Quantitative Software Management, Inc.
2000 Corporate Ridge, Suite 700
Mclean, VA 22102
703.790.0055 ● 703.749.3795 (fax)
info@qsm.com ● www.qsm.com

AGILE BY THE NUMBERS

Outline

- What are the core issues with software development and maintenance?
- Improvement measures (Silver bullets?)
- Why do software projects succeed or fail?
- Agile by the numbers
  - Is Agile a silver bullet
- Some problems in paradise
  - Agile issues

Core Issues with Software

- Cost, Schedule, Quality are hard to manage and are often unpredictable
- Frequently do not meet requirements

Why does this matter?

- Software is pervasive and life as we know it would cease without it
- Money. A huge cost component for business, government, military, communications, and our personal lives

Core Issues with Software

- What is the desired state for software?
  - Predictable
  - Meet requirements
  - Become more efficient over time (productivity improvement)
- New tools and improvement initiatives are best understood in this context

Improvement Initiatives

- Silver Bullet: A direct and effortless solution to a problem. An action that cuts through complexity and provides an immediate solution to a problem*

Some software improvement initiatives

- Structured programming
- 3gl/4gl languages
- Case tools
- Code generators
- CMMI
- Cloud computing
- GUI's
- OO Development
- ERP packages
- SOA
- Internet

Improvement Initiatives Classified

- Most measures aimed at software improvement have focused on tools, processes, or both.

Tools	Process
3/4 GL Languages	Structured Programming
Case Tools	CMMI
Code Generators	OO Programming
GUI's	ERP Packages
Internet	Internet
	SOA

Silver Bullets

- “There is no single development, in either technology or management technique, which by itself promises even one order of magnitude improvement within a decade in productivity, in reliability, in simplicity.”

  - Frederick Brooks in “No Silver Bullet – Essence and Accidents of Software Engineering”

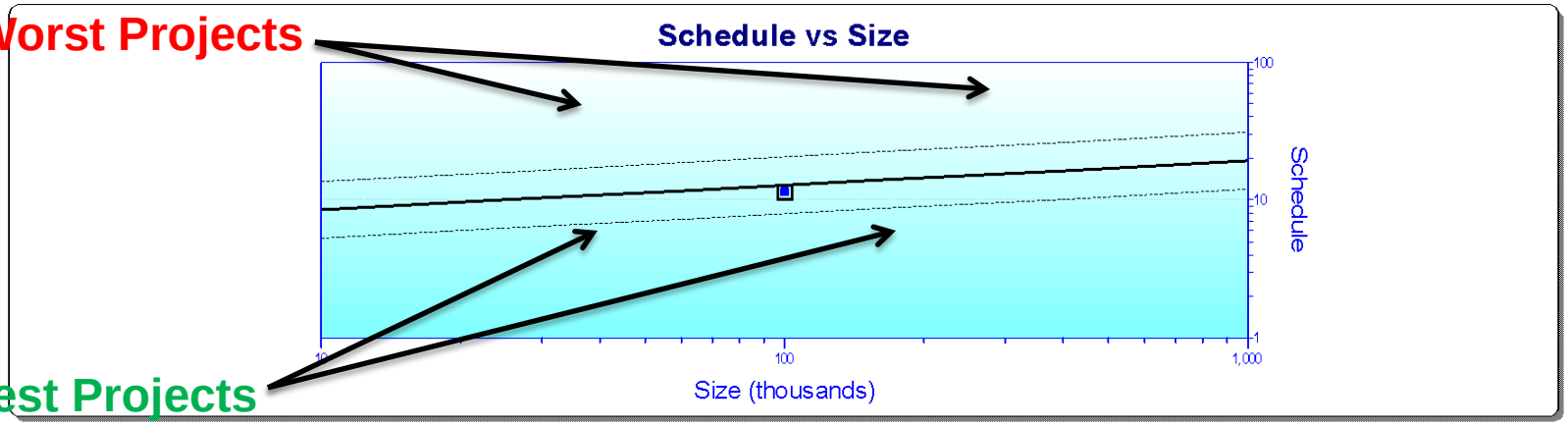
Success or Failure
Best and Worst Projects

