scram™

SCHEDULE COMPLIANCE RISK ASSESSMENT METHODOLOGY

High Maturity, But Yet You're Late?

Presented by:

Emanuel R. Baker, Ph.D. – Software Engineering Consultants, Inc.

Betsy Clark, Ph.D. - Software Metrics, Inc.

Adrian Pitman - Capability Acquisition and Sustainment Group

Angela Tuffley – RedBay Consulting Pty Ltd.

Agenda

- The problem
- The remedy
- How it works
- Outputs
- Conclusion

© Copyright 2018 Commonwealth of Australia

THE PROBLEM

© Copyright 2018 Commonwealth of Australia

Problem - 1:

What? Late Again?

© Copyright 2018 Commonwealth of Australia

Problem - 2:

- Systems getting more complex
  - Taking longer to develop
  - Late deliveries are too common
  - Getting far more costly with cost overruns too common
- Notable examples
  - California High Speed Rail
  - Joint Strike Fighter (F-35)
  - ACA web-based user interface
  - Los Angeles DWP billing system
- Attempts to correct
  - Adopt CMMI, ISO 9000, etc.
  - Quality has improved
  - Hasn't necessarily fixed the late delivery problem

© Copyright 2018 Commonwealth of Australia

Historical view of development cycle time

- Aircraft development cycle durations from contract award to IOC
  - World War II: 20 months
  - 1960s/1970s: ~ 70 months
  - 1980s/1990s: ~ 9 years
  - Today: 15 or more years
- Why?
  - World War II
    - Aircraft was relatively simple (compared to today)
    - Aircraft was airframe, engine, and armament
  - Today
    - Aircraft can be characterized as warfighter inside a flying computer
    - Aircraft functions are controlled by mega-millions lines of code
    - IOC now harder to estimate

© Copyright 2018 Commonwealth of Australia

What's the hangup?

- CMMI: a best practices model
  - Characterizes the kinds of practices effective organizations employ to develop systems
  - High maturity organizations use process performance models (PPMs)
  - PPMs used to control critical processes affecting project performance
    - Schedule
    - Quality
    - Etc.
  - PPMs used to predict things like project completion date or quality at product delivery
- ISO 9000
  - A quality standard: not a model
  - More high level
  - Focus is more on addressing the kinds of things a quality plan should contain, rather than the kinds of practices to employ

© Copyright 2018 Commonwealth of Australia

PPMs

- PPMs are probabilistic models
  - Monte Carlo simulations
  - Reliability models
  - Rayleigh distributions
- Goal is established from business needs
  - Improved quality
  - Time to market
  - Timely delivery
- Model constructed from determining the influence of subprocesses on business goal
  - Peer review effectiveness
  - Time to achieve clarity on customer requirements
  - Subprocess output will vary from instantiation to instantiation

© Copyright 2018 Commonwealth of Australia

What's missing?

- PPMs have their limitations
- Factors related to risk typically not included
  - Risk characterized by likely impact and probability of occurrence
  - Characteristics tend to be constant
    - Likely impact and probability of occurrence stay constant each time evaluated unless some external event (like a requirements change) impacts risk assessment
- PPMs focus on overall process performance
  - Component process performance is not constant
  - Since PPMs do not generally utilize evaluations of risk-related factors, PPMs may often miss the mark on schedule prediction

© Copyright 2018 Commonwealth of Australia

So, what do we do?

- Utilize a methodology like SCRAM as an adjunct to PPMs to zero in on realistic dates

© Copyright 2018 Commonwealth of Australia

THE REMEDY

© Copyright 2018 Commonwealth of Australia

What does SCRAM mean?

- Go away!

- Secure Continuous Remote Alcohol Monitoring
  - As modelled here by Lindsay Lohan

SplashNewsOnline.com/Hollywood.tv

- Schedule Compliance Risk Assessment Methodology

© Copyright 2018 Commonwealth of Australia

SCRAM

- SCRAM is a methodology that has evolved from reviews of Capability Acquisition and Sustainment Group (CASG) Projects of Interest and Concern

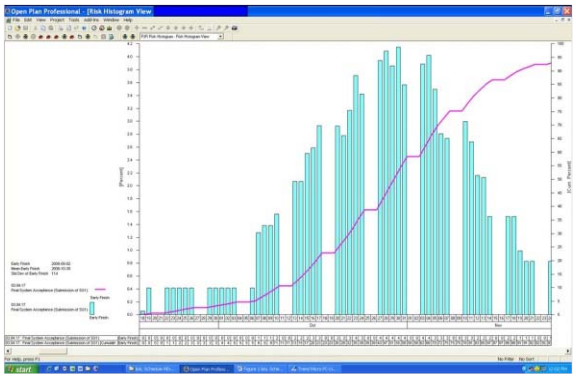
Schedule is almost always the primary concern of our project stakeholders

- SCRAM is a key component of CASG's initiatives to improve schedule performance

© Copyright 2018 Commonwealth of Australia

What is SCRAM?

- SCRAM is used to identify issues and risks to schedule compliance
  - quantifies the schedule impact of issues and risks using scientific analysis techniques
    - Schedule Monte Carlo Simulation
    - Software Parametric Modelling



- Embodies best practices from
  - systems and software engineering
  - schedule development and project execution

- Facilitates improved business practices based on feedback from reviews, (i.e. identification of systemic root causes / issues)

© Copyright 2018 Commonwealth of Australia

Diversity of SCRAM Reviews

- SCRAM has been proven in a number of technology domains with projects of varying size and complexity

- The CASG SCRAM Team has completed reviews on CASG major acquisition projects:
  - Aerospace
  - Enterprise Resource Planning
  - Command and Control
  - Information Communication Technology
  - Communications / Telecommunications

  Maritime

  Training

  Satellite Ground Station

© Copyright 2018 Commonwealth of Australia

SCRAM Testimonial

- SCRAM has been applied successfully to the F-35 Joint Strike Fighter Program in the USA (web search “F-35 Australian SCRAM”)
  - Six SCRAM reviews were conducted from 2011 to 2015 (on-board and ground software)

    “The SCRAM reviews on the F-35 Program were extremely helpful to us. SCRAM gave us new techniques that allowed us to better understand the complexities of our software development. Within two weeks of coming in, the SCRAM reviews were able to point out areas where we were going to have problems. SCRAM also gave us new techniques for measuring the progress of software development and for predicting how long the software development was going to take. In 2014, I briefed the SCRAM results to the Defense Acquisition Board. Of all the organizations that were making estimates, the SCRAM estimates, in hindsight, were the most accurate.”

