Carnegie Mellon University

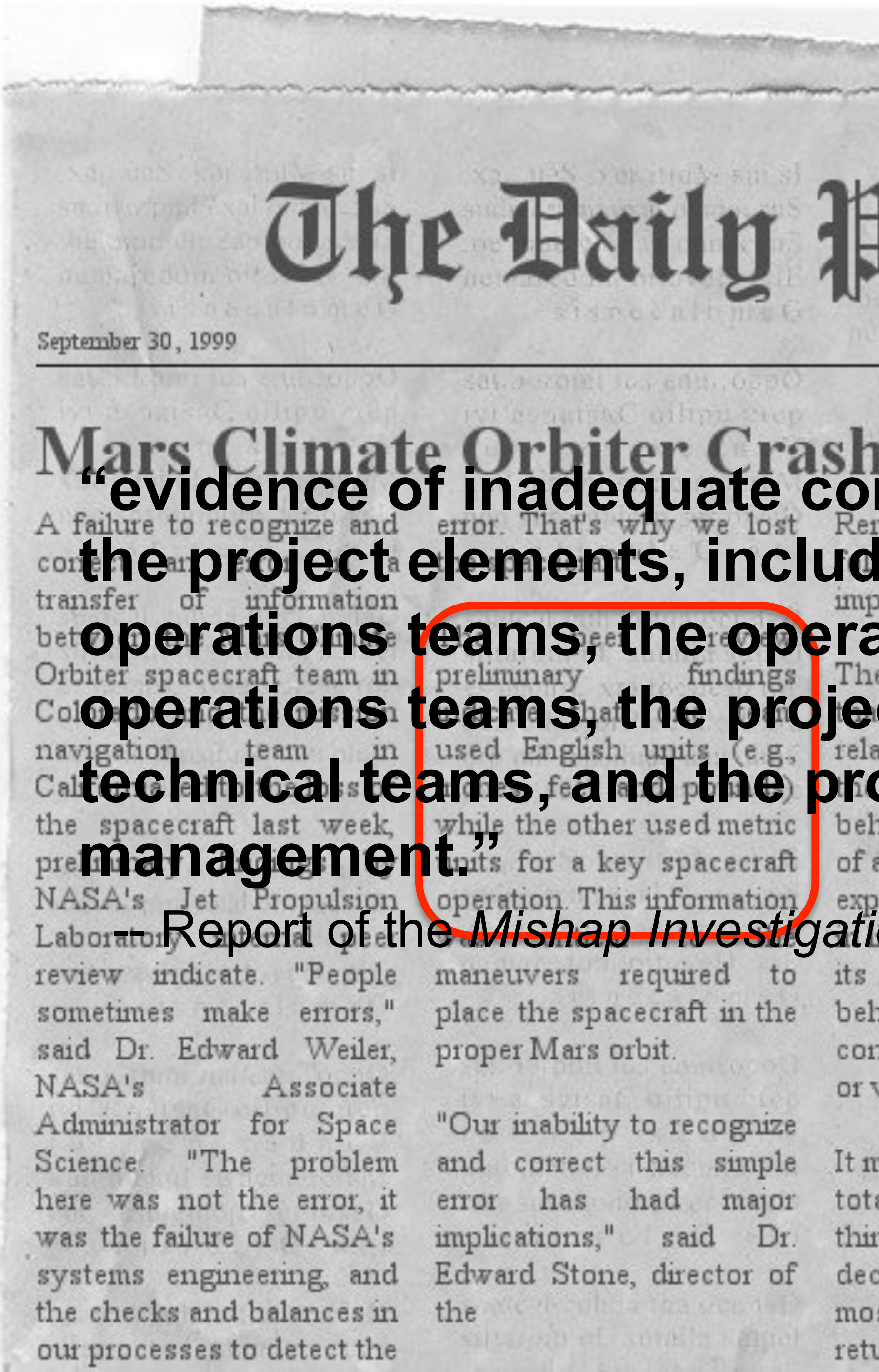
institute for SOFTWARE RESEARCH

Socio-Technical Coordination

James Herbsleb

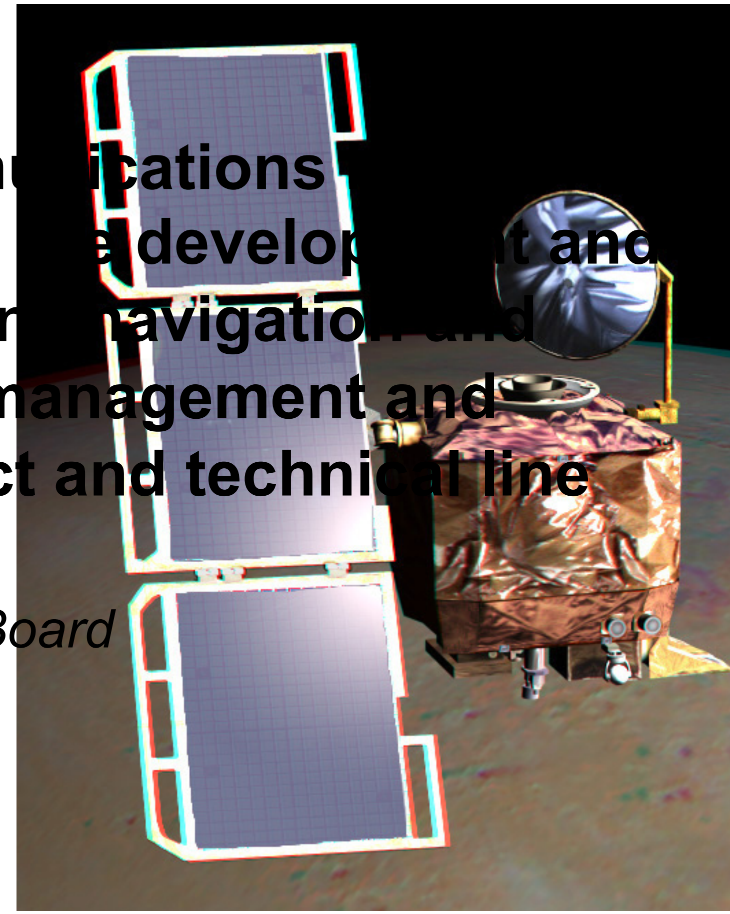
jdh@cs.cmu.edu

Mars Climate Orbiter

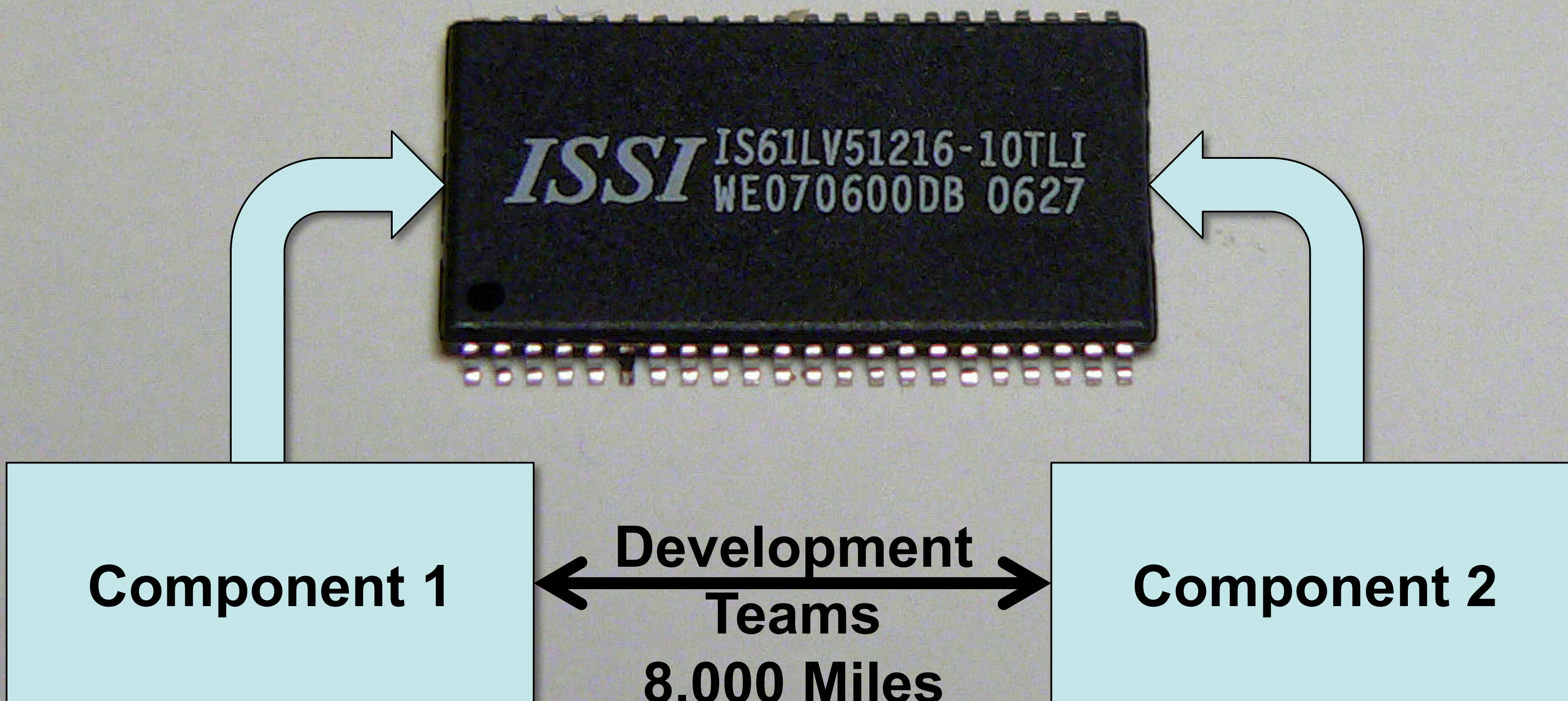


"evidence of inadequate communications the project elements, including the development and operations teams, the operations navigation and operations teams, the project management and technical teams, and the project and technical line management."

-- Report of the *Mishap Investigation Board*



Shared Memory



Coordination Failures

Managing dependencies among tasks.

