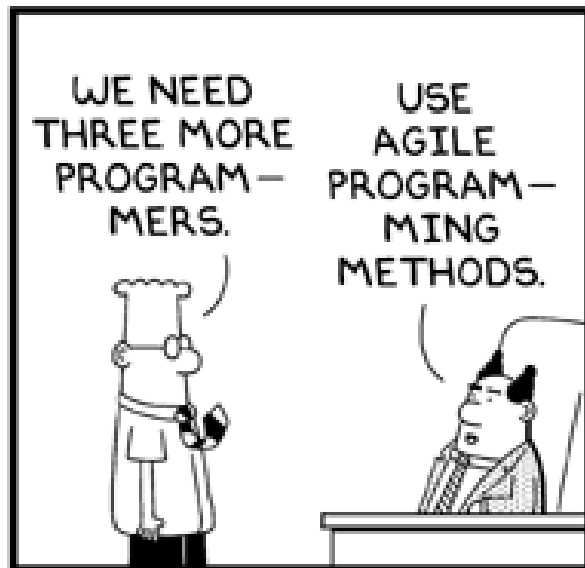


Empirical Evidence of Agile Methods:

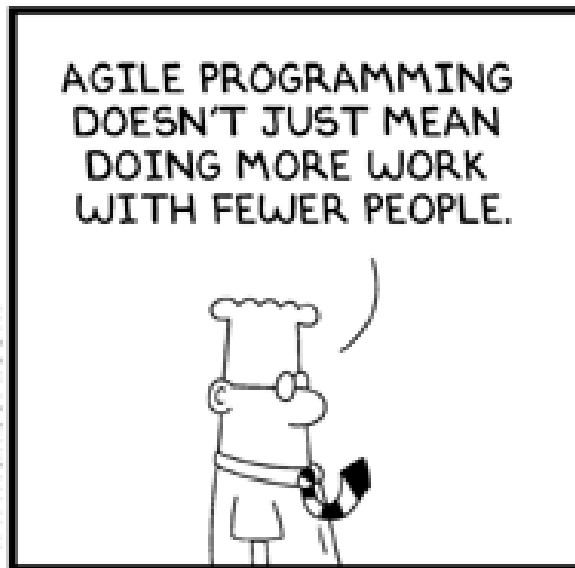
ce que nous savons,
ce que nous pensons savoir et
ce que nous voulons savoir

Grigori Melnik
Microsoft patterns & practices





www.dilbert.com scottadams@aol.com



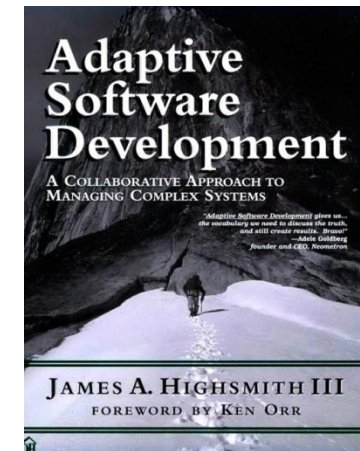
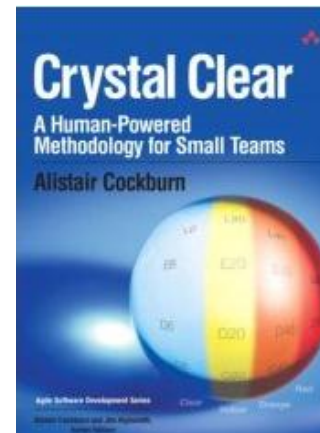
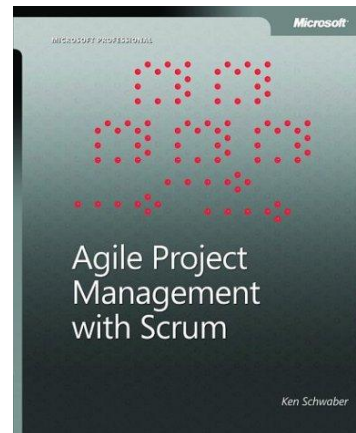
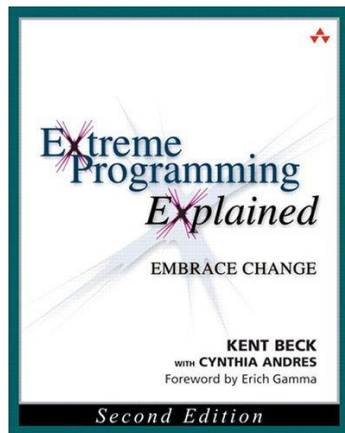
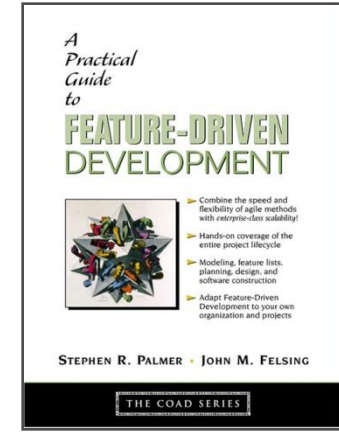
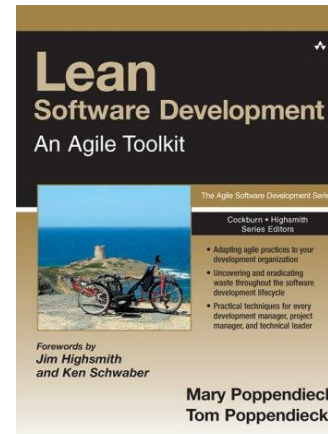
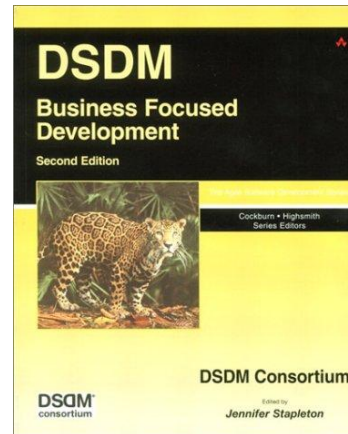
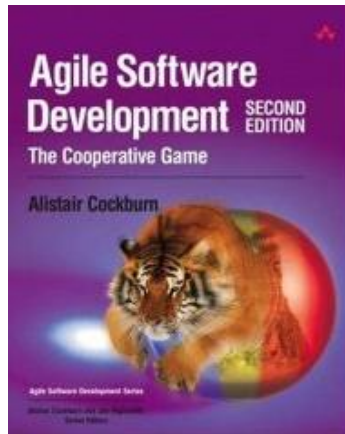
© 2005 Scott Adams, Inc./Dist. by UFS, Inc.



© Scott Adams, Inc./Dist. by UFS, Inc.

patterns & practices

proven practices for predictable results



...

patterns & practices
proven practices for predictable results

The Issue ...

Do agile methods work because of their engineering and management practices or because the people who introduce them are simply very good developers ?

Understanding a discipline demands observation, model building, and experimentation.

