Business Value of Lean Thinking

Capitalizing Upon Lean Thinking Principles to Rapidly Create Innovative Products & Services

Dr. David F. Rico, PMP, CSEP, EBAS, BAF, FCP, FCT, ACP, CSM, SAFe, DevOps, AWS

Website: http://davidfrico.com ● LinkedIn: http://linkedin.com/in/davidfrico ● Twitter: @dr_david_f_rico

Agile Cost of Quality: http://www.davidfrico.com/agile-vs-trad-coq.pdf
DevOps Return on Investment (ROI): http://davidfrico.com/rico-devops-roi.pdf

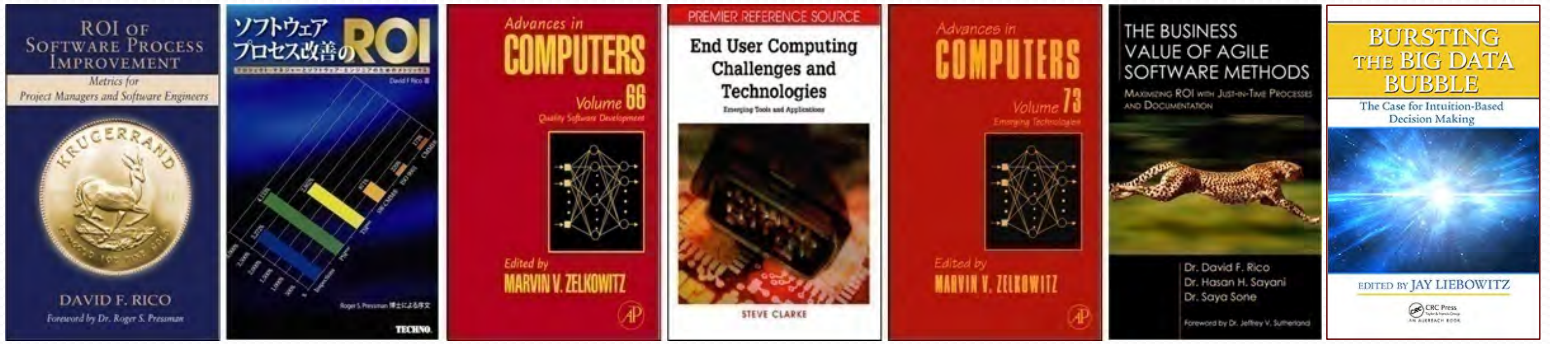
Dave's NEW Business Agility Video: http://www.youtube.com/watch?v=hTvtsAkL8xU
Dave's NEWER Scaled Agile Framework SAFe 4.5 Video: http://youtu.be/1TAuCRq5a34
Dave's NEWEST Development Operations Security Video: http://youtu.be/qrWRoXSS9bs
Dave's BRAND-NEW ROI of Lean Thinking Principles Video: http://youtu.be/wkMfaPAxO6E
Dave's REALLY-NEW ROI of Evolutionary Design Principles Video: http://youtu.be/TcXI26ClRb0
Dave's EXTREMELY-NEW ROI of Organizational Agility Principles Video: http://youtu.be/HOzDM5krtes

DoD Fighter Jets versus Amazon Web Services: http://davidfrico.com/dod-agile-principles.pdf
Principles of Collaborative Contracts: http://davidfrico.com/collaborative-contract-principles.pdf
Principles of Lean Organizational Leadership: http://davidfrico.com/lean-leadership-principles.pdf
Principles of Evolutionary Architecture: http://davidfrico.com/evolutionary-architecture-principles.pdf
Principles of CI, CD, & DevOps - Development Operations: http://davidfrico.com/devops-principles.pdf
Principles of SAFe Transformations - Scaled Agile Framework: http://davidfrico.com/safe-principles.pdf
Principles of Maximizing SAFe ROI - Scaled Agile Framework: http://davidfrico.com/safe-roi-principles.pdf
Principles of Lean-Agile - Contract Statements of Work (SOW): http://davidfrico.com/agile-sow-principles.pdf

Economic Value of Agile Businesses, Enterprises & Organizations - http://davidfrico.com/value-of-business-agility.pdf

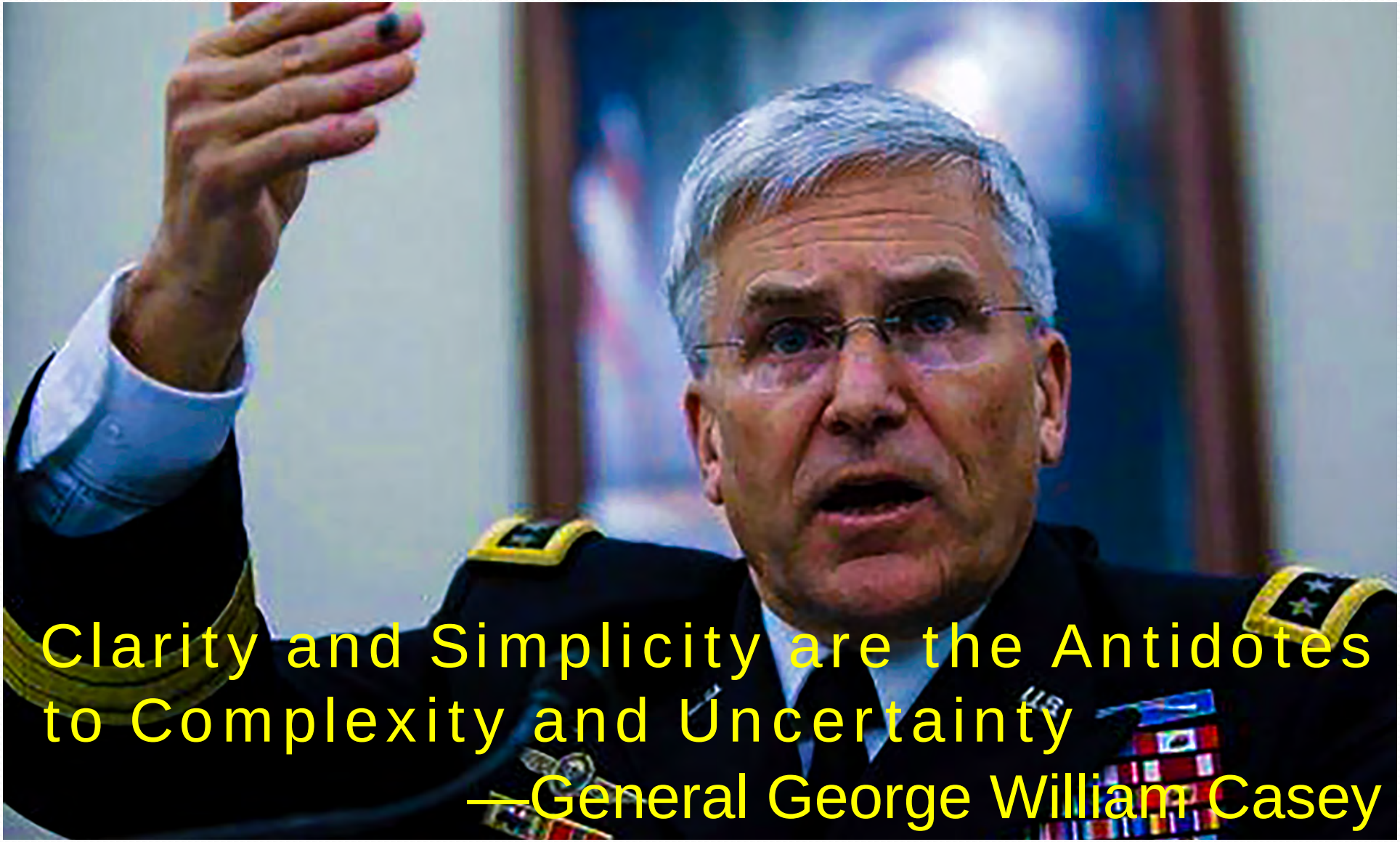
Author Background

- Gov't contractor with 39+ years of IT experience
- B.S. Comp. Sci., M.S. Soft. Eng., & D.M. Info. Sys.
- Large gov't projects in U.S., Far/Mid-East, & Europe