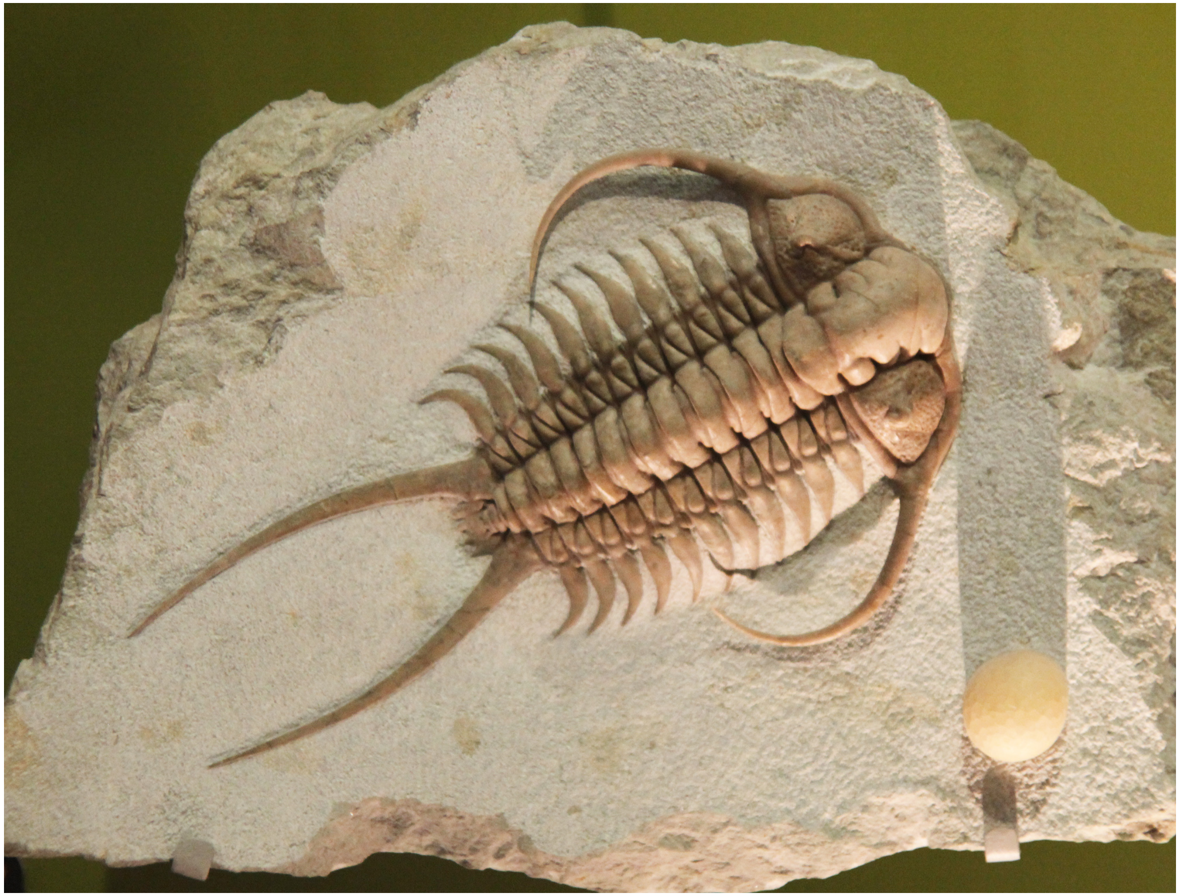
-- Malone and Crowston (1994)

A Brief, Selective, and Biased History of Coordination

How do we coordinate work in software engineering?

We've applied an astonishing variety of techniques.

Paleozoic Era



Coordination and its Discontents

Coordination Technique of Choice of a Previous Generation

“The ultimate method for managing . . . activity with a small group of 10 or 20 people is 10 hours of meetings a day. And then you go work 5 hours.”

Curtis, B., Krasner, H. and Iscoe, N. A field study of the software design process for large systems. *Communications of the ACM.*, 31, 1988, 1268-1287.

What the Data Showed

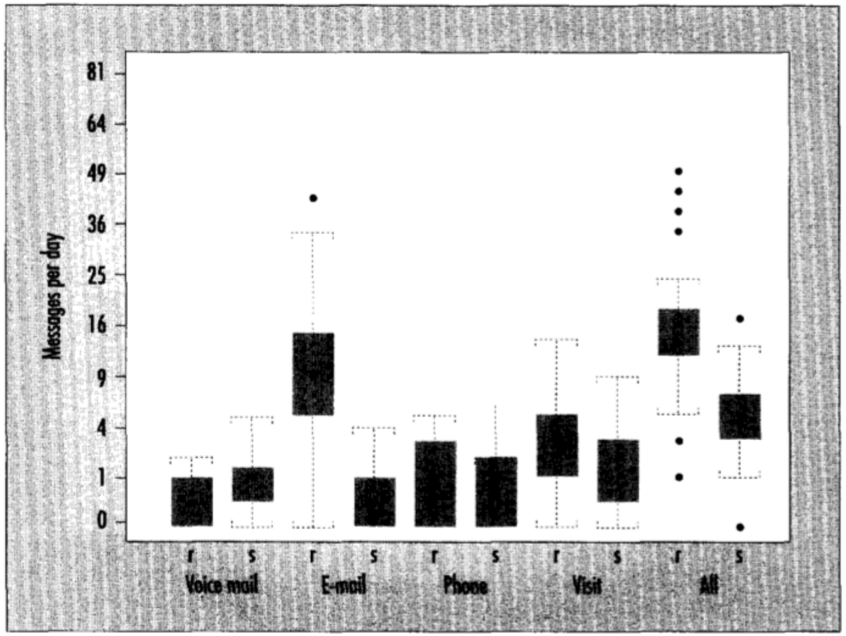


Figure 7. Messages sent and received across four media types. The figure shows the number of messages sent and received, by media type and according to whether they were received (r) or initiated (s). We applied a square-root transformation to stabilize the variance. Each box contains data on all seven study subjects across five days of observation per subject.

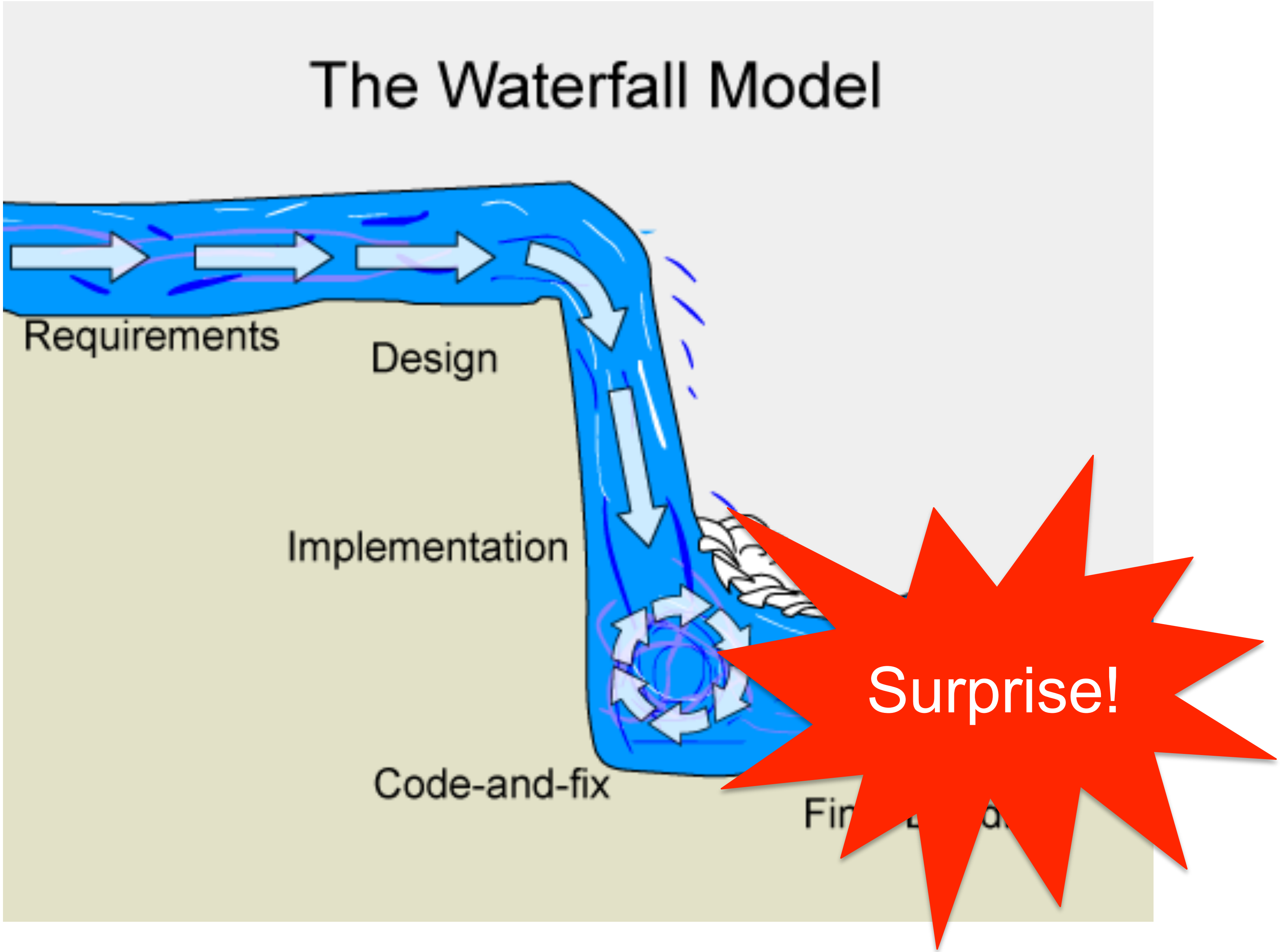
"There are seven unique personal contacts per day on average, representing continuing interactions"

"Direct observation showed us that developers spend about ***75*** minutes per day in unplanned interpersonal interactions."

Perry, D. E., Staudenmayer, N. A. and Votta, L. G. People, Organizations, and Process Improvement. *IEEE Software*, 11, 1994, 36-45.

Not Just Meetings . . .

There are also methods to the madness


The Waterfall Model
Requirements
Design
Implementation
Code-and-fix
Surprise!

Not Just Methods . . .

Processes!

Software Process

SOFTWARE PROCESSES ARE SOFTWARE TOO

Leon Osterweil

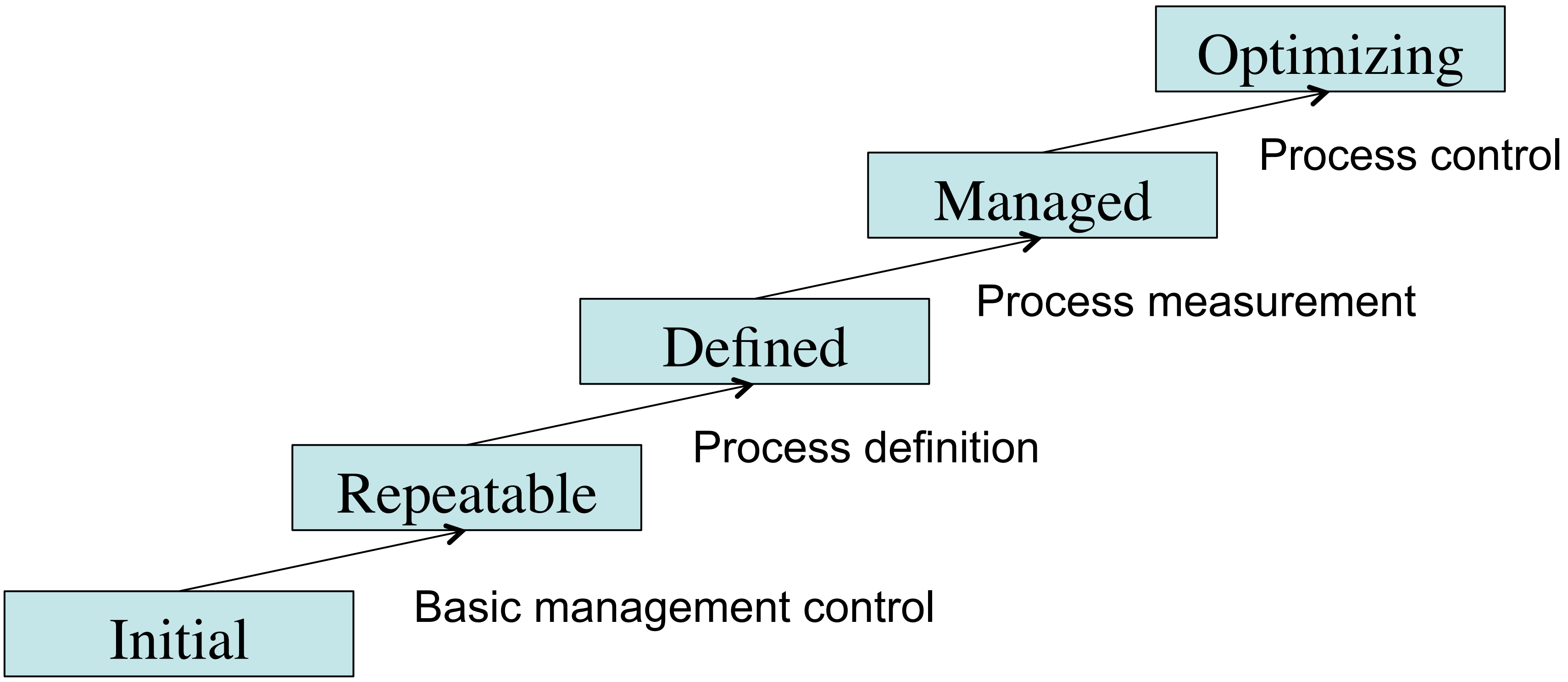
University of Colorado Boulder, Colorado USA