    -- Testimonial from Lt. Gen. Chris Bogdan, Program Executive Officer, F-35 Program (24 March 2017)

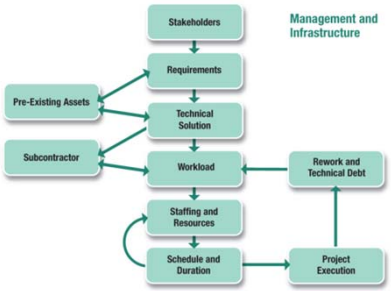
© Copyright 2018 Commonwealth of Australia

HOW IT WORKS

© Copyright 2018 Commonwealth of Australia

Root Cause Analysis of Schedule Slippage (RCASS)

- Has evolved from our experiences in conducting reviews
- Shows logical dependencies and linkages between information categories
- Covers project planning and project execution



- Used by projects as a guide to:
  - Monitor project schedule performance
  - Identify issues and emerging risks to schedule slippage
    - Identify root causes of issues and risks
    - Select appropriate measures to serve as leading indicators for monitoring risk realisation

© Copyright 2018 Commonwealth of Australia

Categories of Information

Requirements

Workload

Rework & Technical Debt

Management & Infrastructure

Stakeholders

Subcontractor

Staffing & Resources

Pre-Existing Assets

Project Execution

Management & Infrastructure

Schedule & Duration

Technical Solution

© Copyright 2018 Commonwealth of Australia

Organizing Project Information

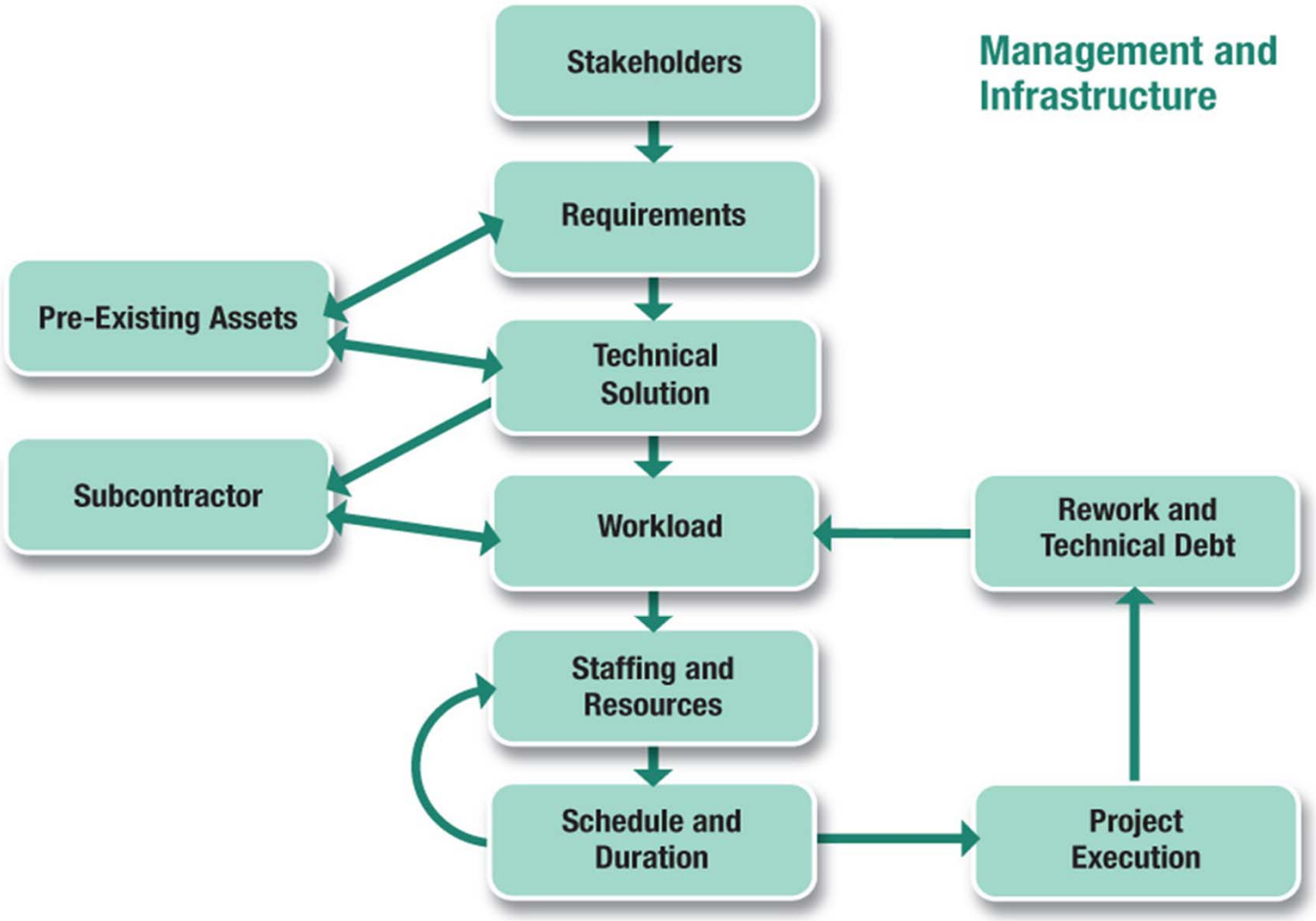
Program Managers are flooded with information, making it difficult to distinguish between symptoms and root causes of schedule slippage

To de-clutter and organise the massive amounts of information, SCRAM Assessors utilise a thought model

Root Cause Analysis of Schedule Slippage (RCASS)