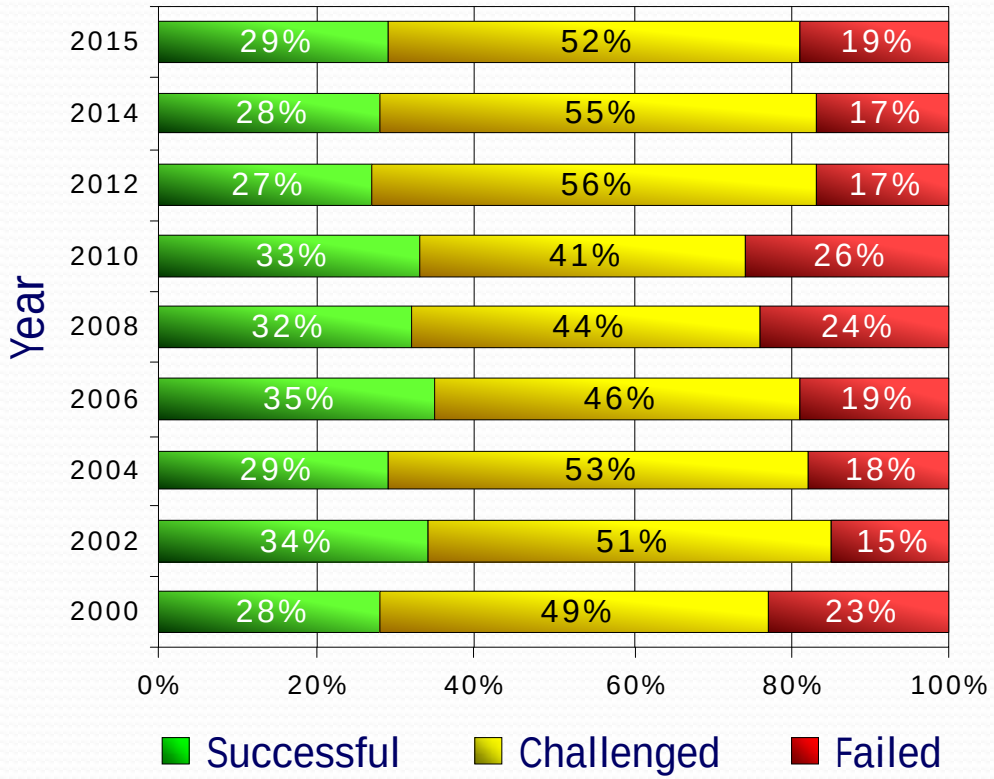
- ✓ Career IT project management, systems and software engineering PROCESS coach.
- ✓ Supported numerous billion-dollar enterprise digital transformation initiatives for 35+ years.
- ✓ Clients multi-billion government agencies, Fortune 500 conglomerates, and international IT firms.
- ✓ Included NASA's Space Station, Japanese Firms, Navy Fighters, NRO Satellites, and Intel Clouds, etc.
- ✓ Supported Digital Transformations at leading energy, healthcare, financial, and DoD enterprises and firms.
- ✓ Supported virtual casefile systems, data warehouses, data lakes, cloud migrations, and enterprise architectures.
- ✓ Specialized in Lean, Agile, Scrum, Scaled Agile Framework (SAFe), CI, CD, DevOps, DevSecOps, and Cloud Computing.
- ✓ Quickstart SAFe rollouts for critical portfolios, solutions, programs, projects, and new product development initiatives.
- ✓ Provides one-on-one and small group coaching services for C-levels, directors, managers, tech leaders, and developers.
- ✓ Skills include Lean, Agile, Scrum, SAFe, DevSecOps, Agile assessments, metrics, toolsets, dashboards, and case studies.
- ✓ Public speaker, author, blogger, trainer and holds over 13 professional certifications including SAFe SPC 5.0 and AWS CCP.
- ✓ Supported HHS, CMS, IRS, Exelon, ODNI IC-CIO, Intel, DoD, DoJ, USPS, NASA, DARPA, DISA, U.S. Air Force, Army, and Navy.

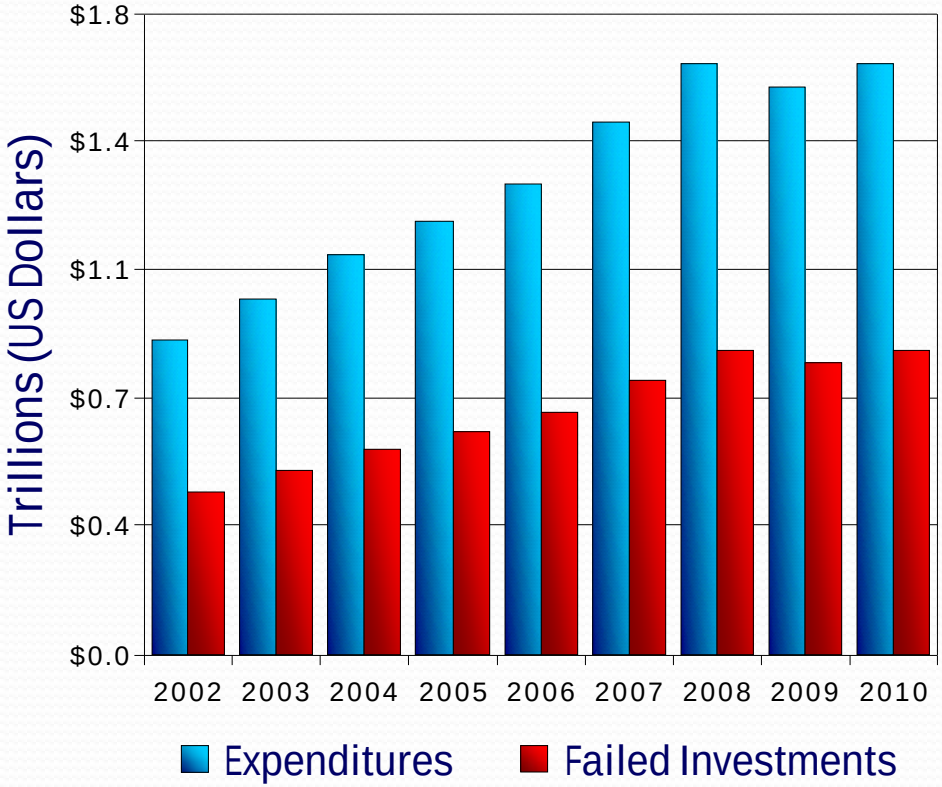
General Casey on Lean Thinking



Traditional Project Failure Rates

- Challenged and failed projects hover at 67%
- Big projects fail more often, which is 5% to 10%
- Of $1.7T spent on IT projects, over $858B were lost



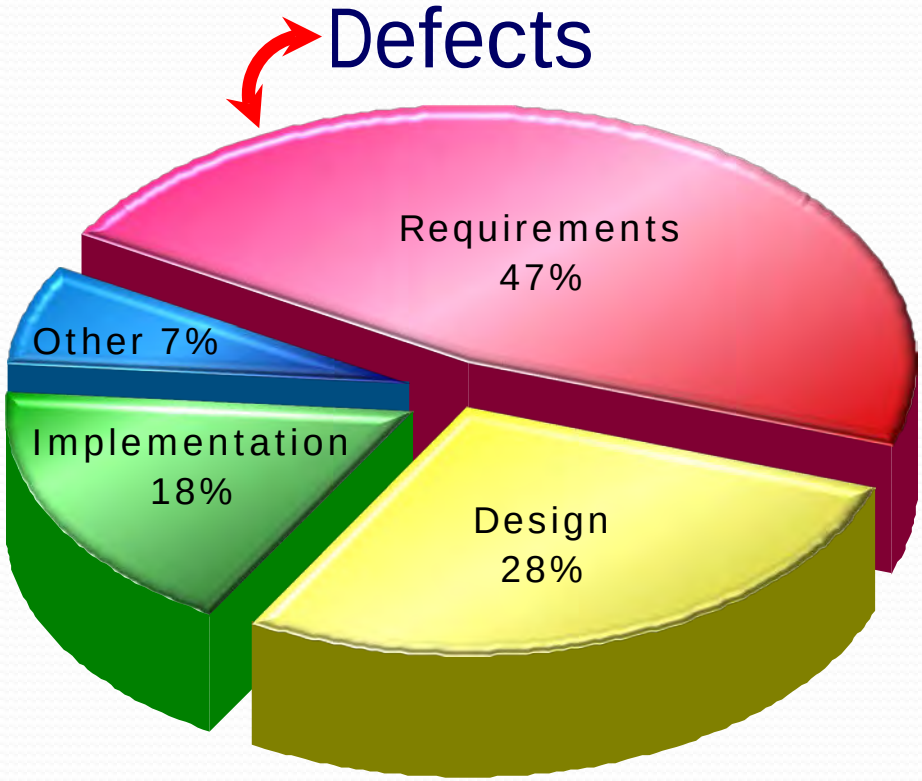


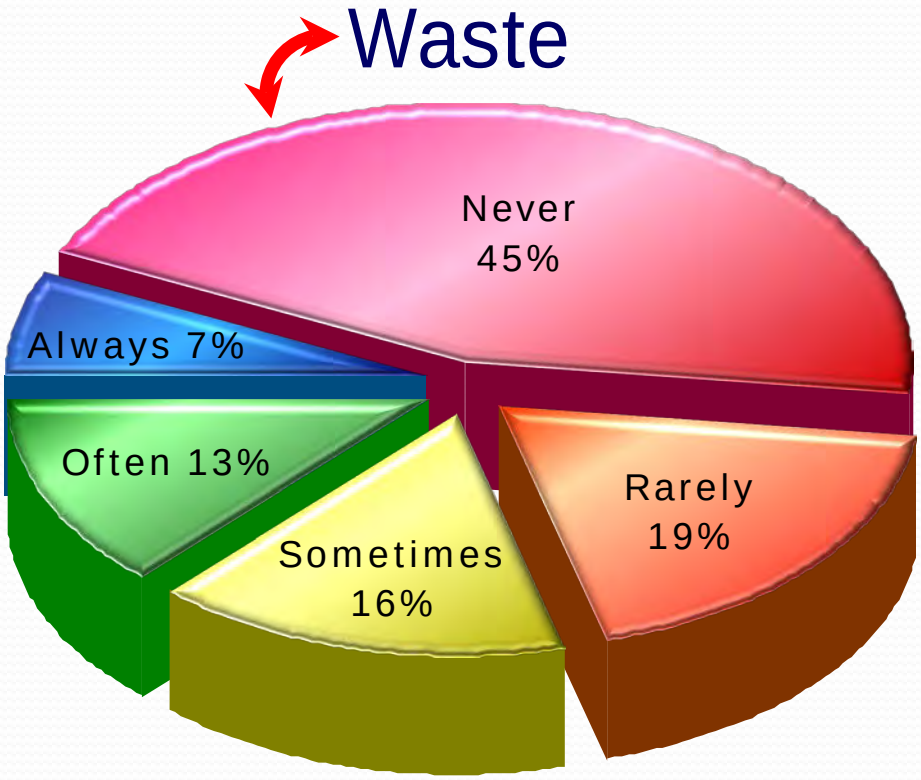
Standish Group. (2015). *Chaos summary 2015*. Boston, MA: Author.

Sessions, R. (2009). *The IT complexity crisis*: *Danger and opportunity*. Houston, TX: Object Watch.