1. The Nature of Process.

The major theme of this meeting is the exploration of the importance of .ul process as a vehicle for improving both the quality of software products and the the way in which we

description defines a class or set of objects related to each other by virtue of the fact that they are all activities which follow the dictated behavior. We shall have reason to return to this point later in this presentation.
For now we should return to our consideration of the intui-

Maturity Framework 1988

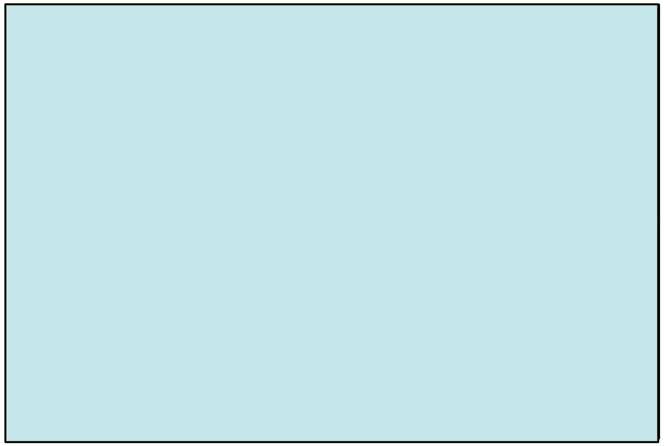


*Humphrey, W. S. (1988). Characterizing the software process: a maturity framework. *Software, IEEE, 5*(2), 73-79.

Not Just Process . . .

- Don’t forget product structure!

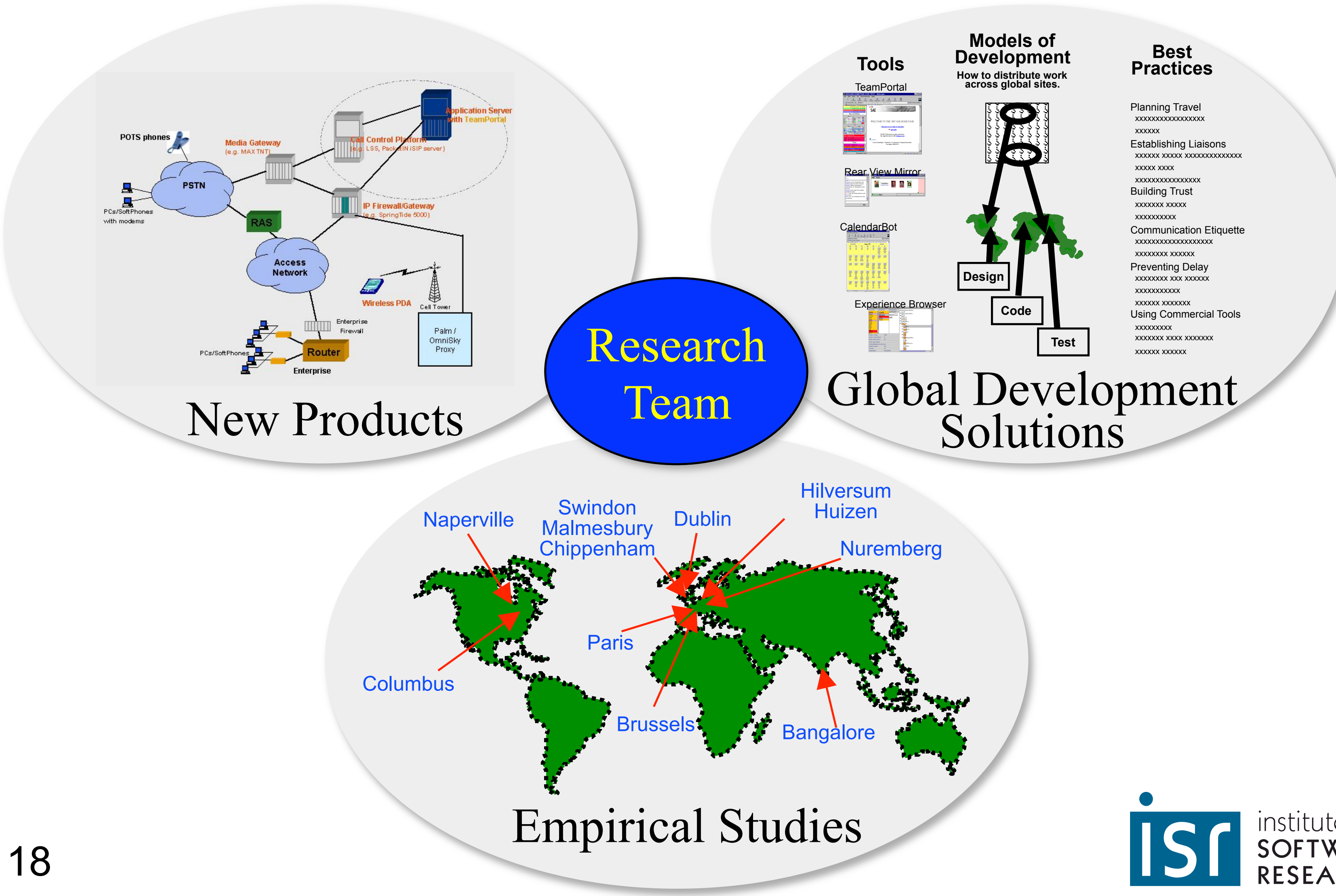
Modularity and information hiding



Parnas, D. L. On the Criteria to be Used in Decomposing Systems into Modules. *Communications of the ACM*, 15, 12 1972, 1053-1058.