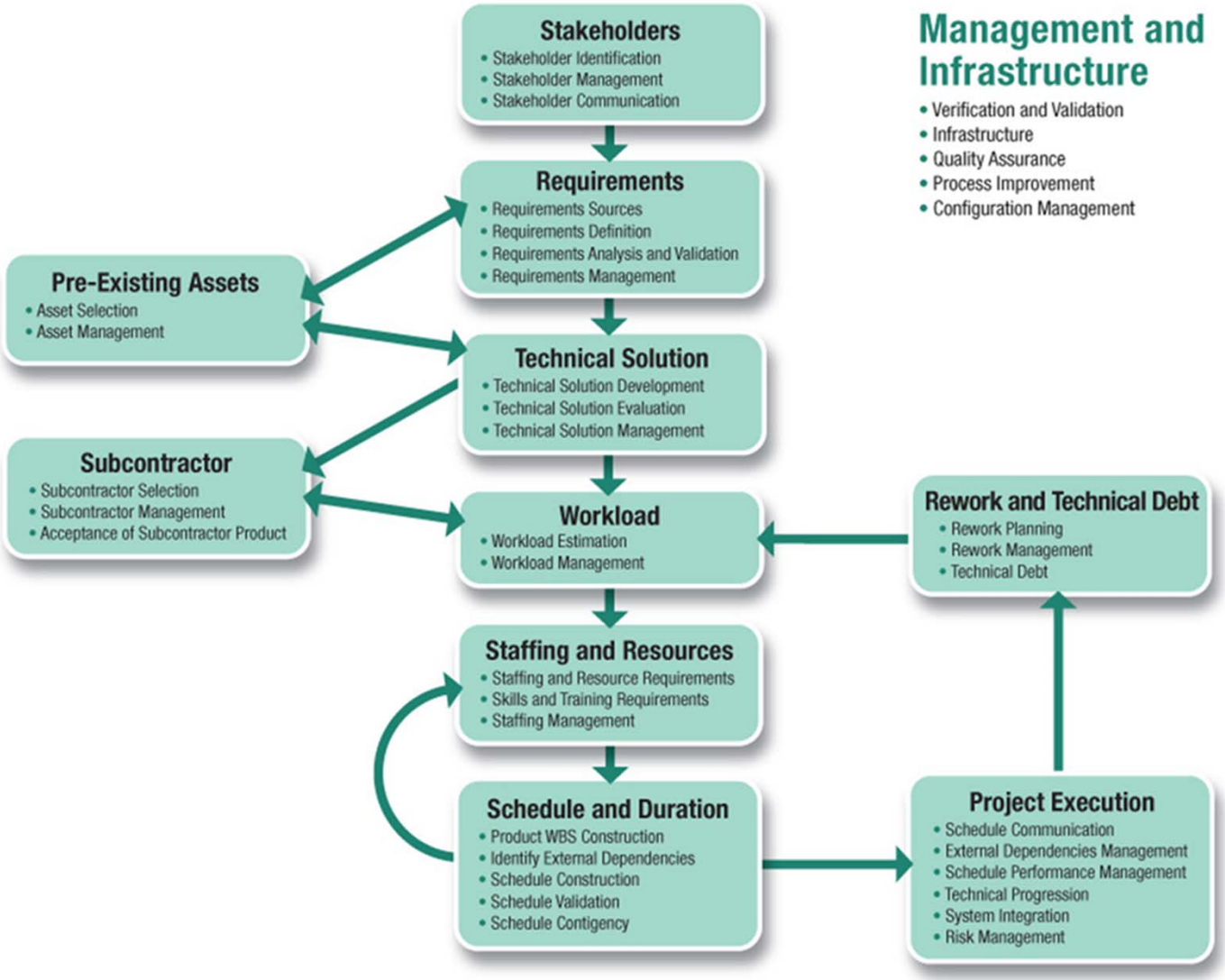
© Copyright 2018 Commonwealth of Australia

Root Cause Analysis of Schedule Slippage (RCASS) Model



© Copyright 2018 Commonwealth of Australia

SCRAM PR/AM Processes



© Copyright 2018 Commonwealth of Australia

Example SCRAM Sub-category

REQ-SRC - Requirements Sources

Purpose

The purpose of Requirement Sources is to ensure that all relevant sources of requirements are considered in eliciting and gathering requirements.

Outcomes

As a result of successful implementation of this process:

REQ-SRC-1 Capability Definition including Operational Concepts and Environments are established

REQ-SRC-2 Customer/Stakeholder Product Needs, Expectations and Constraints are elicited

REQ-SRC-3 Customer/Stakeholder Requirements are established based on Customer/Stakeholder Product Needs, Expectations and Constraints

REQ-SRC-4 Relevant Legislative and Regulatory Requirements are identified

REQ-SRC-5 System Assurance Requirements are identified

REQ-SRC-6 System Integration and Test Requirements are identified

REQ-SRC-7 Support System Requirements are identified

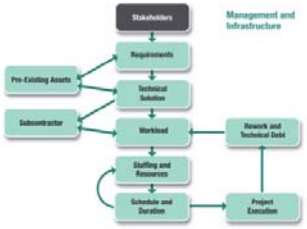
© Copyright 2018 Commonwealth of Australia

SR/MAG Risk Guidance

- Practices and Indicative Work Products refers to an outcome within the sub-category
- Practices
  - provide guidance and information on how to achieve outcomes
  - used by projects to contribute towards constructing a schedule that minimises the risk of schedule slippage
- Indicative Work Products
  - a possible artefact resulting from performing the practice
  - produced by projects to record outputs of work performed
- Practices and indicative work products are recommendations and are NOT mandatory

© Copyright 2018 Commonwealth of Australia

Stakeholders



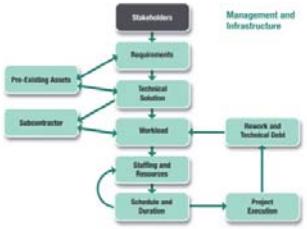
Reflects project turbulence because of difficulties in synchronising the project's stakeholders: users, customers, system engineers, developers, maintainers, others.

"Our stakeholders are like a 100-headed hydra – everyone can say 'no' and no one can say 'yes'."

SCRAM Examples

- Late in development, a critical stakeholder (customer) added a condition for acceptance that removed three months from the development schedule

© Copyright 2018 Commonwealth of Australia

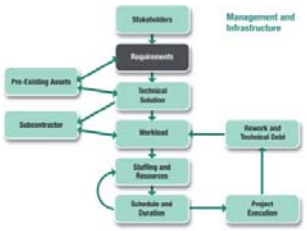


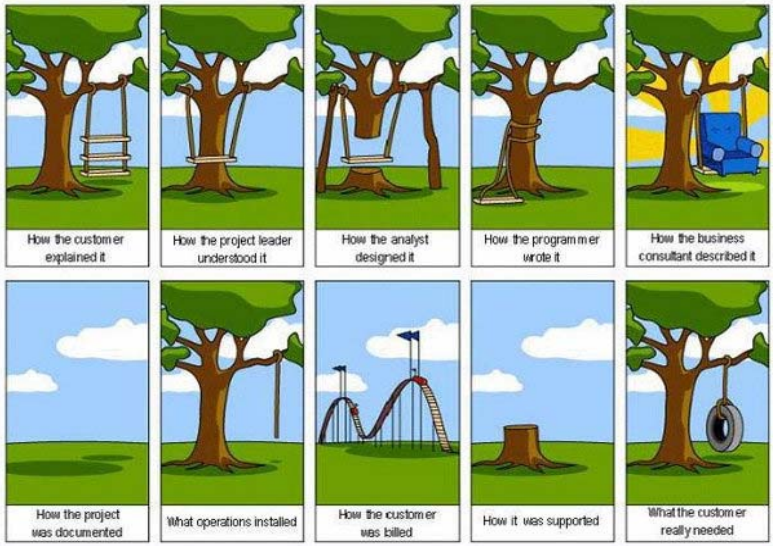
Stakeholders

- Sub-categories
  - Stakeholder Identification
  - Stakeholder Management
  - Stakeholder Communication

© Copyright 2018 Commonwealth of Australia

Requirements





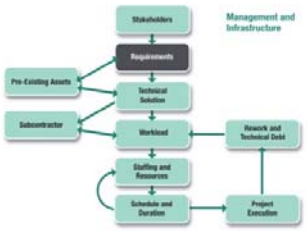
Reflects understanding and stability of the functional requirements, performance requirements, system constraints, standards, etc. used to define and bound what is to be developed