Traditional Defects & Waste

- Requirements defects are #1 reason projects fail
- Traditional projects specify too many requirements
- More than 65% of requirements are never used at all

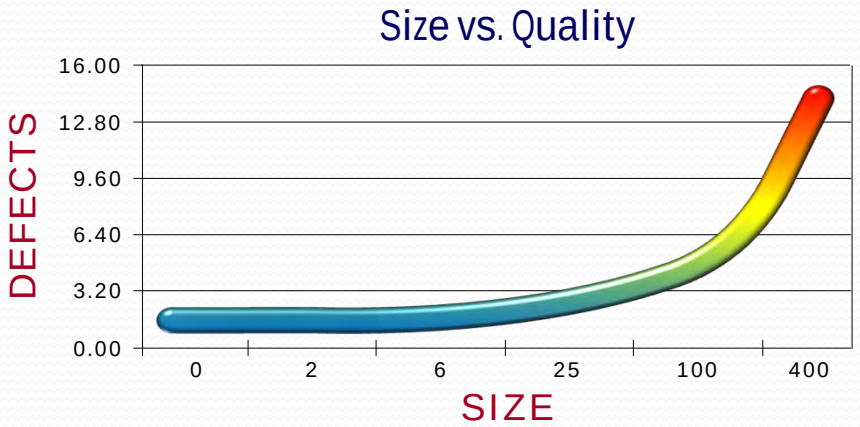


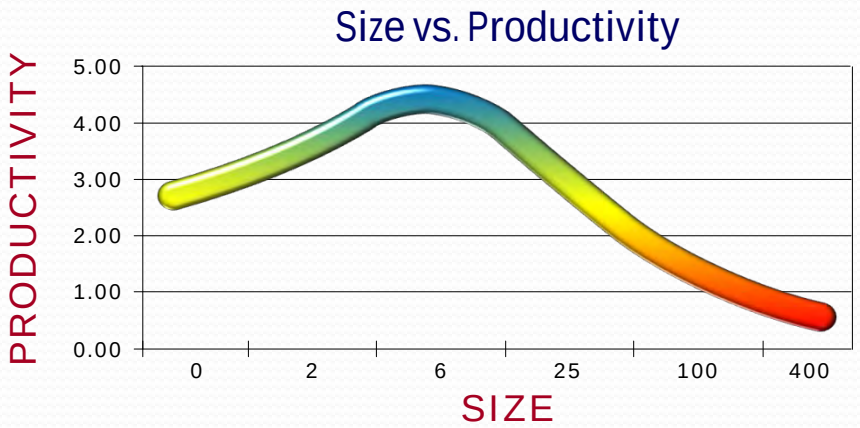


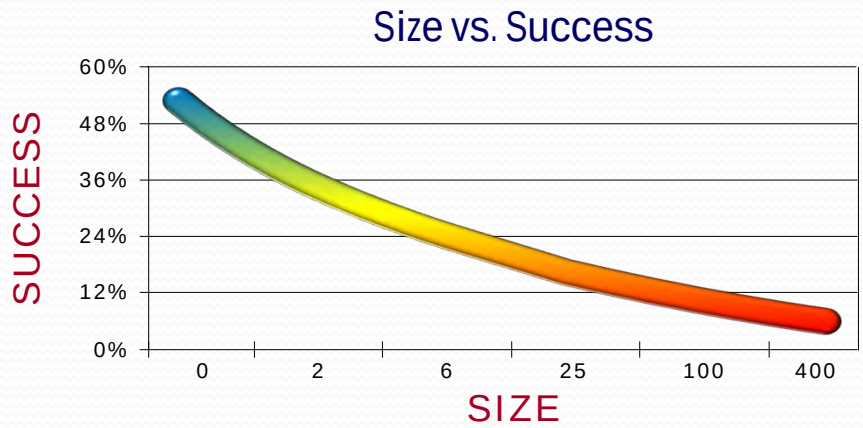
Sheldon, F. T. et al. (1992). Reliability measurement: From theory to practice. *IEEE Software, 9*(4), 13-20
Johnson, J. (2002). ROI: It's your job. *Extreme Programming 2002 Conference, Alghero, Sardinia, Italy*.

Size & Complexity vs. Performance

- Big projects result in poor quality and scope changes
- Productivity declines with long queues/wait times
- Large projects are unsuccessful or canceled



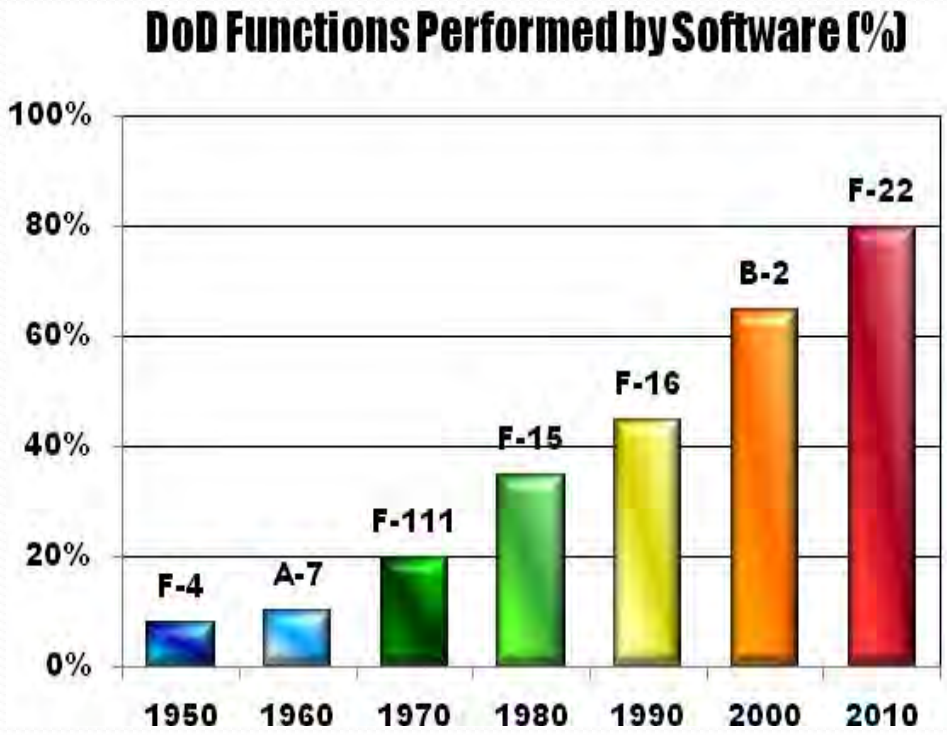


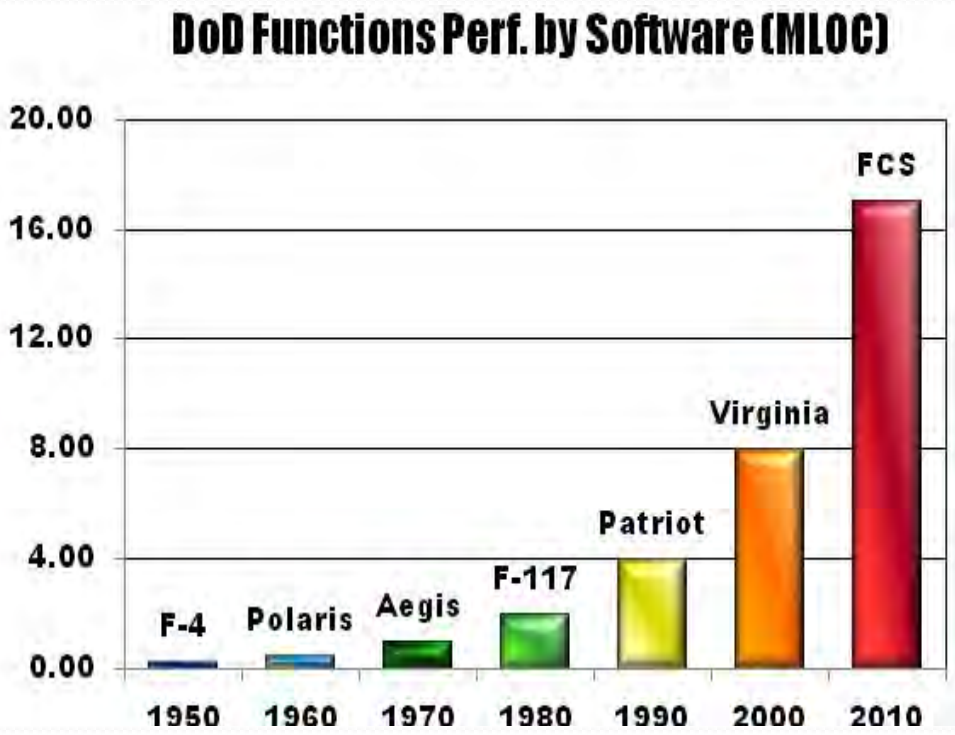


Jones, C. (1991). *Applied software measurement*: *Assuring productivity and quality*. New York, NY: McGraw-Hill.

Complexity U.S. DoD Systems

- No. of software-intensive systems is growing
- 80% of US DoD functions performed in software
- ☞ Major driver of cost, schedule, & tech. performance





Kennedy, M. P., & Umphress, D. A. (2011). An agile systems engineering process: The missing link. *Crosstalk*, *24*(3), 16-20.

