34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

Role-Based Structuring of Systems Engineering Teams

Outline

- Introduction
- Organization of SE Teams
  - Single-Role
  - Multi-Role
  - Hybrid Single- and Multi-Role
  - Embedded Multi-Role
  - Embedded Multi-Role with Lead Ses
  - Underlying Concepts
- Case Studies / Examples
  - Single-Role to Hybrid
  - Embedded Multi-Role on an Agile Project
- Conclusions

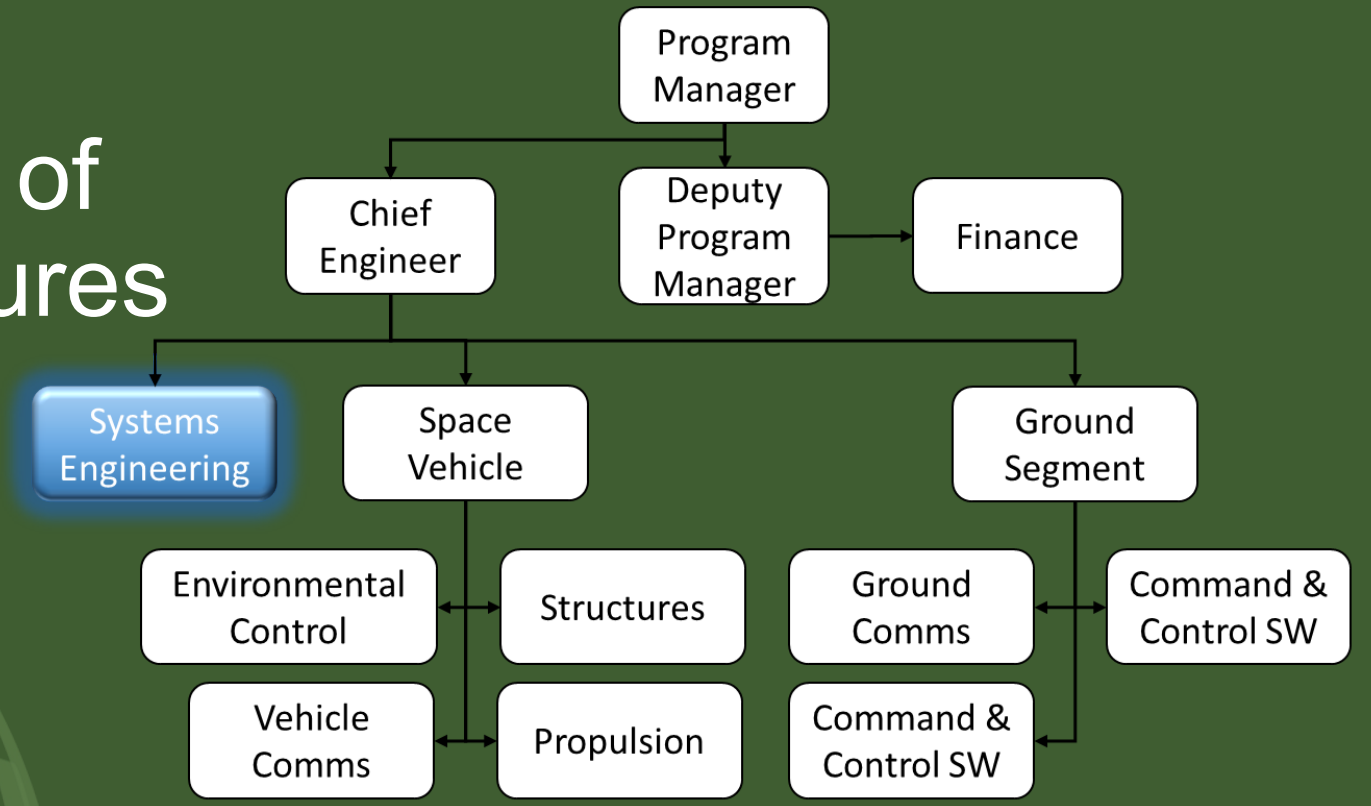
Introduction

- Technical Processes well-defined and understood
- Successful execution of the processes partially dependent on team structure

- **But which structure??**

Introduction

- Abundant definitions of overall project structures
- SEBoK describes functional, matrix, Integrated Product Teams, etc.
- All define SE teams as “closed box” within the overall structure



What is Missing?

- How does the internal structure of the SE team integrate with the execution of technical processes?

- SE processes themselves are highly tailorable
  - Expand this concept to optimize the structure of SE team itself