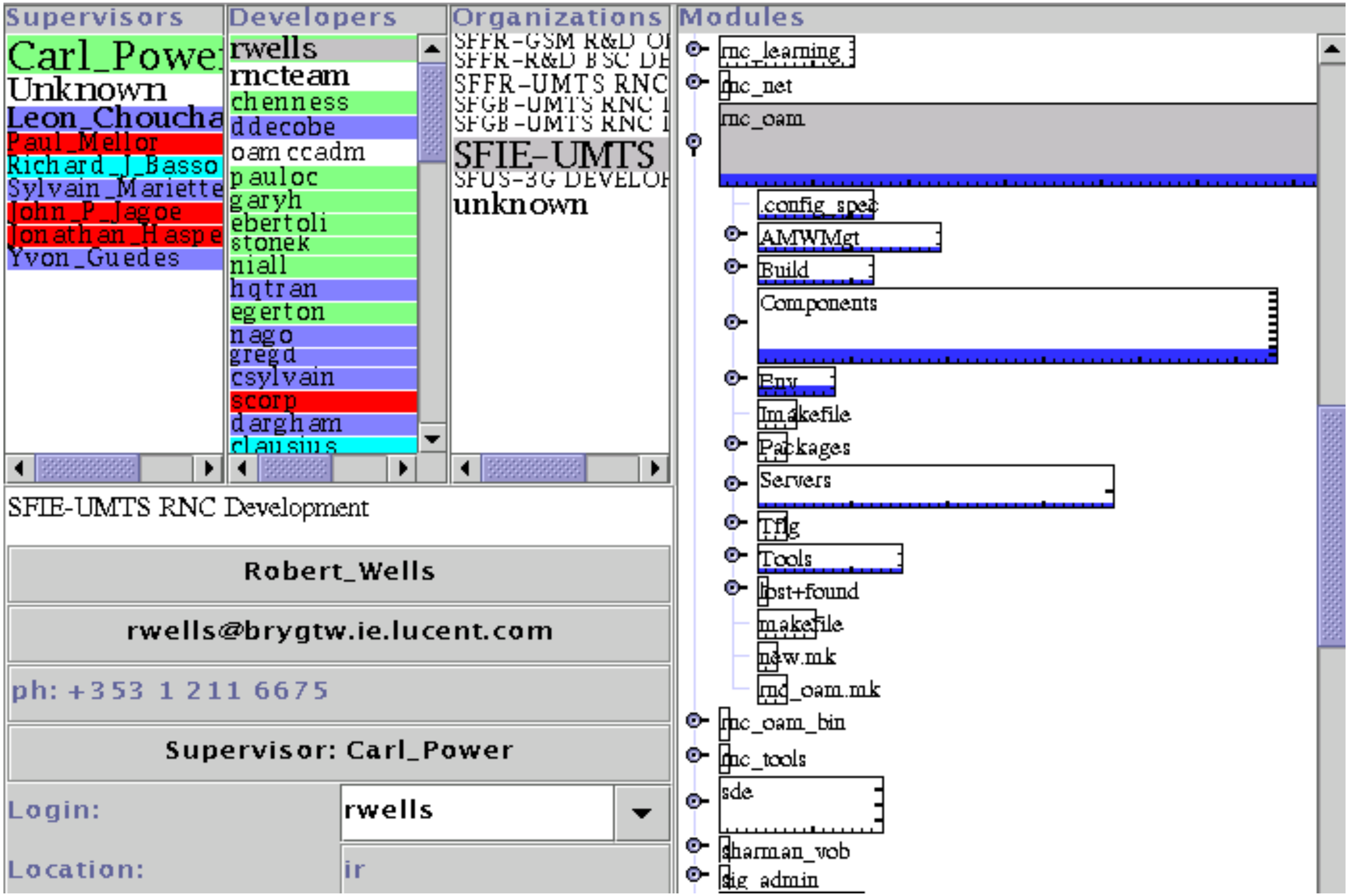
Mesozoic Era

Bell Labs Collaboratory



institute for SOFTWARE RESEARCH

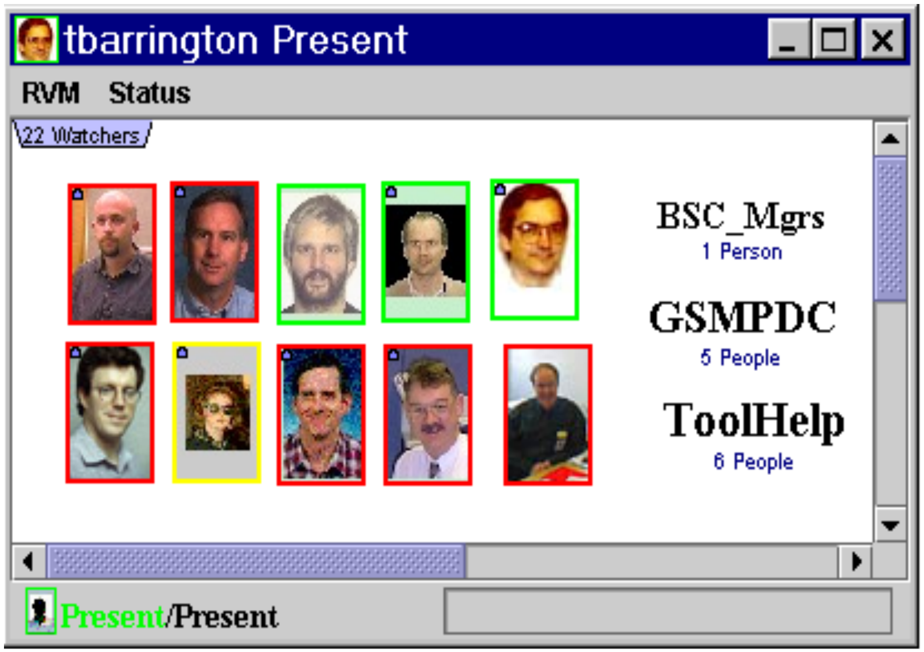
Expertise Browser



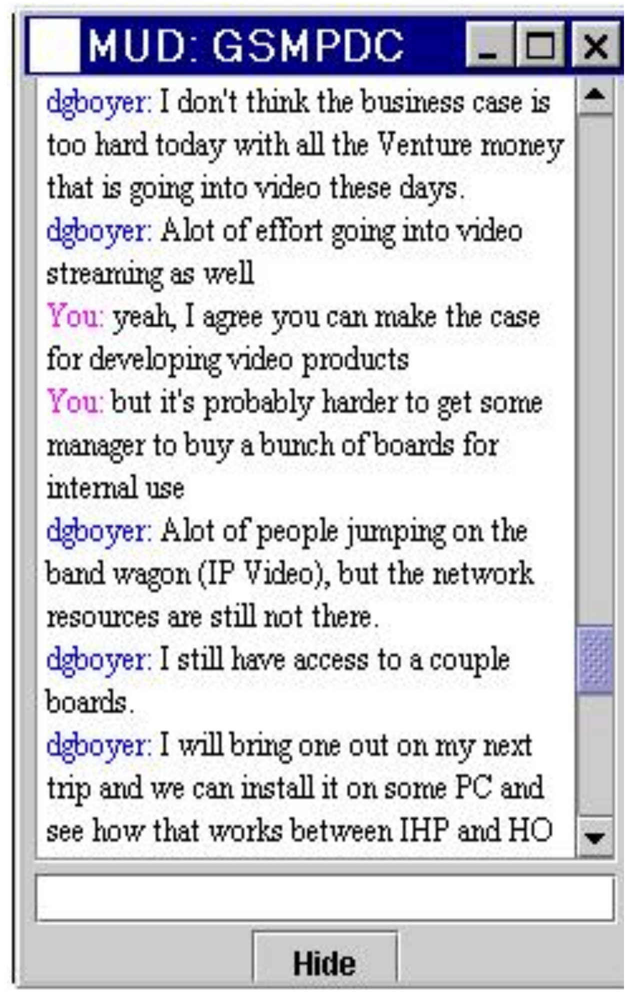
Mockus, A., & Herbsleb, J.D. (2002). Expertise Browser: A quantitative approach to identifying expertise. In Proceedings of *International Conference on Software Engineering*, Orlando, FL, May 19-25, pp. 503-512.

Instant Messaging

Rear View Mirror



Presence Viewer

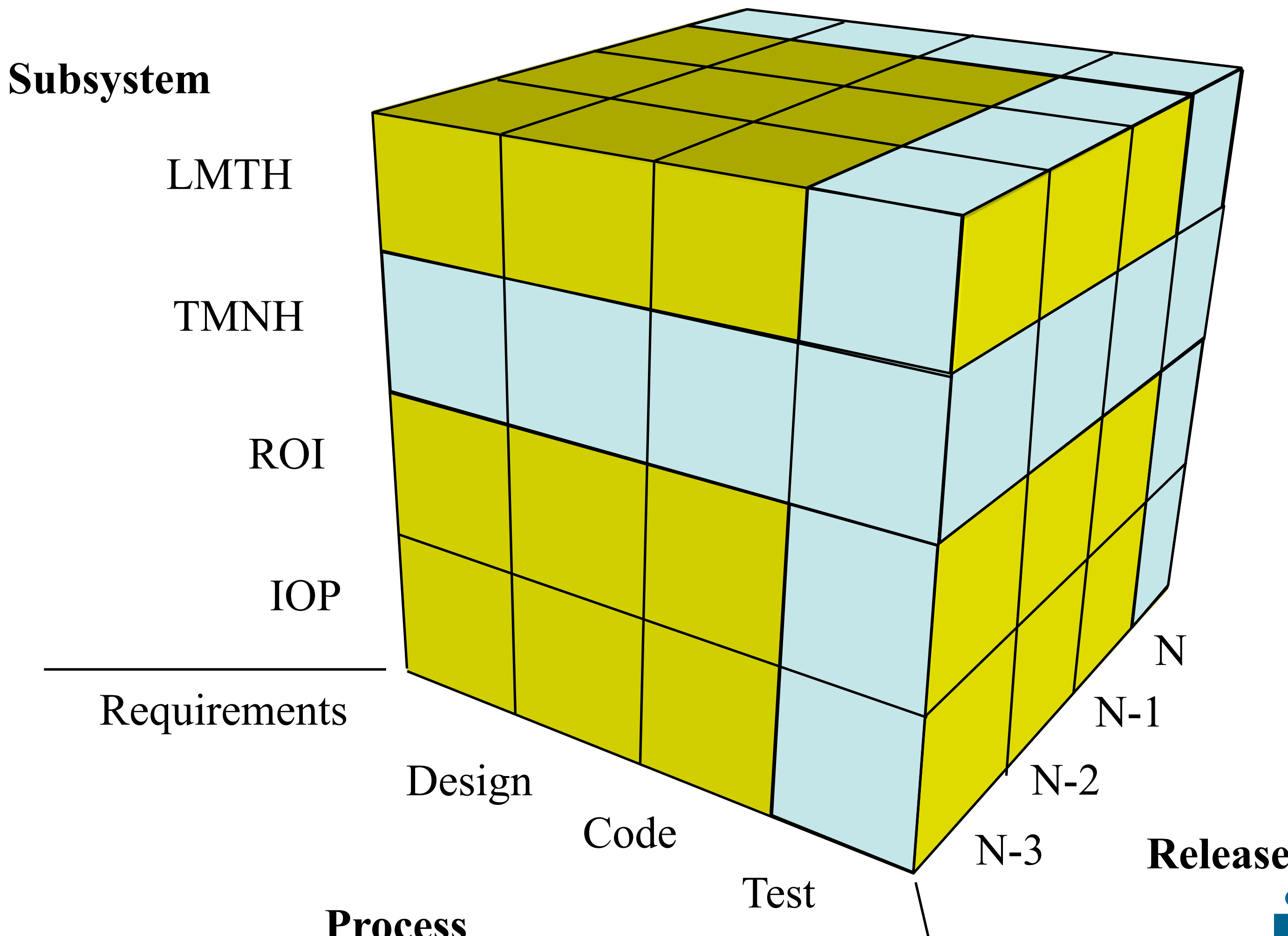


Group Chat

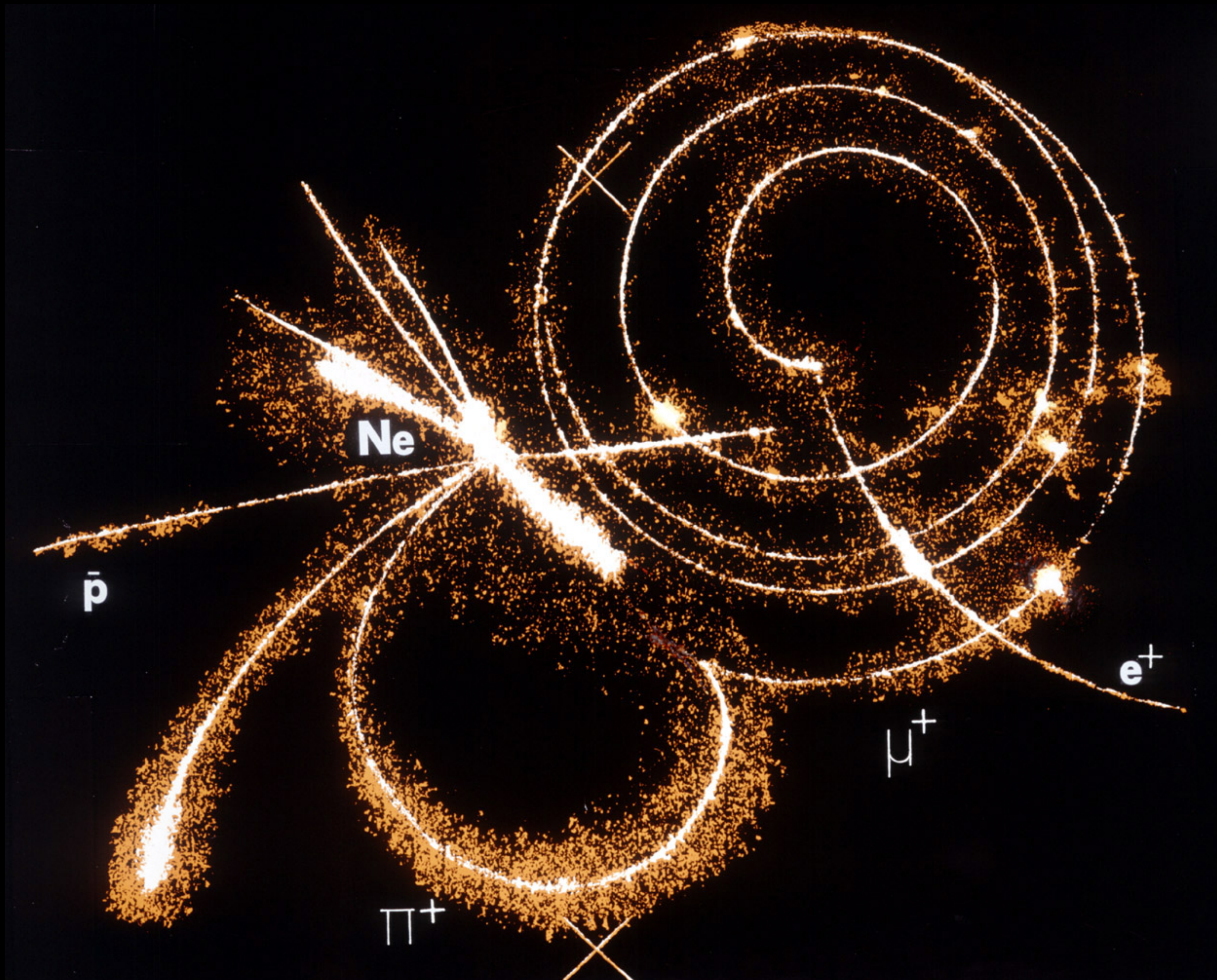
Handel, M. & Herbsleb, J.D. (2002). What is Chat doing in the workplace? Proceedings of ACM *Conference on Computer-Supported Cooperative Work* (CSCW), New Orleans, LA, pp. 1-10.

Herbsleb, J.D., Atkins, D.L., Boyer, D.G., Handel, M., & Finholt, T.A. (2002). Introducing Instant Messaging and Chat into the workplace. In Proceedings of ACM *Conference on Computer-Human Interaction*, Minneapolis, MN, April 20-25, pp. 171-178.

