

Aligning Requirements and Testing - Current Challenges and Solutions

Robert Feldt

Blekinge Tekniska Högskola



Dept of Software Engineering
SERL Sweden (SE Research Lab)



BTH, Karlskrona



SERL Sweden:

6 professors

3 Assoc. professors

2 Assistant professors

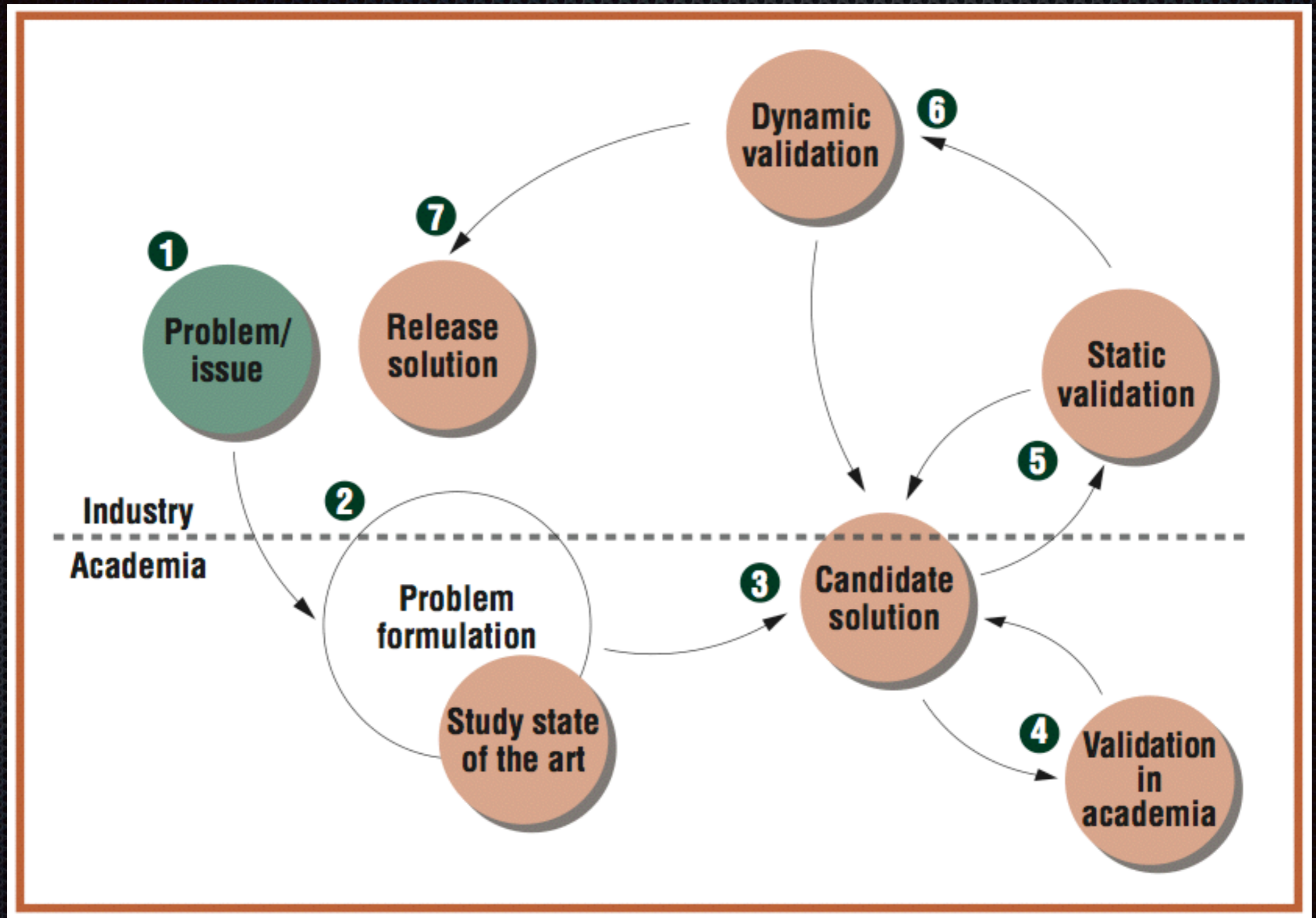
1 post doc

10 PhD students

Ranked 5th in the world in latest
SE&Systems ranking, 1st in EU!



