Large Scale Requirements Engineering: why it is different?

ICEIS 2004

Wednesday, April 16, 2004

Kalle Lyytinen, Case

Motivation

- >50% of large, complex IS development endeavors fail (Gartner, Standish)
- There is a "complexity horizon" beyond which successful design seems random
- The traditional RE strategy: reductive analysis and fine-grained control of components through formalization
- Our contention: the strategy is flawed -- a fallacy of composition and detailed analysis to regain control
- Consider the following…

NASA Overview
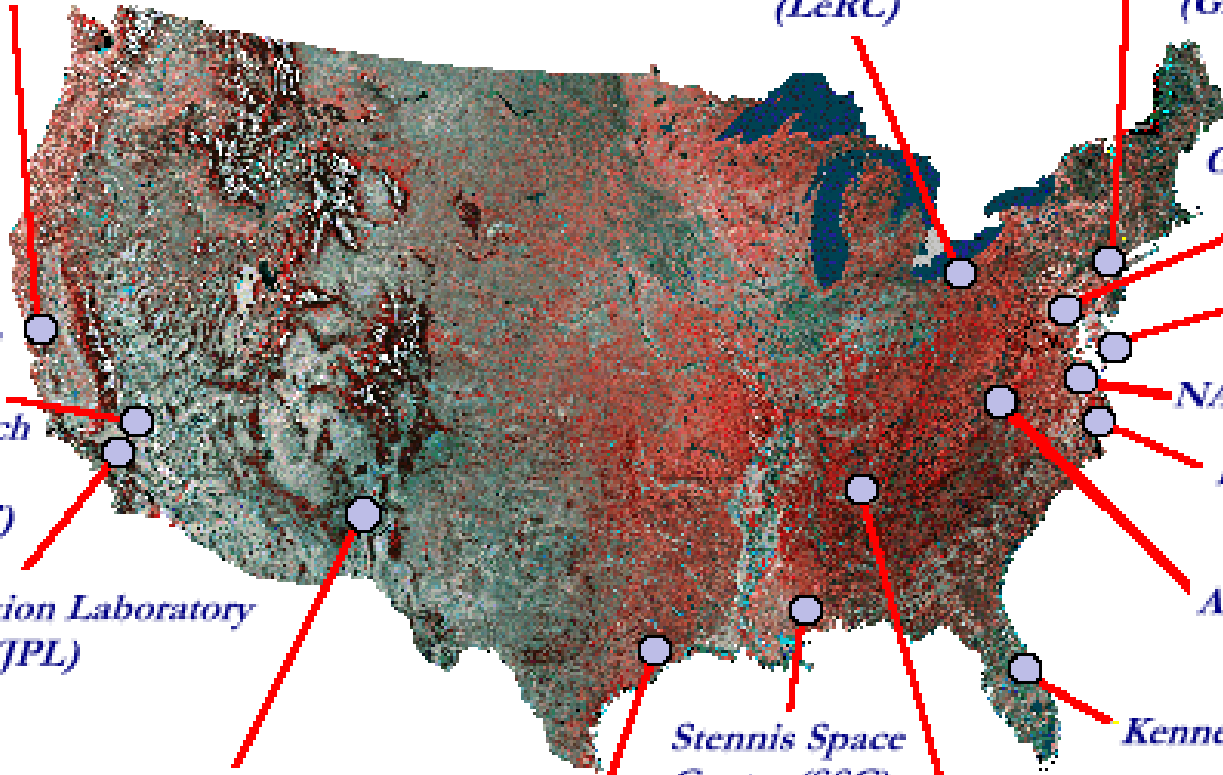
The NASA Problem

"The Office of Management and Budget … requires that Federal agencies maintain a single, integrated financial management system. Because NASA's use of individual non-integrated systems at Headquarters and Centers to meet its statutory and regulatory reporting requirements does not conform …, NASA continues to report a financial management system material weakness."

Source: NASA CFO Act of 1990 5-Year Plan, September 1998

The Full Story

US General Accounting office finds NASA accounting and financial information systems constitute a "material weakness" of the Agency

NASA conducts a two year "study" to determine the appropriate resolution to the GAO finding.

NASA launches the NAFIS Project
- ➔ **NASA Accounting & Financial Information System**
- ➔ **"Everything to Everyone"**
- ➔ **Estimated to take 3 years, cost $10 Million (US)**
- ➔ **New development freeze**

1987	1988	1989	1990	1991	1992	1993	1994	1995

NAFIS CANCELLED!!!
- ➔ **8 years and $33 million later**
- ➔ **Estimated to require 3 more years and $90 million**
- ➔ **Replaced with "off-the-shelf" solution**
- ➔ **NASA CFO: "If we can't find software that works the way we do, we'll change the way we do business."**

5

Where Are They Now?

- 16 years and $100 million into the process
- Core Financial alone will cost more than $110 million (more)
- 2004 will attempt a complete rollout of "shrinkwrapped" Core Financial
- **After 15 – 30 year old systems still running the Agency!**
- **The likelihood of some level of failure with IFMP is still high as the organizational control of the system and its structure is not aligned**