Organizational Models



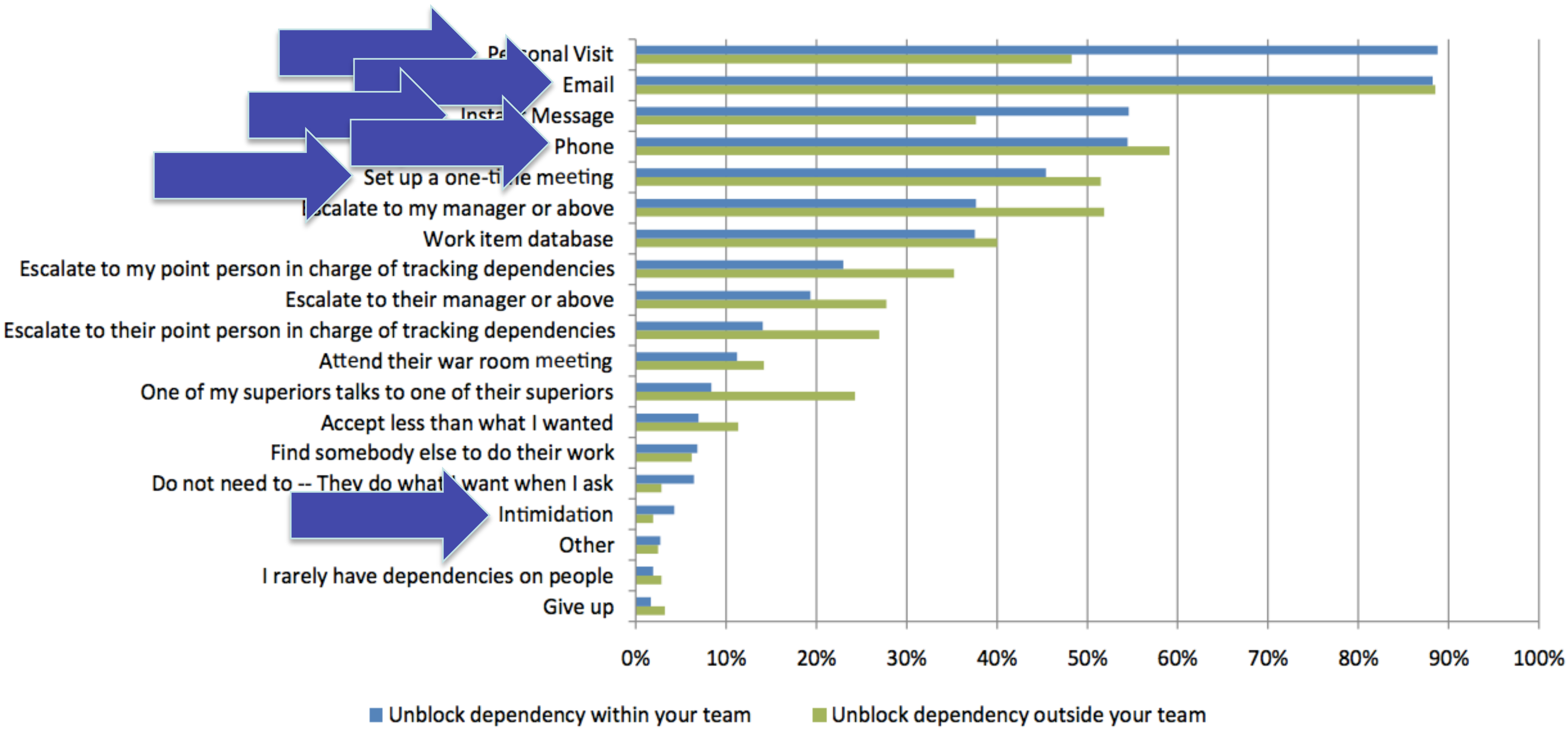
Grinter, R. E., Herbsleb, J. D. and Perry, D. E. The Geography of Coordination: Dealing with Distance in R&D Work. In *Proceedings of GROUP '99* (Phoenix, AZ, November 14-17, 1999).



Ne
p̄
e+
μ+
π+

Cenozoic Era

Meeting Innovation



Begel, A., Nagappan, N., Poile, C., Layman, L. (2009). Coordination in Large-Scale Software Teams. *CHASE*, pp. 1-7.

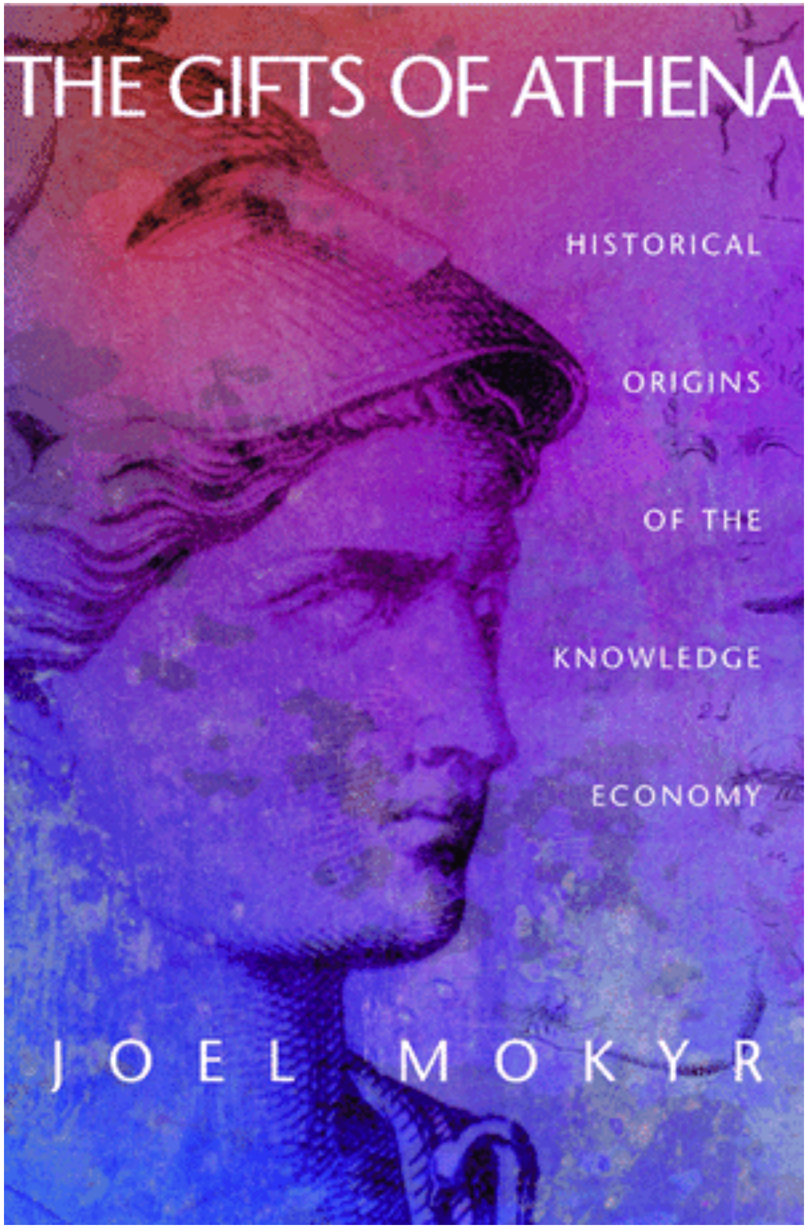
Today

launchpad

So Many Techniques . . .

“Sometimes the magic works and sometimes it doesn’t.”
− *Little Big Man*

This is the history of technology and the evolution of useful knowledge.


THE GIFTS OF ATHENA
HISTORICAL
ORIGINS
OF THE
KNOWLEDGE
ECONOMY
JOEL MOKYR

History of Useful Knowledge

Technique

- Make iron (from 2000 BC)
  - Mix ore, charcoal
  - Apply heat
  - Pour when ready

Epistemic Base

- Metallurgy
  - Eliminate phosphorus
  - Add carbon at right time
  - Reduce oxygen
  - Siemens Martin process (1865)

History of Useful Knowledge

Technique

- Analgesic (1763)
  - Ingest willow bark
  - Pain relief
  - Side effects

Epistemic Base

- Chemistry
  - Salicin
  - Explore related compounds
  - Salicylic acid (1835)

Future Useful Knowledge

Techniques

- Meetings
- Communication tools
- Processes
- Tweets
- Etc.

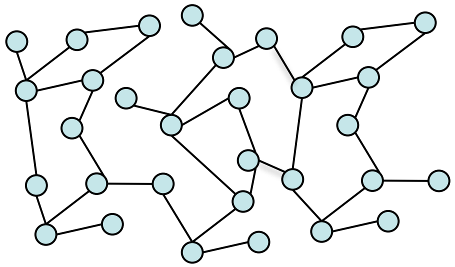
Epistemic Base

- Theory of Coordination

Where to Start?

- Product modularity, task modularity, mirroring
  - Baldwin (2000); Conway (1968); Parnas (1974); Sosa & Eppinger (2004); Colfer & Baldwin (2010)
- Collaboration over distance
  - Olson & Olson (2000); Olson, Malone, & Smith (2001); Olson & Teasley (1996)
- Implicit and explicit coordination
  - Kraut & Streeter (1995); Espinosa, Lerch, Kraut (2004)
- Interdisciplinary theory of coordination
  - Malone and Crowston (1994)
- Social network analysis
  - Krackhardt & Carley (1998)

Socio-Technical Coordination



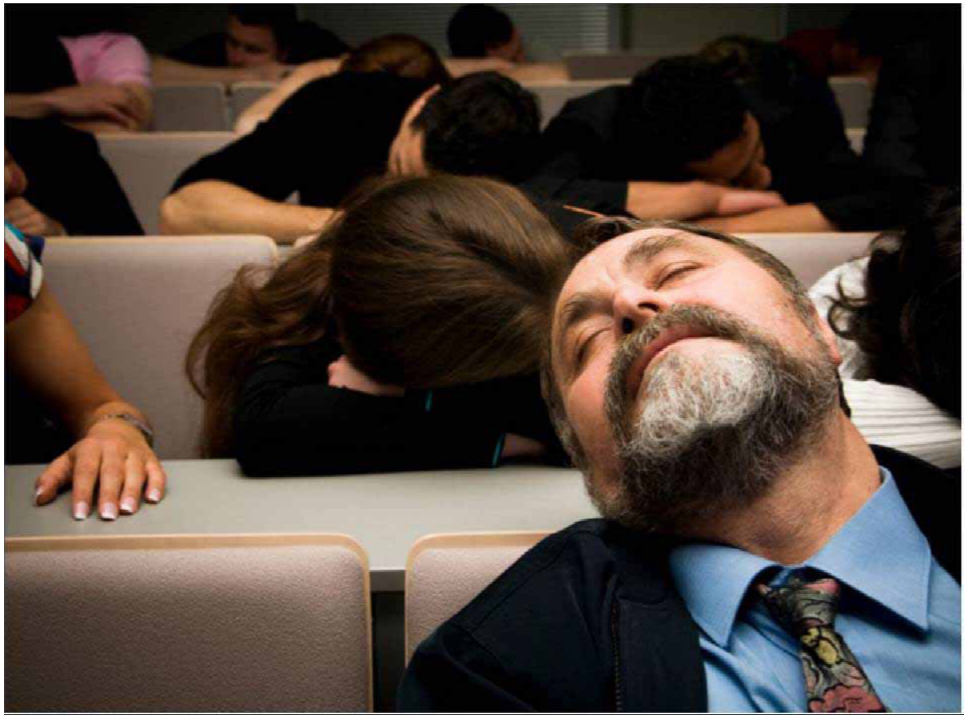
Decisions and Constraints

Network of decisions . . .

