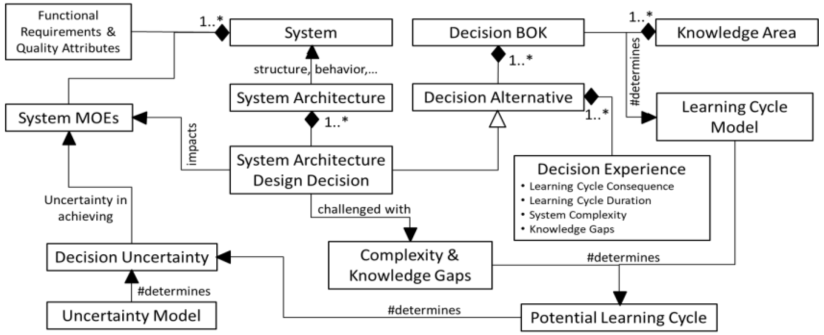
LPPDE, 2023 - April
Half-day Virtual Summit

THE SCIENCE OF IMPROVING INNOVATION



AI - Machine Learning Models for Aiding System Architecture Design Decisions

Dr. Ramakrishnan Raman
Fellow, Honeywell Aerospace
IEEE Distinguished Lecturer
Co-Chair – INCOSE AI Systems Working Group
Chair - IEEE Bangalore Section Systems Council
Fellow – Indian Society of Systems for Science & Engineering
ramakrishnan.raman@incose.net , ramakrishnan.raman@ieee.org
https://www.linkedin.com/in/ramakrishnanraman/

Presentation Outline

Introduction → System Architecture Design → ML Models → Conclusion

Introduction
- Machine Learning
- Complex Systems

System Architecture Design
- Design Decisions
- Learning Cycles

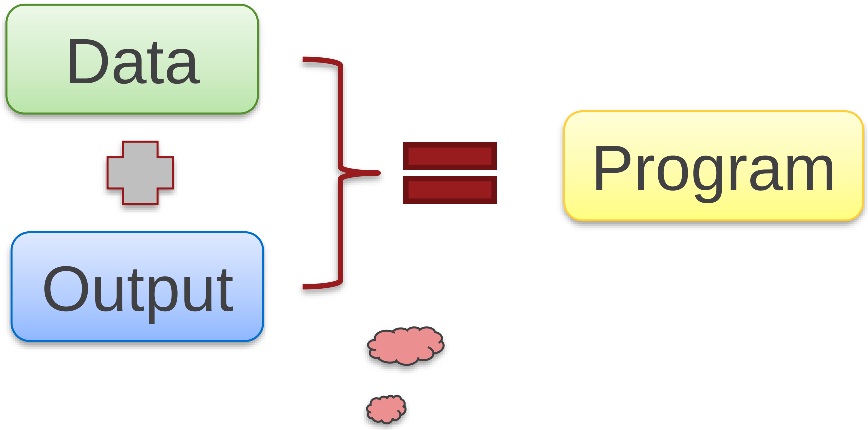
ML Models
- LCC prediction
- Uncertainty Assessments

Conclusion
- Opportunities
- Conclusion

Machine Learning

“Machine learning can be broadly defined as computational methods using experience to improve performance or to make accurate predictions”

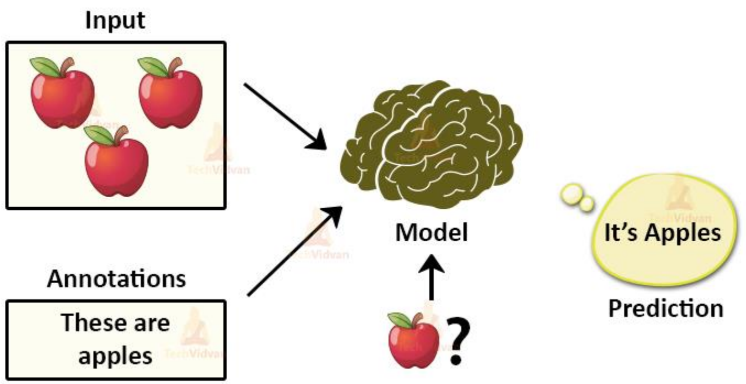
“Machine Learning represents the field of study that allows computer programs to learn without being explicitly programmed”



Src:Mohri M, Rostamizadeh A, Talwalkar A. Foundations of Machine Learning Cambridge, MA: MIT Press; 2012.

Machine Learning

- Machine Learning
  - Supervised Learning
  - Unsupervised Learning
  - Reinforcement Learning



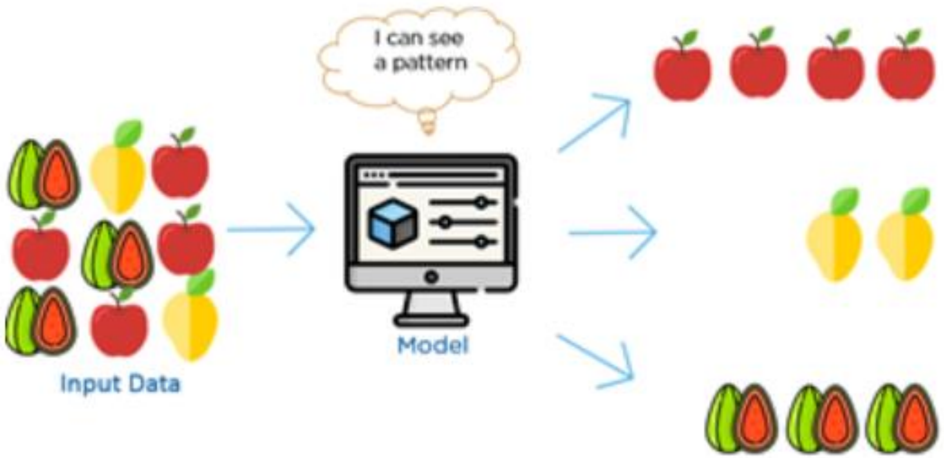
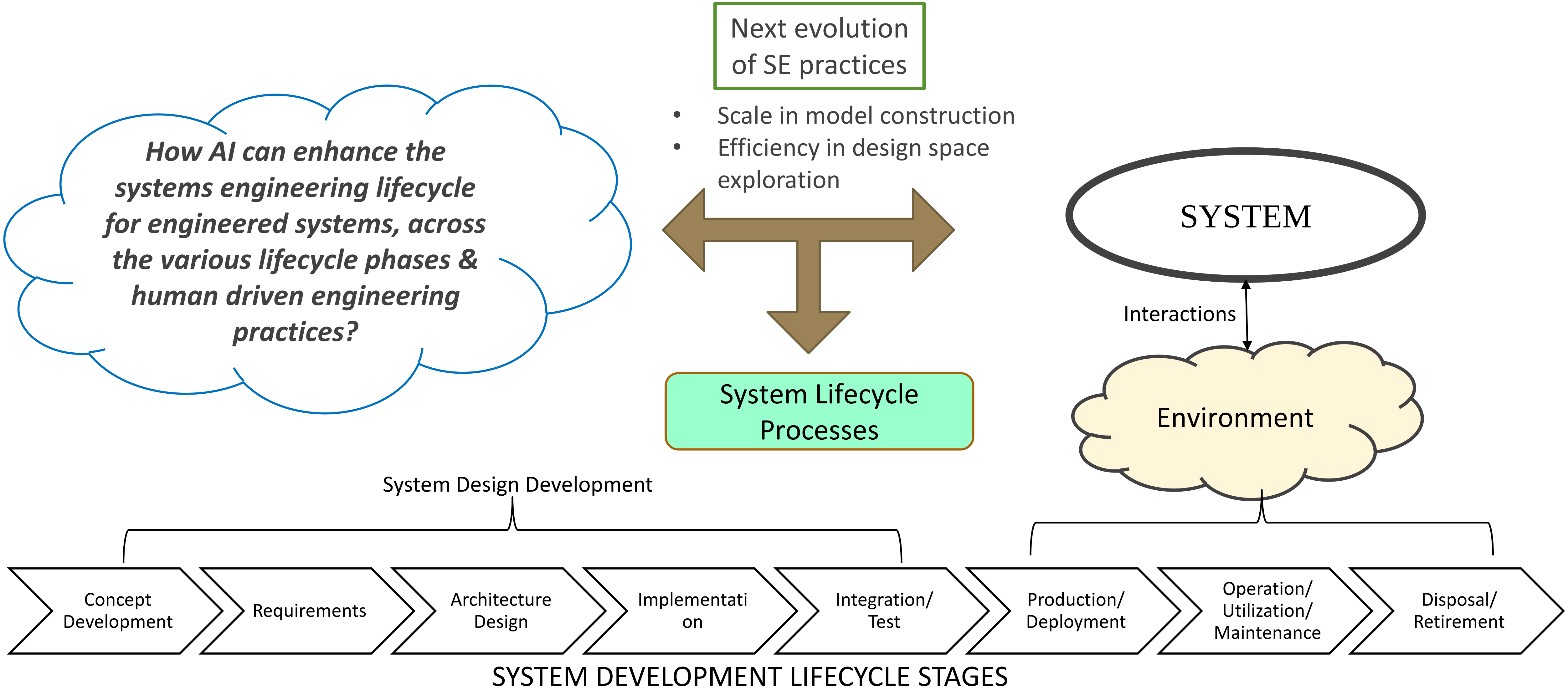
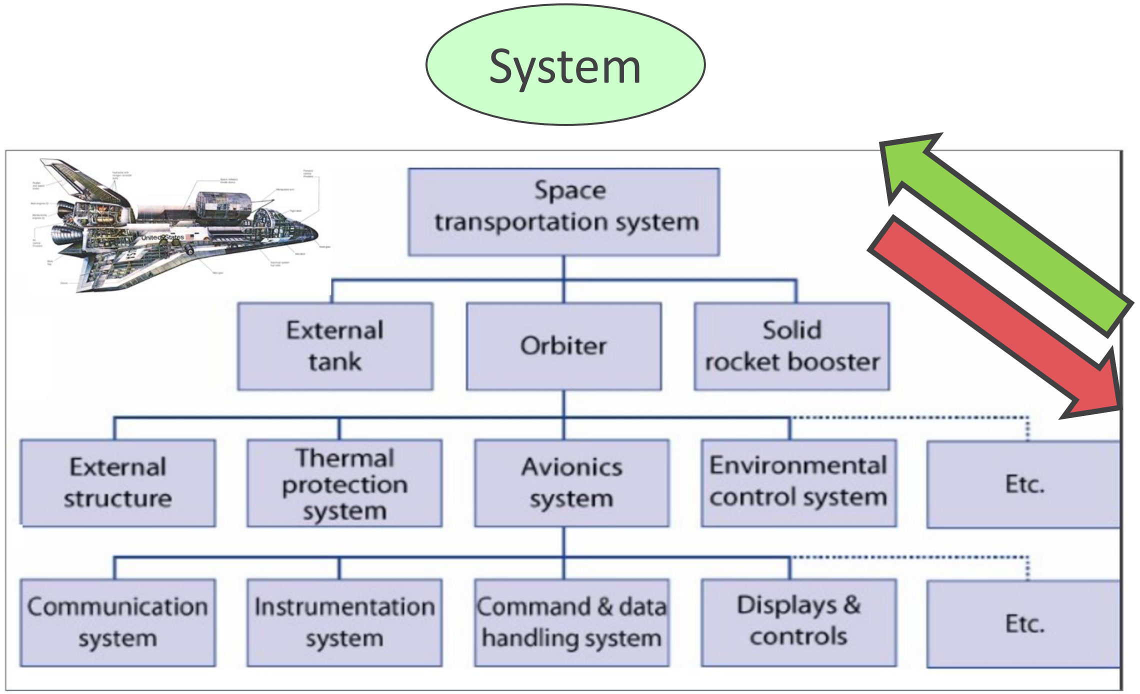


