Fundamental Research in Systems Engineering:

Asking “Why?” rather than “How?”

Chris Paredis
Program Director
NSF ENG/CMMI Engineering & Systems Design, Systems Science
cparedis@nsf.gov
(703) 292-2241

Disclaimer

- Any opinions, findings, and conclusions or recommendations expressed in these slides are those of the author/presenter and do not necessarily reflect the views of the National Science Foundation.

Outline

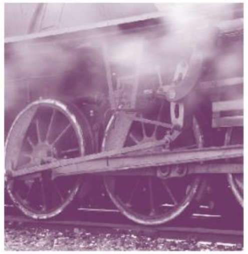
- Asking “Why?”
  - Why do we engineer artifacts?
  - Why do we engineer complex artifacts?
  - Why do we engineer systems of systems?
  - Why do we use systems engineering methods?
- A normative perspective: maximize value
- From best practices to a theoretical foundation
- Key take-aways

Why Do We Engineer Artifacts?


5000 BC


1200 AD


1750 AD


1850 AD


1900 AD


1980 AD


2010 AD

Why?

Because the artifacts:

- Make life easier…
- Increase our chances of survival…
- Result in outcomes that are more preferred…

Add Value!

What do we Mean by Value?

Value is an Expression of the Preferences of the Designer

- Value is an expression of preference – the more an outcome is preferred, the higher the value assigned to it
  - A philanthropist may assign high value to an alternative that significantly increases well-being even if it cannot be produced at a profit
  - An environmentalist may assign high value to environmentally friendly, sustainable alternatives
  - A publicly traded company may assign high value to profitable alternatives
- Value is often expressed in monetary terms
  - If a designer prefers outcome A over outcome B then he/she is willing to pay an amount of $\Delta v = v_A - v_B$ to exchange B for A

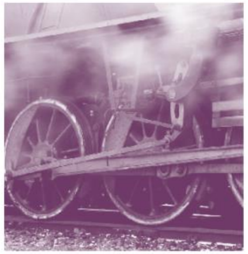
Why Do We Engineer Artifacts?

The Value to the Engineer/Designer May be Indirect


5000 BC


1200 AD


1750 AD


1850 AD


1900 AD


1980 AD


2010 AD

1. Individual designer — artifact for personal use
   - Designer obtains added value directly from artifact use
2. Individual designer — artifact for sale
   - Trading → consumer surplus + producer surplus
   - Through trading, both consumer and producer benefit
3. Designer in firm — artifact for sale
   - Producer surplus received by firm → firm pays designer’s salary
   - Organizing in firms is beneficial because it reduces transaction costs

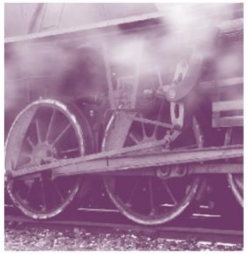
Why Do We Engineer Artifacts?

The Value to the Engineer/Designer May be Indirect


5000 BC


1200 AD


1750 AD


1850 AD


1900 AD


1980 AD


2010 AD

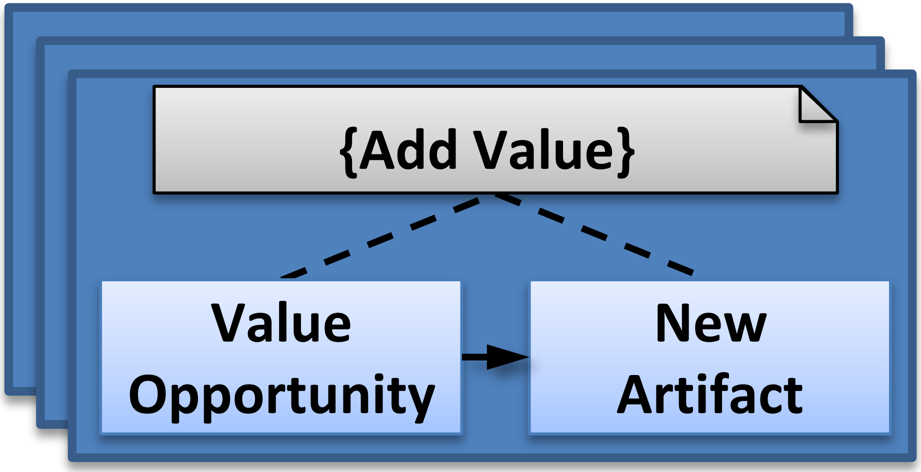
1. Individual designer — artifact for personal use
   –
2. I
   –
   – Through trading, both consumer and producer benefit
3. D
   – alary
   – Organizing in firms is beneficial because it reduces transaction costs

Engineers Design Artifacts…
Because Doing so Adds Value…
to Themselves

Satisfying Customer Needs is not the Primary Goal
but is a Means for Adding Value to the Engineer

Why Do We Engineer Artifacts?

Identifying and Capitalizing on Value Opportunities



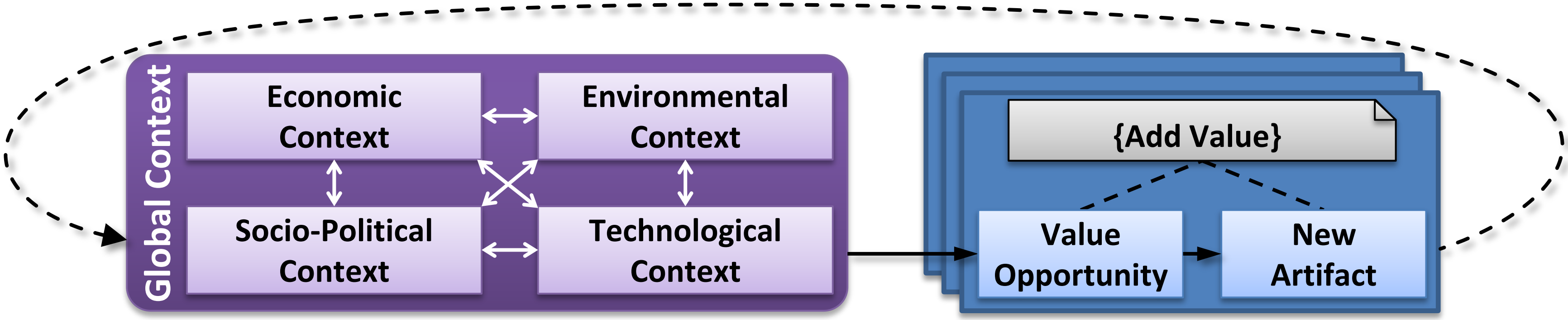
{Add Value}

Value Opportunity

New Artifact

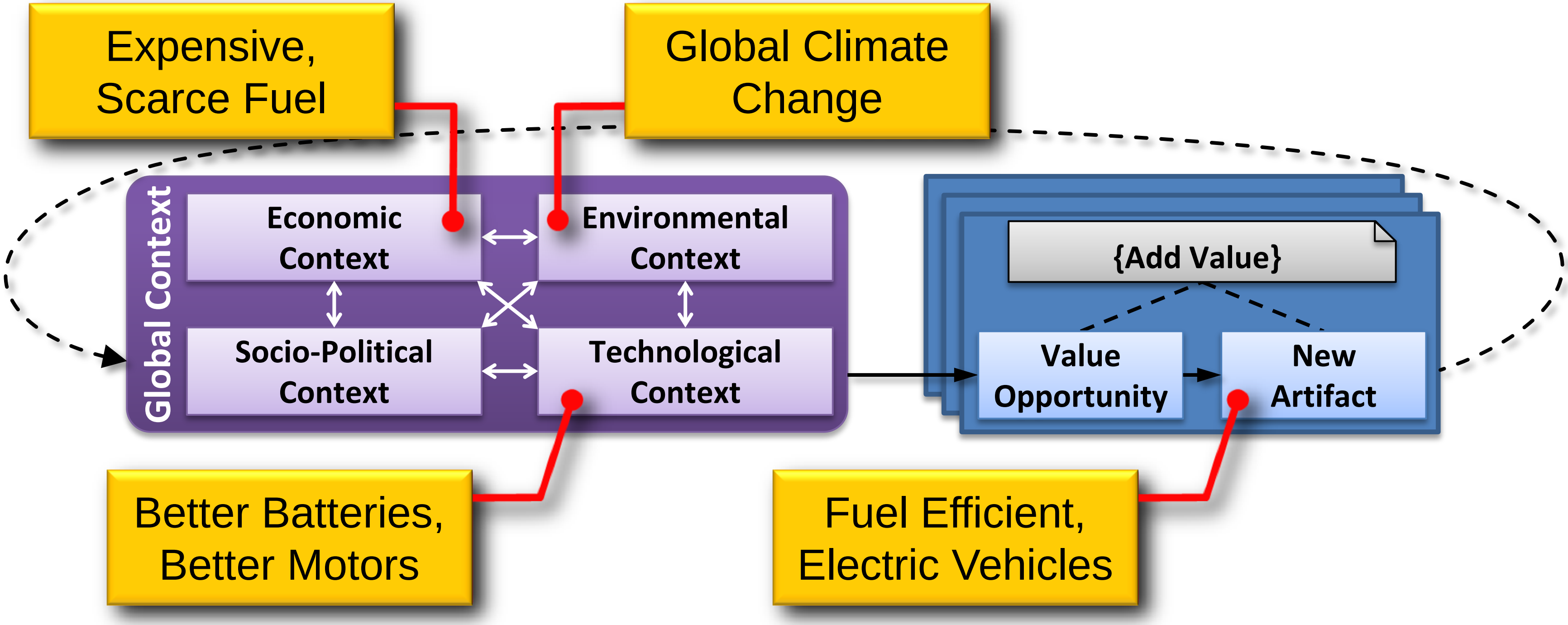
Why Do We Engineer Artifacts?

