given budget, need, milestones
  - Requirements Review – periodic and careful review of requirements, traceability, need
  - Preliminary Design Review (PDR) – conception design acceptable, alternatives
  - Critical Design Review (CDR) – based on PDR – detail, meet requirements, performance, ready to 'light the fuse'. This is called Critical for a reason
  - Test Readiness Review (TRR) – prepared to conduct VER or VAL – test plans, test cases, VER and VAL Process ready
  - Operational Readiness Review (ORR) – given VAL results, assess if system ready to go to ops – and how will this be done?
- Reviews
  - Project Management Review (PMR) – usually month review of Cost, Schedule, Performance, Risks, Staffing, Contractors, Plan Ahead, Action Items
- Use only those you need, combine if it makes sense but to don't streamline too quickly

NOT ALL GATE REVIEWS ARE THE SAME (1)

Table 2 Gate Reviews and Their Focus – Progressively Detailed Through Each Gate

Approval Gate	Gate 1 Project Start	Gate 2 Pre-Design Review (PDR)	Gate 3 Critical Design Review (CDR)	Gate 4 Fielding Review	Gate 5 Retire
Scope	Consistent to Charter	Trade Studies Experience Outsourcing	Agreements Made, MOA	Agreements Ready	Dependent Users
Functionality	Defined Need, Regulatory, Data Sensitivity	Deliver Need, Compliance, Data, … ilities, Performance, KPIs,	Requirement, … ilities, Compliance, Approval, Drawings as Evidence	User Exp, Need, Compliance, Resolved, Interfaces	Connectivity, Environmental, Security
Cost	Notional Budget, Availability of Resources	Phased Budget, Availability of Resources, Licenses	EAC/ETC Prepared Committed Resources,	Travel, Export Issues, Shipping, Transition Plan	Packing and Shipping, Destruction
Schedule	Milestones	Gantt Chart, Development Method, Draft Project Plan/SE Plan	Detailed Gantt, Development Method, Project Plan/SE Plan	Delivery Schedule, Transition Plan	Transfer or Decommission vs 'pulling the plug'
Process	Known Processes	Processes Identified and Documented, in Use, Checklists	Process Results Identified, Completed Checklists	Mature, re0used processes, not the first time this is done, checklists	Retention of needed documentation – as evidence of compliance

Gate 1 – not detailed, notional.
Gate 2 PDR is more rigid and formal
Gate 3 CDR is before lots of resources are approved
Gate 4 Very Formal, ready to go operational?

NOT ALL GATE REVIEWS ARE THE SAME (2)

Approval Gate	**Gate 1 Project Start**	**Gate 2 Pre-Design Review (PDR)**	**Gate 3 Critical Design Review (CDR)**	**Gate 4 Fielding Review**	**Gate 5 Retire**
Communication	Listening to Stakeholders/ Users, Who needs to Know What, Establish Project Comms.	Baseline designated artifacts in a known location, Informing and listening to Team and Stakeholders, 'Grease Communication', Outline Education and Training, Procedures	Baseline artifacts, control changes, informing and listening, 'Greasing', Prepare and Conduct Training, Write Procedures	Notify external stakeholder and dependencies, daily and frequent updates, Inform of Changers/ Awareness, Train and Educate	Capture old documentation, inform stakeholders and dependent users
Procurement	Supplier Feasibility, Non-Disclosure Agreement Alternative	Most suited – cheap and slow, or expensive and fast, trusting provider, ISO, sub-contracts,	Back Up Procurement if needed, open comms with existing sub-contractor, SLA, ISO Audits	Travel, availability 24/7, requirements flow down, communication	Not becoming complacent, Complete and Close Contracts and Invoicing, terminate licenses, close agreements

NOT ALL GATE REVIEWS ARE THE SAME (3)

Approval Gate	Gate 1 Project Start	Gate 2 Pre-Design Review (PDR)	Gate 3 Critical Design Review (CDR)	Gate 4 Fielding Review	Gate 5 Retire
Demand of System	Is there a need? Others providers	Still a need? Viable substitute	Capacity to meet demand, cost recovery (profit). Checklists	Check each user site for demand changes. Adjust performance as needed. Follow Up entering O&M.	Verify demand is transferred seamlessly, inform users and other SE.
External Issues	Political, Environ-mental, Availability of Material, Bandwidth, utilities, survey external threats	Plan for external Issues, high level identification of issues, utilities redundancy (BU Power), slack, margin, insurance	Detail of issue planning, include into Risk Plan, observe external condition	External networks or data available, last check before fielding. Travel, Import/ Export or data restrictions	Treaties and Alliances for data, service. Tacit agreements.

The Key to a successful Gate Review are the quality of the activity leading up to it.

- Section 3.0 Introduction to Processes and Gate Approvals
- Section 3.1.1 Decision Gates
- **Section 3.1.2 A Process – Use a Common Game Plan**
- Section 3.2 CM/DM to Fitting Processes Together – to end

A PROCESS

- A coherent, repeatable and documented set of steps …
  - Inputs, Outputs, Steps, Prerequisite Activities Entry Criteria
  - Embedded Process
  - Iteration and Recursion
- Organizations that deliver systems must have documented and approved processes
- Need Tools, Skill Level and Type of People
- People will learn and know the process well enough they don't need to continually refer but ….
  - Have them read and review periodically
  - Don't let them go stale or out of date
- **Think People and Tools** – supporting the Process

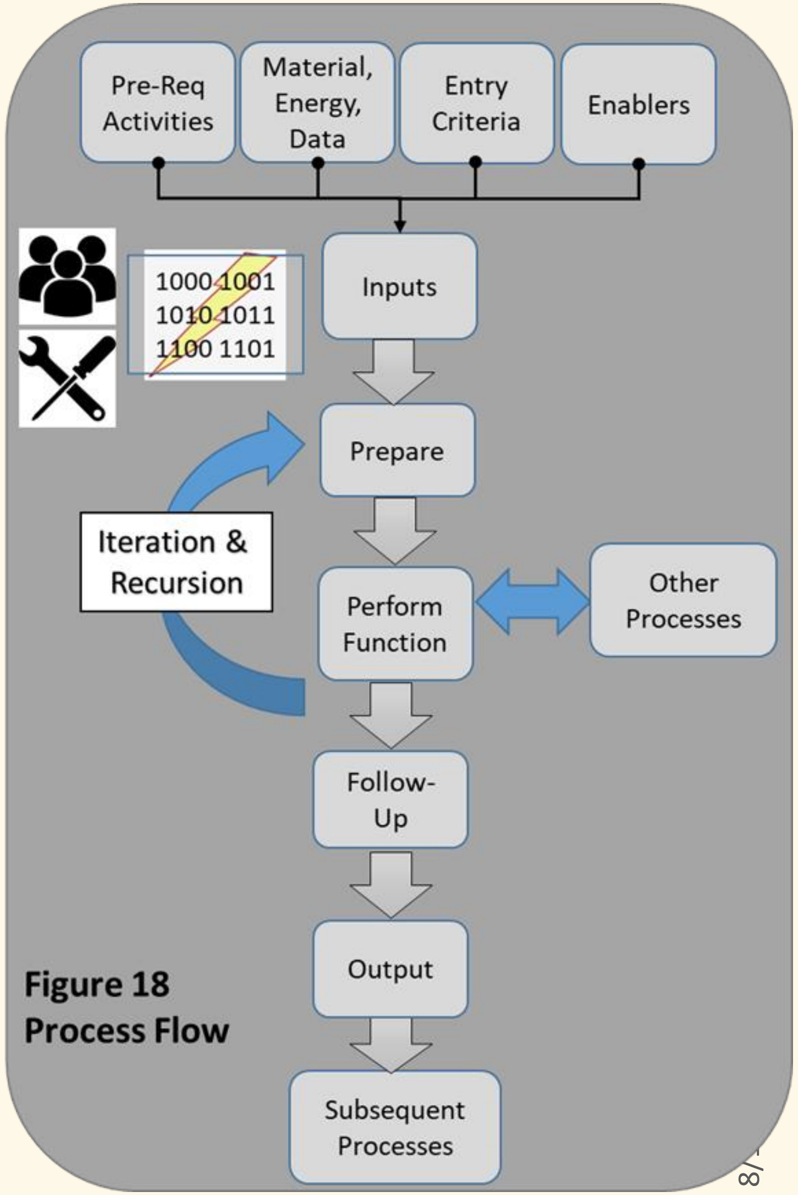


Figure 18
Process Flow

TYPICAL PROCESSES (2)

- The PM and SE must have an **Evaluate and Create** level of understanding of process – **(see Bloom's Taxonomy)**
  - Know when to do Other Activities – Transitions, Opportunities and Issues, Preliminary Design Review, Critical Design Review
  - To Create or Modify Processes – to ensure they add value
  - Use On-Line Processes with URLs built in – printed version may be out of date

Most of the Process Structure is based on Industry Standards and Guidance – especially INCOSE and PMI

3.1.3 to 3.1.15 Documented Processes, & 3.2 Other Processes

EXAMPLES DOCUMENTED PROCESSES

- A Template of a Documented Process
- Manage Project
- Manage Schedule
- Risk Issue, & Opportunity Management (RIO)
- Requirements Management
- Configuration Management/Document Management (CM/DM)
- Integration
- Create Architecture (Architecting)
- Conduct Trade Study
- Manage Defects and Improvements
- Verification and Validation (VER&VAL) (V&V)
- Conduct Project Management Review
- Conduct Technical Exchange Meeting
- Other Processes

These are Generic and May Require More Detail for your Organization

PROCESS TEMPLATE

<table>
<tr><td>Name: Process Name – Make it Active/Verbal</td><td>Purpose: Why are You Performing</td></tr>
<tr><td>Inputs and Prerequisites: Items, Processes, Activities, Documents Used in the Processes, or as Inputs</td><td>Information: General Tips and Information, any special safety, health, security considerations. Can embed a illustrated process flow.
Process Templates May have a Block Flow Diagram</td></tr>
<tr><td>Related Processes/Tasks: Proceeding, Sub-sequent, Parallel and embedded activities and processes</td><td>Tools: Software, Hardware, Facilities </td></tr>
<tr><td colspan="2">Roles & Responsibilities 
• Project Manager – will always be responsible for the success of the project
• Systems Engineer: Compliance and Participation in the process – that only needed processes are performed
• Specialist: Requirements Manager, CM/DM Manager, Subject Matter Experts (SME), etc</td></tr>
<tr><td colspan="2">Process Activities
1. General Sequence of Activities – starting with plan and process, include training and awareness so team know what is being done, their roles
2. Describe Each Step – briefly – it is assumed those performing the task are trained to the needed level.
3. Document What was done, any success, failures, suggested improvements – so others will know esp. those performing subsequent processes.
4. Finish – Close files, put away tools, clean up, report any issues, etc</td></tr>
<tr><td>Output and Results: Expected Results, Artifacts – if not completed the process may not be finished. Some outputs depend on the degree the process was performed. Not every artifact may be needed.</td><td>Change Log (only approved changes)
Original Version 04/01/20
Change 1 : added step 4 in Activities ; 04/11/20
<u>Printed Versions Are UnControlled</u></td></tr>
</table>

Name: Manage Project

Purpose: Plan, Organize, Control, Direct Resources to achieve the expected outcome

Inputs and Prerequisites: Approved Project,

Information: Ongoing Process – start to end of project, many embedded activities; PPT: Cost/Schedule/ Performance & Risk

Related Processes/Tasks: Manage Risk, VER & VAL, Manage Schedule, Manage Cost, Create Architecture

Tools: MS Word, Excel, Tableau, Project Now!

Roles & Responsibilities

- **Project Manager** – success of the project – prepare and execute project plan, build team, awareness and training, deliverables
- **Systems Engineer:** Establish & Comply to SE Management Plan, CM/DM, Risk Management, Verify Enterprise Architecture, Solutioning, Quality, Continual Improvement

Process Activities

1. **Plan Project** – PM creates project plan – include Roles & Responsibilities, Scope, Assumptions, Schedule/Ops Rhythm, Budget, Deliverables. Open and assign charge numbers, establish and enforce agreements with contractors and providers.
2. **Technical Activities** – Systems Engineer – creates and approves SE Management Plan, other Systems Plans, sets up and manages systems tool (CM/DM and Requirements, Architecture Tools), build team
3. **Ops Rhythm** – establish an ongoing patter of meetings/activities – use checklists, action items, PMR, TEMs to monitor and adjust resources to ensure quality and completion
4. **Record Activities** – after an activity or process is completed, record in minutes, reports, test results, and place into CM/DM as proof of compliance.
5. **Communicate and Awareness** – vertical and horizontal - formal – via staff meeting, TEMS, PMRs, few emails. Informal by face to face 'walk around'. Create and deliver formal training for major changes or new tools.
6. **Close of Finish Project** – phase down activities, reward/help team find new jobs, transition needed activities/interfaces, ensure all deliverables delivered, customer feedback obtained, close contracts, KM

Output and Results: Managed tasks, completed tasks accepted by recipient, functioning and tested system

Change Log

Original Version 04/01/20

Change 1 : added step 4 in Activities ; 04/11/20

3.1.4. Manage Project

Se

MANAGE PROJECT - ELABORATION

- Example is 'epic' – should have a process to 'Start Project', 'Manage Project', 'Conduct PMR', etc
  - Break Process Down into
    - Start Project
    - Management Project
    - Manage Schedule
    - Other processes as needed
  - Remember
    - People, Process, Tools
    - Cost, Schedule, Performance and Risk
- The Project Manager and the Systems Engineer are 'joined at the hip' and can almost read each others' minds – trust is very important
- Try to keep 'disagreements between you' behind closed doors

Name: Manage Schedule	**Purpose:** Establish and Maintain a manageable project schedule.
Inputs and Prerequisites: Approved Project, Project Plan, Work Breakdown Structure, Deliverables, Roles /Responsibilities, SOW	**Information:** Schedule may be Gantt Charts, Ops Rhythm, or Increments (for Agile).
Related Processes/Tasks: Manage Project, PMR	**Tools:** MS Project, Primavera

Roles & Responsibilities

- **Project Manager** – success of the project - on time, as expected and to cost
- **Scheduler** – create and update schedule based on system and project needs. Identify and help treat risks
- **Systems Engineer:** Ensure tasks/ deliverables in the project plan. Lead / organize differing tasks to ensure integration, identify risks.

Process Activities

1. **Establish Schedule** – Schedule from solicited project (from proposal) is updated and Rebaselined from the proposal.
2. **Schedule Review Work to Perform** – SOW, Work Breakdown Structure (WBS), List of Deliverables and with project/systems team determines sequence and dependencies, durations – drafts schedule
3. **Lean Out Schedule** – first run at schedule is incomplete – will take iteration and recursion to 'streamline' and ensure completion.
4. **Baseline Schedule** – accepted and used to track work to be done, and work done – and for risk.
5. **Update Schedule** – make approved changes, show completion.

Output and Results: Baselined Schedule or Ops Rhythm, Updated Schedule, Completed Scheduled Tasks.	**Change Log** Original Version 04/01/20 Change 1 : added step 4 in Activities ; 04/11/20

MANAGE SCHEDULE (1) - ELABORATION

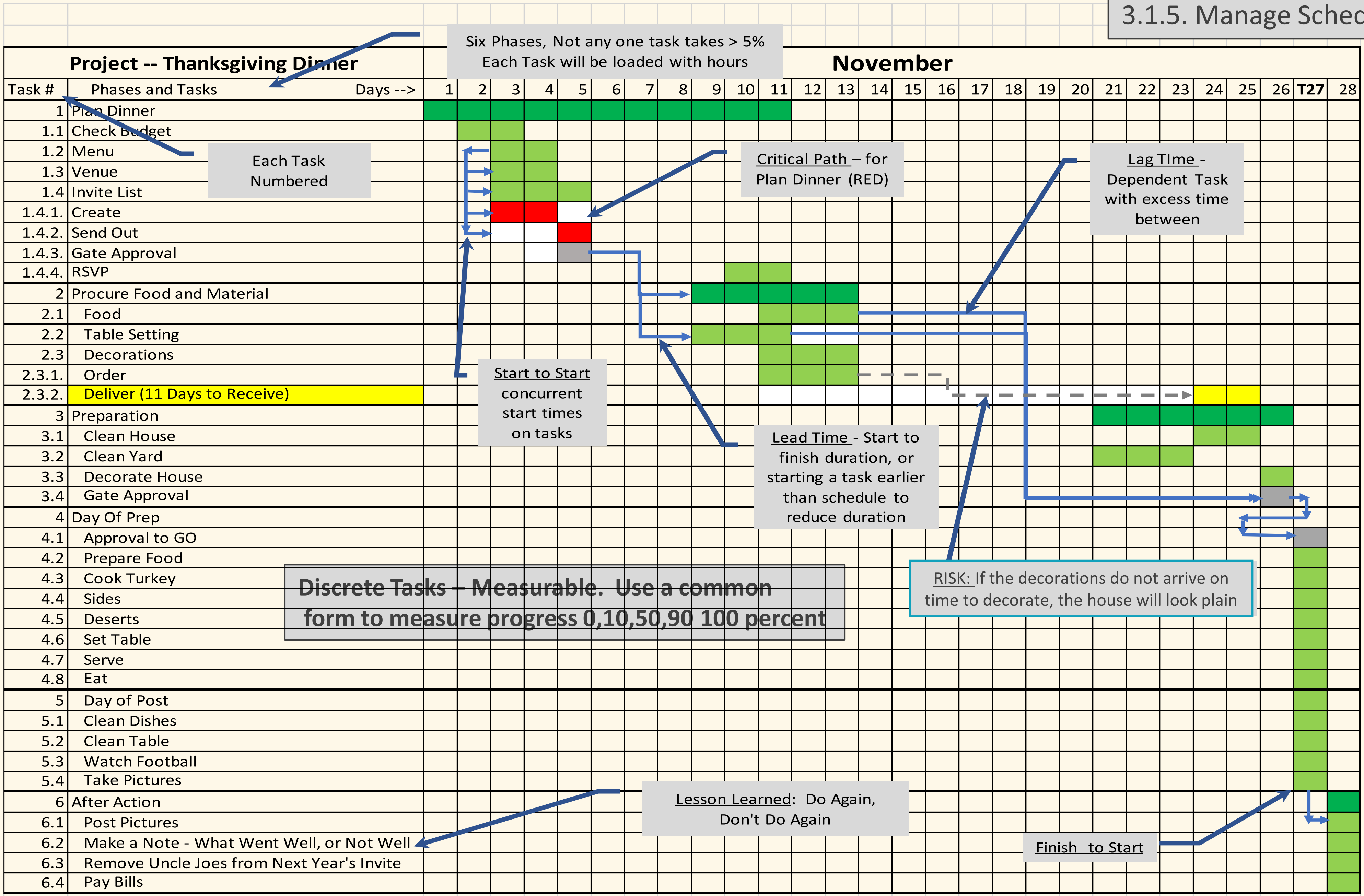
- Break down to smaller tasks – if needed down to level four or five
  - Each task no more then 10 percent of the work – one miscalculation of a task completion could jeopardize the entire project
  - In fact, each task should be about 5 percent or less of entire project
  - Discrete Task – measurable and assign to a task owner
  - Establish and Show Dependencies – Finish to Start, Start to Start, End to End
- How I measure Progress on a Schedule
  - 0 percent – those assigned the tasks has not yet read it
  - 10 percent – the person assigned the task has read it and understands it
  - 50 percent – in process
  - 90 percent – task is finished – but not yet accepted by the recipient
  - 100 percent – task is completed and accepted by the recipient

MANAGE SCHEDULE (2) - ELABORATION

- Scheduling Terms
  - Critical Path – a delay in any one series of tasks – that are dependent on one another will cause a delay in the project finish date
  - Know Critical Path, **and also secondary Critical Path**
  - Lag Time – delay of starting the next, dependent task (breathing room)
  - Float or Slack – time that can be taken without cause a project delay
  - Event Driven – tasks are triggers (to start) when an external and undetermined event occurs. You're not in control
  - For Agile – Velocity – amount of work measured in points that are completed
- Schedule can almost be an art form – a good SE or PM will know where to find more time that others done see – if needed
- Be careful – if using a Gantt Chart during a meeting – people will spend time trying to follow it, figure it out, make mistakes and waste time
- Consider showing only the top level tasks and easy on the dependencies
- This may be a case where too much communication can be a problem

MANAGE SCHEDULE (3) - ELABORATION - GANTT CHART – THIS NOT USED IN AGILE

3.1.5. Manage Schedule



Name: Manage Risk	**Purpose:** Capture and Manage, preventing it from becoming an Issue
Inputs and Prerequisites: Risk Plan, Risk Repository, Trained Team	**Information:** Risk is an event, should it occur, will cause damage to the project or the system.
Related Processes/Tasks: Start Project, Manage Project, Manage Requirements, KM	**Tools:** Spreadsheet, PowerPoint

Roles & Responsibilities

Project Manager – success of the project

Systems Engineer: create a design and solution of acceptable risk – balances Cost, Scheduler, Performance – to deliver the desired capability

Risk Manager: capture, track and manage risk, compliance to the process

Enterprise Risk Manager: Document Risk and Mitigation and make this knowledge available to those in the organization.

Risk Owner: Risk is assigned to someone to managed to lower the Likelihood or the Impact.