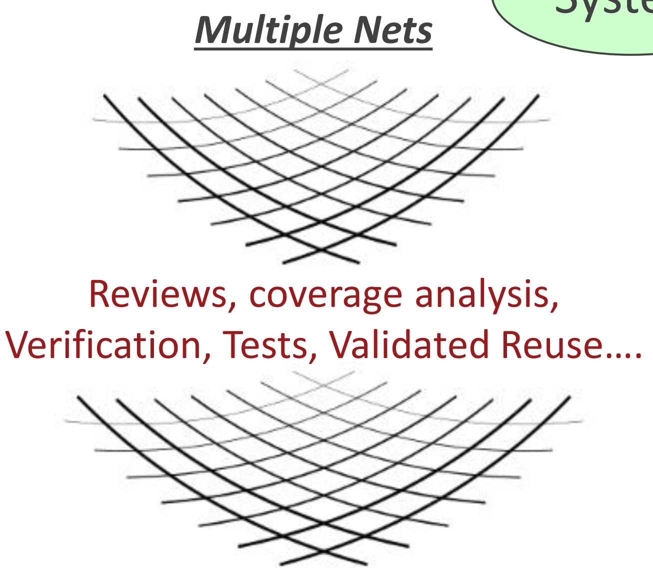
Image Credit: Towards Data Science

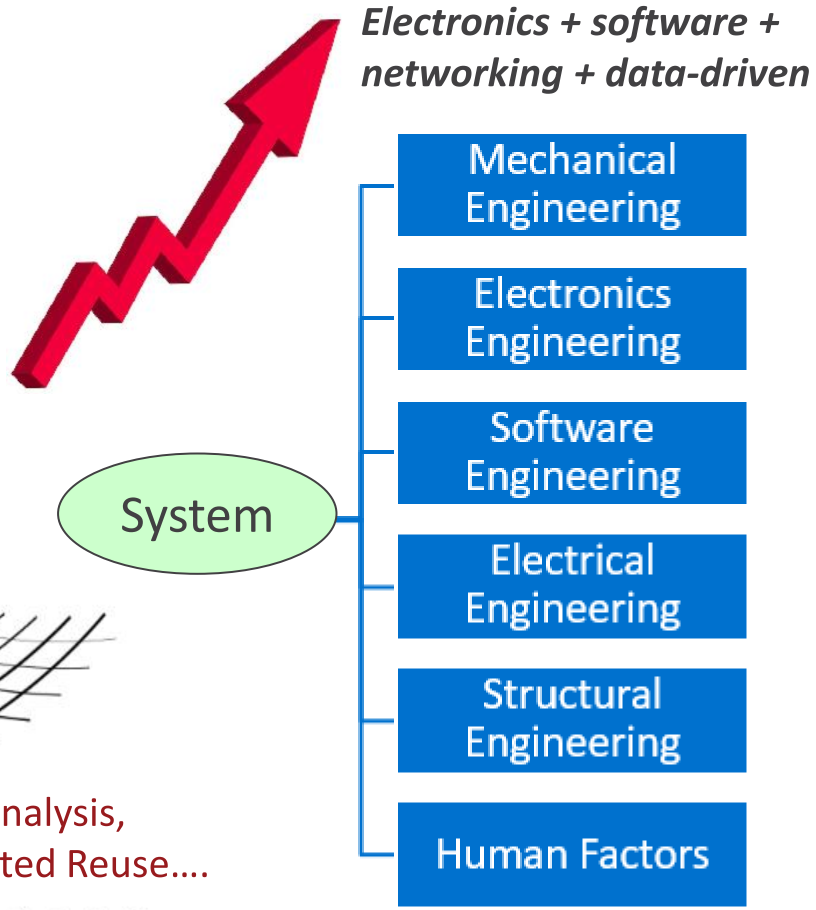
Development Lifecycle



Challenges in complex engineered systems





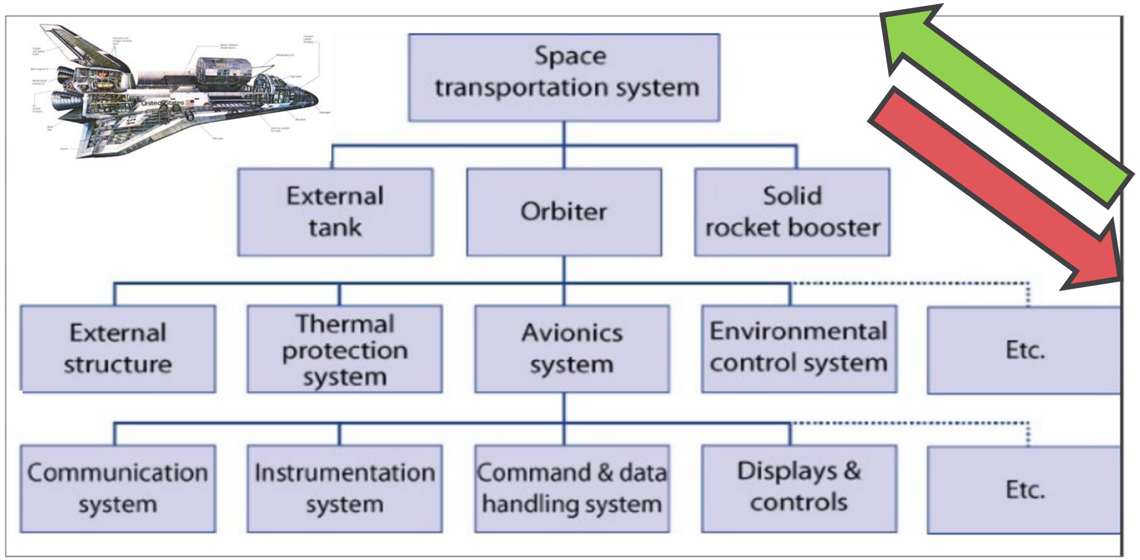


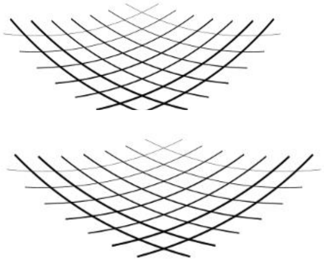
But many new failure areas are appearing …beyond just those driven by component failure

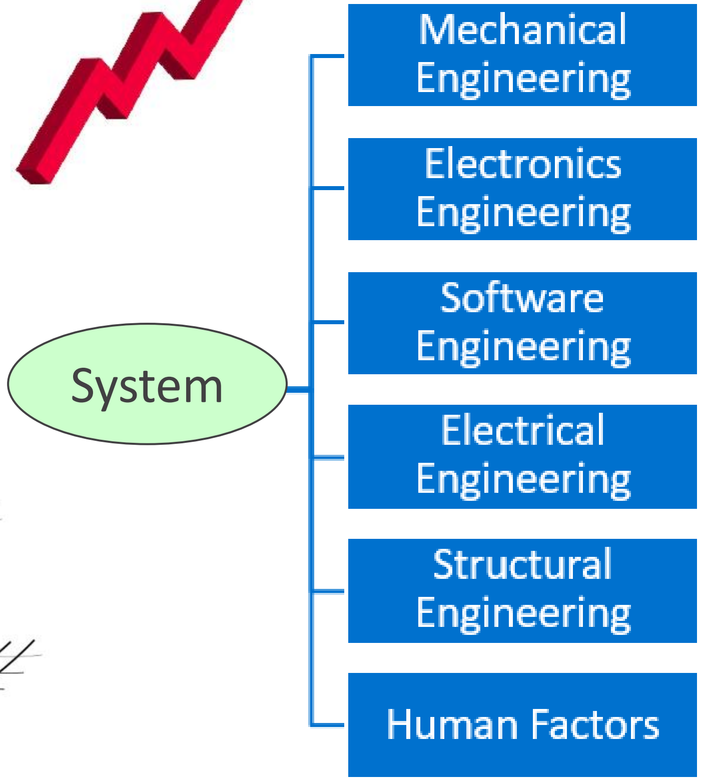
Failure Prone Areas for complex systems

Electronics + software + networking + data-driven

System





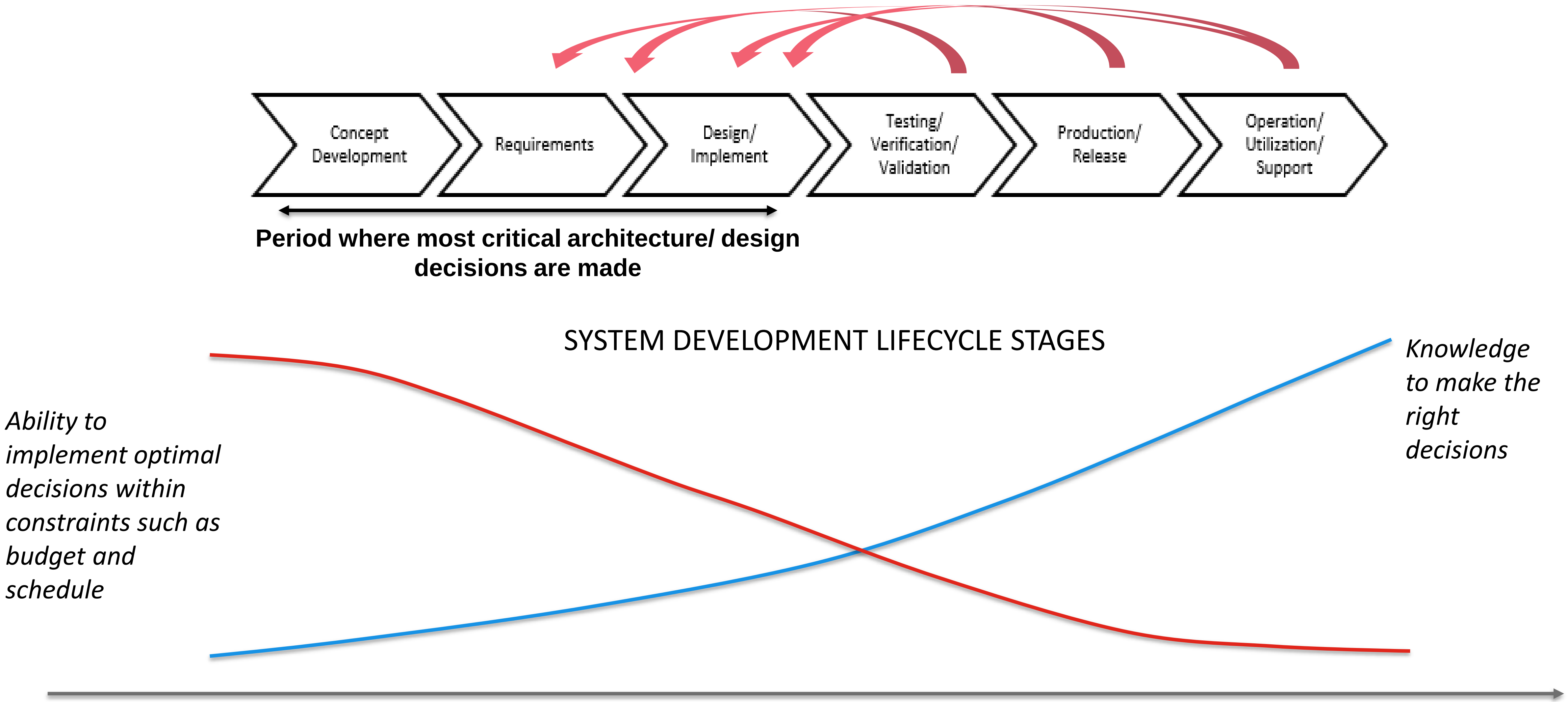


Failure prone areas – layers/ interactions that are not seen

Failure prone areas – interdisciplinary "cracks"