Value Opportunities in a Global Context



Why Do We Engineer Artifacts?

Value Opportunities in a Global Context



Why Do We Engineer *Complex* Artifacts?

Complexity is a By-Product of our Desire for Functionality

Increasingly Rapid Expansion of Functionality

Electronic injection
Cruise control
Central locking

1970

Automatic gearbox
Climate control
ABS
Seat heating
Automatic mirrors

1990

Navigation system
Infotainment system
Adaptive cruise control
Xenon lighting
Voice input
Emergency call
Vehicle assist
Dynamic stability ctrl

2010

Night vision system
Pedestrian detection
Automatic parking
Voice control that actually works
Heads-up display
Integrated into the internet of things
Battery electric (BEV)

…

V2V communication
Driverless

>2015

Why Do We Engineer *Complex* Artifacts?

Added Functionality, but at a Cost

- Additional Functionality → Additional Value Potential
  - Artifact is likely more desired by customers and can be sold at a higher price

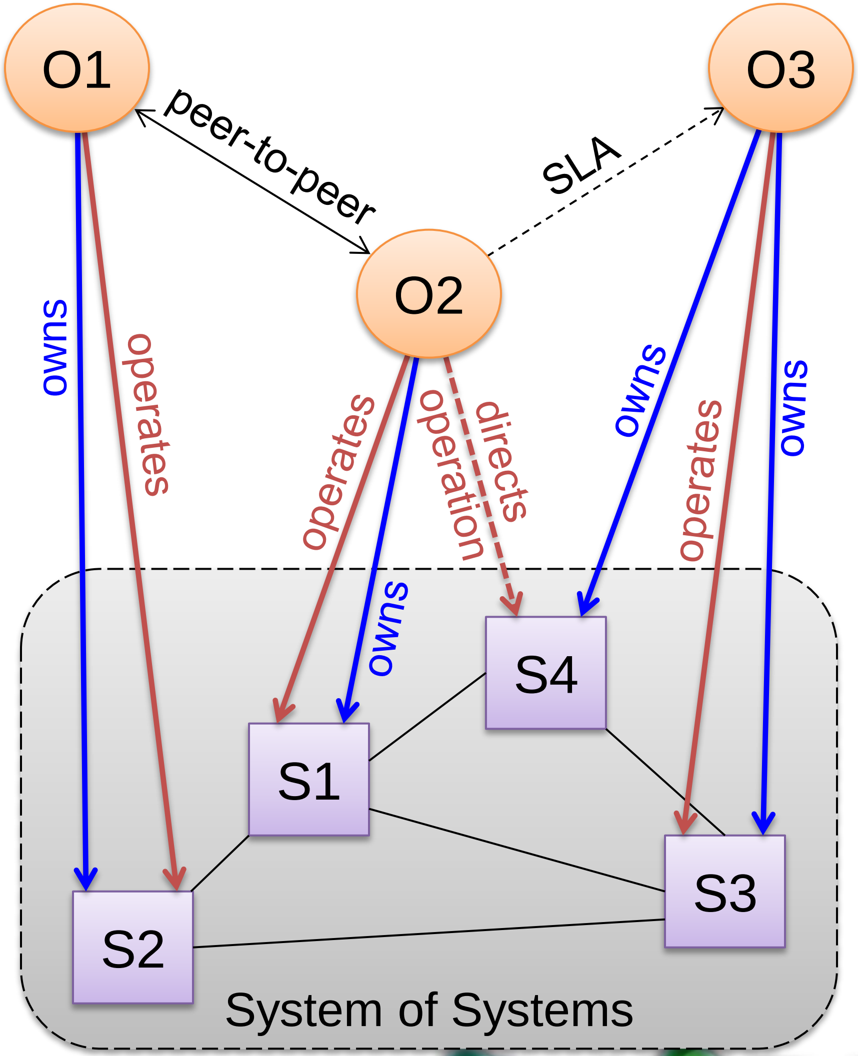
- Additional Complexity → Additional Cost
  - Additional functionality
    → additional parts & interfaces
    → additional experts
    → more complex interactions
    → increased opportunity for failure

Because Additional Functionality Adds Value!

Why Do We Engineer Systems of Systems?

Why Give up Control and Introduce Organizational Complexity?

- SoS Characteristics
  - Evolving over time
  - Multiple owners
  - Multiple, independent designers
  - Operational and managerial independence
- Challenges
  - Socio-technical problem
  - Uncertain, evolving
  - Requires flexibility, interoperability



Why Do We Engineer Systems of Systems?

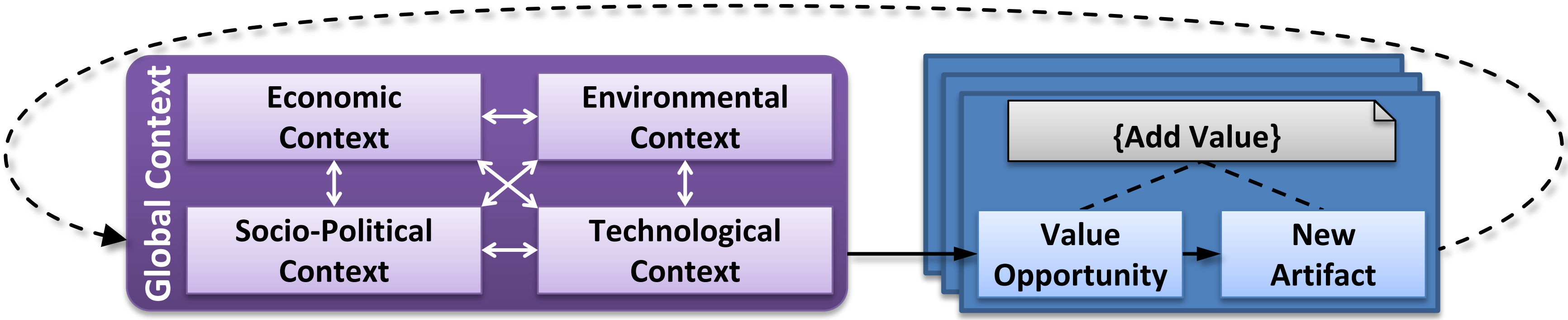
Flexibility, Evolvability, Modularity…Add Value over Time

- Providing increased functionality in a fast-changing global context → added value
- Modularity → Lower entry-cost for innovators
- Modularity → evolvability → continually upgrading functionality with smaller capital investments spread out over time
- Modularity → flexibility → adapt to uncertain future
- Multiple stakeholders → shared risk
- …

Because It May Add Value!

Why Do We Engineer Artifacts?

