TATA CONSULTANCY SERVICES

TATA

Requirements for Design of Software Processes

Kesav V. Nori

TCS Innovation Lab for Business Systems

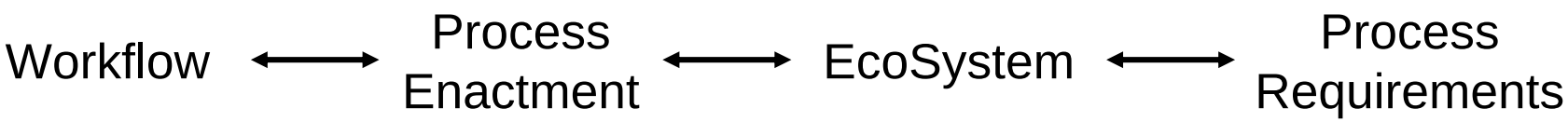
kesav.nori@tcs.com

Experience certainty. IT Services
Business Solutions
Outsourcing

- Thank You ICSP 2008 Programme Committee
- Nervous (never having been a keynote speaker at an International Conference)
- Excited at the prospect of getting feedback

Acknowledgment

- Work in progress as a part of a collaborative research programme between
  - TCS Innovation Lab – Business Systems, Hyderabad, India
  - LASER, University of Massachussetts, Amherst, Ma., USA
    - Complementing Process Technologies
    - Process Research – Outcomes, Process Design
- Collaborators
  - Prof. Leon Osterweil, UMass@Amherst
  - Nikhil Zope, Anand Kumar, Nitin Reddy and Dr. VCS Prasad, TCS ILBS, Hyderabad



Outline

- Scope of study
- Characterizing the Software Process Design Space
  - Quality of Software
  - Responsibility of Process
  - Flexibility, Adaptability and Agility of Process
  - Evolving Process and Product
  - Unity of Process and Product
- Conclusions

Scope of Study

- Industrial Processes in which people perform intellectual tasks
- Problems of definition
- Problems of scale and complexity
- From repeatable outcomes to assured outcomes

Quality of Software

Software is a Virtual Machine

- Software Processes that deliver discrete Virtual Machines
  - Parallels with other industries that define, design, produce and deliver discrete Machines
  - Need to learn from mature engineering industries about assuring Quality

- The function of a Process is to deliver a Quality “Product”
  - **Process Step – “Product” Quality Correlation**
    - “Product” Configuration
    - Reasoning and Analysis
- Non functional properties of a Process: Process Quality
  - Process Configuration
  - Statistical Control
    - Simulation

Software Quality

- Quality is to be built in, not only tested for
- Kano
  - Functional quality is presumed
    - Its absence has negative connotations
  - Non Functional Qualities
    - More is better
    - Delight – unusual – USPs

Configuration of “Product” is not fixed

- Levels of discourse
  - Problem Domain – needs, properties, Qualities
  - Domain Solution Configuration
  - IT Domain Solution
    - Architectural Configuration
    - Non Functional Quality – Engineering Configurations
    - Platform and Middleware relative Configuration
      - Construction Engineering
    - Installation Configuration
    - Release Configuration
    - Deployment Configuration

Process Step – “Product” Quality Correlation

- Each Configuration localizes Qualities discernible at its Level of Discourse
- Each Successive Process Phase is responsible for building those Qualities
- As Process progresses, Qualities accrue

- Quality is built through the process, not only tested after building the “product”

- Inadequacy of Zachman’s Framework

- ISO 9126

- ISO 15504

Responsibility of Process

Responsibility for work in Industrial Organizations

- Governance – Sustained upholding of Values of all Stakeholders
- Management – Delivering Positive Value outcomes to all Stakeholders
  - "Product" and Process Qualities are its basis
- Administration – Facilitating Work in all Processes
- Knowledge Work – the "Shop Floor" for the New Economy
  - Have the wisdom to apply knowledge to build quality in the outcome
    - "Product" Qualities
  - Trained to be "productive"
    - Process Qualities
  - "Offerings"
  - Organization
  - Business Models
  - Psychology of People @ Work