No components would have failed, but failure is due to the complexity in interactions & emergent behavior, as driven by the system design

Decision Scenario



Presentation Outline

Introduction → System Architecture Design → ML Models → Conclusion

Introduction
- Machine Learning
- Complex Systems

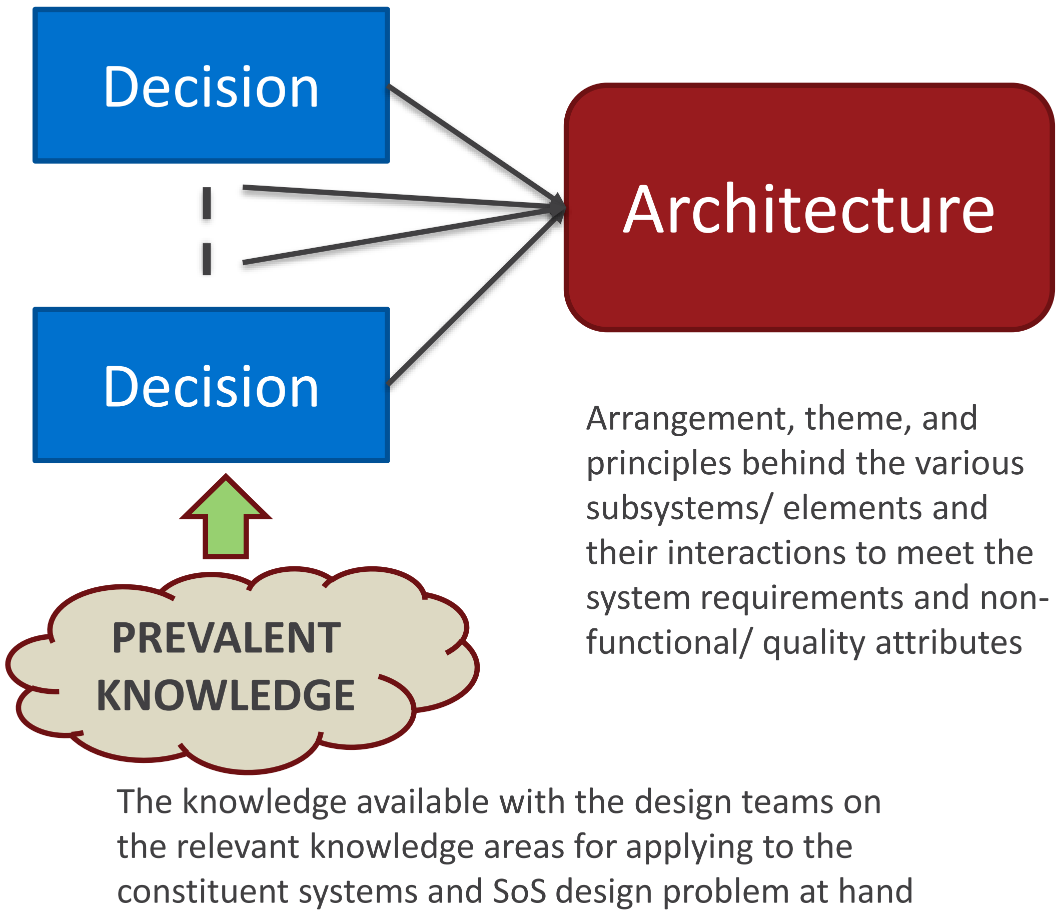
System Architecture Design
- Design Decisions
- Learning Cycles

ML Models
- LCC prediction
- Uncertainty Assessments

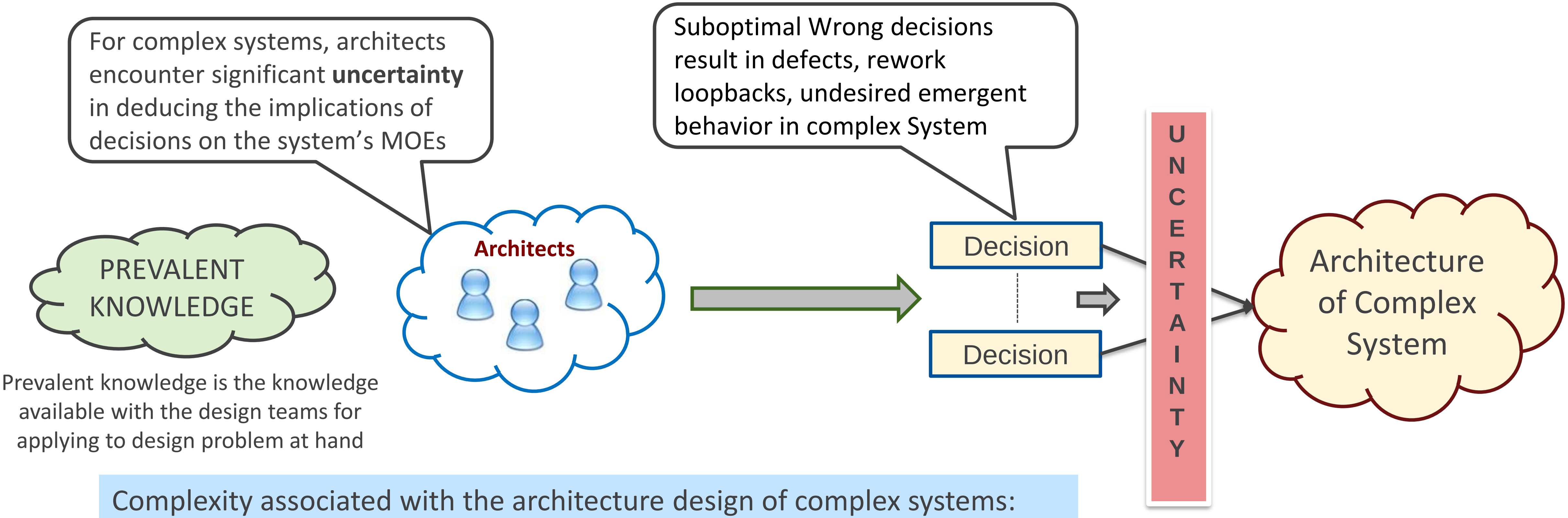
Conclusion
- Opportunities
- Conclusion

Architecting Systems

- The view of architecture, as a set of relevant decisions and corresponding decision based views & decision models, has been well established
- The architectural/ design decisions comprise the specific choices and selections made from the set of all known alternatives of arrangements, themes and principles pertaining to the architecture/ design of the system.
- Arriving at the architecture can be viewed as a string of decisions to be taken, with each decision having one or more alternatives
- Decision making techniques typically involve evaluating the alternatives in terms of how well each meet the requirements, thereby requiring tradeoffs



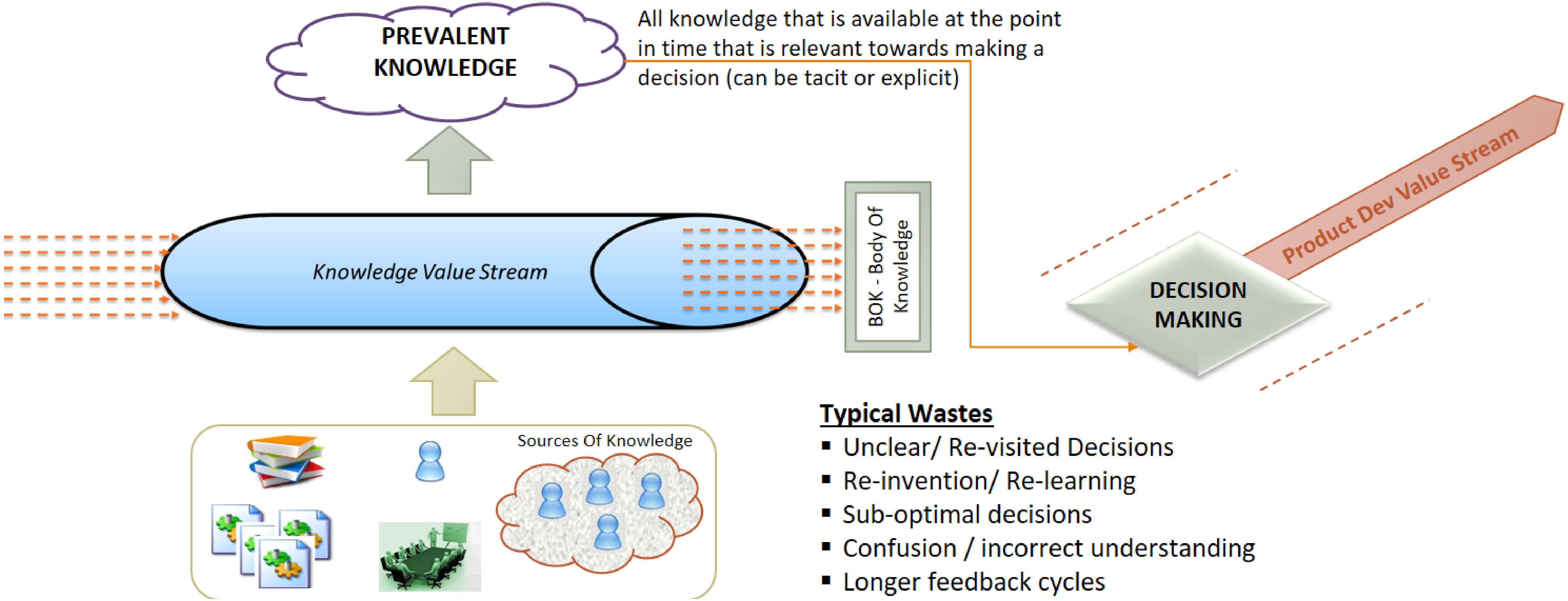
Uncertainty in Architecture Design Decisions



Complexity associated with the architecture design of complex systems:

(a) multiplicity of the number of decisions
(b) diversity of the knowledge areas pertaining to the decision
(c) Significant interdependencies and multiple implications of the decision