Identifying and Capitalizing on Value Opportunities


Global Context

Economic Context

Environmental Context

Socio-Political Context

Technological Context

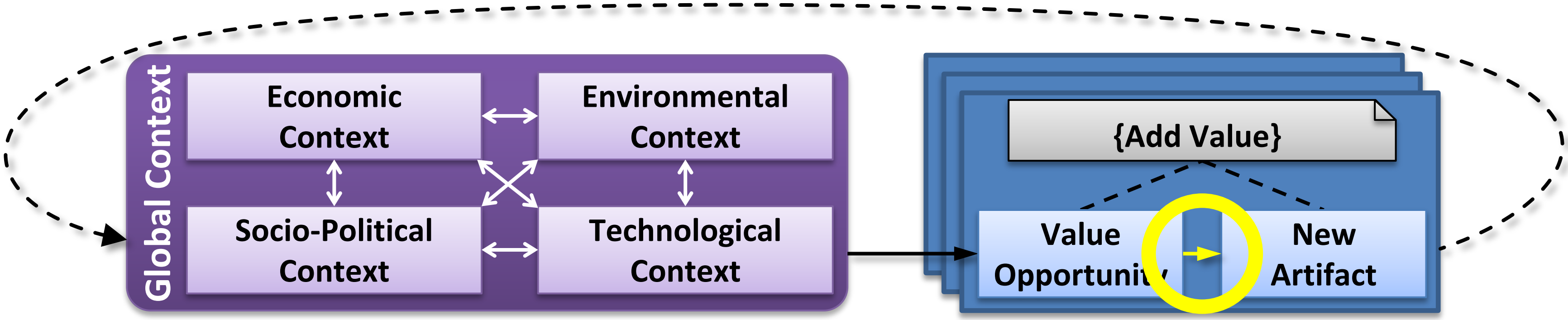
{Add Value}

Value Opportunity

New Artifact

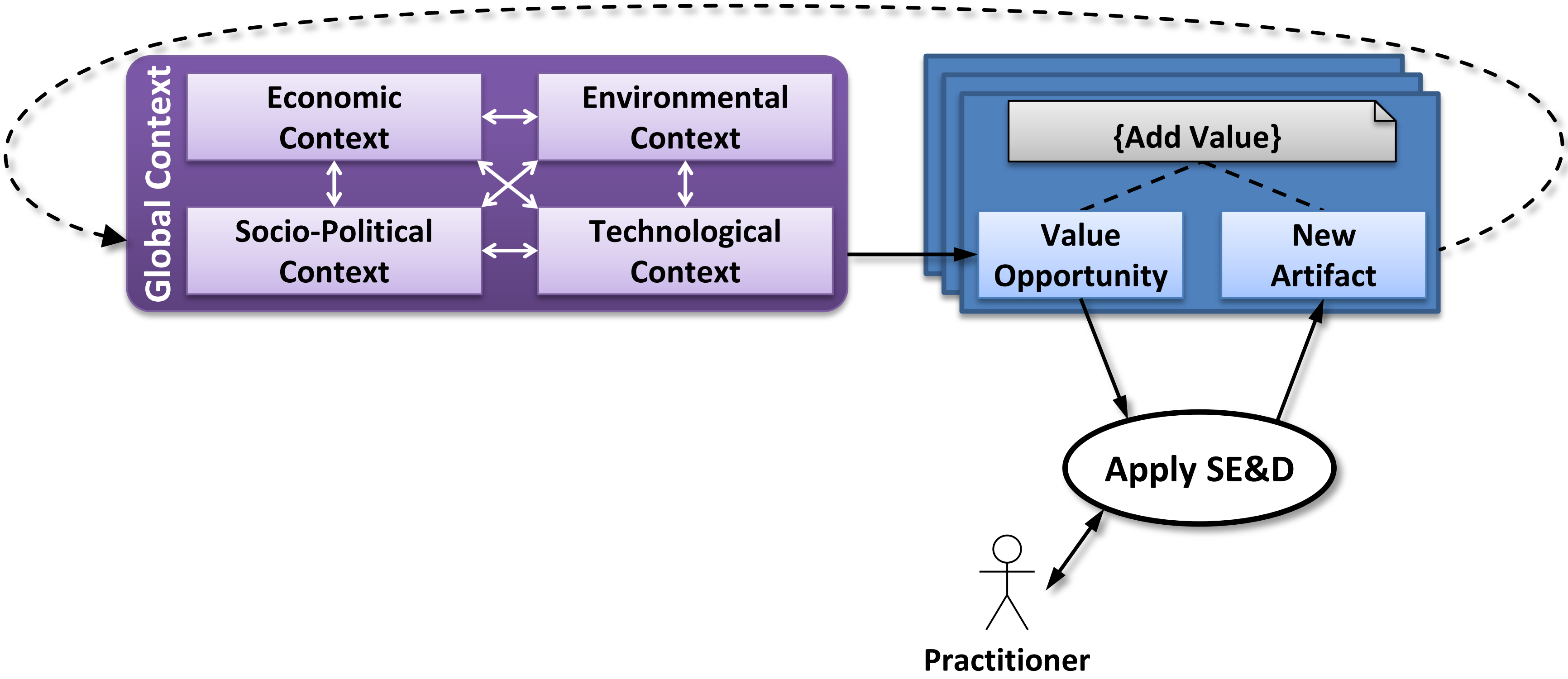
Why Use Systems Engineering Methods?

SE Methods also Influence the Value



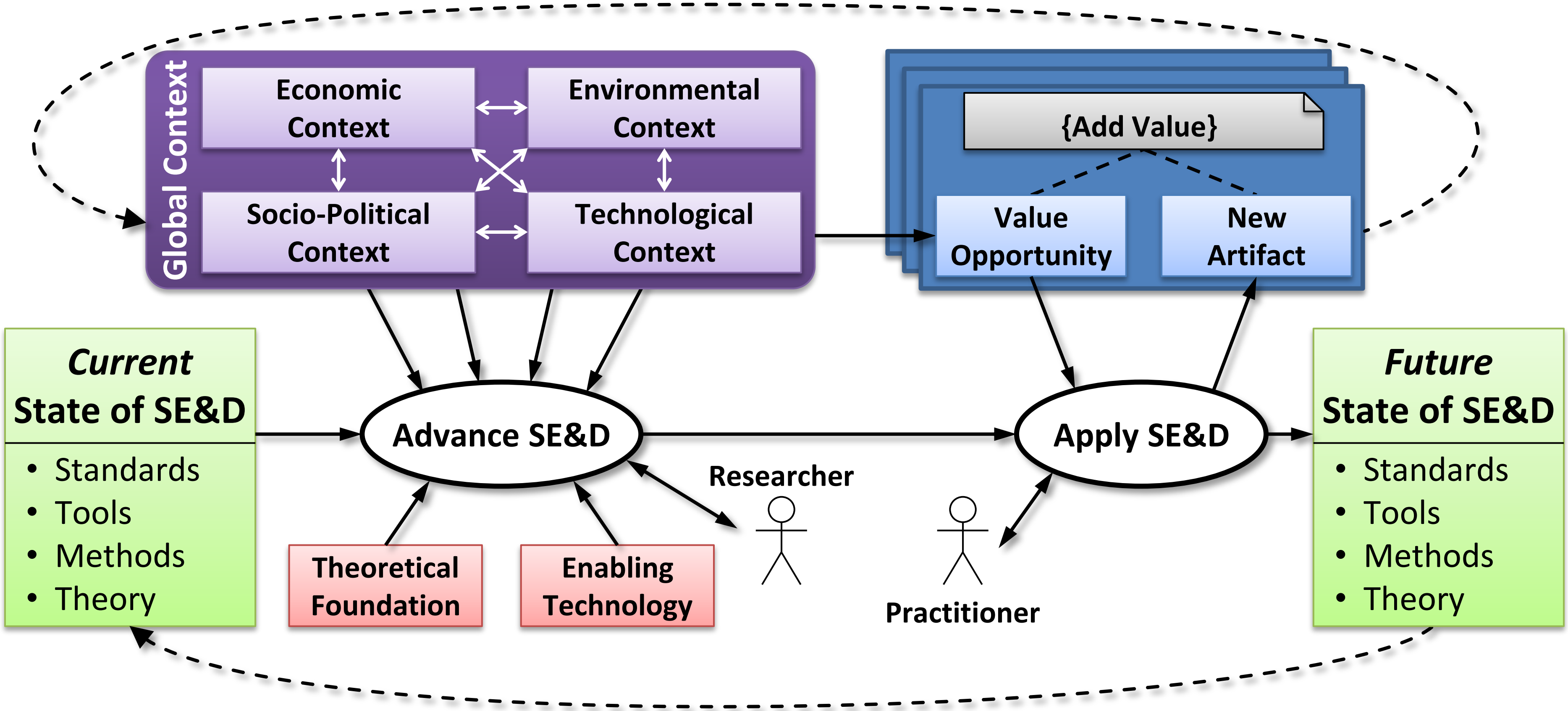
Why Use Systems Engineering Methods?

Better SE Methods Add Value



Why Use Systems Engineering Methods?

Continuous Advances in SE Methods, Driven by Competition



Outline

- Asking "Why?"
  - – Why do we engineer artifacts?
  - – Why do we engineer complex artifacts?
  - – Why do we engineer systems of systems?
  - – Why do we use systems engineering methods?
- A normative perspective: Maximize value
- From best practices to a theoretical foundation
- Key take-aways

Because it
Adds Value!

Outline

- Asking "Why?"
  - Why do we engineer artifacts?
  - Why do we engineer complex artifacts?
  - Why do we engineer systems of systems?
  - Why do we use systems engineering methods?
- ➡ A normative perspective: Maximize value
- From best practices to a theoretical foundation
- Key take-aways

Because it Adds Value!

A Normative Perspective

Maximizing Value

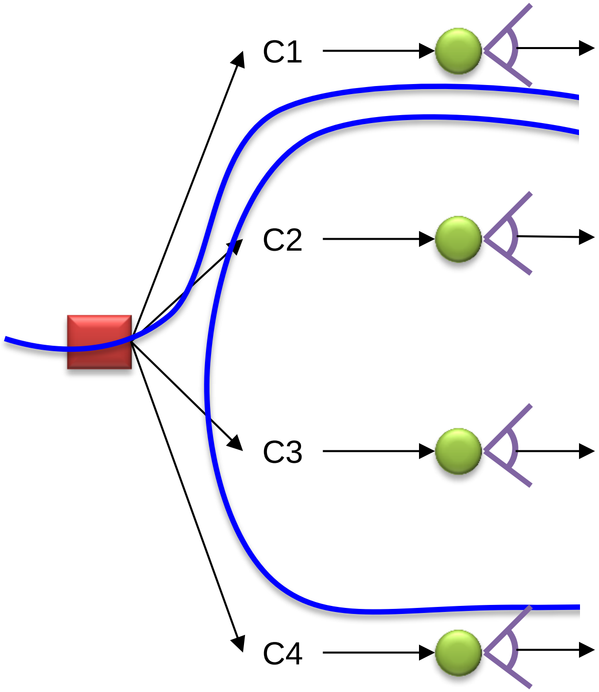
- The goal of SE researchers is to ***improve*** SE
- But what do we mean by “improve”?
  - What makes a good SE method?
  - How do we measure “goodness”?