The State of the Practice

- Largely measurement-free zone
 - Few experiments
 - Even fewer published results
 - Minimal focus on data collection
 - Existing data is often incomplete or tainted
 - High-profile consultants provide anecdotal evidence
- Existing experiments
 - Either too trivial
 - Or have experimental design flaws
 - Vast number of human interaction variables that cannot be controlled
 - Small samples
 - Non-practitioners as subjects
 - Conducted over short time

Experience Reports are Situational

- The soundness of an idea (process, technique etc.) is not absolute
- Context-specific!
- Cross-project comparison is very difficult



Evidence shows

- Agile is entering the **mainstream**
- Teams practicing agile are getting **larger** and more **distributed**
- Agile self-organizing teams report **higher job satisfaction**
- Agile is **not just developer-centric**
- **Rational design decision-making**
- **Testing to the forefront** of development



AGILE ADOPTION – WELCOME TO MAINSTREAM

patterns & practices

proven practices for predictable results



The Story of Agile

- 2000-2001 Suitable contexts
- 2002 Scalability
- 2003 Adaptability
- 2004 Methodologies zoo
- 2005 Convergence
- 2006-2007 Entering mainstream
- 2008-... Agile v2



Agile Adoption (Methods & tools, May 2005)

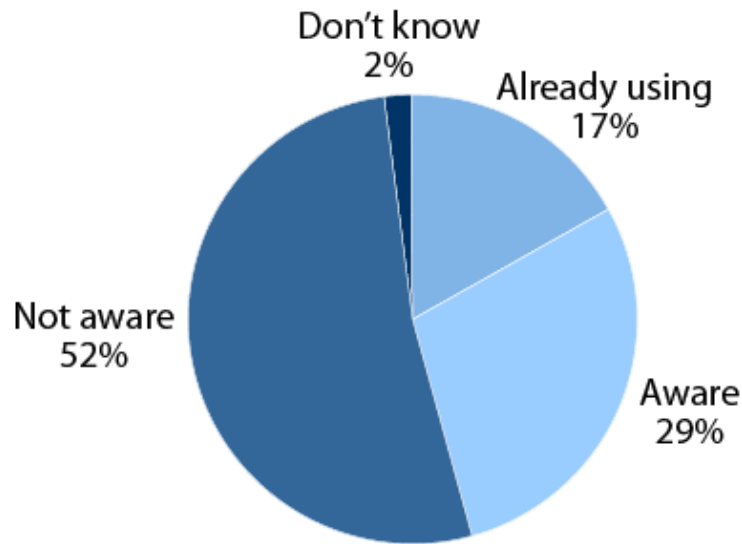
- **At what stage is the agile approach (XP, Scrum, FDD, ...) adoption at your location?**
 - Not aware 26%
 - Not using 16%
 - Investigating 14%
 - Analysed and rejected 3%
 - Pilot projects 4%
 - Partial implementation (adoption of some agile practices) 17%
 - Partial deployment (some projects are using this approach) 12%
 - Deployed (all new projects are using this approach) 8%
- } 41%

N=232, Selection bias

Agile Adoption (Forrester, Sep 2006)

FORRESTER®

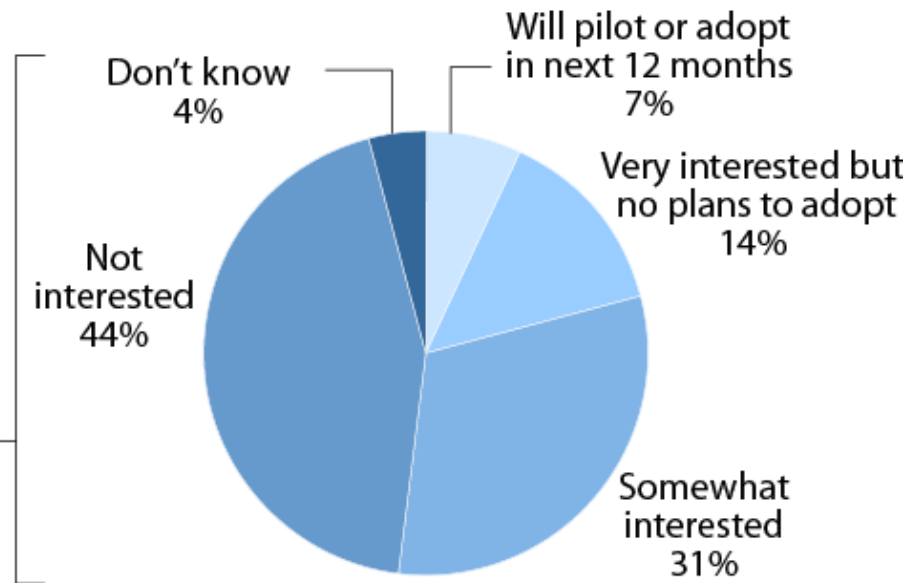
"Are you aware of Agile software development processes?"



Base: 1,078 North American and European enterprises

NB: - not just software companies;
- large companies (>1,000 employees)

"How interested are you in adopting Agile software development processes?"