- Two studies by author
  - 2006 IT projects
  - 2010 Engineering software projects
- Best projects defined as being one standard deviation (σ) better than average for both time to market (schedule) and effort expended
- Worst projects were one σ worse than average for both time to market and effort
- Projects evaluated on 58 criteria for Tools & Methods, Technical Complexity, Personnel, and Re-use

Best Project / Worst Projects

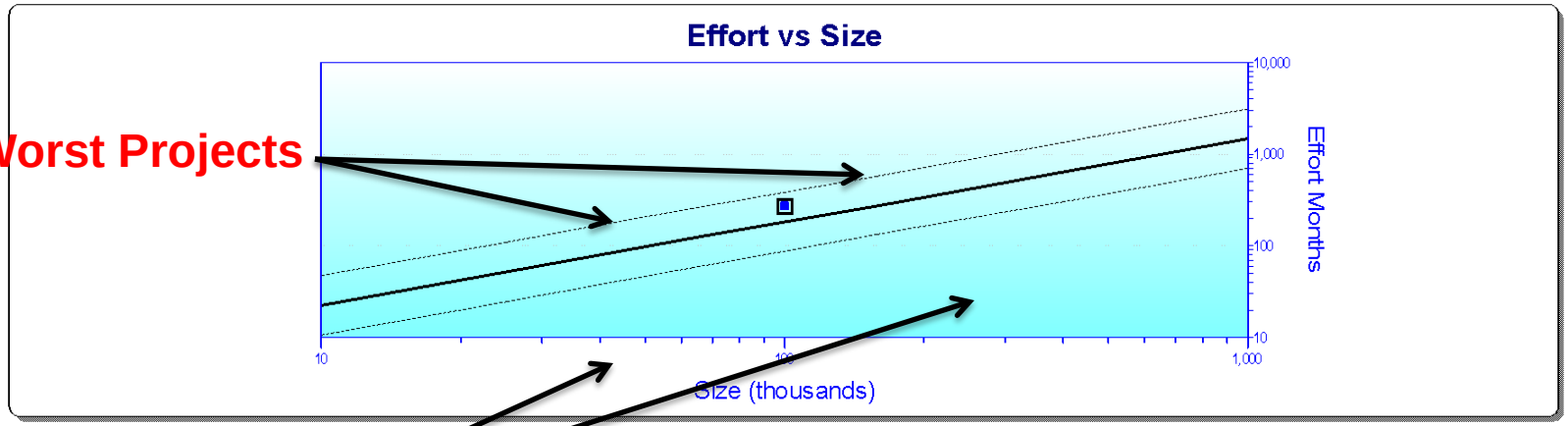
Validate Estimate with History

Worst Projects



Best Projects

Worst Projects



Best Projects

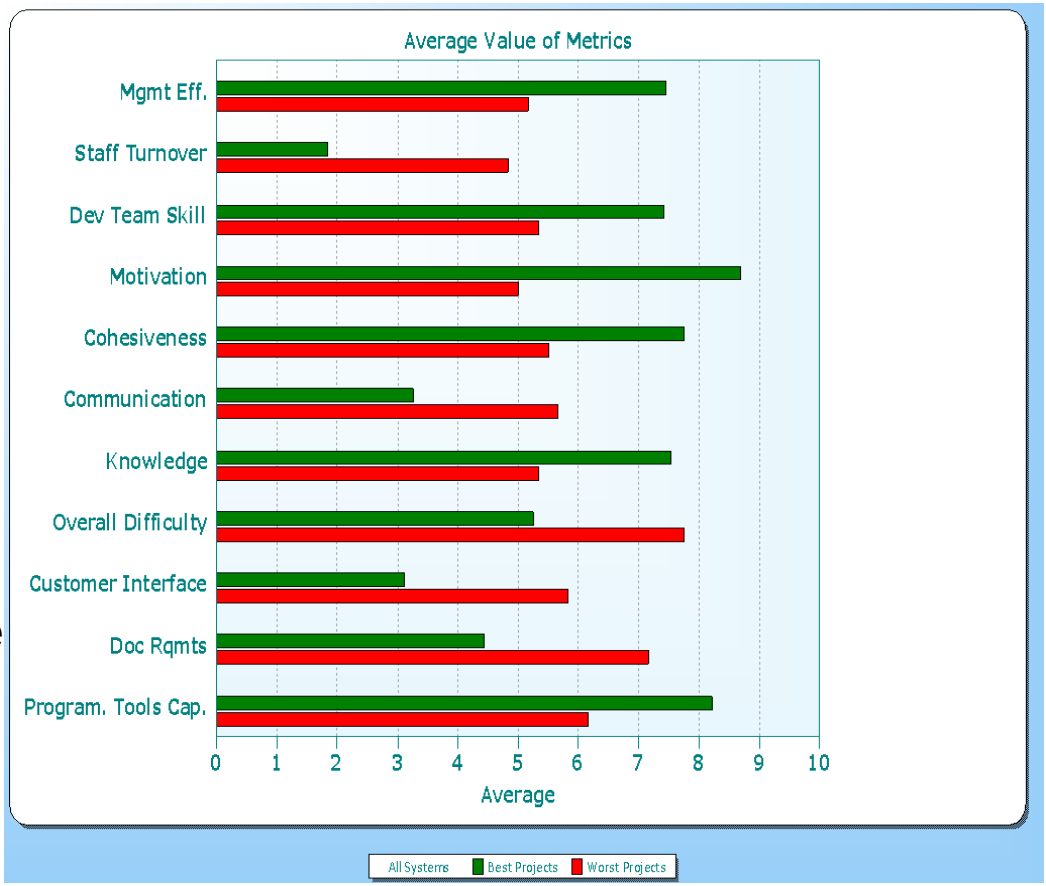
Current Solution | Historical Projects | QSM C&C | Avg. Line Style | 1 Sigma Line Style | Project: Quality Demo

Differentiators

People, Communication, Knowledge
- Management Efficiency
- Staff Turnover
- Team Skill
- Motivation
- Cohesiveness
- Communication
- Knowledge