- Normative Statement:

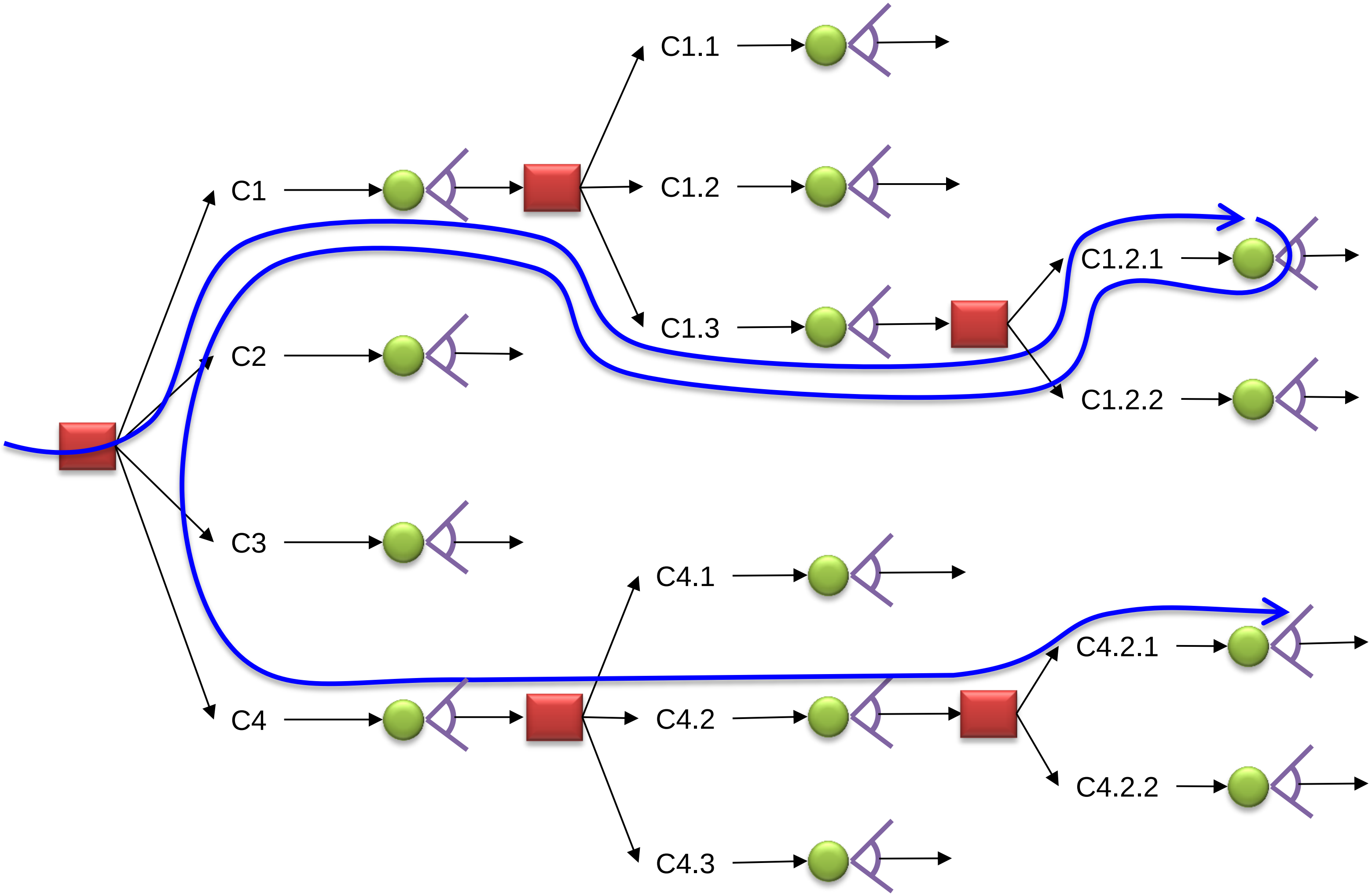
A good systems engineering method helps
a systems engineer achieve outcomes
that are most preferred to him or her

→ Maximize the Engineer’s Value

Why Should Good SE Methods Adopt Gradual Refinement of the System Specification?

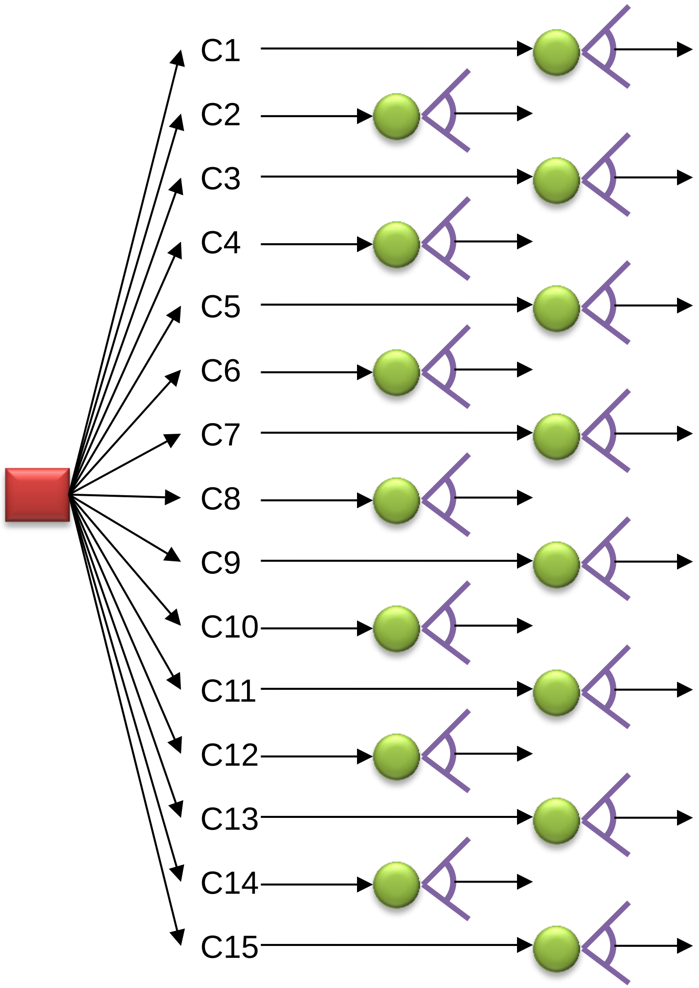


Why Should Good SE Methods Adopt Gradual Refinement of the System Specification?



Why Gradual Refinement?

Gradual Refinement of System Specification



- **Exhaustive Search:**
  Cost of synthesis and analysis is too high

- **Gradual refinement of system specification:**
  is advantageous because it allows for pruning → fewer specifications are considered