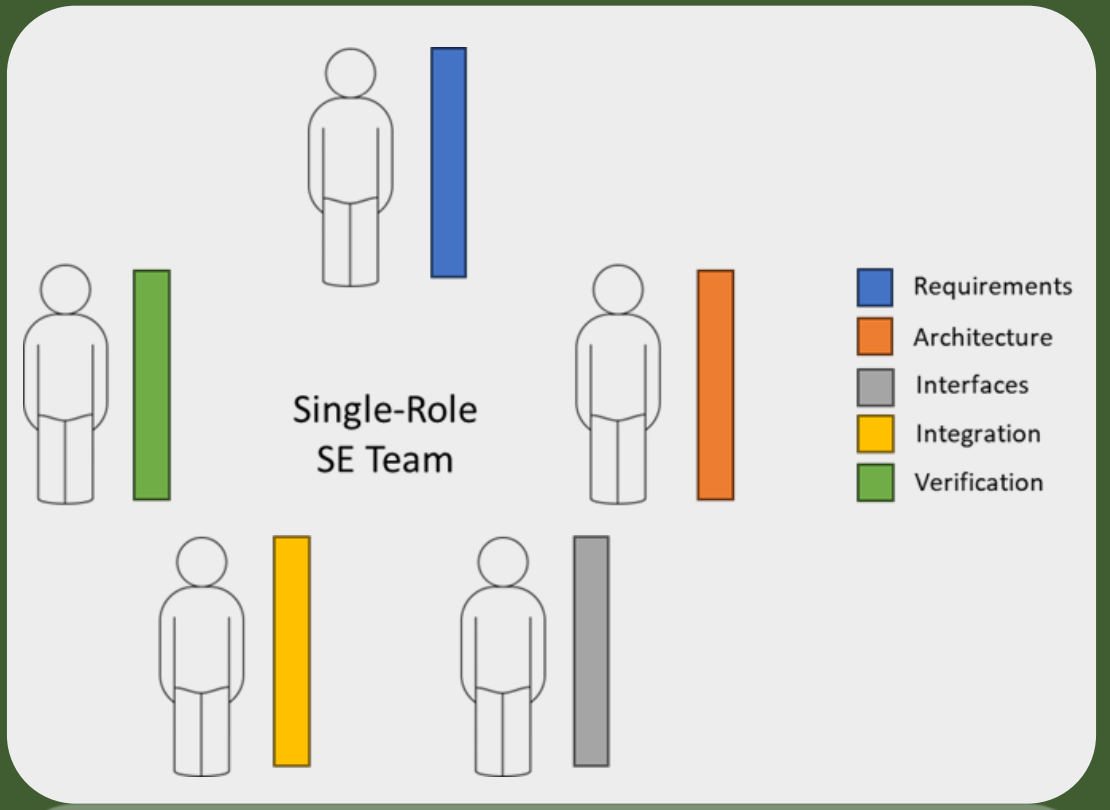
Methods of Structuring SE Teams

Structural Foundations

- Numerous ways to structure SE teams
- Presentation focuses on teams responsible for major technical processes within a large complex project
  - Requirements, Architecture, Interface, Integration, and Verification
- Concepts presented can be expanded to other processes, teams, roles, and project sizes

Single-Role Teams

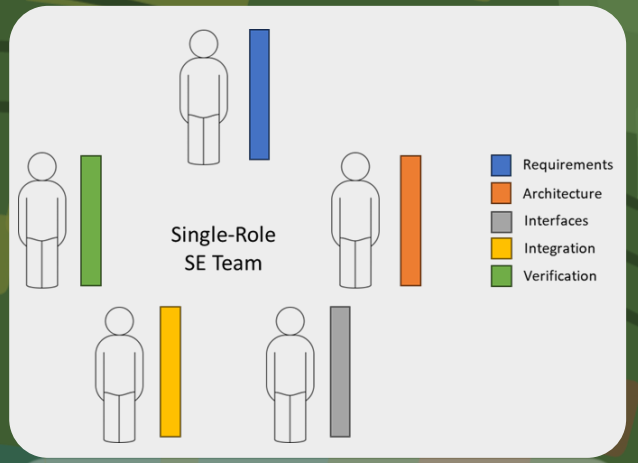
- Simplest structure
- Each member is a specialist with deep knowledge in a single SE discipline
- Best suited for teams with less experienced SEs



Single-Role Teams

Advantages

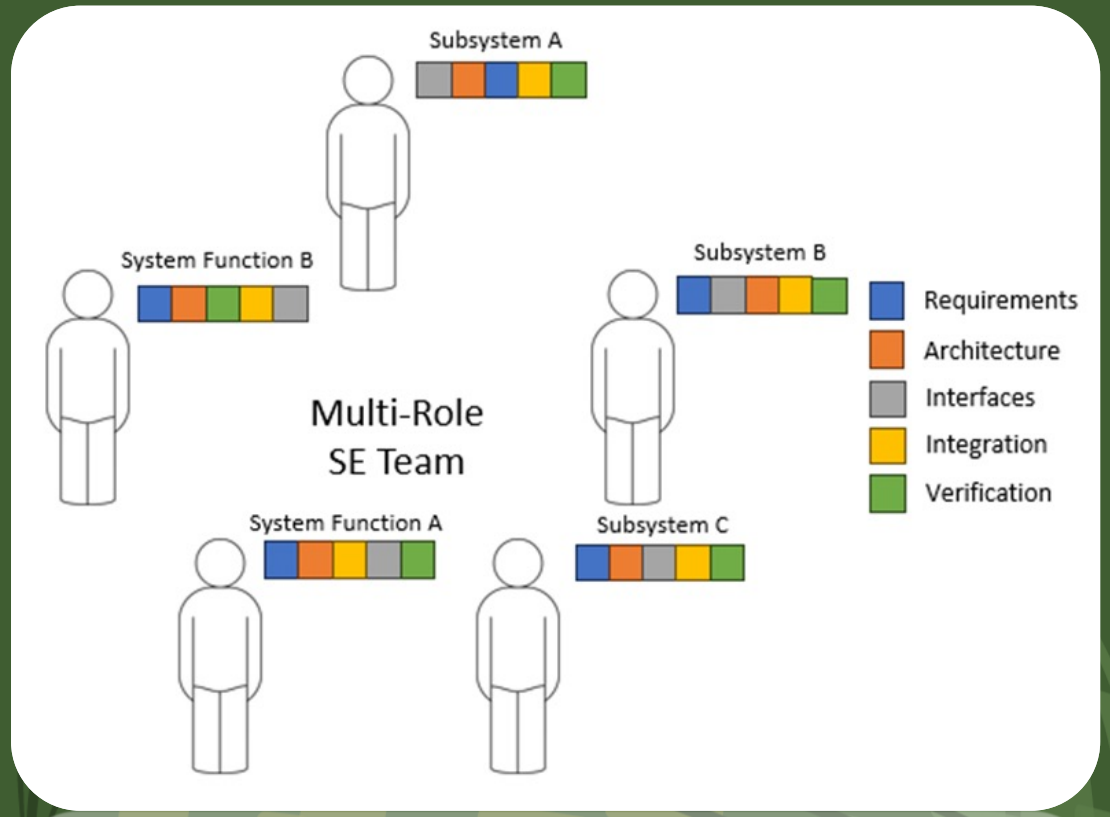
- Easiest to implement
- Team members focus on a single SE discipline
- Other teams can interface with experts



