Noema:
A Paradigm

Ron Burback

October, 1997

The Field of Software Engineering

Control	plan repair, re-planning, process changes, plan optimization, chronic problem management, ...
Measure	number of faults both reported and fixed, lines of code, closeness to plan, resource utilization, performance, ...
Strategies	methodologies, architecture, paradigms, mission, risk analysis, scheduling, priority setting, resource utilization, decision making, life cycle management, ...
Tools	compilers, debuggers, environments, quality assurance, CASE, version control, databases, operating systems, networks, file systems, GUI builders, composition, ...
People	group interactions, skill development, group dynamics, communications, goal setting, ...

Areas of Focus

Control	plan repair, re-planning, process changes, plan optimization, chronic problem management, ...
Measure	number of faults both reported and fixed, lines of code, closeness to plan, resource utilization, performance, ...
Strategies	**methodologies**, **architecture**, **paradigms**, mission, risk analysis, scheduling, priority setting, resource utilization, decision making, **life cycle** management, ...
Tools	compilers, debuggers, **environments**, quality assurance, CASE, version control, databases, operating systems, networks, file systems, GUI builders, **composition**, ...
People	group interactions, skill development, group dynamics, communications, goal setting, ...

Paradigm

- Point of view
- A prototype model used as a template to generate particular solutions
- If all you have is a hammer, everything looks like a nail.
- Using the correct paradigm is worth 100 IQ points.
- If a problem is hard it’s because you are looking at it wrong. Change your point of view until the problem becomes easy.

Traditional Science

- Plato (Rationalism)
- Hierarchical in nature.
- To understand a whole, first understand the parts then combine the knowledge into an understanding of the whole.

Problems with Rationalism

- Complete knowledge of sub-components and their actions and interactions.
- Must be able to decompose a complex systems into simpler parts.

Traditional Science Brick Walls

- Godel's incompleteness theorem of mathematics.
- Heisenberg showed that in physics there are no absolutely accurate measurements and that all measurements are only known to with in an uncertainty.
- Einstein showed that velocity can only be known with respect to a relative reference frame.

Hermeneutics Approach

- The components of a whole system are defined as an interpretation in the context of the whole and the environment.
- There is no full and explicit understanding of neither the components nor the whole system.
- The understanding is never complete.

Hermeneutic Circle

- No absolute facts but only interpretations of content within a context.
- Word in the dictionary.
- Like a spinning toy top... An external observer, one outside the toy top, is given the tasks of riding, or understanding, the toy top. His first attempt is to step onto the toy top and is immediately thrown off. To be successful, first the observer must gain momentum, and match the motion of the toy top, and then, step onto the toy top. One can't understand the hermeneutic circle without first understanding the whole

Noema

- Edmund Husserl (1850) called the Hermeneutic circle paradigm a Noema.
- Noema is an antiquated Greek word for an intellect.
- The understanding of a part is a projection onto the whole. First you understand the whole, then you understand the parts.

Defining Characteristics (Hermeneutic)

- The implicit beliefs within a Noema and assumptions cannot all be made explicit.
- Practical operational understanding of a Noema is more fundamental than detached theoretical understanding.
- A representation of a thing cannot be complete.
- Understanding is fundamentally in the context of the whole and cannot be reduced to activities of individual sub-components.
- A sub-component cannot avoid its interactions with the whole.
- The effects of the sub-components cannot be absolutely predicted.
- All representations of the current state are ephemeral at best.
- Every representation of a sub-component is an interpretation with respect to the whole. Every action of the sub-component affects the whole, even non action.
- The presence of the sub-component affects the whole.

Noemi

- Life.
- The web.
- Free market economy.
- My contention is that a Noema is the correct paradigm for the software engineering of large distributed systems.

Defining Characteristics (Distributed System)

- Knowledge
  - Partial Knowledge
  - Local Knowledge
  - Smart Components
- Change
  - Constant Change
  - Elaboration Tolerance
  - Constant Death and Replacement
  - No Down Time
  - Periods of Rest

Defining Characteristics (Distributed System)

- Communication
  - Global Communication
  - Asynchronous Communication
  - Languages
- Organization
  - Distributed
  - Replication and Groups
  - Massive Interdependency
  - Self Fixing and Self Regulating
  - No Master Control
  - Conscience and Unconscious Actions
  - No System-Wide Clock
  - No Global State
  - Catalyst
  - Common Building Blocks

Defining Characteristics (Distributed System)

- Security
  - A Security (Immune) System
  - Unique Name Space
  - Authentication and Authorization
- Persistence
- Boundaries
  - Environment
  - Recycling System
  - Peripherals

Noemic Engineering

- Small changes. Simple components.
- Evolution.
- Clone from the old.
- Assume that the whole already exists.
- Push off calculations to another component.
- Let the Noema environment do the integration.
- Assume that the environment already exists.

Theory Support

- A Noema is hard to build but easy to change.
- The computational power of a Noema is the same as a type 1 context-sensitive grammar and represented by a linearly bounded Turing machine

Example: Networking

Network - Today

FTP SMTP TELNET ... Others
Transport - TCP/UDP
Network - IP
Physical Link - MAC

Network - Noema

FTP SMTP TELNET ... Others
PTP
Transport - TCP/UDP
Network - IP
Physical Link - MAC

Simplified PTP packet

Encrypted Data	From Principal	To Principal

PTP layer example