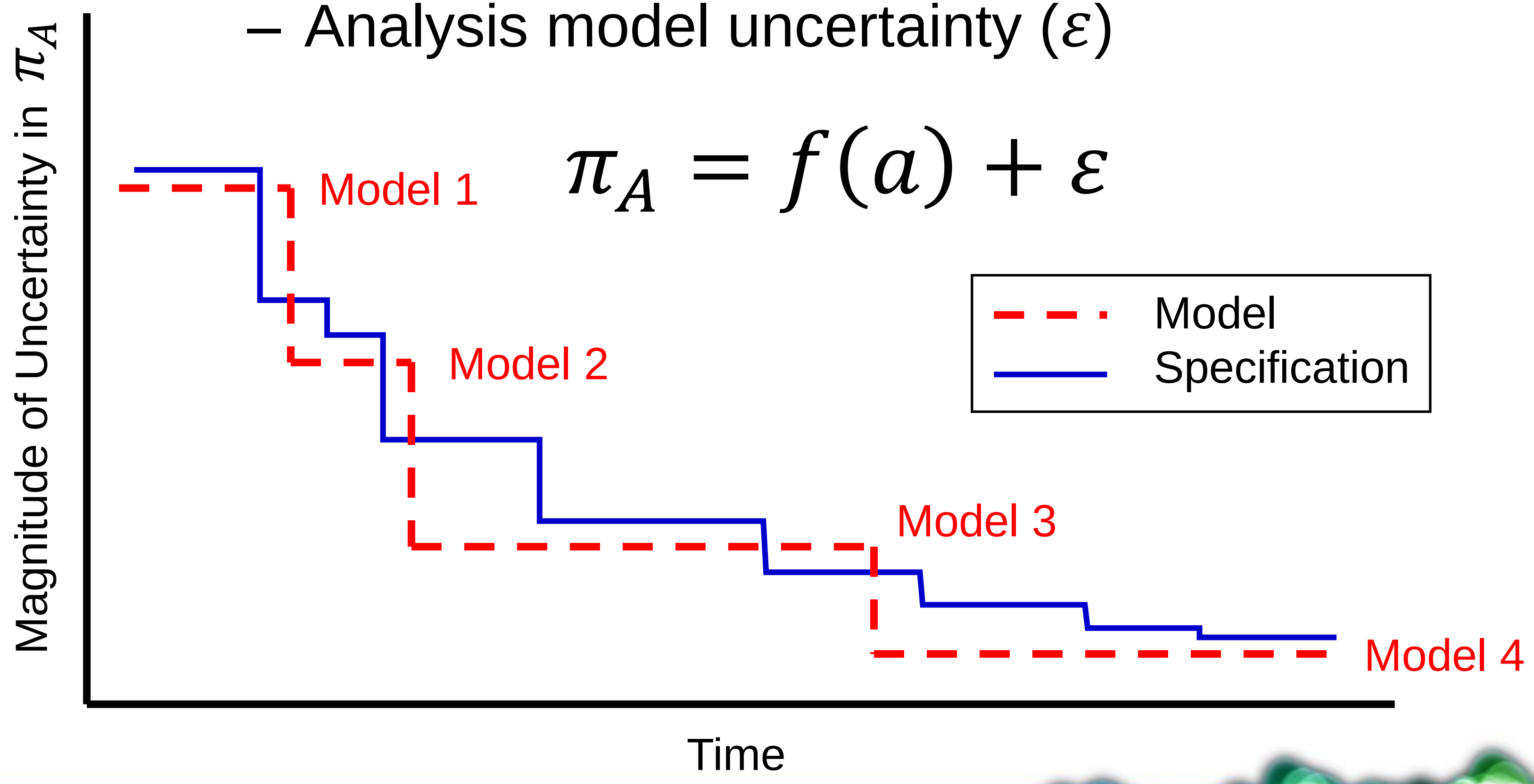
- But caries a risk that the most preferred alternative
  is also pruned

Why Gradual Refinement?

Gradual Increase in Analysis Accuracy

- Uncertainty in prediction of artifact value, π_A, results from:
  - Specification uncertainty (uncertainty in a)
  - Analysis model uncertainty (ε)

$$\pi_A = f(a) + \varepsilon$$



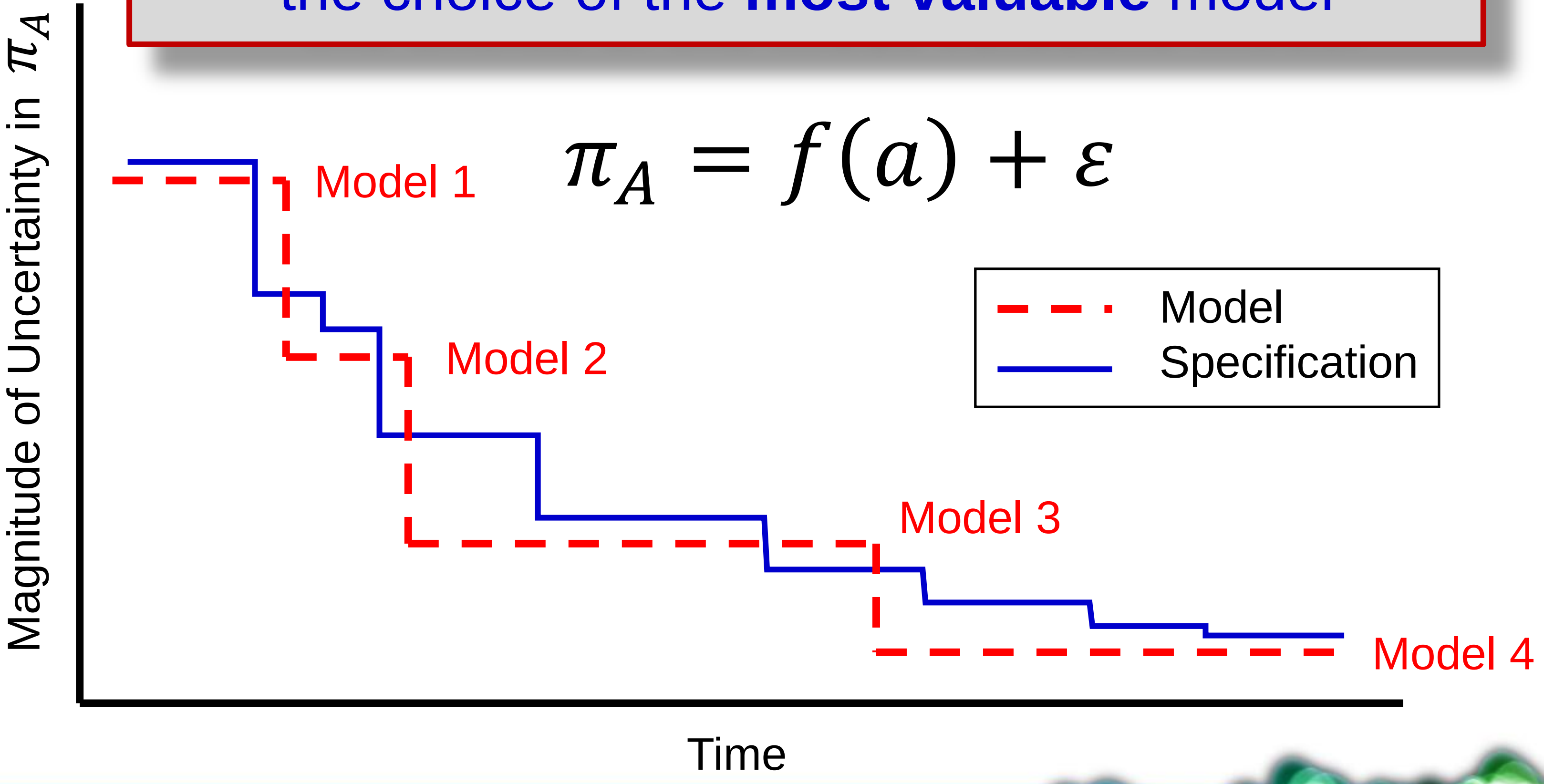
Why Gradual Refinement?

Gradual Increase in Analysis Accuracy

- Unc sults from

How should we choose which model to use?

Value-of-Information Theory guides us to the choice of the **most valuable** model



$$\pi_A = f(a) + \varepsilon$$

Why Gradual Refinement?

An Explanatory Model Justifying Gradual Refinement

- Think of SE as a search process
- Maximizing the value π_A of an artifact a:

$$\mathcal{A}\colon \max_{a\in A} \pi_A(a) \longrightarrow$$

Overlooks importance of uncertainty…

$$\mathcal{A}\colon \max_{a\in A} \mathrm{E}[u(\pi_A(a))] \longrightarrow$$

Overlooks importance of the search process…

Why Gradual Refinement?

An Explanatory Model Justifying Gradual Refinement

- Think of SE as a search process
- Maximizing the value π_A of an artifact a:

$$\mathcal{A}: \max_{a \in A} \pi_A(a) \longrightarrow$$

Overlooks importance of uncertainty…

$$\mathcal{A}: \max_{a \in A} \mathrm{E}[u(\pi_A(a))] \longrightarrow$$

Overlooks importance of the search process…

- The search process requires time and resources:

$$\mathcal{A}: \max_{a \in A} \mathrm{E}\left[u\left(\pi_A\left(a, t(\mathcal{A})\right) - C(\mathcal{A})\right)\right]$$

Systems Engineering: A Search Process

Value of the Artifact minus Development Cost

- Maximization problem becomes Self-Referential!

$$\mathcal{A}: \max_{a \in A} \mathrm{E}\left[u\left(\pi_A\left(a, t(\mathcal{A})\right) - C(\mathcal{A})\right)\right]$$

- Leads to infinite planning recursion
  - To achieve the optimal outcome the problem needs to be optimally framed
  - To find the optimum frame, the framing problem needs to be optimally framed
  - …

→ heuristics are required

Systems Engineering: A Search Process

Artifact is the Outcome of a Process

- Maximizing the value π_A of an artifact that results from a process p:

