26th annual INCOSE international symposium

Edinburgh, UK

July 18 - 21, 2016

Architecture as a Solution Schema for a Class of Problems

Swaminathan Natarajan, Anand Kumar

TCS Research

Prof. Kesav Nori

Indian Institute of Information Technology, Hyderabad

TATA CONSULTANCY SERVICES

Experience certainty.

Motivation

- Express our viewpoint on What is architecture?
  - Articulate our empirical understanding of architecture for pedagogical purposes
  - Distinguish architecture from point solution design
- Sharply formulate what it means to take an architectural approach to a problem: help learners build mental models
  - Current expositions of architecture clear about the outcomes of architecting, less precise about the nature of the activity

Arising from our experience in teaching course at TCS for budding architects

Articulate common understanding – no intent to propose anything new

Typical Perspectives

Fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution

- Normative restriction of design freedom
- Architecture is outside in (stakeholders, context), design is inside-out (how to meet requirements)
- Design solves a defined problem. Architecture shapes problem & solution spaces, relationships between them
- Civil architecture: shaping space. Form to achieve function, deliver stakeholder value, including aesthetics
- Goes beyond specific current problem to define general schema e.g. product lines, domain reference arch

Challenge: Align and encompasses all these

Proposed Framing

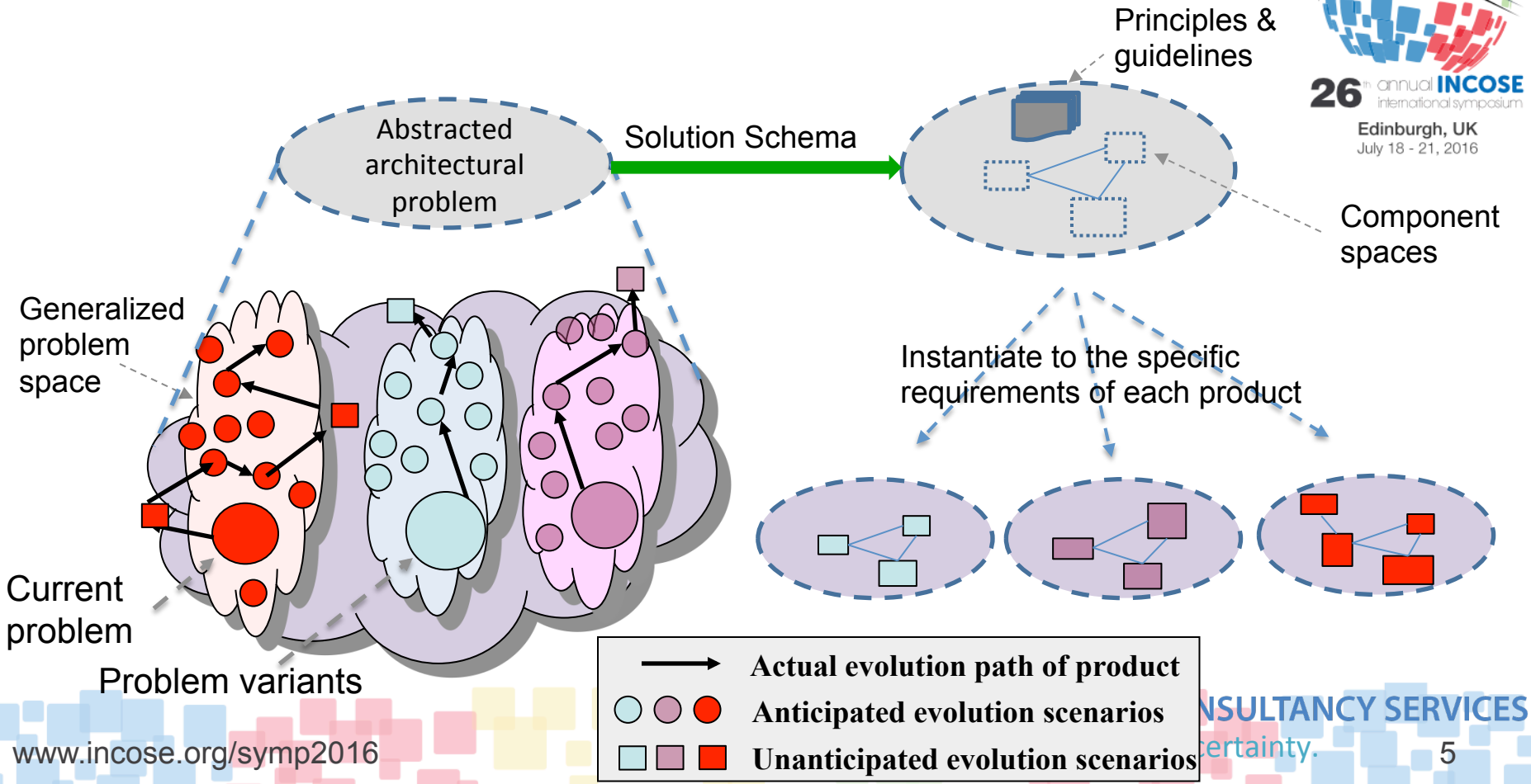
Architecture is a solution schema for a class of problems
that asserts the capacity to deliver targeted stakeholder value