Process Activities

1. **Identify Risks** – all stakeholders can identify risk, and report to the SE, PM or Risk Manager
2. **Document Risks** – Risk Manager vets risk (if accepted) into a common risk leger and chart.
3. **Risk Monitoring** – Risk Manager provides oversight of the process, ensure compliance.
4. **Risk Closure -** Risk Owner create plan, execute plan to reduce impact and likelihood of risk.

Output and Results: Documented Risk, Risk Treatment Plan, Triggers, Closed Risk, Identified Issues	**Change Log** Original Version 04/01/20 Change 1 : added step 4 in Activities ; 04/11/20

RISK MANAGEMENT (1) – ELABORATION

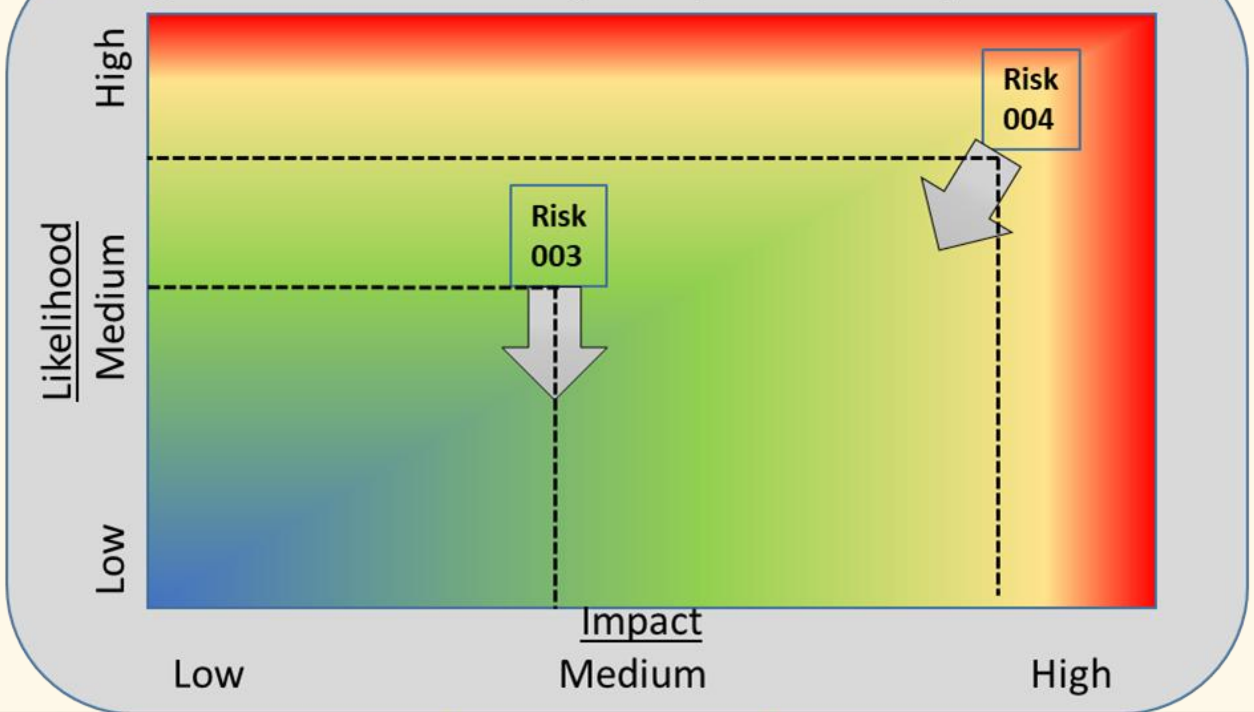
- Make sure it not nebulous or epic 'solve world hunger' or its just something hard to do – these are not risks
  - Make it an If…., Then … Statement.
  - Assign an Owner
- Risk Treatment Plan – Owner will create this – the step of actions to eliminate the risk
- Risk Monitoring – Periodic Oversite of risk – status and follow up. Have a monthly or weekly Risk Meeting
- Risk Closure – when declared the risk is low or eliminated – by the PM or Quality

RISK MANAGEMENT (2) - ELABORATION

- Risk **Impact** and **Likelihood**
  - High – mission critical impact, severely degrading or halting the mission
  - Medium – hamper or degrade the mission, may temporarily stop the mission or system. Will increase cost or delay schedule
  - Low – inconvenience, will not stop mission, moderate delay or cost increase – no use to manage Low Risk Tasks or Systems – these may be infinite
- Risk Treatment – how to manage it
  - Owner creates a plan to manage the risk, and work the tasks to reduce risk
  - Transfer – third party, sub-contractor, a SME
  - Mitigate – decrease the impact if the risk should occur
  - Avoid – Change the plan – extend the schedule, add resources
  - Accept – commonly done on Low Risk – if for systems that are poorly managed
- Document the Risk
  - Ledger – with unique number, name, risk statement, owner, treatment plan, and update
  - Risk Cube – X Coordinate (Impact) and Y Coordinate (Likelihood)

RISK MANAGEMENT (3) - ELABORATION

Figure 17 Risk Cube and Ledger – Pay Me Now or Pay Me Later



Risk	Name	Owner	Mitigation
003		Peter Drucker	
If … Then Risk Statement will go here			
004		Warren Buffett	
If …. Then risk statement will go here			

RISK MANAGEMENT (4) - ELABORATION

- Common Activities to Reduce Risk
  - Management Reserve (MR) or Budget Reserve – set budget aside to pay for added resources needed to avoid a risk – this seldom works – few want to leave no $$ on the table
  - Prototyping – determine if desired capability is achievable (very common in DOD)
  - Ver and Val – higher risk projects require more rigor in V&V
  - Component Testing – Critical System – check all components. If less critical – test first one in each lot. May also test samples.
  - Back Up or Contingency Plan – have another contractor in the wings, Have a Quick Reaction Capability (that may cost more) but there if needed.
  - Similar Systems – learn from someone else's pain
  - Architectural designs – that load balance, have back ups, multiple networks
  - Back Up Plans – with spares, back up date and user accounts,
  - Continuity of Operations (COOP) – back up site ready when needed – Hot Back Up – Seamless transition (runs in parallel) vs Warm Back Up – take some time to bring up. Practice and train for these.
  - Pre-Staging Equipment and Spares – esp for equipment that is prone to failure
  - Work at Risk – perform work, procure material before the payment is approved to cut down on critical path. Working at Risk usually requires corporate approval.

RISK MANAGEMENT (5) – ELABORATION

- Risk, **Issue and Opportunity**
  - **Issue** – when a Risk has been realized – requires immediate action and triage
  - **Opportunity** – an unplanned, often fleeting event that if captured and managed well, will decrease cost or schedule, improve performance
    - Opportunity is not without risk – if you fail to capture the opportunity you may experience risk
    - Manage the Opportunity
  - Risk is OK – and expected or we would not make progress –
    - Think of our ancestors, and the Era of Enlightenment
    - But Manage the Risk

SIX Slides Spent on Risk Management – because its important –
Don't 'check the box' or perform an academic activity
If there is poor Risk Management – the project/system will show it

TOP TEN QUOTES ON RISKS – COLLECTED BY RICHARD BRANSON

10. “The biggest risk is not taking any risk … In a world that’s changing really quickly the only strategy that is guaranteed to fail is not taking risks.” – Mark Zuckerberg

9. “If you don’t play you can’t win.” – Judith McNaught

8. “Life is inherently risk. There is only one big risk you should avoid at all costs, and that is the risk of doing nothing.” – Denis Waitley

7. “Why not go out on a limb? Isn’t that where the fruit is?” – Frank Scully

6. “A ship in harbor is safe, but that is not what ships are built for.” – William G.T. Shedd

5. “When you take risks you learn that there will be tines when you succeed and there will be times when you fail, and both are equally important.” – Ellen DeGeneres

4. “There is freedom waiting for you,
On the breezes of the sky,
And you ask “What if I fall?’
Oh but my darling,
What if you fly?”
-- Erin Hanson

3. “I am always doing that which I cannot do, in order that I may learn how to do it.” – Pablo Picasso

2. “Don’t be afraid to take a big step. You can’t cross a chasm in two small jumps.” – David Lloyd George

1. “Two roads diverged in a wood … I took the one less travelled by, and that has made all the difference” - Robert Frost

Name: Manage Requirements	**Purpose:** Analyze, Determine, Document WHAT is Needed and to what Specification, and Transform to a Systems Perspective
Inputs and Prerequisites: Bounded and Scoped Problem, New Mission or New Business Operation, Enterprise Architecture, Regulatory and Compliance	**Information:** Trace Requirements From Source to Testing, to When and How Delivered
Related Processes/Tasks: Start Project, Systems Analysis, CM/DM, Manage Project, Manage Risk, TEM< Create Architecture, VER & VAL	**Tools:** DOORS
Roles & Responsibilities • **Project Manager** – success of the project • **Systems Engineer:** create a design and solution of acceptable risk – balances Cost, Scheduler, Performance – to deliver the desired capability • **Operations/User –** communicate need and operational requirements to Requirements Manager and SE • **Requirements Manager** – elicit, document, synthesize, decompose, trace requirements through entire lifecycle	
Process Activities 1. **Document Problem** into a Needs Statement 2. **Write and Document** the CONOPS 3. **Write** Each **Functional/User** Requirement – illicit the requirements from the User 4. **Identify Key Performance Parameters, MOEs, MOPs** 5. **Decompose** Requirements and Document 6. **Gain Approval** of Functional/User Requirements 7. **Document/Decompose/Synthesize/Functional Analysis - Systems Requirements -** 8. **Gain Approval** of System Requirements/Document into Tools	
Output and Results: Needs Statement, CONOPS, Functional/User Requirements, Systems Requirements, Key Performance Parameters, MOPs and MOEs, Hierarchy, Specification Tree	**Change Log** Original Version 04/01/20 Change 1 : added step 4 in Activities ; 04/11/20

MANAGE REQUIREMENTS
TYPES OF REQUIREMENTS (1) -- ELABORATION

- Functional Requirements –
  - succinct and discrete statement from the operator, user perspective – a function of the system
  - The User Shall ….. **The User Shall …..**
  - For Agile – **As an Operator I want to ……**
  - Functional Requirements Come from the User/Operator/Maintainer
- Systems Requirements
  - What the System Will Do to Fulfill the Functional Requirement
  - For the Architecture and Developer (will not make sense to the User) **The System Shall ….**
  - Synthesized and Decomposed from the Functional Requirements
- Security Requirements –
  - Physical, virtual, administrative requirements to ensure system confidentiality, integrity and availability (ICD 503)
  - Part of the Functional but called out separately

MANAGE REQUIREMENTS

TYPES OF REQUIREMENTS (2) -- ELABORATION

- **Key Performance Parameters (KPP)** – characters of the system that must function or system will not fulfill its purpose.
- **MOPS and MOEs** – aid in creating alternative solutions – More on this in VAL
  - Methods of Performance –Methods of Evolutions – a subset and smaller description of MOPs – Several MOEs make a MOP.
- **Interface** – connection allowing data, power, control, form, and fit between two ore more
- **Size, Weight, Power** – defined area, budgeted weight, power budget and heat output – over a given period of time. Esp. important for aerospace
- **Regulatory** – Federal Acquisition Authority (FAR), Privacy and Security, Policy and Law, Health Insurance Portability and Accountability Action (HIPAA) and many more!
- **--- ilities – Related to the 'User Experience'**
  - Availability – ready for use, when and where needed – and perform as expected
  - Usability – ease of use – simple, clear, intuitive, layout and feel – relates to Human Systems Integration
  - Maintainability – ease of maintenance, cost and time needed to maintain, minimal disruption
  - Scalability – allowing for future grow
  - Reliability – little or no failure, or failure occurs how quickly will it recover – Mean Time Between Failure (MTBF)

REQUIREMENTS CONSIDERATIONS

- Bounded – define system perimeter and interfaces to other systems
- Human Interfaces – presentation, ergonomics, time and motion, safety and health
- System Interfaces and Integration – as agreed and documented in a MOA or API
- System Function – define what that portion of the system does – make it design agnostic
- Supportability – maintenance, time to repair, monitoring
- Environmental Conditions – externally and induced – moisture, temperature, vibration
- Verification Criteria – how do you know if it passed Verification
- Size, Weight, Power -- Components and elements given a power budget, and weight budget, size limitations
- Form and Fit – Mating of two separate parts to connect or release.

REQUIREMENTS PROCESS

- Derived Requirements – discover more requirements as detail is defined, and document why the derived requirement is needed
- Functional Analysis – group into like functions – all security requirements in one group, all processing requirements into another, all flight control into a another group etc
- Forming up a Hierarchy based on Functional Analysis – but may be unbalanced – some portions of the hierarchy will be overloaded
- System Specification – form the hierarchy – balanced to the needed detail, down to the Configuration Item (CI)
- Traceability – from source, to system, to architecture System Specification and down to the CI, and test case

WHERE DID THIS REQUIREMENT COME FROM?

Only You Can Prevent Orphaned Requirements !

Figure 18 RVTM - Trace Requirements From Source to Solution

From The User or Regulatory Compliance – these are decomposed to a discrete level

One Functional to Many System Requirements – for the developer

Part of the Design Satisfies the Requirement

VER - Test is written to the require ment

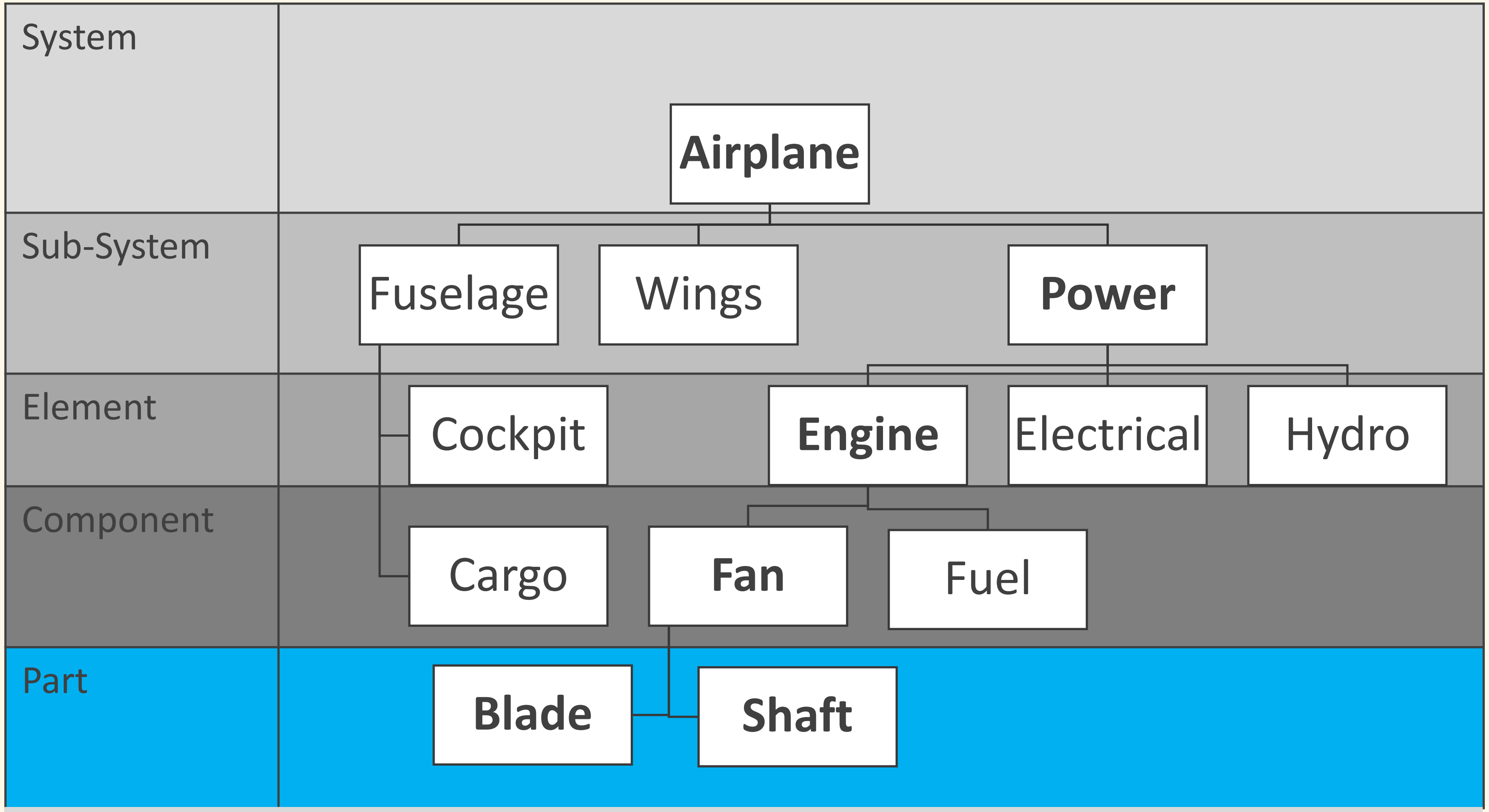
Requirements Traceability Matrix – from Functional Requirements to System Requirements to Test Cases							System Requirement	Design Component	Test Case
Req #	Priority	Source	Statement	Date Entered	Person Entering	Comments			
XXX.001	H	CONOPs 3.1	The User Shall ...	5 Jan 2020	I Newton		The System Shall	A12	
XXX.002	H	CONOPs 3.1	The User Shall ...	5 Jan 2020	I Newton		The System Shall		
XXX.003	H	FCC xxx	The User Shall ...	5 Jan 2020	I Newton		The System Shall		
XXX.004	M	FAR	The User Shall ...	5 Jan 2020	I Newton		The System Shall		
XXX.005	L	Style Guide	The User Shall ...	5 Jan 2020	I Newton		The System Shall		


From The User or Regulatory Compliance – these are decomposed to a discrete level
One Functional to Many System Requirements – for the developer
Part of the Design Satisfies the Requirement
VER - Test is written to the require ment

Requirements Traceability Matrix – from Functional Requirements to System Requirements to Test Cases							System Requirement	Design Component	Test Case
Req #	Priority	Source	Statement	Date Entered	Person Entering	Comments			
XXX.001	H	CONOPs 3.1	The User Shall ...	5 Jan 2020	I Newton		The System Shall	A12 XXXX	
XXX.002	H	CONOPs 3.1	The User Shall ...	5 Jan 2020	I Newton		The System Shall	A12 1121	
XXX.003	H	FCC xxx	The User Shall ...	5 Jan 2020	I Newton		The System Shall	B12 1454	
XXX.004	M	FAR	The User Shall ...	5 Jan 2020	I Newton		The System Shall	B15 5678	
XXX.005	L	Style Guide	The User Shall ...	5 Jan 2020	I Newton		The System Shall		

HIERARCHY – AN EXAMPLE

3.1.7. Requirements Management
3.1.10 Architecting



- A System Hierarchy Breaks the System Down into Understandable Portions
- Define the names of each level of the hierarchy and establish a vocabulary early in the project

The Truth About Requirements

If you have some experience in Systems Engineering, you may now be thinking that although the CONOPS and the Functional Requirements are to describe WHAT the system will do, they often describe HOW the system will perform the functions. This WHAT versus HOW is challenging under any circumstance in the Systems world. The more a CONOPS or Functional Requirement describes HOW, the fewer options the designers have for innovative ideas of new technology or processes. A Functional Requirement that dictates a solution to build a high-speed buggy will preclude the use of the automobile. A requirement that dictates a fixed phone with a 30-foot cord would eliminate a wireless phone. There is no easy answer to this except the user and operator, and the systems engineer must have an open dialog that fosters innovation into WHAT the user needs and HOW a system will function.

Name: Configuration Management/ Document Management (CM/DM)

Purpose: Document and Approve Characteristics and Attributes of the System and of the Project. Control and Management the Baseline

Inputs and Prerequisites: Approved Project, Artifacts to approve, approved tool CM/DM Plan, CM/DM Process, CI, System Specification

Information: Trace Requirements From Source to Testing, to When and How Delivered

Related Processes/Tasks: Start Project, Manage Project, Manage Risk, VER & VAL, Requirements

Tools: CM/DM Repository**,** Chef, Puppet, CM Now!

Roles & Responsibilities

- **Project Manager** – success of the project
- **Systems Engineer:** Ensure Compliance, Capture and Document Configuration Items, Manage Change, Awareness
- **CM/DM Manager:** Write CM/DM Plan, Manage CM/DM Process, Manage/Control Change to Process/ Configurations, Awarer

Process Activities

1. **Plan for CM/DM** – write or verify Plan and Process are suitable, Create Configuration Items (CI) Schema, train team
2. **Capture CI and Artifacts** – team proposed base documents – approved at CM/DM Board, enter into the Repository, Baselined
3. **CM/DM Changes** – as changes are proposed, consider risk, requirement, improvement or defect, VER/VAL, and how change will be made, and awareness, fallback. Calorize Change as needed from Major, Minor, or Administrative or something similar
4. **Capture Minutes** from CM/DM Board and enter into CM/DM
5. Make Change – make change according to approved plan, for major changes perform during least amount of risk to mission. Document changes, monitor and 'fall back' if needed
6. **Conduct CM/DM Audits** – Physical accuracy of Baselined CI as documented to the 'as is'.