Complexity and Evolution

- Complex systems in complex domains are irreducible and incompressible (Cilliers)
- "Successful" complex systems are evolutionary not revolutionary (revolutionary change is incremental!)
- Complex systems entail evolutionary requirements analysis and understanding what makes them evolve
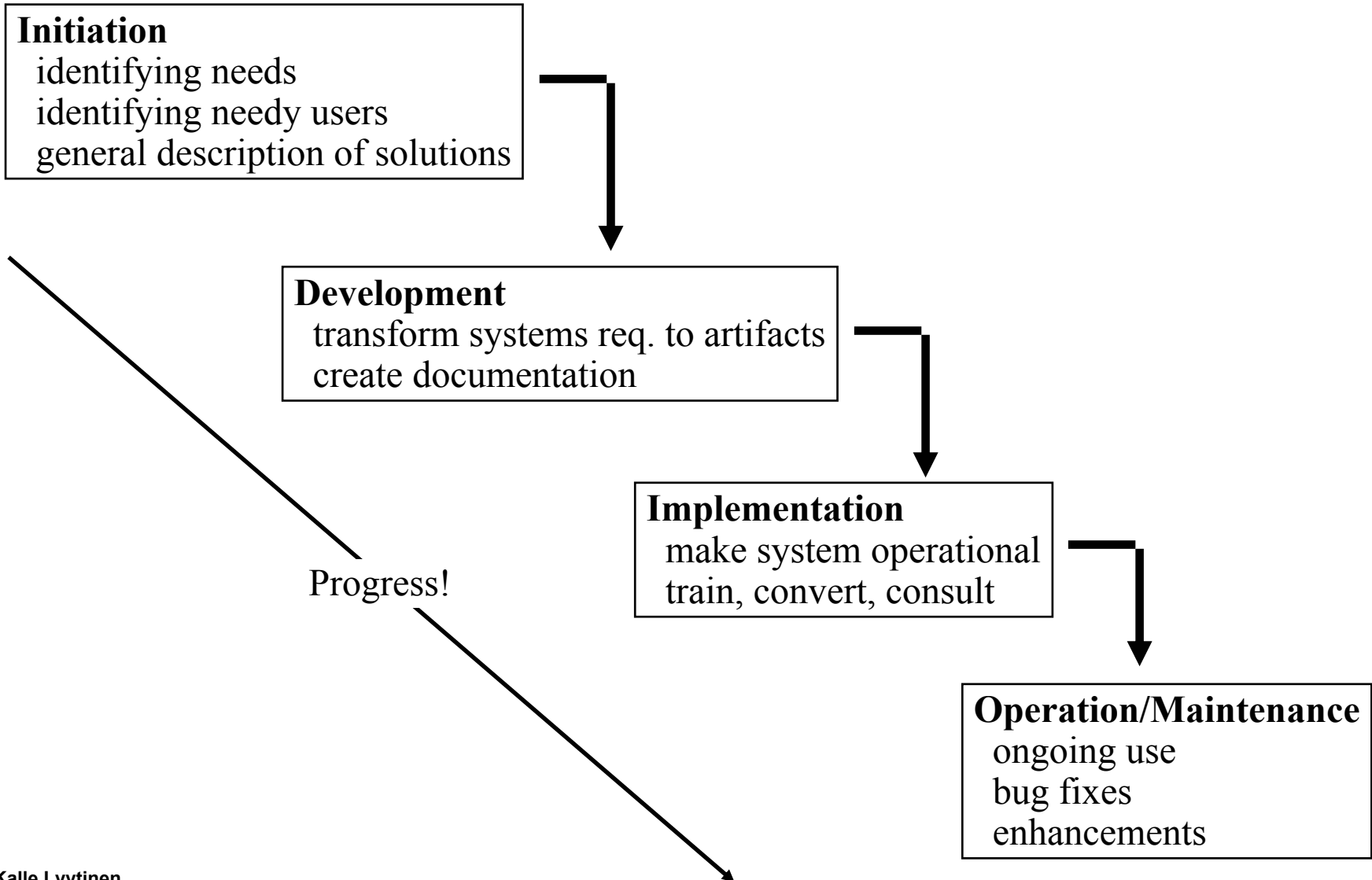- Evolutionary processes occasionally produce equifinal results, but not by controlled design
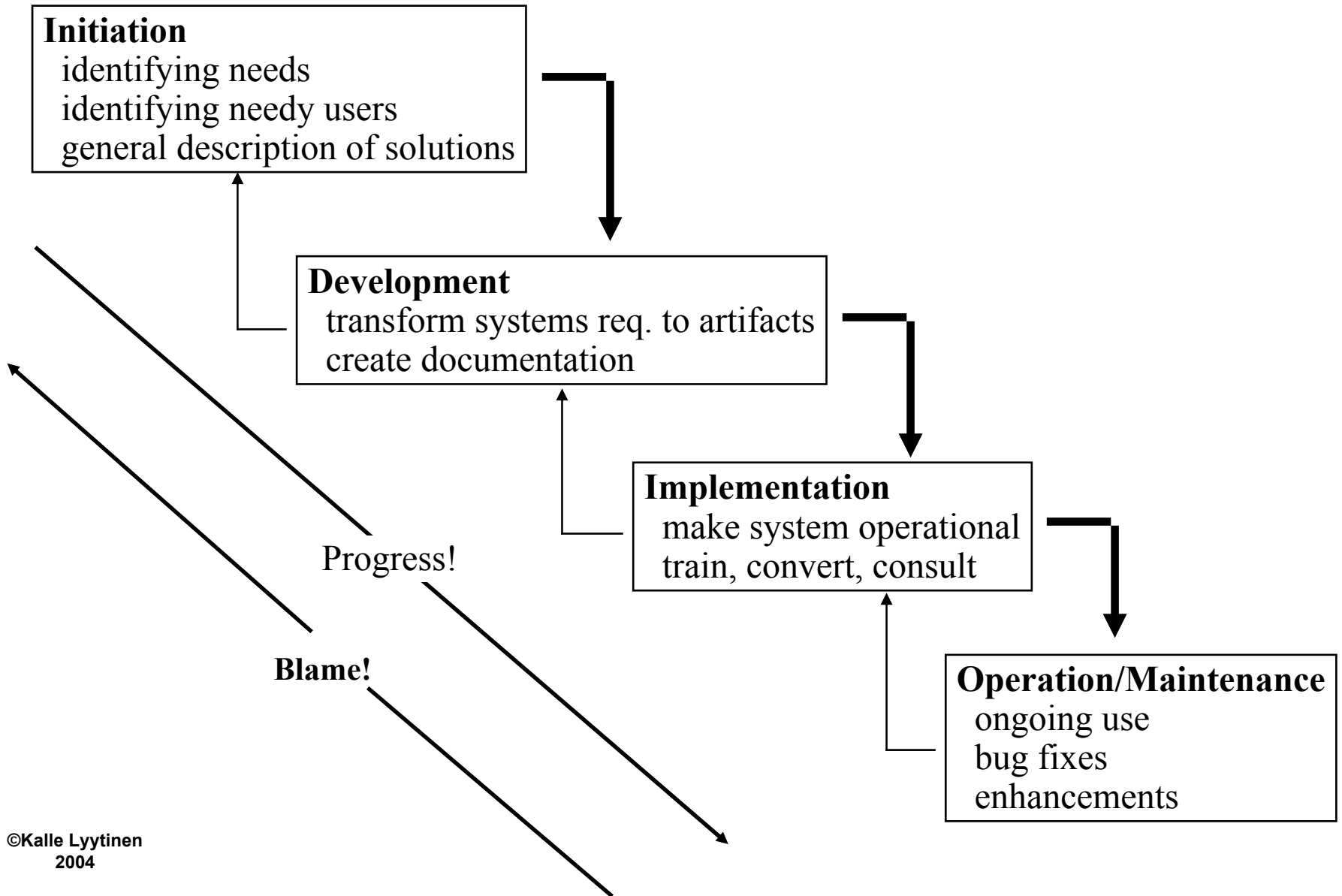
The Challenge

- We have built very complex systems for complex domains (e.g. wireless services, POTS, Internet)

- We just can't do it on purpose every time, at least not very well.

- We need a different strategy than the one we have been following.

The Challenges of System Design

- The hardest part of design is identifying the set of problems to be addressed (Brooks 1995, Carroll 2000)

- There is rarely only one clear problem to choose to address (March and Heath 1994) (functional instability)

- Initial system design involves

  - Satisfying the needs of conflicting, overlapping groups of stakeholders

    - Not all can win (politics!)
    - Still, want as many to win as possible

- Problem identification → System design *project* identification and selection

  - The system's success or failure is directly tied to its ecology within the environment

    - Examples: Suchman (1987), Orlikowski (1991), Bowers, Button, and Sharrock (1995)

Responsibility Push-back

- **When you cannot identify clear endogenous or actional causes for problems, you look "upstream" to the priors in the environment**

- **The existing order that gives rise to task domains and their tasks is the *environment***

- **The environment is taken as a natural order, not immutable, but not socially constructed, either**

- **Failures are usually attributed *to failure in requirements analysis*.**

- **This constitutes "push-back" along the causal pathway identified in nearly all system-building models.**

A Classic Case in Point

Initiation
 identifying needs
 identifying needy users
 general description of solutions

Development
 transform systems req. to artifacts
 create documentation

Progress!

Implementation
 make system operational
 train, convert, consult

Operation/Maintenance
 ongoing use
 bug fixes
 enhancements

Refinement on the Idea

CASE CASE WESTERN RESERVE UNIVERSITY

Initiation
 identifying needs
 identifying needy users
 general description of solutions

Development
 transform systems req. to artifacts
 create documentation

Implementation
 make system operational
 train, convert, consult

Progress!

Blame!

Operation/Maintenance
 ongoing use
 bug fixes
 enhancements

13

A Critique

- The models say nothing how solutions or requirements emerge and how they move within the process

- Agency dynamics of system development
  - Specification as contract (save your ass!)
  - Reluctance to "cross the barrier"

- The necessary reform: systems people must move into the domain space as principals (new logic of responsibility)

Five Reversals

- Complex system development **creates** goals; it does not follow them
- **Constraint**s are more important than goals in complex system development
- Optimality in relation to goals is dysfunctional: strive to **satisfice** (Simon)
- **Solutions drive problems**- not the other way round
- **Problems and solutions** do not follow uni-directional causality- they **are mutually created and dependent**

A New Strategy

- An ongoing interplay of discovery on both supply and demand sides (solutions / problems)

- Iterative movement between emerging articulation of problems and potential solutions

- Objectives:
  - continual search for "killer issues" that will render the systems or process moot (constraints)
  - continual adjustment of both problems and solutions toward convergence (protracted walk in functional and political ecologies)

16

A Prolegomena For High Level Requirements Analysis

- **The biggest hurdle is understanding the need to do it: doing the right thing is easy once you know it; the hard part is figuring it out**

- *The required reversal*: **do not look for the "right" way; focus on avoiding the wrong ways.**

- **The most important issues are those that you assume will not change, because those will shape the emerging order.**

- *High level requirements analysis* **is a task of narrowing the search space (functional and political ecology) by interpreting design options in light of what will not change, and use that to calibrate action spaces in the residual.**