SCRAM Examples

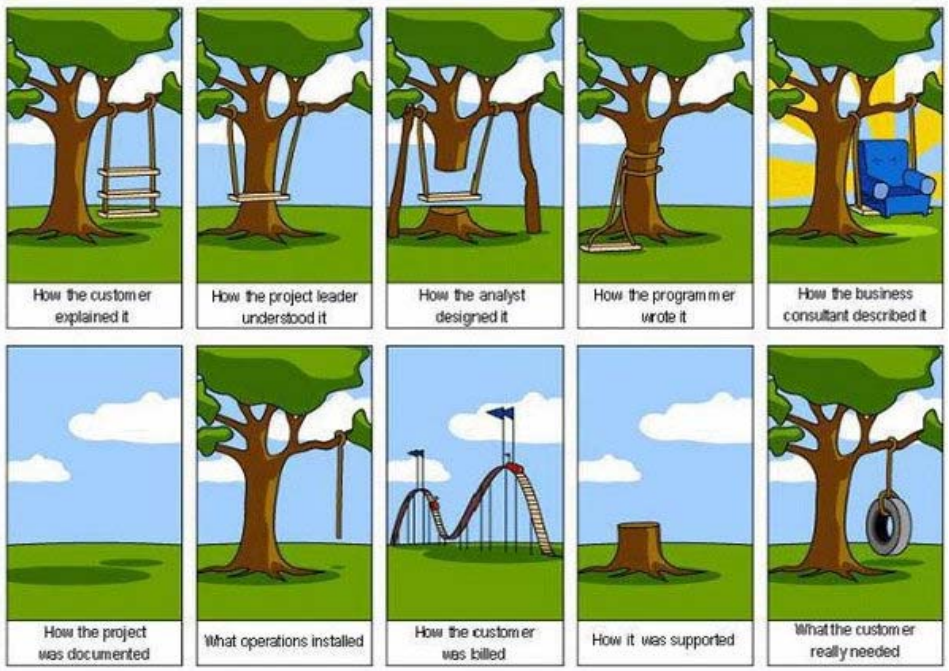
- A large ERP project had two system specifications – one with the sponsor/customer and a different specification under contract with the developer – would this be a problem?

© Copyright 2018 Commonwealth of Australia

Requirements

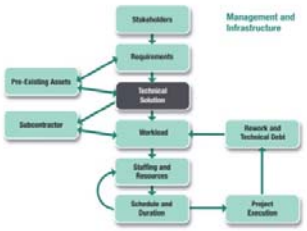


- Sub-categories
  - Requirements Sources
  - Requirements Definition
  - Requirements Analysis & Validation
  - Requirements Management



© Copyright 2018 Commonwealth of Australia

Technical Solution



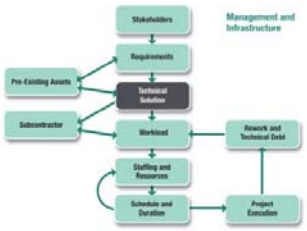
Reflects the design considerations and approaches needed to ensure that the chosen solution is architected, logically and physically designed to align with the business enterprise architecture, satisfy functional and non-functional requirements (quality attributes) and optimised to meet system development and sustainment life- cycle objectives.

SCRAM Examples

- The tendered Technical Solution was not compliant with functional performance requirements for the system to be deployable by air and sea transportation
  - Mutually exclusive transport requirements

© Copyright 2018 Commonwealth of Australia

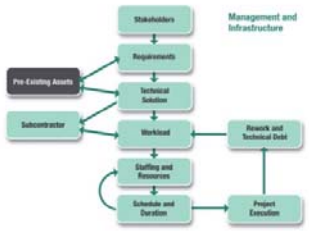
Technical Solution



- Sub-categories
  - Technical Solution Development
  - Technical Solution Evaluation
  - Technical Solution Management

© Copyright 2018 Commonwealth of Australia

Pre-existing Assets



Reflects products developed independent of the project that will be used in the final product, i.e. an asset that reduces the amount of new work that has to be done on a project.

"It doesn't do what we thought…"

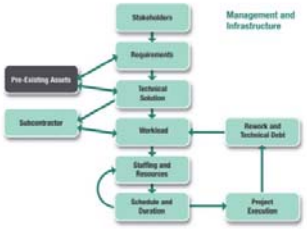
"There is a lot of functionality we don't need."

SCRAM Examples

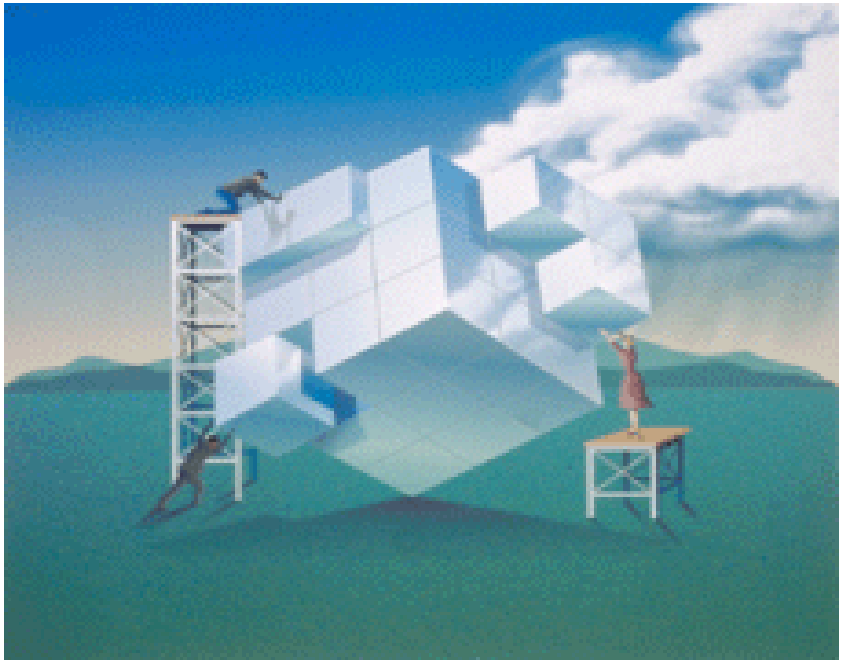
- COTS products that didn't work as advertised, resulting in additional work
- COTS product required a "technology refresh" during development as the project was years late (cost the project $8M)

© Copyright 2018 Commonwealth of Australia

Pre-existing Assets

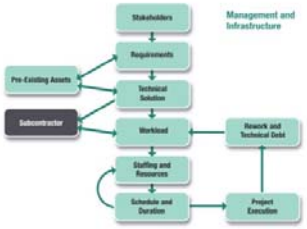


- Sub-categories
  - Asset Selection
  - Asset Management



© Copyright 2018 Commonwealth of Australia

Subcontractor



Reflects subcontractor products or services that will be delivered as a part of the overall system.