Formulate architectural problem space
by abstracting from specific current problem

Generate solution schema that places normative constraints on design solution space
such that targeted stakeholder value can be achieved for any problem within the class

Express in the form of elements, their interrelationships,
desired relationships between the system and its contexts,
principles and guidelines governing their design and evolution

Conceptual View



Relationships to Practice

- Abstractions identified in problem & solution space
- Capture stakeholder needs & scenarios e.g. ATAM evaluation
- Pull back from specific requirements to categories
  - Think about quality requirements in terms of level of stringency etc
- But typically do not formulate an "architectural problem" explicitly

Should we always generalize the problem?

Generalization typically involves costs (performance, cycletime etc), limited by need to assert capacity to achieve stakeholder value

What about one-off problems with no evolution concerns ?

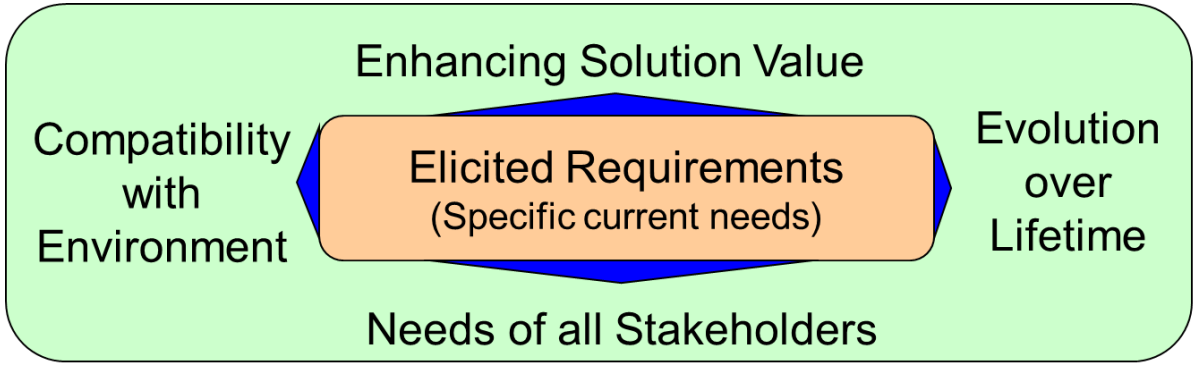
Architecture focus is on patterns that generate stakeholder value e.g. simplify construction and verification

What about systems with no explicit architecture?

"All systems have architecture, even if it is not explicit"
Architecture is the set of decisions that endure across change and variation

Formulating the Architectural Problem

1 **Scope the architectural problem space**



Usually done as part of requirements rather than architecture

Results in scenarios against which architecture is analyzed

- Also constraints on the architecture, to facilitate integration & conformance with its environments

Formulating Architectural Problem - 2

2

Pull back from specific requirements

- Abstractions that generalize functions to functionality spaces
  - E.g. “Media feeders”, “marking engine”, “finishing devices”
- Commonalities and variabilities → generalized to concern areas and constraint spaces
  - E.g. “Auditing requirements”, “compliance requirements”, “decimation policies”, “business process”
  - Details bound for each specific product at requirements and design time
- Generalize quality requirements to stringency levels, categories
  - E.g. “Near-zero downtime”, “thousands of transactions per minute”

Formulating Architectural Problem - 3

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

3 **Address holism: Desired relationships to context**

Architecting goals often framed as desired relationships between system and context