The Possible

Comparatively Likely

Unlikely

Viable
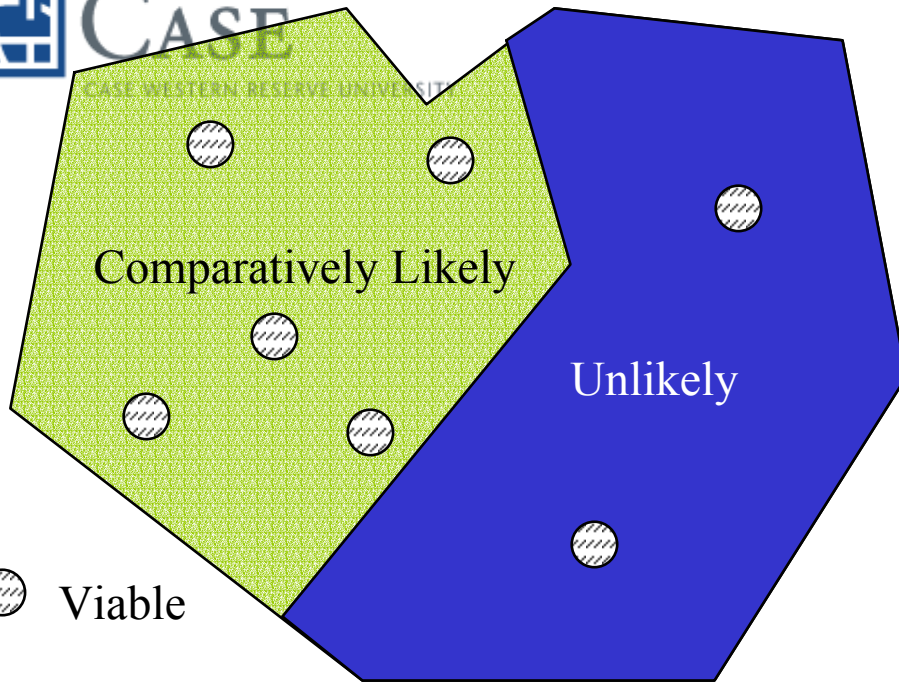
Retail electronic commerce: the changes in the high-level order are likely to be modest; high-level design needs and constraints are largely already mapped out for us.

Wholesale electronic commerce: fundamental changes at the high-level suggest need for radical rethinking of what should be done.

- **The search for local optimality is at least to some degree blind.**
- **Viable alternatives might exist within the unlikely space (c.f. Progressive).**
- **What can be achieved- not what ought to be achieved. Ideas are opportunity-driven; implementation is constraint-driven.**
- **When high-level factors are not changed, the emerging solution will be an incremental change. When they are changed, the emerging solution will be a major departure in which prediction becomes impossible.**
- **The focus on the what will not change becomes more important as the prospect of high-level requirements dislocation grows.**

Features of the HLRE model

- requirements map *solutions* with *problems*
- both are *socially constructed , fluid* (M:N mapping)
- *RE* is a satisfacing, "garbage-can" process that involves mapping and reconstructing *problem* and *solution* spaces

> solutions--> problems--> solutions

- Requirements are emergent and discovered through a contracted process that yields a **functional ecology** of requirements
- what will and will not change involves finding stabilities in the socio-technical network **a political ecology**
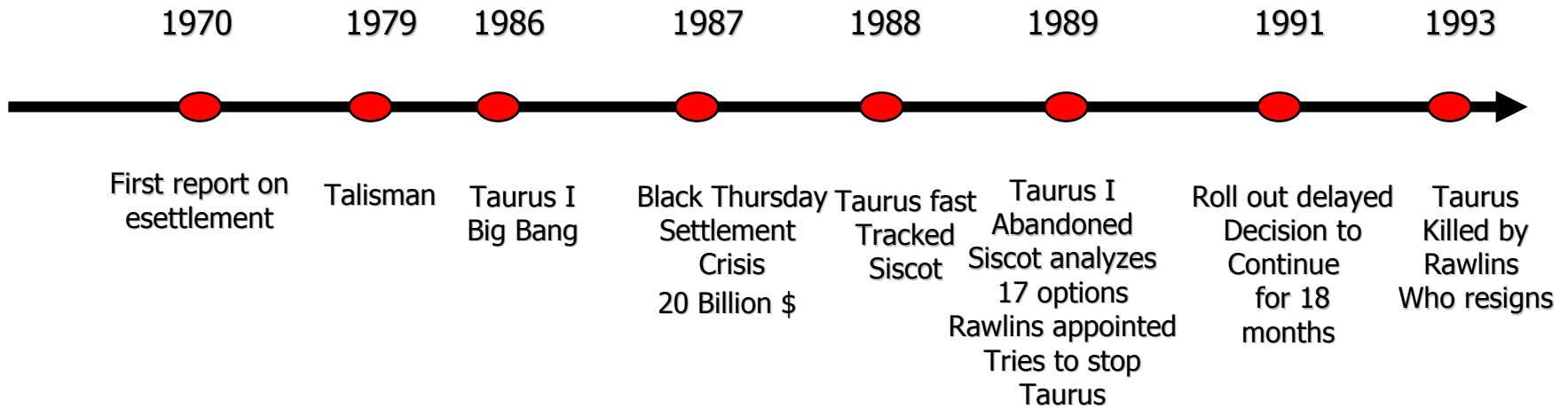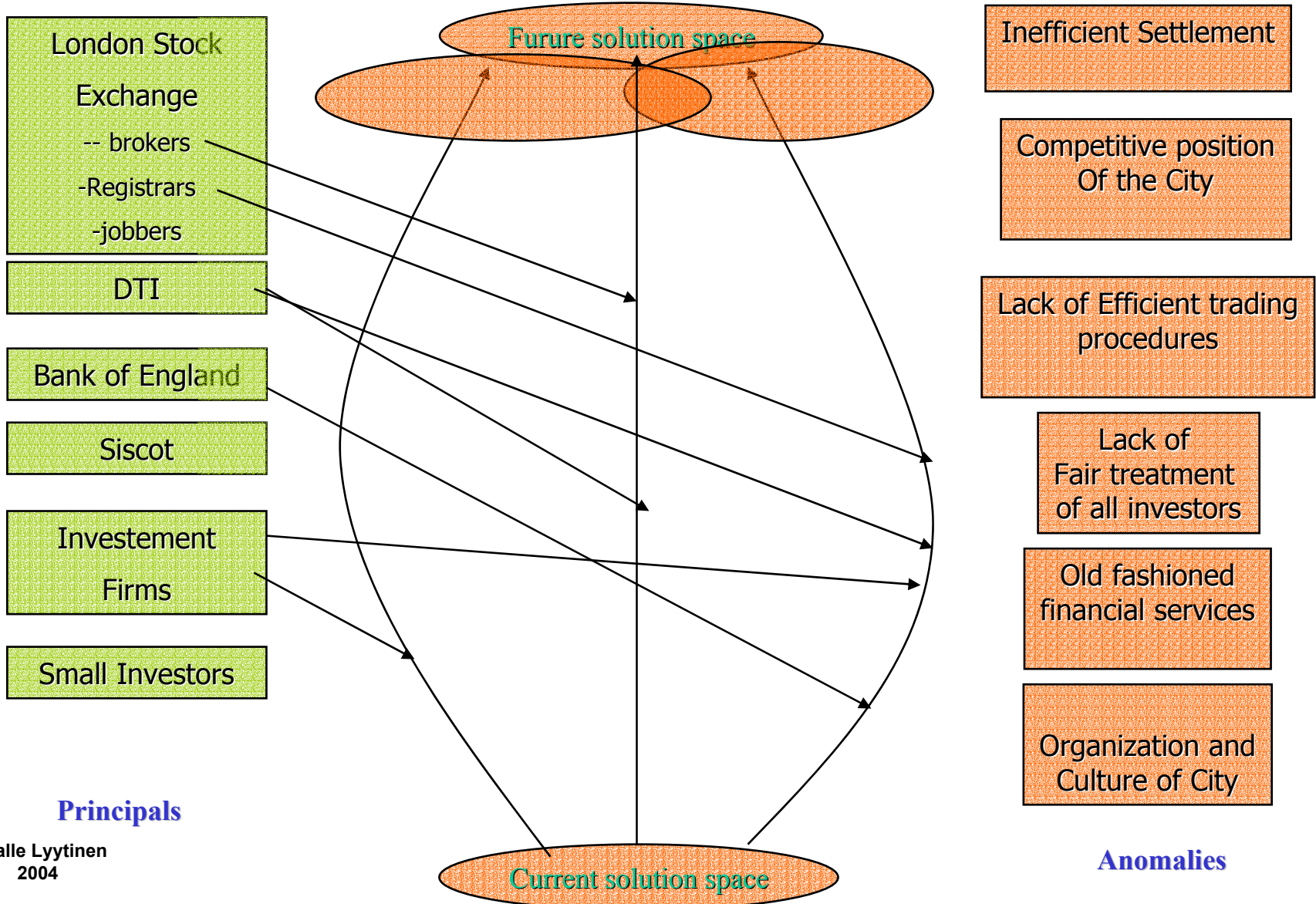- Thereby HLRE can be characterized as *heterogeneous engineering*
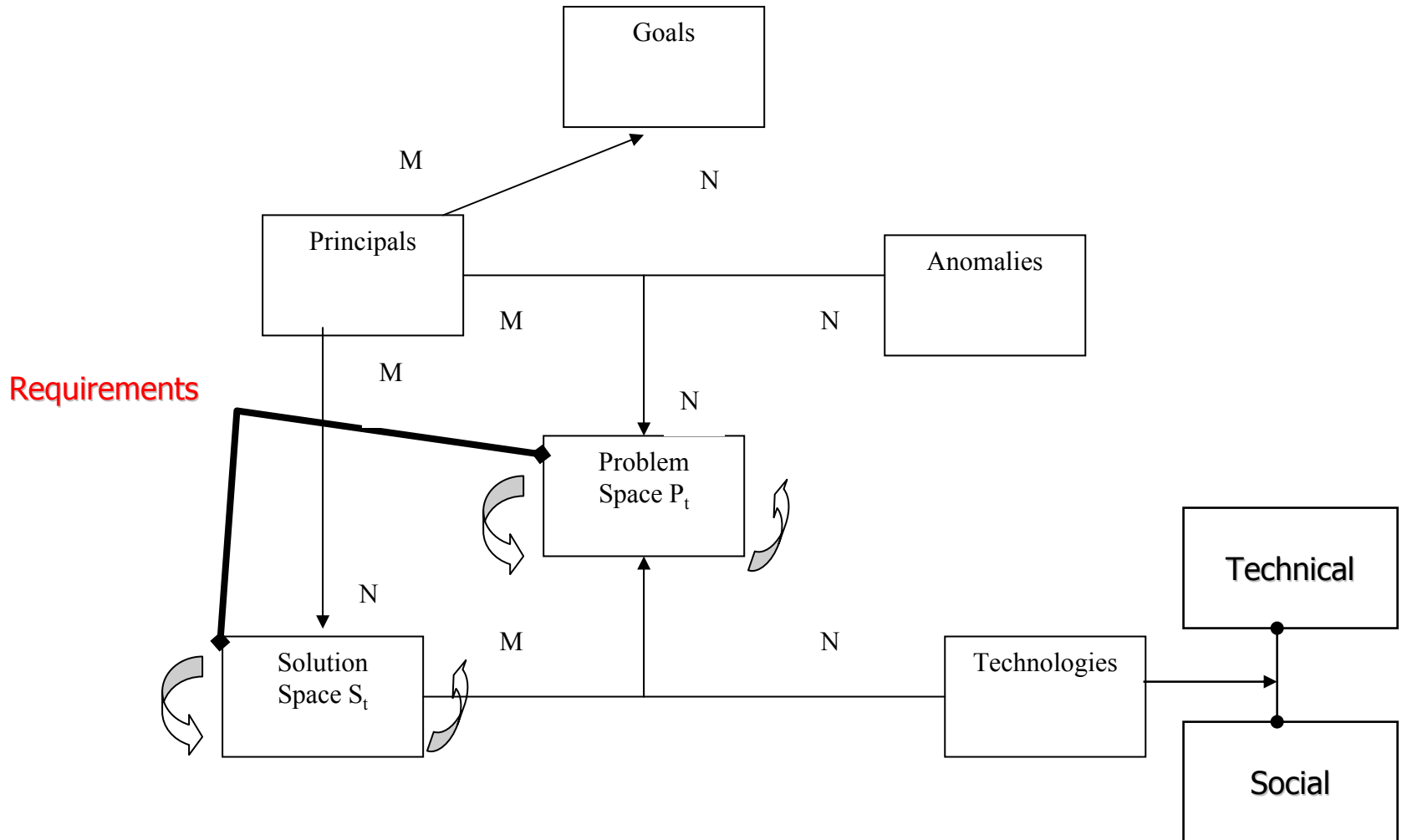
What Does this mean?The Taurus Case