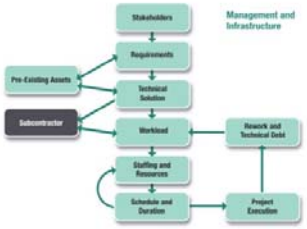
If the subcontractor doesn't perform, additional work required by the Prime

SCRAM Examples

- Prime and sub-contractor schedule not linked or aligned

© Copyright 2018 Commonwealth of Australia

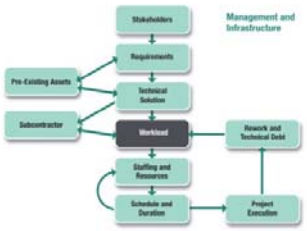
Subcontractor



- Sub-categories
  - Subcontractor Selection
  - Subcontractor Management
  - Acceptance of Subcontractor Product

© Copyright 2018 Commonwealth of Australia

Workload



Reflects the quantity of work to be done and provides a basis for estimating effort/staffing and duration

"Unrealistic expectations based on inaccurate estimates are the single largest cause of software failure."

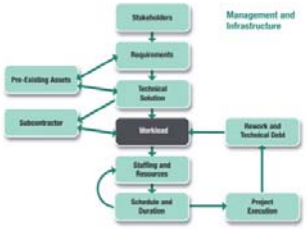
» Futrell, Schafer

SCRAM Examples

- Source lines of code is typically underestimated
- Contract data deliverables workload often underestimated by both contractor and customer
- System Integration labs underestimated resulting in additional cost to build another lab (inadequate basis of estimates)

© Copyright 2018 Commonwealth of Australia

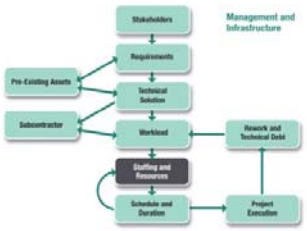
Workload



- Sub-categories
  - Workload Estimation
  - Workload Management

© Copyright 2018 Commonwealth of Australia

Staffing & Resources



Reflects the availability, capability and experience of the staff necessary to do the work as well as the availability and capacity of other resources, such as test and integration labs.

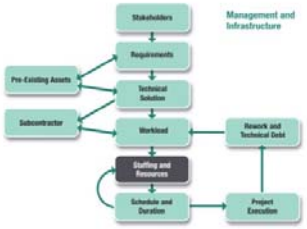
Bringing on people to solve a slippage problem may make it worse (especially late in the project)

SCRAM Examples

- Parent company sacking workers as soon as job is completed
  - Workers went on a “Go Slow”
- Scheduling staff for 12 hours a day (to recover schedule)

© Copyright 2018 Commonwealth of Australia

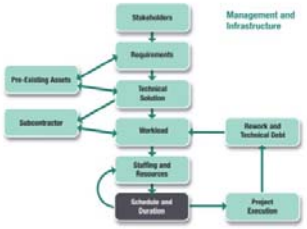
Staffing & Resources

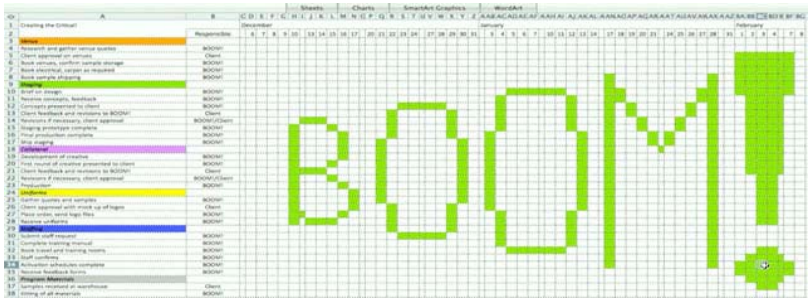


- Sub-categories
  - Staffing and Resource Requirements
  - Skills and Training Requirements
  - Staffing Management

© Copyright 2018 Commonwealth of Australia

Schedule & Duration





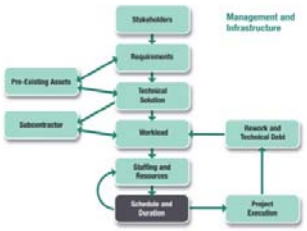
Reflects the task sequencing and calendar time needed to execute the workload by available staff and other resources (e.g. test labs).

SCRAM Examples

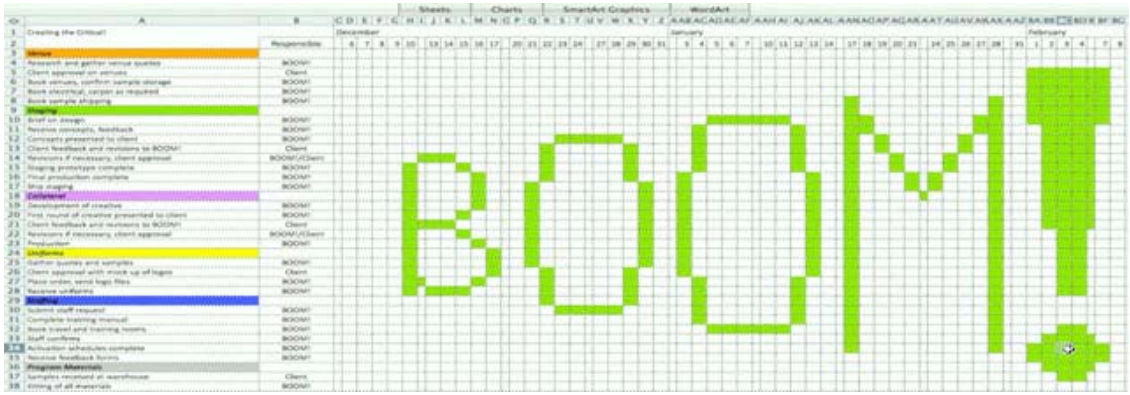
- No effective integrated master schedule to provide an overall understanding of the completion date of the project
  - 13 subordinate schedules
  - Failed Health Checks
  - Critical Path went subterranean!

