

Agile Approach to Accelerate Product Development Using an MVP Framework

Never Stand Still

Capability Systems Centre, Australian Cybersecurity Centre, School of Engineering and Information Technology



Boris Kazakevich – Principal Engineer, Moog Australia

- Professional engineer with 30+ years of experience in the industry
- Mechanical Engineer by trade, specialising in motion control
- Managed complex projects, and led engineering teams
- Took an interest in Systems Engineering, formalising it in study MsSysEng (UNSW)



Dr. Keith Joiner, CSC – Senior Lecturer and Researcher, UNSW

- Air Force aeronautical engineer, project manager and teacher for 30 years
- Former Director-General of Test and Evaluation
- BEng Aeronautical Engineering, a Master of Aerospace Systems Engineering, Ph.D. in Calculus
- Teaches master programs for systems engineering, project management and cybersecurity, and to undergraduates for aircraft design



Agenda

1. Background and Complexity
2. Existing Model Challenges
3. Agile methodology
4. Proposed Concept
5. Summary

Background

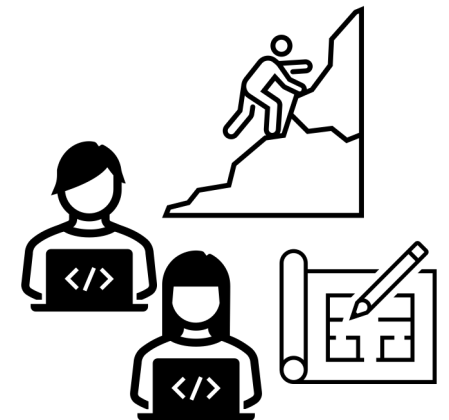
Voice of Customer

- Capability development is taking too long
- Capability development attracts a significant cost
- Capability development is inflexible, little tolerance to changes
- Capability development is very transactional, and milestone focused



Voice of Process

- Requirements Engineering takes a significant portion of development
- Locking requirements early leads to untested assumptions and risks
- Little opportunity to innovate due to the expectation the system works on the first attempt
- Testing at the end of development focuses on the acceptance and verification leading either to compromises or stressful rework



Complexity

“Complexity characterises the behaviour of a system or model whose components interact in multiple ways and follow local rules, leading to non-linearity, randomness, collective dynamics, hierarchy, and emergence” (source Wikipedia)