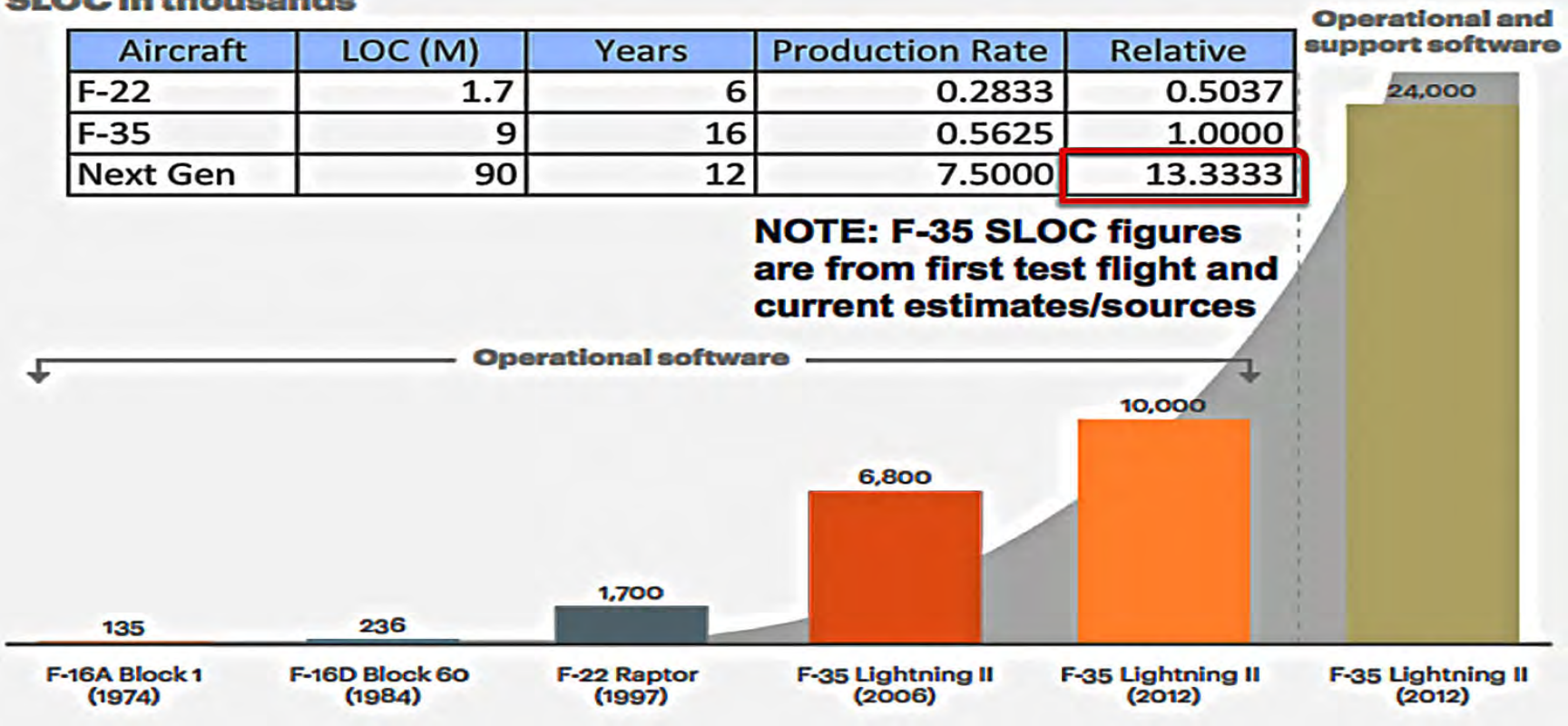
Complexity in U.S. DoD Fighter Jets

- Software in U.S. DoD avionics growing exponentially
- 10x growth from F-16 to F-22 (& another 10x to F-35)
- Productivity must grow by 10x for next gen systems

SLOC in thousands

Aircraft	LOC (M)	Years	Production Rate	Relative
F-22	1.7	6	0.2833	0.5037
F-35	9	16	0.5625	1.0000
Next Gen	90	12	7.5000	13.3333

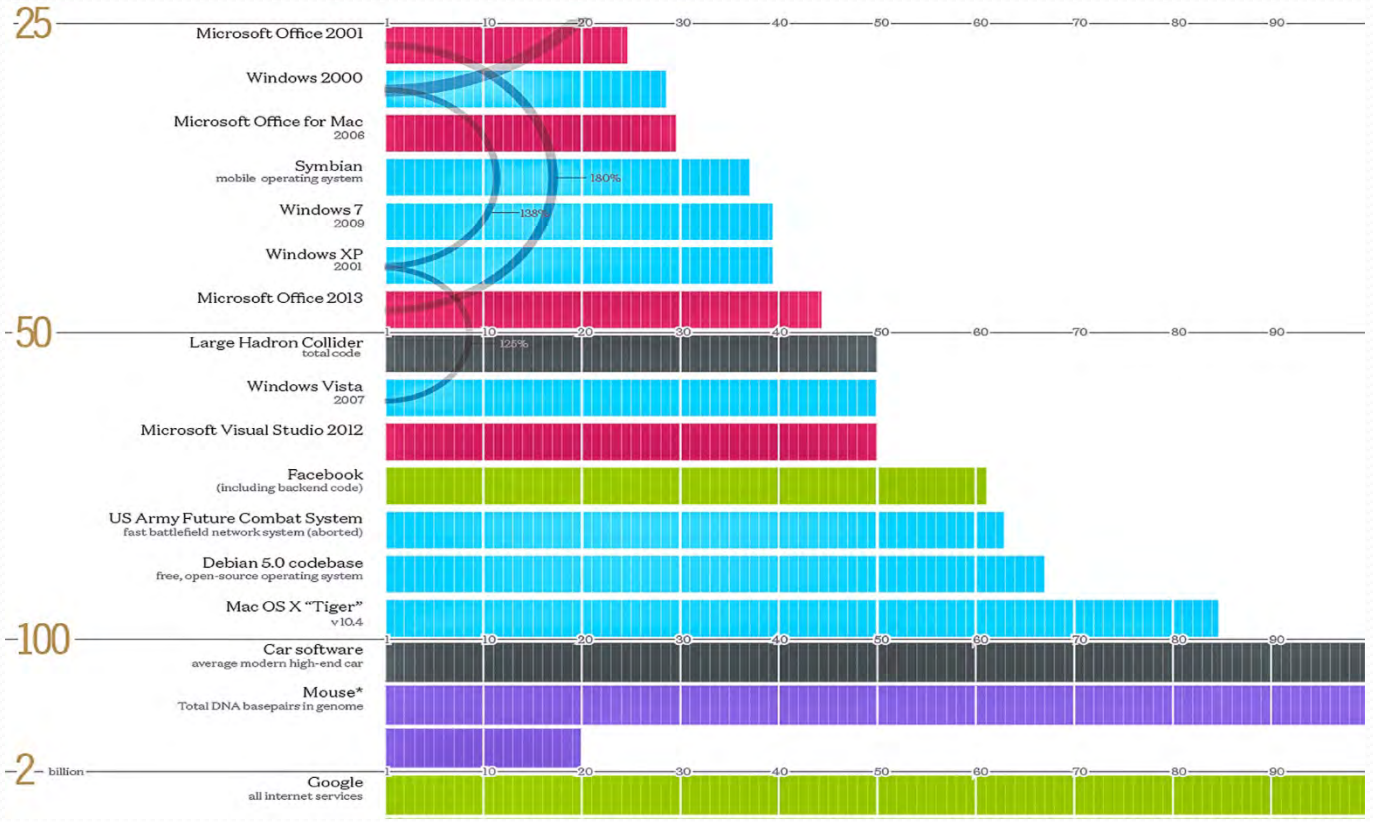
NOTE: F-35 SLOC figures are from first test flight and current estimates/sources



Blackburn, M. R. (2014). *Transforming systems engineering through a holistic approach to model centric engineering*. Washington, DC: Stevens Institute of Technology.

Complexity in Software Systems

- Software systems increased to billions of lines of code
- Software systems will grow to trillions of lines of code
- Productivity must increase by 1,000x in 21st century



McCandless, D. (2015). *Codebases*: *Millions of lines of code*. Retrieved January 26, 2020, from https://informationisbeautiful.net/visualizations/million-lines-of-code

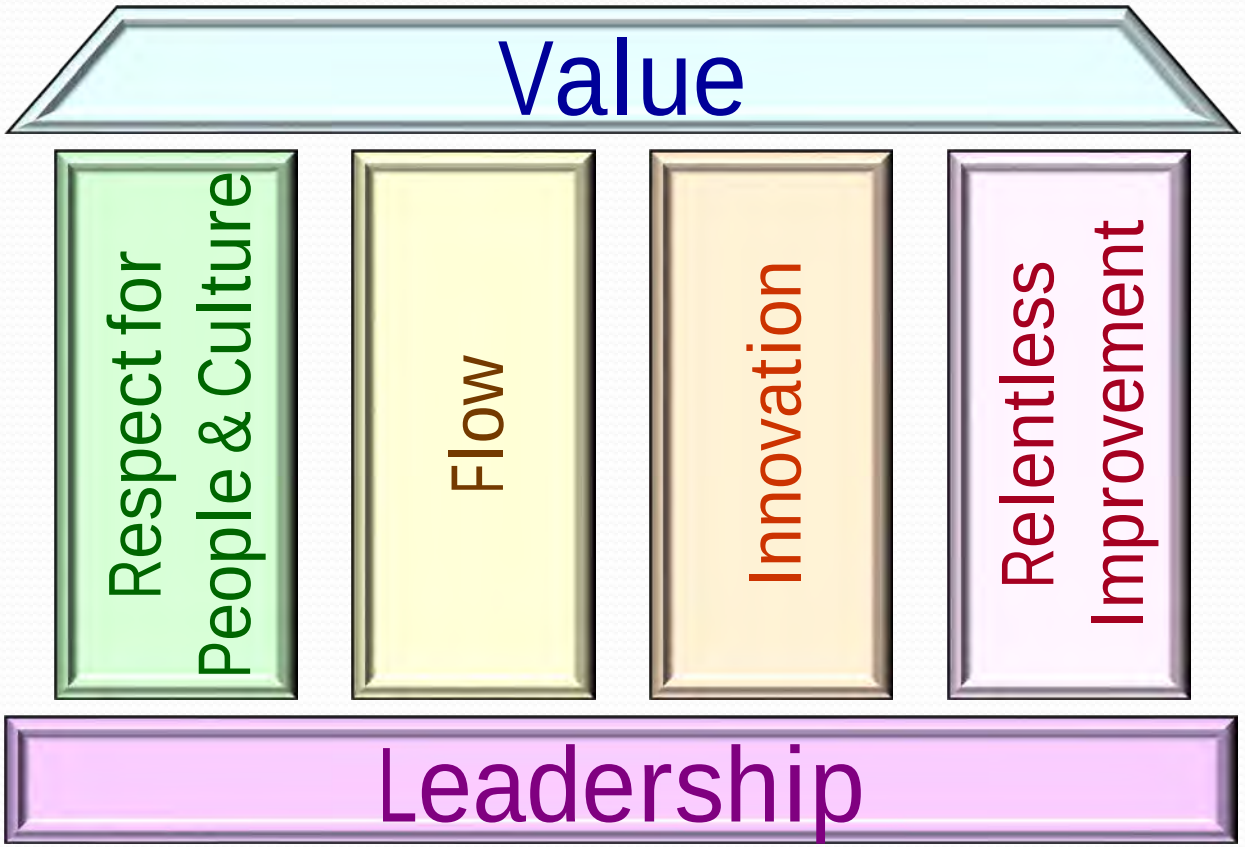
What is Lean Thinking

- Lean (lēn): Property consisting of being thinness, slimness, and skinniness; To be extremely slender
  - *A* customer-driven *product development process that delivers the maximum amount of* business value
  - *An economical way of* planning *and* managing *the development of complex new products and services*
  - *A product development process that is* free of excess waste, *capacity, and non-value adding activities*
  - Just-enough, *just-in-time, and right-sized product development* processes, documentation, *and* tools
  - *A product development approach that is* **ADAPTABLE TO CHANGE** *in customer needs and market conditions*

Womack, J. P., & Jones, D. T. (1996). *Lean thinking*: *Banish waste and create wealth in your corporation*. New York, NY: Free Press.