Disadvantages

- Federated structure and mindset => Holistic integration more difficult
- Multiple SEs to interface with
- Risk of miscommunication increases
- Little room for growth and skill expansion
- No opportunity for understanding the full system

Multi-Role Teams



- Simple structure
- Each member has broad knowledge of multiple SE disciplines
- Applied knowledge to entire system
- Best suited for teams with more experienced SEs

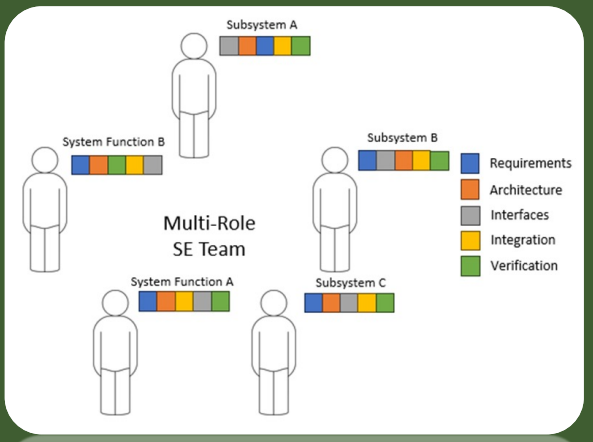
Multi-Role Teams

Advantages

- Individuals focus on an entire system/subsystem
- Allows members to expand skill sets
- Reduces federated SE teams and provides single person for subsystems
- Communication and integrated assistance more effective