- **The facts (Drummond: Escalation in Decision Making)**
  - Largest software development initiative in Europe 1990-1993
  - Estimated cost 55 million $ and 150 million $ from
  - cost c.a. 150 Million $ for London Stock Exchange, 580 million$ from the securities industry
  - Abandoned inn 1993 when the projected delivery was moved to 1996 and the cost to 250 $
  - Reasons for spiraling cost and difficulty:
    - escalating requirements
    - Difficulty to match requirements with feasible implementation (though it was doable)
    - Escalating problems, goals and principals
  - **Radical initiative: restructuring of job-floor operations and introducing totally electronic settlement for LSE**
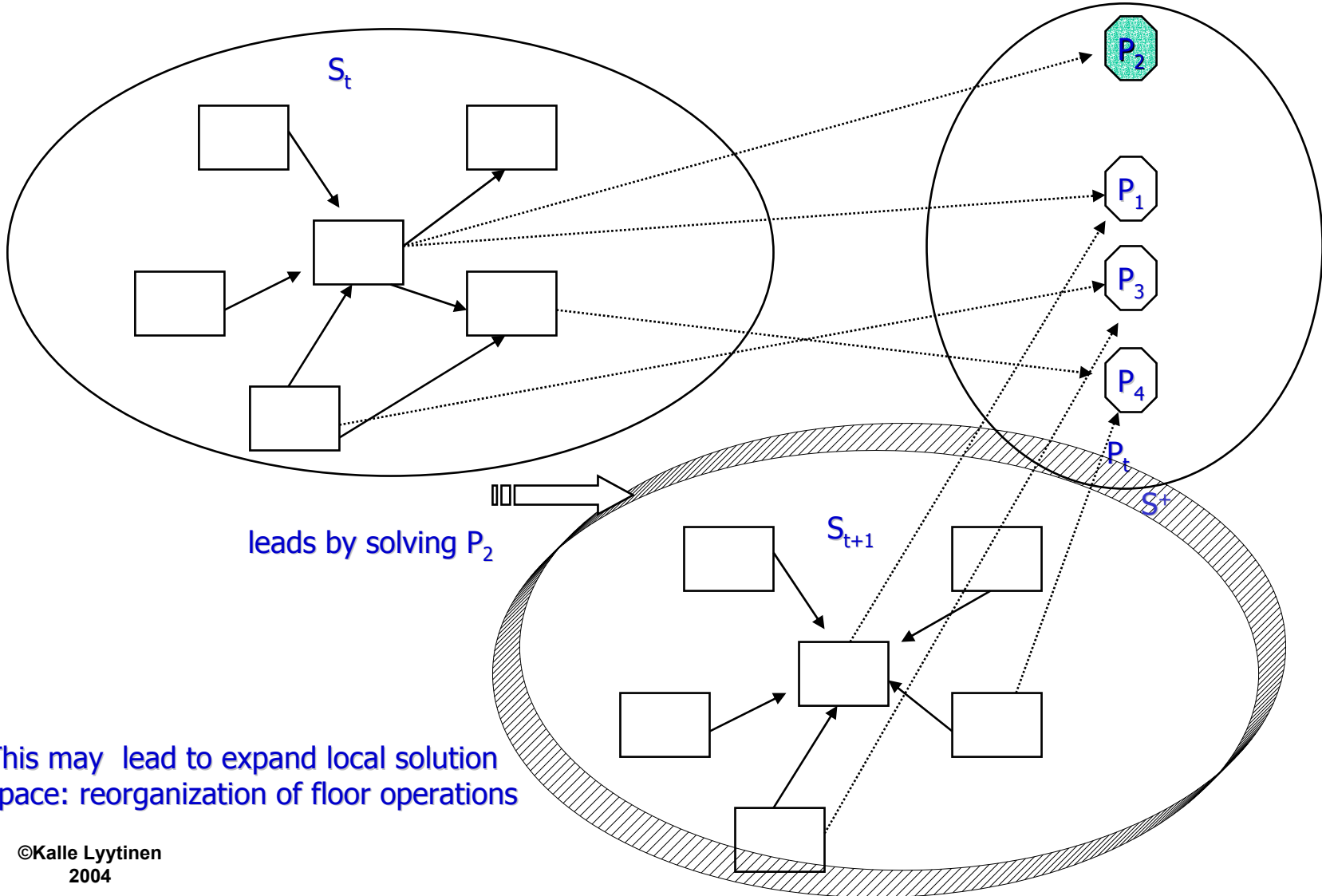
The history