How we often work (or try to work ;)



Trends

Topic Keywords	2012	% 2012
Software Engineering	13,354	100 %
Requirements	2173	16 %
Design	4618	35 %
Programming	2760	21 %
Testing OR Verification	1349	10 %
Req AND Testing	289	2.2%
Human Factors	90	0.7%
Social OR Sociology	348	2.6%
Psychology	68	0.5%
Personality	29	0.2%

→ 50.1%

What is Alignment?

We take a broader view and introduce:

Alignment = “*adjustment of RE and ST efforts for coordinated functioning & optimized product development*”

Alignment-as-activity = “*act of adjusting/arranging efforts involved in RE & ST so they work better together*”

Alignment-as-state = “*condition of RE & ST efforts having established a coordinated functioning*”

Previous work - in one page

- * **Involve testers in RE** => better Testing [Damian05]
- * **Rich interaction RE<->Test** => higher test coverage, manage risks, increased productivity [Chisan05]
- * **Traceability is well researched**
 - * Focus is on technical issues & tools
 - * Many claimed benefits, but also: volatile artefacts and not enough time to update traces [Cleland-Huang03]
- * **Model-based testing indirectly aligns**
 - * Detailed Req models => automated testing, but costly



Singapore Univ
of Tech, Singapore



Wahlmers,
Eborg



LTH, Lund



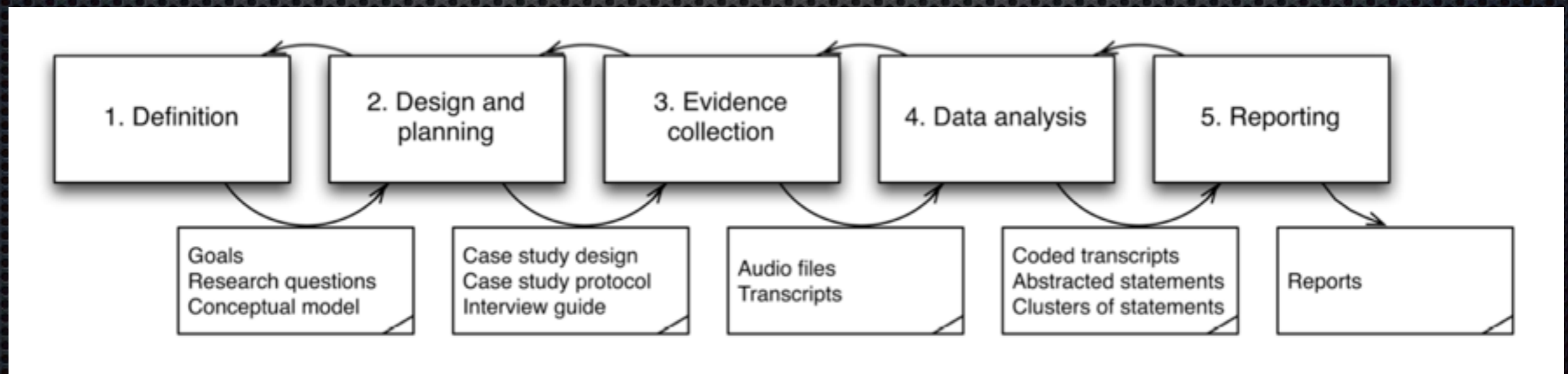
BTH, Ronneby



Ok, so what did we do?