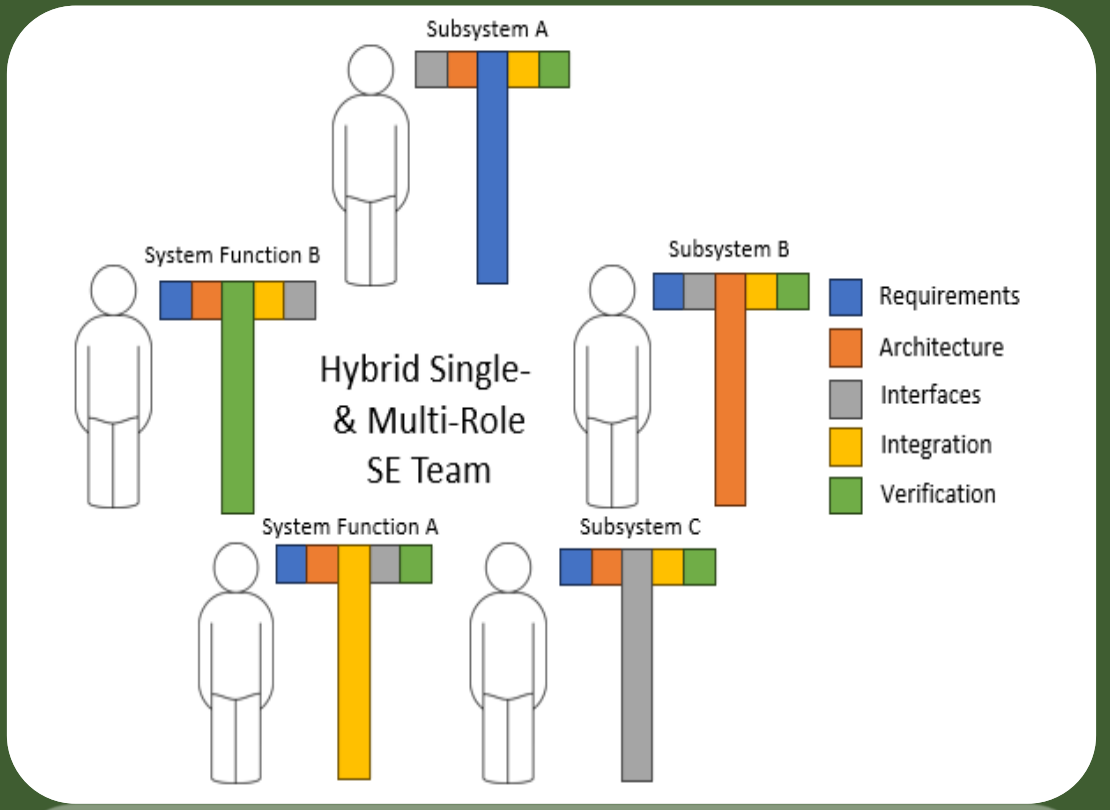
Disadvantages

- No experts in SE disciplines => increase risk of poor definition
- No opportunity for understanding the full system
- Federated structure still exists with the separate SE team



Hybrid Single- and Multi-Role Teams

- Combination of previous structures
- Individual responsible for entire system and for an SE discipline
- T-Shaped Engineers
- Best for teams with experienced/expert SEs and with few less experienced SEs to grow



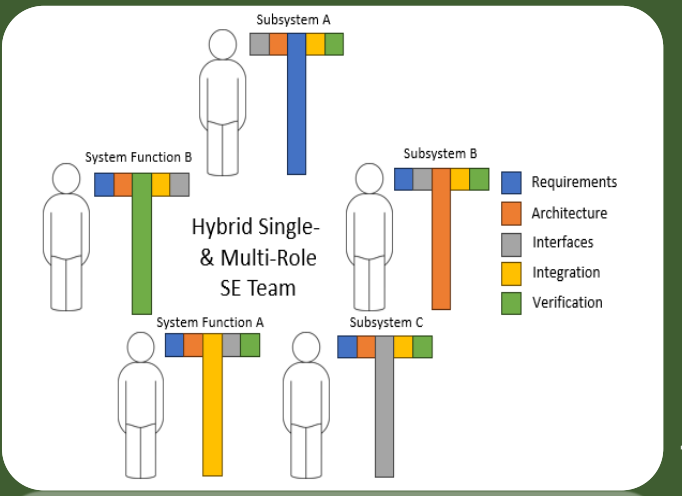
Hybrid Single- and Multi-Role Teams

Advantages

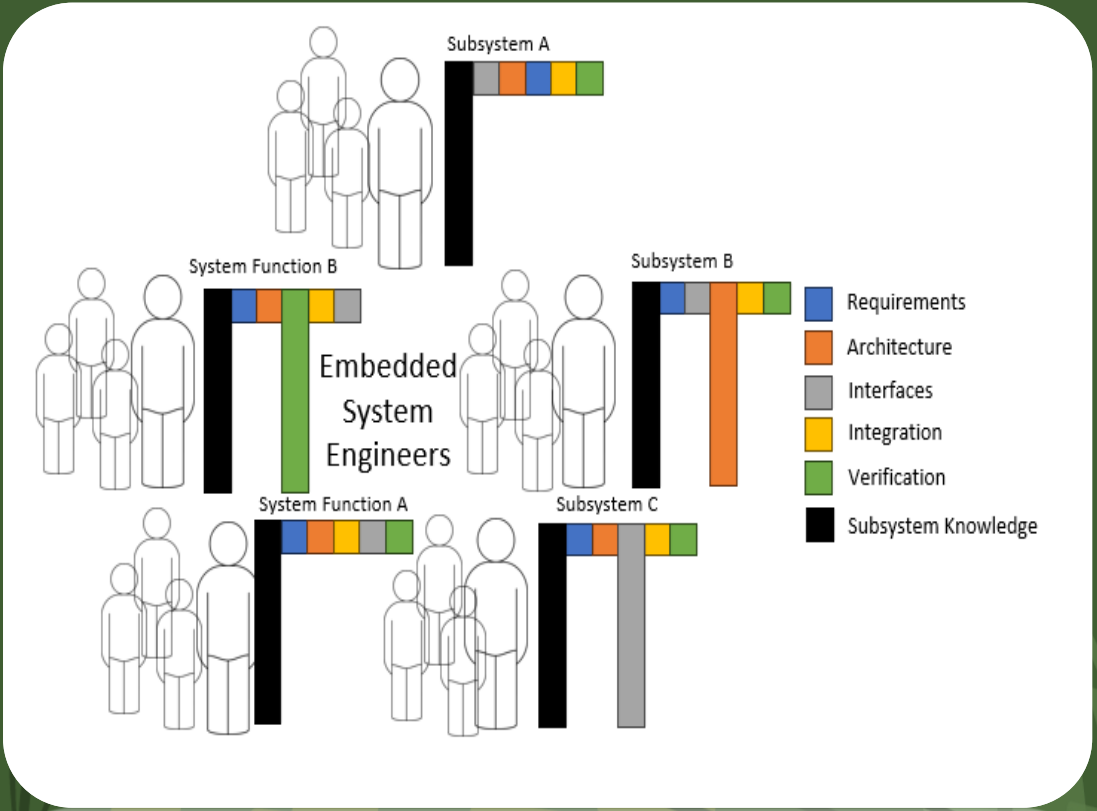
- Single interface between SE and other teams
- Design authority interactions reduced
- SE specialists ensure general design knowledge exists alongside specific system knowledge
- Increase in growth potential for team members based on team wants/needs