1970	1979	1986	1987	1988	1989	1991	1993
First report on esettlement	Talisman	Taurus I Big Bang	Black Thursday Settlement Crisis 20 Billion $	Taurus fast Tracked Siscot	Taurus I Abandoned Siscot analyzes 17 options Rawlins appointed Tries to stop Taurus	Roll out delayed Decision to Continue for 18 months	Taurus Killed by Rawlins Who resigns

Taurus reconfigured

Principals

London Stock Exchange
-- brokers
-Registrars
-jobbers

DTI

Bank of England

Siscot

Investement Firms

Small Investors

Furure solution space

Current solution space

Anomalies

Inefficient Settlement

Competitive position Of the City

Lack of Efficient trading procedures

Lack of Fair treatment of all investors

Old fashioned financial services

Organization and Culture of City

S: Solution Space: Current settlement   P: Problem space: Speed, cos, competittion

S_t

P_2

P_1

P_3

P_4

P_t

S^*

S_{t+1}

leads by solving P_2

This may lead to expand local solution
space: reorganization of floor operations

Requirements redefined

1. **A set of relationships** between a solution space and a problem space
   - no more fixed than an evolving understanding of the features of both the solution space and the problem space
   - requirements are not *discovered* but *constructed*
   - requirements provide links to solution space contextualization (functional ecology)

S: Solution Space

P: Problem space

S_t

R_o

R_o

R_o

R_1

R_2

R_3

R_4

P_2

P_1

P_3

P_4

P_t

S_{t+1}

26
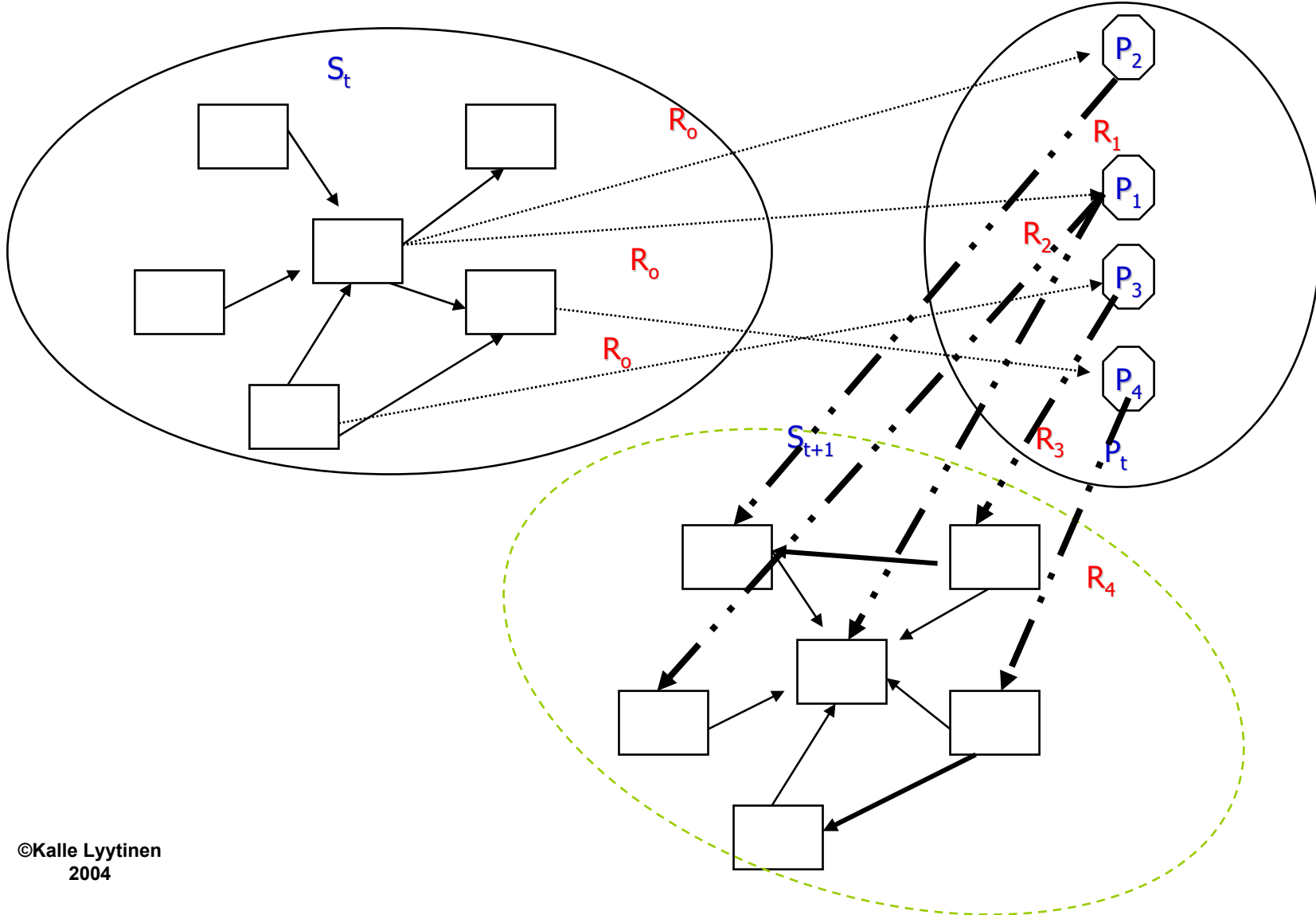
Requirements Redefined

2. A typology of requirements: 1) *an objective requirement*: a want, need or desire that corresponds to a problem as contextualized by a part or all of a current solution. Thus a functional requirement forms a relationship that comes from a solution looking for a problem.

2) a *constraint:* a restraining condition imposed upon a solution within S_{t+1} as contextualized by a problem within P_t.

How does HLRE work in practice?

- *Framing the Investigation*

- Empirical Examination of *In Situ* High Level Project Selection and Requirements Analysis with NMP process (50 interviews, 6 review meetings recorded)

- How does HLRE interact with project scoping and assessment and how does it get done?

Site Selection: NMP

- **Repeatably Successful Process**
  - Project selection and requirements analysis process has been historically stable, repeatable and produced consistently viable results
  - Has a history of successfully producing large, complex systems

- **Barriers to Entry**
  - Project selection is politically and economically sensitive
    - Do not want to share confidential information with 3rd parties
  - Observation would influence and bias outcomes
    - Restricted access (if at all) to sensitive meetings

The Field Site

- The New Millennium Program (NMP) at the Jet Propulsion Laboratory (JPL): A group in a NASA research laboratory located in Southern California

- The NMP program's mission: Space flight validate new technologies that are deemed important to NASA's future science missions
  - This includes maturing new technologies (TRL 4 → TRL 7)

- "Open and Fair" Competition
  - US Congressional Mandate
  - Process must be well documented

- New process cycle starts once a year
  - 11 cycles started, 5 in progress (each c.a. 100 million $)

©Kalle Lyytinen
2004

Potential Technology Flight Validation Needs

Exploration of the Solar System

Aeroassist Ballutes

Microspacecraft

Solar Sail
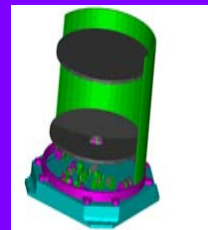
Optical Communication

Sun Earth Connection

Data Synthesis

Robotic Assembly

Advanced Instruments

Autonomy & On-board Processing

Constellation Operation

Tethers

Drag Free Inertial Sensors
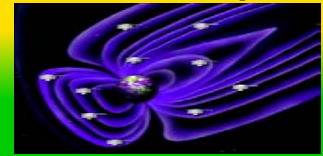
Gossamer Optics