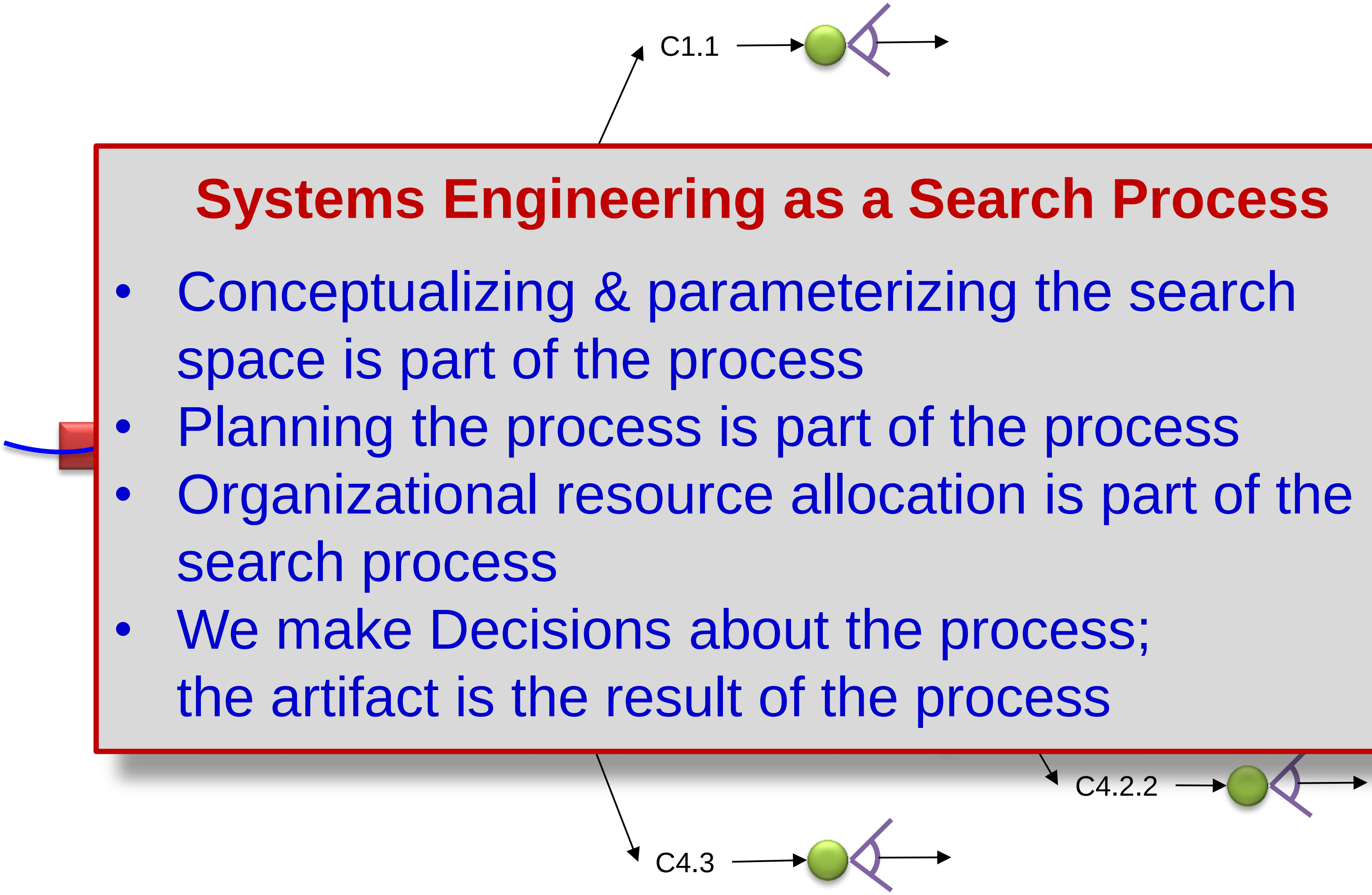
$$\mathcal{P}: \max_{p\in P} \mathrm{E}\left[u\left(\boxed{\pi_A\left(a(p), t_p(p)\right)} - \boxed{C_p(p)}\right)\right]$$

- No longer self-referential, but still dynamic in the sense that future process steps depend on the outcomes of previous process steps
- Search strategy, p, and resulting artifact are inextricably linked

→ Must make a tradeoff between artifact value and search time & cost

Systems Engineering: A Search Process

Artifact is the Outcome of a Process



Systems Engineering as a Search Process

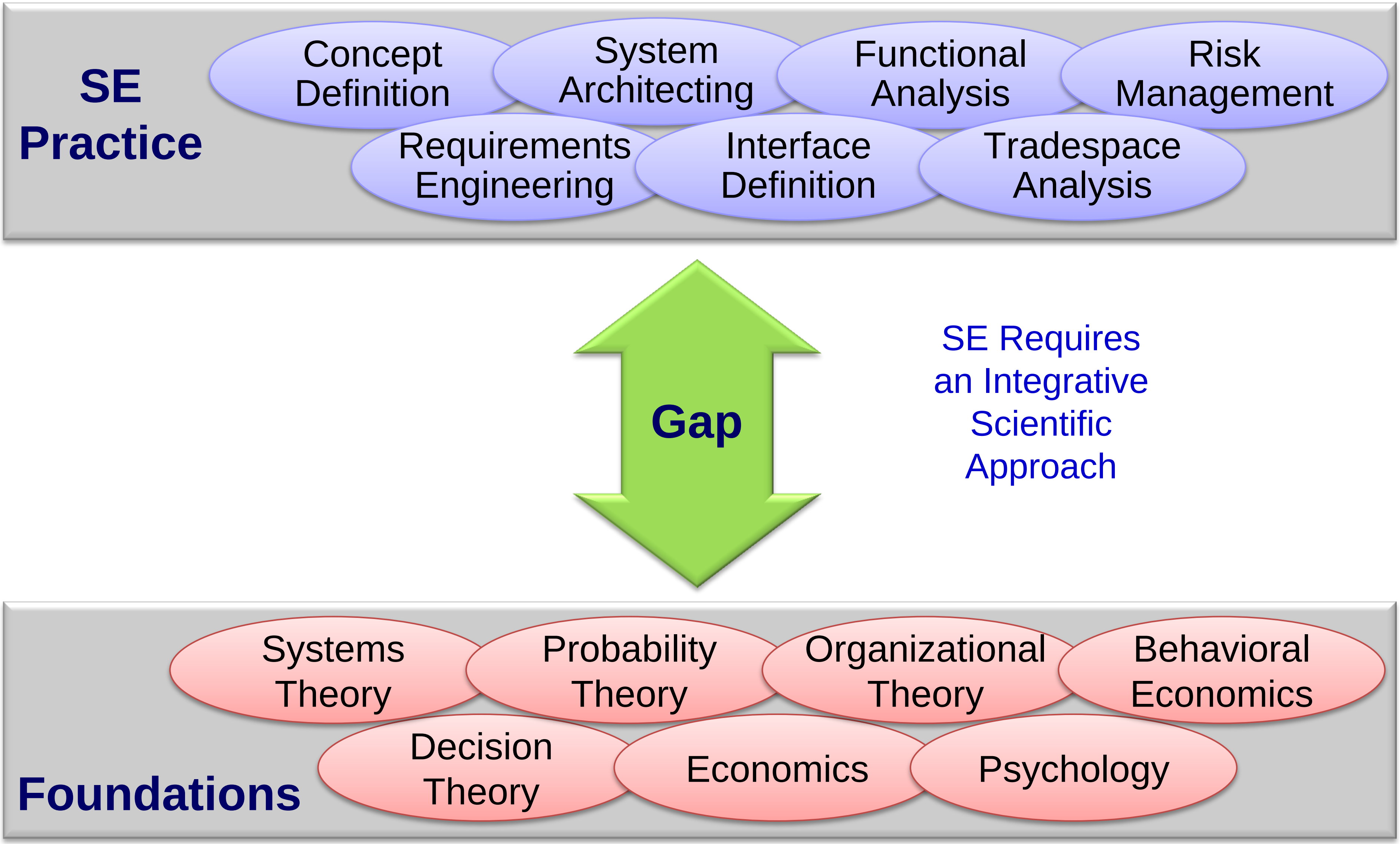
- Conceptualizing & parameterizing the search space is part of the process
- Planning the process is part of the process
- Organizational resource allocation is part of the search process
- We make Decisions about the process;
  the artifact is the result of the process

Outline

- Asking “Why?”
  - Why do we engineer artifacts?
  - Why do we engineer complex artifacts?
  - Why do we engineer systems of systems?
  - Why do we use systems engineering methods?
- A normative perspective: Maximize value
- ➡ From best practices to a theoretical foundation
- Key take-aways