© Copyright 2018 Commonwealth of Australia

Schedule & Duration

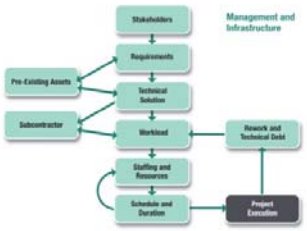


- Sub-categories
  - Work Breakdown Structure Construction
  - Schedule Construction
  - Identify External Dependencies
  - Schedule Validation
  - Schedule Contingency



© Copyright 2018 Commonwealth of Australia

Project Execution



Focuses on monitoring and controlling the execution of the project in accordance with the project schedule

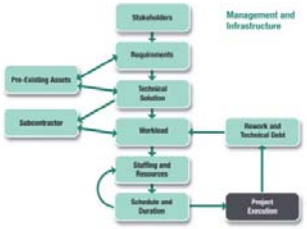
Experience from multiple SCRAM reviews has highlighted the need to focus on System Integration and Technical Progression

SCRAM Examples

- The schedule was not available to program staff or stakeholders
  - Undergoing a schedule tool transition for approx. 2 years
- No System Integration Plan
- No “red” risks on a program undergoing a major contract overrun breach

© Copyright 2018 Commonwealth of Australia

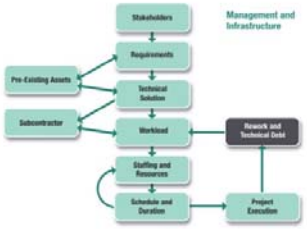
Project Execution



- Sub-categories
  - Schedule Communication
  - External Dependencies Management
  - Schedule Performance Management
  - Technical Progression
  - System Integration
  - Risk Management

© Copyright 2018 Commonwealth of Australia

Rework and Technical Debt



Reflects additional work caused by the discovery of defects in the product and/or associated artefacts, work that is deferred for short-term expediency (Technical Debt) and their resolution

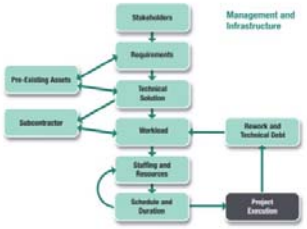
Technical Debt includes suspension of peer reviews, short-cuts in unit test, postponing functionality until later.

Rework is often underestimated or not planned for.

SCRAM Examples

- Suspension of peer reviews led to a bow wave of defects found in System Test
- Accrual of Technical Debt with no repayment plan

© Copyright 2018 Commonwealth of Australia

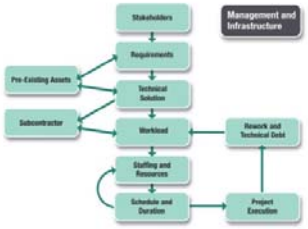


Rework and Technical Debt

- Sub-categories
  - Rework Planning
  - Rework Management
  - Technical Debt

© Copyright 2018 Commonwealth of Australia

Management & Infrastructure



Addresses the factors that impact the efficiency and effectiveness of getting work done, e.g. work processes, use of management and technical software tools, management practices, etc.

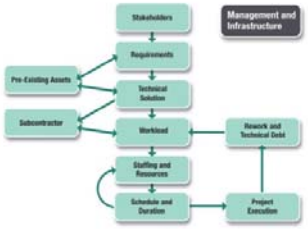
Includes processes for Verification and Validation, Infrastructure, Quality Assurance, Process Improvement and Configuration Management

SCRAM Examples

- Change management process and tools could not keep pace with requirements for System Integration and Test
- Lack of fidelity and qualification of integration and test lab
- Lack of software quality assurance

© Copyright 2018 Commonwealth of Australia

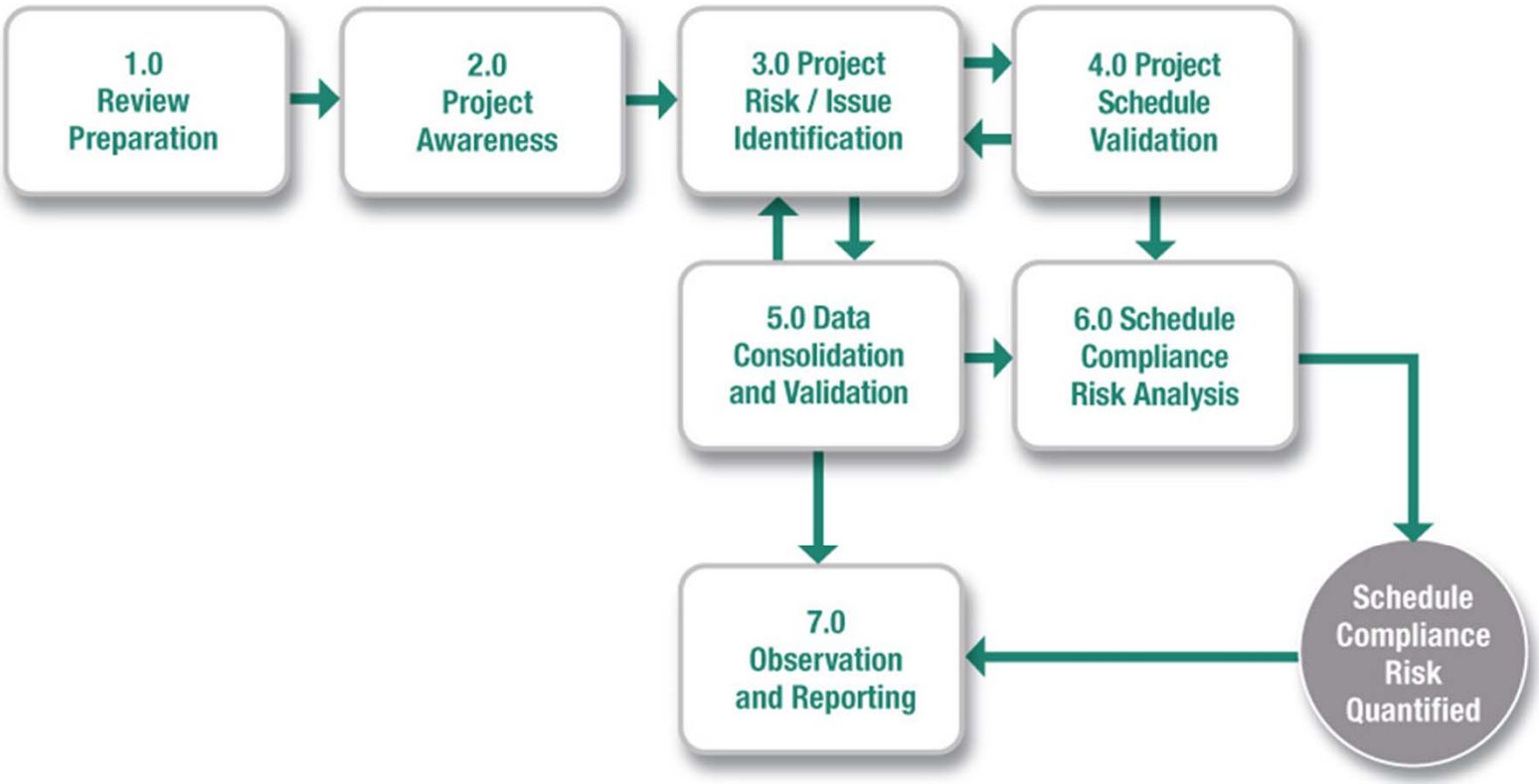
Management & Infrastructure



- Sub-categories
  - Verification and Validation
  - Infrastructure
  - Quality Assurance
  - Process Improvement
  - Configuration Management

© Copyright 2018 Commonwealth of Australia

SCRAM Review Process



© Copyright 2018 Commonwealth of Australia

SCRAM Review Key Principles

- Minimal Disruption
  - Artefact Review (plans, procedures, model evidence) conducted offline
  - Information is collected one person at a time
  - Interviews typically last an hour
- Independent
  - Review team members are organisationally independent of the program under review
    - Some SCRAM reviews have been joint contractor/customer team – facilitates joint commitment to resolve review outcomes
- Non-advocate
  - All significant issues and concerns are considered and reported regardless of origin or source (Customer and/or Contractor).