Thermal Control Precision Metrology

Light Weight Deployable Precision Structure

Precision Formation Flying

Structure & Evolution of The Universe
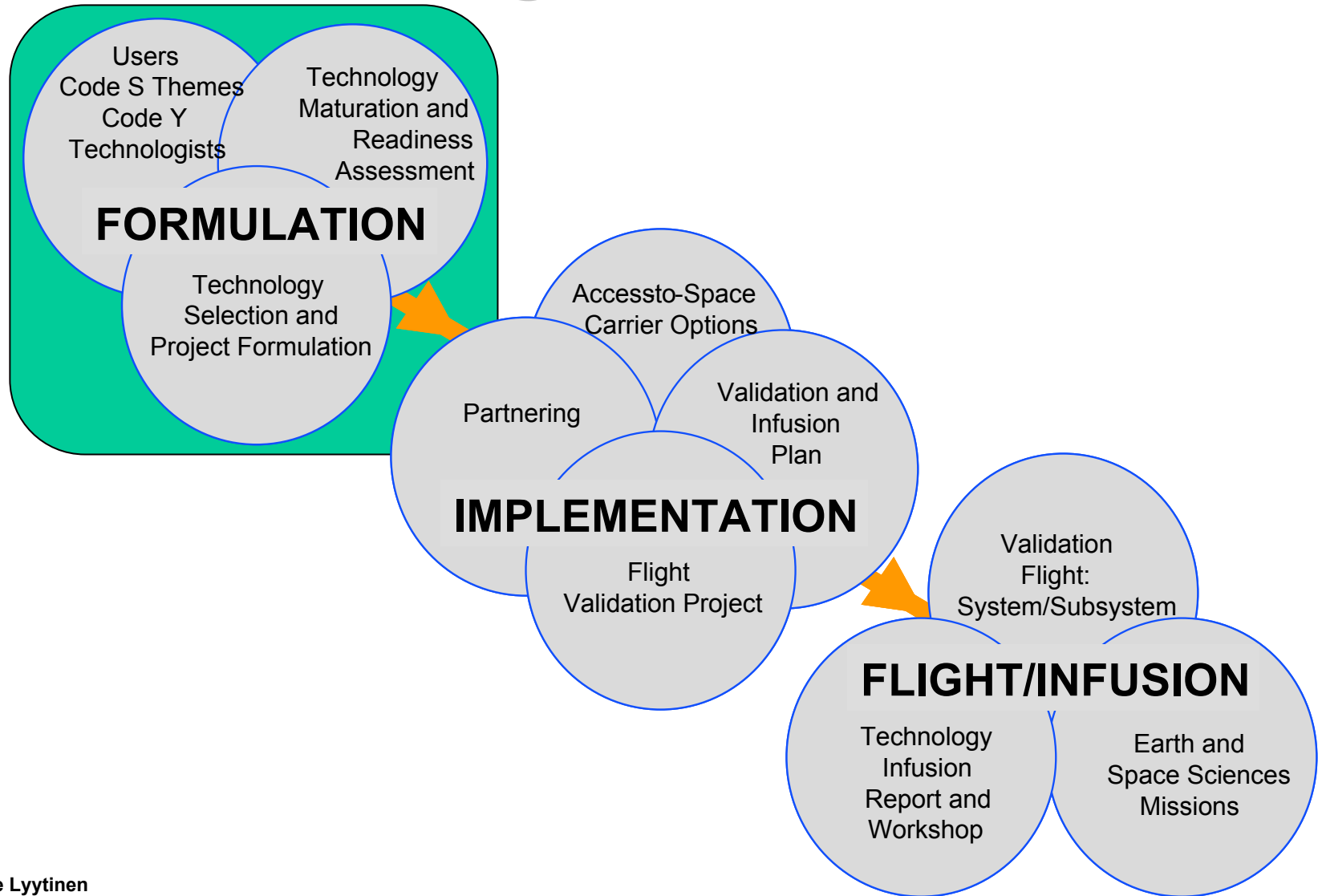
Astronomical Search for Origins
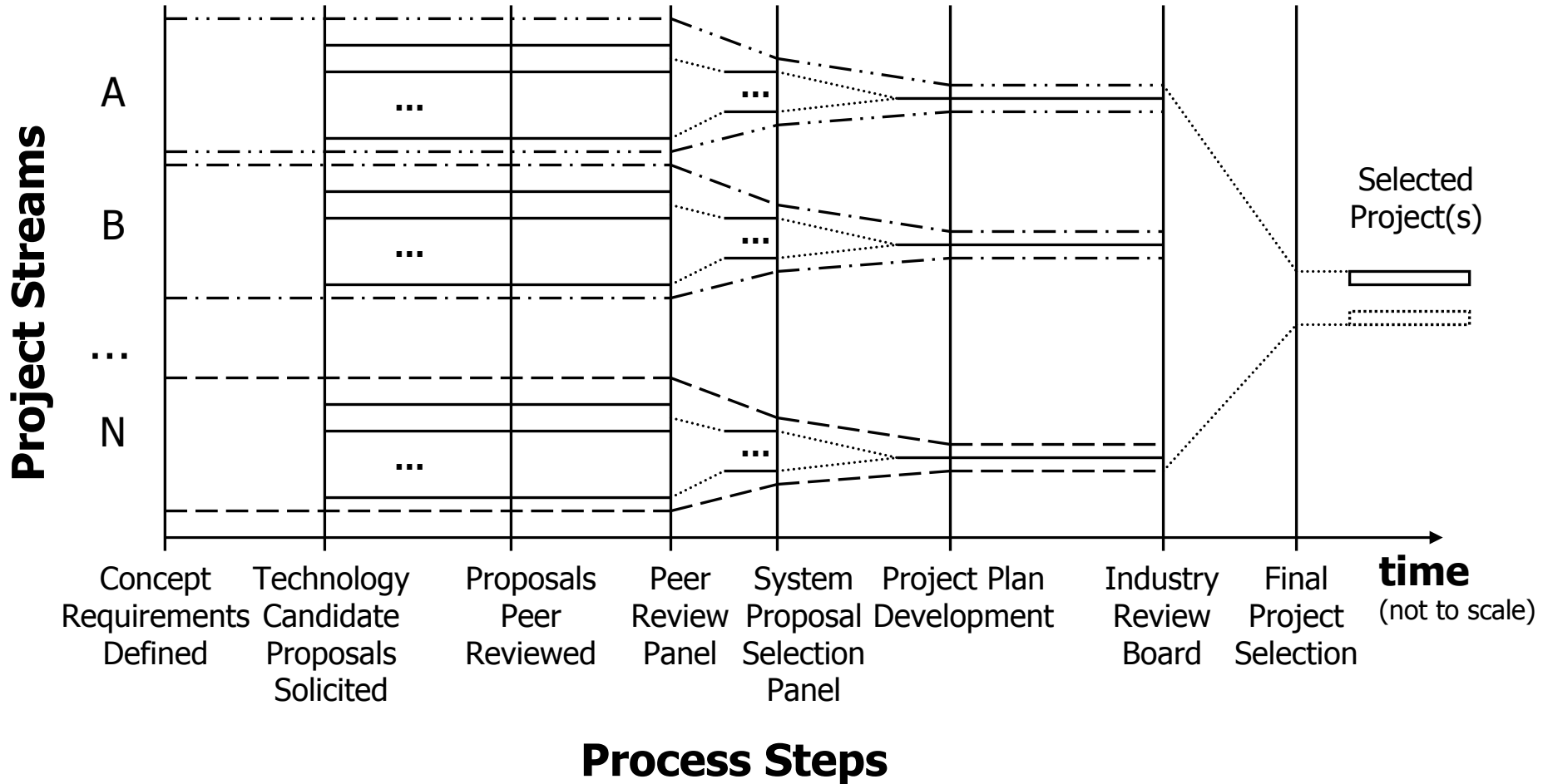
Classic Project Formation