Complexity
- Overall Difficulty
- Customer Interface
- Documentation Rqmts

Tools
- Programming Tools Capability



Things that Don't Matter

Data Complexity

Integration Complexity

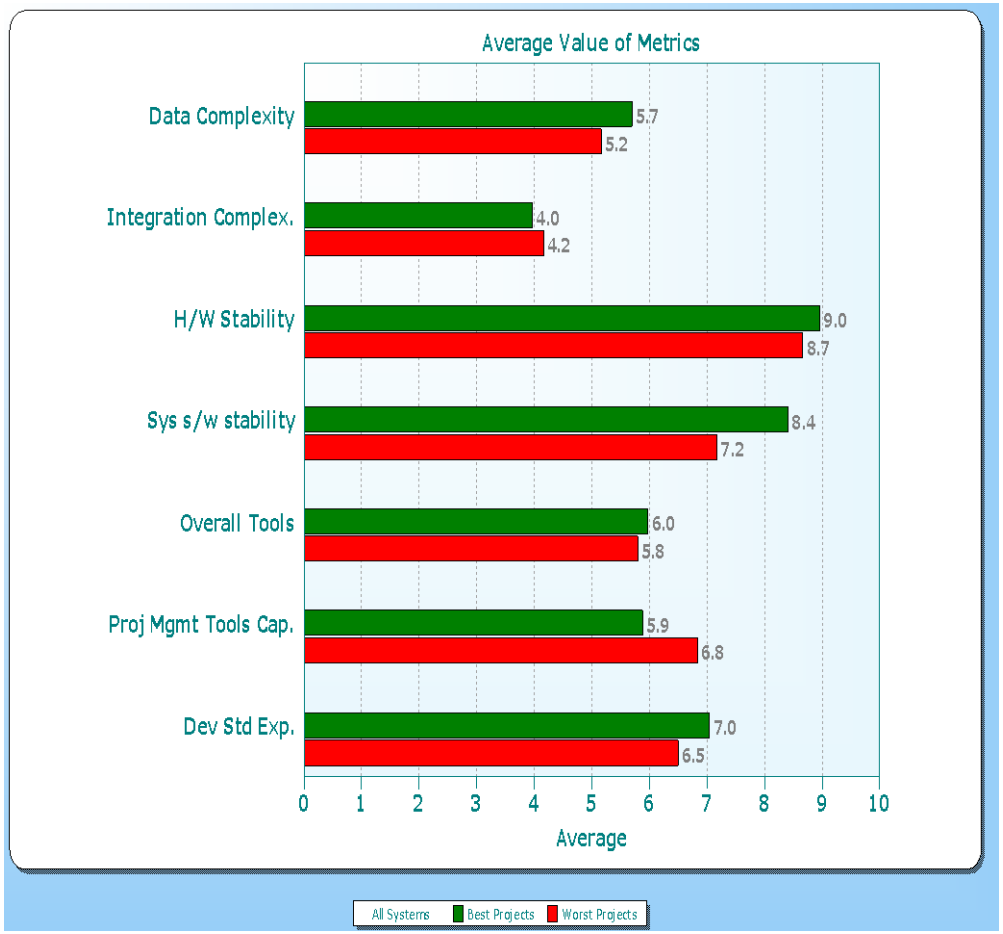
Hardware Stability

System Software Stability

Overall Tools Capability

Project Mgt Tools Capability

Development Standards Experience



Best Projects / Worst Projects Results

- Results from both the IT and Engineering projects were very similar
- The biggest differentiators between productive and unproductive projects were in the areas of people, communication, and knowledge
- Many project improvement efforts focus on tools and processes
- An interesting tidbit: Project software languages were not correlated with either Best or Worst projects

The Promise of Agile: Agile Manifesto

- Individuals and Interactions over processes and tools
- Working Software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan
- Key traits
  - Frequent delivery
  - Business people and developers work together daily
  - Face to face conversations