Disadvantages

- Little opportunity for anyone to understand the full system
- Entire SE team needed for holistic view
- Federated structure still exists with the separate SE team

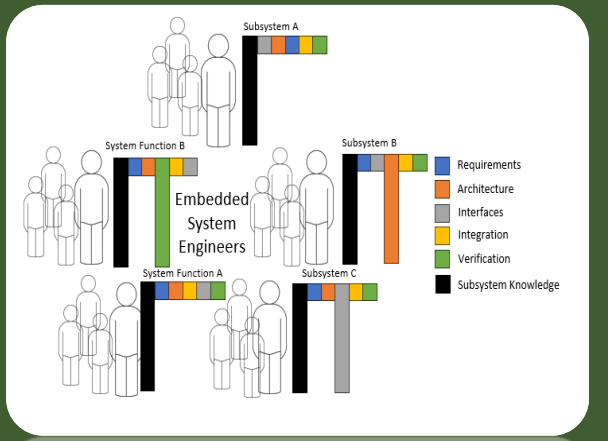


Embedded Multi-Role Teams



- No separate SE team, SEs are part of the subsystem team
- Well-versed in multiple SE disciplines and may have deep knowledge in one or more
- Best for teams with those who understand SE, knowledgeable on the specific system, and maintain Systems Thinking mindset

Embedded Multi-Role Teams



Advantages

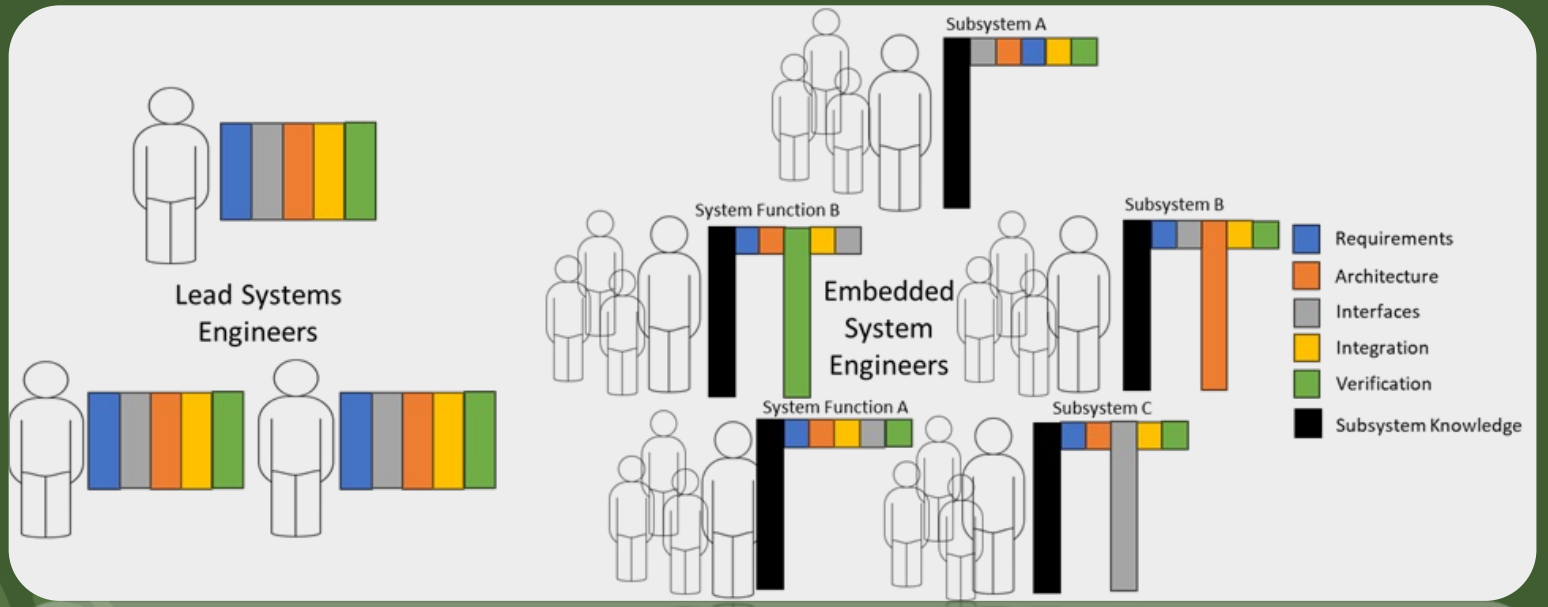
- Design definition, implementation, and verification more efficient
- Changes occur faster
- Structure eliminates the federated team perspective
- Real-time top-down and bottoms-up communication

Disadvantages

- Risk of SEs focusing on area of responsibility and losing sight of the bigger picture
- Requires team of experienced/expert SEs
- Full system-level knowledge still missing
- Reduced growth potential

Embedded Multi-Role with Lead SEs Teams

- Addition of expert SEs to the embedded structure
- Most complicated to implement
- Lead SE bridges high-level system definition and lower-level design