- **First, Project Selection**
  - Determine project choices
  - Choose a project to fund and develop
- **Then, Requirements Analysis**
  - Determining stakeholders' wants, needs, and constraints for a project
  - Requirements Analysis traditionally follows Project Selection
- *Does this classic view match what is happening in the field with NASA?*
- How does NASA successfully build large, complex systems; is it consistent with this view?
  - How are requirements determined *in situ* for large, complex systems?
  - What is the relationship between requirements determination and project formation?
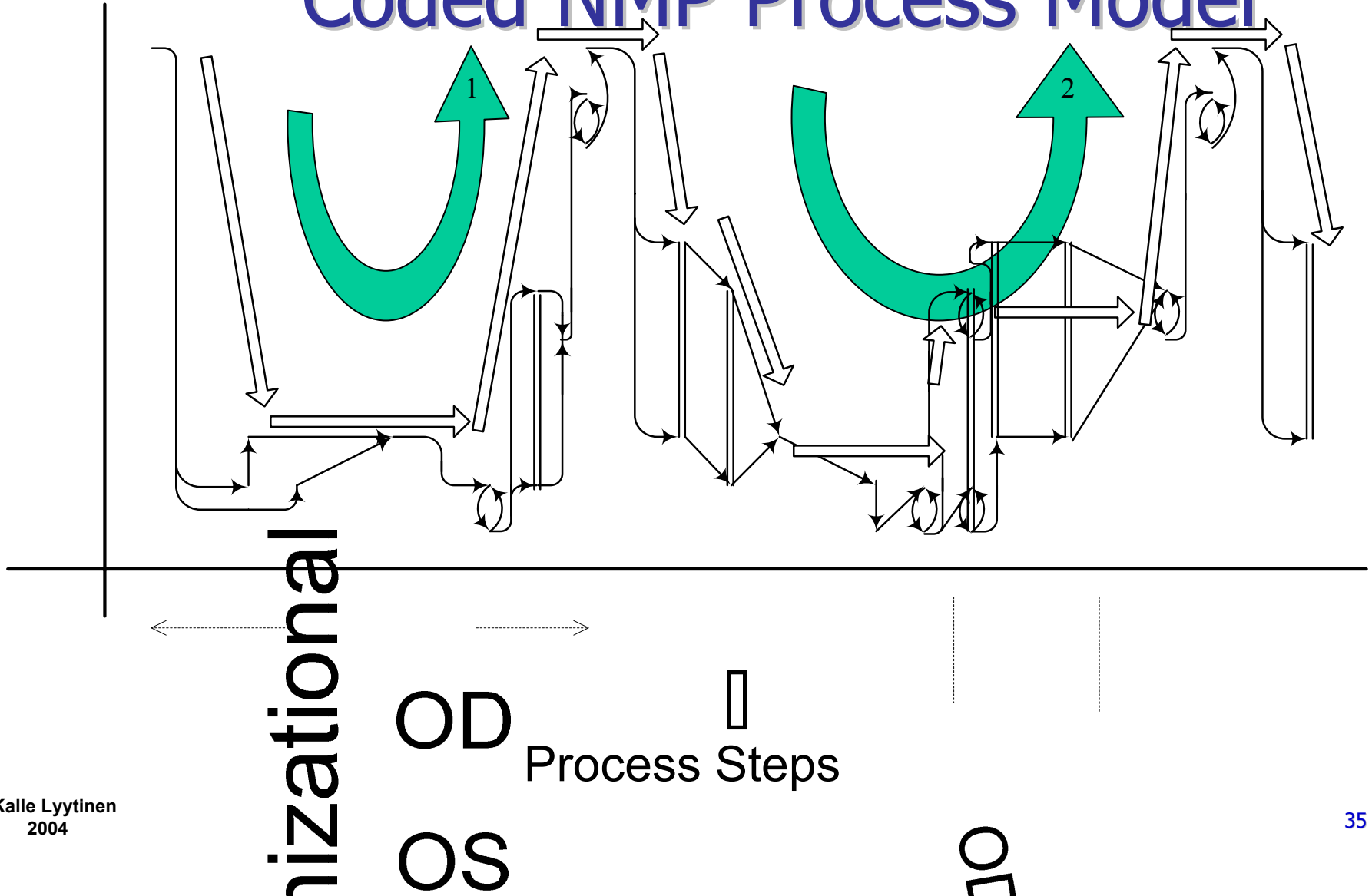
NMP Flight Validation Process

- ~9-10 mo. per project selection cycle
- 6 mo. used in project plan development

Project Streams

A

B

...

N

Selected Project(s)

Process Steps

| Concept Requirements Defined | Technology Candidate Proposals Solicited | Proposals Peer Reviewed | Peer Review Panel | System Proposal Selection Panel | Project Plan Development | Industry Review Board | Final Project Selection |

time (not to scale)
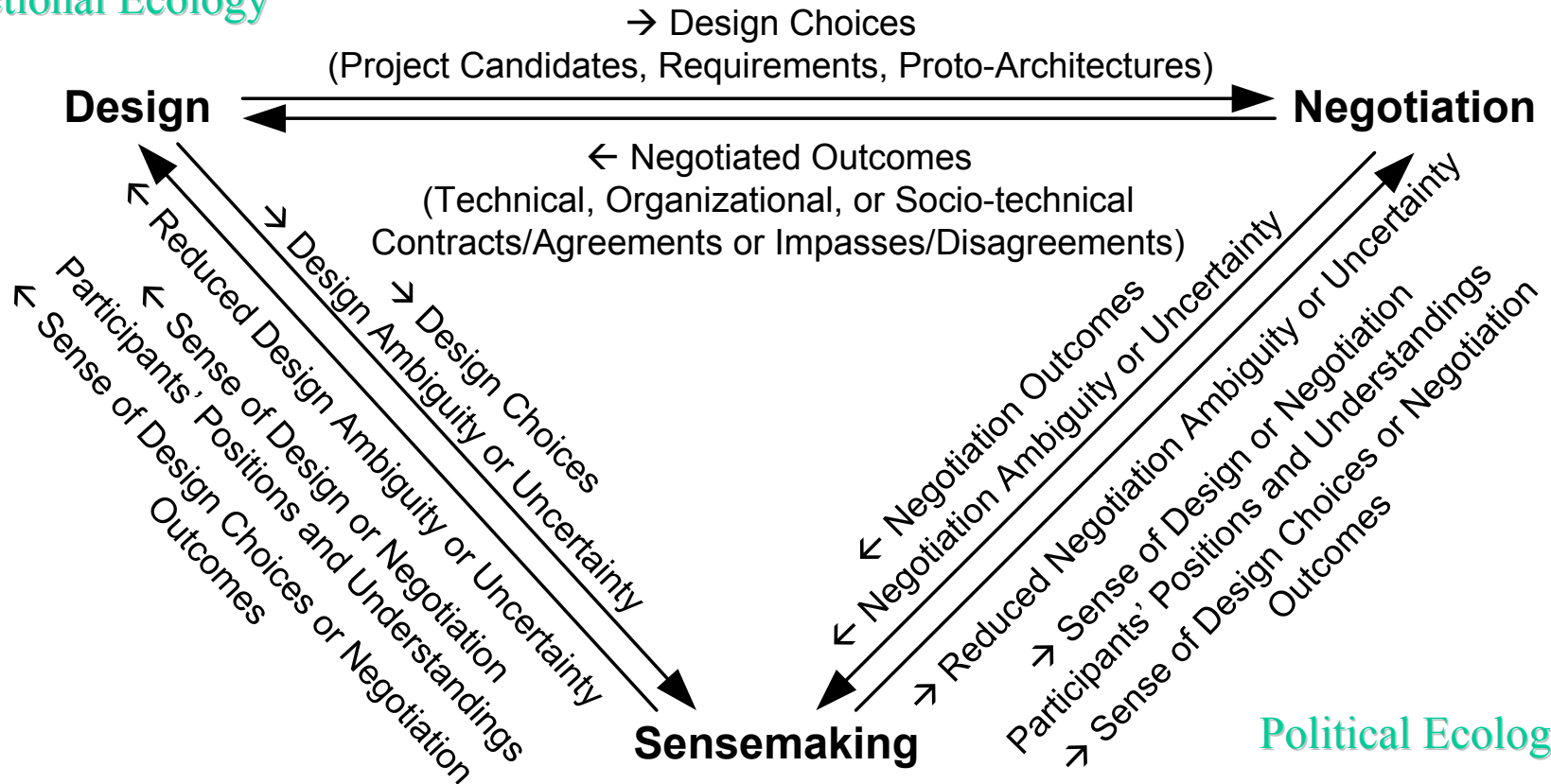
Coded NMP Process Model