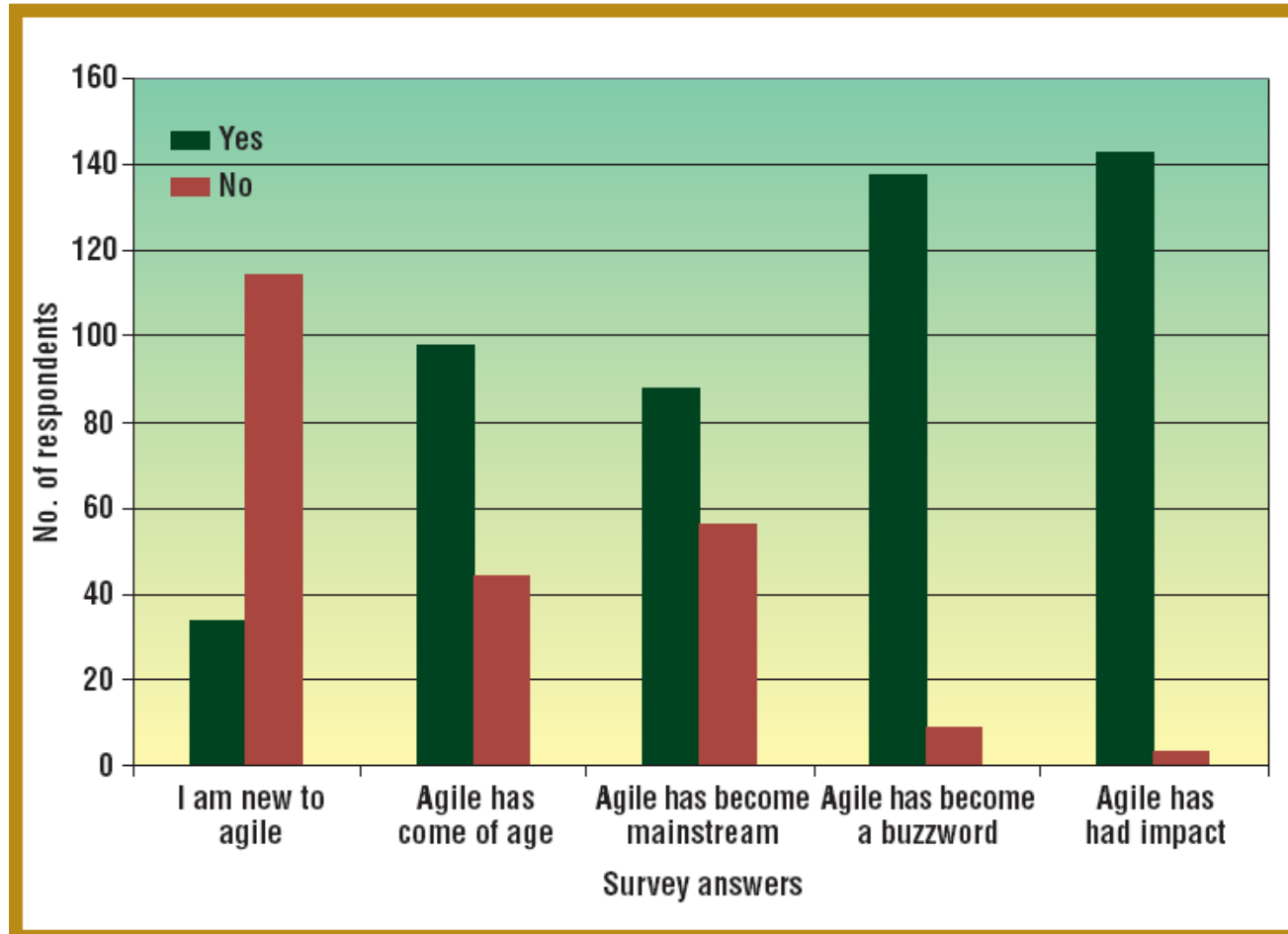


Base: 318 enterprises that are aware of but not already using Agile software development processes

patterns & practices

proven practices for predictable results

IEEE Software Blitz Perception Survey (Erdogmus 2007)



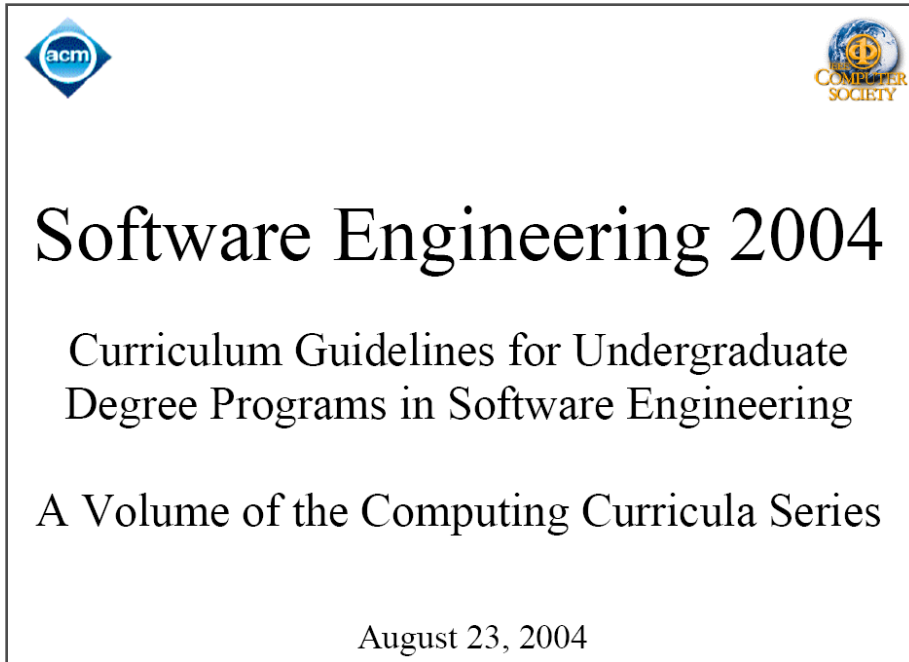
NB: N=150, selection bias

Professional conferences

- XP/Agile Universe + ADC series = Agile Conference
 - annual, since 2001, (230 attendees)
 - 2007 (1,200 attendees)
 - 2008 (1,700 attendees expected, with 400 presenters!)
- European XP conference
 - since 2002, annual
- Many XP Days and AgileOpen around the world
- OOPSLA
 - becoming more and more agile context-rich
 - merger?
- Agile Development Conference (SQE)
 - new, commercial, still gets 400+ attendees



In Academia...



www.computer.org/education/cc2001/SE2004Volume.pdf

- Include "agile methods" as Essential in both Software Life Cycles course and Project Management course
- Individual practices:
 - refactoring (E)
 - test-first design (D)

Agile Adoption Rates Increase With Company Size

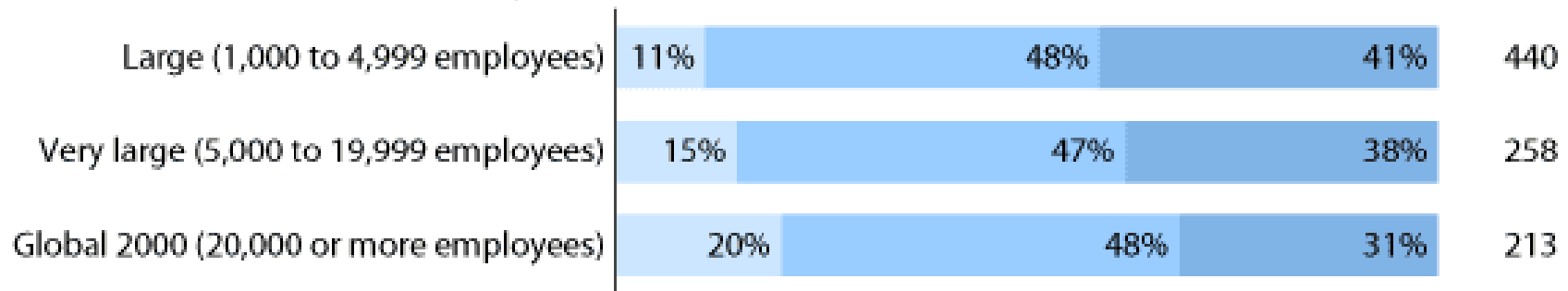
FORRESTER

"Are you aware of Agile software development processes?"

