N degrees of Separation:
Multi-Dimensional Separation of Concerns

Peri Tarr, Harold Ossher, William Harrison
IBM Watson Research Center
ICSE, 1999

Stanley M. Sutton Jr.
EC Cubed, Inc.

Presented by David Gonzalez
Key Insights (1/2)

• A dominant decomposition (features, objects) based on separation of concerns forces structuring software by one dimension of separation at a time, resulting in sparse functionalities if viewed by another concern.

• Provides a model for simultaneous, non-invasive multi-dimensional decomposition, hyperslicing, and composition of artifacts using hypermodules, while avoiding the need for new software formalisms.
Key Insights(2/2)

• A hyperslice is a set of conventional modules that encapsulate a non-dominant concern.

• A hypermodule is a set of hyperslices that obey a given rule.

• Considered instances of the model: subject-oriented and aspect-oriented programming, contract-base composition, role models, adaptive programming, etc.
Problem

Concerns:
Features in requirements
Objects in design
Interface and Implementation classes in code

Sparse functionality due to the “tyranny of dominant concern”:
**Scattering** – single requirement affects multiple artifacts
**Tangling** – multiple requirements affect an artifact
Approach

Goals:
low impact of changes
More Reuse
More Traceability

Steps:
1. Identify concerns (feature, unit of change, customization, data or object).
2. Define primitive and compound units within artifacts.
3. Decompose into hyperslices.
Questions?

- Does hyperslicing reflect what its name mean?
- Besides analytical process, what proof do they present to offer to evaluate the model?
- Is a hyperslice useful?
- Method overloading, polymorphisms. Where do they fit?