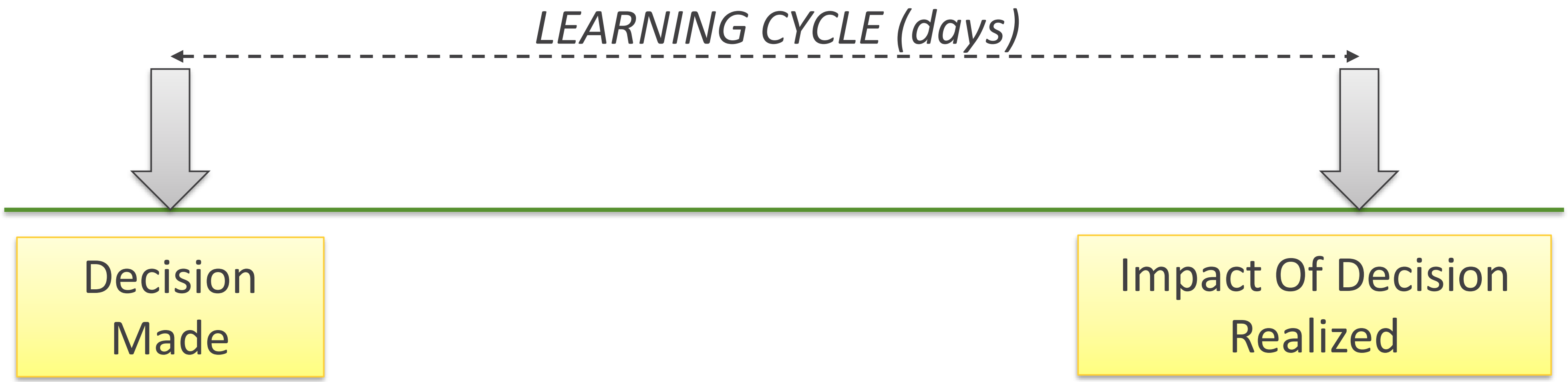
Knowledge Value Stream



Typical Wastes

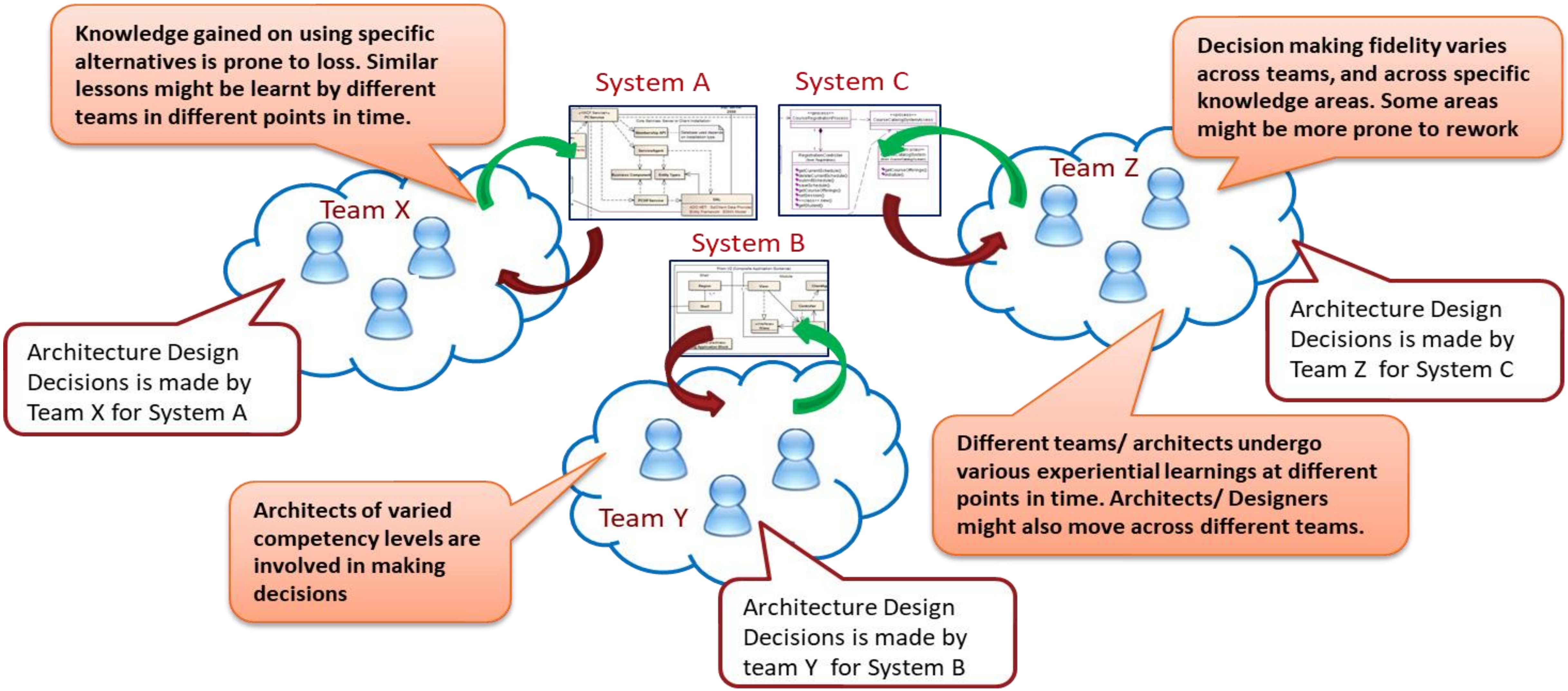
- Unclear/ Re-visited Decisions
- Re-invention/ Re-learning
- Sub-optimal decisions
- Confusion / incorrect understanding
- Longer feedback cycles

Learning Cycles



- ❑ Yes, that is the right decision
- ❑ It’s ok, though suboptimal, we can proceed
- ❑ No, we need to patch up (surgery)
- ❑ No, we need to redo/ rework (loop back)

Organizational Scenario - Decisions



Presentation Outline

Introduction → System Architecture Design → ML Models → Conclusion

Introduction

- Machine Learning
- Complex Systems

System Architecture Design

- Design Decisions
- Learning Cycles

ML Models

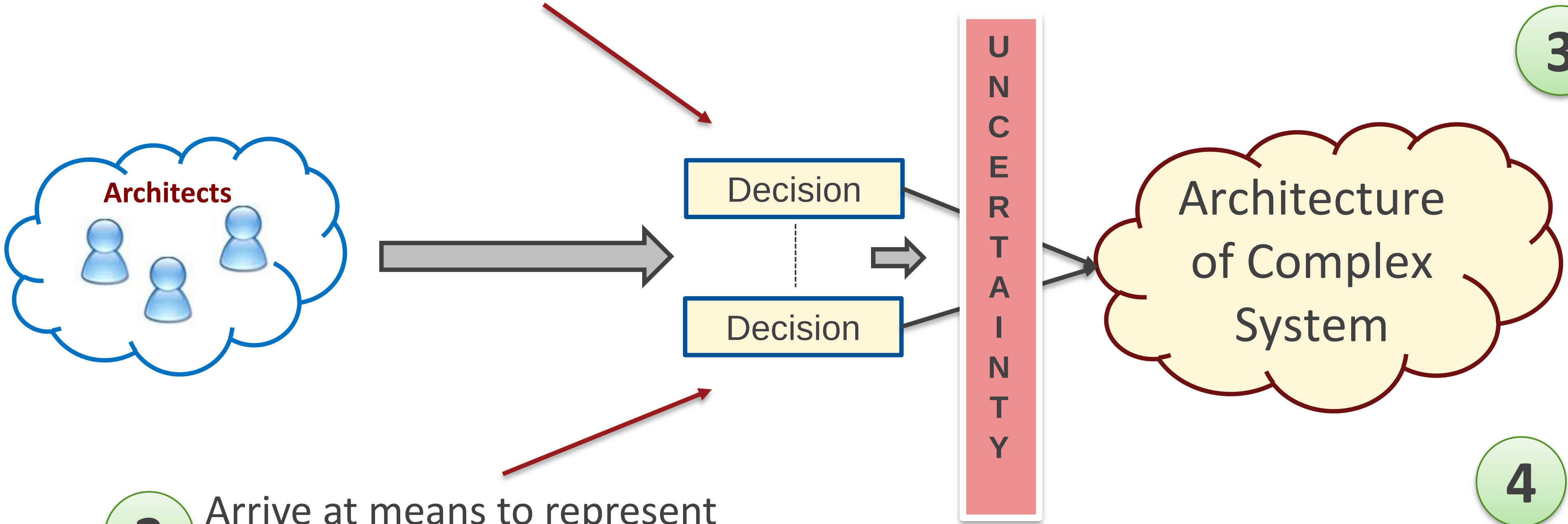
- LCC prediction
- Uncertainty Assessments