A Theoretical Foundation for SE

Bridging the Gap Between Best Practices and Foundations



SE Practice

Concept Definition

System Architecting

Functional Analysis

Risk Management

Requirements Engineering

Interface Definition

Tradespace Analysis

Gap

SE Requires an Integrative Scientific Approach

Foundations

Systems Theory

Probability Theory

Organizational Theory

Behavioral Economics

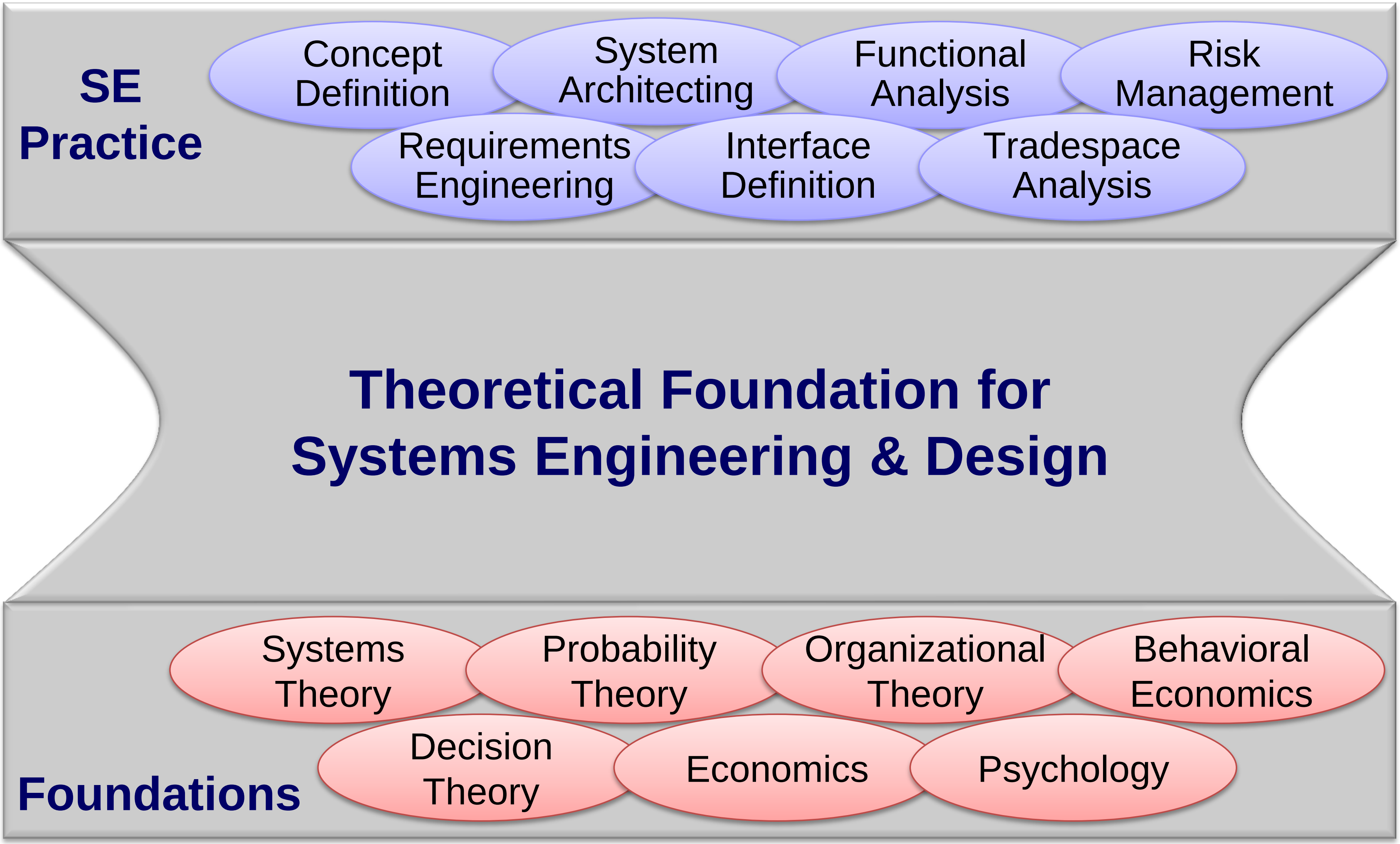
Decision Theory

Economics

Psychology

A Theoretical Foundation for SE

Bridging the Gap Between Best Practices and Foundations



SE Practice

- Concept Definition
- System Architecting
- Functional Analysis
- Risk Management
- Requirements Engineering
- Interface Definition
- Tradespace Analysis

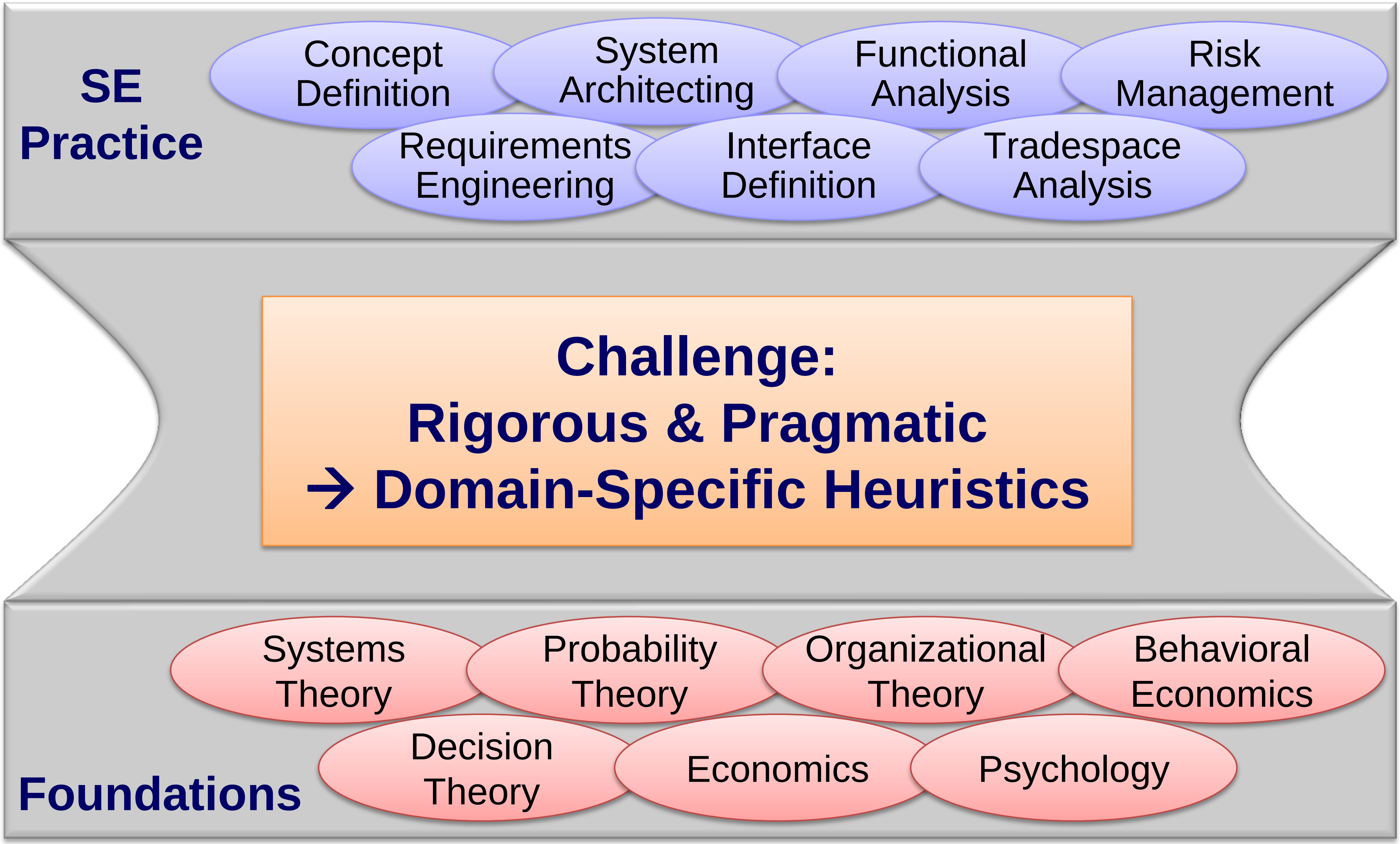
Theoretical Foundation for Systems Engineering & Design

- Systems Theory
- Probability Theory
- Organizational Theory
- Behavioral Economics
- Decision Theory
- Economics
- Psychology

Foundations

A Theoretical Foundation for SE

Bridging the Gap Between Best Practices and Foundations



SE Practice

- Concept Definition
- System Architecting
- Functional Analysis
- Risk Management
- Requirements Engineering
- Interface Definition
- Tradespace Analysis