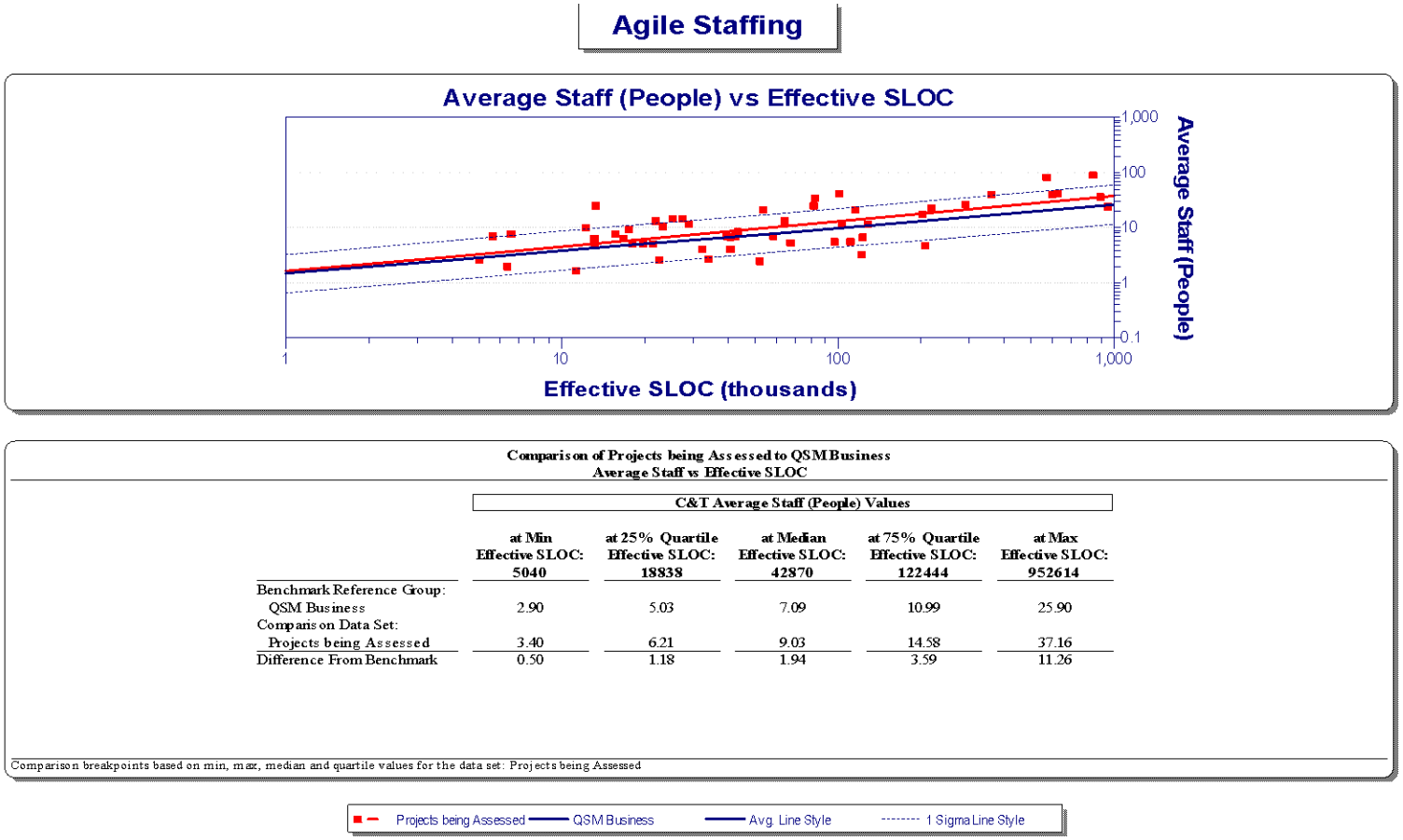
The Promise of Agile

- It appears that Agile development embraces the People, Knowledge, and Communication traits that were found in highly successful projects
- Agile is very focused on the social component of software development
- So, how well do Agile projects compare to traditional development?

Demographics

- 64 recently completed Agile projects
- 12 different companies
- 87% business, 7% scientific applications, 6% system software
- Team size clustered in 5-10 and 20-50 ranges
- Median size 42.9k lines of code
- Median effort 47 staff months
- Median staff 7.5
- Median duration 6.1 months
- Principally new development and major enhancements

Agile Staffing



Comparison of Projects being Assessed to QSM Business
Average Staff vs Effective SLOC

	C&T Average Staff (People) Values				
	at Min Effective SLOC: 5040	at 25% Quartile Effective SLOC: 18838	at Median Effective SLOC: 42870	at 75% Quartile Effective SLOC: 122444	at Max Effective SLOC: 952614
Benchmark Reference Group:					
QSM Business	2.90	5.03	7.09	10.99	25.90
Comparison Data Set:					
Projects being Assessed	3.40	6.21	9.03	14.58	37.16
Difference From Benchmark	0.50	1.18	1.94	3.59	11.26

Comparison breakpoints based on min, max, median and quartile values for the data set: Projects being Assessed

Projects being Assessed — QSM Business — Avg. Line Style ------ 1 Sigma Line Style