Conclusion

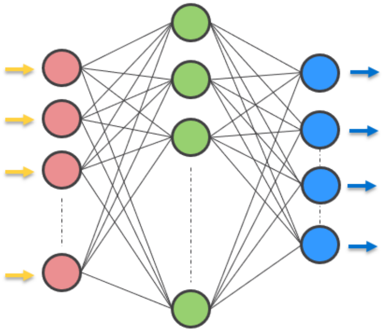
- Opportunities
- Conclusion

ML Model Approach

1 Capture the experienced architecture design decisions, and label “good” and “not good” decisions



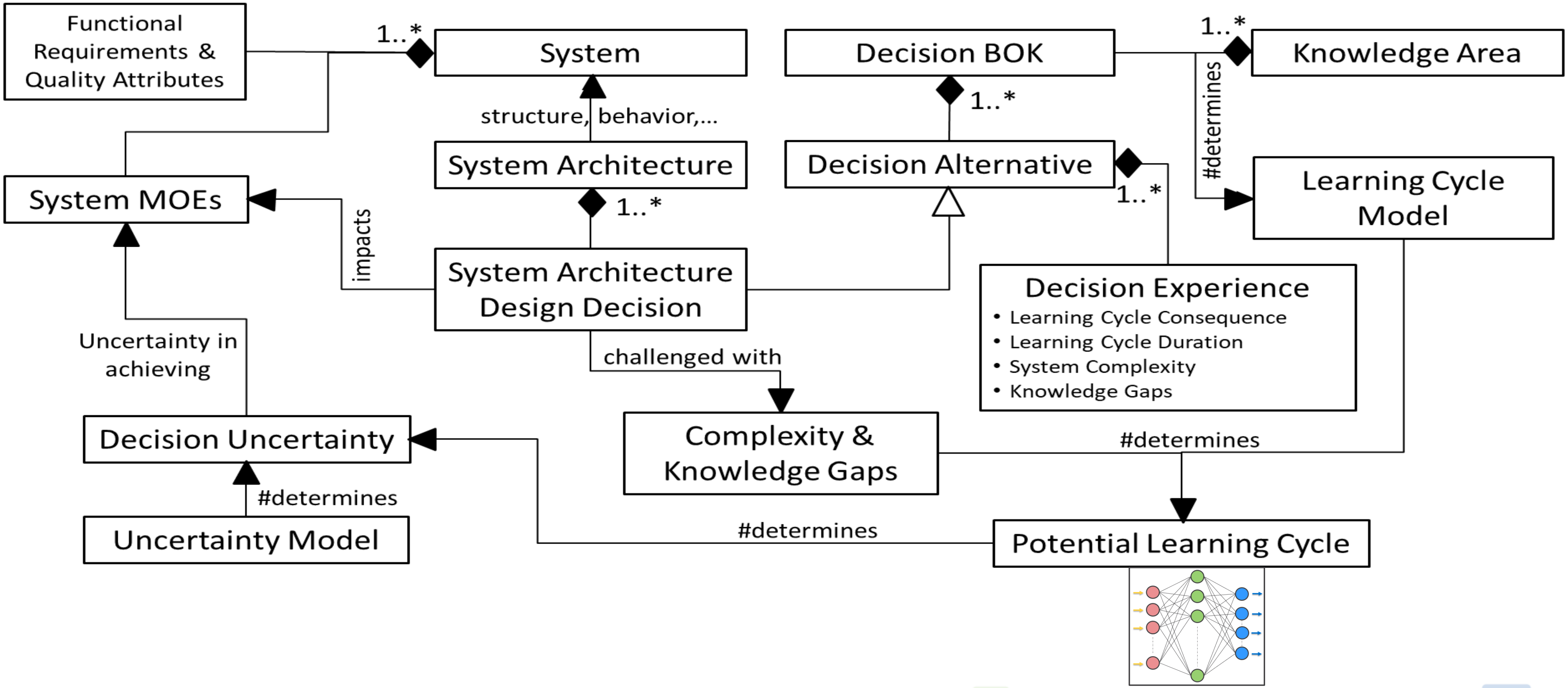
3 Train ML Models



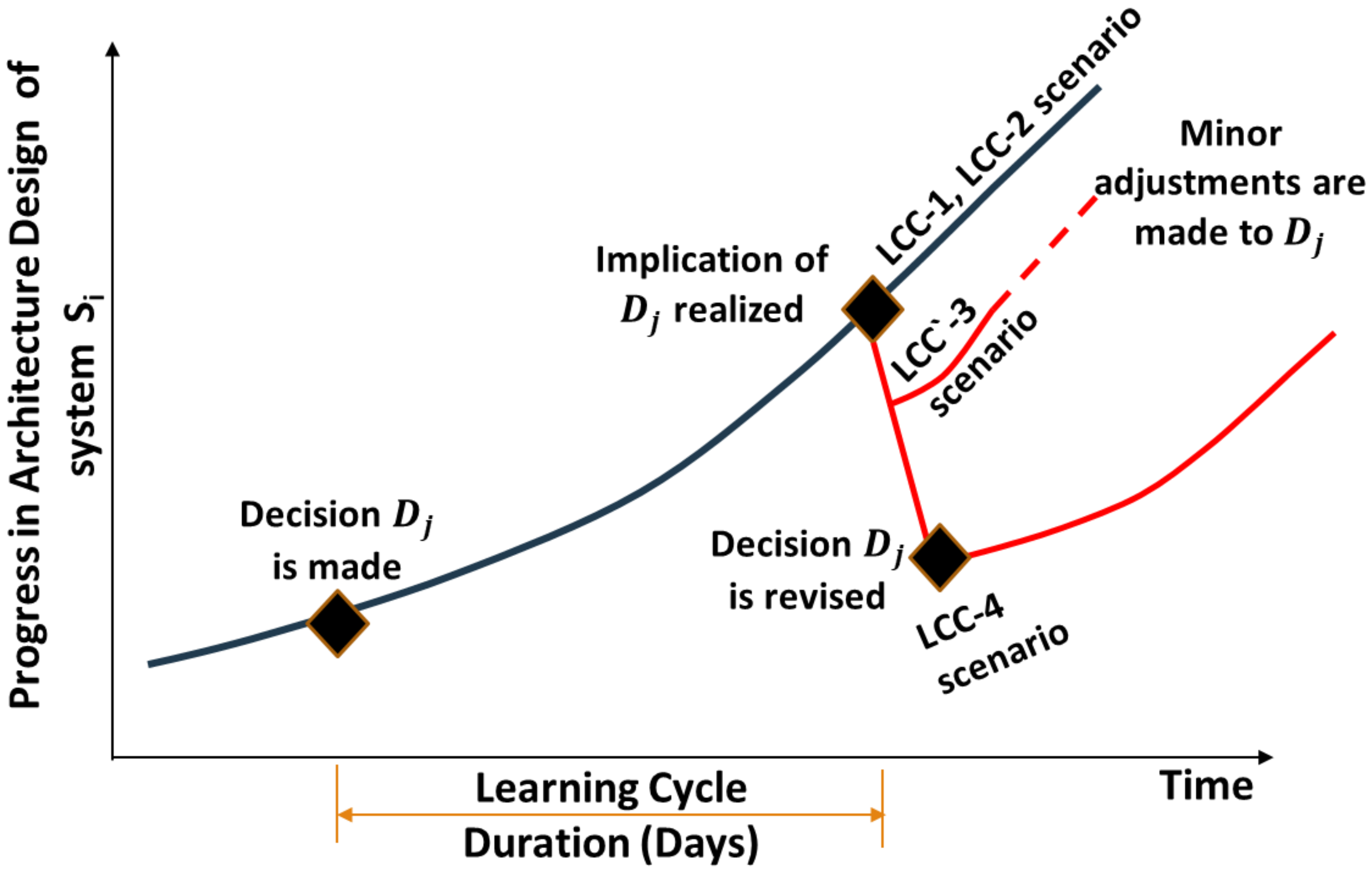
4 Predict & Assess Decision Uncertainty

2 Arrive at means to represent decisions – in terms of attributes

Domain Model



Learning Cycle Consequences



LCC-1	The decision is the optimal decision, and does not inhibit any of the requirements or expected behaviors of the system
LCC-2	The decision is not the optimal one, but nevertheless, can be "lived with", i.e. does not impact any critical requirements or behaviors of the system
LCC-3	The decision is not optimal, and it might require some amount of rework, minor correction or "surgery"
LCC-4	The decision needs to be significantly reworked, requiring a loop-back to the point where the decision was taken. In extreme cases, the budget or resources required for the rework might be far in excess of what is available or allowable for development

Learning Cycle Consequence

ML Model – Learning & Prediction Phase

Feedback the architecture
design decision experiences

Learning Phase

- Codify architecture design decision experiences
- Analyze Learning Cycle Space
- Train the Learning Cycle Model [machine learning]
- Formulate Uncertainty Model

Prediction Phase

- For new architecture design decision, codify the alternatives considered
- Use Learning Cycle Model to predict potential learning cycles
- Use Uncertainty Model to assess uncertainty and factor in decision for choosing a specific alternative
- Monitor & reassess decision uncertainty with time

Provide Learning Cycle Model
& Uncertainty Model

Codification of Decision Experience

Decision (#2)
OS Middleware

Alternative (#1)
<<h1=2, h2=3, h3=1, h4=2>>

Alternative (#2)
<<h1=1, h2=2, h3=1, h4=2>>

Alternative #1 was used in a system with complexity measure 4. There was 8 knowledge gaps that existed when the decision was made. The decision was found to be optimal (LCC-1) and this was realized after a long duration

Decision #2: OS Middleware : Alternatives - Attributes

h1 Scheduling: co-operative = 1, rate monotonic=2,…

h2: NVM writing: periodic = 2. on demand = 3,…

h3: Buffer: fixed = 1, variable=2, ring buffer=3,…

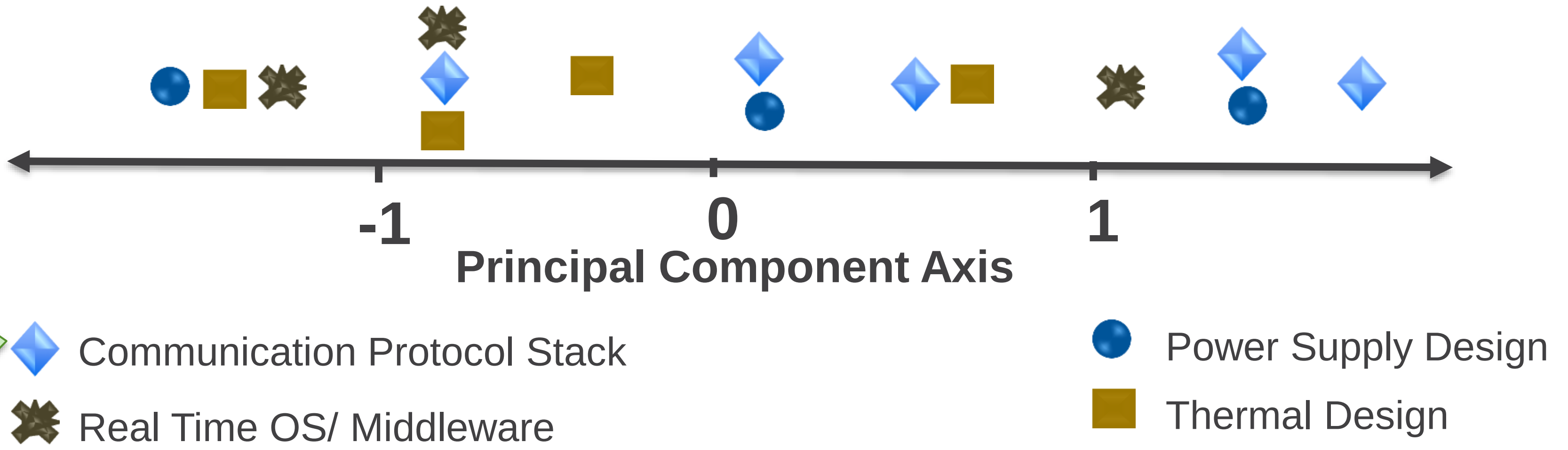
h4: thread life management: ….