Already using Aware Not aware

Company size

Number of respondents

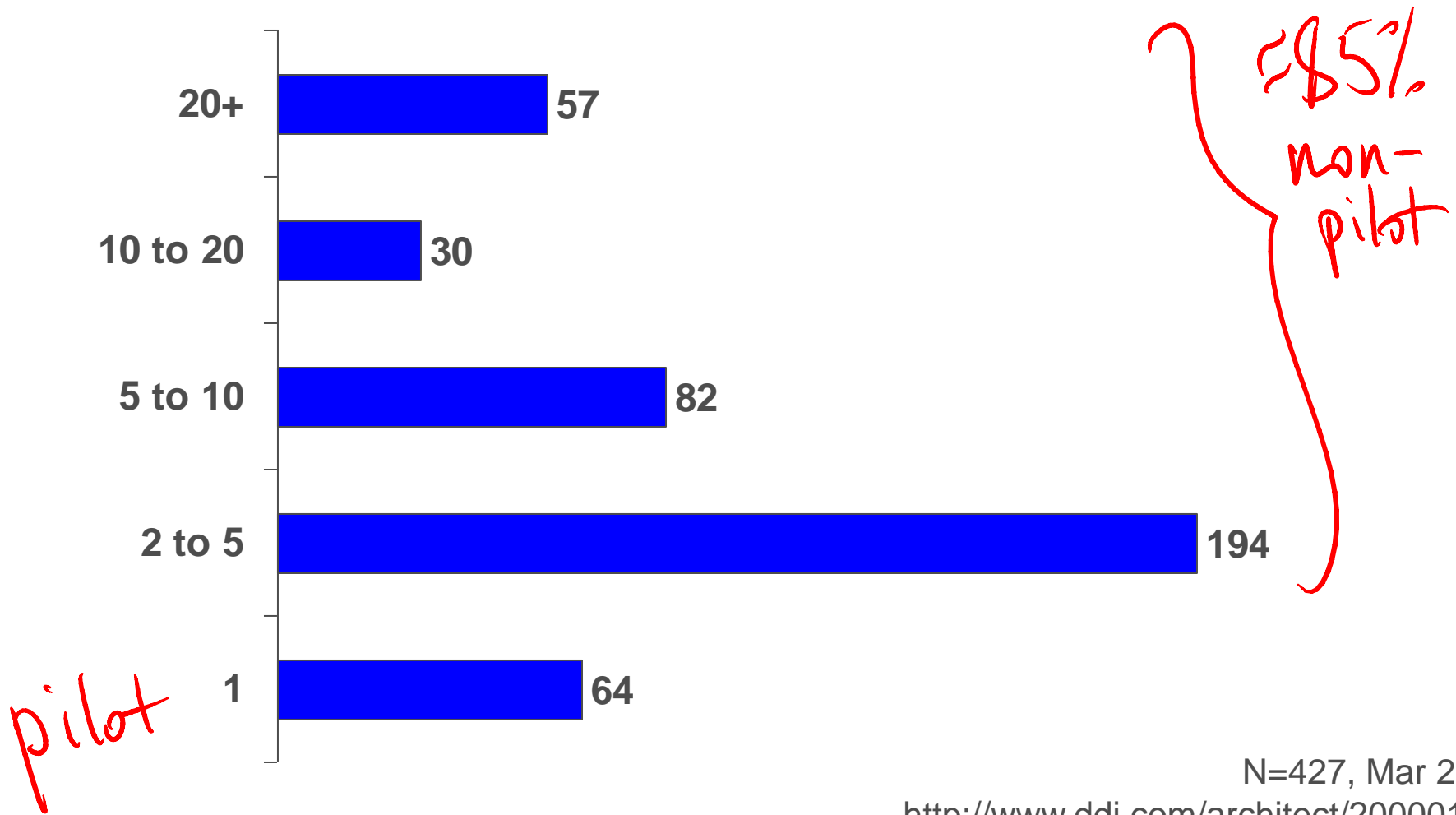


N=911, Nov 2005

patterns & practices

proven practices for predictable results

Number of Agile Projects Run (Ambler 2007)



N=427, Mar 2007

<http://www.ddj.com/architect/200001986>

patterns & practices

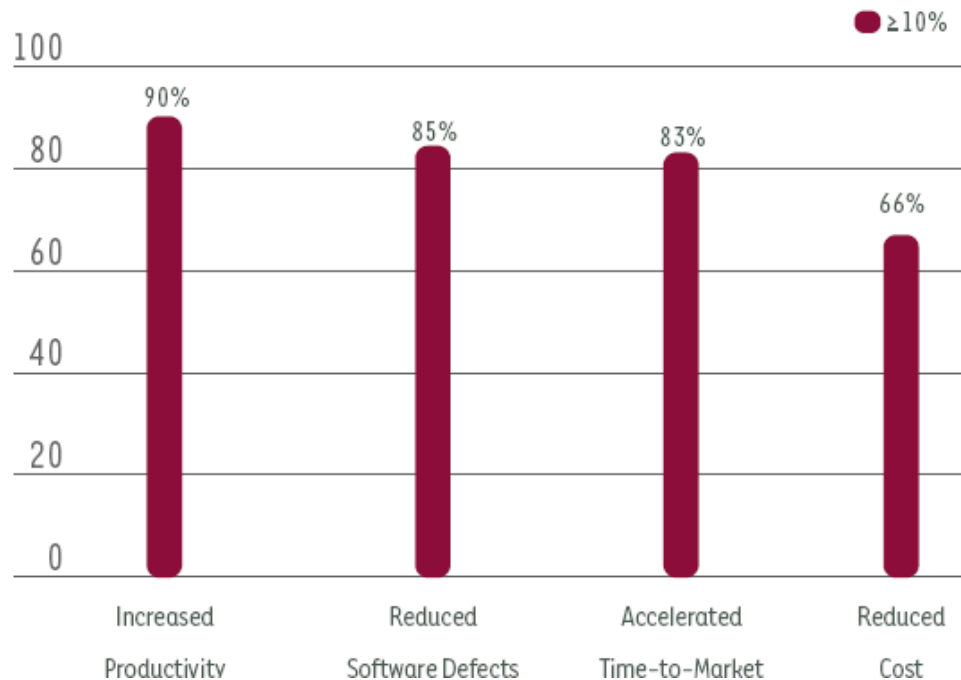
proven practices for predictable results

Value Realized (VersionOne 2006, 2007)

Estimate SPECIFIC IMPROVEMENTS you have actually realized from implementing Agile practices

	≥10%	≥25%
Accelerated time-to-market	86%	60%
Increased productivity	87%	55%
Reduced software defects	86%	55%
Reduced cost	63%	26%

N=722, Jul 2006
VersionOne
Global Survey



N=1,700, Jul 2007
VersionOne
2nd Global Survey

Boeing Case Study (Bedoll 2003)

- Critical system: Boeing Commercial Airplane Electrical design
- First project – Tayloristic, heavy-weight
 - *“invested 60 man-years, and failed”*
- Second project – Agile
 - *“invested 4 man-years, and continues to be dramatically successful”*

Aspect	Success, Agile	Failure, Tayloristic
Business Process	The tool evolves incrementally, along with the business process.	The process is build after the tool is done, or the tool is created using imagined business process.
Customer Involvement	Daily contact and hands-on evaluation of new-feature code by the users within days of requirements definition.	Monthly paper-design reviews, and hands-on feature evaluation by the users nine months after requirements.
Simplicity and Focus	A single group of 60 users; a single airplane program with a small set of airplanes.	2000 users, all five airplane programs with all airplanes past, present, future (several thousand planes).
Development Tools and Processes	ACCESS, SQL Server, Visual Basic, lean development processes (but still a formal testing and release process)	C++, Motif, heavy-weight, paper-intensive development processes.

Key Agile Practices (VersionOne 2007)

Which of the following practices do you employ?