Main goals:

1. understanding challenges in REW alignment
2. identify common practices used in industry



Six (6) companies involved

Company	A	B	C	D	E	F
Type of company	Software development, embedded products	Consulting	Software development	Systems engineering, embedded products	Software development, embedded products	Software development, embedded products
# employees in software development of targeted organisation	125-150	135	500	50-100	300-350	1,000
# employees in typical project	10	Mostly 4-10, but varies greatly	50-80	software developers: 10-20	6-7 per team, 10-15 teams	Previous process: 800-1,000 person years
Distributed	No	Collocated (per project, often on-	Yes	Yes	Yes	Yes
						Waterfall
Duration of a typical project	6-18 months	No typical project	1-5 years to first delivery, then new software release for 1-10 years	1-5 years to first delivery, then new software releases for 1-10 years	1 year	Previous process 2 years
# requirements in typical project	100 (20-30 pages HTML)	No typical project	600-800 at system level	For software: 20-40 use cases	500-700 user stories	Previous process: 14,000
# test cases in a typical project	~1,000 test cases	No typical project	250 at system level		11,000+	Previous process 200,000 at platform level, 7,000 at system level
Product Lines	Yes	No	Yes	Yes	Yes	Yes
Open Source	Yes	Yes. Wide use, including contributions	Yes, partly	No	No	Yes (with new agile process model)

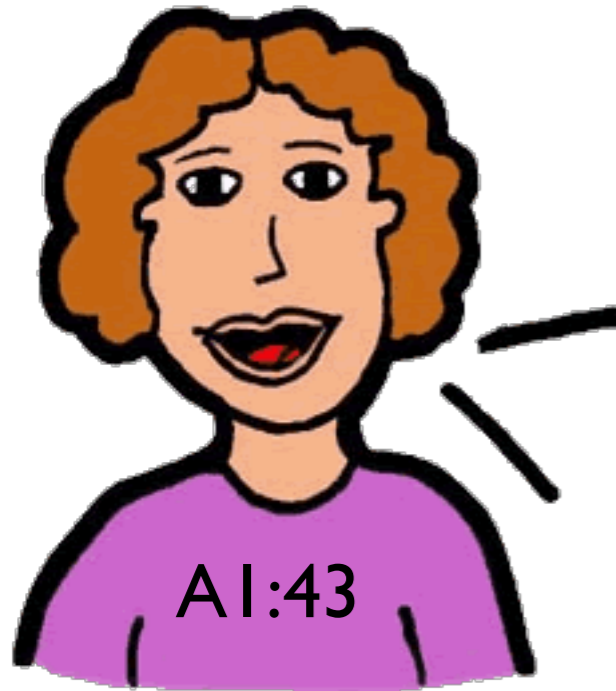
Seven (7) roles involved

Role	A	B	C	D	E	F
Requirements engineer						F1 (senior), F6 (senior), F7 (senior)
Systems architect				D3 (junior)	E1 (senior)	F4 (senior)
Software developer		B1 (junior), B2 (senior), B3 (senior)				F13 (senior)
Test engineer	A2 (senior)		C1 (senior), C2 (junior)	D2 (senior)	E3 (senior)	F9 (senior), F10 (senior), F11 (junior), F12 (senior), F14 (senior)
Project manager	A1		C3 (senior)	D1		F3 (junior),
	(junior)			(senior)		F8 (senior)
Product manager	A3 (senior)				E2 (senior)	
Process manager						F2 (junior), F5 (senior), F15 (junior)