Codification of Learning Cycle Experiences pertaining to Architecture Design Decisions

Decision ID	h1	h2	h3	h4	Sys Comp	K-Gaps	LCC	LCD
2	2	3	1	2	4	8	1	3
1	1	2	2	1	3	12	3	2
3	2	3	1	2	1	5	3	2
1	1	2	2	1	2	6	2	1
2	1	2	1	2	3	11	4	3

PCA of Decisions

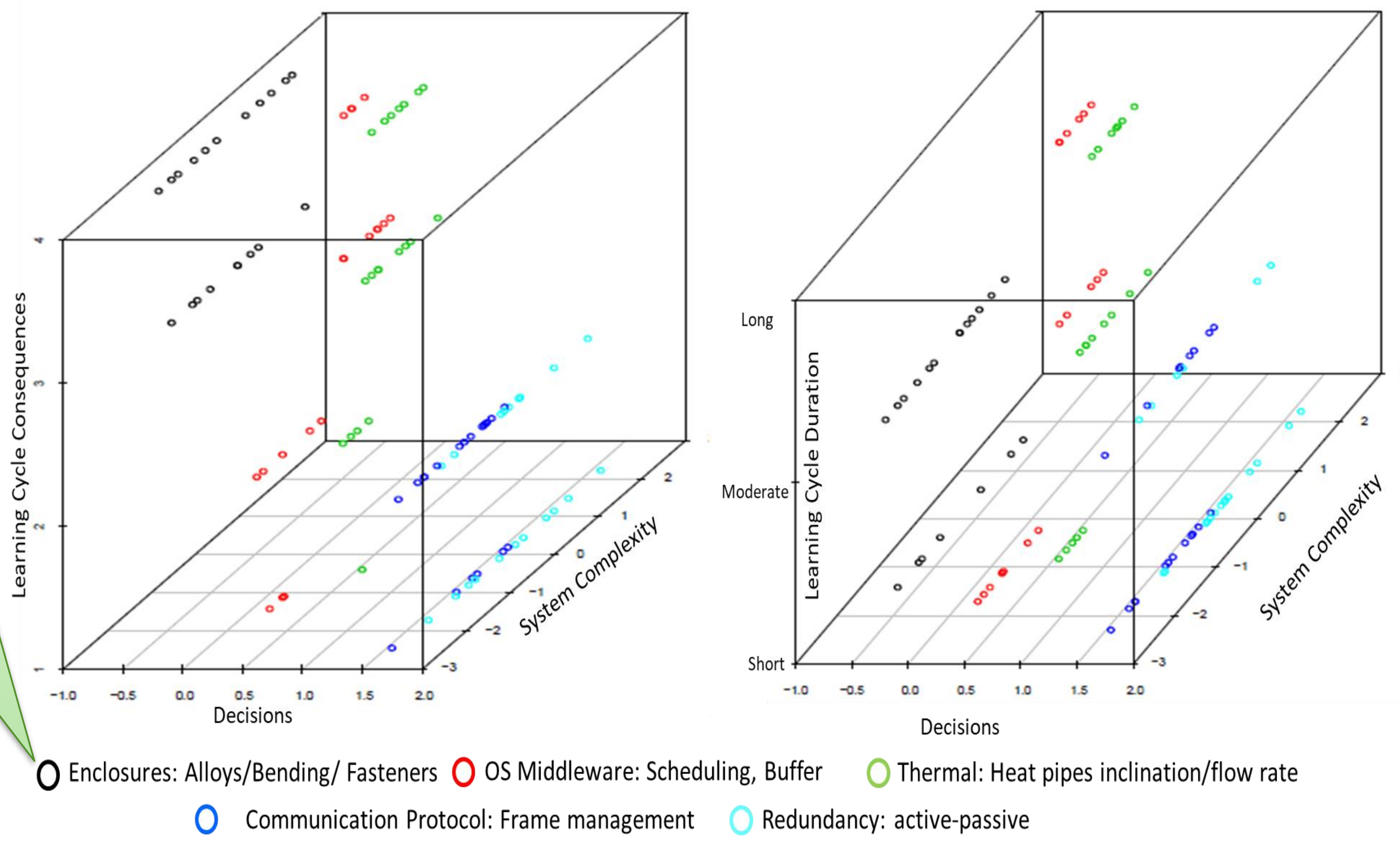
All the decisions taken pertaining to Communication Protocol Stack manifest as five distinct clusters (-0.81, 0.14, 0.54, 1.57, 1.82 on the Principal Component Axis),



- ❑ PCA adopted to reduce the dimensionality of the multi-dimensional decision space
- ❑ Clustering of the decision points along the reduced single dimension of the decision space enables easy visualization and analysis in terms of distinct clusters and characteristics

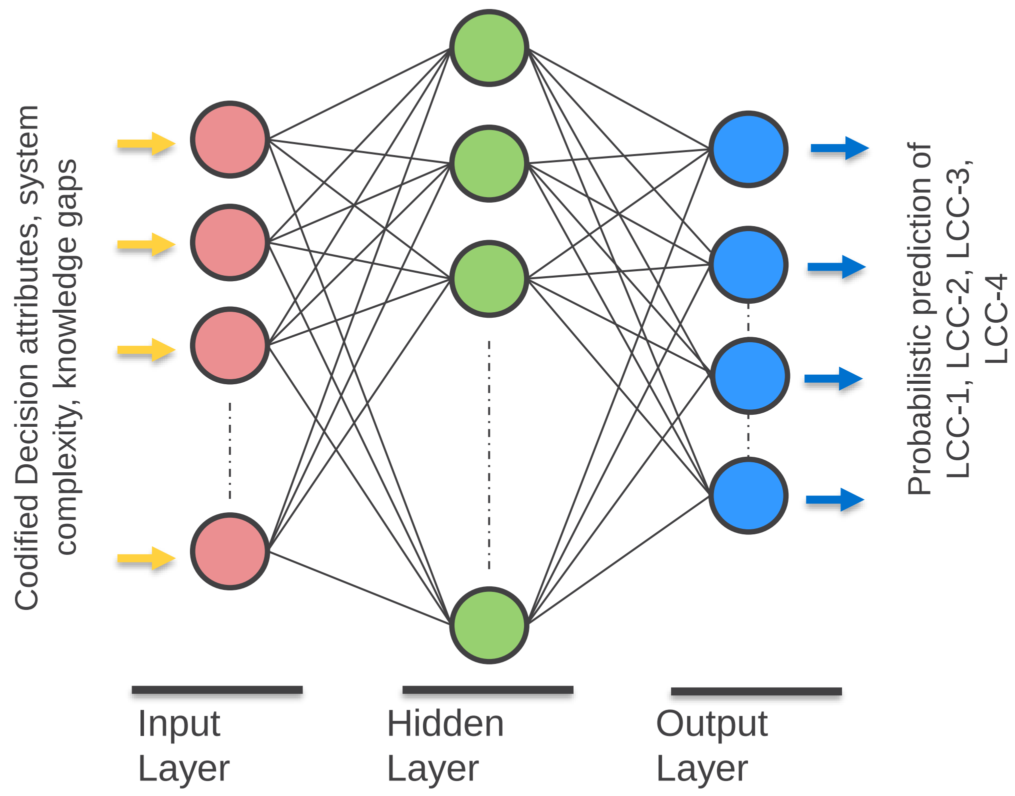
Learning Cycle Space

Decisions pertaining to Enclosures: Alloys/ Bending have mostly experienced LCC-3 and LCC-4, across a wide range of varied system complexity and knowledge gaps (low and high).



- Provides insights into the architecture design decision-making experience
- Represents the experiential knowledge pertaining to the various architecture design decisions taken for different systems

ML Model



Activation Function

$$sigmoid(z) = g(z) = \frac{1}{1 + e^{-z}}$$

$$g'(z) = \frac{d}{dz} g(z) = g(z)(1-g(z))$$

Input Layer: $a^{(1)} = x$

Hidden Layer: $z^{(2)} = \Theta^{(1)} a^{(1)}$

$a^{(2)} = g(z^{(2)})$

Output Layer: $z^{(3)} = \Theta^{(2)} a^{(2)}$

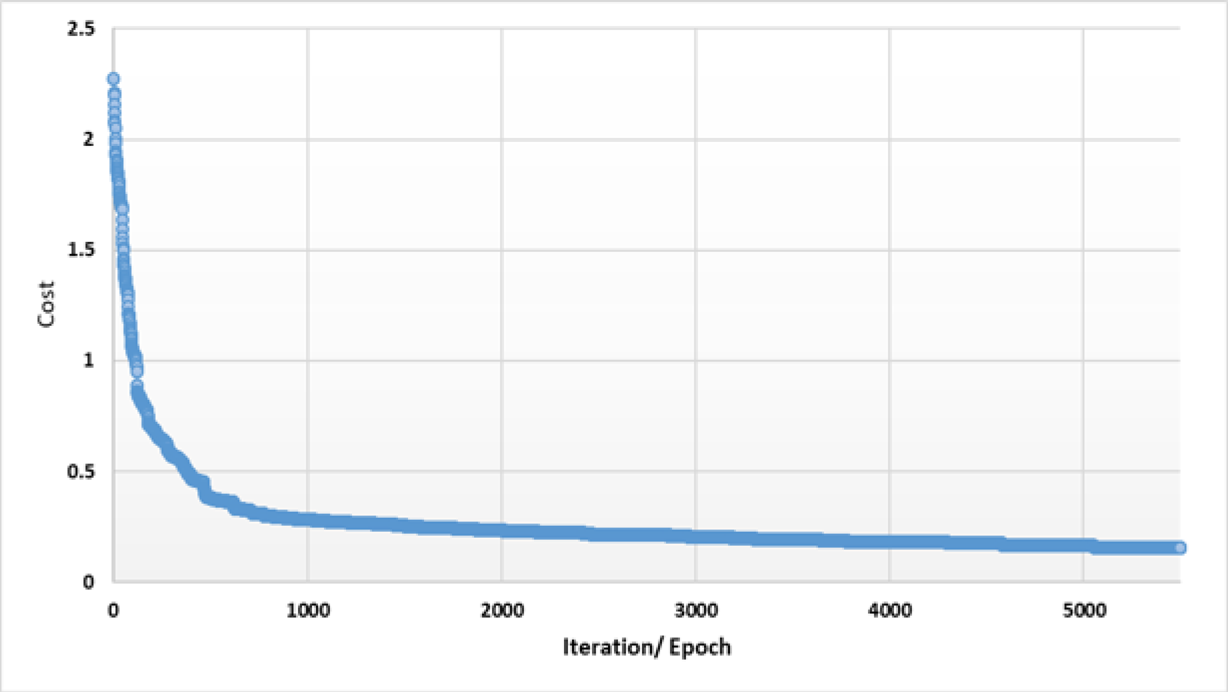
$a^{(3)} = g(z^{(3)}) = h_\theta(x)$

- Machine Learning methods adopted to train the Learning Cycle Model
- The past decision experiences form the Training Set
- The Learning Cycle Model learns about the learning cycles experienced, pertaining to various decisions taken over the period of development and evolution of various systems in the organization

ML Model Performance

PREDICTION					
LCC-1	27.4%	0.8%	0.0%	0.0%	97.3% 2.7%
LCC-2	0.4%	30.5%	0.0%	0.0%	98.8% 1.2%
LCC-3	0%	0.8%	15.8%	0.4%	93.2% 6.8%
LCC-4	0%	0.4%	0.4%	23.2%	96.8% 3.2%
	98.6% 1.4%	94.0% 6.0%	97.6% 2.4%	98.4% 1.6%	96.9% 3.1%
	LCC-1	LCC-2	LCC-3	LCC-4	