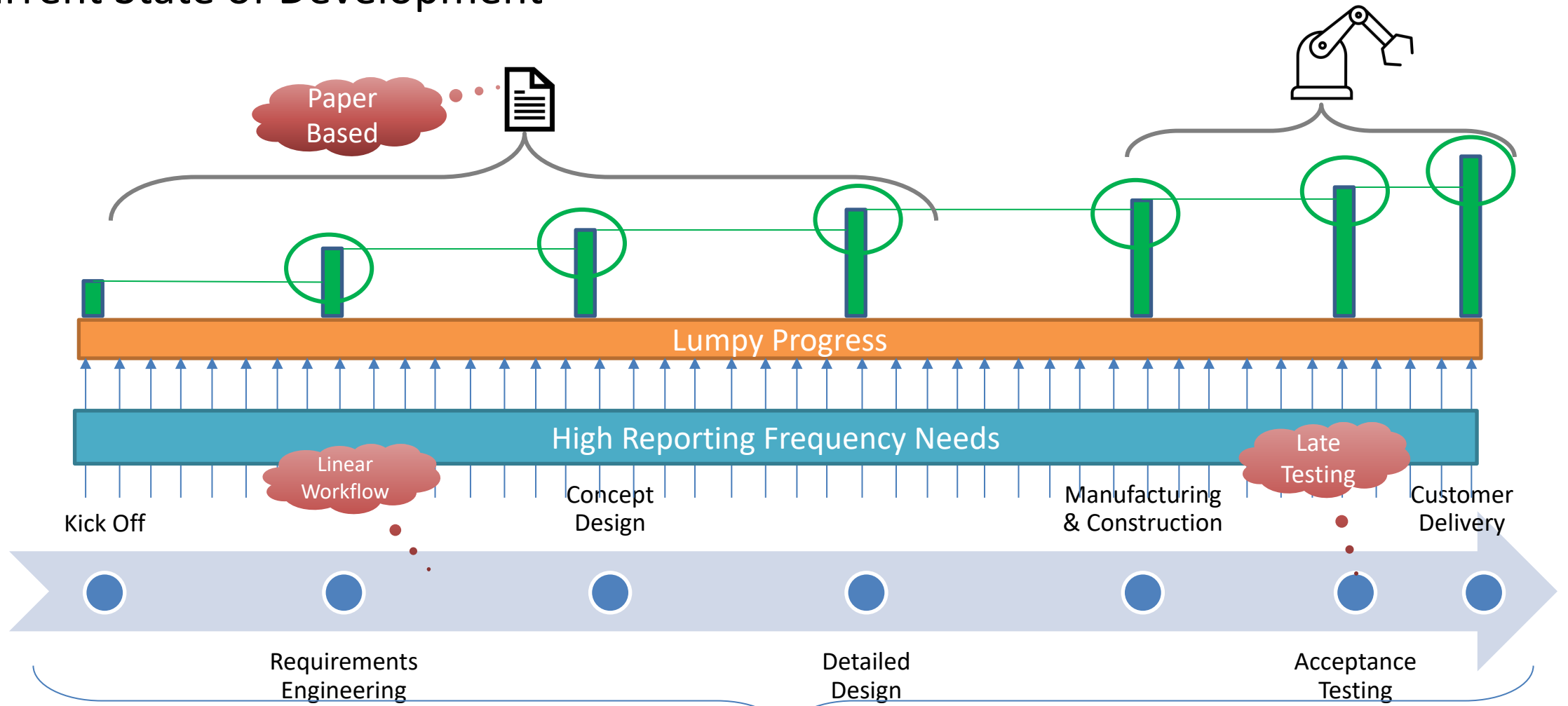
Modern Capability

- Combination of advanced subsystems of various maturity level
- Software Centric (Firmware, Control, Safety, Security)
- Integrates into System of Systems (Mechanical, Electrical, Software)
- Expected to be Modular, Adaptable, Resilient
- Automated now, Autonomous tomorrow



“... traditional, linear project management tools and techniques, while still necessary, are often insufficient to manage the complexities of 21st-century projects...” (International Centre for Complex Project Management)