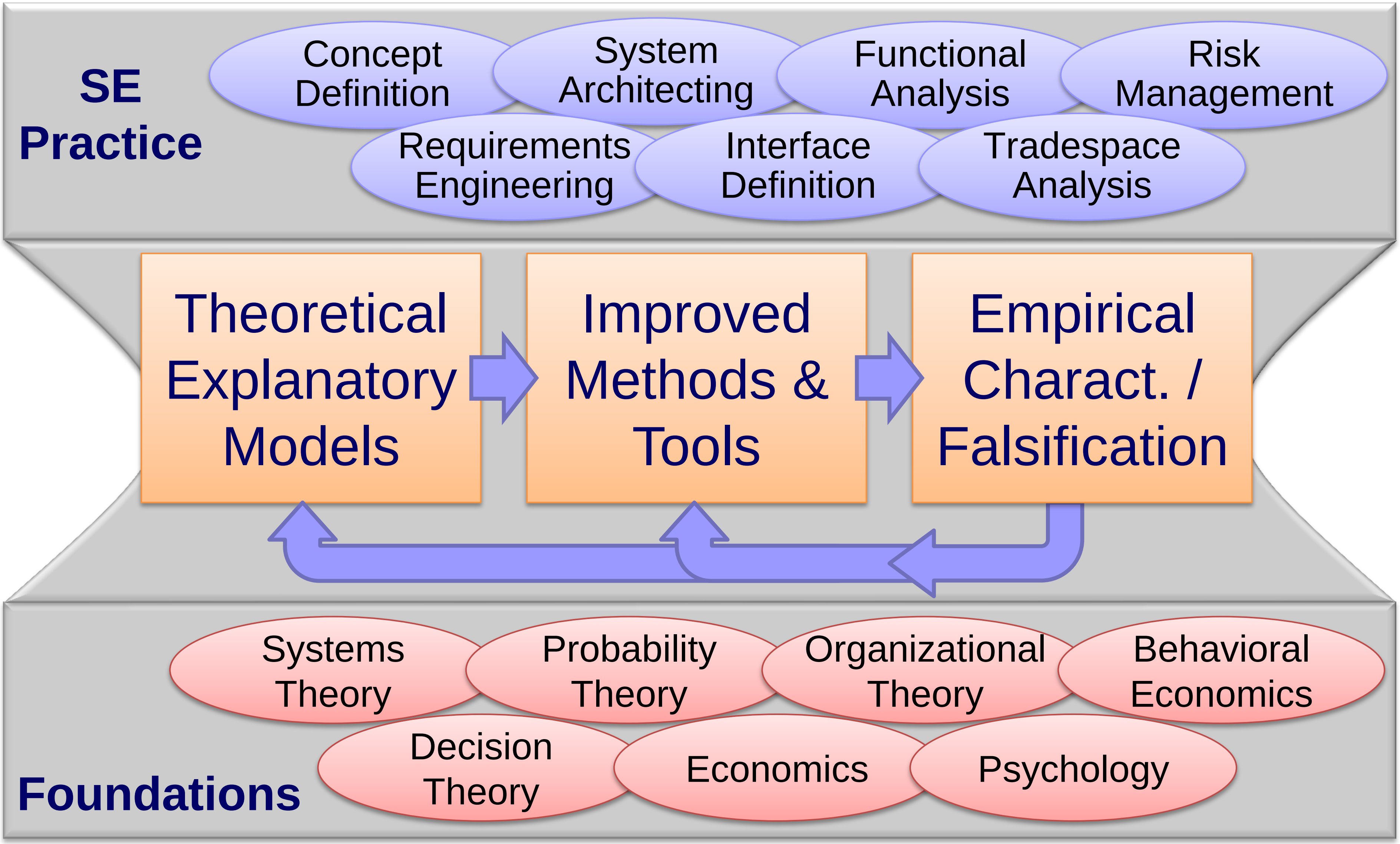
Challenge:
Rigorous & Pragmatic
→ Domain-Specific Heuristics

- Systems Theory
- Probability Theory
- Organizational Theory
- Behavioral Economics
- Decision Theory
- Economics
- Psychology

Foundations

A Theoretical Foundation for SE

Explanatory Models Supported by Empirical Evidence



SE Practice

Concept Definition

System Architecting

Functional Analysis

Risk Management

Requirements Engineering

Interface Definition

Tradespace Analysis

Theoretical Explanatory Models

Improved Methods & Tools

Empirical Charact. / Falsification

Systems Theory

Probability Theory

Organizational Theory

Behavioral Economics

Decision Theory

Economics

Psychology

Foundations

Key Take-Aways

- In engineering practice we ask: “How?”

  In research on engineering practices we should ask:

  “Why?”

- We engineer new artifacts…

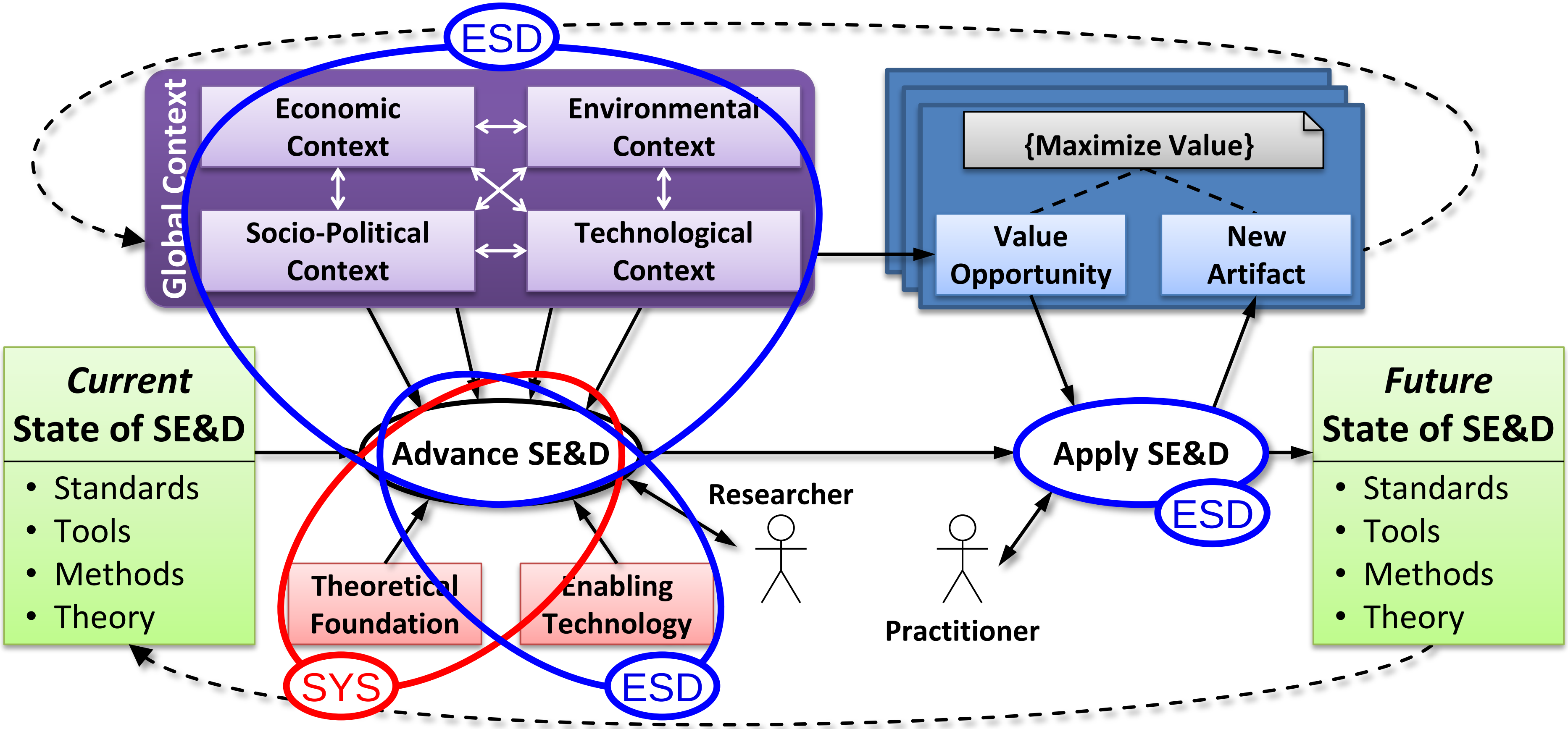
  because doing so adds value… to the engineer

- Normative: Good SE methods maximize value

- Goal: A theoretical foundation that consists of explanatory models supported by empirical evidence

ESD & SYS Program Overview

What is the Scope of Each Program?



http://tinyurl.com/ESD-SYS

ESD & SYS Program Opportunities & Logistics

What you need to know to submit your proposal

- Unsolicited proposals submission windows
  - Fall: September 1-15
  - Spring: February 1-15
- Typical scope of proposals: 1-2 PIs, 1-2 PhD students, 3 yrs
- CAREER – proposals accepted for both SYS and ESD
  - Deadline: sometime mid-July 2015
  - Solicitation number: NSF 14-532 → to be updated for 2015
  - Budget: $500,000
- Interested in being a panelist?
  - E-mail me a 1-page description of your background & interests
- More info at:
  - ESD: https://www.nsf.gov/funding/pgm_summ.jsp?pims_id=13340
  - SYS: https://www.nsf.gov/funding/pgm_summ.jsp?pims_id=504788

Some References & Introductory Material

- H.A. Simon, *Sciences of the Artificial – 3rd Edition*, MIT Press, 1996.
- G. Hazelrigg, *Fundamentals of Decision Making for Engineering Design and Systems Engineering,* http://www.engineeringdecisionmaking.com/, 2012.
- G.S. Parnell, P.J. Driscoll, D.L. Henderson, *Decision Making in Systems Engineering and Management (2nd Edition)*, Wiley, 2010.
- J.M. Bernardo, A.F.M. Smith, *Bayesian theory*, Wiley, 2000.
- R. Gibbons, *Game Theory for Applied Economists,* Princeton University Press, 1992.
- D. Kahneman, *Thinking, Fast and Slow*, Farrar, Straus and Giroux, 2011.
- J. Brickley, J. Zimmerman Jr., C.W. Smith, *Managerial Economics & Organizational Architecture (5th Edition)*, McGraw-Hill, 2008.
- B.D. Lee, C.J.J. Paredis, "A Conceptual Framework for Value-Driven Design and Systems Engineering," *24th CIRP Design Conference*, Milan, Italy, April 14-16, 2014.