Reasoning about process responsibility

- Governance – Organization Chart
- Management – Organization Chart
- Administration – Process Chart
- Knowledge Work – Configuration Structures

- From parts and their interconnections (composition) to the whole
- Reductionism
- Holism

Process Support

- Process Step
  - Checklists
    - Constraints
    - Claims and Audit Trails
    - Sampling and Quality Assurance
  - Exceptions
    - Governance, Management, Administration, Knowledge Work
    - Organization wide procedures for exception handling
    - Rework

Flexibility, Adaptability, Agility and Composability of Process

Flexibility

- Least Commitment Principle
  - The problem is not fully known (the detailed understanding is always relative to the agent who performs the process-steps)
  - Expand the process on demand

Adaptability

- Partial Evaluation
- Adapt to Qualities – Product and Process
- Adapt to configurations, configuration items
- Adapt to agents capabilities
- Adapt to changing environment

Agility

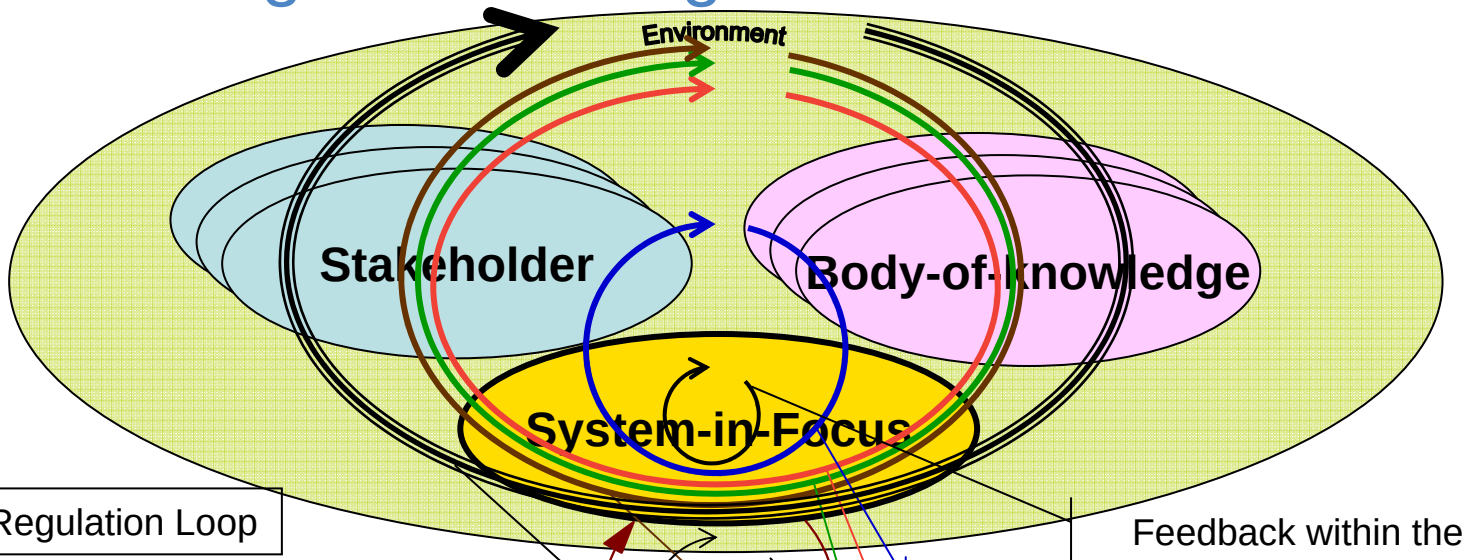
- Change is constant: requirements, development environment, emerging technologies
- Delayed binding reduces the propagation of change

Abstraction and Composability

- Is the Process Diagram enough?
  - Configuration structure
  - Governance structure
  - Administration Structure
  - Knowledge Work Structure
    - Roles and capability maturity structure