Current State of Development

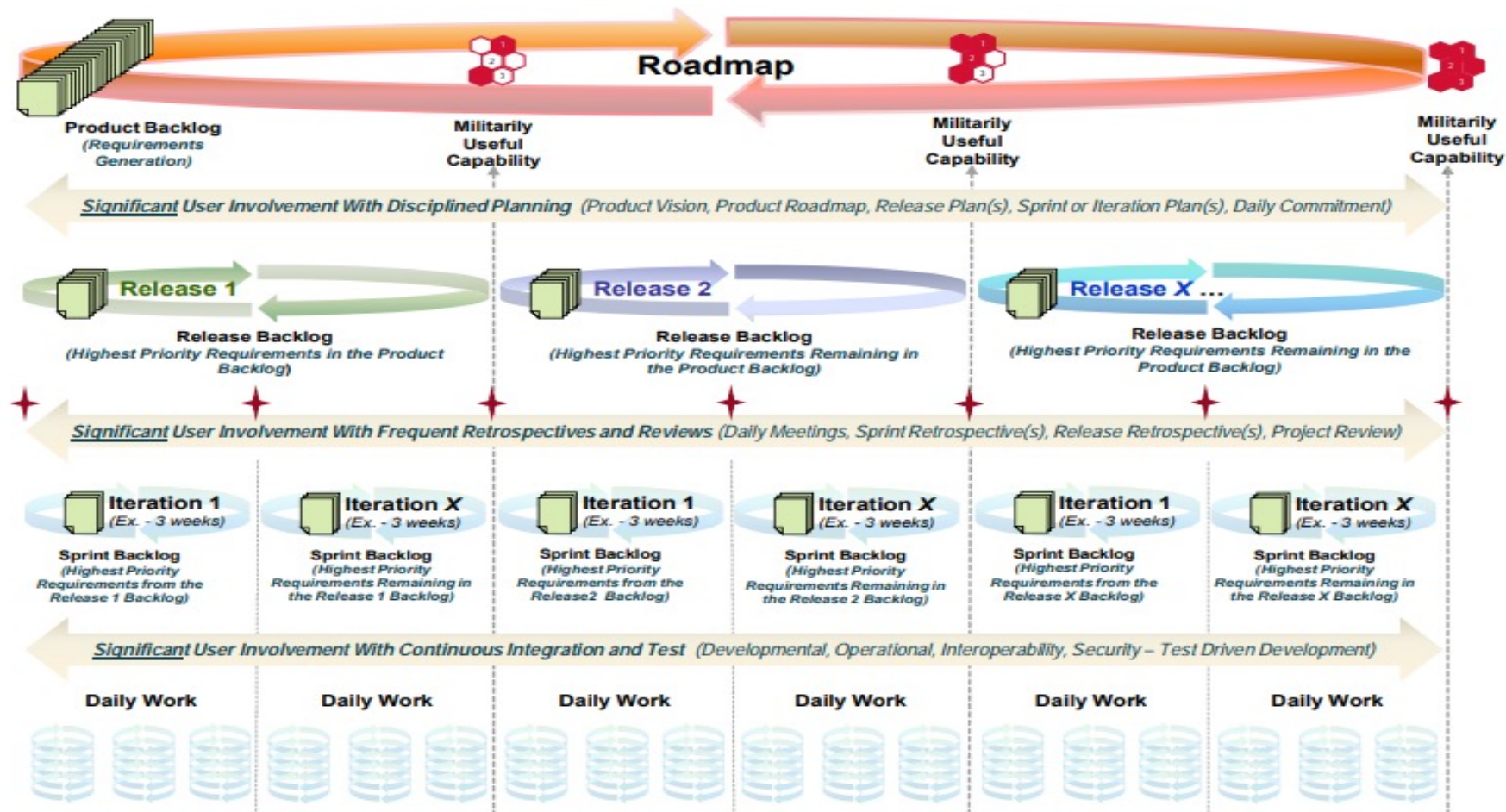


Milestones






3-5 years

Agile Methodology

“.. a project management philosophy characterised by highly skilled workers, working in self-managed teams to create products and services that are effective, efficient and value added...” (Kaitlynn M. Castelle et al)



Waterfall and Agile

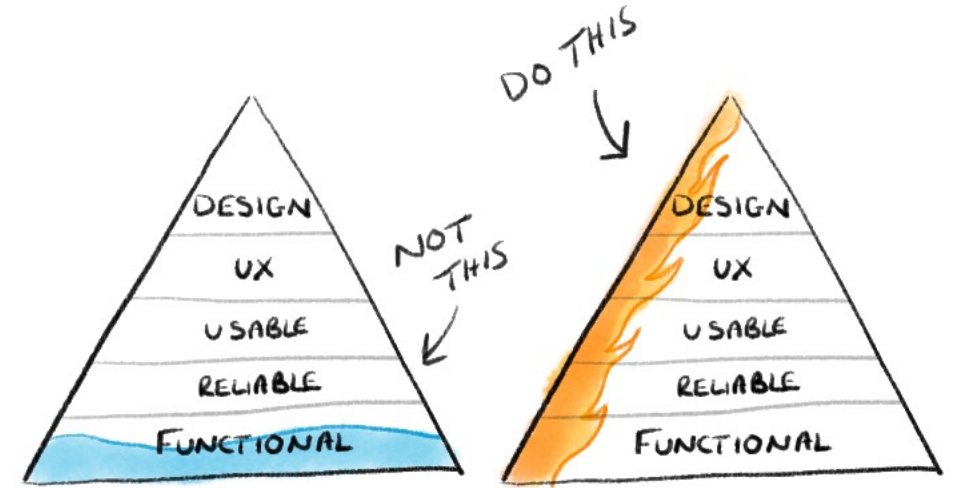
	Waterfall	Agile
 <p>Planning is critical</p>	<p>Plan the work—especially the budget, schedule, and deliverables—to the maximum extent possible before beginning any design or code.</p>	<ul style="list-style-type: none"> • Near-term plans contain more detail, while plans further out on the time horizon contain fewer details. • The overall vision is broken down into a roadmap, which is further broken down into release plans, which are further broken down into sprint or iteration plans, which are further broken down into daily plans. • Requirements are prioritized. • Cost and schedule estimates are prepared for each capability at a high level. Relative estimation versus absolute estimation is employed. • Frequent planning sessions (at the beginning of each iteration) result in detailed, high-fidelity plans. • Risks are assessed and risk mitigation influences planning.
 <p>Managed Requirements</p>	<p>Lock down requirements to prevent gold-plating and scope creep.</p>	<ul style="list-style-type: none"> • No requirements can be added to an iteration once it has started. • New requirements are evaluated by the stakeholders and prioritized thus preventing gold-plating and scope creep.
 <p>Ongoing Reporting</p>	<p>Institute multiple reviews to provide senior leadership oversight as well as to serve as gates for continued work.</p>	<ul style="list-style-type: none"> • The customer is involved in all aspects of planning and testing. Customer (in the form of the product owner) is involved daily. • There are reviews at the end of each iteration that serve as gates to further work.
 <p>Task Completing</p>	<p>Move forward in a step-by-step, sequential manner and only when all parts of the previous steps were complete.</p>	<ul style="list-style-type: none"> • The code base is integrated and tested daily. • The code base must pass all tests before and after integration. Regression testing is typically done each night.
 <p>Documents managed</p>	<p>Capture all details with extensive documentation.</p>	<ul style="list-style-type: none"> • There is an overall plan. • There are requirements descriptions. • There are cost and schedule estimates. • There are risk assessments. • There is training material (as appropriate). • There is documentation (as appropriate). • There are lessons learned (based on retrospectives).