Output and Results: Managed Configuration, CI Baselined and Approved, Traceable and diagnosable configuration

Change Log

Original Version 04/01/20

Change 1 : added step 4 in Activities ; 04/11/20

CM/DM (1) – ELABORATION

- Terms
  - Baseline – approved set of artifacts and their value – the characteristic detail. May be future (to be), Current (as is), or Past Baseline (for falling back)
  - Configuration Item – a discrete item that is controlled, measurable, approved
  - Artifact – CM/DM is document driven – so all CIs are to be documented. The audit will verify that they system is built as designed.
- How Much CM/DM?
  - Can the system be reproduced with baseline documentation ?
  - Is the Cost of putting the CI under CM/DM less than the mission impact if the CI were not known?
  - Will an uncontrolled change to the system NOT cause mission failure?
  - If any are NO – put in in CM
- Examples of Artifacts: Schedules, Contract Documents, Requirements, Project Plan, SE Management Plan, Risks/Issues/Opportunies, Meeting Minutes, Budgets, Architectures, Test Plans, Test Cases, Test Results, User Stories, Interfaces, O&M Manuals, Agreement …..

RELATED TO ARCHITECTING INTERFACES AND INTEGRATION

HOW TO MAKE IT ALL FIT TOGETHER (1)

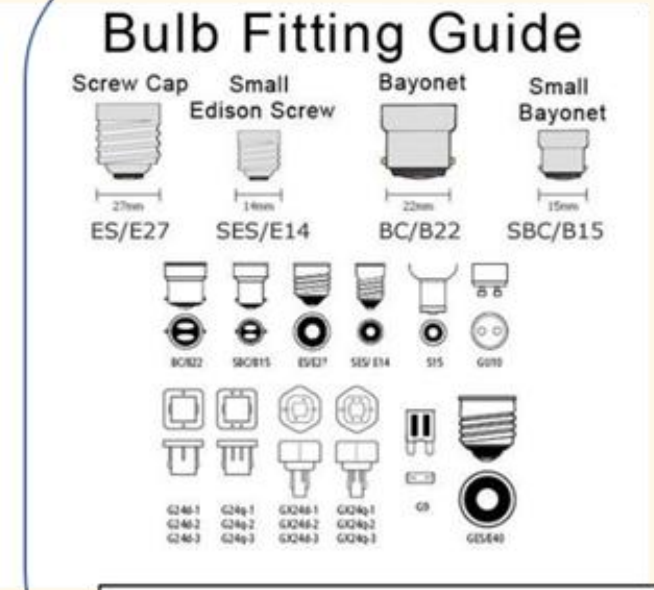


Figure 22 Interfaces – two objects fitting together, and Integration is the entire System Working Together

Interoperability, Interfaces and Integration – don't take it for granted – make these testable requirements

HOW TO MAKE IT ALL FIT TOGETHER (2)

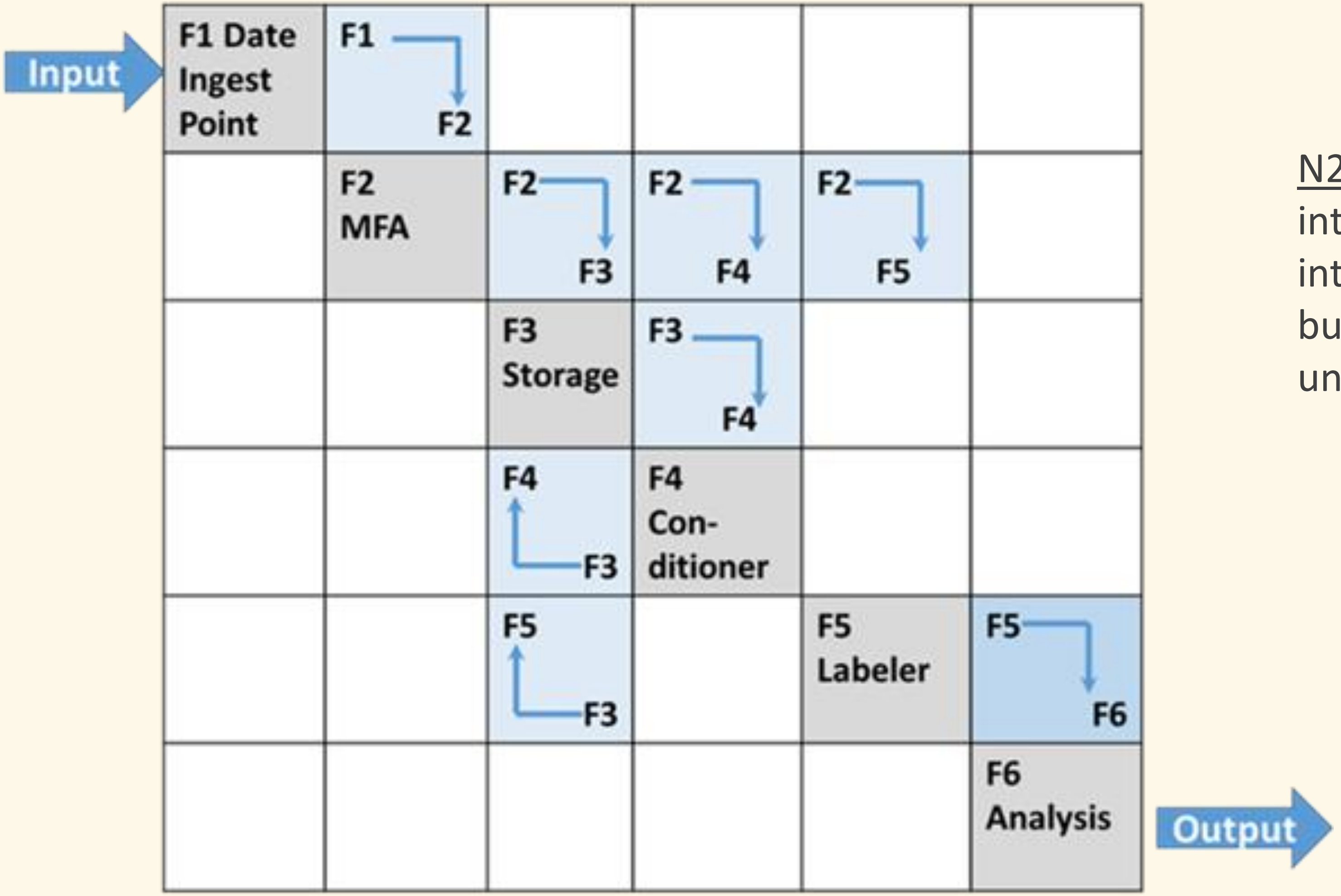


Figure 3-9 an Example of an N2 Chart – Leave No Outputs Behind

N2 Chart – one of many tools to aid in interfaces, leading to integration – leave no interface behind – helps during design, build, test but also at retirement when unplugging system

Name: Conduct Trade Study

Purpose: Given a situation with two or more alternatives to choose, make the most suitable selection.

Inputs and Prerequisites: Alternative and criteria for selection.

Information: Two or more alternatives to choose. May be a contractor, product, service to select. This process may be performed by any stakeholders during any time of the lifecycle.

Related Processes/Tasks: Manage Requirements, Architecting, Manage Risk, CM/DM, Modeling & Simulation

Tools: Excel and Common Desktop Apps

Roles & Responsibilities

- **Project Manager** – success of the project
- **Systems Engineer:** Ensure criteria established, weight factors, critical issues identified, dates, times, etc
- **Person Conducting Trade Study** – often the contract/sub-contract manger, but also the SE or any stakeholder

Process Activities

1. **Trade Study Alternatives** – the selection that must be made, the pros/cons, critical info for date/time, performance - weighted factors.
2. **Peer Review** – to ensure Trade Study Alternatives are correctly established – modify as needed.
3. **Conduct Research** – without make the assessment, consider most viable alternatives by research, trade shows, modeling and simulation, performance data, and conduct analysis. Score the results. Sophisticated Trade Study may 'mask' the product or service provider to reduce bias.
4. **After Data is Collected** – score the results and peer review, check.
5. **Conduct Selection –** recommend – decide contingent on level of authority (Table of Authority)
6. **Document the Results** – in project plan, system plan, architecture, in KM and ensure relevant stakeholders aware.

Output and Results: Product or Service Source is Selected, Action Items

Change Log
Original Version 04/01/20
Change 1 : added step 4 in Activities ; 04/11/20

Name: Architecting	**Purpose:** Satisfies the System Requirements
Inputs and Prerequisites: System Requirements, Risk Management, Trade Study, TEM, VER & VAL, Existing Architectures and Drawings for Possible Re-Use	**Information: Language of System Engineers,** Defined Notation and Vocabulary, Bounded System
Related Processes/Tasks: Start Project, Manage Project, Manage Risk, VER & VAL, CM/DM, Modeling & Simulation	**Tools:** PowerPoint, Excel, CORE, M&S Tools

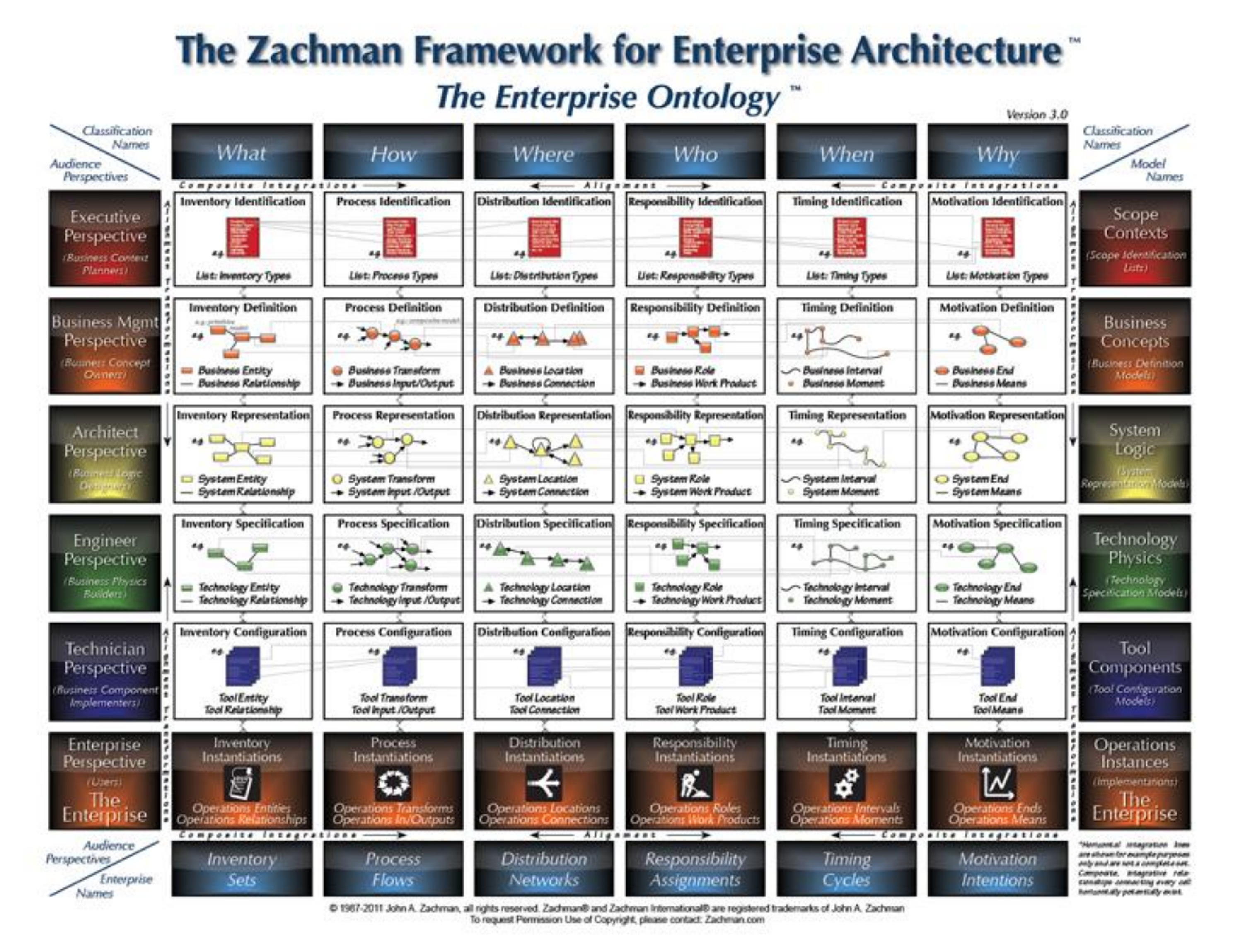
Roles & Responsibilities

- **Project Manager** – success of the project
- **Systems Engineer:** Create a compliant solution, suitable value – balance cost and schedule, risk
- **Architect** – work closely with operations, requirements manager to understand need, requirement, create alternatives, aid in the selection of alternatives

Process Activities

1. **Review and Understand** Need, CONOPS, Functional, System Requirements. Review Enterprise Architecture.
2. **Parse 'like' requirements into a hierarchy** – so all security requirements one set of function, user interface in another set, processes function into another, for aerospace – navigational in another set – this is defining sus-systems, elements
3. **Start High Level and Work Down** – System Architecture, Sub-System, etc. Created needed flow diagrams, Date Structures, Monitoring and Control, KPP, MOE and MOP
4. **Define Interfaces** – one system interconnects to others – standards, agreements, etc – leads to integration.
5. **Document and Notate** – common and agreed to 'language of systems engineers'. Estimate cost (BOM).
6. **Evaluate several alternatives** – Use TEMS and Trade Studies, full value of team to estimate risks, costs, performance. Make suitable selection
7. **Record** the 'to be' future architecture, use version # – use DM/DM Process
8. **Make Changes** – as updates proposed include in the 'to be' and perform same set of activities.

Output and Results: Baselined Architectures, Flow Diagrams, Bill of Material (or list of items), Managed Risks	**Change Log** Original Version 04/01/20 Change 1 : added step 4 in Activities ; 04/11/20



Could be
Planner
Owner
Designer
Builder
Maintainer
Contractor
Supplier
Citizen
User

Source: O'Rouke, Fishman, Selko; Enterprise Architecture – Use the Zachman Framework; Thompson Course Technology, 2003

ARCHITECTING – ELABORATION (1)

- Highest Level - Enterprise Architecture
  - Strategic Document that captures and bounds the organizational and systems interests
  - Based on Who, What, Where, How, When – of the stakeholders – Operators, Maintainers, Systems Engineers, Buyers, Recipients of the Emergent Behavior, etc
  - From Clinger- Cohen Act 1998 – which came form the Zackman Framework

- 1st and 2nd Level - System Architecture
  - First and second layer of System/Sub-System – still high level, interfaces and function
  - Often used to brief Senior Managers and those not fully technically astute

- 3rd and Lower Level - Systems Drawings
  - To answer the requirements to include Key Performance Parameters (KPP), MOPS, MOEs
  - Elements, Components, detailed interfaces, data structures, formats, policies
  - To what the Developers Build
  - Create Differing Alternatives with Pros/Cons – make a suitable decision based on pre-established criteria on Cost, Schedule, Perform, Risk

- Typical Architecting Artifacts/Documents

Operational	Physical Drawing	**Dictionary**
Flow Diagrams	Data Architecture	Interfaces
Network	Ontology and Nodes	**Narrative Description**
List/Bill of Material	**Configuration Management Ledger**	**Architects Name**

CONDUCT TRADE STUDY - EXAMPLE

Decision Matrix Example for Battery			ENTER SCORES	Extend Old Battery Life	Buy New Batteries	Collect Experient Data With Alternative Experiment	Cancelled Experiment
CRITERIA	Mandatory (Y=1/N=0)?	Weight	SCALE				
Mission Success (Get Experiment Data)	1	30	3 = Most Supportive 1 = Least Supportive	2	3	3	0
Cost per Option	0	10	3 = Least Expensive 1 = Most Expensive	1	2	3	1
Risk (Overall Option Risk)	0	15	3 = Least Risk 1 = Most Risk	2	1	2	3
Schedule	0	10	3 = Shortest Schedule 1 = Longest Schedule	3	2	1	3
Safety	1	15	3 = Most Safe 1 = Least Safe	2	1	2	3
Uninterrupted Data Collection	0	20	3 = Most Supportive 1 = Least Supportive	3	1	2	1
WEIGHTED TOTALS in %		100%	3	73%	60%	77%	0%

Name: Manage Defects and Improvements

Purpose: Operations and Maintenance (O&M) – process to capture defects or potential enhancements to a system.

Inputs and Prerequisites: Operational System includes Users, Maintainers, Support. Usually considered to have met IOC, KM and Other Similar Problems

Information: Defects or Potential Improvements, Predict Issues or improvement - Predictive Analyses

Related Processes/Tasks: Trade Study, TEM, CM/DM

Tools: Service Now! HP Help Desk, Excel, Telephone/Email

Roles & Responsibilities

- **Project Manager** – success of the project
- **Operations Manager** – if project is delivered and is in O&M, the PM is replaced by the Ops Manager
- **Systems Engineer:** Frequently review performance metrics, reported defects, trends, ensure process compliance.
- **Service Desk Manager** – trained workforce, with correct and usable tools, timely completion and resolution.

Process Activities

1. **User/Operator Has Problem or Observe Potential Improvement** – call, send email, report via online – date/time, name, contact number, system behavior, error codes, operational impact must be documented and reported
2. **Service Desk** – Completes documentation (Ticket), assesses critical level (1,2,3,4), initial analysis and trouble shooting, resolve on the spot if able using other similar documented problems
3. **Elevate the Problem** – until it is resolved – may require system modification using Manage Requirements, V&V, CM/DM
4. **Make Change or Correction** – Awareness to informing user, to more complex changes may include training.
5. **Document Results** in CM/DM and KM

Output and Results: Updated or Closed 'Ticket', KM Updated, CM/DM Completed

Change Log
Original Version 04/01/20
Change 1 : added step 4 in Activities ; 04/11/20

MANAGE DEFECTS – ELABORATION

- Aerospace and Large Systems
  - Organization Level Maintenance – at the work site, usually small repair or preventive
  - Intermediate Level Maintenance – regional maintenance site – fly or ship product
  - Depot Level Maintenance – usually one depot per system – scheduled overall, upgrades
  - Factory Level Maintenance – Rebuild, remodel - good airframe but needs modernizing – B52D to B53 G or H
- Level Repair or Maintenance (usually for IT)
  - Level 0 - Automated or Self-service such as password reset, knowledge base lookup.
  - Level 1 – Filters Help Desk Calls and basic support, troubleshooting, use of a documented knowledge base of previous and known questions. 'Tickets' are used to document the call and if needed for complex problems are elevated to Level 2
  - Level 2 – resolve break, fix and configuration problems, hardware and software repair. May be 'touch' maintenance where a technician is dispatched.
  - Level 3 – generally used to resolve infrastructure problems. Synonymous to Depot Maintenance.
  - Level 4 – complex problems that will require and external specialist to resolve.
- Predictive Maintenance – via acoustics, vibration , pattern of behavior, data, deterministic and empirical

Name: Verification and Validation (VER & VAL) or V&V	**Purpose: Verify (VER)** the system and its parts are designed, built and delivered as outline by the requirements **Validate (VAL)** that the system to be delivered will meet the expected operational need, in its expected environment.
Inputs and Prerequisites: CONOPS, Functional Requirements as documented in requirements registry, Risk Management, Architecture, O&M Process, Defect Management Plan & Process, Test Readiness Review (TRR). IOC for VAL	**Information:** VER and VAL are normally two separate processes but are combined here for succinctness and due to many similarities. VER and VAL start early but become front and center during and right after development.
Related Processes/Tasks: Manage Risk, CM/DM, Modeling & Simulation	**Tools:** PowerPoint, Excel, CORE, Test Tools

Roles & Responsibilities

- **Project Manager** – success of the project
- **Systems Engineer:** Ensure all requirements are traced to a test case, and ready to take output of test results.
- **Requirements Manager** – all requirements in repository, traced from source to solution
- **Verification Manager** – create a test case for each requirements, procure resources to conduct testing
- **Validation Manager** – create test plan/ scenarios for system performance in expected environment - CONOPS.