What is the Lean Value System

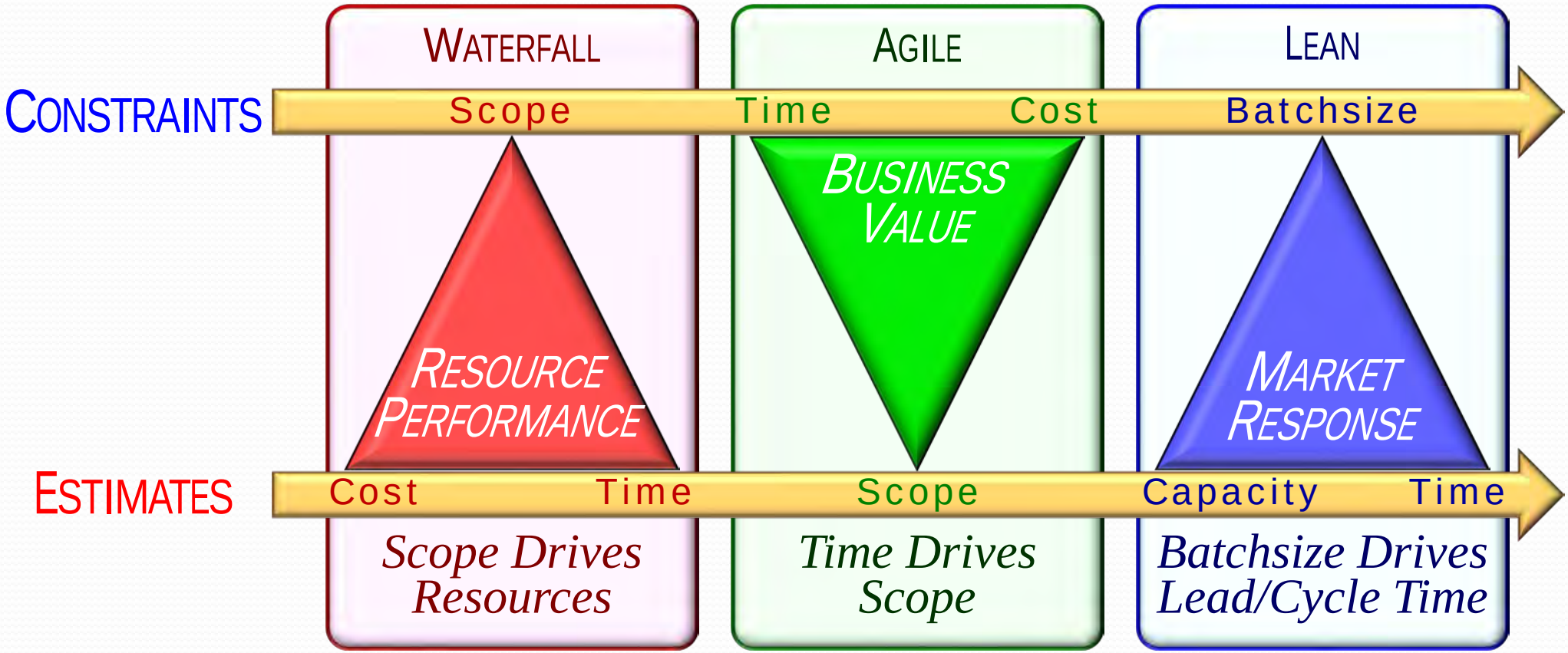
- Time-centric way to compete on speed & time
- Customer-centric model to optimize cost & quality
- Pull-centric alternative to wasteful mass production



Leffingwell, D. (2017). *The SAFe house of lean*. Retrieved February 19, 2018, from http://www.scaledagileframework.com

Lean Thinking Goldilocks Zone

- Traditional project management is scope-based
- Agile project management is primarily time-based
- ☞ Batchsize, capacity, & time key to market response



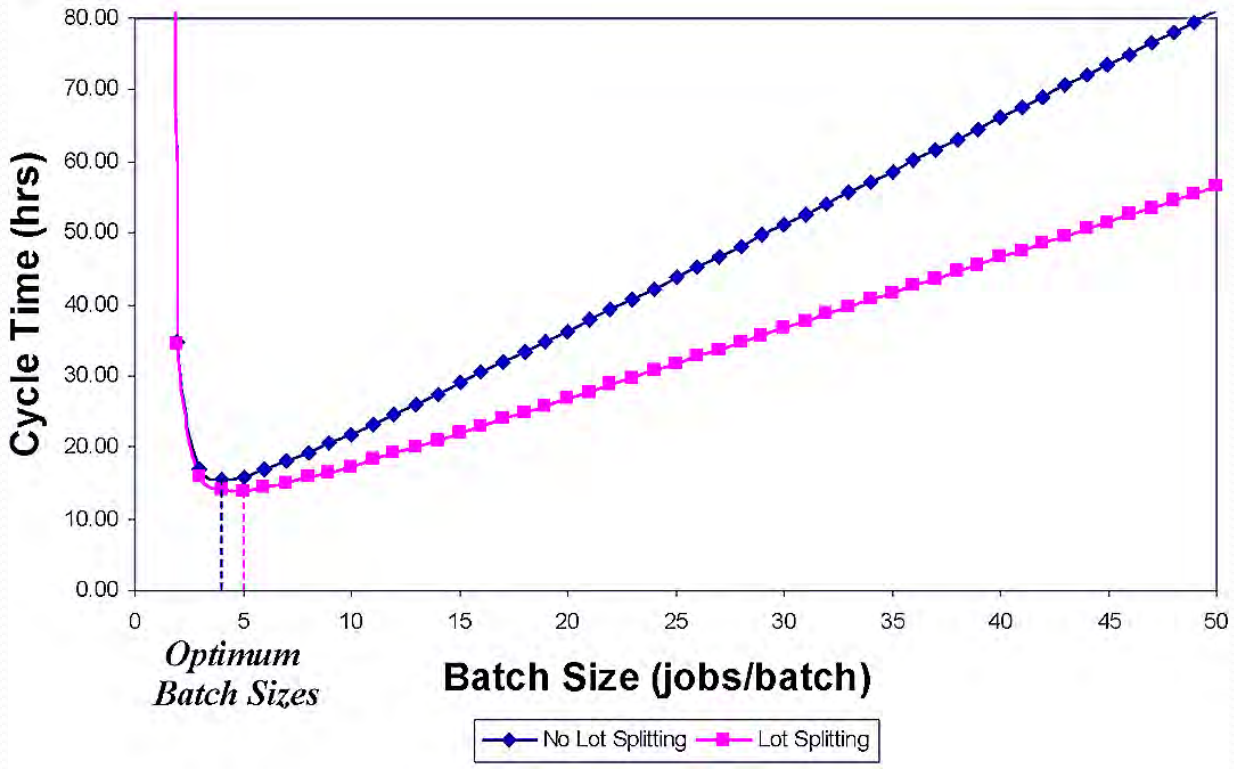
Rico, D. F. (2017). *Lean triangle: Triple constraints*. Retrieved December 17, 2017, from http://davidfrico.com/lean-triangle.pdf

Sylvester, T. (2013). *Waterfall, agile, and the triple constraint*. Retrieved December 16, 2017, from http://tom-sylvester.com/lean-agile/waterfall-agile-the-triple-constraint

Pound, E. S., Bell, J. H., Spearman, M. L. (2014). *Factory physics*: *How leaders improve performance in a post-lean six sigma world*. New York, NY: McGraw-Hill Education.