Lessons

- Two cycles of feedback from different layers of organizational responsibility

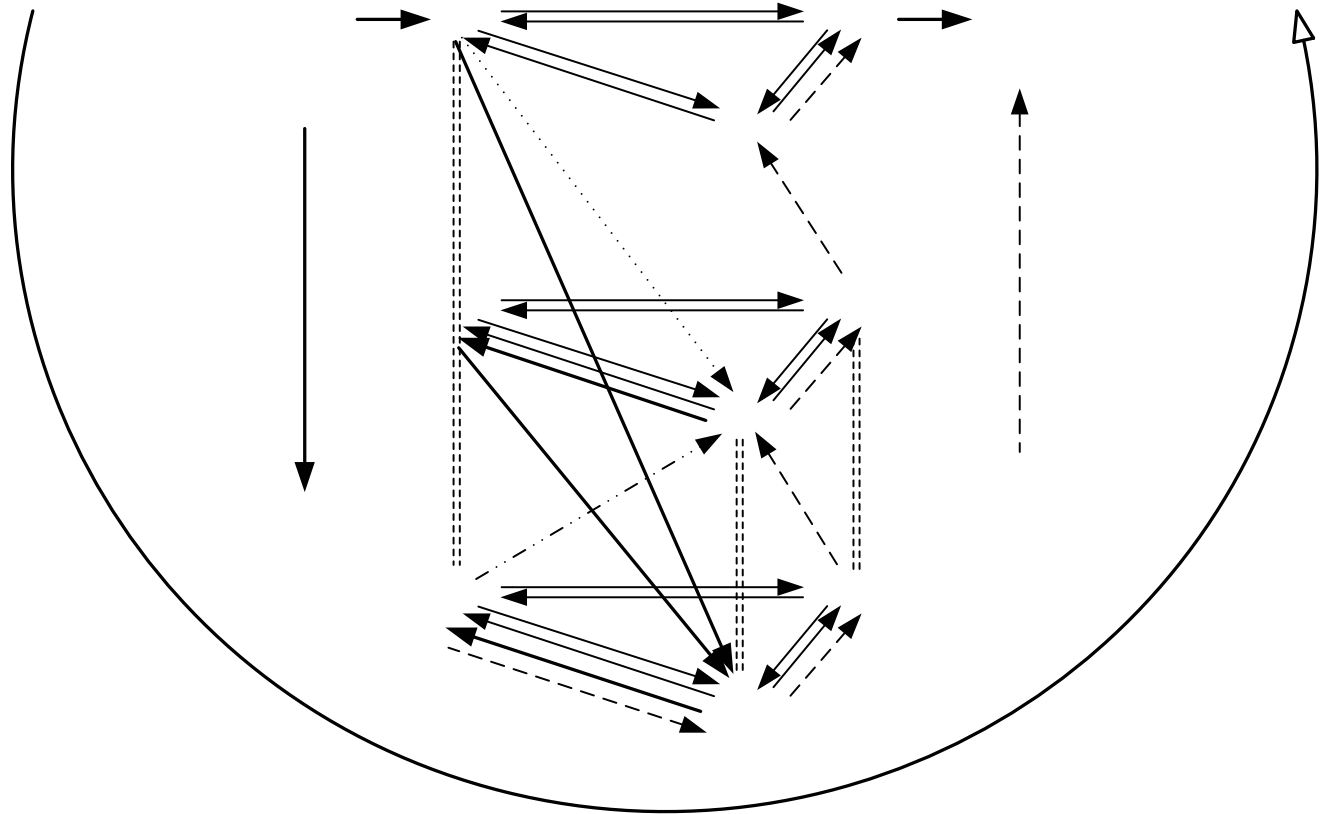- Drives a U shape cycle of design-sense-making and negotiation

DNS Activity Triangle

Project Selection and Requirements Analysis

- Procedures are generalizable *authority-activity components*
  - Engage technical, project, and organizational authority levels
  - Design, sensemaking, and negotiation primary activities
- Requirements analysis always performed at the different authority levels
  - Allows each level to manage their own view and control of requirements (REs as boundary objects)
  - Allows each level to learn and build upon that which was learned by the other levels (Res as boundary objects)
  - *Negative requirements* key in project candidate survival (focus on constraints)

Organizational Requirements Analysis

- An organization must combine effective authority-activity components to create stable S-P-FS Alignments for large, complex projects- this is not a Robinson Crusoe adventure

- Right organization of authority/activity components is critical for successful HLRE

- *Organization as Requirements Analyst*
  - Individual analyst is not equipped nor empowered to handle the organizational complexity of large system design
  - Individual analyst as shepherd for moving between levels of authority and different concerns of system design
  - Necessity to organize negotiation and sense-making through boundary objects

HLRE Boundary Objects in NMP

Design	Specifications (Requirements, Project) Proto-architectures (Concepts)

Sensemaking	Individual or Collective:
	    Sense of Design (Requirements, Projects, Proto-architectures)
	    Sense of Agreement/Disagreement
	    Sense of Process Participants Positions or Understandings

Negotiation	Functional (Technical, Socio-technical) Agreement/Contract Political (Organizational Power and Resource Allocation) Agreement/Contract

- Help build and reinforce legitimation
- Propel the system design process
- Used to form and stabilize the ecology of requirements (both functional and political)
- Connectors on the DNS Triangle

Attributes of Requirements and Projects
Co-Determination

- Multiple Parallel Competitive Requirements Analysis (several solution-problem alignments)
- Balance of Power (or lack thereof) amongst divergent authority levels
  - Disparate Stakeholder Power
  - Requirements Power Inequality
  - Consensus Emergence
- Requirements analysis was performed at the different authority levels
  - Manage their own of requirements
  - Reflective application of their requirements
  - Learn and build upon that which was learned by the other levels
  - Design validation
- *Negative requirements* key in project candidate assessment for "killing" issues; validates the logic of falsification in requirements determination

Implications

- all HLRE solutions are heterogenous (otherwise they are not complex)

- requirements engineering involves satisficing

- requirements involve also a discovery of goals and projections of new solution capabilities by different principals and policy making

- R_o and R_c form the basis for requirements specification: **A requirement specification is the set of requirements selected for attention by the principal requirements owners (which may shift)**

- **If you change any: principals, goals, solution spaces, problems (and by transitivity technologies) your requirements are likely to change**

- Requirements create dynamic relationships between **specific** solution spaces and a specific problem space such that the principal 's objectives are realized in the context of constraints

Conclusions

- requirements emerge through the identification of anomalies or changes in the solution space (technical opportunities)

- requirements lead to altered perceptions of solutions and problems and thus to an escalating "learning" cycle

- in large scale systems this necessitates local adaptation and heuristics where one examines *what should not be changed (what must be retained in the current solution space as a basis for emerging order)*

- Often failed requirements involve imitations, are developed randomly, or impose only technical changes thus ignoring the heterogenous nature of the HLRE

Implications

- requirements fail often due to "insufficient" or incorrect specificity (cf. NASA IFMP) w.r.t. S-P alignments

- specificity impossible to achieve due to bounded rationality, changing environmental conditions, limited resources, problem blossoming or political shifts (goals change)

- the number of requirements for a given problem spaces is non-linearly growing and drastically affected by small changes in the problem space, or fixation of the solution space (requirements paralysis)

Conclusions

- Killer problems: high risk or "impossible" move from the problem space (single or interrelated sets of problems) --> there are also killer requirements

- The process of HLRE involves incremental adaptation of S-P-S mappings over time at different levels of authority (design downwards, negotiate upwards)

- These processes must be aligned with authority structures that are drawn upon in stabilizing the mappings

Future research

- **Future research**
  - examine situations with several principals, conflicting goal sets (political ecology of requirements); how negotiation gets done (interorganizational systems)
  - case studies of how and why specific RE processes succeeded and-or failed with specific configurations and types of software
  - empirical analysis of enacted RE processes to validate and refine the model
  - development of prescriptive framework for the HLRE moves

Literature

- Bergman M, King J., Lyytinen K. (2002): "Large scale requirements analysis revisited: The need for understanding the political ecology of requirements engineering, *Requirements Engineering Journal*

- Bergman M, King J., Lyytinen K., (2002): "Large Scale Requirements Analysis as Heterogeneous Engineering", *Scandinavian Journal of Information Systems,* 14, 1, pp.37-56

- Bergman M., Mark G. (2003): "In situ analysis: a deeper analysis of the relationship between requirements determination and project selection", RE'03, Essen, Germany

- Bergman M., Lyytinen K., Mark G (2004): "Requirements as boundary objects", working paper

©Kalle Lyytinen
2004

48

The Killer Requirements

The death of virtuality