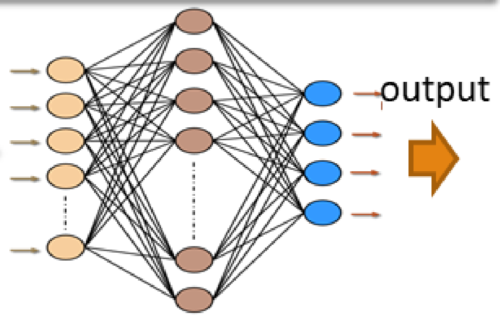
TARGET



Validation

Decision Experience pertaining to architecture design decisions							
DecisionID	**h1**	**h2**	**h3**	**h4**	**SysComp**	**K-Gaps**	**LCC**
2	2	1	1	2	3	11	4
1	3	2	1	1	1	5	1
3	3	1	2	2	3	6	4
1	1	2	2	1	3	9	3
2	2	3	1	2	4	9	2

input

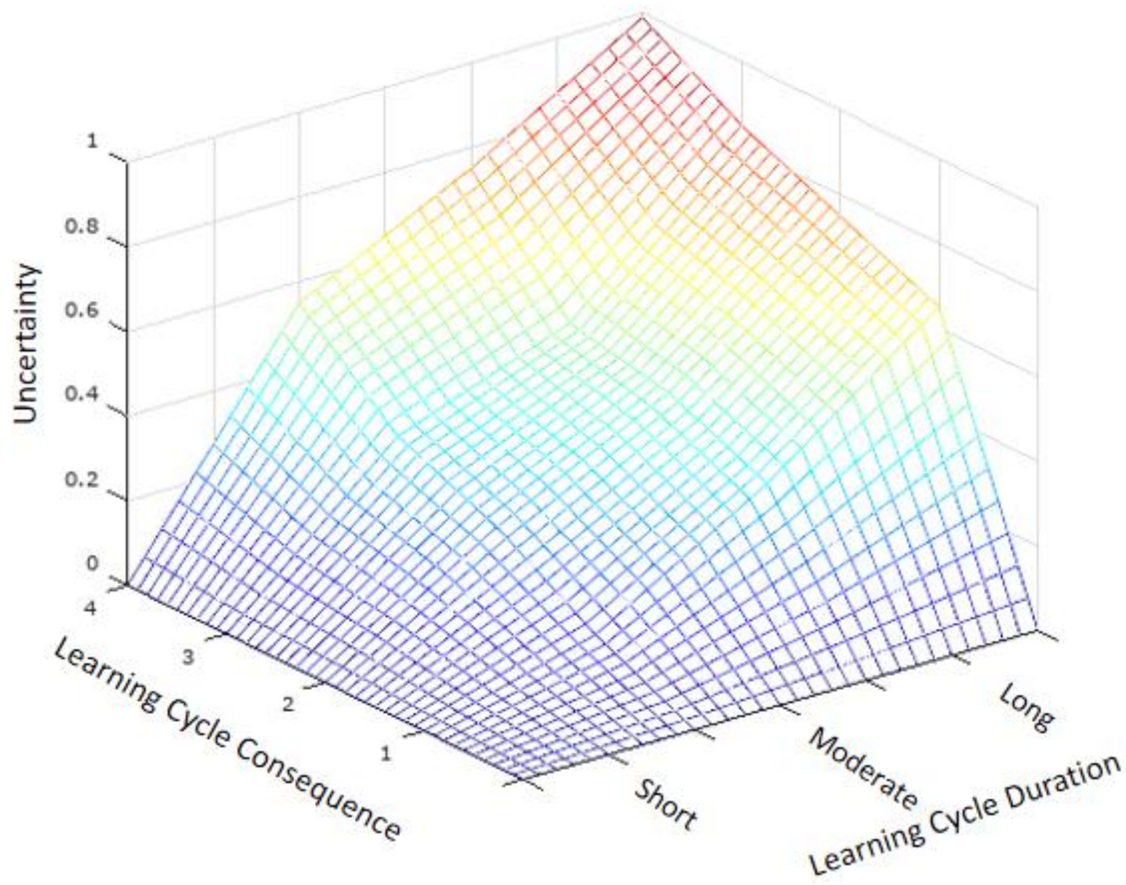


Learning Cycle Model

Probabilistic predictions of Learning Cycle Consequences			
LCC-1	**LCC-2**	**LCC-3**	**LCC-4**
0.0016817217	0.0061270562	0.0000796866	0.9549523432
0.9999969555	2.3560401E-11	2.2714304E-06	0.0000422213
0.0000170234	1.2163202E-11	1.2490132E-10	0.9997610606
0.0004465759	0.0008018442	0.9963371391	0.0072283499
0.0342013134	0.9708418369	6.1549942E-07	0.0001666969

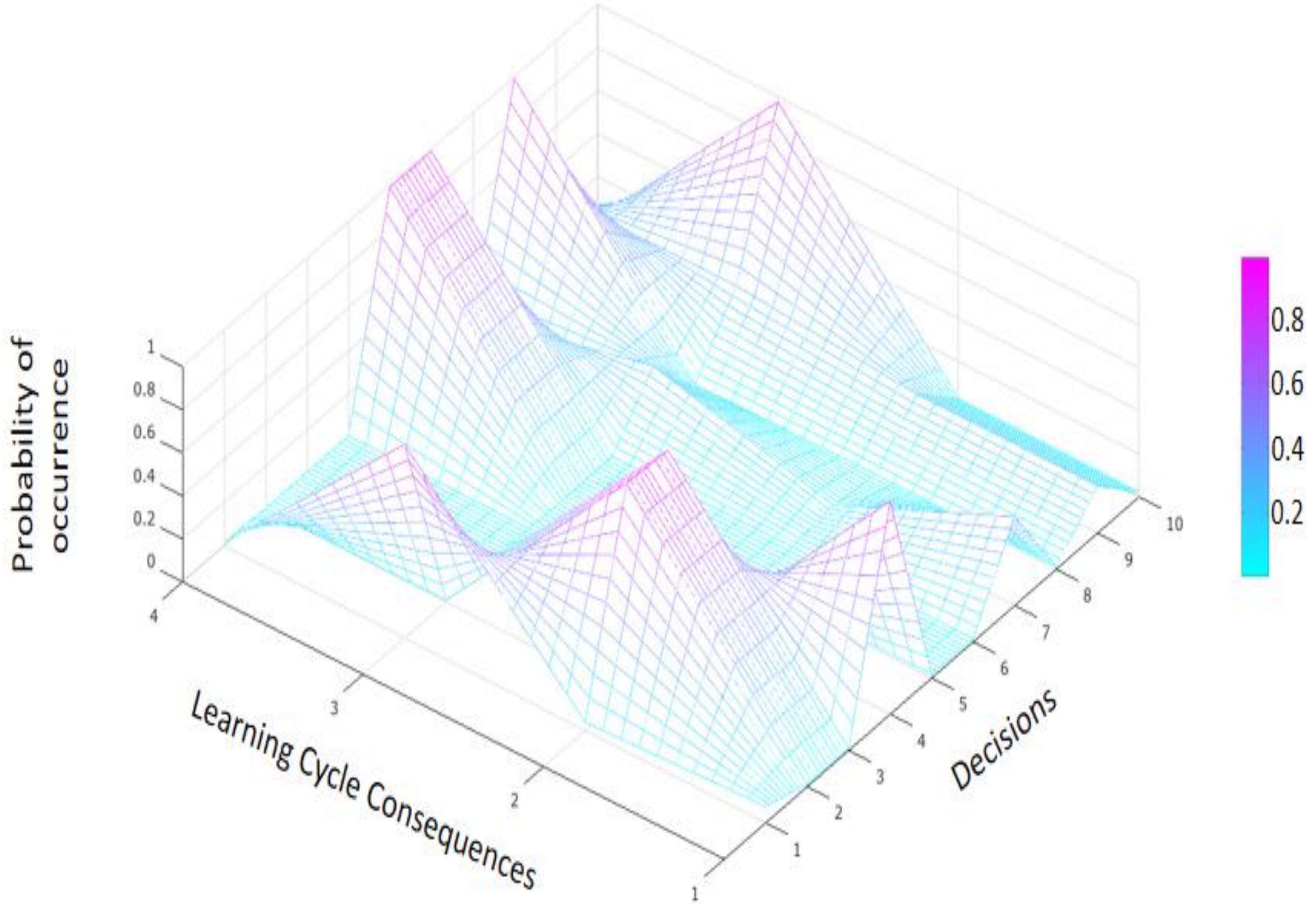
- Learning Cycle Consequence prediction: Validation is in terms of the highest probability prediction for the specific learning cycle consequence, in tandem with the actual experience
- Similar approach is done to predict learning cycle duration, categorized as short-moderate-long

Uncertainty Model



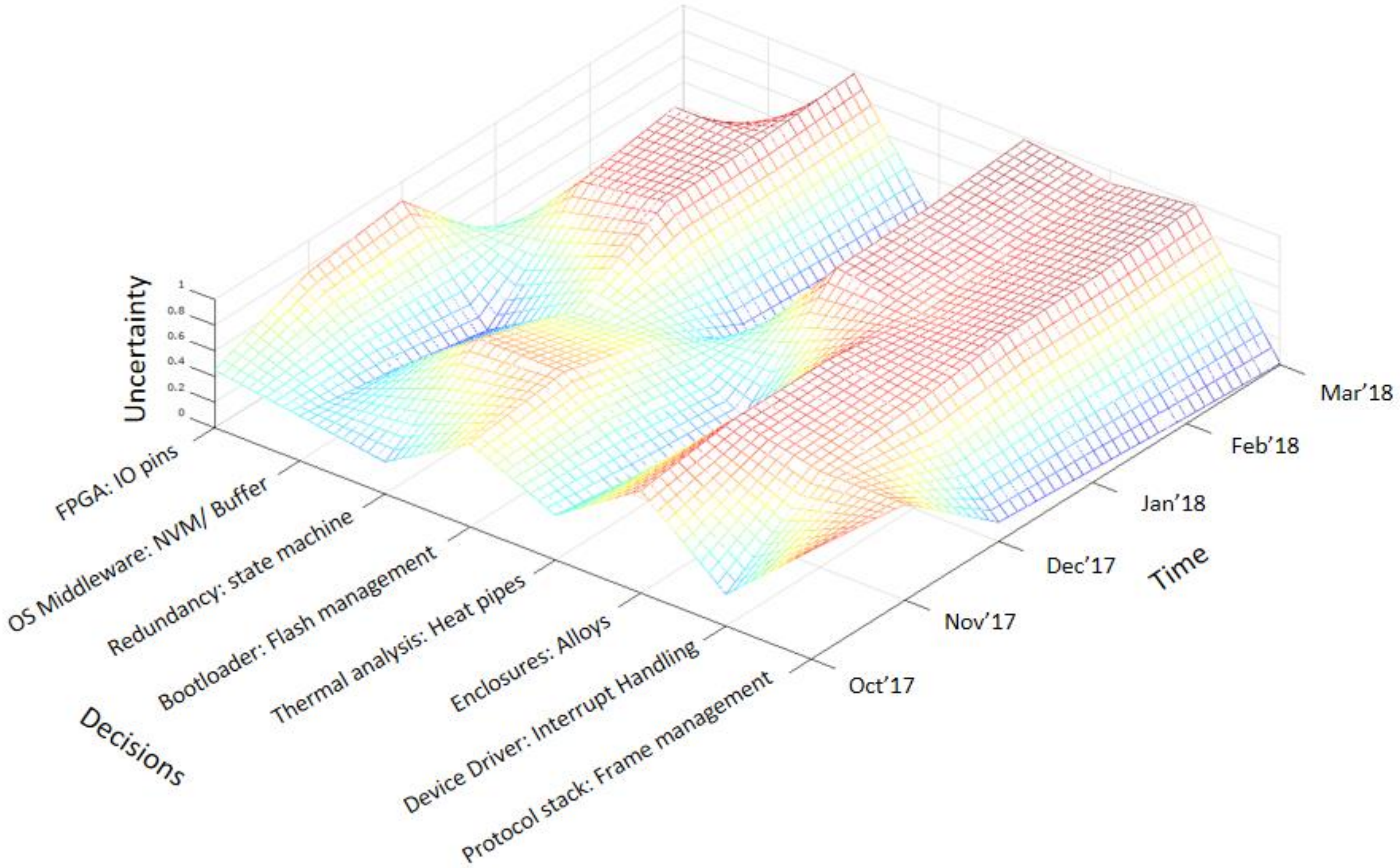
- Built as a surface that is formulated based on the Learning Cycle Consequences and Learning Cycle Duration
- Higher uncertainty (maxima point on the uncertainty surface) is associated for LCC- 4 with long Learning Cycle Duration
- The lowest uncertainty (minima point on the uncertainty surface) is associated with LCC-1 with short Learning Cycle Duration
- A team or organization to appropriately calibrate the uncertainty surface based on factors such as knowledge areas, culture, performance of the teams and organizational stage gate processes