Evolving Process and Product

Understanding the evolving need



The Internal Regulation Loop

The Systems Thinking Loop

The Social Agreement Loop

The Design Loop

The Holism Loop

The Survival / Emergence / Emancipation Loop

Feedback within the system

Feedback between the Observer and the Observed

Feedback between multiple observers

Implemented Solution

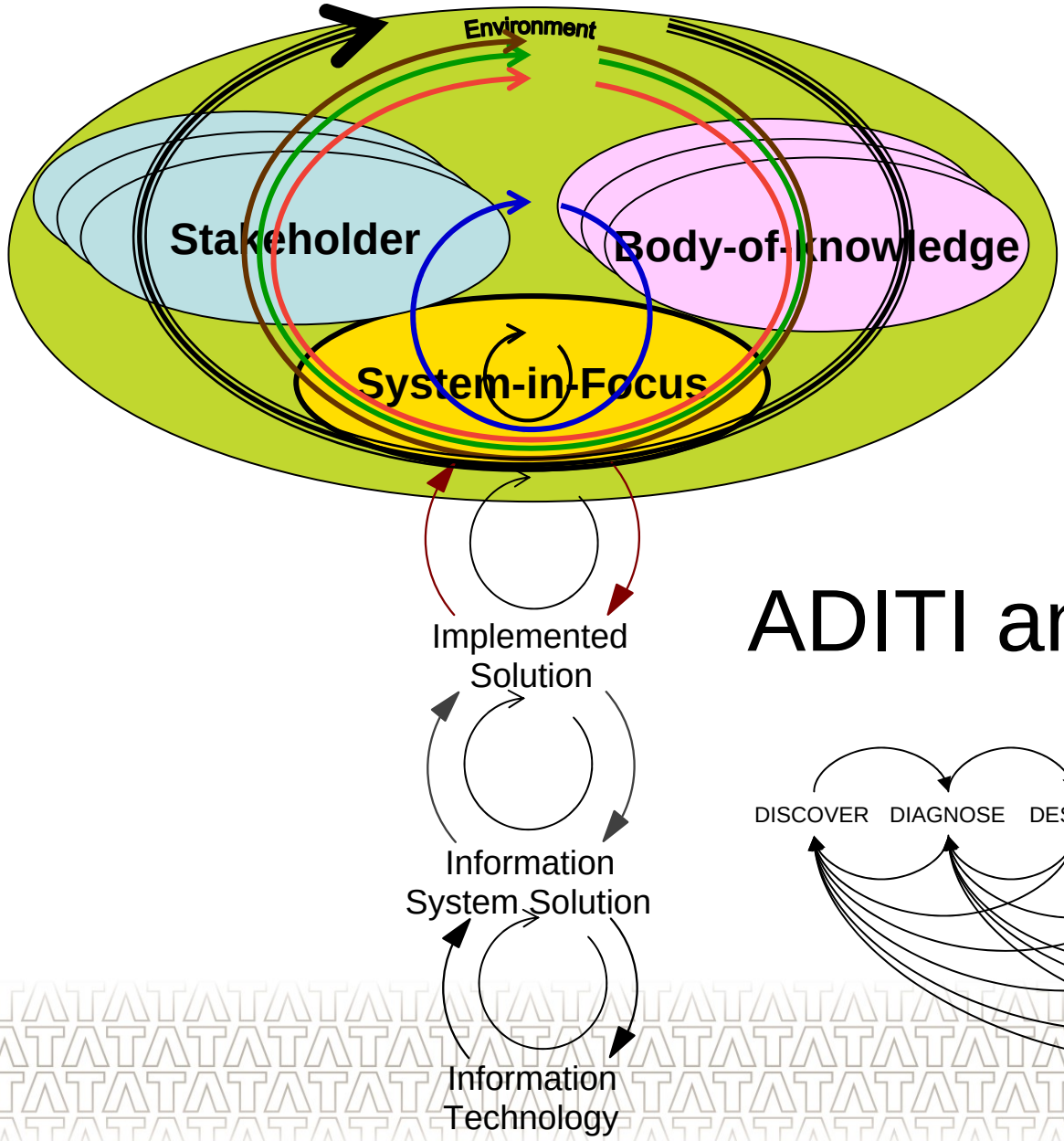
Feedback between System-in-Focus and Body-of-Knowledge