Batchsize vs. Cycle Times

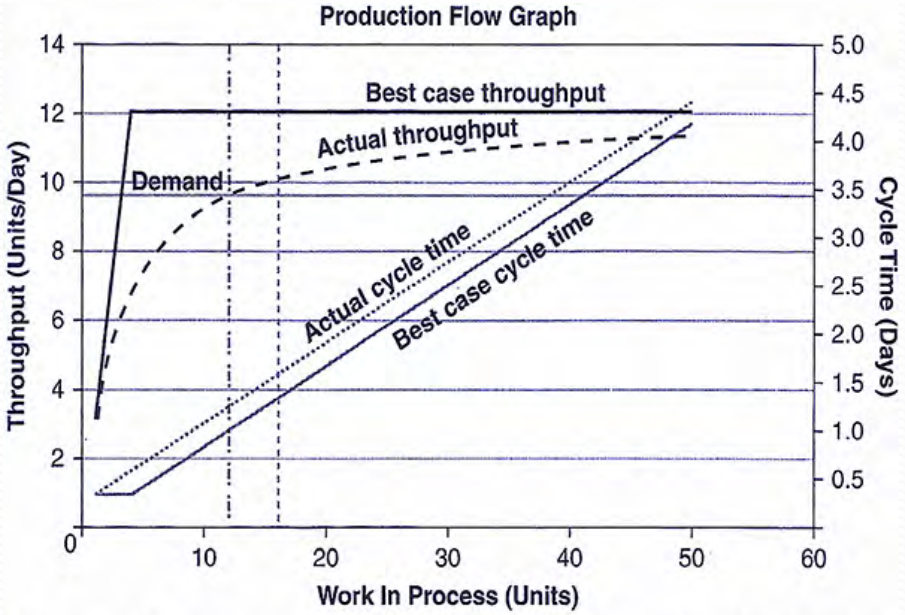
- Increasing batchsizes elongates cycle times
- Splitting batchsizes decreases cycle times a bit
- Decrease or split batchsizes to decrease cycle times



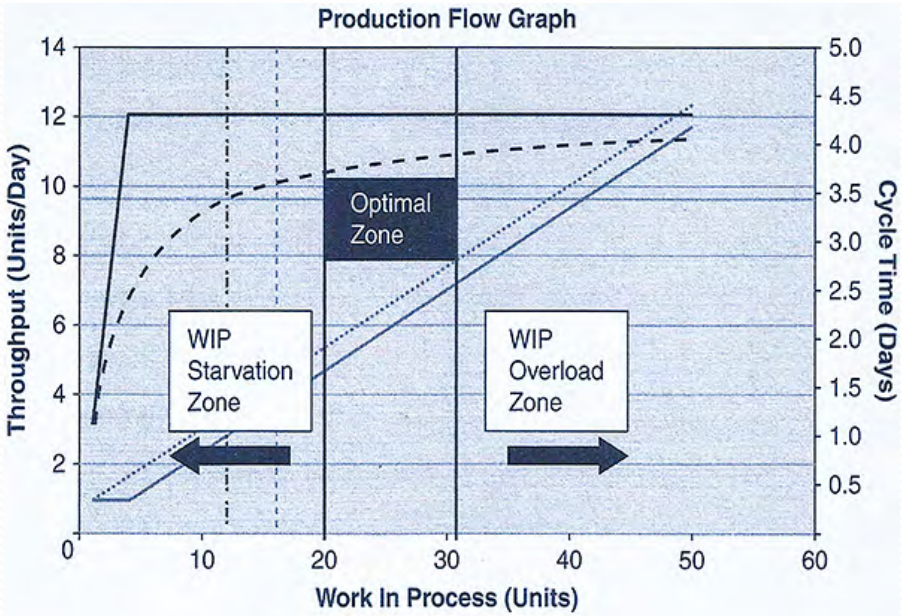
Matejcik, F. J. (2007). *Operations planning*. Rapid City, S.D.: Electronic University Consortium.

WIP vs. Throughput/Cycle Time

- Increasing WIP elongates throughput & cycle times
- Far too little or far too much WIP is also suboptimal
- Decrease WIP to decrease throughput/cycle times



A Visual Representation of Production-Flow Behavior

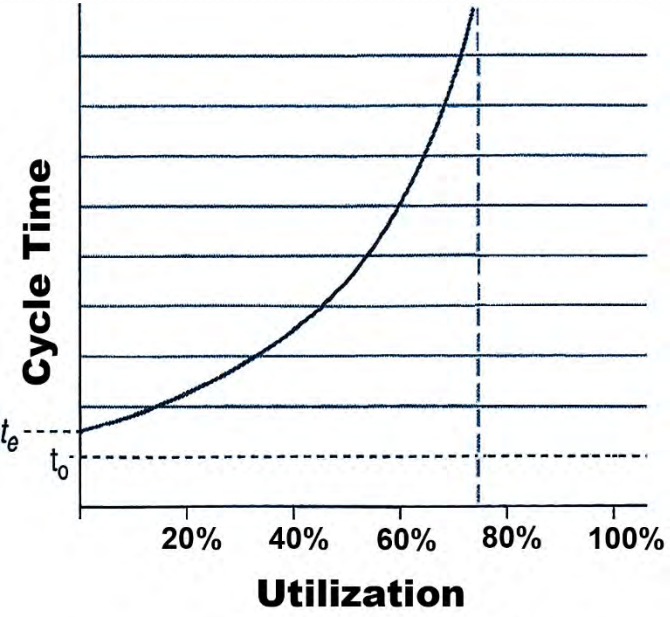


The Optimal WIP Zone

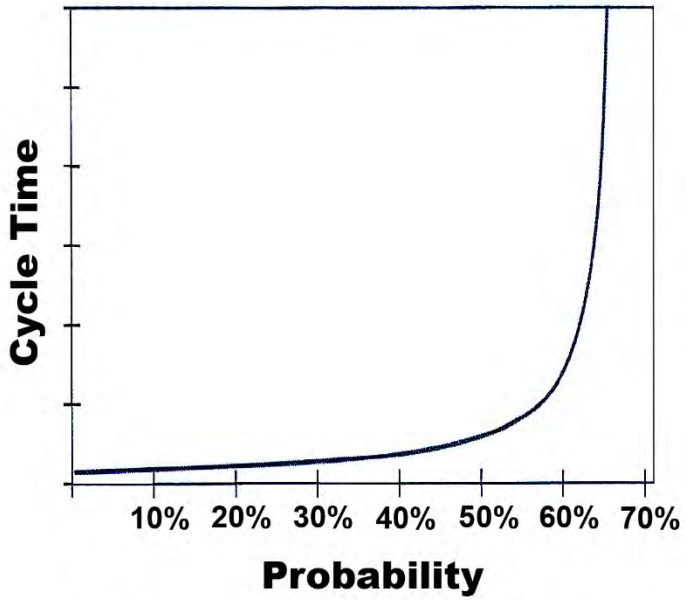
Pound, E. S., Bell, J. H., Spearman, M. L. (2014). *Factory physics*: *How leaders improve performance in a post-lean six sigma world*. New York, NY: McGraw-Hill Education.

Utilization vs. Cycle Times

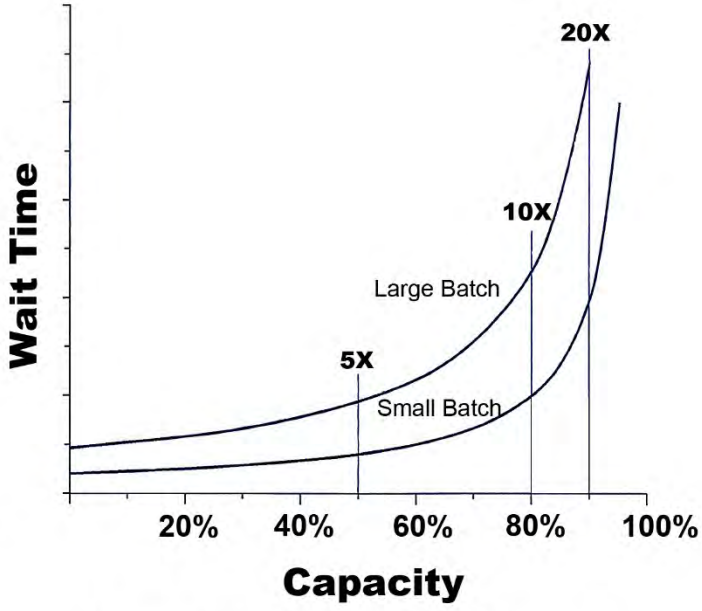
- Increased utilization elongates cycle times
- Increasing cycle times reduces system quality
- ☞ Decrease utilization to speed up & increase quality



Corrupting Influence of Variability



Cycle Time and Probability of Rework

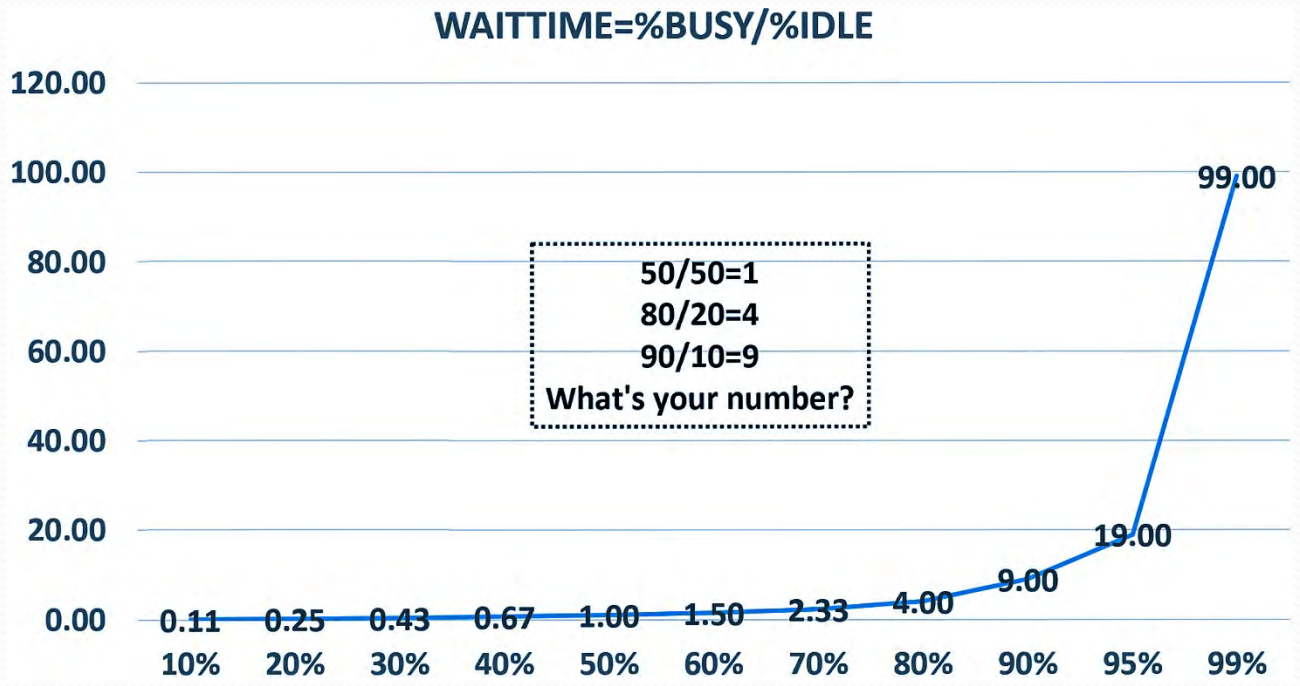


Rising Wait Times and Full Capacity

Smith, P. G. (2018). *Flexible product development*: *Agile hardware development to liberate innovation.* San Francisco, CA: Jossey-Bass.