. . . establishes a coordination problem . . .

. . . that the organization must solve.

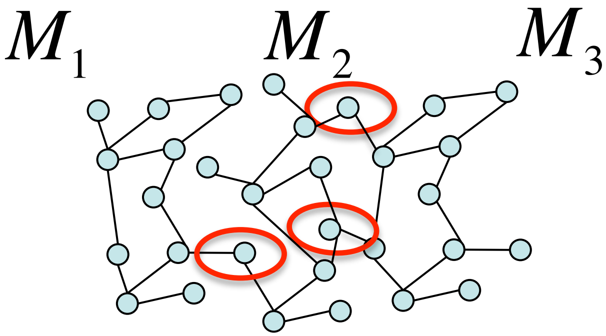
Expressed More Formally . . .



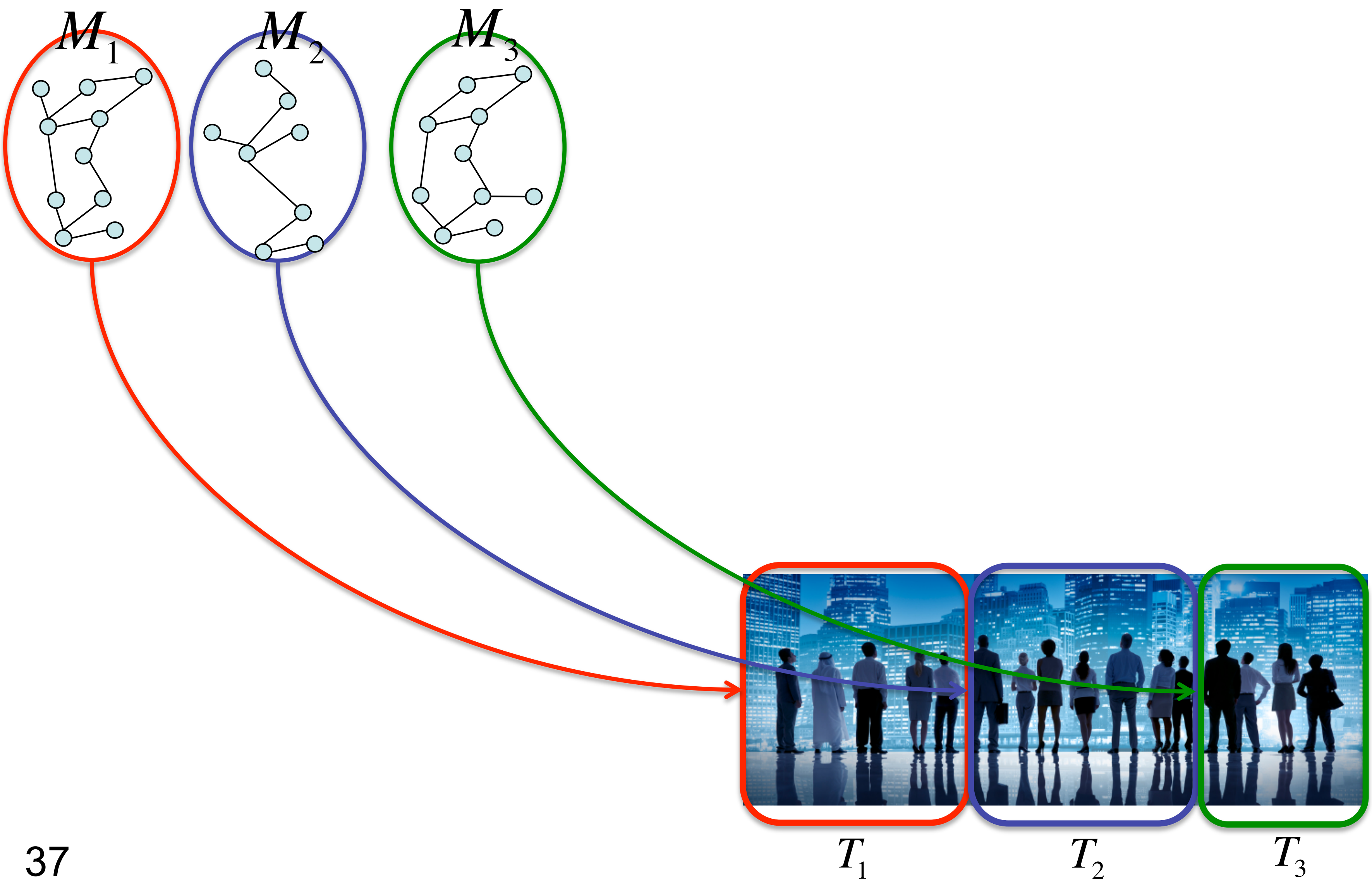
Herbsleb, J.D., & Mockus, A. (2003). Formulation and preliminary test of an empirical theory of coordination in software engineering. In Proceedings, *ACM SIGSOFT Symposium on the Foundations of Software Engineerin*g, Helsinki, Finland, September 1-5, pp. 112-121

Herbsleb, J.D., Mockus, A., Roberts, J.A. (2006). Collaboration in Software Engineering Projects: A Theory of Coordination*. International Conference on Information Systems*, Milwaukee, WI.

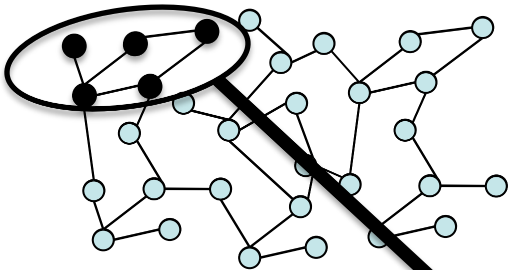
Example: Modularity and Teams



Example: Modularity and Teams



Socio-Technical Coordination



Decisions and Constraints

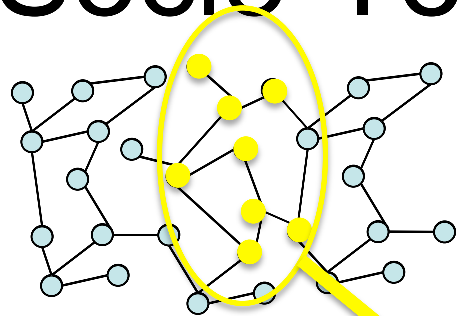
Network of decisions . . .

. . . establishes a coordination problem . . .

Geography

. . . that the organization
must solve.

Socio-Technical Coordination



Decisions and Constraints

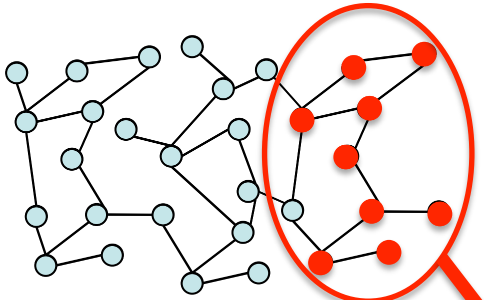
Network of decisions . . .

. . . establishes a coordination problem . . .

Prior work history

. . . that the organization must solve.

Socio-Technical Coordination

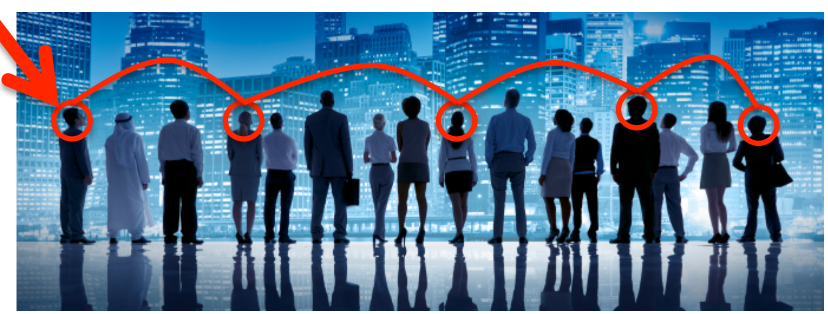


Decisions and Constraints

Network of decisions . . .

. . . establishes a coordination problem . . .

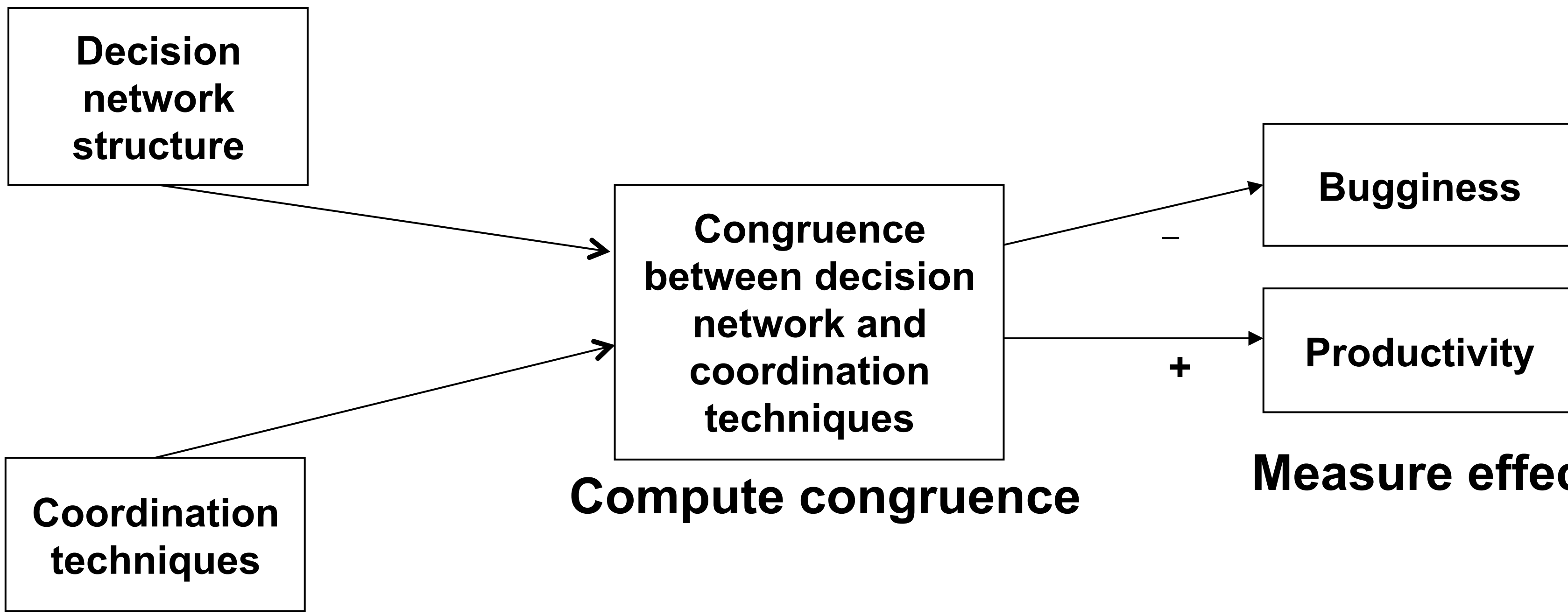
Create communication channels



. . . that the organization must solve.