Iteration Planning	65%
Unit Testing	60%
Daily Standup	55%
Release Planning	54%
Continuous Integration	50%

N=1,700

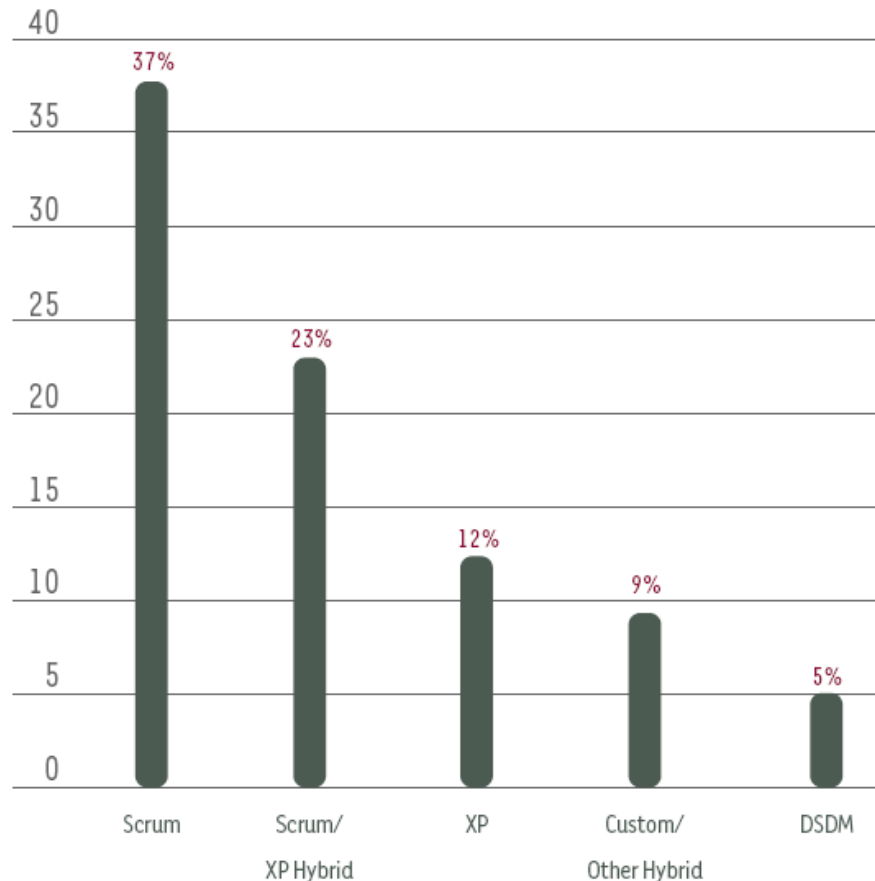
patterns & practices

proven practices for predictable results



Key Agile Methodologies (VersionOne 2007)

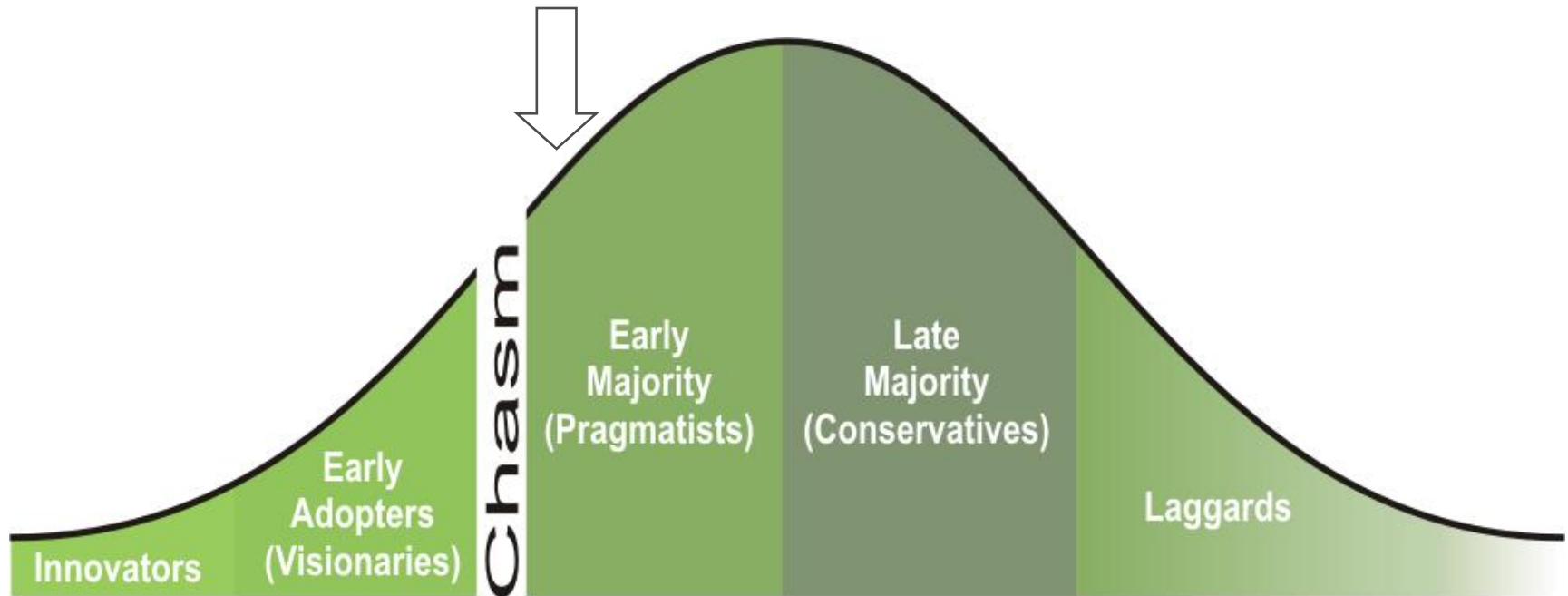
Which Agile methodology do you follow most closely?



N=1,700

patterns & practices
proven practices for predictable results

Have Agile Methods Crossed the Chasm Yet?