30 x 90mins semi-structured interviews



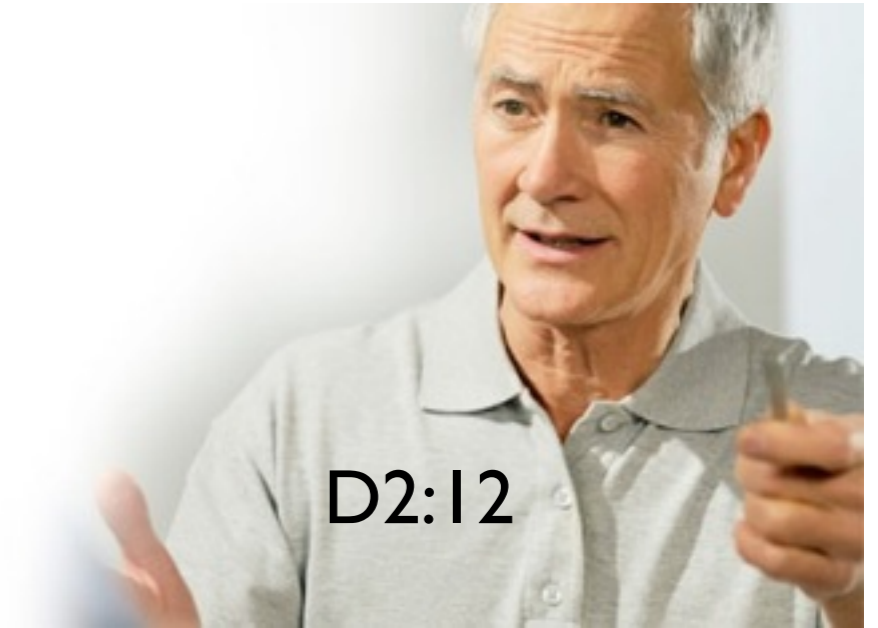
Results - in general



AI:43

*‘[with misaligned requirements]
there wasn’t a bug, but the
functionality was implemented in
such a way that it was hard to do
what the customer [originally]
intended’*

Results - in general

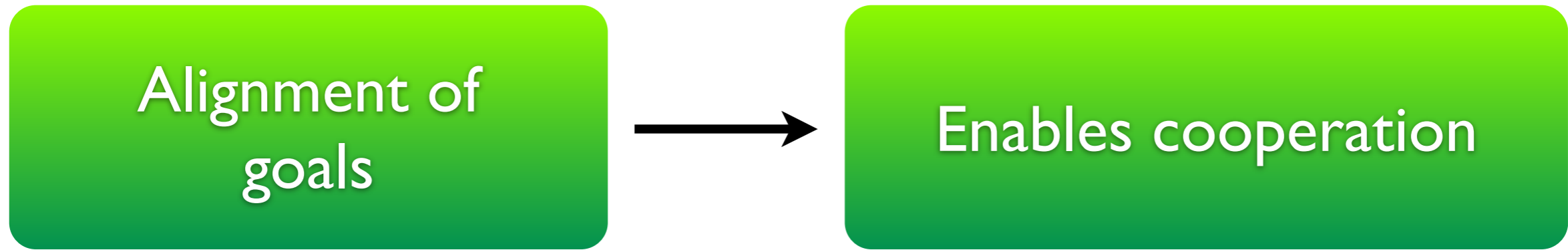


builds customer trust since good alignment allows the company to 'look into the customer's eyes and explain what have we tested... on which requirements'