© Copyright 2018 Commonwealth of Australia

SCRAM Review Key Principles

- Non-attribution
  - Information obtained is not attributed to any individual
  - Focus is on identifying and mitigating the issues/risk
- Corroboration of Evidence
  - Significant Findings and Observations based on at least two independent sources of corroboration
- Rapid turn-around
  - One to two weeks spent on-site
  - Executive out-briefing presented at end of second week
  - Written report two weeks later

© Copyright 2018 Commonwealth of Australia

SCRAM Review Key Principles

- Sharing Results, Openness and Transparency
  - For the parametric modelling component of a SCRAM Review, the organisation under review may witness data analysis and challenge results
  - Preliminary out brief of findings is delivered prior to departure from review site
  - Builds cooperation and trust
  - Builds confidence in the schedule forecast
  - SCRAM Team is the final arbiter

© Copyright 2018 Commonwealth of Australia

The SCRAM Review Team

- Review conducted by a small team including:
  - Engineers (to validate engineering related BoEs, work load estimates, identify project issues and risks, and provide inputs for schedule risk assessment)
    - Supplemented by domain specific subject matter experts as necessary
    - For software intensive development projects, at least one team member should be proficient in software parametric modelling
  - Scheduler experienced in Project schedule tool
    - Validates schedule – conducts schedule health checks
    - Performs Monte Carlo risk modelling with inputs from engineering team members

© Copyright 2018 Commonwealth of Australia

SCRAM Assessor Qualification Framework

- Three levels of SCRAM Assessors
  - Provisional SCRAM Assessor
    - Completed SCRAM training and passed exam
  - Certified SCRAM Assessor
    - Participated in SCRAM Reviews
  - SCRAM Lead Assessor
    - Lead SCRAM Reviews

- SCRAM Principal
  - Lead SCRAM Reviews
  - SCRAM Instructors
  - SCRAM Model Developers

© Copyright 2018 Commonwealth of Australia

OUTPUTS

© Copyright 2018 Commonwealth of Australia

Software-Intensive Projects

- SCRAM answers two basic questions
  - What is the root cause of schedule slippage?
  - When will the project deliver?

- For software-intensive projects, SCRAM uses a commercial model to predict when the software will be delivered
  - Software is often the “long pole in the tent”

© Copyright 2018 Commonwealth of Australia

Estimation versus Forecasting

- Estimation (cost and schedule) occurs before a project begins and is based on characteristics of the software to be developed
- Forecasting (cost and schedule) occurs during development and is based on objective measures of performance to date
- There are a number of software estimation models
  - COCOMO II, SLIM-Estimate, SEER-SEM, PRICE S
  - Estimates are derived from a number of parameters (attributes) that characterise the software including
    - Size
    - Application complexity
    - Experience/ and capability of developers
- Some estimation models allow you to enter constraints (e.g., no more than 50 people, must be completed with 18 months)
- Output is an estimate of schedule and effort

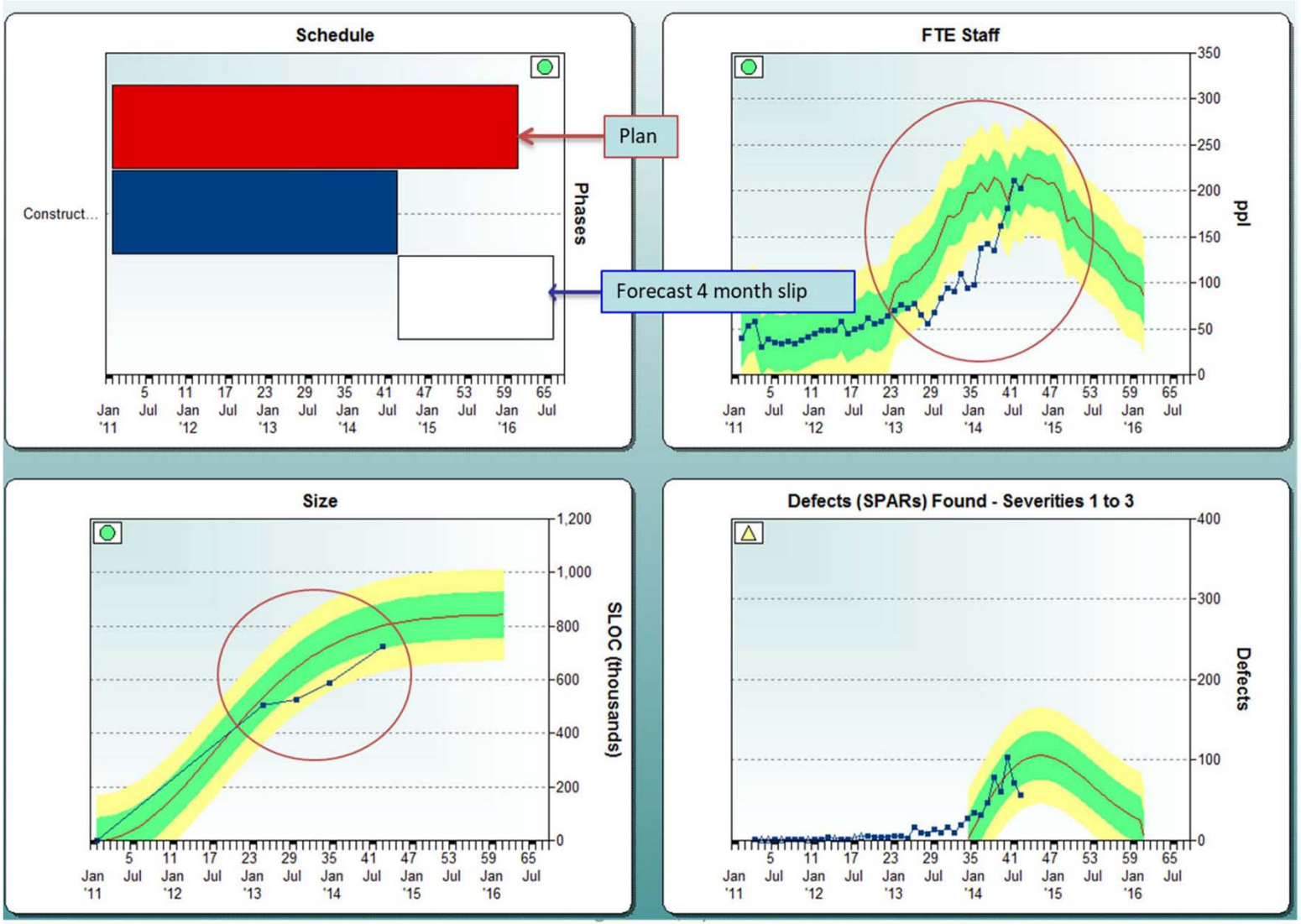
© Copyright 2018 Commonwealth of Australia