Research Challenges

Measure structure of network

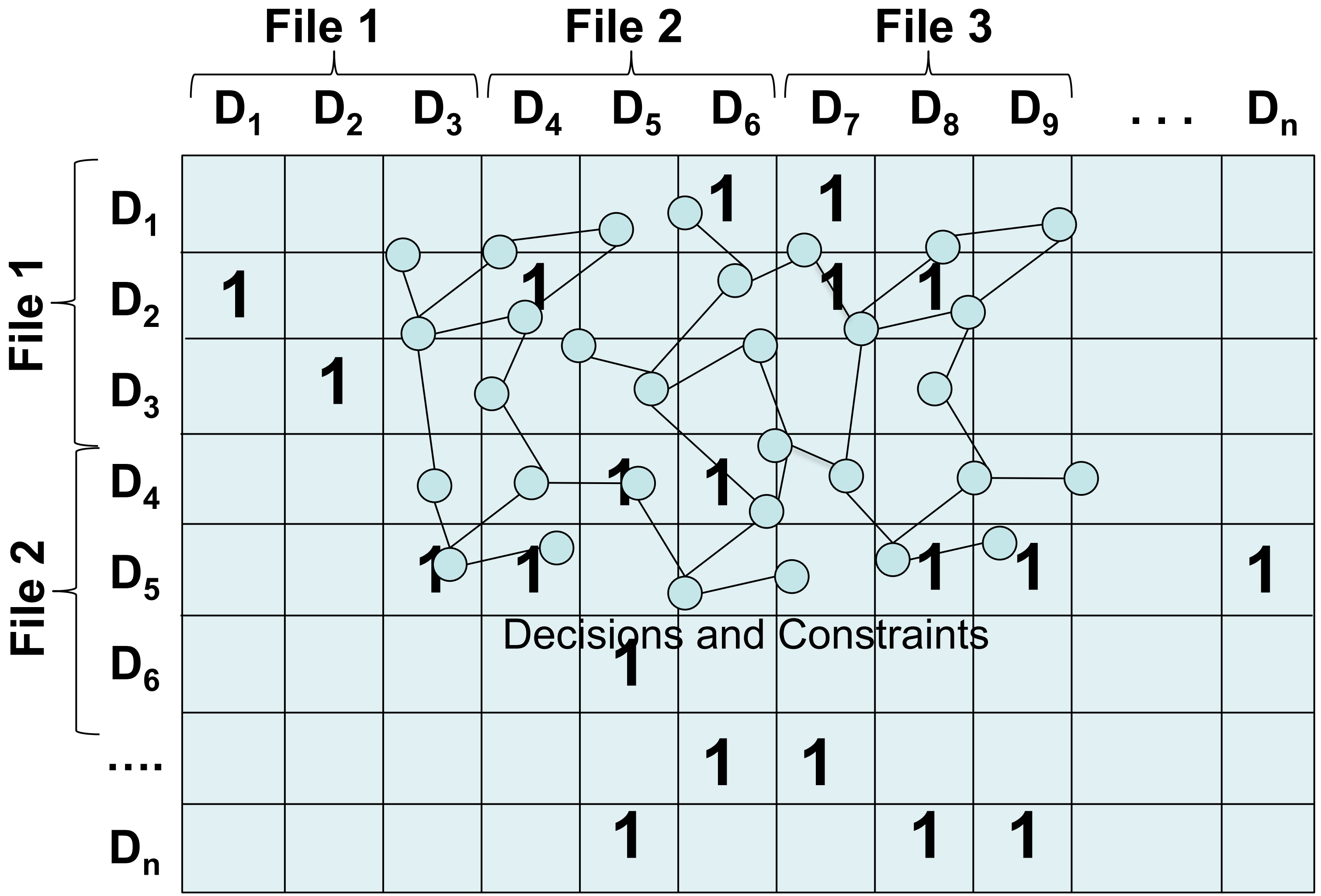


Compute congruence

Measure effects

Measure coordination techniques

Decision Constraint Matrix (DC)



Cataldo, M., Wagstrom, P., Herbsleb, J.D., Carley, K. (2006). Identification of coordination requirements: Implications for the design of collaboration and awareness tools. In Proceedings, *ACM Conference on Computer-Supported Cooperative Work, Banff Canada,* pp. 353-362.

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	. . .	F_n
F_1						1	1				
F_2	1			1			1	1			
F_3		1									
F_4					1	1					
F_5			1	1				1	1		1
F_6					1						
....						1	1				
F_n					1			1	1		

Decision Assignment Matrix (DA)

	File 1			File 2			File 3				
	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	. . .	D_n
	1					1	1				
		1									
			1	1							1
					1			1	1		

Cataldo, M., Wagstrom, P., Herbsleb, J.D., Carley, K. (2006). Identification of coordination requirements: Implications for the design of collaboration and awareness tools. In Proceedings, *ACM Conference on Computer-Supported Cooperative Work, Banff Canada,* pp. 353-362.

Carnegie Mellon University

Decision Assignment Matrix (DA)

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	. . .	F_n
	1					1	1				
		1									
			1	1							1
					1			1	1		

Cataldo, M., Wagstrom, P., Herbsleb, J.D., Carley, K. (2006). Identification of coordination requirements: Implications for the design of collaboration and awareness tools. In Proceedings, *ACM Conference on Computer-Supported Cooperative Work, Banff Canada,* pp. 353-362.

isr institute for SOFTWARE RESEARCH

DA

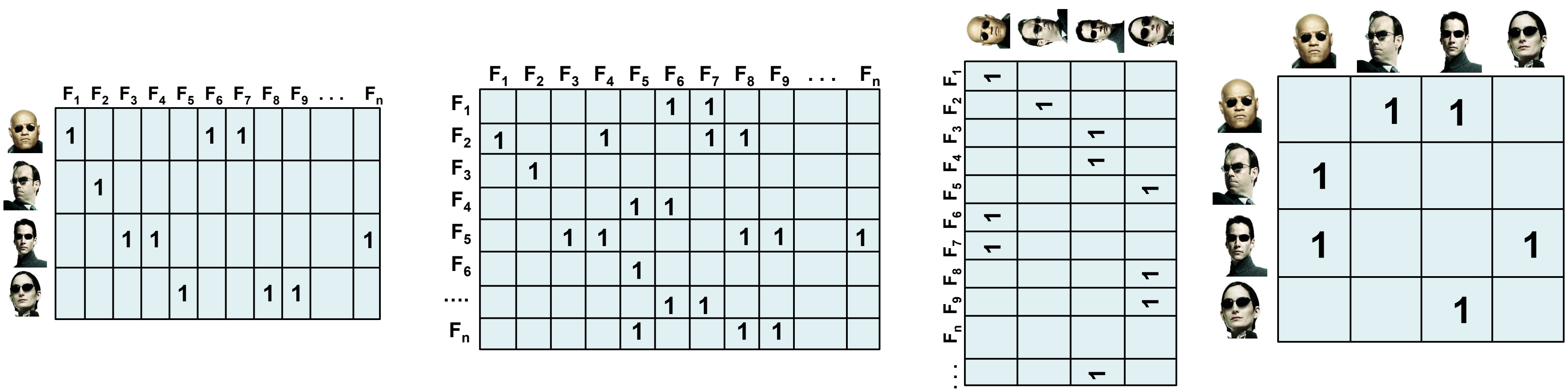
	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	. . .	F_n
	1					1	1				
		1									
			1	1							1
					1			1	1		

	F_1	F_2	F_3	F_4
	1			
		1		
			1	1

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	. . .	F_n
F_1						1	1				
F_2	1			1			1	1			
F_3		1									
F_4					1	1					
F_5			1	1				1	1		1
F_6					1						
....						1	1				
F_n					1			1	1		

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	. . .	F_n
	1					1	1				
		1									
			1	1							1
					1			1	1		

$$DA \quad \times \quad DC \quad \times \quad DA^{T} \quad = \quad C_R$$



C_R Coordination Requirements

Coordination Activities and Congruence

Coordination Requirements (C_R)


	1	1	
1			
1			1
		1	

Actual Coordination (C_A)


	1	1	
1			
1			1
		1	

- Team structure
- Geographic location
- Use of chat
- On-line discussion

Congruence **= proportion of nonzero cells in C_R that are also nonzero in C_A**

Cataldo, M., Wagstrom, P., Herbsleb, J.D., Carley, K. (2006). Identification of coordination requirements: Implications for the design of collaboration and awareness tools. In Proceedings, *ACM Conference on Computer-Supported Cooperative Work, Banff Canada,* pp. 353-362.

Impact on Productivity

Table 2: Results from OLS Regression of Effects on Task Performance ($^{+}$ p < 0.10, * p < 0.05, ** p < 0.01).

	Model I	Model II	Model III	Model IV
(Intercept)	2.987**	3.631**	1.572*	1.751*
Dependency	0.897*	0.653*	0.784*	0.712*
Priority	-0.741*	-0.681*	-0.702*	-0.712*
Re-assignment	0.423*	0.487*	0.304*	0.324*
Customer MR	-0.730	-0.821	-0.932	-0.903
Release	-0.154*	-0.137*	-0.109*	-0.098*
Change Size (log)	1.542*	1.591*	1.428*	1.692*
Team Load	0.307*	0.317*	0.356*	0.374*
Programming Experience	-0.062*	-0.162*	-0.117*	-0.103*
Tenure	-0.269*	-0.265*	-0.239*	-0.248*
Component Experience (log)	-0.143*	-0.143*	-0.195*	-0.213*
Structural Congruence		**-0.526***		**-0.483***
Geographical Congruence		**-0.317***		**-0.312***
MR Congruence		**-0.189***		**-0.129***
IRC Congruence		**-0.196***		**--**
Interaction: ReleaseX Structural Congruence		**0.007**		**0.009**
Interaction:ReleaseXGeographical Congruence		**-0.013**		**-0.017**
Interaction: Release X MR Congruence		**-0.009+**		**-0.011+**
Interaction: Release X IRC Congruence		**-0.017***		**--**
N	809	809	1983	1983
Adjusted R^2	0.787	0.872	0.756	0.854

Cataldo, M., Wagstrom, P., Herbsleb, J.D., Carley, K. (2006). Identification of coordination requirements: Implications for the design of collaboration and awareness tools. In Proceedings, *ACM Conference on Computer-Supported Cooperative Work, Banff Canada,* pp. 353-362.