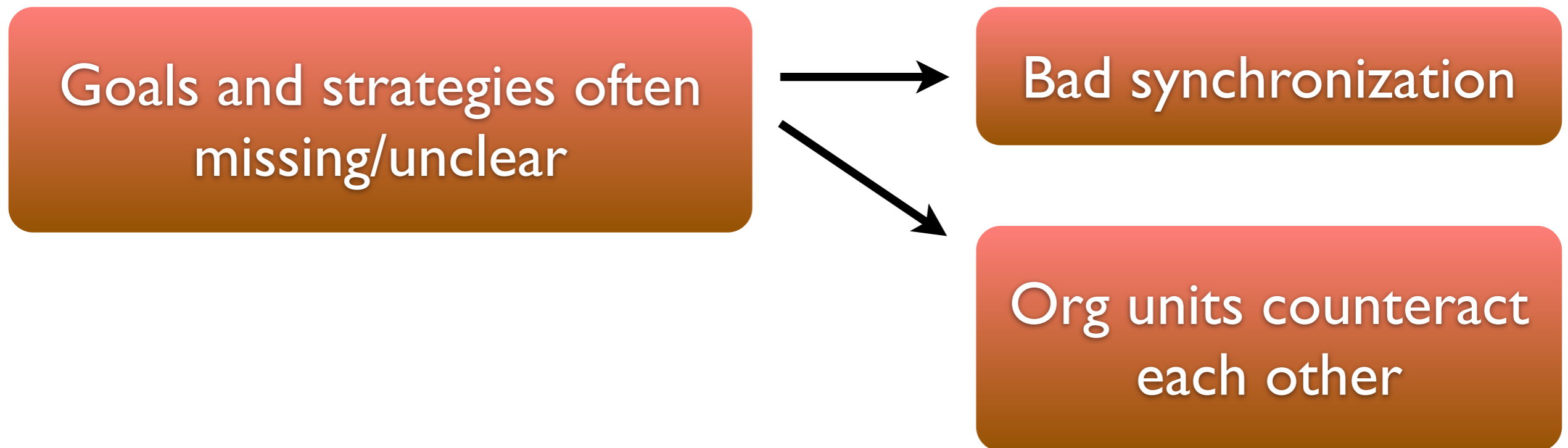
Results - challenges

	Id	Challenge	Company					
			A	B	C	D	E	F
	Ch1	Aligning goals & perspectives organisation	X	X	X		X	X
	Ch2	Cooperating successfully	X		X	X	X	X
Req spec quality	Ch3.1	Defining clear and verifiable requirements			X	X	X	X
	Ch3.2	Defining complete requirements		X		X	X	X
	Ch3.3	Keeping requirements documents updated						X
VV quality	Ch4.1	Full test coverage	X	X	X	X		X
	Ch4.2	Defining a good verification process						X
	Ch4.3	Verifying quality requirements		X		X		X
	Ch5	Maintaining alignment when requirements change	X		X			X
Req's abstract levels	Ch6.1	Defining requirements at abstraction level well matched to test cases				X		X
	Ch6.2	Coordinating requirements at different abstraction levels	X					X
Tracea bility	Ch7.1	Tracing between requirements and test cases	X	X	X	X		X
	Ch7.2	Tracing between requirements abstraction levels		X	X	X		
	Ch8	Time and resource availability			X		X	X
	Ch9	Managing a large document space			X	X		X
	Ch10	Outsourcing of components or testing				X		X

CI: Aligning goals throughout org



BUT:



CI: Aligning perspectives throughout org

Alignment of
perspectives on
problem/solution
domain



Better communication:
externally & internally

BUT:



EI:20

Systems
architect

when there is 'higher expectations on the product than we [systems architect] scoped into it' a lot of issues and change requests surface in the late project phases

CI: Aligning perspectives throughout org

Alignment of perspectives on problem/solution domain



Better communication: externally & internally

BUT:



A2:105

Test engineer

for higher abstraction levels there are no attempts to synchronize, for example, the testing strategy with the goals of dev projects to agree on important areas to focus on

C2: Successful Co-op/communication

Close co-op between
roles and units



Less friction & better
alignment

Prod. manager: ‘an “us and them” validation of product level requirements is a big problem’

Company F: lack of early co-op in validating reqs result in late discovery of failures to meet reqs. Dev project say: ‘We did not approve these requirements, we can’t solve it’

Company B: ‘We have succeeded with mapping requirements to tests since our process is more of a discussion’