Utilization vs. Wait Times

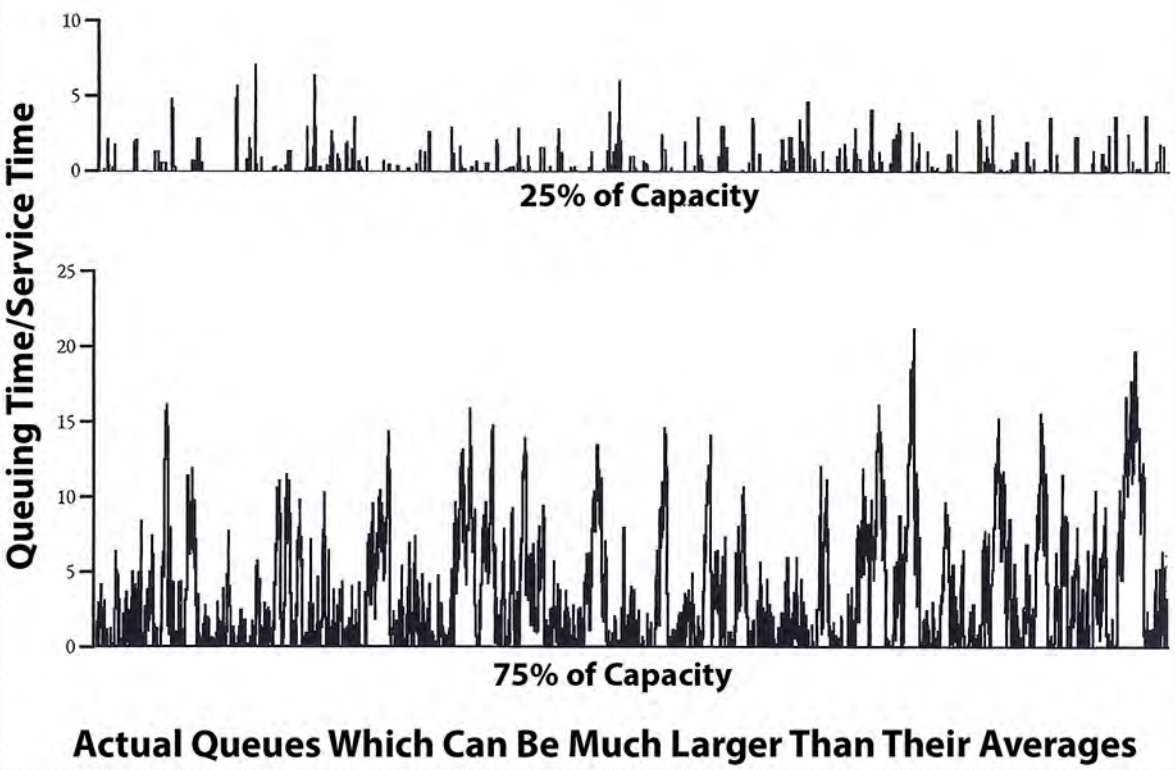
- Increased utilization elongates wait times
- Wait times quadruple at 80% utilization rates
- ☞ Decrease utilization to speed up & increase quality



Prugh, S. (2018). *SAFe, ITIL, and DevOps.* SAFe Summit, Washington, DC: USA.

Capacity vs. Queue-Service Times

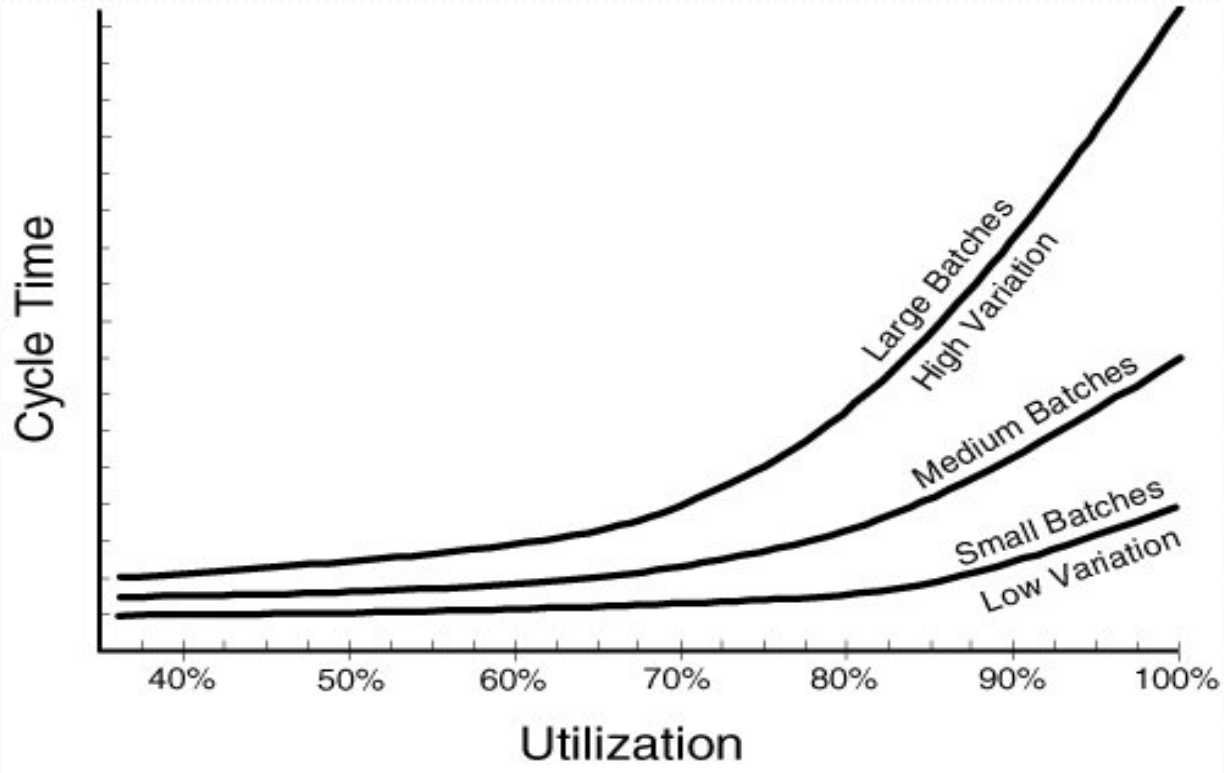
- Increased utilization elongates queue & service times
- Increased utilization does not reduce service time
- ☞ Decrease utilization to decrease waiting times



Smith, P. G. (2018). *Flexible product development*: *Agile hardware development to liberate innovation.* San Francisco, CA: Jossey-Bass.

Batchsize vs. Variation (Defects)

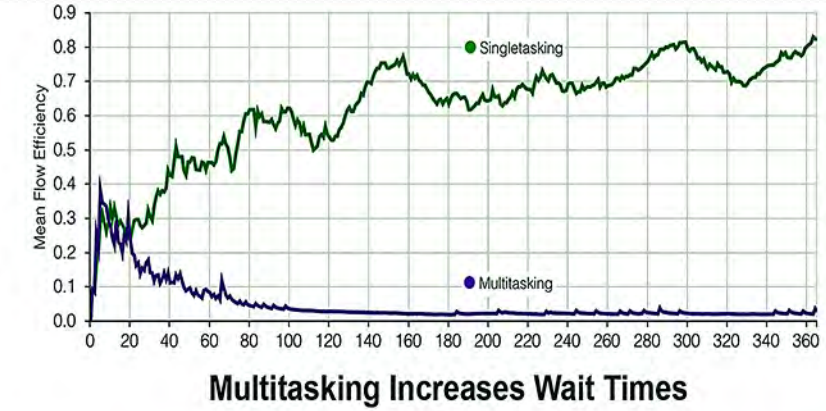
- Larger batchsizes increase variation (defects)
- Reducing batchsizes reduces variation (defects)
- ☞ Decrease batchsizes to decrease variation (defects)

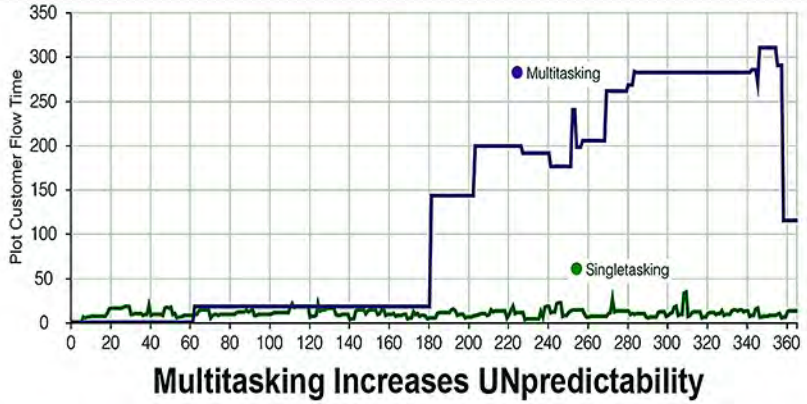


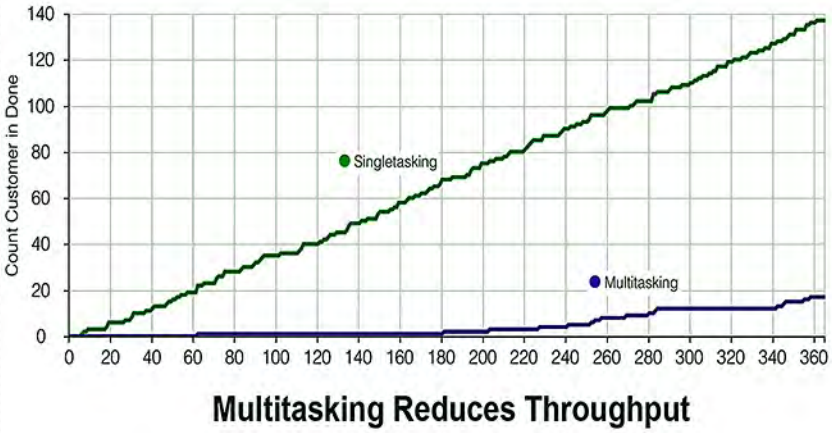
Fifoot, T. (2016). *Improving innovation through batch size optimisation*. Melbourne, Australia: Scrum Australia.