Adapted from Palmquist, 2013

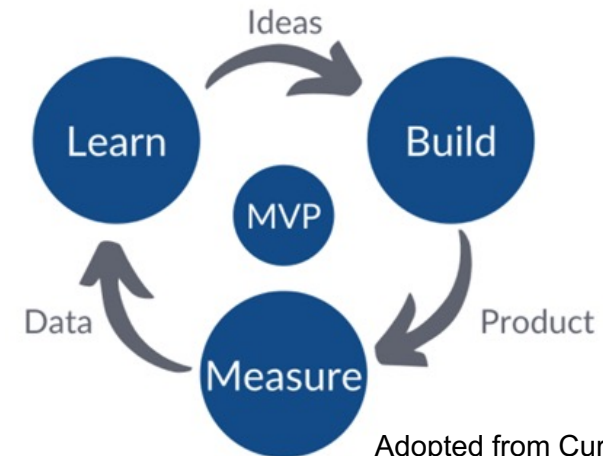
Minimum Viable Product

“That version of a new product which allows a team to collect the maximum amount of validated learning about customers with the least effort.” (Eric Reis)

- Focus on the core issue needed to be solved for the customer
- Rapid and ongoing testing within a set budget
- Market validation in real-time with real users
- Shorter development time
- Reduced cost because of shorter development time and focused effort

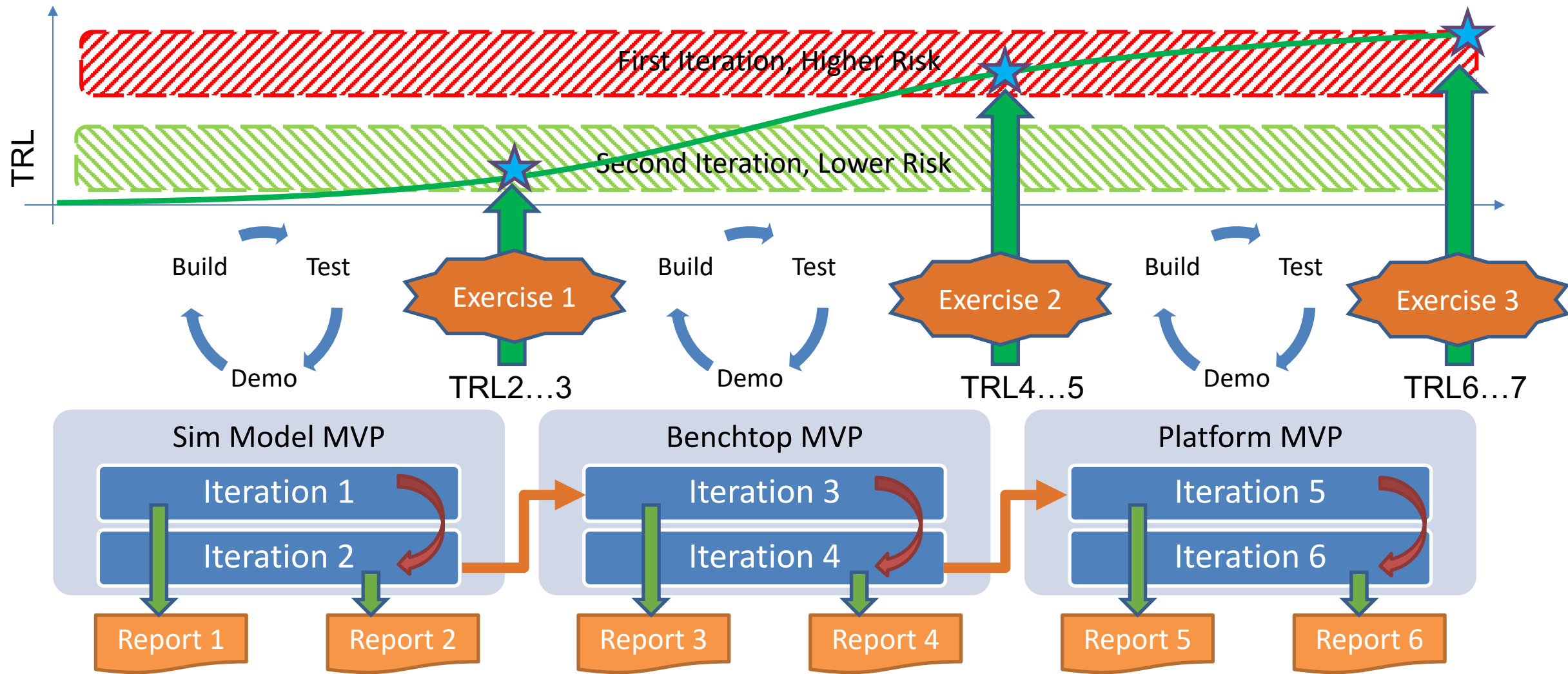


Adopted from Optimus (2023)