- Requires experienced engineers for embedded role and expert SEs for lead role

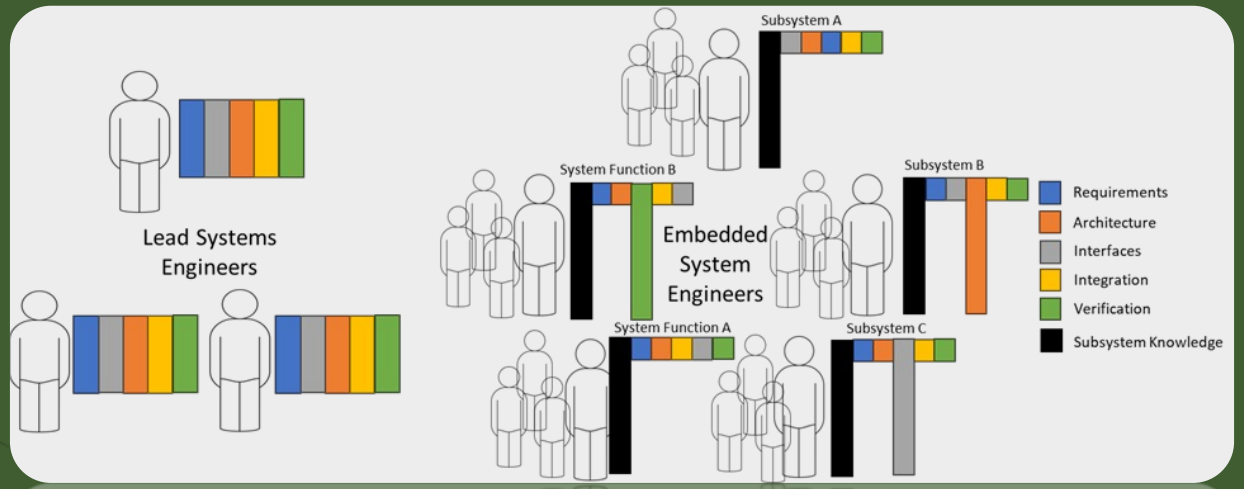
Embedded Multi-Role with Lead SEs Teams

Advantages

- All advantages of Embedded structure
- Full system-level knowledge achieved
- More centralized approach to definition and decisions
- Opportunity for others to grow to lead roles

Disadvantages

- Team size may be large
- Lead SEs may fall to “information overload”



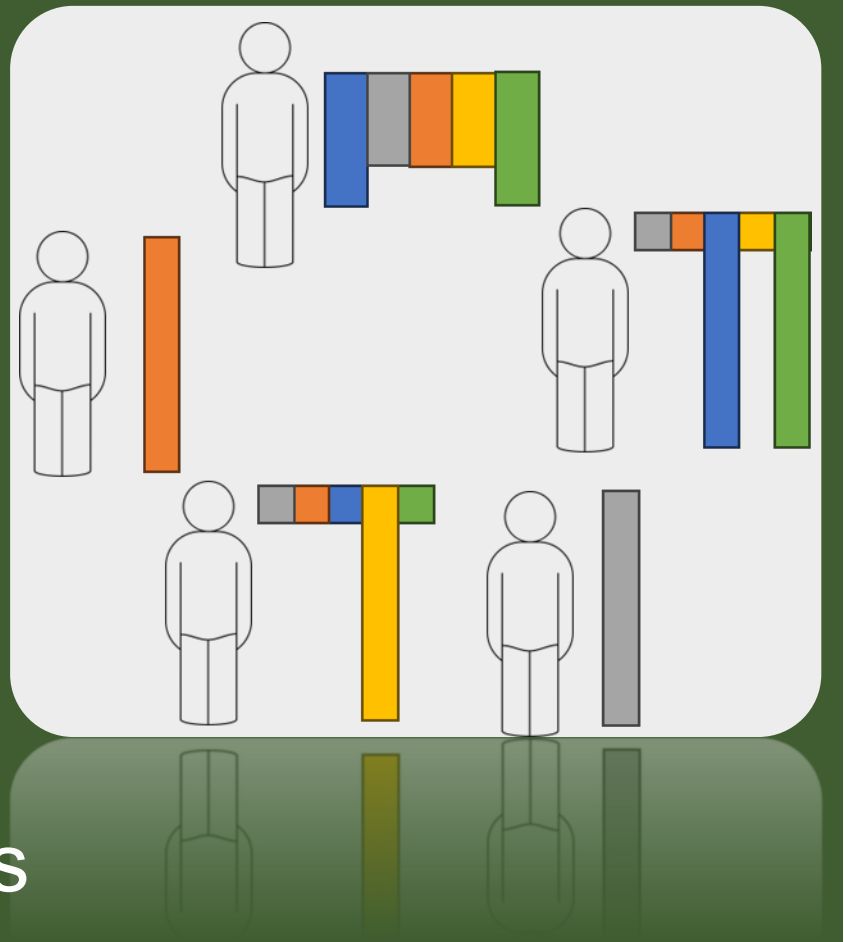
Underlying Concepts

1. All structures suffer from single point of failure
   - Work on a specific area halts if the responsible person is not available
   - Can lead to project delays
   - Cross-team training eliminates single points of failure and promotes a growth mindset
2. Lead SE can be added to any structure to increase full system-level understanding
3. Structures are *TAILORABLE* and scalable!
   - Combine / remove aspects of structures
   - One person can fill multiple roles
4. These were only a sample of possible SE team structures, more exist!