Forecasting Software Duration

- Data comes from the contractor(s) – must be validated by the contractor(s)
  - Never take data and just “run with it”
- The majority of our time “modelling” is spent summarizing and validating the data
  - That’s a critical step often missed by estimators
  - Running the model is the easy part
- We ALWAYS show the results to the contractor(s) first
  - We don’t surprise them by having the results “pop up” at a briefing to someone else
  - This goes a long way to building trust

© Copyright 2018 Commonwealth of Australia

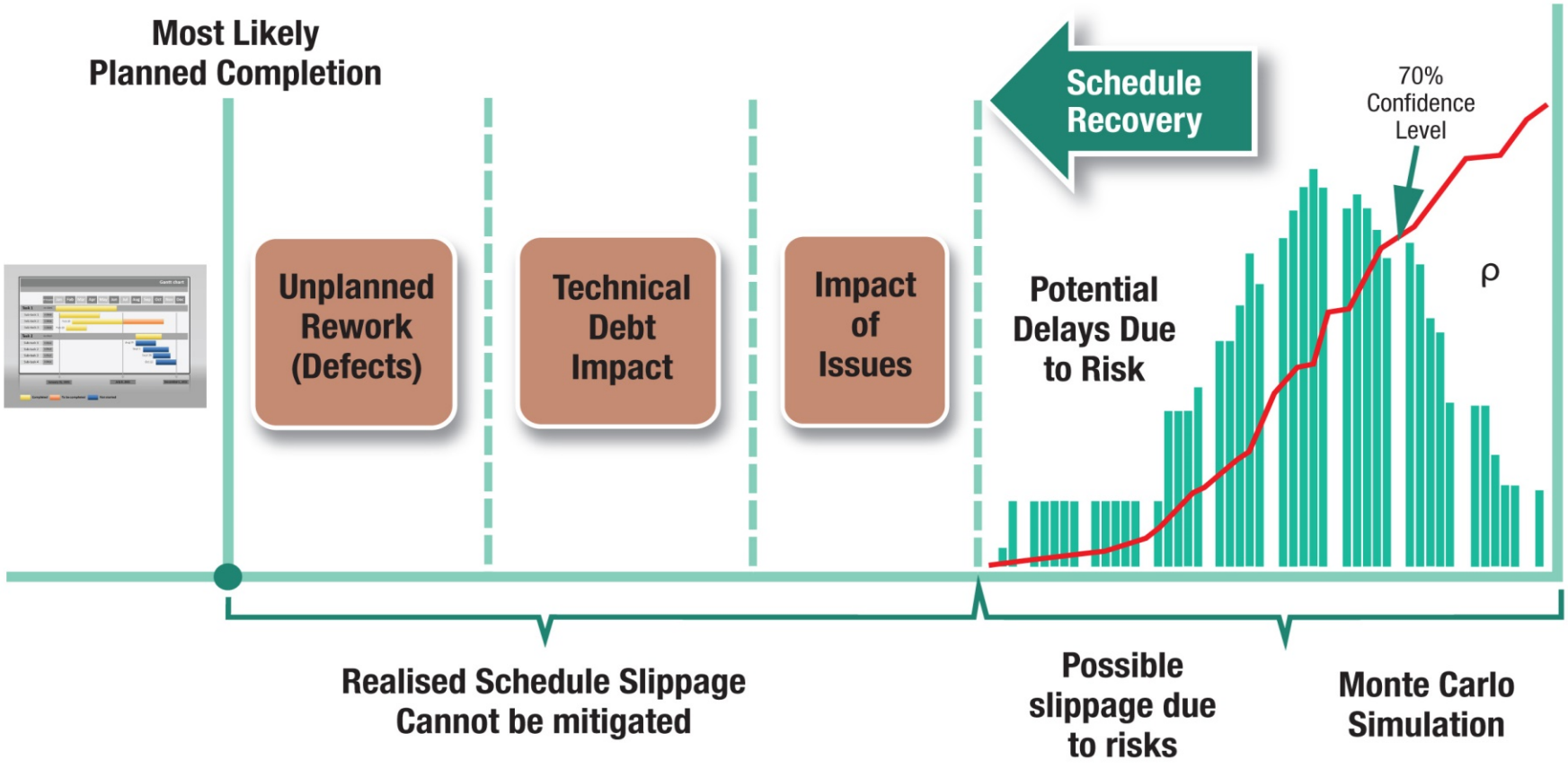
Example Forecast



© Copyright 2018 Commonwealth of Australia

SCRAM Identifies and Quantifies Schedule Slippage Root Causes and Risk

Causes of Project Slippage and Potential Risk Delays



© Copyright 2018 Commonwealth of Australia

CONCLUSION

© Copyright 2018 Commonwealth of Australia

….In conclusion

- Many projects still plagues by late deliveries
  - Even organizations that have implemented aggressive process improvement have underperformed on cost and schedule
  - CMMI high maturity organizations also miss the mark
    - Utilize PPMs, but PPMs have limitations
    - Risk-related factors not always considered
- CASG has developed SCRAM as an independent schedule review process
  - Model includes 11 risk-related factors
  - Uses Monte Carlo simulation to develop numerical probability of meeting project critical milestones
  - Used successfully on 40 projects, including F-35
- SCRAM is a tool that effectively complement the use of PPMs and other models to estimate realistic schedules

© Copyright 2018 Commonwealth of Australia

Contact Information

- Emanuel R. Baker, Ph.D.
  - Software Engineering Consultants, Inc.
  - (310) 278-7241
  - erbaker@swengcon.com
- Betsy Clark, Ph.D.
  - Software Metrics, Inc.
  - (703) 754-0115
  - betsy@software-metrics.com
- Adrian Pitman
  - Capability Acquisition and Sustainment Group
  - +61 (0404) 815665
  - adrian.pitman@gmail.com
- Angela Tuffley
  - RedBay Consulting Pty Ltd.
  - +61 (7) 3206 7774
  - angela.tuffley@gmail.com

© Copyright 2018 Commonwealth of Australia