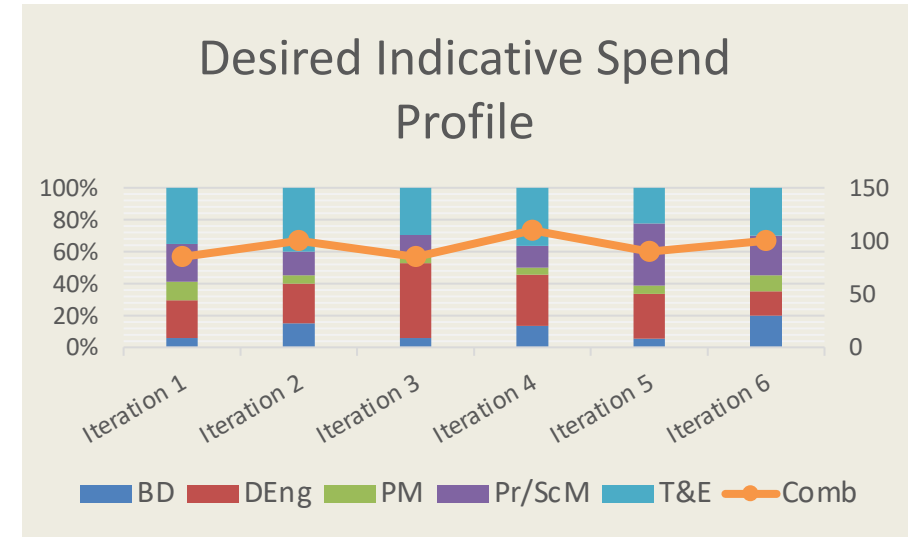
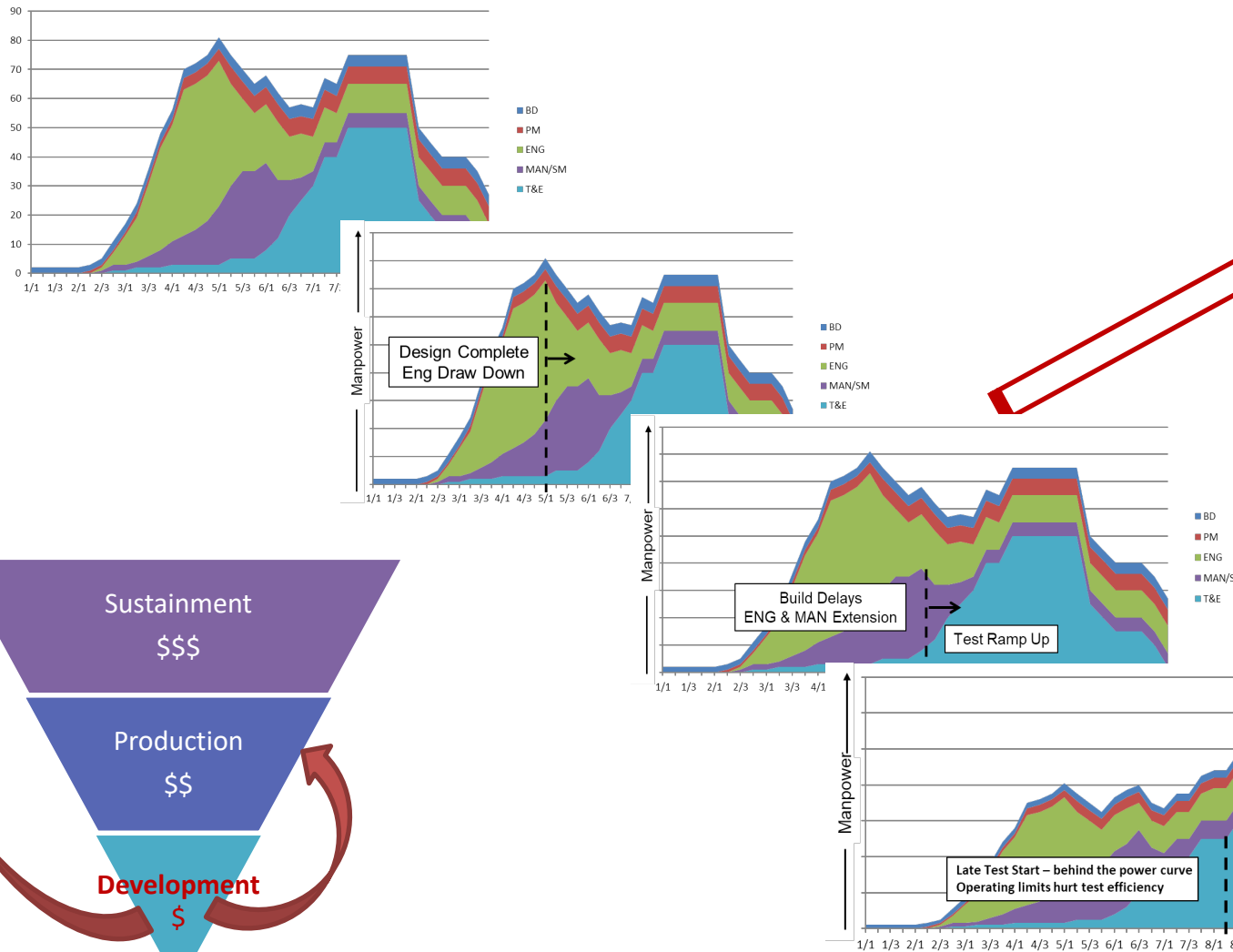