Results - practices #1

Cat.	Id	Description	Company					
			A	B	C	D	E	F
Requirements	P1.1	Customer communication at all requirements levels and phases		X	X	X	X	X
	P1.2	Development involved in detailing requirements	X	X				X
	P1.3	Cross-role requirements reviews	X		X	X	X	X
	P1.4	Requirements review responsibilities defined					X	X
	P1.5	Subsystem expert involved in requirements definition				X		X
	P1.6	Documentation of requirement decision rationales					S	S
Validation	P2.1	Test cases reviewed against requirements						X
	P2.2	Acceptance test cases defined by customer		X				
	P2.3	Product manager reviews prototypes	X				X	
	P2.4	Management base launch decision on test report						X
	P2.5	User / Customer testing		X		X	X	X
Verification	P3.1	Early verification start					X	X
	P3.2	Independent testing			X	X	X	
	P3.3	Testers re-use customer feedback from previous projects				X	X	X
	P3.4	Training off-shore testers			X			

Results - practices #2

Change	P4.1	Process for requirements changes involving VV	X		X	X	X	X
	P4.2	Product-line requirements practices	X		X			
	P5	Process enforcement			X			S
Tracing	P6.1	Document-level traces	X					
	P6.2	Requirements-test case traces						X
	P6.3	Test cases as requirements	X					X
	P6.4	Same abstraction levels for requirements and test spec			X	X		
	P7	Traceability responsibility role			X	X	X	
Tools	P8.1	Tool support for requirements and testing	X		X	X	X	X
	P8.2	Tool support for requirements-test case tracing	X		X	X	X	X
	P9	Alignment metrics			X	X	X	X
	P10	Job rotation				S		S

Alignment metrics

Job rotation

P9: Measure alignment

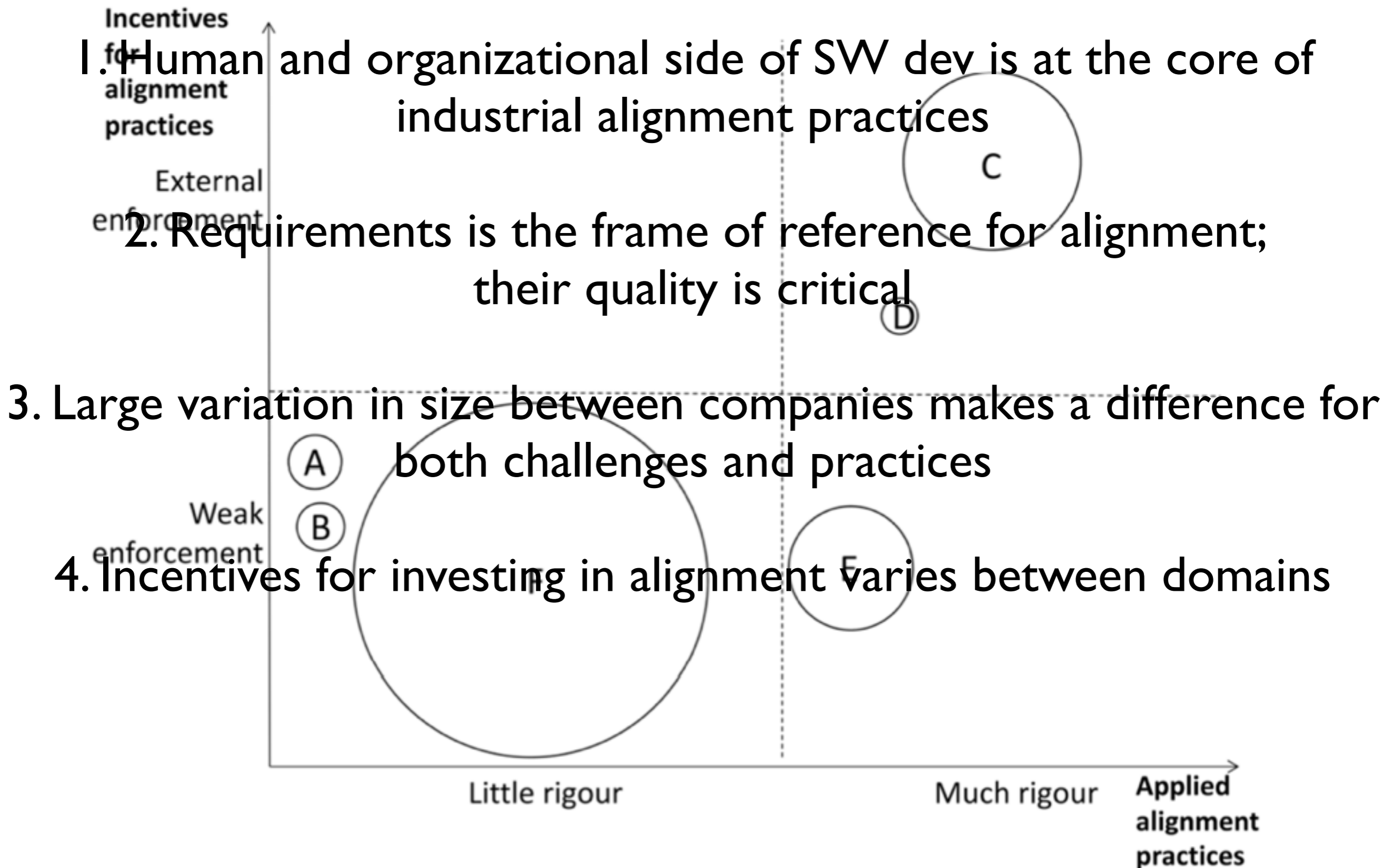
Company C: 'we measure how many requirements are already covered with test cases and how many are not' (through req and test management tool)