- "Comply with applicable regulations"
- "Scale as transaction load increases, maintaining response time goals"
- "Ability to develop and release new features with short cycletimes"

Hold for anticipated and unanticipated changes to system and context

↓

Need to rely on patterns and experiential knowledge

Patterns create capacity → Design establishes desired relationships → Often people processes to maintain relationships when context changes

During Architecture | During Instantiation | During Evolution

Solution Space Architecting

- Functionality spaces → component spaces
- Collaboration patterns reflect commonalities (relationships and interactions from the domain)
- Variabilities addressed by patterns that enable late binding e.g. process externalization, rules
- Patterns that address each stakeholder concern, establish & maintain relationships
- Synthesis into schema + principles; description and analysis based on concern viewpoints

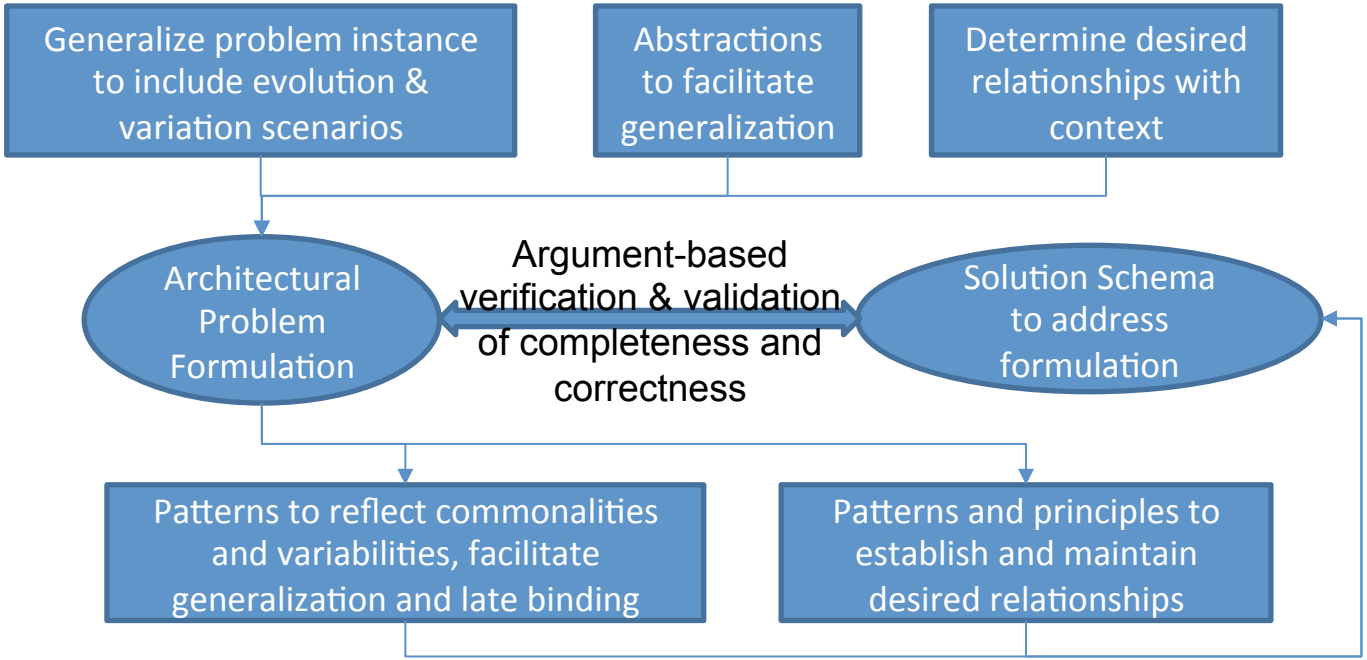
- ✓ Functional correctness arguments based on (de)compositional logic of problem domain
- ✓ Quality concerns verification based on capacity of patterns to deliver desired levels of quality
- ✓ Coverage of formulated architectural problem

complemented by

- ✓ Scenarios-based verification: can address problem requirements and scenarios while conforming to the architecture

Much of the above part of current practice (e.g. ATAM analysis). Our formulation sharpens practice by providing a basis for asserting completeness and correctness within some confidence level

Summary



- ✓ Normative restriction of design freedom
- ✓ Architecture is outside-in (stakeholders, context), Design is inside out (how to meet requirements)
- ✓ Shapes problem & solution spaces
- ✓ Fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

Thank you!

Questions?

Feedback?