Adopted from Curtis, B.(2020)

MVP-based Capability Development Model



Proposed Cost Model – Balanced Spend



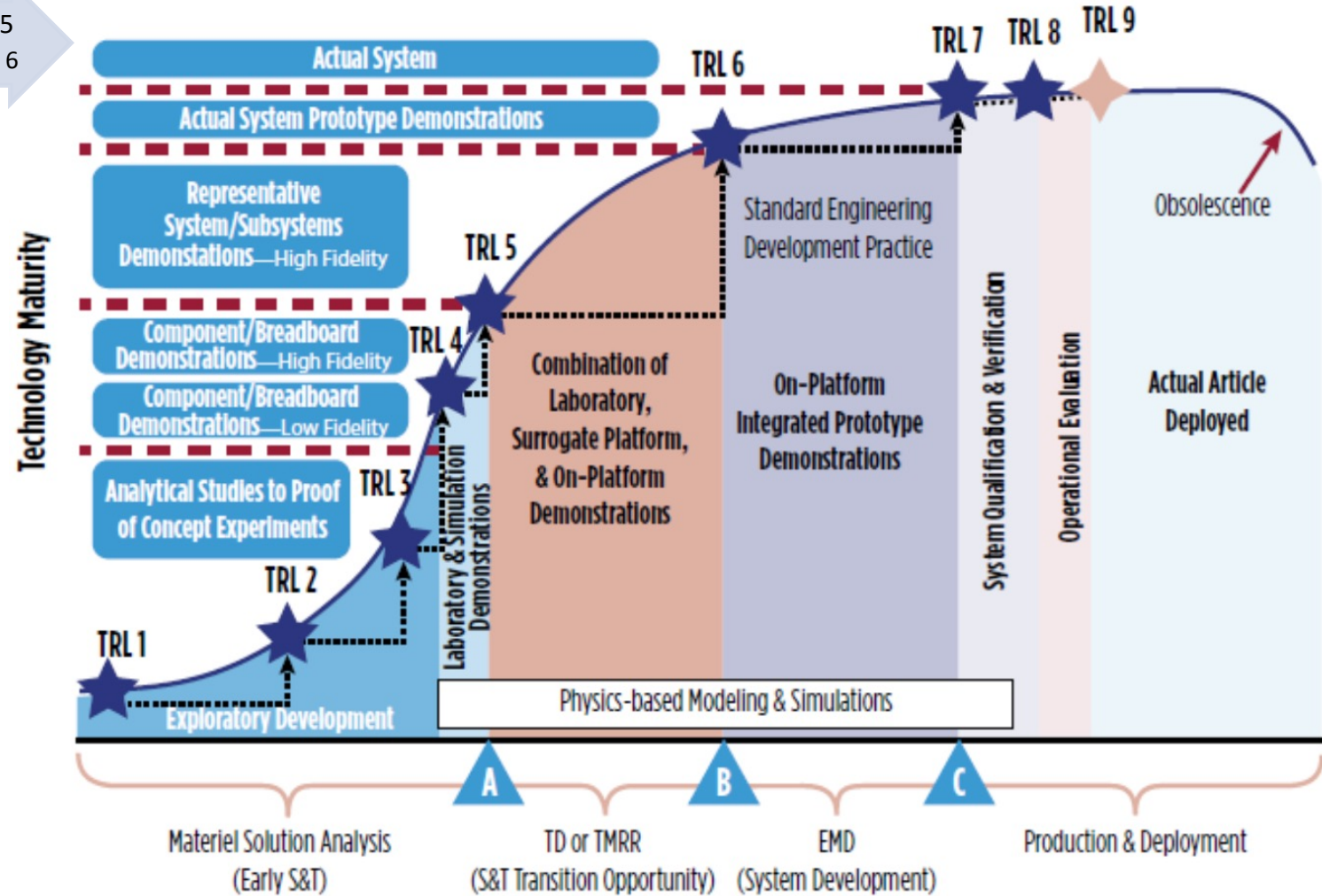
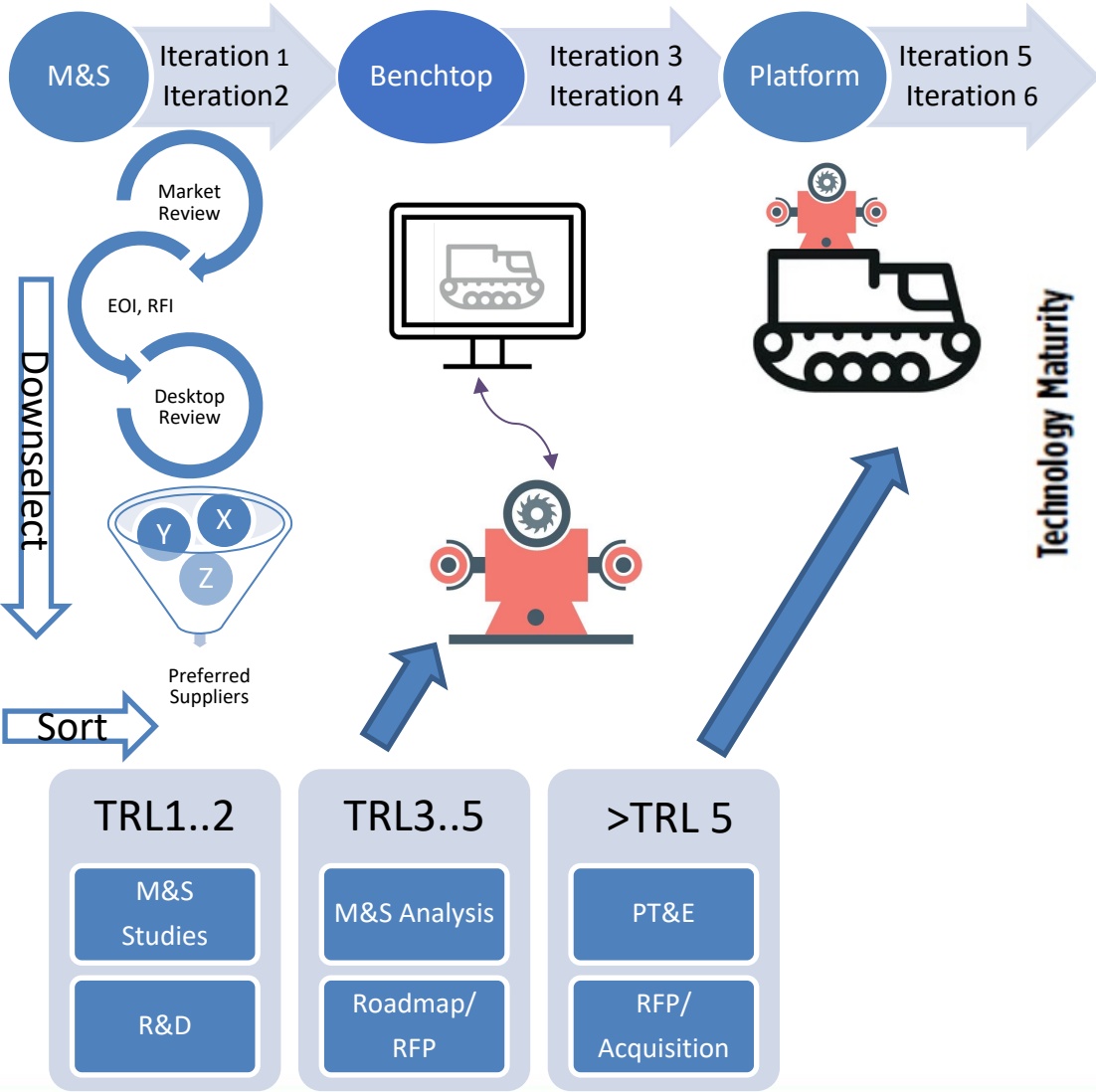
Business Development Design Engineering Project Management Procurement Supply Chain Management Test and Evaluation Combined Spend

Design – Manufacture – Test – Produce The Seven Buckets

- 1 The right people in the right place at the right time
- 2 Determining what you're going to build – the design
- 3 Determining validation & verification requirements (T&E SOW)
- 4 Determining data requirements
- 5 Determining instrumentation requirements
- 6 Integrated manufacture & test
- 7 Efficient test planning & execution