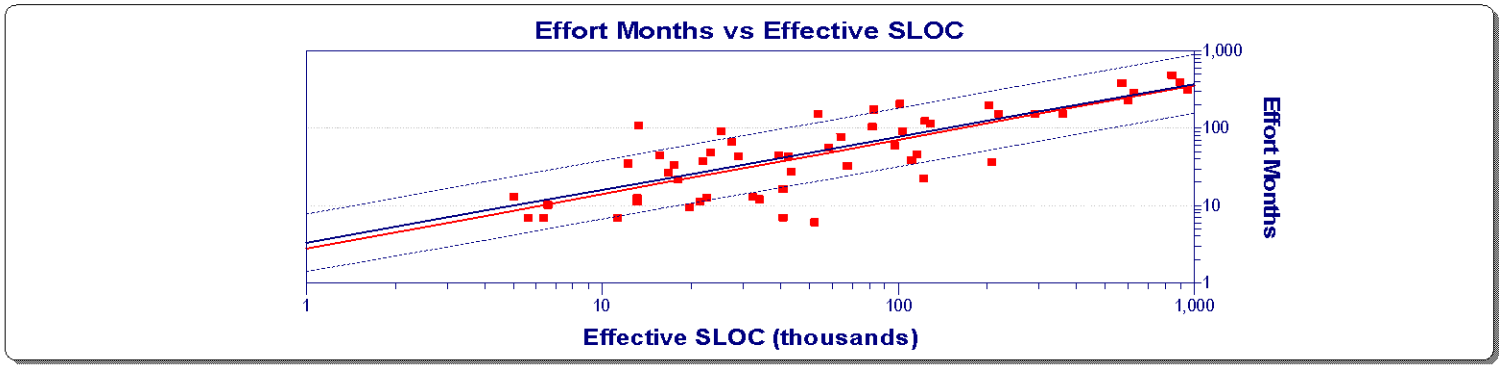
The blue trend lines in this and subsequent graphs are the QSM business average with plus & minus 1 standard deviation. The red line is the Agile dataset average

Agile Staffing Observations

- The agile projects use slightly more staff than non-agile business projects although the trend is very similar

Agile Effort

Agile Effort



Comparison of Projects being Assessed to QSM Business
Effort Months vs Effective SLOC

	C&T Effort (PM) Values				
	at Min Effective SLOC: 5040	at 25% Quartile Effective SLOC: 18838	at Median Effective SLOC: 42870	at 75% Quartile Effective SLOC: 122444	at Max Effective SLOC: 952614
Benchmark Reference Group: QSM Business	9.99	24.63	43.24	88.68	361.08
Comparison Data Set: Projects being Assessed	8.63	21.85	39.01	81.74	347.02
Difference From Benchmark	-1.36	-2.78	-4.23	-6.94	-14.06

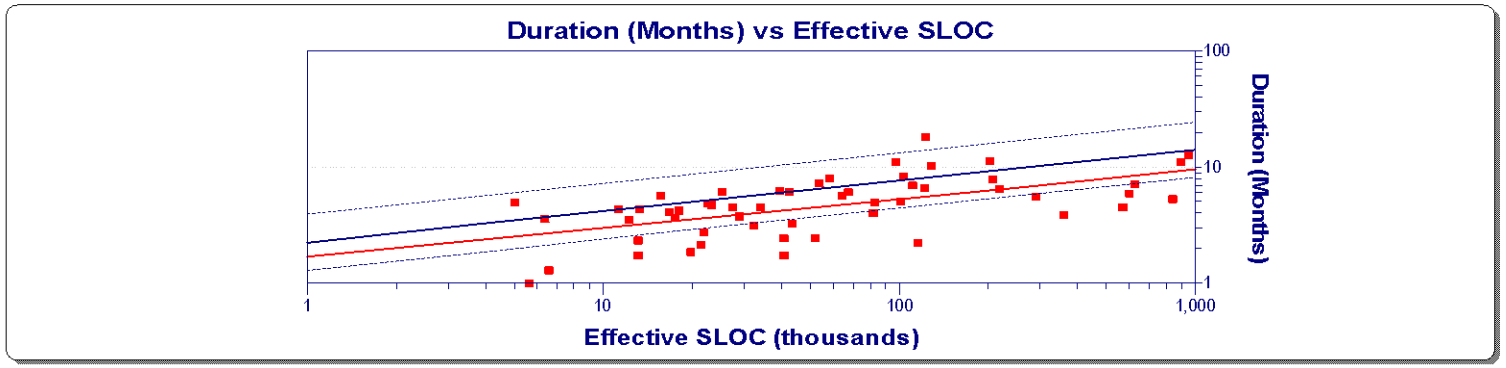
Comparison breakpoints based on min, max, median and quartile values for the data set: Projects being Assessed

Projects being Assessed — QSM Business — Avg. Line Style -------- 1 Sigma Line Style