Prediction of LCC



- For architecting the system, various decisions and corresponding feasible alternatives are enlisted, along with the corresponding knowledge gaps
- The decision-making process requires the architects to analyze the set of possible alternatives pertaining to each decision
- The ML Model is used to predict the potential learning cycles for the shortlisted alternatives

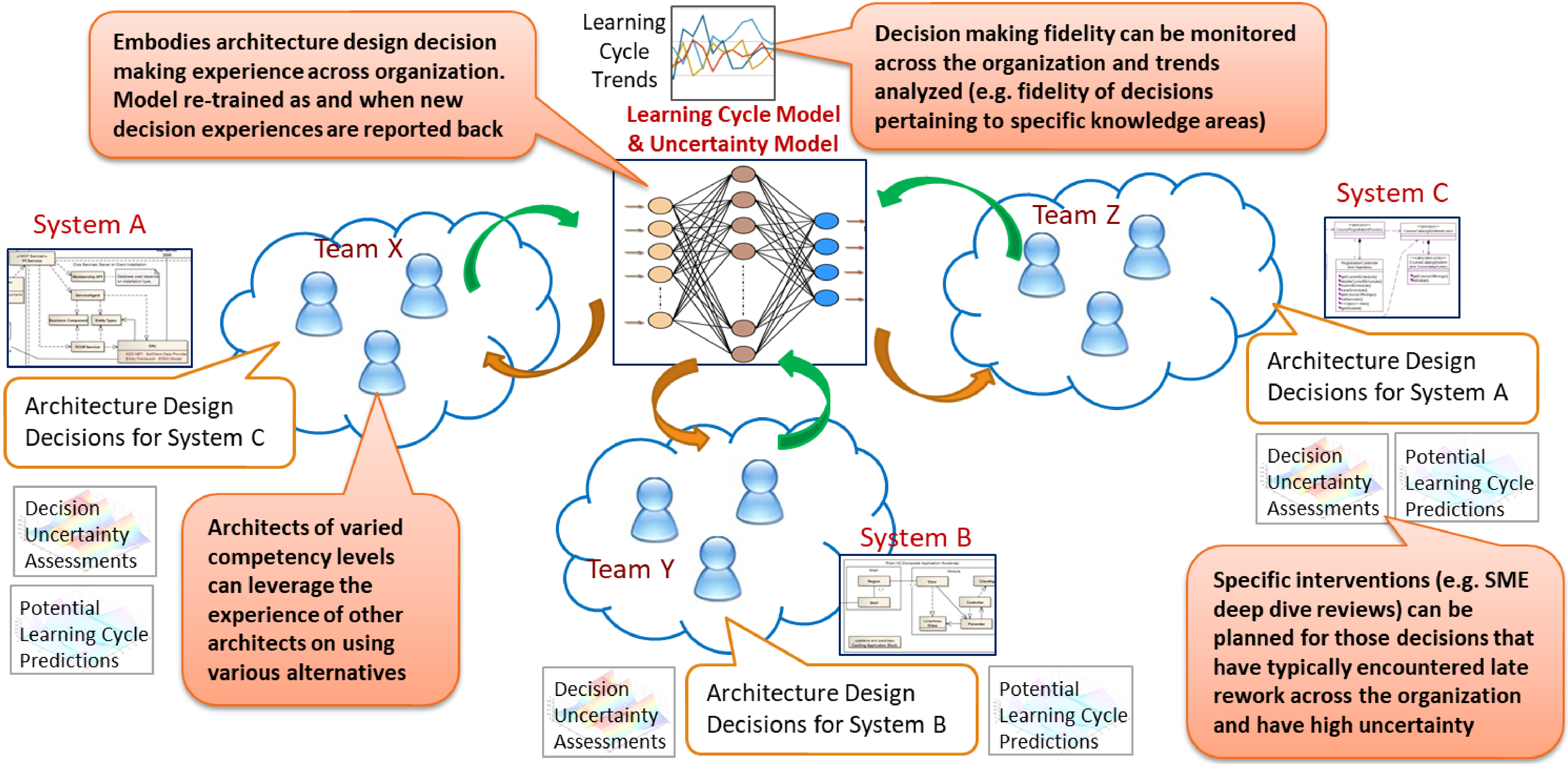
Uncertainty Assessments



- Based on the learning cycle consequence and duration probabilities for the selected alternative, the Uncertainty Model formulated in [L3] is used for assessment of the corresponding uncertainty

- As the development progresses, the uncertainty is to be re-assessed since there will be changes in the knowledge gaps associated with the decision

Benefits



Embodies architecture design decision making experience across organization. Model re-trained as and when new decision experiences are reported back
Learning Cycle Trends
Decision making fidelity can be monitored across the organization and trends analyzed (e.g. fidelity of decisions pertaining to specific knowledge areas)
Learning Cycle Model & Uncertainty Model
System A
Team X
Team Z
System C
Architecture Design Decisions for System C
Architecture Design Decisions for System A
Decision Uncertainty Assessments
Potential Learning Cycle Predictions
Architects of varied competency levels can leverage the experience of other architects on using various alternatives
Team Y
System B
Decision Uncertainty Assessments
Architecture Design Decisions for System B
Potential Learning Cycle Predictions
Decision Uncertainty Assessments
Potential Learning Cycle Predictions
Specific interventions (e.g. SME deep dive reviews) can be planned for those decisions that have typically encountered late rework across the organization and have high uncertainty

Presentation Outline

Introduction

- Machine Learning
- Complex Systems

System Architecture Design

- Design Decisions
- Learning Cycles

ML Models

- LCC prediction
- Uncertainty Assessments

Conclusion

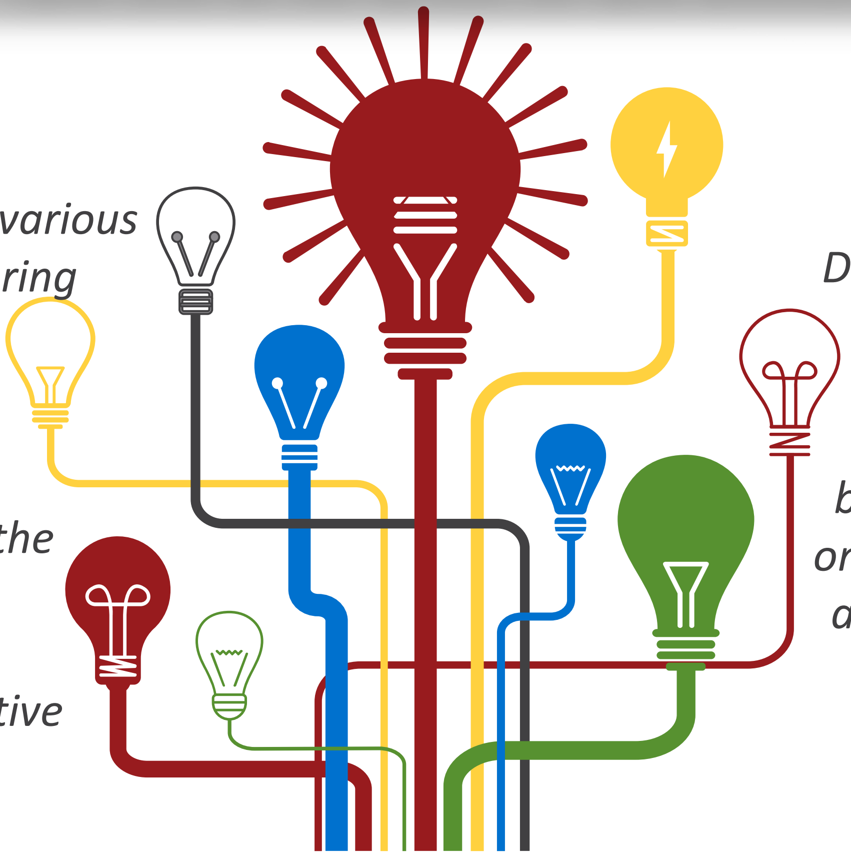
- Opportunities
- Conclusion

Opportunities...

Quality of the engineered system
Cycle time for the various life cycle activities

Assist in arriving at various corner test cases during verification

Advise system architects on the various architecture design decisions options based on intelligence built from collective prior experience of decisions taken in earlier systems.



Digital/synthetic environments (e.g., digital twins) being leveraged to understand various lifecycle operation scenarios and providing better insights to systems engineers on understanding the implications of architecture design decisions on the engineered systems

Publications

https://doi.org/10.1002/sys.21517

SYSTEMS ENGINEERING
The Journal of The International Council on Systems Engineering
INCOSE

Volume 22, Issue 6
SOSE Special Issue
November 2019
Pages 538-560

REGULAR PAPER

Decision learning framework for architecture design decisions of complex systems and system-of-systems

Ramakrishnan Raman, Meenakshi D'Souza

First published: 07 November 2019 | https://doi.org/10.1002/sys.21517 |

https://onlinelibrary.wiley.com/doi/toc/10.1002/(ISSN)1520-6858.Best-Papers-of-2019

Best Papers of 2019

Virtual Issues | First published: **14 March 2021** | Last updated: **16 March 2021**

Each year, the International Council on Systems Engineering (INCOSE) selects a Best Paper award from among the papers published the previous year in Systems Engineering. The Editorial Board considers the papers in this virtual issue to be among the best from those published in 2019. We hope this virtual issue will provide some insight into the latest research and developments in systems engineering.

Export Citation(s) Download PDF(s)

SYSTEMS ENGINEERING
The Journal of The International Council on Systems Engineering
INCOSE

Table of Contents

doi: 10.1109/TEMSCON.2017.7998412

2017 IEEE Technology & Engineering Management Conference (TEMSCON)

Knowledge value stream framework for architecting complex products

Ramakrishnan Raman ; Meenakshi D'Souza

Conclusions

- *Exponentially increasing complexity of systems, exacerbated by new technologies*

- *Increasingly difficult to analyze various implications of operational scenarios on the system design - Unhandled system states, Unhandled operational scenarios/ conditions, Incomplete/ wrong assumptions*

- *Opportunity to leverage AI-ML to advise on the appropriate decisions upstream: learn from the decision learning cycles experienced*

THANK YOU