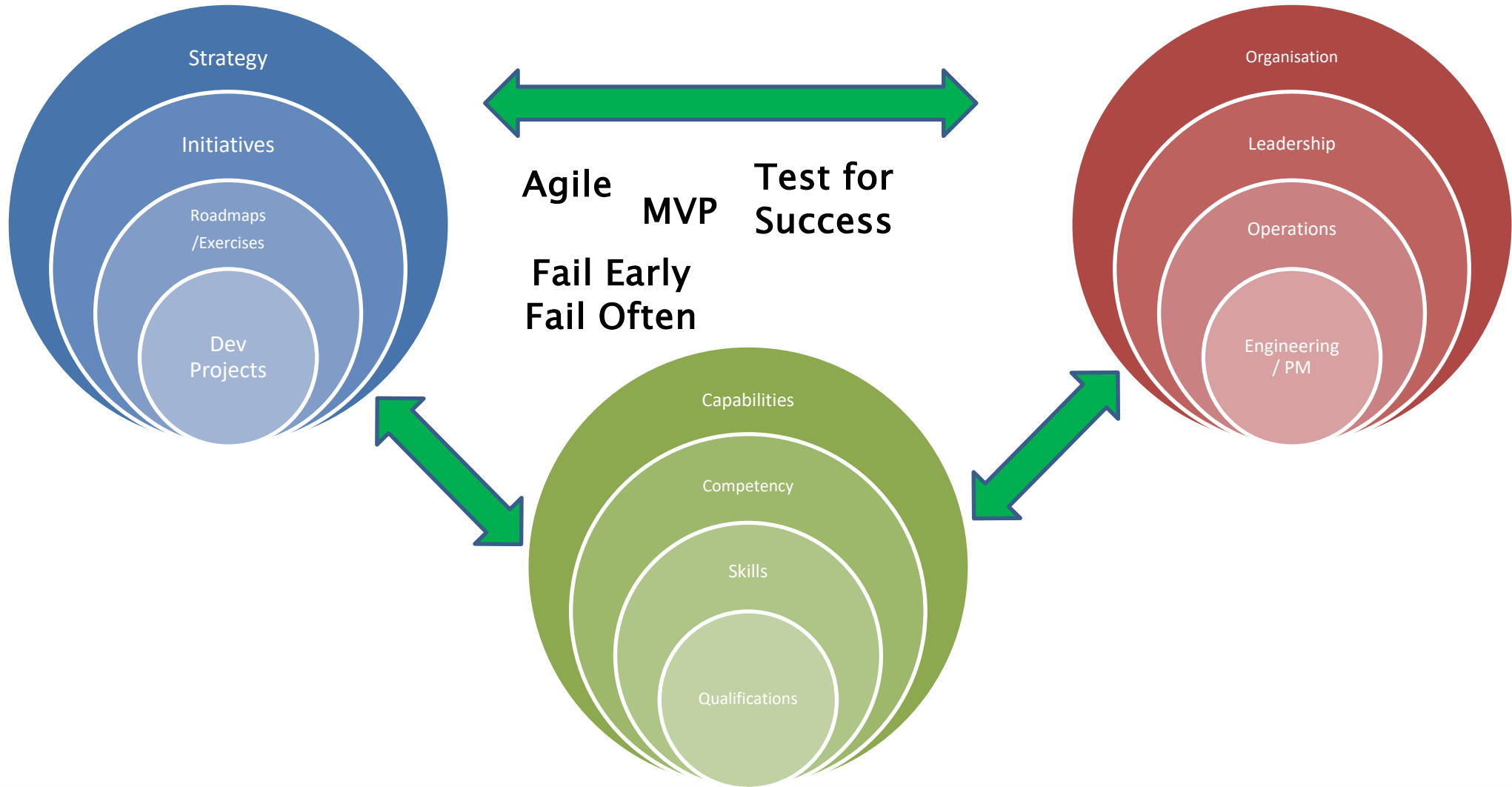
Adapted from Tomeny, T.E. ITEA Conference 2015, based aerospace larger projects

Change in Solicitation Model



Adopted from Copeland et al (2015)

Business Model



Change Management

Current

- Develop Detailed Requirements Upfront
- Project Management on Fixed Milestones reports
- Deliver Final Capability without iterations
- Business Development delivers reports only
- Rigorous management of the engineering team
- Paper-based reporting
- Lessons learned at the end

Transitioning

- Agile Training across the organisation
- Business Management Systems alignment with Agile methodology
- Focus on behavioural change
- Process corrections using reflections and feedback
- Practice, practice, practice

Future

- Requirements refinement through activity
- Continuous capability growth
- Business and Engineering work cooperatively
- Activity-based reporting
- Focus on technical competencies through self-organised team
- “Maximise the amount of work note done to drive simplicity”
- Reflections drive continuous improvements and behavioural changes

Summary

- Proposed new MVP-based Framework for Defence Acquisitions (more details in referenced paper)
- Quick Survey using QR-code on presented material and interest in attending Face Validity Workshop
- Alternatively welcome exploring or commencing the activity under proposed framework



[Survey Link](#)