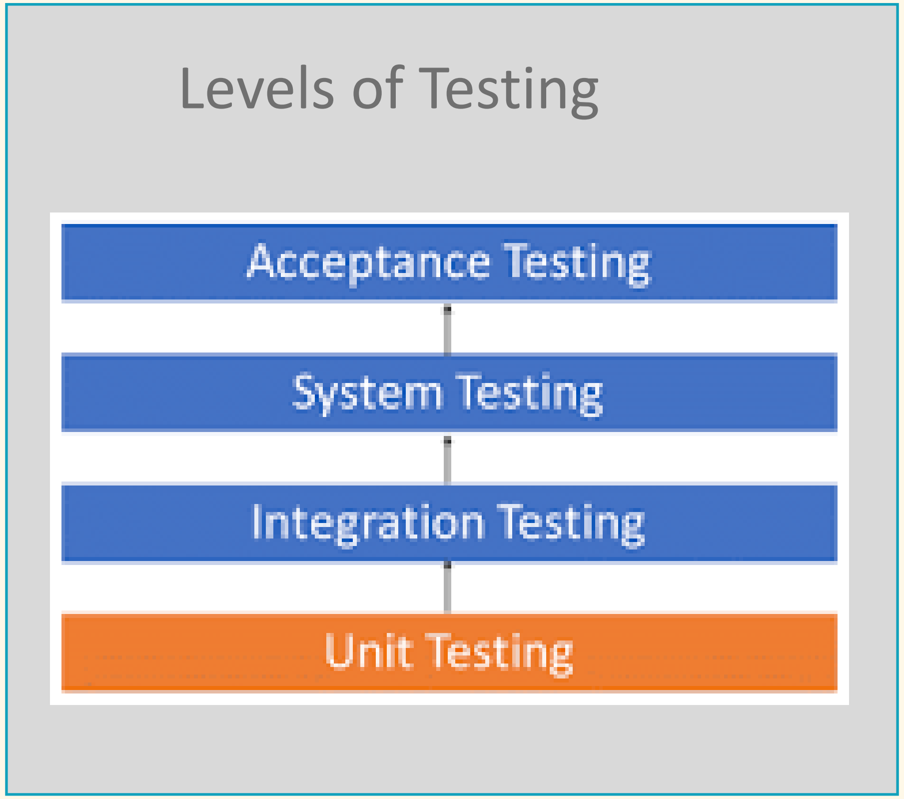
Process Activities

1. **Prepare for Verification– work with Requirements Manager and Operations to create test case for each requirements,**
2. **Ensure Resourced** – test tool, capture test results, test environment, test product, testing time, data
3. Conduct Testing – according to plan and process
4. **Capture and Analyze Test** – if complete, accurate – conduct test again as needed. Capture problems or issues
5. **For Validation** – create Use Cases or Scenarios to perform based on the system in its operational environment. Work with Operations Community to ensure training is performed, maintenance and support activities in place.

Output and Results: Test Plan, Test Cases, Test Results, Defects and Improvements, Results of KPP, MOPs/MOE.	**Change Log** Original Version 04/01/20 Change 1 : added step 4 in Activities ; 04/11/20

VER AND VAL (1) – ELABORATION

- Types of VER and VAL
  - Test – controlled conditions, real or simulated.
  - Inspection – visual or dimensional exam.
  - Demonstration – correct operations against observable characteristics.
  - Analysis – modeling and simulation.
  - Analogy or similarity (type) – evidence of similar element.
  - Simulation – performed on models.
  - Sampling – statistical sample.
- System 'Check Out' – is not formal testing but used to prepare a system at start up – for connectivity, or basic function



VER AND VAL (2) – ELABORATION

- **VER**
  - For critical – no fail systems – ongoing unit testing as being developed
  - Agile – upgrade made and some Verification can occur overnight – to provide assessment if its ready to implement
- **VAL**
  - **Measures of Performance (MOP)** – the physical and functional attributes relating to the system operation, measured or estimated under specific testing or operational environmental conditions. MOPs measure the design and performance requirements.
  - **Measure of Evaluation (MOE)** – the overall operational success criteria from the users' point of view. MOEs are design agnostic. MOEs are derived from MOPs.
  - **Key Performance Parameters (KPP)** – a requirement that is so important that if not met, the system is not considered successful.
  - **Technical Performance Measurements (TPM)** – specific measurements used to assess design and build progress. TPMs are few, and are select criteria that provide insight to the Systems Engineer and PM on the project's state-of-health.
- **Two More Terms**
  - **Initial Operating Capability (IOC)** – the basic deployed system delivering the needed mission capacity that allows for Validation.
  - **Full Operational Capability (FOC)** – once IOC is approved, the remainder of the system is deployed and once all deployed FOC is declared.

Name: Conduct Project Management Review (PMR)	**Purpose:** Communicate the stakeholders of the project and systems status, plans and illicit stakeholders in making decisions.
Inputs and Prerequisites: Project Plan, Schedule, Budget (EAC/ETC), Risks, Deliverables, Staffing, Action Items	**Information:** Includes Cost, Schedule and Perform, People Process, Tools and Risks. May be Project or Program Management Reviews, conducted approximately monthly.
Related Processes/Tasks: Manage Risk, CM/DM, Modeling & Simulation	**Tools:** PowerPoint, Projector, Computer, Room

Roles & Responsibilities

- **Project Manager** – success of the project, lead the PMR, prepare presentation, collect information
- **Systems Engineer:** Collect information from SMEs, Tech Team and other engineers, synthesize and provide to PM
- **Finance/Contracts** – provide information and reporting to the PM to include in PMR. Attend PMR.
- **Other Expertise** – depending on project and systems activities, others who may attend are V&V Manager, Requirements Manager, Task Managers, Usually Quality Manager attends.

Process Activities

1. **Compare Communication** - PM, SE and others periodically compared communication form stakeholders and customer to ensure all topics (emerging, new, ongoing) are captured. Review formal documentation of same.
2. **Project Manager** – extend 'data call' to relevant stakeholders – pertaining to the PMR. Check Contract to ensure required topics are included. Include suspense date when due back. Best to put into the Ops Rhythm.
3. **Prepare for PMR** – send out agenda, read ahead, reserve facility, check to ensure equipment functions, have a note taker. Often the PMR will include a written report – be sure the report and the briefing are consistent.
4. **Conduct PMR** – 1) Introduce and Purpose 2) Accomplishment 3) Cost (EAC/ETC), Schedule, Performance 4) Risks, 5) Plans or Requested Changes/Questions 6) Staffing 7) Review Actions
5. **Complete PMR** – Record Meeting and Actions Taken – obtain approval and put into CM/DM

Output and Results: Record of Meeting, Action Items, change requests.	**Change Log** Original Version 04/01/20 Change 1 : added step 4 in Activities ; 04/11/20

Name: Conduct Technical Exchange Meeting (TEM)

Purpose: Resolve a technical or programmatic problem.

Inputs and Prerequisites: Risk Management, Requirements, Defects, Architectures, Manage Defects, Action Items

Information: TEMs cover a range of issues, and start at the beginning, go through all development to O&M and even to retirement and system disposition. May be quite formal but often less formal is better.

Related Processes/Tasks: Manage Risk, CM/DM, Modeling & Simulation, Architecting, Analysis,

Tools: White Board, PowerPoint, Excel, Modeling and Simulation Tools, Meeting Room

Roles & Responsibilities

- **Project Manager** – success of the project
- **Systems Engineer:** Scope and scale, bound problem to resolve. Ensure problem, alternatives and solution are documented, and included into the system and project plans.
- **Requirements Manager** – all requirements in repository, traced from source to solution
- **Any Engineer, Technician, Manager** – given a problem to resolve may call a TEM with peers, to resolve.

Process Activities

1. **Define the problem** – document, bound the problem, determine key participates.
2. **Coordinate with Management or Peers** – prepare agenda, problem statement, send invites. Est. decision criteria.
3. **Draw Problem, Background, Restraints** – collaborate and brainstorm, sketch problems and potential solutions.
4. **Conduct Trade Study or Assessment** – include problem statement, selection criteria, assumptions, alternatives
5. **Conduct Modeling and Simulation** – as needed, and as part of the TEM and Trade Study
6. **TEM Make Recommendation** – given this info from Trade Study, Modeling and Sim, and White Board
7. **Systems Engineers and Team Make Selection** – contingent on their level of authority (Table of Authorities)
8. **Incorporate into Plan** – get needed change approval or Use CM/DM process, and awareness to team

Output and Results: CM/DM, KM, Action Items, Record of Meeting

Change Log
Original Version 04/01/20
Change 1 : added step 4 in Activities ; 04/11/20

DON'T BE A WATERMELON

Don't Be a Watermelon

I've attended many a PMR (or other meetings) where the project or system looks good on the outside but is wrought with problems on the inside. The metrics and charts don't always tell the truth. Eventually, those problems on the inside reveal themselves in the undesired emergent behavior. Hiding these problems demonstrate a lack of sincerity with ourselves and the recipients of our system. Your organization should have the core values to disclose problems and resolve as a team, - early, and have the leading indicators to identify and prevent problems – don't be a watermelon.

3.1.15 Conduct Technical Exchange Meeting (TEM)

Name: Conduct Technical Exchange Meeting (TEM)

Purpose: Resolve a technical or programmatic problem.

Inputs and Prerequisites: Risk Management, Requirements, Defects, Architectures, Manage Defects, Action Items

Information: TEMs cover a range of issues, and start at the beginning, go through all development to O&M and even to retirement and system disposition. May be quite formal but often less formal is better.

Related Processes/Tasks: Manage Risk, CM/DM, M&S, Architecting

Tools: White Board, PowerPoint, Excel, Modeling & Sim. Tools, Meeting Room

Roles & Responsibilities

- **Project Manager** – success of the project
- **Systems Engineer:** Scope and scale, bound problem to resolve. Ensure problem, alternatives and solution are documented, and included into the system and project plans.
- **Requirements Manager** – all requirements in repository, traced from source to solution
- **Any Engineer, Technician, Manager** – given a problem to resolve may call a TEM with peers, to resolve.

Process Activities

1. **Define the problem** – document, bound the problem, determine key participates.
2. **Coordinate with Management or Peers** – prepare agenda, problem statement, send invites. Est. decision criteria.
3. **Draw Problem, Background, Restraints** – collaborate and brainstorm, sketch problems and potential solutions.
4. **Conduct Trade Study or Assessment** – include problem statement, selection criteria, assumptions, alternatives
5. **Conduct Modeling and Simulation** – as needed, and as part of the TEM and Trade Study
6. **TEM Make Recommendation** – given this info from Trade Study, Modeling and Sim, and White Board
7. **Systems Engineers and Team Make Selection** – contingent on their level of authority (Table of Authorities)
8. **Incorporate into Plan** – get needed change approval or Use CM/DM process, and awareness to team

Output and Results: CM/DM, KM, Action Items, Record of Meeting

Change Log
(Printed Copies May Not Be Current)

Ver	Date	By
.1	04/01/20	BLS

CONDUCT TEM – ELABORATION

3.1.15 Conduct Technical Exchange Meeting (TEM)

- Conduct Technical Exchange Meeting (TEM)
- May occur anytime, often on short notice
- Given the choice of planning a TEM and losing momentum, or conducting an impromptu TEM and resolving a problem, conduct the TEM NOW
    - May be very structured or more informal but in any case
    - Have an objective and a defined problem
    - Have the right people attending (may not be perfect attendance though)
    - Document the discussion and the result, actions, etc and give to others to see, post on SharePoint/Common Directory
- TEMS are not a renegade process (because they may be impromptu and less formal) - still needs sanctioned and result known to the relevant stakeholders

OTHER ACTIVITIES AND PROCESSES

- Establish Agreements – between two or more parties, usually external
  - Memorandum of Agreements, Interfaces, Contracts, mutual project plans
  - By Working for an organization, you implicitly agreed to follow their process
  - Service Level Agreements (SLA)
- Transition
  - From Old to New Contractor, Terminate a Contract
  - In Expanding a Project, Program, System – added scope, new organizations
  - Organizational Changes – in Structure
  - Project - Know expectations/requirement, plan, build consensus, make agreements, think People, Process, Tools/Cost, Schedule, Performance and Risk
  - Sign Off Sheets, Checklists – and retain these in CM/DM
- Knowledge Management
  - Document your knowledge and pass it on
  - Tacit Knowledge, Tribal Knowledge is undocumented – experience
  - Knowledge is of capital value – and should be treated as such
  - Related to Lessons Learned and Knowledge Management – learn and improve
  - Feedback from Stakeholder and Customers – have a formal process and factor into your system and business
  - Informal Feedback – gain a dialog and trust with customers
- Customer Feedback
  - Formally and Informally Capture Customer Feedback, Comments, Surveys, Discussions
  - Some of the most valuable feedback is informal, spontaneous – pay attention, carefully solicit their opinions and challenges
  - Customer Relationship Management and Tools (CRM) new and very valuable tool for multiple people to capture, document customer comments, visits, telephone calls, etc
- Continual Improvement –
  - To not only fix discrepancies but to implement ideas, efficiencies
  - Section 5 has more – Tricks of the Trade

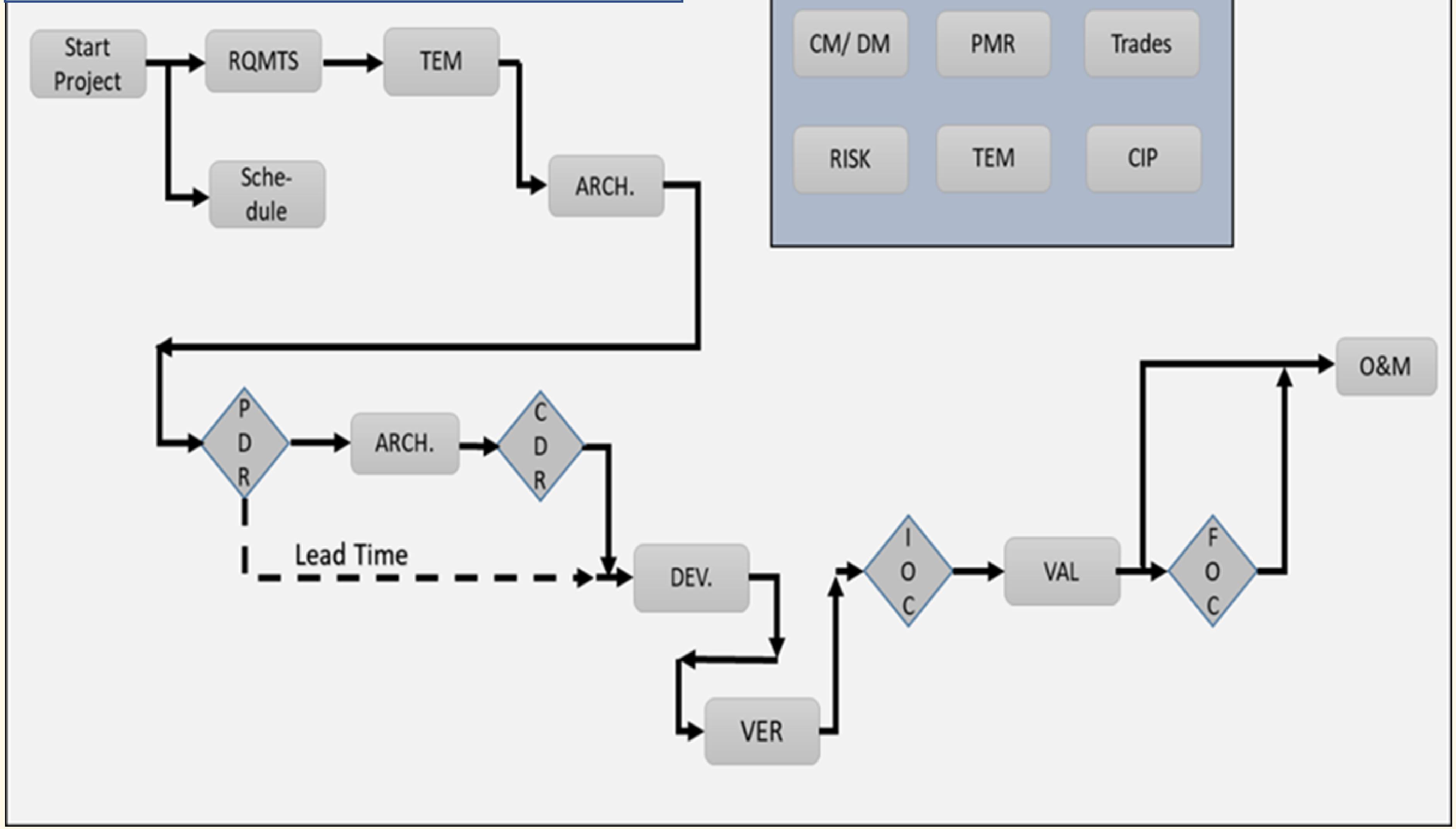
More information that can be, or should be included in formal processes are in Section 5 as Tricks of the Trade

BEST PRACTICES

- Activities that deliver the best results
- May be Industry Best Practices, or for a single Organization
  - Systems, Business, HR, Environmental
  - Big or Small Business, Organizational or for Government, Non-Profit, etc
  - Seems to have come about over the lasts several years
- May be formally documented or informal and not documented
- My Question ….
- Why not just document your Best Practices (to the needed level) and make a documented process?
- Why would anyone choose something other then 'Best Practices'
- Sounds like a substitute for Documented Processes

Simple and Notional Process Flow – Note the On-Going Processes versus the more Sequential Processes – if one process fails it may jeopardize the success of others. Some organization hyperlink for easy navigation.

Figure 3-12 Simple Process Flow



A Rant

I will challenge any team or person who states **they do not have processes**, or **they don’t need processes**. Everyone has a process, so the questions are ‘Are processes consistent across the team? Are processes performed the same by each person every day? Do you have one process and others have another process, and do you expect everyone else to go out of their way to do it your way?’

SECTION 3 TYPICAL PROCESSES SUMMARY

- Why Processes – Creatures of Habit, Focus less on routine and more on problem solving
- Systems Engineers and Project Managers – perform only those that add value.
- Its Not a Perfect World
- Typical and Most Important Processes

Manage Project	Manage Schedule
Manage Risk	Manage Requirements
Architecting	VER & VAL
Technical Exchange Meeting (TEM)	Conduct Trade Study
Manage Defects	Project Management Review (PMR)
Integration	Transitions
Configuration Management/Document Management	Agreements (MOA, Service Levels)

SECTION 3 - KNOWLEDGE VERIFICATION

1. What Processes Must Exist or there are signs of a problem? (subjective)
2. What are two elements of risk that are measured at High, Medium, Low?
3. What is a risk cube and a risk log/ledger? What's in it?
4. What is decomposition of requirements?
5. How can you ensure you leave no requirements behind or that you don't pick up orphaned requirements?
6. What method can help you make a quantitively decision about making a choice of product or service?
7. When transitioning to operations what would you expect to have completed?
8. What are the seven types of VER and VAL?
9. What are the six types of VER and VAL?
10. What is the difference between interfaces and integration?
11. What is your favorite flavor of ice cream?

WHICH SYSTEMS AND PROJECTS ARE WATERMELONS?

1. The systems engineer has qualified team and all the needed tools – and is thinks no processes are needed because the team is qualified to perform it their way.

2. The Project Manager has a good account of the schedule and the requirements only.

3. The System Engineer is taking the Big Bang testing approach for the complex system by waiting until the system is completely assembled, then do all testing (VER) at once.

4. The System Engineer takes most of the time thinking about the goals and requirement, and thereby leaving little time about how it will be achieved.

5. I'll know its right when I see it.

6. A discrete requirement take ten words but is valued at a thousand words.

7. The Project Manager has good information on the budget and actual costs, the schedule, and the performance, and what risks there are.

8. The Systems Engineer focuses on fixing the results and that the inputs and processes will take care of themselves.

RECOMMENDED CASE STUDIES

SECTION 4 SAY WHAT YOU MEAN, MEAN WHAT YOU SAY

- The Language of Systems Engineers
  - Recall Section 0 & 'What We Have Here is A Failure to Communicate'
  - A Few of the Languages of Systems Engineers – to capture and resolve problems, then to collaborate on, and communicate these problems
- Organizations Who Make the Language and Provide Guidance

A FEW OF THESE LANGUAGES

- Department of Defense Architectural Framework (DODAF)
  - Drawings for Operations, Networks, Data Schema, and three more
  - Operational View 1 (OV-1) is very common – but infrequently called an OV-1
- Integrated Definition Methods (IDEFx) – six diagrams – block flows, functional, data, ontology
- Uniform Markup Language (UML) – 15 drawings for Software Development
- Systems Modeling Language (SysML) – nine drawing types for Systems Engineering – grew out of UML
- U.S. Military Standards – MILSPECs, Handbooks, Standards, Performance Specs, Detail Specs.

DEPARTMENT OF DEFENSE ARCHITECTURAL FRAMEWORK (DODAF)

- Clinger- Cohen Act 1996 - Recognizes the need for Federal Agencies to improve the way they select and manage IT resources and states, “information technology architecture, with respect to an executive agency, means an integrated framework for evolving or maintaining IT and acquiring new IT to achieve the agency’s strategic goals and information resources management goals.” Chief Information Officers are assigned the responsibility for “developing, maintaining, and facilitating the implementation of a sound and integrated IT architecture for the executive agency”.