Agile and non-Agile projects use nearly the same amount of project effort for projects with similar amounts of delivered functionality

Agile Schedule Length

Agile Schedule Duration



Comparison of Projects being Assessed to QSM Business
Duration (Months) vs Effective SLOC

	C&T Duration (Months) Values				
	at Min Effective SLOC: 5040	at 25% Quartile Effective SLOC: 18838	at Median Effective SLOC: 42870	at 75% Quartile Effective SLOC: 122444	at Max Effective SLOC: 952614
Benchmark Reference Group:					
QSM Business	3.45	4.90	6.10	8.07	13.94
Comparison Data Set:					
Projects being Assessed	2.54	3.52	4.32	5.61	9.34
Difference From Benchmark	-0.91	-1.38	-1.78	-2.46	-4.60

Comparison breakpoints based on min, max, median and quartile values for the data set: Projects being Assessed

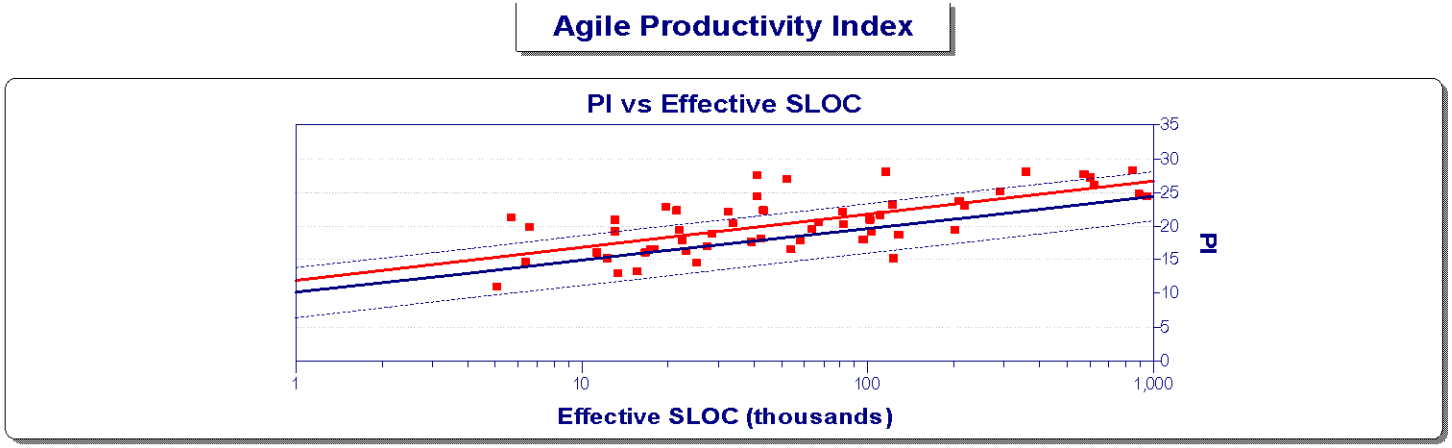
Projects being Assessed — QSM Business — Avg. Line Style -------- 1 Sigma Line Style

Agile projects complete much more rapidly

Agile Schedule Observations

- Agile projects complete much more quickly than non-agile projects while expending about the same amount of effort (Cost)
- Since schedule is frequently an important project driver, this is a significant advantage

Agile Productivity Index (PI)