Company E & F: Also measure req coverage but say there is a lot of judgement involved and the metrics are only partial: "If you have one requirement, that requirement may need 16 test cases to be fully compliant. But you implement only 14 out of those. And we don't have any system to see that these 2 are missing.'

PI0: Job Rotation

Company D & F: Suggested as a way to increase contact network and experiences and over time create more aligned perspectives in the organisation. Key for alignment is individuals and their experiences and willingness to communicate and align with others.

Discussion and Analysis



Acknowledgement

Note that I was only a small part of this work! Credits should go mainly to the PhD students involved and to the largest group involved (Lund).

E. Bjarnason, P. Runeson, M. Borg, M. Unterkalmsteiner, E. Engström, B. Regnell, G. Sabaliauskaite, A. Loconsole, T. Gorschek, and R. Feldt, "Challenges and Practices in Aligning Requirements and Verification & Validation: A Case Study of Six Companies", Journal of Empirical Software Engineering, 2013.

http://www.robertfeldt.net/publications/bjarnason_2013_alignment_challenges.html

Questions: robert.feldt@bth.se

Table 1. The RE distances included in the Gap Model and in the Gap Finder.

	<i>Type of distance</i>	<i>Between</i>
PEOPLE	D1 Geographical distance Physical distance of desks	Roles related to requirements and testing
	D2 Organisational distance Distance between organisational units	
	D3 Psychological distance Perceived effort to communicate	
	D4 Cognitive distance Difference in knowledge	
ARTE-FACTS	D5 Adherence distance Difference betw. documented content and perception of agreement or reality	Artefact and reality
	D6 Semantic distance Difference in meaning	Artefacts
	D7 Navigational distance Effort to navigate between	
ACTIVI-TIES	D8 Temporal distance Time between activities, e.g. specifying and using a requirements specification	Activities

[Bjarnason2013]

Harmony?





Kyoto

Sydney

Atlanta

Barcelona

New Delhi

Minneapolis

Paris.....



www.re14.org



CI: Aligning perspectives throughout org

Alignment of
perspectives on
problem/solution
domain



Better communication:
externally & internally



F13:29

Software
developer

‘if both [Req eng & SW Dev] have
a common perspective [of
technical possibilities], then it
would be easier to understand
what [requirements] can be set
and what cannot be set’