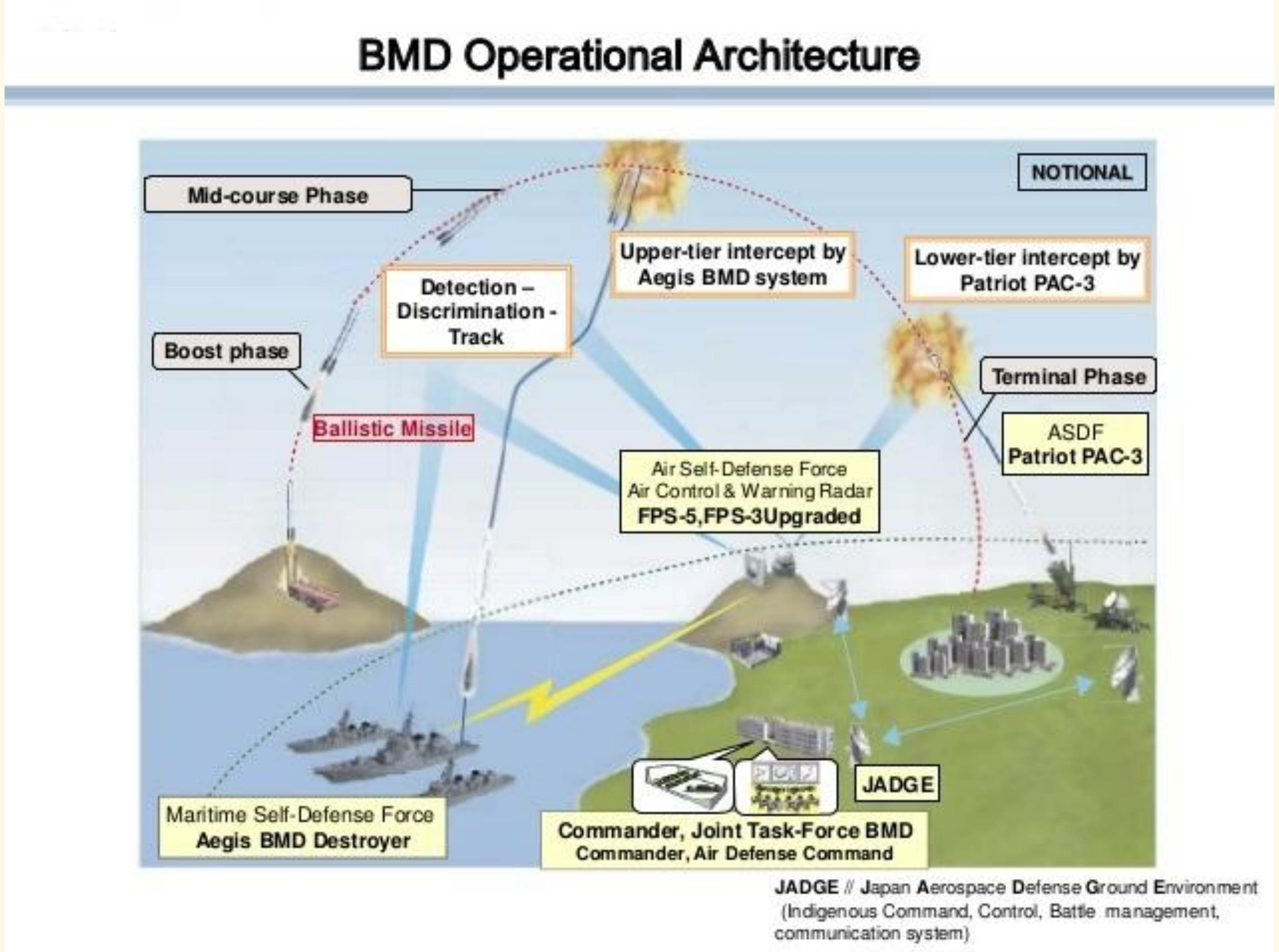
- Over 20 Views and Viewpoints (elaborations of Views) – most common are OV-1 Operational, OV-2, SV-1 Systems View

- DODAF is/has become a global ‘standard’ – MODAF and other commercial versions

https://dodcio.defense.gov/Library/DoD-Architecture-Framework/

DoDAF V2.0 / DoDAF V1.5	Operational Viewpoint	Systems Viewpoint	Services Viewpoint	All Viewpoint	Standards Viewpoint	Data & Information Viewpoint
AV-1				AV-1		
AV-2				AV-2		
OV-1	OV-1					
OV-2	OV-2					
OV-3	OV-3					
OV-4	OV-4					
OV-5	OV-5a, OV-5b					
OV-6a	OV-6a					
OV-6b	OV-6b					
OV-6c	OV-6c					
OV-7						DIV-2
SV-1		SV-1	SvcV-1			
SV-2		SV-2	SvcV-2			
SV-3		SV-3	SvcV-3a, SvcV-3b			
SV-4a		SV-4				
SV-4b			SvcV-4			
SV-5a		SV-5a				
SV-5b		SV-5b				
SV-5c			SvcV-5			
SV-6		SV-6	SvcV-6			
SV-7		SV-7	SvcV-7			
SV-8		SV-8	SvcV-8			
SV-9		SV-9	SvcV-9			
SV-10a		SV-10a	SvcV-10a			
SV-10b		SV-10b	SvcV-10b			
SV-10c		SV-10c	SvcV-10c			
SV-11						DIV-3
TV-1					StdV-1	
TV-2					StdV-2	

OV-1 EXAMPLE – OFTEN NOT CALLED AN OV-1



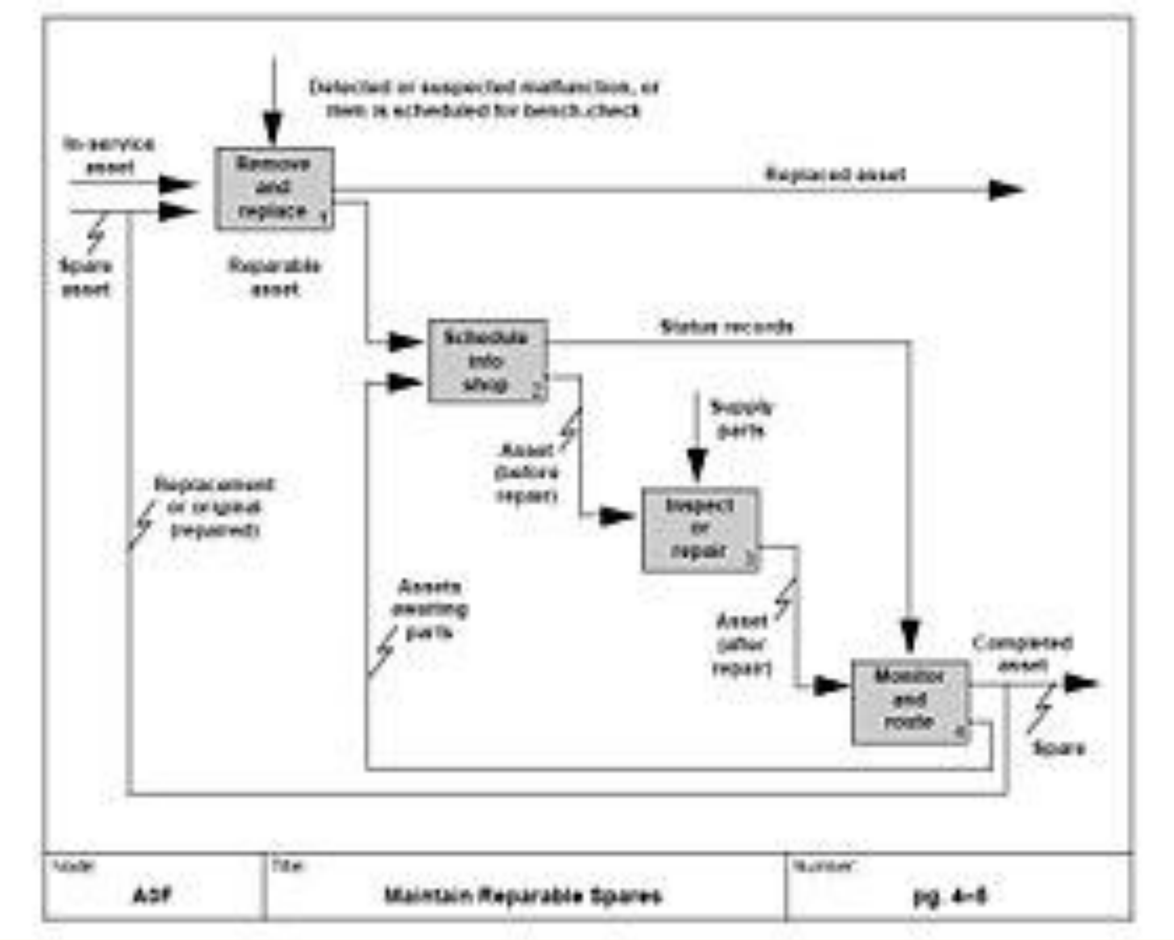
https://spfusa.org/research/u-s-japan-missile-defense-cooperation-increasing-security-and-cutting-costs/

INTEGRATED DEFINITION METHODS (IDEFX)

- IDEF0 - Function Modeling – this is the most common IDEF drawing, and often the only IDEF drawing used. It includes the inputs as Controls, Inputs and Mechanisms to go into the Function, and the resulting output. That output leads to another function block with the same type of inputs. IDEF0 inherently illustrate a decomposed and time-based set of functions.
- IDEF1 - Information Modeling – illustrates the structure, relationship and semantics of information.
- IDEF1X - Data Modeling – modeling of meta-data mainly for IT
- IDEF2 - Simulation Modeling – interfaces but later repurposed for timing and synchronization of system entities.
- IDEF3 - Process Description Capture – relationships between objects, entities.
- IDEF4 - Object Oriented Design – for component-based systems, to enough detail to allow development.
- IDEF5 - Ontology Description Capture – objects, their naming conventions and relationships.
- IDEF6 - Design Rationale Capture – ensure purpose in creating enterprise-oriented systems by asking questions about the design, alternative, and why alternatives are selected.

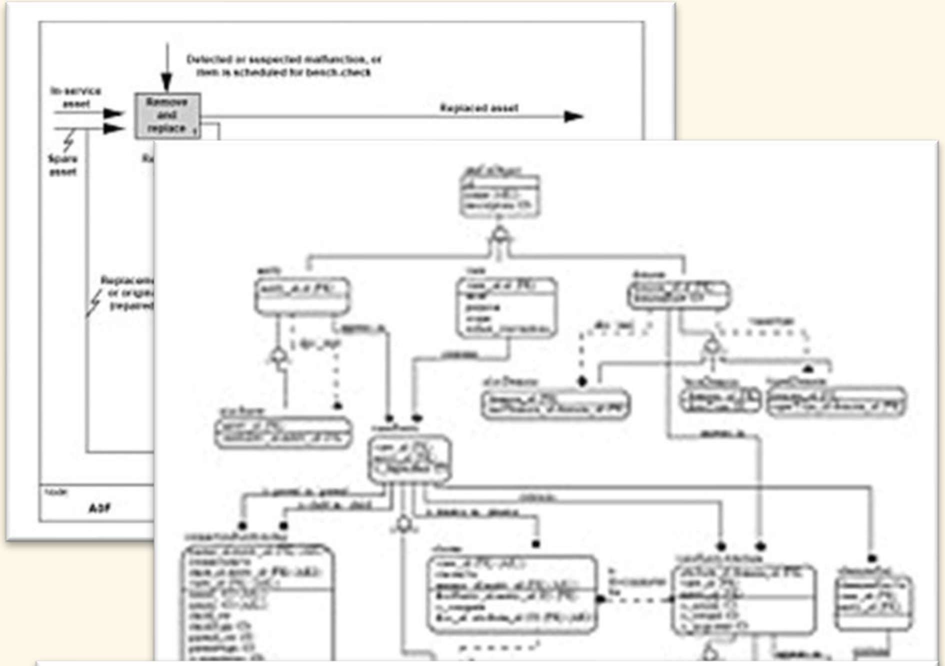
https://www.idef.com/idefo-function_modeling_method/

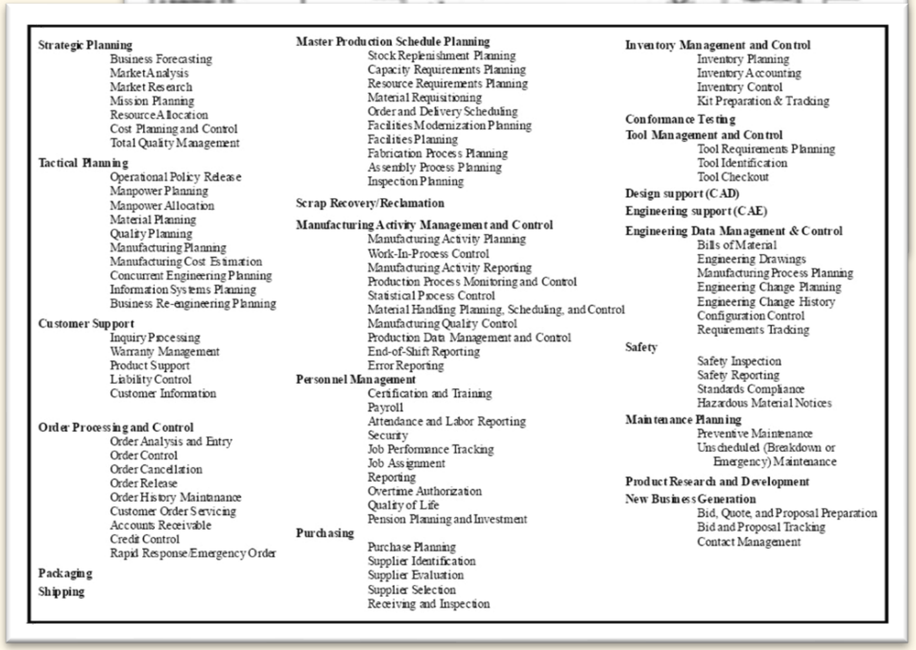
IDEF0 DRAWING – BLOCK FLOW DIAGRAM



More IDEFx Drawings

- Network
- Hierarchy
- Data Structure
- Organization





Strategic Planning
- Business Forecasting
- Market Analysis
- Market Research
- Mission Planning
- Resource Allocation
- Cost Planning and Control
- Total Quality Management

Tactical Planning
- Operational Policy Release
- Manpower Planning
- Manpower Allocation
- Material Planning
- Quality Planning
- Manufacturing Planning
- Manufacturing Cost Estimation
- Concurrent Engineering Planning
- Information Systems Planning
- Business Re-engineering Planning

Customer Support
- Inquiry Processing
- Warranty Management
- Product Support
- Liability Control
- Customer Information

Order Processing and Control
- Order Analysis and Entry
- Order Control
- Order Cancellation
- Order Release
- Order History Maintanance
- Customer Order Servicing
- Accounts Receivable
- Credit Control
- Rapid Response/Emergency Order

Packaging

Shipping

Master Production Schedule Planning
- Stock Replenishment Planning
- Capacity Requirements Planning
- Resource Requirements Planning
- Material Requisitioning
- Order and Delivery Scheduling
- Facilities Modernization Planning
- Facilities Planning
- Fabrication Process Planning
- Assembly Process Planning
- Inspection Planning

Scrap Recovery/Reclamation

Manufacturing Activity Management and Control
- Manufacturing Activity Planning
- Work-In-Process Control
- Manufacturing Activity Reporting
- Production Process Monitoring and Control
- Statistical Process Control
- Material Handling Planning, Scheduling, and Control
- Manufacturing Quality Control
- Production Data Management and Control
- End-of-Shift Reporting
- Error Reporting

Personnel Management
- Certification and Training
- Payroll
- Attendance and Labor Reporting
- Security
- Job Performance Tracking
- Job Assignment
- Reporting
- Overtime Authorization
- Quality of Life
- Pension Planning and Investment

Purchasing
- Purchase Planning
- Supplier Identification
- Supplier Evaluation
- Supplier Selection
- Receiving and Inspection

Inventory Management and Control
- Inventory Planning
- Inventory Accounting
- Inventory Control
- Kit Preparation & Tracking

Conformance Testing

Tool Management and Control
- Tool Requirements Planning
- Tool Identification
- Tool Checkout

Design support (CAD)

Engineering support (CAE)

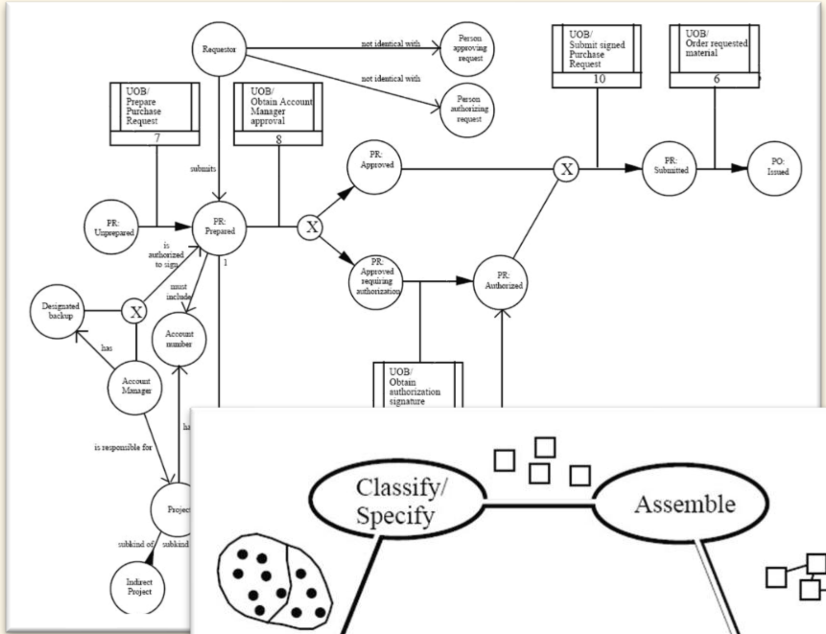
Engineering Data Management & Control
- Bills of Material
- Engineering Drawings
- Manufacturing Process Planning
- Engineering Change Planning
- Engineering Change History
- Configuration Control
- Requirements Tracking

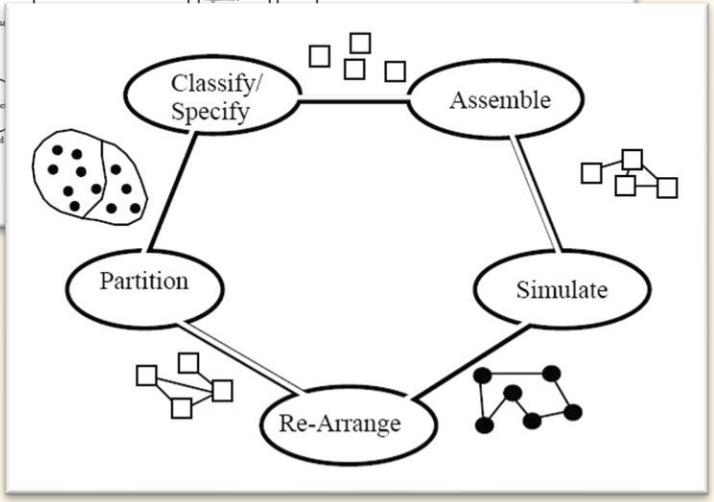
Safety
- Safety Inspection
- Safety Reporting
- Standards Compliance
- Hazardous Material Notices

Maintenance Planning
- Preventive Maintenance
- Unscheduled (Breakdown or Emergency) Maintenance

Product Research and Development

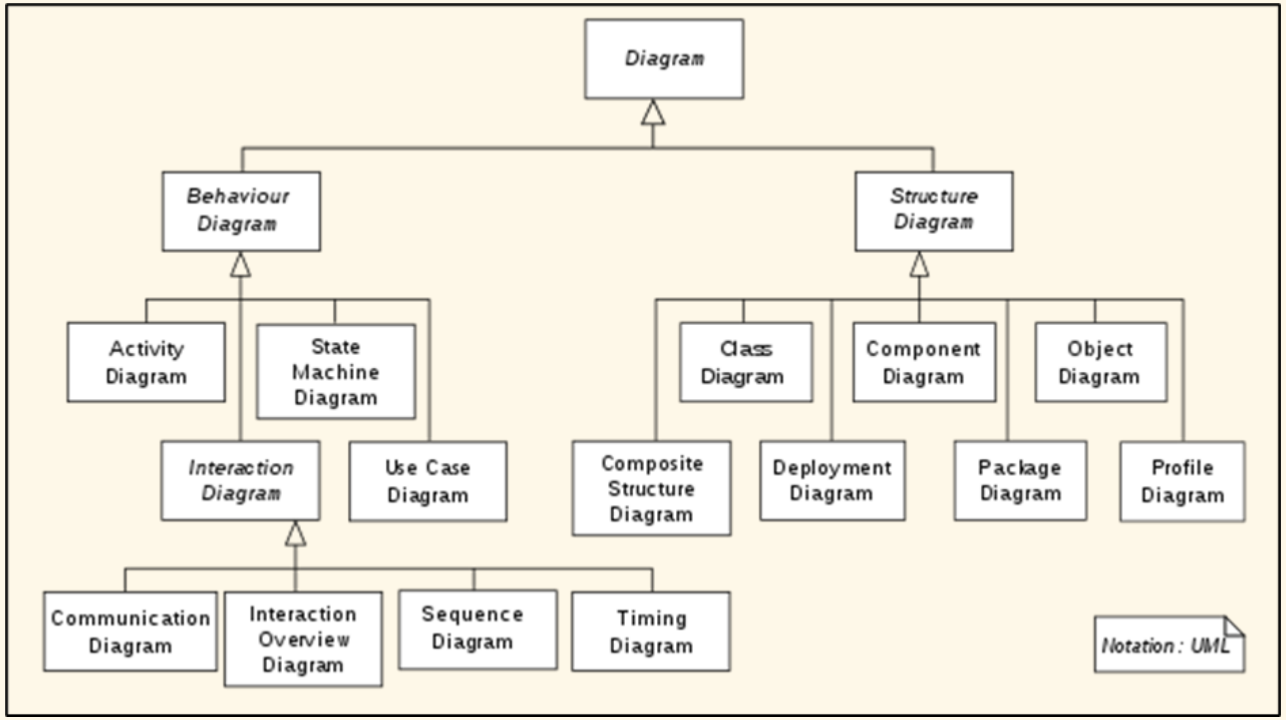
New Business Generation
- Bid, Quote, and Proposal Preparation
- Bid and Proposal Tracking
- Contact Management