Case Studies / Examples

Single-Role to Hybrid Multi-Role

- Led an SE team comprised of mixed experience
  - Lead SE and requirements/verification expert (10+ yrs)
  - Experienced systems architect (10+ yrs)
  - Experienced interface manager (10+ yrs)
  - Mid-level, broad knowledge engineer (5+ yrs)
  - Entry-level SE (~1 yr)
- Traditional waterfall lifecycle
- At risk of missing major milestone events



Single-Role to Hybrid Multi-Role

- One or two experts for each SE discipline
- Other systems and team members had single person for specific SE disciplines
- Less experienced engineers paired with more experienced to facilitate growth and reduce single points of failure

Gaps in downstream definition, missed communications, unclear goals, and contradictions among SE disciplines

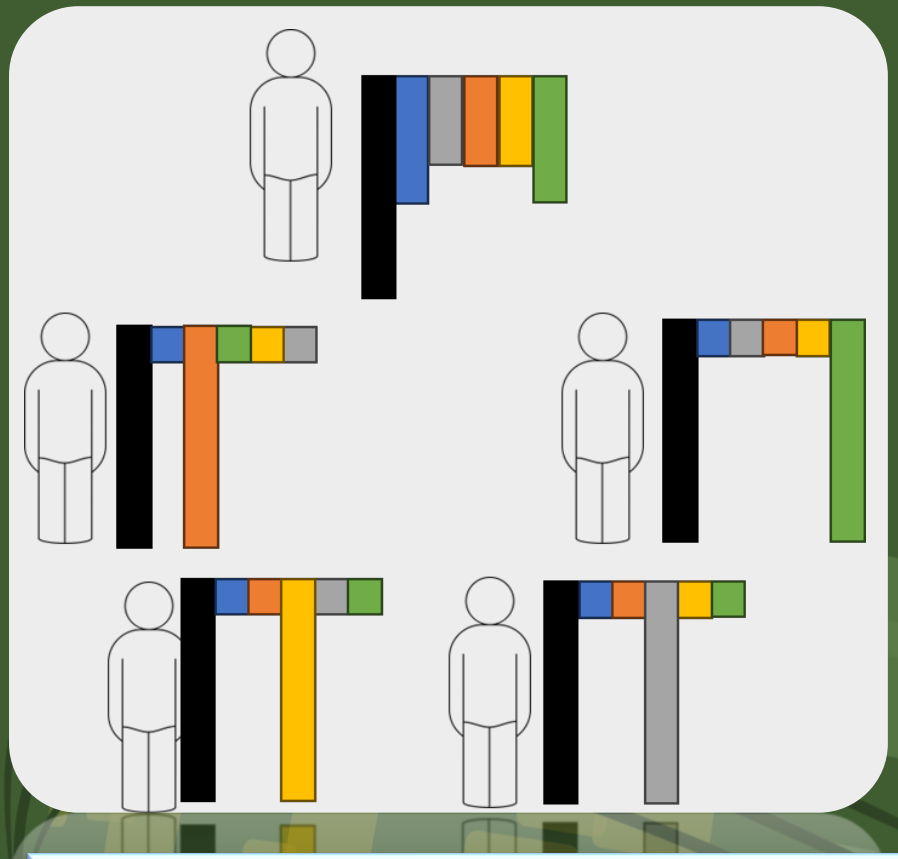
Single-Role to Hybrid Multi-Role

- Worked with each team member to identify strengths, weaknesses, and desires
  - Team members wanted more challenging work; expert in one area hindering growth and career
- Elicited feedback from downstream teams and ideas on how to improve
  - Too many interfaces and inconsistent direction