Moore, G. Crossing the Chasm, 1991, 1999, 2002

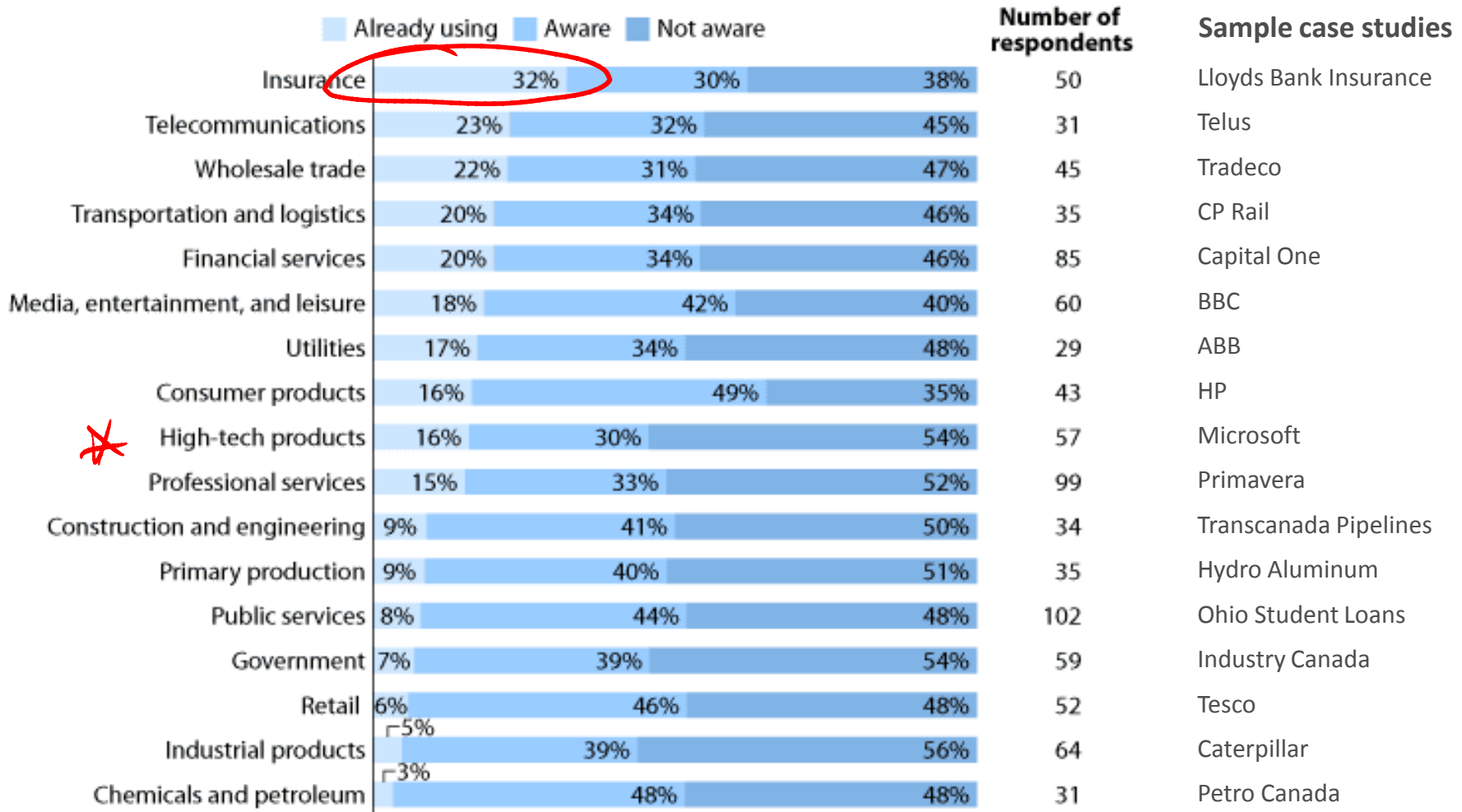
patterns & practices

proven practices for predictable results

Agile Adoption By Industry

FORRESTER

"Are you aware of Agile software development processes?"



N=911, Nov 2005

patterns & practices

proven practices for predictable results

AGILE = HAPPY TEAMS ?

patterns & practices

proven practices for predictable results



Effects on Job Satisfaction and Turnover



patterns & practices

proven practices for predictable results

Who cares?

happy teams → productive teams

happy teams → lower turnover

patterns & practices

proven practices for predictable results



Who cares?

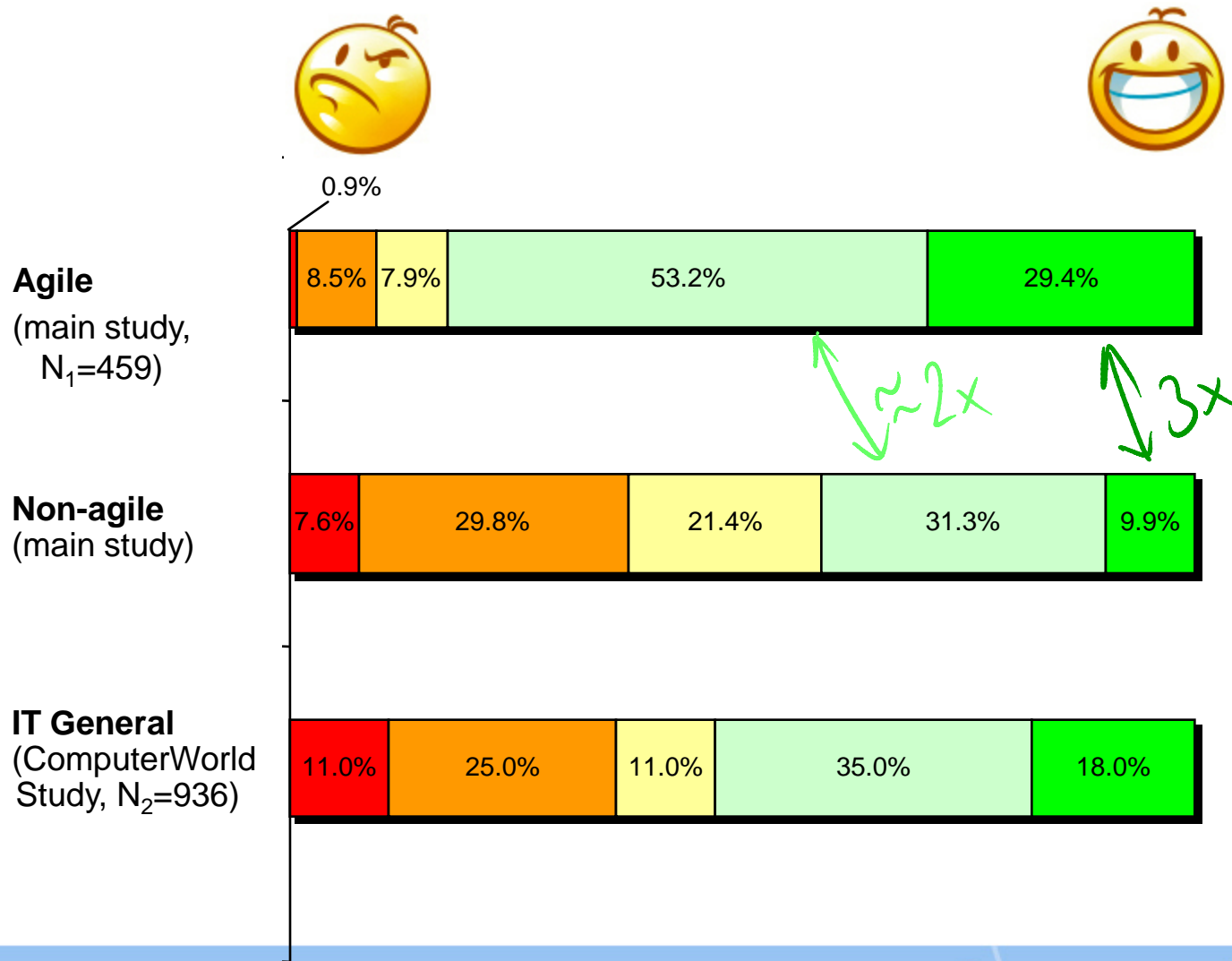
happy teams ~~→~~ productive teams *myth*

happy teams *✓* → lower turnover *fact*




⇒ considerable economic effect:

turnover costs **70%-200%** of employee's annual salary (e.g. Meta Group Report)