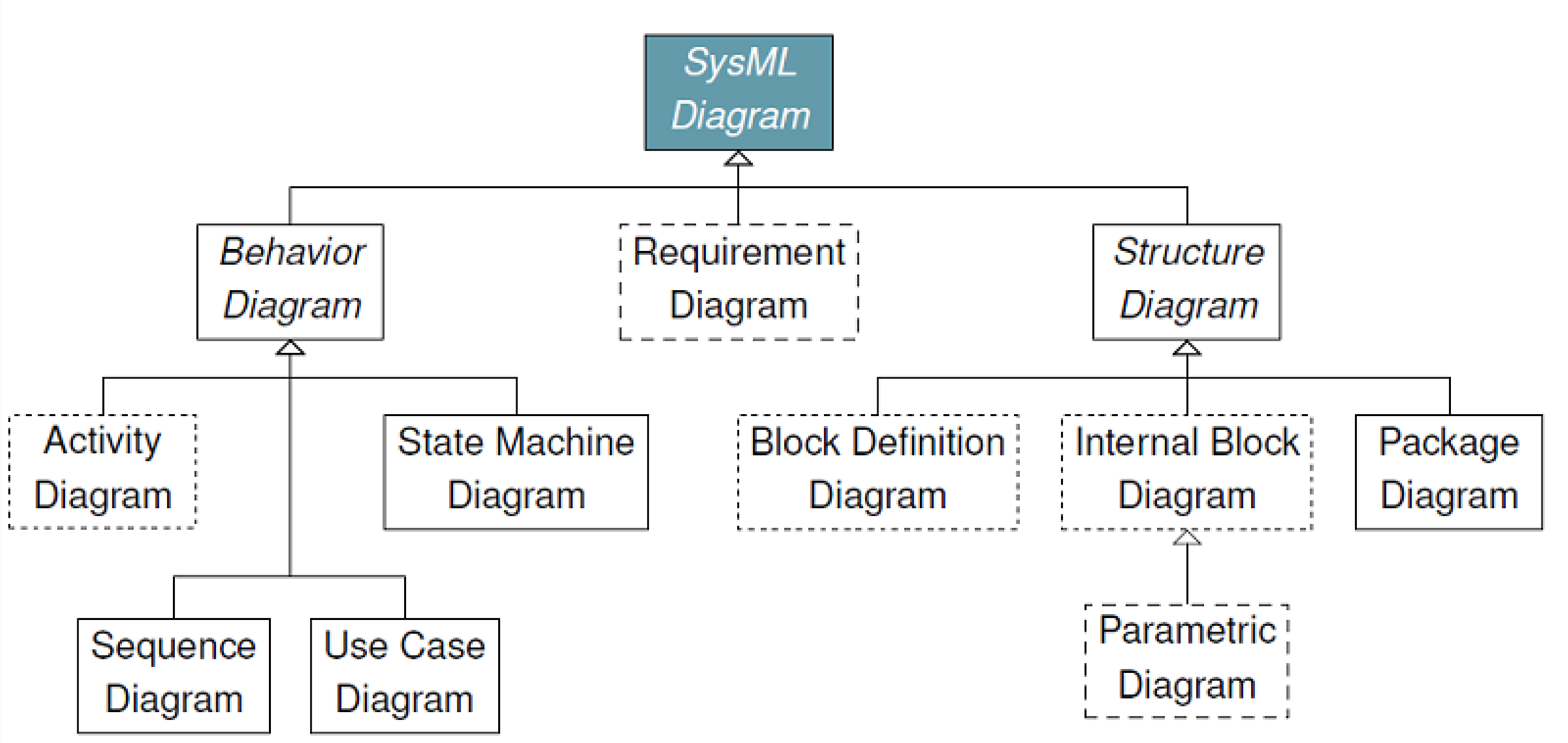
UNIFIED MODELING LANGUAGE (UML)

- UML - graphical representation providing a set of diagram formats and works as a model that has two different views 1) static with objects, attributes, operations and relationships, and 2) Dynamic (or behavior).
- Rational Software in 1994 - small team created a standardized modeling language for software engineering.
- More object-oriented programming.



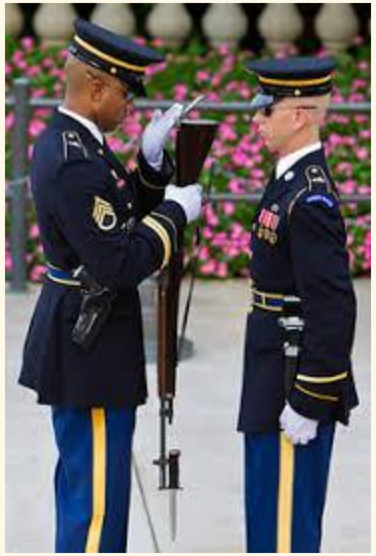
SYSTEMS MODELING LANGUAGE (SYSML)

- Grew out of UML – but as a general purpose System Engineering Language
- Nine Diagrams – less and more simple than UML



OBJECT MANAGEMENT GROUP®

U.S. MILITARY STANDARDS



‘The nice thing about standards is that there are so many of them to choose from.’ – *Andres S. Tanenbaum, Professor of Computer Science, Vrije Universiteit, Amsterdam*

- Four Formats
  - MIL-HDBK – Defense Handbook
  - MIL-SPEC – Defense Specification
  - MIL-STD - Defense Standard
  - MIL-PRF - Performance Specification
  - MIL-DTL - Detail Specification

- Most Applicable to Systems Engineering
  - MIL-STD-498, on software development and documentation
  - MIL-STD-499, on Engineering Management (System Engineering)
  - MIL-STD-810, test methods for determining the environmental effects on equipment
  - MIL-STD-1472, Human Engineering

https://en.wikipedia.org/wiki/United_States_Military_Standard

SUMMARY OF THE LANGUAGES FOR SYSTEMS ENGINEERS

		Works Well With					
Method	Used For	Requirements	Architecting	Decision	Develop	VER	VAL
DODAF	All Systems	Y	Y	N	N	Y	Y
UML	Software	Y	Y	N	N	N	N
SysML	All Systems	Y	Y	N	N	N	N
MIL-STD 498	Software	Y	Y	Y	Y	Y	Y
C4	Software but can be used in Other Systems	N	Y	Y	N	N	N
ISO/IEC 15504	SW Dev. Standards and Framework	Y	Y	Y	Y	Y	Y

GUIDING BODIES AND STANDARDS BODIES

- **Defense Acquisition University (DAU)** – DOD Accredited College – offer advanced degrees, certifications, Systems, Program Management and Acquisition Training – often sets the standard (or uses the standard) that others follow. This is the premier systems school in the United States. (IMHO)
- **Internet Engineering Task Force (IETF**) - open standards organization, which develops and promotes voluntary Internet standards, in particular the standards that comprise the Internet protocol suite.
- **World Wide Web Consortium** - member organizations that maintain full-time staff working together in the development of standards for the World Wide Web
- **Institute of Electrical Engineering (IEEE**) – professional association for electronic and electrical engineering.
- **International Organization for Standardization (ISO)** – international standard-setting body composed of representatives form various national standards, for manufacturing, agriculture, healthcare.
- **American National Standards Institute (ANSI)** – private non-profit – oversees development of voluntary consensus standard for products, services, processes, systems, personnel
- **Federal Communications Commission (FCC)** – legislation and enforcement - spectrum
- **Federal Aviation Administration (FAA)**
- **International Telegraphic Union (ITU)** – chartered by the UN for information and communication technologies

- PMP® Project Management Professional (PMP)
- PMI-SP® PMI Scheduling Professional (PMI-SP)®
- PgMP® Program Management Professional (PgMP)®
- PfMP® Portfolio Management Professional (PfMP)®
- CAPM® Certified Associate in Project Management (CAPM)®
- PMI-PBA® PMI Professional in Business Analysis (PMI-PBA)®
- PMI-ACP® PMI Agile Certified Practitioner (PMI-ACP)®
- PMI-RMP® PMI Risk Management Professional (PMI-RMP)®
- PMI-SP® PMI Scheduling Professional (PMI-SP)®

https://www.incose.org/about-incose

Expert Systems Engineering Professional – 20 years documented experience, substantial technical leadership, meet board, 317 Certified.

Certified Systems Engineering Professional – Five Years of Documented Experience, Broad Knowledge, Pass Written Exam, Able to ‘Find Their Way’, 2,098 Certified

Associate Systems Engineering Professional – Zero Years Experience (best to be working as an SE), Pass Written Exam, 1,074 Certified

https://www.sebokwiki.org/wiki/INCOSE_Systems_Engineering_Handbook

KNOWLEDGE VERIFICATION

- Why Is It Important to Have a Common Language?
- Name one (or more) of the Languages of Systems Engineers
- The Languages illustrated here apply to which areas
  - Architecting
  - Interfaces
  - Integration
  - Requirements Collecting
  - VER and VAL
  - WHAT the User Does
- What is INCOSE and PMI?

SUMMARY OF SECTION 4
SAY WHAT YOU MEAN, MEAN WHAT YOU SAY

- The Language of Systems Engineers
  - Recall Section 0 & 'What We Have Here is A Failure to Communicate'
  - A Few of the Languages of Systems Engineers – to capture and resolve problems, then to collaborate on, and communicate these problems
- Organizations Who Make the Language and Provide Guidance

RECOMMENDED CASE STUDIES

- TBD - ~ 30 mins
- Corp of Discovery – 1 hour

SECTION 5 SUPPORTING SKILLS

- Associated Skills and Professions
  - Systems Analyst
  - Operational Research
  - Usability Engineering/Human Systems Integration
  - Human Factors Engineering
  - EOHS
- Quality and Quality Manager
- ‘Tricks’ of the Trade
  - QFD – Voice of the Customer
  - Continual Improvement
  - Predictive Analysis
  - Decision Tree Analysis
  - Linear Algebra – Throughput, Max/Min Profit, Efficiencies
  - Root Cause and Corrective Action
  - Blooms Taxonomy
- Poised to Deliver

SYSTEMS ENGINEERS NEED SOME HELP TOO

- Systems Analysis – problem solver – analyze, design, implement solutions to tough problems – not only in IT but across differing domains. (contrary to Wikipedia)
- Operational Research – analytically solves advanced problems, capability development, mission assurance – statistical analysis, linear algebra for throughput, and load balancing, max, min of profit and loss, etc
- Usability Engineering/Human Systems Integration – study of structured methods enabling human to machine interfaces – user friendly, hand-eye motion **(think SysML)**
- Human Factor Engineering – psychological and physiological principles to engineer and design products, processes, and systems – to reduce human error, increase productivity and enhance safety and comfort – human interfaces with a device.
- Environmental, Occupational Health, and Safety (EOHS) – biological, chemical, physical factors affecting human health and the environment – as designed and built into systems. Includes transport, storage, use and disposal of systems, parts and output, includes the 'unseen' such as RF
- Thermal Engineering – design and assess the thermal energy created and omitted by a product, often in a confined space or in which there is limited power.
- Information Systems Security Managers (ISSM) – ensure and certify that systems are protect sensitive information and are reliable, and that the process outlined in the Risk Management Framework are followed. Directives for this activities are *ICD 503 Intelligence Community Information Technology Systems Security Risk Management, Certification and Accreditation*, and *ISO/IEC 27000 Information Security Management Systems*.

QUALITY

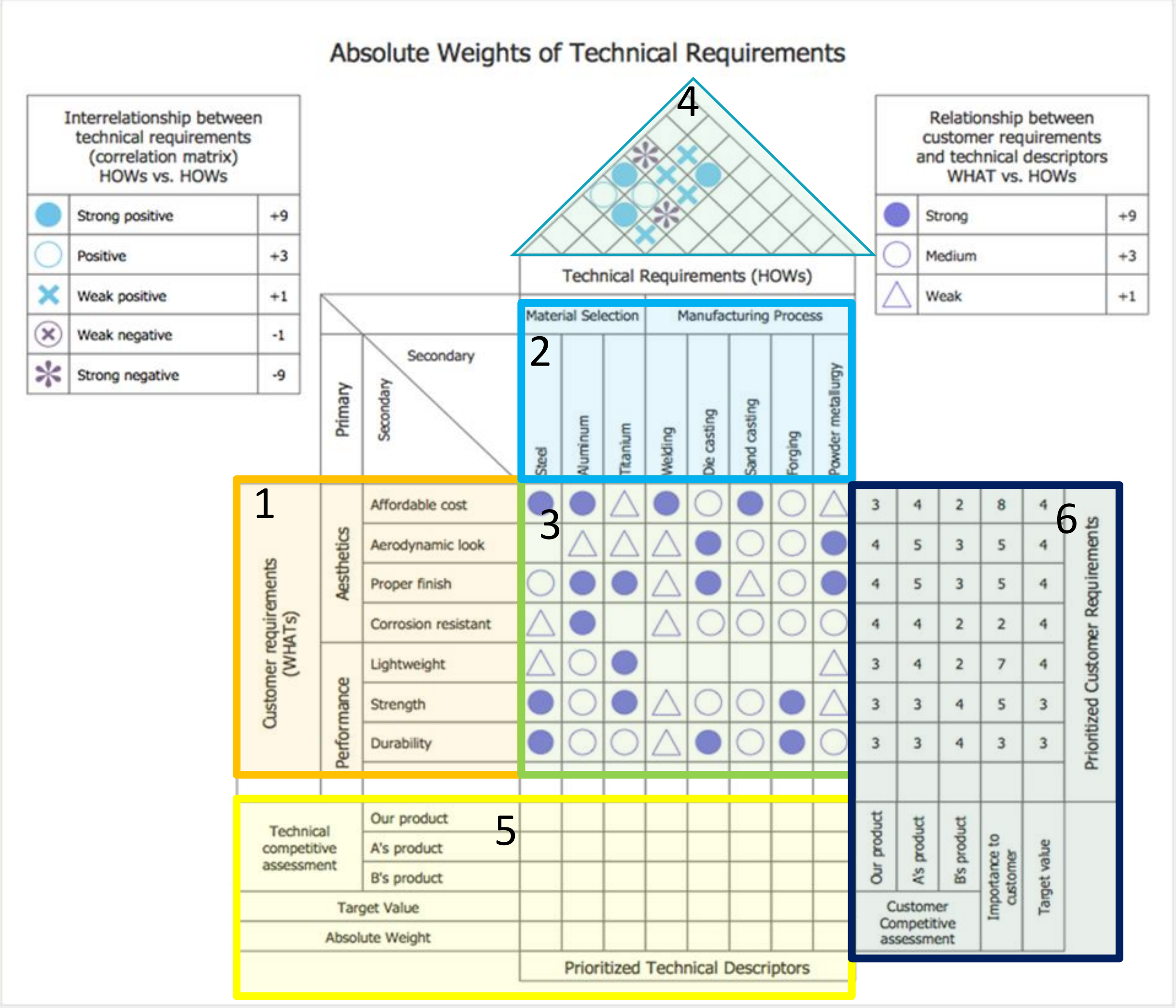
- Quality starts at the beginning of any program, continues through the entire program to when completed, and is everyone's job.
- The customer defines quality – so that's why its important to keep the 'voice of the customer', and their involvement
- Most organizations will have a Quality Policy and Training.
- Quality Manager – not assigned to any specific program but at a higher level to maintain autonomy, provides oversite and surveillance of processes and products during the activities and after performed. Most often done by way of audits and monitoring records.
- Think people, process, tools.
- Complex Systems live or die based on quality – few chances to re-do's.
- For no fail systems every piece or part may be checked and measured for accuracy. For less Critical systems a sample is taken or the first item checked

Why Do We Need a separate person called Quality Manager, or a Quality Department? Shouldn't we just do the best job we can? But you need oversight, establish and maintain a quality culture, training, especially for critical systems.
See W. Edwards Demming and Joseph M. Juran

QFD – HOUSE OF QUALITY

Figure 5-1 House of Quality

Aligns the Voice of the Customer to Systems Engineering



Quality → **F**unction → **D**eployment (QFD) or House of Quality

- Established 'Voice of the Customer' at project origins
- Determine What You Control and Cannot Control
- Match Most Desirable Needs of the Customer to Most Suitable Technical Capability
- Measure Competitive Threat and Capability
- Helps with Assessing – Should We Do This ?

1. Far Left – add Voice of Customer – what they want
2. Control Factors – what we (the provider/seller) controls
3. Center Box - Relationship between Customer Requirements and Technical Descriptors – this is what the customer wants (WHATs) compared to what you can deliver (HOW) - create and agree to a value – see upper right corner of illustration
4. Top – The Roof – cross ref each in the cell that they meet and rate (each control factor) and scale how well they compliment (the relationship) each. See scale in box on uppe left
5. Importance Factor (along bottom) multiply each cell in center to weight and put the value on Importance Factor Row – the higher the most you must focus on because that's what you can control, and what is important
6. Competition Rating – rate how your competition does in each of the 'Voice of the Customer' Create a rating such as 1 fail, 2, marginal, 3 Sat, 4 Excellent, 5 Outstanding
7. You may or should have multiple competitors

CONTINUAL IMPROVEMENT

- Circular – to repeat process and each time improve
  - Identify –
    - Process, but also people and tools are measured, errors noted, improvements and ideas
    - Document – what, who, how, when – data driven
  - Plan how to implement – a requirement, Model and Simulate, develop and Verify/Validate
  - Execute –perform awareness, monitor, have fallback ready, make the change
  - Review – monitor and check, via metrics – repeat as needed
- Process especially important in repeating, manufacturing processes
- Risk comes from variance, an external event, and **Disrupters –** new technology or new process that changes the industry

Figure 5-2 Continual Improvement

Variance on a process causes risk

DECISION TREE – TO MAKE COMPLEX DECISIONS SEVERAL 'MOVES' AHEAD. CONSIDER ALTERNATIVES, BREAK INTO A TREE AND PLACE VALUE ON EACH. CONSIDER THE LIKELIHOOD – FROM PAST EXPERIENCE OR USE OF PROBABILITIES

Figure 44 – Decision Tree Analysis - Best Decisions
https://hbr.org/1964/07/decision-trees-for-decision-making