- **Current team structure needed changed**

Shift to a Hybrid Multi-Role with Lead SE was apparent

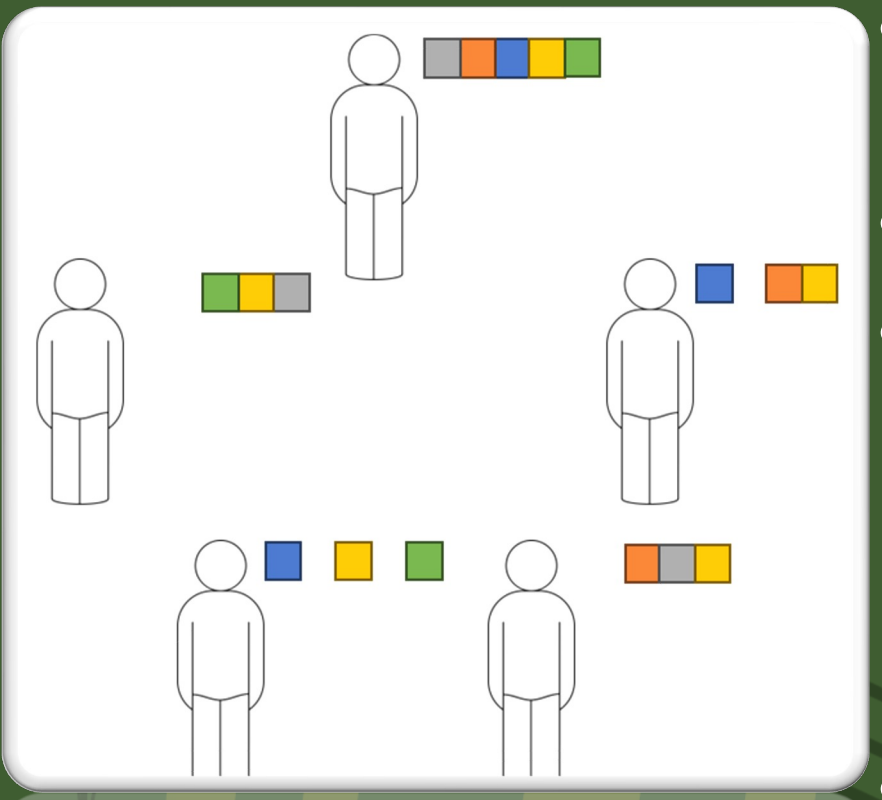
Single-Role to Hybrid Multi-Role



- Restructured the team giving each SE one or two subsystems and a specialist role for an SE subdiscipline
- Downstream teams now had a single interface for all design definition and experts remained for each discipline
- Team members expanded skills into SE areas they wanted to learn, learning from more experienced engineers in that area

Team morale increased, system definition improved, downstream interactions more effective, program regained schedule!

Embedded Multi-Role SEs on Agile Project



- Medium-sized software project utilizing Agile development approach
- Unclassified and classified aspects
- Teams already established as a loose unclassified only multi-role team
  - Overall lacking in some SE disciplines such as architecture and interfaces
- Product Owners were developers, leaving SEs to do PO work

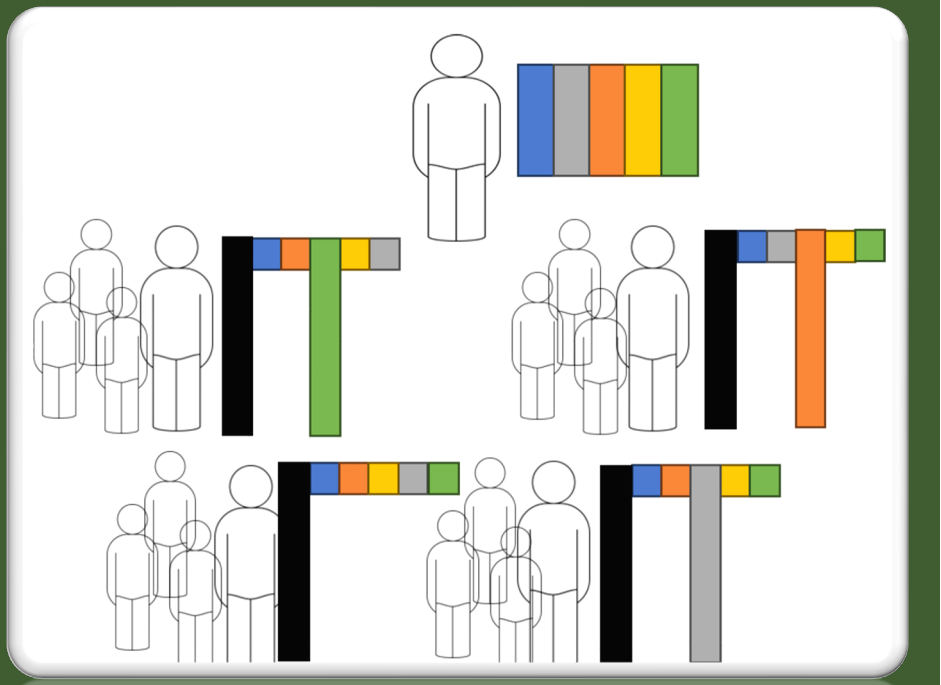
Embedded Multi-Role SEs on Agile Project

- Lack of SE expertise and formal structure lead to insufficient design detail for Features and Stories
- Development teams had knowledge gaps in critical areas for software development (i.e. interfacing, requirements, customer desire)
- Planning took three days or longer and key dependencies were missed
- SEs not cleared to classified work => little understanding of holistic system and “big picture”

Lack of structure and poor documentation led to lengthy development timelines and rework

Embedded Multi-Role SEs on Agile Project

- Restructuring and refocusing was necessary
- Created an Embedded Multi-Role team with a Lead SE
- Got more SEs briefed or on a team with a briefed Product Owner
- Lead SE was “Release Train Engineer”, facilitated cross-collaboration, and ensured holistic design definition



Embedded Multi-Role SEs on Agile Project

- Being cleared allowed SEs to understand the larger mission and end user needs
- Formal structure with ownership at the development team level and an overarching lead
- More effective design definitions and dependency tracking

Effective design/dependency definitions, planning time reduced to less than a day, development velocity increased, rework drastically reduced!

Conclusions

Conclusions

- Programs have a team structure where an SE team is a part, but the structure of the SE team itself is often neglected
- Few possible structures of an SE team discussed with advantages and disadvantages
- Two real-world examples presented proving the value of defining the SE team structure considering team dynamics, growth opportunities, and program needs

Thinking about SE team structure improves team dynamics and leads to more effective Systems Engineering

Thank you!

Questions

34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

www.incose.org/symp2024

#INCOSEIS