Feedback between Models

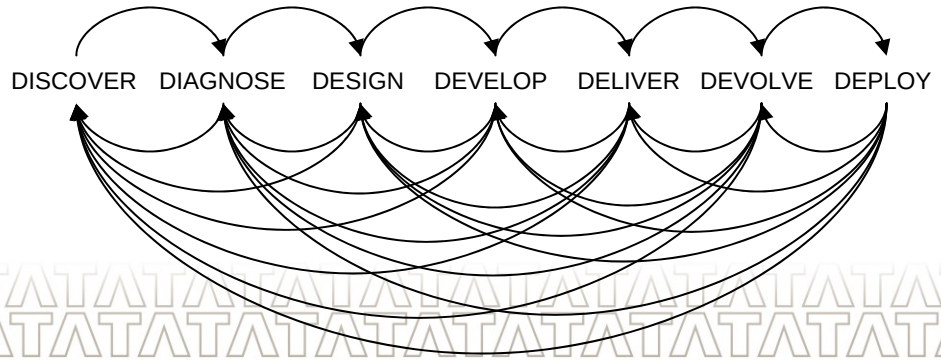
Information System Solution

Feedback between Environment and System

Information Technology



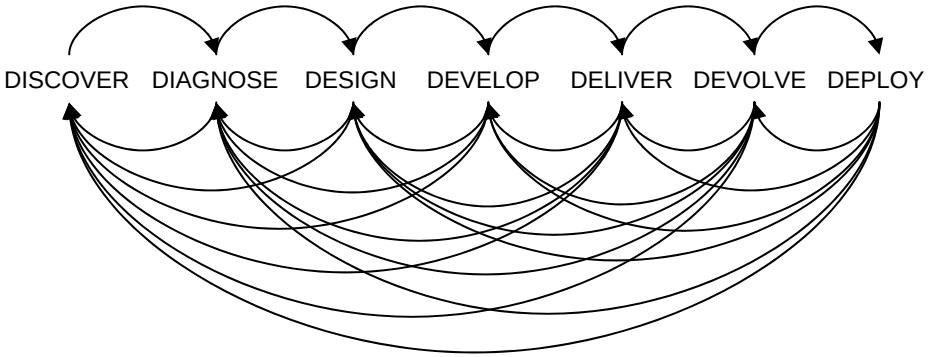
ADITI and DIKSUCHI



Unity of Process and Product

Unity of Process

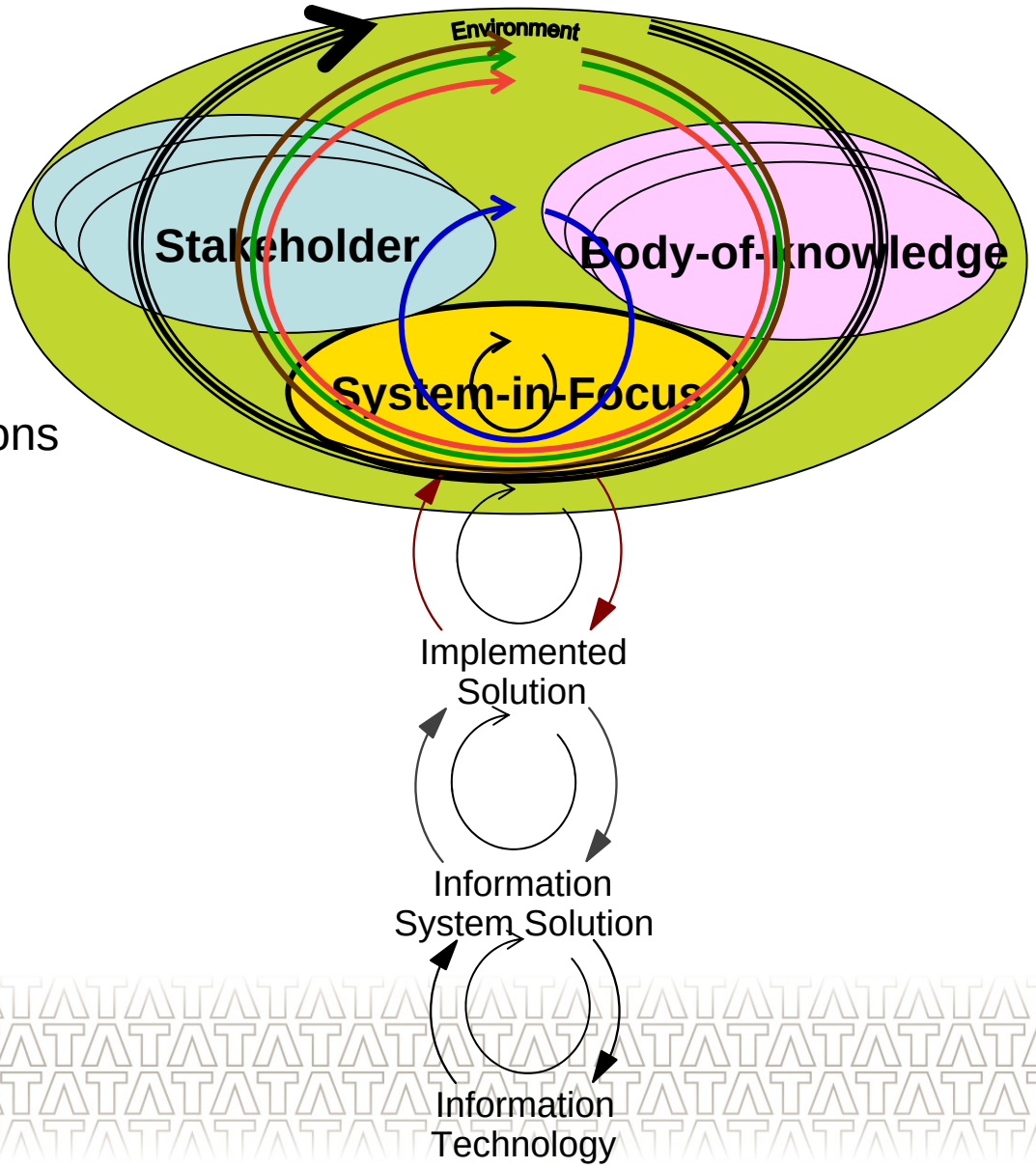
- There is just the holistic problem solving process: DIKSUCHI
- Its steps are specialized to the needs of the Software Process



Unity of Product

- Configurations characterize the "Product"

- Levels of Discourse: Configurations at each level
  - Domain Requirements
  - Domain Solution
  - IT Solution Architecture
  - IT Solution Engineering
  - IT Construction Engineering
  - IT Installation Engineering
  - IT Release Engineering
  - IT Deployment Engineering



Conclusions

And miles to go before we sleep …

- Learning from this exercise
  - Need for interdisciplinary research
  - It is the ecosystem of the Technology that determines its evolution
  - Technology provides a handle on imperatives (workflow, enactment, …)
    - The internal needs of the engine possess us, the engineers of technology
  - At another level Technology ameliorates the needs in an ecosystem
    - The external needs determine the uses of the engine to reach desired goals
- Further Work
  - Specializing Diksuchi to the Software Process
  - Generating the Software Project Process
  - Layering the QMS for binding on demand
  - Specializing Discrete Event Simulation to the needs of Process Quality Analysis
  - Defining a Type System and its Unification for Software Process
  - Defining a Reasoning System for Deducing Properties of Process Outcomes

Join me;
look at the
world in the

Thank You!