Comparison of Projects being Assessed to QSM Business
PI vs. Effective SLOC

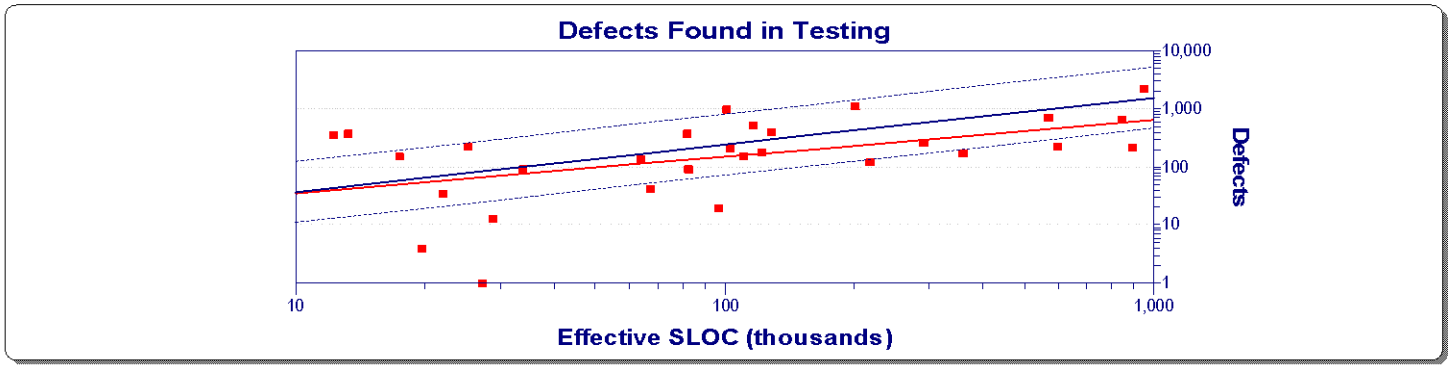
	PI Values				
	at Min Effective SLOC: 5040	at 25% Quartile Effective SLOC: 18838	at Median Effective SLOC: 42870	at 75% Quartile Effective SLOC: 122444	at Max Effective SLOC: 952614
Benchmark Reference Group: QSM Business	13.50	16.22	17.92	20.08	24.32
Comparison Data Set: Projects being Assessed	15.38	18.19	19.93	22.17	26.53
Difference From Benchmark	1.88	1.97	2.02	2.08	2.21

Comparison breakpoints based on min, max, median and quartile values for the data set: Projects being Assessed

Productivity indices for Agile projects were significantly higher than the business average

Agile Quality

Agile Quality



Comparison of Projects being Assessed to QSM Business
Errors (SysInt-Del) vs. Effective SLOC

	Errors (SysInt-Del) Values				
	at Min Effective SLOC: 12240	at 25% Quartile Effective SLOC: 28013	at Median Effective SLOC: 101274	at 75% Quartile Effective SLOC: 254563	at Max Effective SLOC: 952614
Benchmark Reference Group: QSM Business	44.08	86.29	244.75	516.94	1507.89
Comparison Data Set: Projects being Assessed	39.64	67.04	151.55	272.02	628.54
Difference From Benchmark	-4.44	-19.25	-93.20	-244.92	-879.35

Comparison breakpoints based on min, max, median and quartile values for the data set: Projects being Assessed

Projects being Assessed — QSM Business — Avg. Line Style -------- 1 Sigma Line Style

Agile projects produced fewer defects

In Summary

Typical Sized Agile and Business IT Projects				
	Agile	Business IT	Difference	%Difference
Size in SLOC	42,900	42,900		
Average Staff	9	7.1	1.9	26.8%
Devel. Duration (Mths)	4.3	6.1	-1.8	-29.5%
Effort Months	39	43	-4.0	-9.3%
Defects (testing)	152	245	-93.0	-38.0%
Productivity Index	19.93	17.92	2.0	11.2%

- Agile projects outperform conventional development in Productivity, Quality, and Time to Market
- Staffing levels are higher; but overall effort is slightly lower while achieving significant schedule compression

Some Problems in Paradise

- Large projects require more process formality
  - Change control & Configuration Management
- Regulatory environment may not be compatible with Agile
- Legal requirements & corporate/enterprise requirements
- Minimum marketable features may be very large on big projects
- Budget and schedule constraints are real and legitimate

Summary

- Agile is an effective software development strategy
  - Particularly effective at compressing schedule on small to medium size projects
  - Lower defect levels
- Requires investment in training and practice
- Agile is not a panacea for all software development issues
- A good choice; but not for every situation

Questions?

Contact Information

Donald Beckett,
Principal Consultant
Quantitative Software Mgt.
don_beckett@qsm.com
T: 360-638-0097
C: 703-785-1408