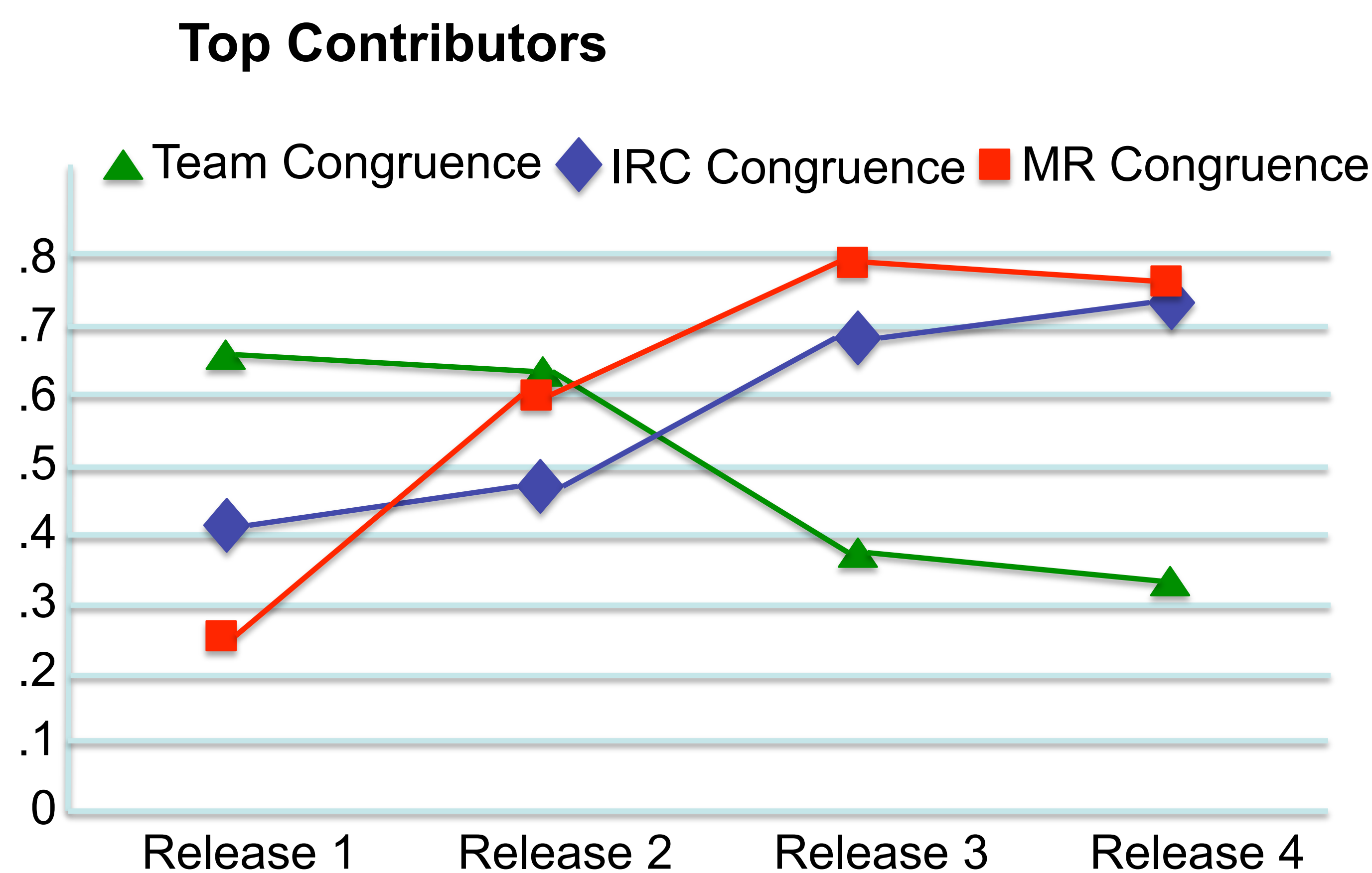
Impact on Bugginess

	Model I	Model II
LOC (log)	1.125**	1.136**
Avg. Lines Changed (log)	1.128**	1.121**
Number Logical Dep. (log)	2.219**	2.109**
Clustering Logical Dep. (log)	0.012**	0.012**
Coordination Req. Dep. (log)	2.187**	1.962**
Structural Congruence		0.281*
Geographical Congruence		0.317
MR Congruence		0.209**
IRC Congruence		0.271**
Model Fit		
N	3980	3980
Model χ2	1663**	1859**
df	5	9
Deviance Explained	0.302	0.335
Model Comparison χ2	--	196.24**

(+ $p < 0.10$; * $p < 0.05$; ** $p < 0.01$)

Cataldo, M., & Herbsleb, J. D. (2013). Coordination Breakdowns and Their Impact on Development Productivity and Software Failures. *IEEE Transactions on Software Engineering* 39(3), 343-360.

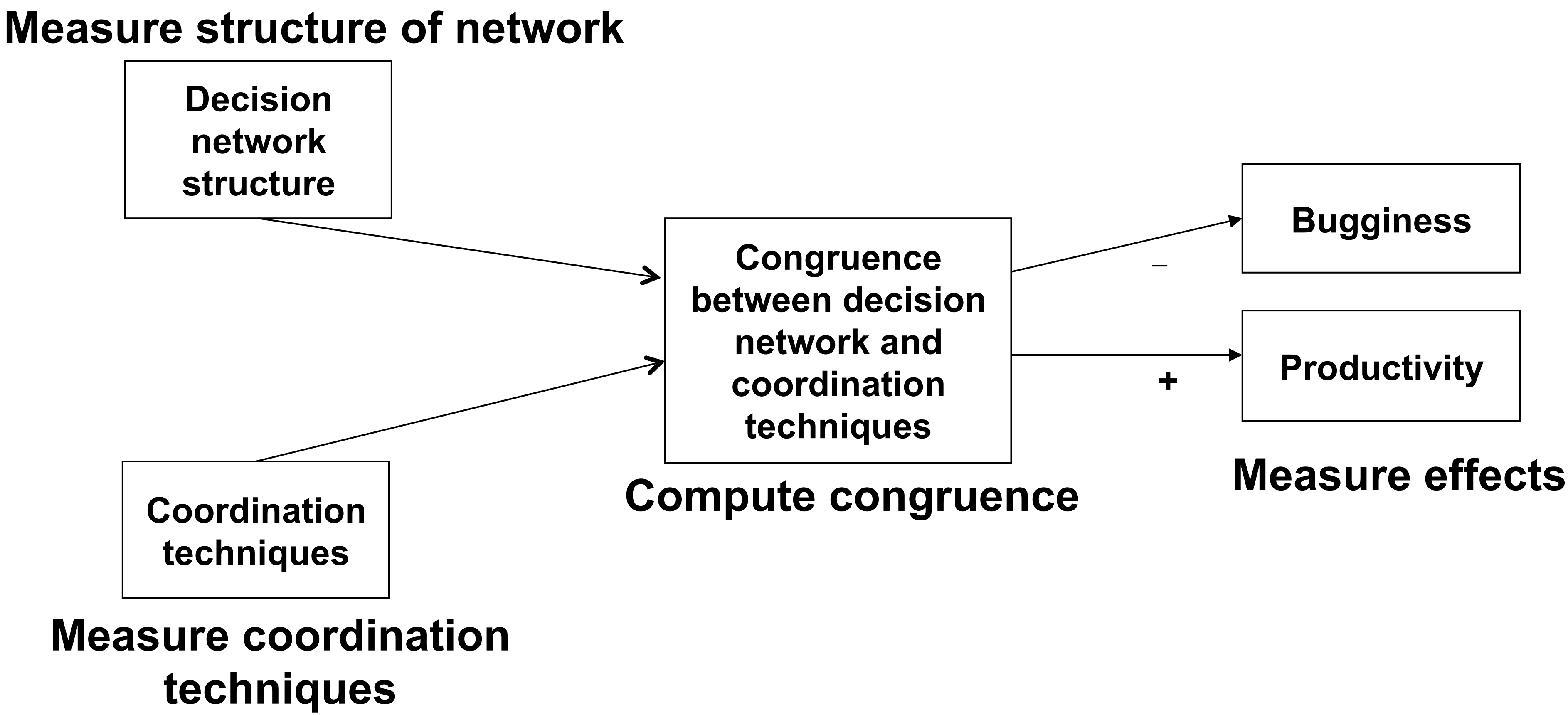
Congruence Over Time



Cataldo, M., Wagstrom, P., Herbsleb, J.D., Carley, K. (2006). Identification of coordination requirements: Implications for the design of collaboration and awareness tools. In Proceedings, *ACM Conference on Computer-Supported Cooperative Work, Banff Canada,* pp. 353-362.

Carnegie Mellon University

Research Challenges

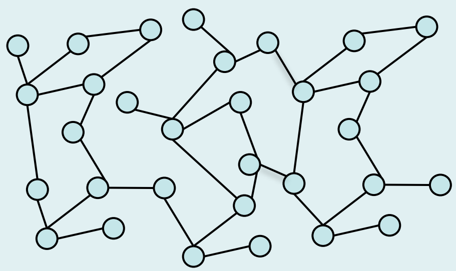


institute for SOFTWARE RESEARCH

Selected Work on Congruence in Software Engineering

- Kwan, Schröter, & Damian (2011)
  - Examined the relationship of congruence to build success.
- Kwan & Damian (2011)
  - Developed an aggreagted congruence measure based on multiple awareness mechanisms
- Avritzer, Paulish, Cai, & Sethi (2010)
  - DSMs to represent architectural dependencies and social communication networks, compute congruence
- Kwan, Schröter, & Damian (2009)
  - Developed a weighted congruence measure
- Sarma et al (2009)
  - Designed Tesseract for visualizing social networks, dependency networks, and congruence
- Bird, et al (2009)
  - Used socio-technical network measures to predict failure-prone components
- Bolici, Howison, & Crowston (2009)
  - Examined stygmergy as a mechanisms for establishing congruence in open source projects
- Valetto, Chulani, & Williams (2009)
  - Analyzed costs and risks of different approaches to close congruence gaps
- Valetto, et al (2007)
  - Develop a graph-theoretic algorithm for computing congruence

Theory and Social Coding



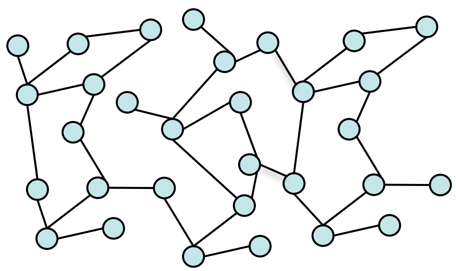
Decisions and Constraints

- Repositories: clumps of decisions
- Decision owners ≅ git access
- Constraints ≅ “uses” relation

. . . establishes a coordination problem . . .

This problem is typical of open source

Social Coding



Decisions and Constraints

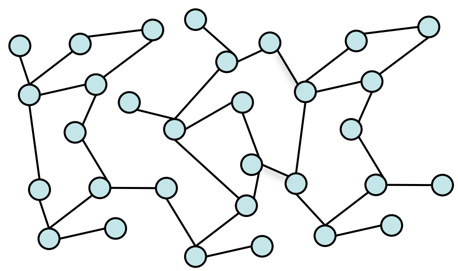
- Repositories: clumps of decisions
- Decision owners ≅ git access
- Constraints ≅ “uses” relation

. . . establishes a coordination problem . . .

. . . that the organization must solve.

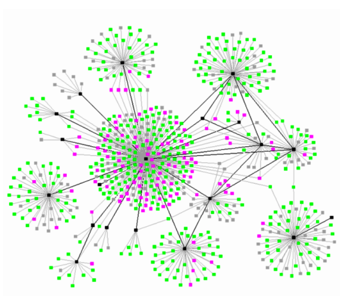
- Tool Affordances
- Adjustment & intervention
- Power asymmetries
- Hard power, soft power
- Audience and accountability

We're at the Beginning



Decisions and Constraints

- **Not just code!**
- **Popular frameworks, libraries, APIs**
- **Temporal order, pace of decisions**
- **Predict early, use in planning**



- **Match decision networks with techniques**
- **How to plan, correct, adjust**

- **What is the full set of techniques?**
- **Substitute, complement, compose?**
- **Role of new, social and transparent media?**

Takeaways

- We have a great many coordination techniques, what we need is a theory
  - We have made a start – we are at the beginning
- A good theory will incorporate the social and the technical
  - Either alone is “one hand clapping”

An Observation

- Human dimension increasingly taken into account in software engineering, e.g.
  - API usability
  - End user programming
  - Tools and environments

Another Observation

- Practical value versus enduring value
  - We need a portfolio of research
  - Validation of a technique can have immediate value
  - Theory development will not yield immediate practical results, yet
- In the long run, “There is nothing so practical as a good theory.”
  - Kurt Lewin (1959)

Yet Another Observation

- Software engineering research is based in behavioral science as strongly as it is based in computer science.

Questions?

Collaborators

- Kathleen Carley
- Marcelo Cataldo
- Laura Dabbish
- Audris Mockus
- Anita Sarma
- Colleen Stuart
- Jason Tsay
- Patrick Wagstrom