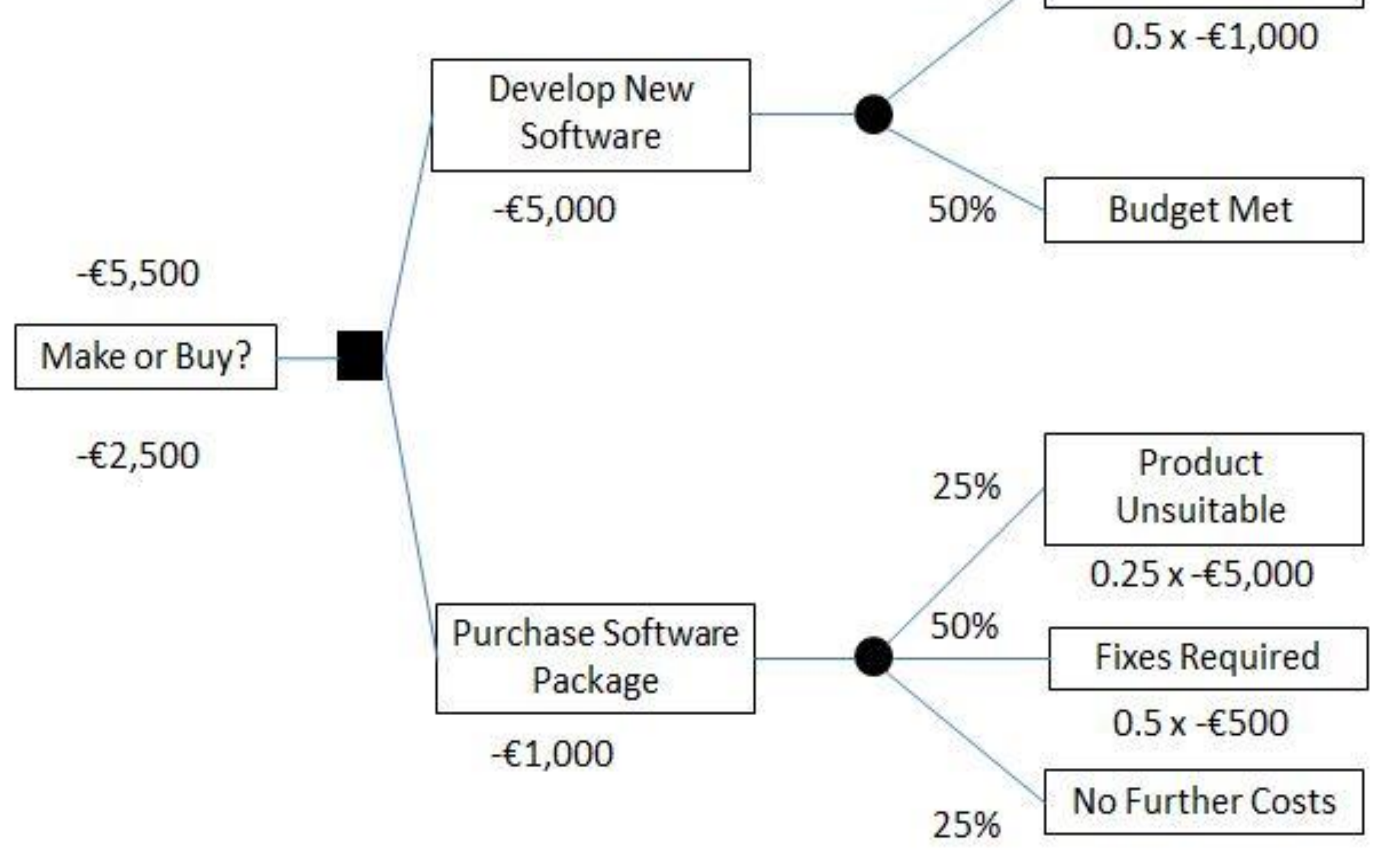
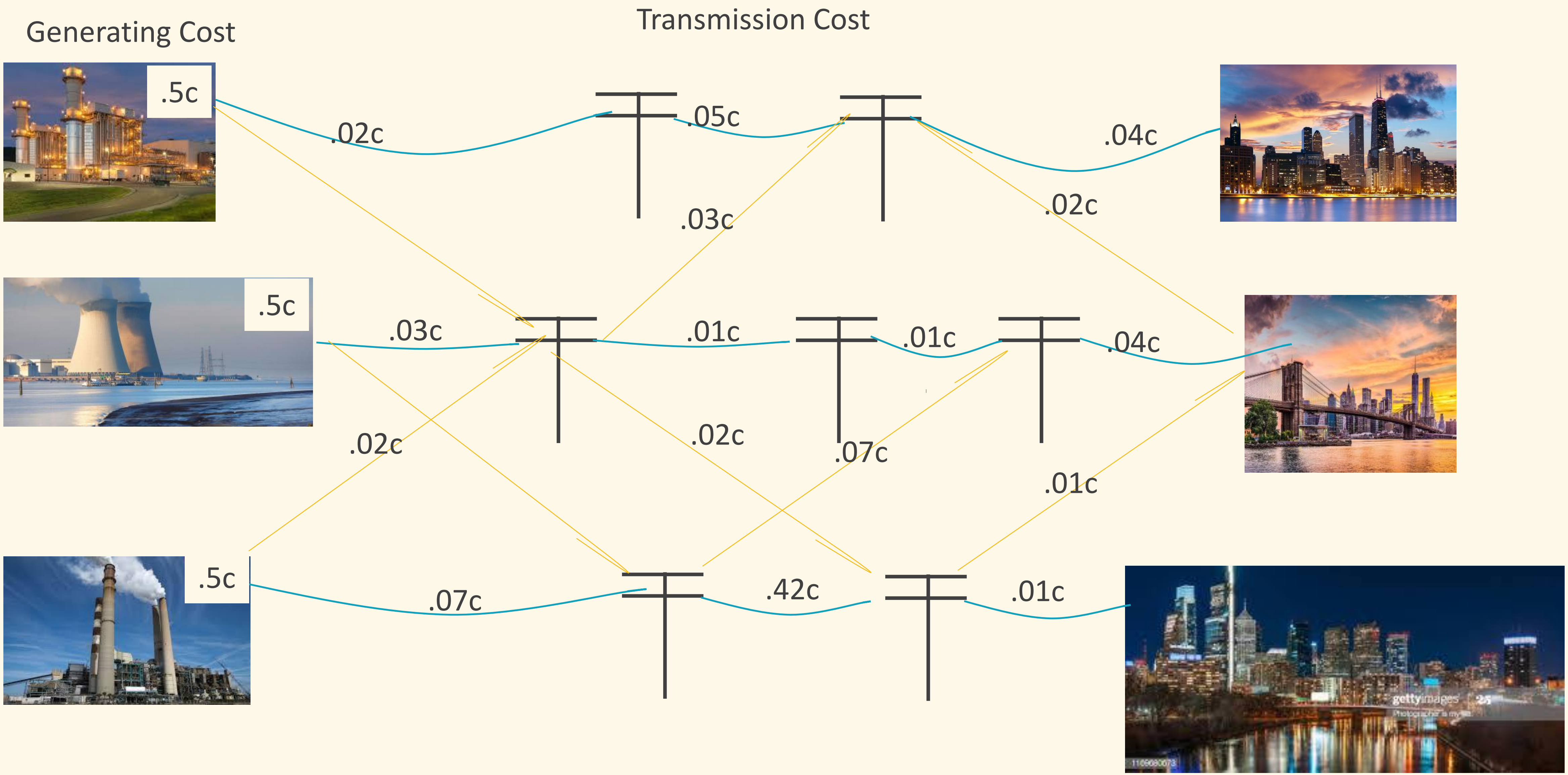
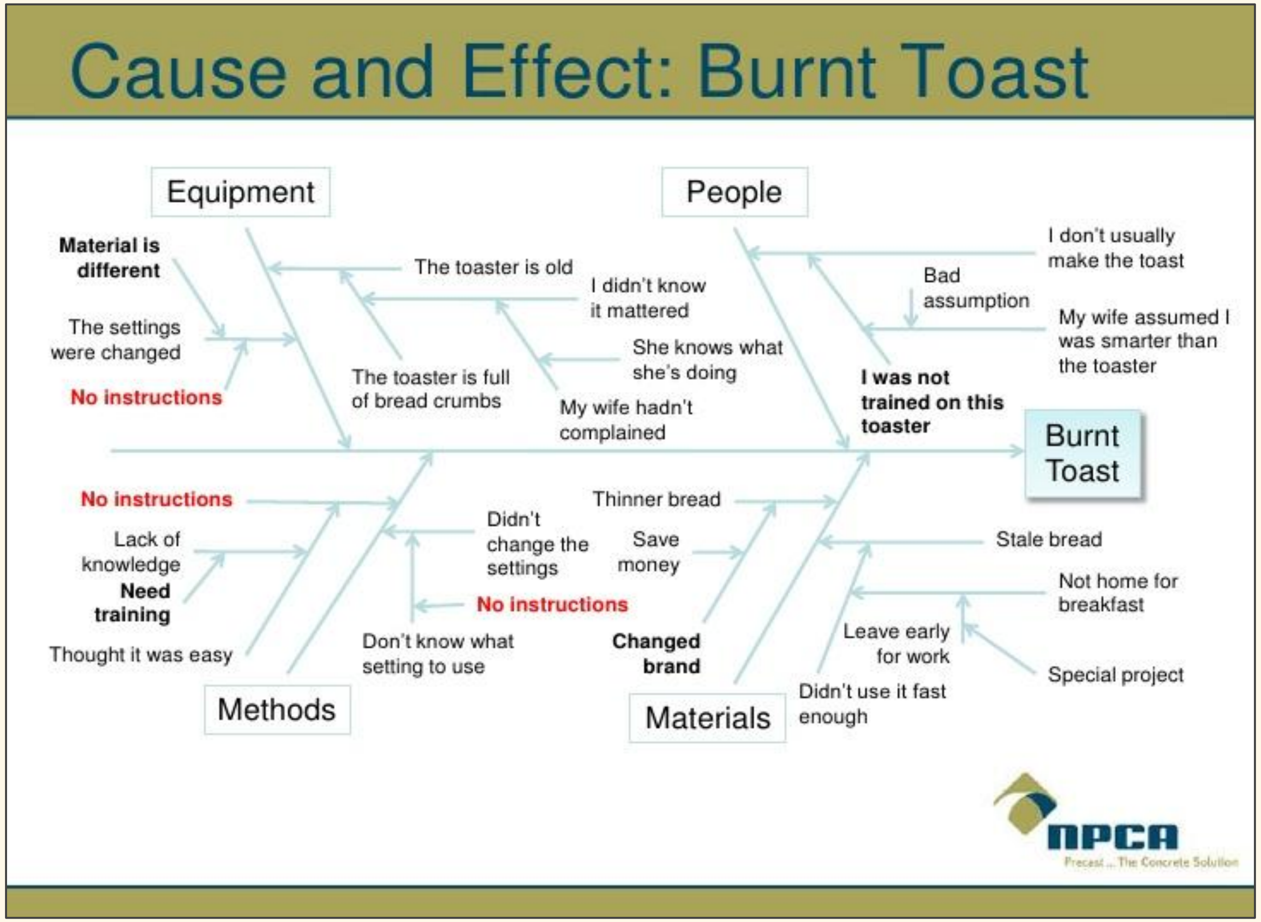


FIGURE 45 LINEAR ALGEBRA AND THROUGHPUT



ROOT CAUSE AND CORRECTIVE ACTION (RCCA) – STARTING AT THE PROBLEM, DIGRESS CONSIDER THE ELEMENTS THAT CAUSED IT. USE SUBJECT MATTER EXPERTS, DATA, EVIDENCE (THINK RECORDS) AND THOSE WHO WERE AWARE.



BE POISED TO DELIVER

- Open Architecture – that can easily (and quickly) accept interfaces, commercial products.
- Indefinite Delivery/Indefinite Quantity (IDIQ) Contracts – existing contract with a series of task orders that can be quickly enacted, without having to go through an extensive acquisition process.
- Anticipate potential needs and requirements and stage a provider with potential designs.
- Providers can have supply agreements already established (often critical path activities).
- Providers could have some candidate designs completed if given requirements.
- Stage equipment, keep hot and warm spares.
- Concurrent design and development – require close team coordination so the pieces can integrate.
- Use mature technologies (TRL 6-9) that requires little testing and preparation time.
- Move activities out of the critical path early by completing these tasks such as security accreditation.
- Keep signature authority and approval authority to a minimum by delegating and empowering. Many an hour and day have been lost awaiting an executive signature that was probably not really needed.
- Best Practices and Processes – confirm that they are current and efficient, so when used it not like the first time they were ever used.

Remember Cost, Schedule, Performance, Risk – if you are quick it may cost more, if you expect high performance – this may not be for you.

HOW PEOPLE THINK AND LEARN

- Learning Model – Blooms Taxonomy – Levels of Learning
  - Remember – recall facts and basic concepts
  - Understand – explain ideas and concepts
  - Apply – Use information in new situations
  - Analyze – Draw connections among ideas
  - Evaluate – Justify a stand or position
  - Create – Produce new or original work
- This is useful in communicating, or teaching – to the appropriate level

HOW PEOPLE THINK AND LEARN

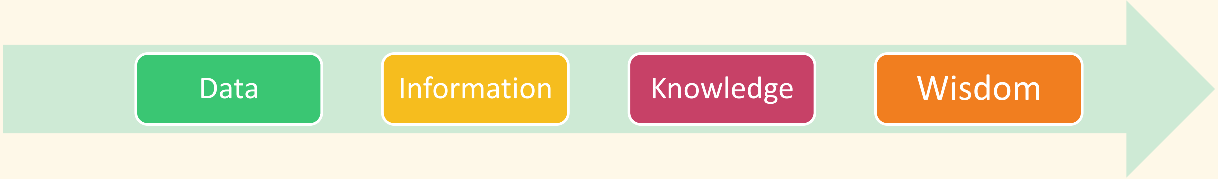
- Reasoning
  - Inductive Reasoning –
    - multiple premises – all believed to be true that are combined to obtain a specific conclusion
    - Define a Hypothesis and then find evidence that its true
    - Bottoms Up Logic – Antidotal
    - Sounds Holistic
  - Deductive Reasoning
    - Collecting and evaluating multiple sources of evidence
    - Top Down Logic
    - Empirical
    - Sounds Like Reductionist

HOW PEOPLE THINK AND LEARN

- Deterministic and Stochastic
  - Deterministic – given an data set (random, output) – expected range and anticipated.
  - Types are Empirical, Knowledge, Driven, Process Driven
  - Predictive – used on Big Data
- Stochastic – large data sets with result having greater ranges, often unanticipated
  - Input is Random
  - Output is Variable
  - Use Monte Carlo to find 80/20 of the data set

HOW PEOPLE THINK AND LEARN

- **Black Swan Events** – 'We didn't see it coming'
  - An implausible event, outside the mindset of the possible, once in a lifetime event
  - Severe Consequences
  - Standard Forecasting cannot predict, data samples
  - **BUT** – in hindsight we should have seen it coming, we didn't pay attention, and Black Swans are really subjective
  - What Black Swans are out there?



SECTION 5 SUPPORTING SKILLS - SUMMARY

- Associated Skills and Professions
  - Systems Analyst
  - Operational Research
  - Usability Engineering/Human Systems Integration
  - Human Factors Engineering
  - EOHS
- Quality and Quality Manager
- Poised to Deliver – quick but expensive
- ‘Tricks’ of the Trade
  - QFD – Voice of the Customer
  - Continual Improvement
  - Predictive Analysis
  - Decision Tree Analysis
  - Linear Algebra – Throughput, Max/Min Profit, Efficiencies
  - Root Cause and Corrective Action

SECTION 6 WHAT'S NEXT IN SYSTEMS ENGINEERING? (MY OPINION)

- Systems Engineering in Business
  - The Triad of Systems in Business
  - Systems Engineering in Business
- Trends
  - Leaning more to Holistic Thinking
  - Three Generations of Systems Engineering
- Session Summary
- Take Away

FROM DR GERALD BROCK AT GWU ~ 1999

- Must have the convergence of Technology, Policy and Economics for a credible business
  - Technology – many theories exists but only when science and technology can make it a reality
  - Policy – reflection of societies will – deregulation, safety and health, security.
  - Economics – using limited resources to create and deliver a product at a desirable price – where demand and supply cross.
- Not Available Today but could be soon – and the impediment
  - FEDEX/UPS long haul drones - Policy
  - Virtual surgery - Policy
  - Block chain - Policy
  - Human travel to other constellations – Technology, Policy
  - Foam Bricks – Policy, Economics

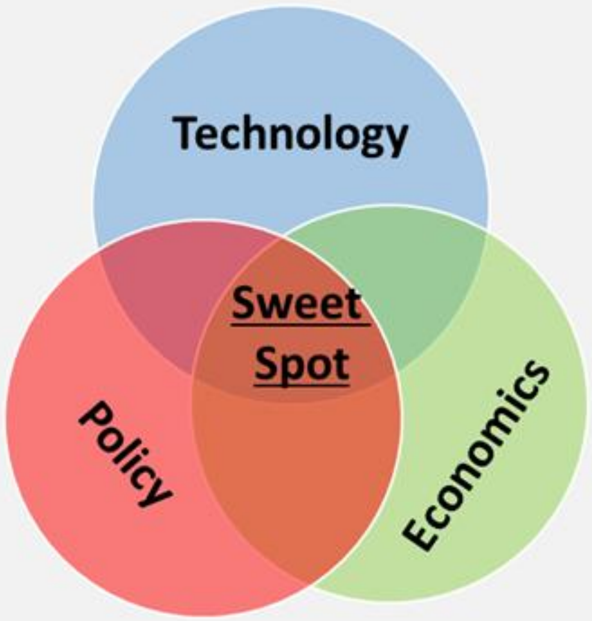


Figure XX – The Triad of Business

SYSTEMS ENGINEERING IN BUSINESS

CAPTURE NEW & KEEP EXISTING BUSINESS – MOSTLY APPLIES TO GOVERNMENT BUSINESS

- Given an emerging opportunity – start early, start often, shape the market
- Eventually – Potential Offerors submit White Papers on solving a complex problem, Market Surveys, Conferences, Q&A
- Have notional solution – rough est. of cost, risk, timeline – in time to select alternatives
- DRFP – SOW, Evaluation Criteria, Contacts and Pricing Info, Requirements – more questions and comments ← **the architecture should be done with a price estimate, form a Probably of Win.**
- Follow Up – anticipate the questions, response to Final Proposal Revision (FPR) – be thorough – **sometimes they are trying to tell you something (we want you but fix 'this')**
- Don't be arrogant – submit something that is **compliant, compelling and executable**
- The Best Way to Win New Business is to **execute well on what you have** (organic growth) – build an excellent reputation

'YOU ARE A SYSTEMS ENGINEER!'

1.5 Who Are Systems E

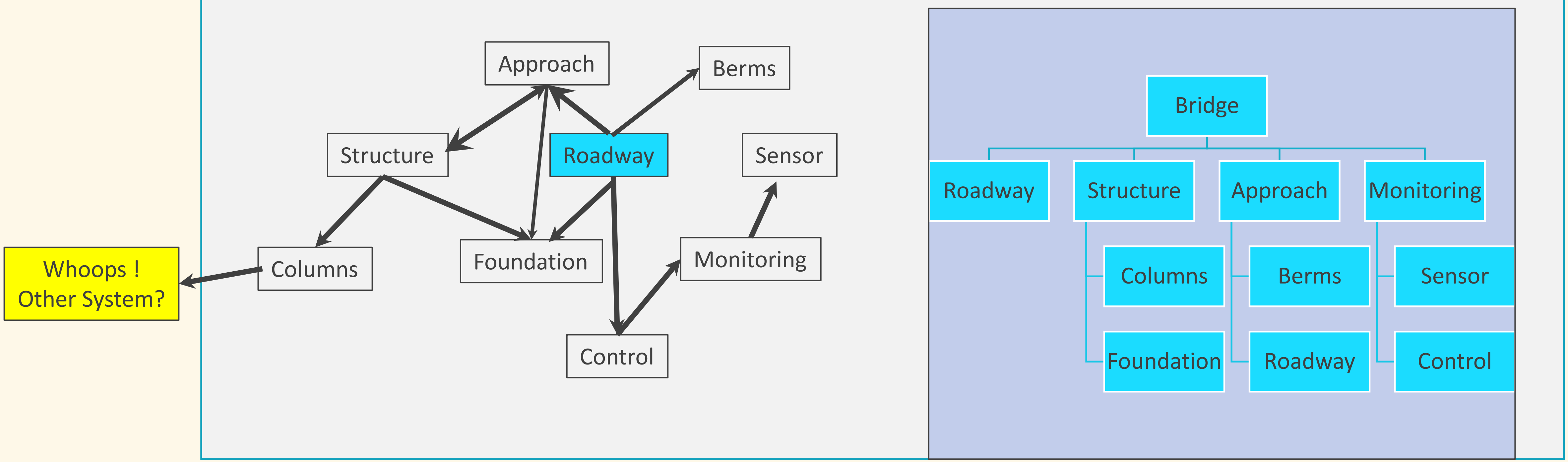
Table 1: Systems Engineers are usually a blend of Thinkers or Perspectives but people naturally lean one way or the other	
Synthesis ← Some of Each →	Analysis
Systems Thinkers Holistic, **bottom up** approach of **synthesizing** the behavior of the small parts, how it contributes to the larger. The ability to understand the cause and effect, cascading events, interconnections of actions and an ontology.	Systems Perspective Reductionist and hierarchy approach, **top down analysis**, inputs – process – output and the results; abstractions and models, learn and understand the system that includes parts, what is required for each part, and anticipating the emergent behavior.
Systems Professionals Built on Thinkers and Perspective – Certifications and University Degrees – the custodian of the Systems Engineering Profession. Systems Professional can determine the best way to deliver a solution, to include the processes, documentation, risks – even if it's a hybrid or new method.	
Systems Engineer Development – 70% experience/OJT, 20% mentoring, 10% training	

REFRESHER – HOLISTIC AND REDUCTIONIST

THINKER - **HOLISTIC** DISCOVER THE SYSTEM THROUGH OBSERVATION AND TRACING

PERSPECTIVE - **REDUCTIONIST** PERCEIVES IT AS A STRUCTURE AND HIERARCHY

Figure 48 How a Systems **Thinker** see Bridge, versus one with a Systems **Perspective**



THREE GENERATIONS OF SYSTEMS ENGINEERING – BRIAN’S OPINION

- First Generation of Systems Engineering 1920s to 1970s
  - Early to Mid 20th Century – Leaned towards a **Reductionist** Approach
  - Top Down, Decompose Requirements, break it down into a **hierarchy**
  - Works well when not certain how you’ll build it, or if no such system already exist
  - Result – many stove piped systems that didn’t connect well – but integration became more importance
- Second Generation of Systems Engineering – 1970s to 2000s
  - 21st Century – starting to become **more Holistic**
  - Virtualization, Service Oriented Architectures (SOA), Interoperability between Stove Pipes, Re-Use What Worked Before
  - Services and apps introduced daily and only the best will survive
  - In essence – users define their desired function and tap into the system that delivers it – streamlines the lifecycle
  - This is consistent with MBSE – that most systems and services already exist in some for – just model how you’ll integrate and use it
- Third Generation of Systems Engineering – we are in it
  - **More Holistic and Synthesis**, bottoms Up but **traditional Reductions still needed** – Requirements, Development, V&V, O&M, Disposition of Systems,
  - Vast Majority of Systems Capability is already built a ‘constellation of interconnected systems – and Systems Engineers simply need to tap into it
  - Requires defined interfaces and standards, improved modeling and simulation

DIGITAL INFORMATION ECO-SYSTEM

- Distributed, adaptive, open social-technical system(s) – self-organization, scalability, sustainability.
- Loosely coupled and defined interfaces - to allow connection.
- Digital environments – applications and services introduced and those services of value – **survive**.
- Enabled by Large Data, Interfaces, High Speed Networks, Wireless, Size/Weight/Power - need
- **Can SE Manage the Digital Information Eco System? What are the opportunities and the risks?**
- Our job as Systems Engineers is to manage this emerging Digital Eco-System and an interconnected world

Figure 6-2 Digital Information Eco Systems - Only the Strong Will Survive, and Beware of Predators

https://medium.com/@davidedjensen_99356/building-a-digital-ecosystem-for-the-planet-557c41225dc2

TRENDS - SYSTEMS, METHODS, ARTIFACTS RE-USE AND SHARING (AND THE DIGITAL ECOSYSTEM), VIRTUALIZATION

- Was Corrective, the Preventive, Optimizing **and now Predictive**
  - Large data samples, collection and sensors, processing power
  - Recall Deterministic and Stochastic, Black Swans, Finding and ‘Connecting Dots’
  - We can anticipate behavior and events better than ever

- Service Oriented Architecture (SOA)
  - Predominate during Client Server Architecture
  - Needs Transport BUS, distributed services called on when needed.
  - Defined and **loosely coupled** interfaces make it flexible and open
  - May be **(and is)** subsumed by more recent trends

- Model Based Systems Engineering (MBSE)
  - Uses systems and subsystems that are currently available and ‘taps’ into their capability – their Emergent Behavior instead of creating new.
  - Re-Use of Artifacts, Code, Process – to save time and money.
  - Still Need the regular activities and approvals of a lifecycle – Requirements, Ver and Val, Gate Approvals, Agreements, Integration
  - Interfaces must be well defined to allow connection
  - Continually Model the **Emergent Behavior** from these systems until it suits the user need.