Job Satisfaction (Melnik/Maurer 2005)



Results: Overall Satisfaction by Job Roles and Levels of Agile Experience (N=482)

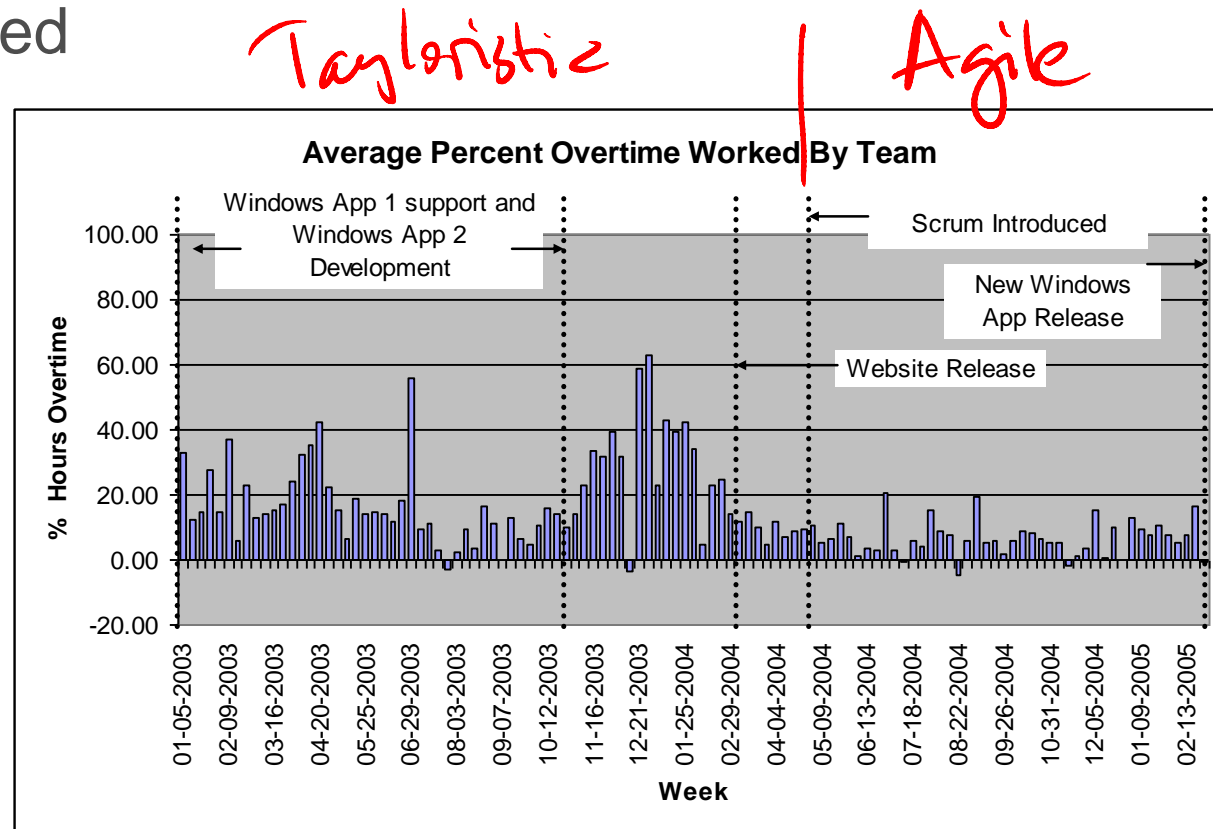
Role	Level of experience with agile	Overall Satisfaction					Grand total
		Very dis-satisfied	Somewhat dissatisfied	Neither satisfied nor dissatisfied	Some-what satisfied	Very satisfied	
	Practice agile now		2 3%	3 5%	39 60%	21 32%	65 100%
	Do not know what agile methods are	1 17%	1 17%	1 17%	3 50%		6 100%
	Haven't practiced but interested in trying	1 9%	3 27%	1 9%	4 36%	2 18%	11 100%
	Have practiced before but not now			2 33%	2 33%	2 33%	6 100%
	Have tried agile in training environment		1 25%	1 25%	1 25%	1 25%	4 100%
	Practice agile now	3 1%	25 11%	19 8%	121 51%	67 29%	235 100%
	Do not know what agile methods are		5 20%	6 24%	10 40%	4 16%	25 25%
	Haven't practiced but interested in trying	5 7%	24 33%	20 27%	19 26%	5 7%	73 100%
	Have practiced before but not now	4 13%	8 27%	4 13%	13 43%	1 3%	30 100%
	Have tried agile in training environment	2 14%	3 21%	5 36%	2 14%	2 14%	14 100%
	Practice agile now			1 9%	5 45%	5 45%	11 100%
	Have practiced before but not now		1 100%				1 100%
	Have tried agile in training environment				1 100%		1 100%

patterns & practices

proven practices for predictable results

Sustainable Pace (Mann/Maurer 2005)

- 2 year longitudinal case study
- Researcher embedded in small development team
- Scrum is introduced
- Overall results:
 - Reduced overtime
 - Increased customer satisfaction



MODELING & DESIGN DECISION MAKING

patterns & practices

proven practices for predictable results



Design Decision Making in Small Agile Teams (Zannier/Maurer 2006)

- ✓ ■ Rational = consequential choice
 - Concurrent comparison of tradeoffs
- Naturalistic = sequential evaluation
 - No tradeoffs
- Qualitative study (action research)
 - 3 companies, 23 developers

! Contrary to
DTSTTC&W

On Design (Ambler 2007)

- 93% of agile teams do whiteboard modeling
- 77% of agile teams do some requirements envisioning up front
- 77% of agile teams also do a bit of architectural envisioning up front

N=427

TEST-DRIVEN DEVELOPMENT

patterns & practices

proven practices for predictable results



Test-Driven Development (Industry Subjects)

Family of studies	Type	Development time analyzed	Legacy project?	Organization studied	Software built	Software size	No. of participants	Language	Productivity effect	Quality effect
Sanchez et al. ⁶	Case study	5 years	Yes	IBM	Point-of-sale device driver	Medium	9–17	Java	Increased effort 19%	40% [†]
Bhat and Nagappan ⁷	Case study	4 months	No	Microsoft	Windows networking common library	Small	6	C/C++	Increased effort 25–35%	62% [†]
	Case study	≈ 7 months	No	Microsoft	MSN Web services	Medium	5–8	C++/C#	Increased effort 15%	76% [†]
Canfora et al. ⁸	Controlled experiment	5 hours	No	Soluziona Software Factory	Text analyzer	Very small	28	Java	Increased effort by 65%	Inconclusive based on quality of test
Damm and Lundberg ⁹	Multi-case study	1–1.5 years	Yes	Ericsson	Components for a mobile network operator application	Medium	100	C++/Java	Total project cost increased by 5–6%	5–30% decrease in fault-slip-through rate; 55% decrease in avoidable fault costs
Melis et al. ¹⁰	Simulation	49 days (simulated)	No	Calibrated using Klondike-Team and Quinary data	Market information project	Medium	4 [‡]	Smalltalk	Increased effort 17%	36% reduction in residual defect density

...