TRENDS - SYSTEMS, METHODS, ARTIFACTS RE-USE AND SHARING (AND THE DIGITAL ECOSYSTEM),VIRTUALIZATION

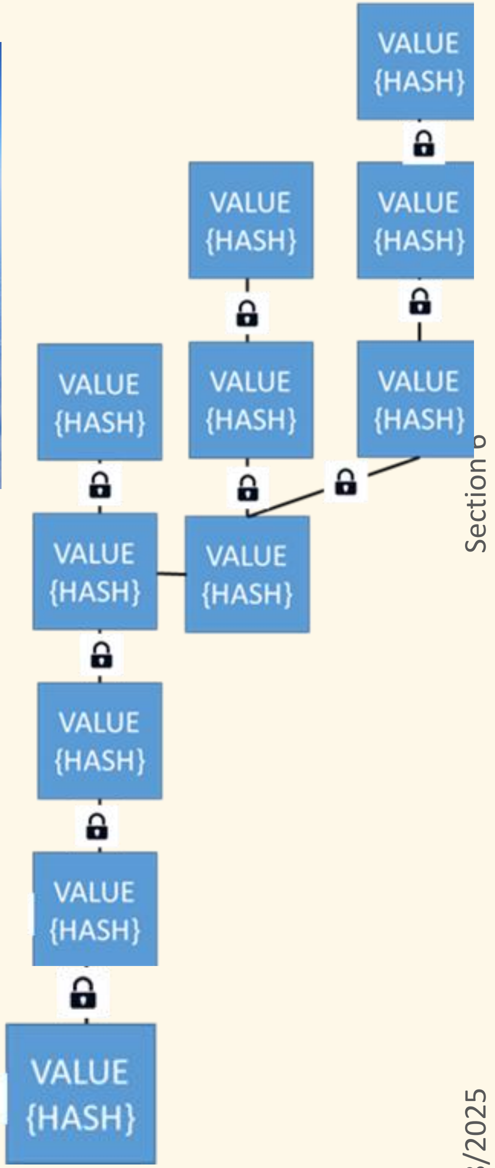
- Cloud Computing
  - Use What You Need – pay as you go – return when finished so others can use
  - Quickly Procure services – in minutes and not days
  - Predominant in Data Storage and Compute, also Security, Transport, Analysis
  - We May Not be coming to the point that we know the real cost of systems, esp IT
- For System Engineering
  - This does not circumvent processes, approvals – but essentialize it
  - Still need requirements and traceability
  - The System already exists – where service or part of the system will satisfy that particular requirement?
  - Not limited to traditional IT – now includes telecom, aerospace
  - Allows for improved systems integration
  - And predictive analysis
- But Not without challenges
  - Cloud Computing – various private, hybrid and public clouds are part of the digital eco-system
  - Becoming more holistic and evolutionary

Services are becoming more of a utility,
and used among assorted missions and business cases.

WHAT WE SHOULD DO –
DO IN THE VIRTUAL WORLD AS YOU WOULD DO IN THE PHYSICAL WORLD

- Strengthen or write policy pertaining to privacy of private and proprietary information
  - International, Federal, State and Local Law
  - Corporate Policy
- Discover and apply tools to verify the integrity and source of information on the Digital Information Eco-Systems.
- Technical Approach – may be counter to applying good standardization to systems as we discussed so far.
  - Mitigate risk by dividing risk among compartments – each needing authentication to pass from one compartment to the other.
  - Fortify Systems from intentional intrusion, spoofing, degradation, theft – and codify by law and strong penalties
  - Firewall Systems but also set traps, audit logs and alerts, cyber-AI, unstandardized configuration, differing protocols, equipment – so that just because one system is compromised doesn't mean others are too.
  - Periodically change configurations on system so if one is being discovered, it changes before harm can be done.

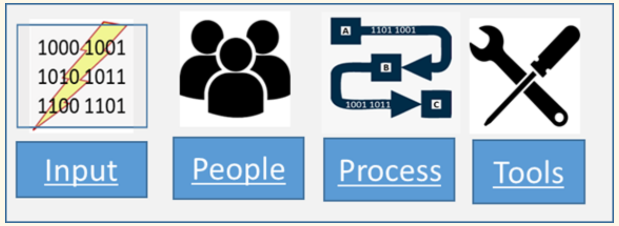
- The Digital Eco System suggests that systems are already built and only a function needs to integrate to it.
- The systems approach has tilted to a holistic approach, synthesizing, bottoms up – i.e. **Systems Thinker**



An anti-electricity cartoon from the 1900s. This is how some of y'all look talking about 5G.

FINAL SUMMARY

- Get the Foundation right – Organizational, Roles and Responsibilities, Leadership and Management, Vision and Purpose, Communicate
- Era of Enlightenment
  - Free to Think, Science
  - Emergent Behavior – large systems drew resources from its smaller parts
- Early Systems – 20th Century
- Systems are – transportation, communication, medical system, aerospace
- Systems Vee and Lifecycle – pick most suitable – many to choose
  - Need, Concept, Requirements – trace to end
  - Decompose and System Hierarchy
  - Design – Model and Simulate
  - Obtain Approvals and a Consensus – Gate Reviews
  - Interfaces and Integration
  - VER and VAL
  - Ops and Maintenance
- Trends – Eco Systems and in interconnected world, more holistic – **manage it**



SYSTEMS ENGINEERING CHECKLIST FOR SUCCESS

- **Remember the foundation** – organization, leadership, management, vision, purpose roles and responsibility, effective communication, everyone counts. Recognize your weaknesses and work to improve, let others perform better than you can, and vice versa. Compliment in public, counsel in private.

- **Think People, Process, Tools** – don't take any of these for granted – work hard to synchronize their activities, take care of all of them. Give your team what they need to succeed.

- **Think Cost, Schedule, Performance and Risk** - most suitable approach is not always the least expensive, or the fastest. Remember Murphy.

- **Have a plan that is suitable to your project and system** – what are you doing and why are you doing it? Plenty of methods and models to choose or make a hybrid.

- Consider the top-down **reductionist** and the **holistic** approach – both are effective if used properly – both require a plan, requirements and objective, resources, managed risks, accurate documentation and records

- **Special Attention to the 'Digital Information Eco Systems'** – know system boundaries, maintain control of your system and project, your apps and services. Know interfaces, and your agreements with others.

- **Risks of the Digital Information Eco**-Systems – Do in the Virtual World as You Would Do in the Physical World – firewall entries, compartmentalize the system, subsystem, don't have too much risk in any one compartment and encrypt and authenticate all entries

- **Re-Use** – Code, Scripts, Material, Plans, Lessons Learned, Processes that work. Use and build on to existing capability – think MBSE and Model and Simulation.

- **Encourage but manage innovation** – especially in the 3rd Generation of Systems Engineering – don't allow things to stall out because of thwarted motivation but don't let things cascade beyond your control. Think the Biz, Tech, Policy Venn Chart

BIBLIOGRAPHY (1)

- 97th Congress of the United States of America. (1982, May 21). *The Telecommunications Act of 1982 (H.R. 5158, 97th Congress): Provisions and Controversies.* Retrieved from Every CRCReport: https://www.everycrsreport.com/reports/IP182.html
- Andrews, E. (2015, February 5). *9 Things You May Not Know About the Ancient Sumerians*. Retrieved from History Channel: https://www.history.com/news/9-things-you-may-not-know-about-the-ancient-sumerians
- ANSI. (2020, May 9). *About ANSI*. Retrieved from American Nations Standards Institute: https://www.ansi.org/
- Armstrong, P. (n.d.). *Blooms's Taxonomy*. Retrieved from Venderbilt University Center for Teaching: https://cft.vanderbilt.edu/guides-sub-pages/blooms-taxonomy/
- ASQ. (2020, May 1). *American Society of Quality*. Retrieved from American Society of Quality: https://asq.org/
- Assistant Secretary of Defense/R&E. (2011, April). Technology Readiness Level (TRL) (Technology Readiness Assessment (TRA) Guidance. The Pentagon, Virginia, USA.
- Audrertsch, T. (2005). The Digital Ecosystesm Research Vision: 2010 and Beyond. *Semantic Scholar*.
- Bell, J., & Defense, D. S. (2007, March 10). *Acquisition Notes.* Retrieved from http://acqnotes.com/wp-content/uploads/2014/09/DoD-4151.18-H-Depot-Maintenance-Capacity-Utilization-Measurement-%E2%80%93-10-Mar-2007-Change-1.pdf
- Body, I. S. (2020). *ISO*. Retrieved from https://www.iso.org/home.html
- Branson, R. (2016, March 17). *My top10 quotes on risk*. Retrieved from Virgin: https://www.virgin.com/richard-branson/my-top-10-quotes-risk
- Brock, G. D. (1999). Trachtenberg School of Public Policy & Public Adminstration . *TCOM Class Presentation* . Virginia, U.S.A.: The George Washington University (Columbian College of Arts and Sciences).
- Burge, D. S. (2011). *The Systems Engineering Tool Box.* Warwickshire: Burge Hughes Walsh.
- C4 Model. (n.d.). *C4 Model for Visualising Software Architeture - Contezxt, Containers, Components and Code*. Retrieved from C4 Model: https://c4model.com/
- Carrick, P. (2018, November 5). *Who Invented the Magnetic Compass?* Retrieved from https://historydaily.org/who-invented-the-magnetic-compass: https://historydaily.org/who-invented-the-magnetic-compass
- CMMI Institute. (2020, May 9). *CMMI*. Retrieved from Capability Maturity Model Integration: https://cmmiinstitute.com/
- Cockburn, A. (2007). *Agile Software Development - The Cooperative Game (2nd Ed) .* Boston: Pearson Education, Inc.
- DAU. (2020, May 9). *About Us*. Retrieved from Defense Acquisition University: https://www.dau.edu/about
- Defense Acquisition University. (2020, May). *Life Cycle Cost*. Retrieved from DAU: https://www.dau.edu/acquipedia/Pages/search.aspx?k=cost%20of%20doing%20system%20engineering#k=systems%20engineering%20cost
- Deming, W. Edwards. (n.d.). *The W. Edwards Deming Institute*. Retrieved from https://deming.org/
- (2013). *Deterministics vs. stochastic models.* Raleigh: NC State.
- DoD CIO. (2010, August). *The DoDAF Architecture Framework Version 2.02.* Retrieved from U.S. Department of Defense - CIO: https://dodcio.defense.gov/Library/DoD-Architecture-Framework/

BIBLIOGRAPHY (2)

- *Engineering*. (2020, June 26). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Engineering
- Esper, M. (2020, May 6). FCC's Decision Puts GPS at Risk. *Wall Street Journal*, p. 14.
- Estefan, J. A. (2008, May 23). *Survey of MBSE Methodologies.* Retrieved from IAN Sommerville: https://iansommerville.com/software-engineering-book/web/spiral-model/
- Freshworks. (2020, April). *What is ITSM?* Retrieved from Fresh Service: https://freshservice.com/itsm
- Gates, B. (n.d.). Retrieved from Brainy Quotes: https://www.brainyquote.com/lists/authors/top-10-bill-gates-quotes
- Georgiev, G. (2019, Nov 5). *"All Models Are Wrong" Does Not Mean What You Think It Means*. Retrieved from The Startup: https://medium.com/swlh/all-models-are-wrong-does-not-mean-what-you-think-it-means-610390c40c9c
- Georgiev, G. (2019, Nov 5). *The Start Up*. Retrieved from Medium: https://medium.com/swlh/all-models-are-wrong-does-not-mean-what-you-think-it-means-610390c40c9c
- Glenn, C., & Gray, L. (2010). *The Hodges Harbrace Handbook (17ed).* Boston: Wadsworth.
- Hersey, P., & Blanchard, K. (1969). *Management of Organizational Behaviour: Utilizing Human Resources.* Englewood Cliffs: Prentice-Hall.
- Hirshorn, Steven R. (2007). *NASA System Engineering Handbook.* Washington, DC: National Aeronautics Space Administration .
- History.com Editors. (2020, February 21). *History Channel*. Retrieved from Enlightement: https://www.history.com/topics/british-history/enlightenment
- Honour, E. C. (2013). *Systems engineering return on investment.* Adelaide: University of South Australia - Defence and Systems Institute - School of Electrical and Information Engineering.
- IBM. (2020, May 11). *Overview of Rational DOORS*. Retrieved from IBM Knowledge Center: https://www.ibm.com/support/knowledgecenter/SSYQBZ_9.5.0/com.ibm.doors.requirements.doc/topics/c_welcome.html
- *IDEF - Integrated DEFinition Methods (IDEF)*. (2020, April). Retrieved from IDEF0 Funtion Modeling Method: https://www.idef.com/idefo-function_modeling_method/
- IDEF. (n.d.). *IDEF Family of Methods*. Retrieved from Integrated DEFinition Methods (IDEF): https://www.idef.com/
- IEEE. (2020, April). *Institute of Electrical and Electronics Engineers*. Retrieved from IEEE Advancing Technology for Humanity: https://www.ieee.org/
- INCOSE. (2020, May 9). *INCOSE - About Systems Engineering*. Retrieved from International Council on Systems Engineering: https://www.incose.org/about-systems-engineering/about-systems-engineering
- *Innovative Architects*. (2020, July 10). Retrieved from http://local.innovativearchitects.com/
- International Council on Systems Engineering (INCOSE). (2011). *Systems Engineering Handbook - A Guide for System Life Cycle Processes and Activities.* San Diego: SE Handbook Working Group.
- International Council on Systems Engineering (INCOSE). (2011). Systems Engineering Handbook - A Guide for System Life Cycle Processes and Activities. In I. SE Handbook Working Group, *Systems Engineering Handbook* (p. 5). San Diego.
- International Council on Systems Engineering (INCOSE). (2020, April 4). *Collaboration Portal "Connect"*. Retrieved from https://www.incose.org/
- ISO. (2020, May 9). *ISO Home*. Retrieved from International Council on Systems Engineering: https://www.iso.org/home.html
- ISO 9000. (2015). *International Standards Organization (ISO)*. Retrieved from ISO 9000 Family Quality Management: https://www.iso.org/iso-9001-quality-management.html
- ISO/IEC. (2020, May 11). *ISO/IEC 27001 Information Security Management*. Retrieved from International Standards Organization: https://www.iso.org/isoiec-27001-information-security.html
- ISO/IEC/IEEE 42010:2011. (2017). *International Standards Organization*. Retrieved from Systems and Software Engineering - Architecture Descriptions: https://www.iso.org/standard/50508.html
- ITU. (2020, May 9). *About ITU*. Retrieved from International Telecommunication Union (ITU): https://www.itu.int/en/Pages/default.aspx
- Kant, I. (1996, June 28). *What is Enlightenment.* Retrieved from Columbia University - Sources of Medieval History: http://www.columbia.edu/acis/ets/CCREAD/etscc/kant.html
- Knighton, A. (2017, Nov 19). *How The Spear Transformed Warfare - From Ancient Times To The Age of Gunpower*. Retrieved from War History Online: https://www.warhistoryonline.com/instant-articles/spear-transformed-warfare-mm.html
- Kobren, B. C. (2017, May 11). *Updated DoD Acquisition Life Cycle Wall Chart*. Retrieved from Defense Acquisition University: https://www.dau.edu/training/career-development/logistics/blog/Updated-DoD-Acquisition-Life-Cycle-Wall-Chart

BIBLIOGRAPHY (3)

- Le Merle, M. C., & Davis, A. (2017). *Corporate Innovation in the Fifth Era.* Corte Madera, CA: Cartwright.
- Madhavan, G. (2020, May 5). The COVID Recovery Comes Down to Engineering. *Wall Street Journal*, p. 14.
- Magee, John F. (1964). *Decision Trees for Decision Making*. Retrieved from Harvard Business Review: https://hbr.org/1964/07/decision-trees-for-decision-making
- McConnell, John (V Adm). (2008, Sept 15). *ICD 503.* Retrieved from Director of National Intelligence ICD 503 Intelligence Community Information Technology Systems Security Risk Managment, Certification And Accreditation: https://www.dni.gov/files/documents/ICD/ICD_503.pdf
- Monat, Jamie; Gannon, Thomas. (2018, Sep 12). *Applying Systems Thinking to Engineering and Design.* Retrieved from Worcester Polytechnic Institute : https://www.mdpi.com/2079-8954/6/3/34
- *National Institute of Standards and Technology*. (n.d.). Retrieved from Digital Ecosystem - Search Results: https://www.nist.gov/fusion-search?s=digital%20ecosystem&start=30&sort=relevance
- NCSU. (2013). *Deterministic vs. stochastic models.* Retrieved from NS State University - Dept of Statistics: https://www4.stat.ncsu.edu/~gross/BIO560%20webpage/slides/Jan102013.pdf
- NPCA. (2010, March). *Corrective Action and Root Cause Analysis*. Retrieved from Slideshare: https://www.slideshare.net/
- OD&I. (2008, September 15). *OD&I Documents.* Retrieved from Office of the Director of National Intelligence: https://www.dni.gov/files/documents/ICD/ICD_503.pdf
- PMI. (2020). *PMI Entry Page*. Retrieved from Project Management Institute: https://www.pmi.org/
- Santayana, G. (n.d.). *Wikiquote*. Retrieved from George Santayana: https://en.wikiquote.org/wiki/George_Santayana
- Schulze, E. (2019, Jan 17). *CNBC*. Retrieved from Everything you need to know about the Fourth Industrial Revolution: https://www.cnbc.com/2019/01/16/fourth-industrial-revolution-explained-davos-2019.html
- Smith, A. (1776). *The Wealth of Nations.* Scotland: Strahan, William; Cadell, Thomas.
- Strickler, Kris. (2020, July). *Oregon.gov*. Retrieved from Oregon Department of Transportation: https://www.oregon.gov/odot/About/Pages/Library.aspx
- Surg, J. C. (2014, January 25). *US National Library of Medicine, National Institutes of Health.* Retrieved from PubMed: https://www.ncbi.nlm.nih.gov/pubmed/24406559
- SysML. (n.d.). *What is SYSML*. Retrieved from Object Management Group (OMG): https://www.omgsysml.org/index.htm
- Szalay, J. (2018, September 18). *Who Invented Zero*. Retrieved from Live Science: https://www.livescience.com/27853-who-invented-zero.html
- UML. (n.d.). *UML Profiles, Specifications, Training*. Retrieved from Unified Modeling Language: https://www.uml.org/
- Union, I. T. (2020, April). *ITU - Committed to connecting the world*. Retrieved from https://www.itu.int/en/Pages/default.aspx
- Vanderbilt University. (2020, March). *Center for Teaching*. Retrieved from https://cft.vanderbilt.edu/guides-sub-pages/blooms-taxonomy/
- *Wikipedia*. (2019, November 27). Retrieved from Digital Ecosystem: https://en.wikipedia.org/wiki/Digital_ecosystem
- Wikipedia. (2020, March 12). *Babylonian Mathematics*. Retrieved from https://en.wikipedia.org/wiki/Babylonian_mathematics
- Wikipedia. (2020, April 25). *Hubble Space Telescope*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Hubble_Space_Telescope
- Wikipedia. (2020, April 17). *Reductionism (ontological)*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Reductionism
- Wikipedia. (2020, February 21). *United States Military Standard*. Retrieved from WIKIPEDIA: https://en.wikipedia.org/wiki/United_States_Military_Standard
- Wikipedia ITSM. (2020, April 8). *IT Service Management*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/IT_service_management
- Wikipedia Six Sigma. (2020, March 15). *Wikipedia*. Retrieved from Six SIgma: https://en.wikipedia.org/wiki/Six_Sigma
- Woolsey, James. (2020, May 7). *About DAU*. Retrieved from Defense Acquisition University (DAU): https://www.dau.edu/about
- Zackman, J. A. (2008). *The Concise Definition of The Zackman Frameworks by: John A. Zachman.* Retrieved from Zackman International Enterprise Architecture: https://www.zachman.com/about-the-zachman-framework

VERSIONS

Ver	Date	Change	By	
1.0	2 Aug 2021	Base Version	Brian S	
1.1	8/18/2025	Edits & Typo Corrections	Brian S	