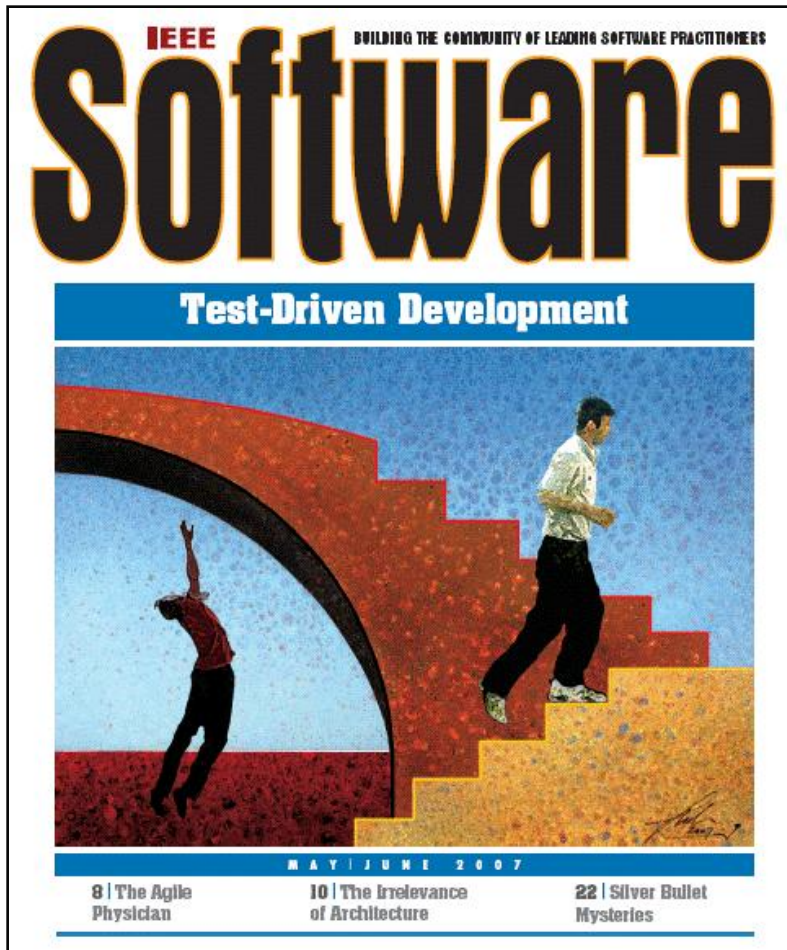
patterns & practices

proven practices for predictable results

Test-Driven Development (Academic Subjects)

Family of studies	Type	Development time analyzed	Legacy project?	Organization studied	Software built	Software size	No. of participants	Language	Productivity effect	Quality effect
Flohr and Schneider ¹⁵	Quasi-controlled experiment	40 hours	Yes	University of Hannover	Graphical workflow library	Small	18	Java	Improved productivity by 27%	Inconclusive
Abrahamsson et al. ¹⁶	Case study	30 days	No	VTT	Mobile application for global markets	Small	4	Java	Increased effort by 0% (iteration 5) to 30% (iteration 1)	No value perceived by developers
Erdogmus et al. ¹⁷	Controlled experiment	13 hours	No	Politecnico di Torino	Bowling game	Very small	24	Java	Improved normalized productivity by 22%	No difference
Madeyski ¹⁸	Quasi-controlled experiment	12 hours	No	Wroclaw University of Technology	Accounting application	Small	188	Java	n/a	-25 to -45% [†]
Edwards ²⁰	Artifact analysis	2-3 weeks	No	Virginia Tech	CS1 programming assignment	Very small	118	Java	Increased effort 90%	45% [†]
Pančur et al. ²¹	Controlled experiment	4.5 months	No	University of Ljubljana	4 programming assignments	Very small	38	Java	n/a	No difference
George ²²	Quasi-controlled experiment	1-3/4 hours	No	North Carolina State University	Bowling game	Very small	138	Java	Increased effort 16%	16% [†]
Müller and Hagner ²³	Quasi-controlled experiment	10 hours	No	University of Karlsruhe	Graph library	Very small	19	Java	No effect	No effect, but better reuse and improved program understanding

TDD in Non-Trivial Contexts



IEEE Software May/June 2007:

- Control systems design
- GUI development
- Database development
- Incorporating performance testing in TDD

Acceptance Testing and Storytest-Driven Development

How do business and technology experts utilize STDD in the software development lifecycle?

What kinds of benefits and limitations does STDD manifest?



Acceptance Testing and Storytest-Driven Development – Understandability

- Can executable acceptance tests describing customer requirements be easily understood and implemented by a technology expert with no background in STDD?




Melnik/Read/Maurer 2005:
Technology Experts' Perspective (N=12/42)

Melnik/Maurer/Chiasson 2006:
Technology Experts' Perspective (N=9/22)

Melnik 2006, 2007: 2 Field Studies



Acceptance Test Authoring

- Are **business experts** (pairs) on agile projects
- **capable** of
- **effectively** 
- **authoring** their **functional** requirements
- in the form of **executable acceptance tests (storytests)**
- and **communicating** those to the development team? 

“good” acceptance tests:

- credible;
- appropriate complexity;
- coverage of major functionality;
- easy to read.

Source: Melnik/Chiasson/Maurer 2006:
Business Experts' Perspective (N=9/18 + 9/22)

Acceptance Testing and Storytest-Driven Development

- Storytest-driven development stimulates **thinking** and is correlated with enhanced **communication** about requirements in software teams
- Executable acceptance test specifications can serve as sufficient evidence of requirements **traceability**
- **Weak tool support** presents a serious limitation and effects maintainability and scalability



Introducing p&p Acceptance Testing Guidance series

- 1) Acceptance testing fundamentals
- 2) Test automation and test patterns
- 3) Acceptance test-driven development – tool support
- <http://codeplex.com/TestingGuidance>
- Call for participation: Online Survey!

Summary (do, reflect, learn!)

- Welcome to the **mainstream!**
- **Experience reports** and case studies are valuable
- Teams practicing agile are getting **larger** and more **distributed**
- Agile teams report **higher job satisfaction**
- Agile is **not just developer-centric**
- Initial evidence of **rational design decision-making**
- **Testing to the forefront** of development
- Need to go beyond surveys and do more experimentation
- Do your own pilot projects to
 - generate support for future agile initiatives
 - learn what works
 - play with new ideas by encouraging innovation and risk taking

Agile 2008 World Conference
August 4-8, 2008
Toronto, Canada

*With
a
dedicated
French
track!!!*



www.agile2008.org

Q & A



grigori.melnik@microsoft.com