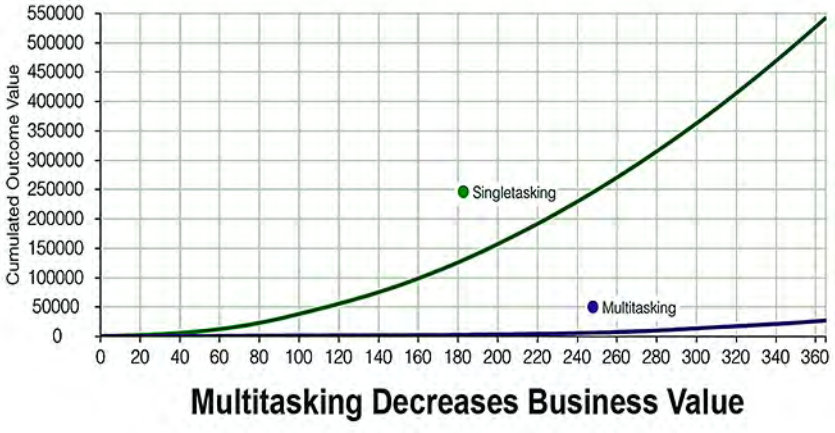
Multi-Tasking & Performance

- Increased multi-tasking decreases performance
- Multi-tasking decreases throughput & predicability
- ☞ Decrease multitasking to increase & business value









Willuda, S. (2019). Simulating the negative consequences of multitasking on flow, throughput, and value generation. San Francisco, CA: Medium.Com

Multi-Tasking & Context Switching

- Increased multi-tasking decreases project time
- Increased multi-tasking increases context switching
- ☞ Decrease multitasking to speed up & increase quality

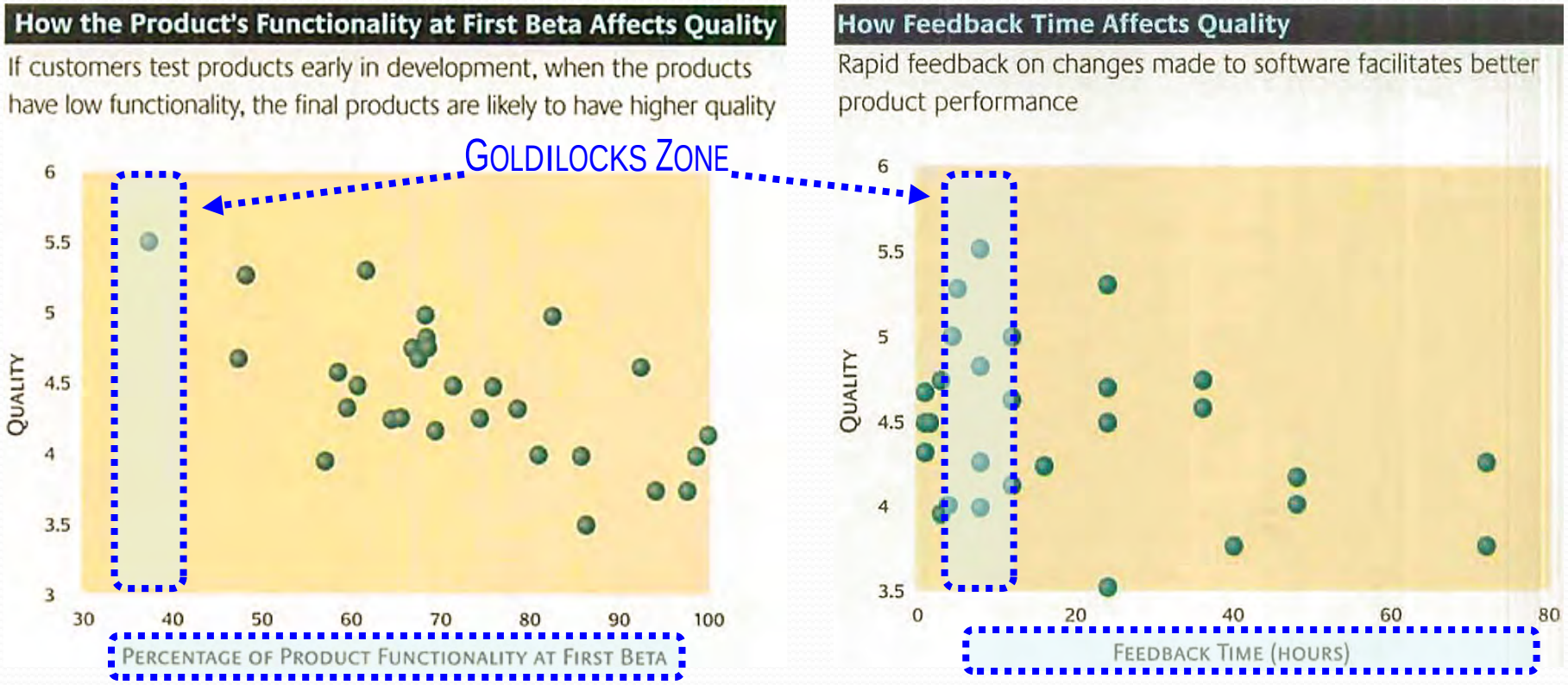
Number of Simultaneous Projects	% Time Available per Project	% Loss to Context Switching
1	100	0
2	40	20
3	20	40
4	10	60
5	5	75

The Impact of Multi-Tasking & Context Switching on Productivity

Thompson, K. W. (2019). *Solutions for agile governance in the enterprise* (*SAGE*): *Agile project, program, and portfolio for development of hardware and software products*. Vancouver, CA: Sophont Press.

Incremental Development

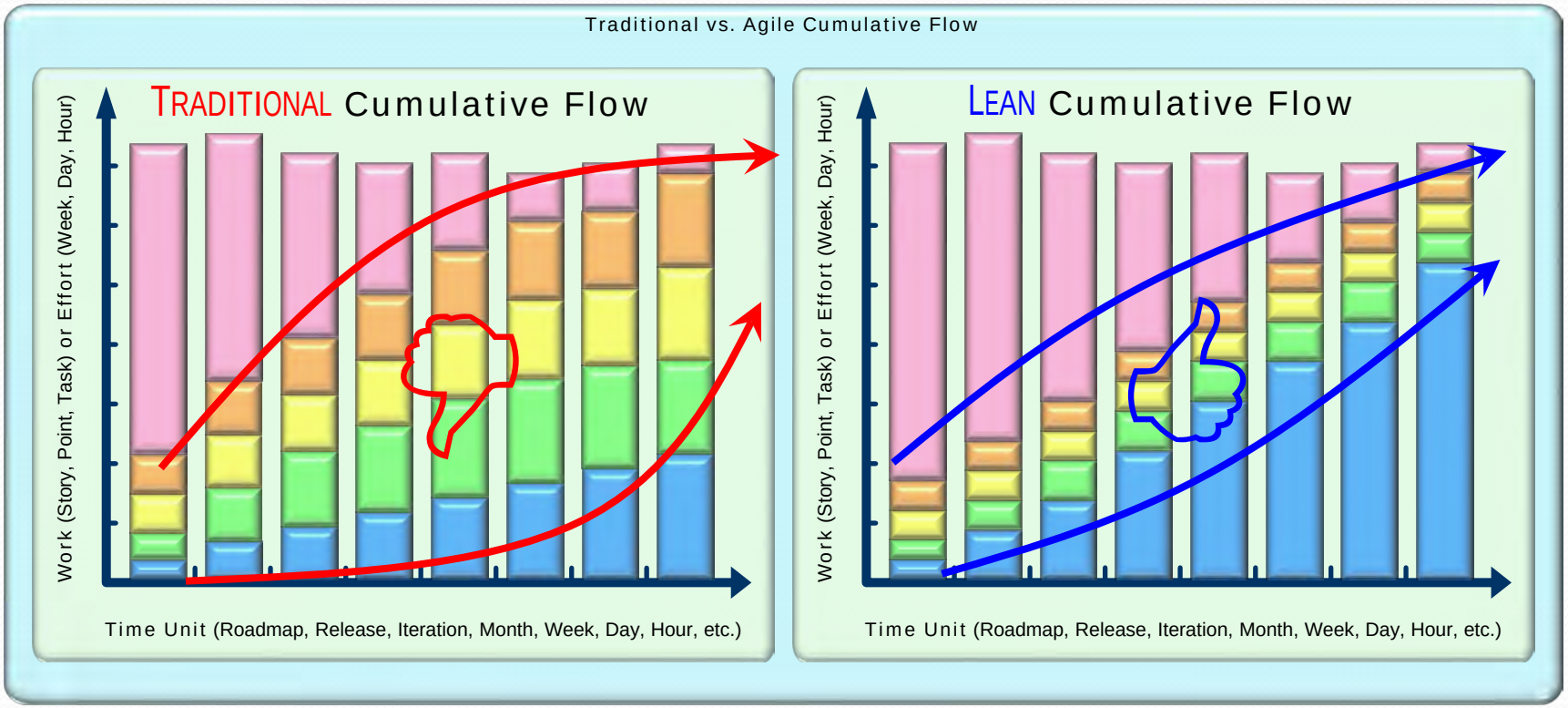
- Incremental development improves quality
- Fast feedback cycles also improves quality too
- Perform fast, incrementalism to improve quality



MacCormack, A. (2001). Product development practices that work: How Internet companies build software. *MIT Sloan Management Review*, *42*(2), 15-24.

Limiting Work in Process (WIP)

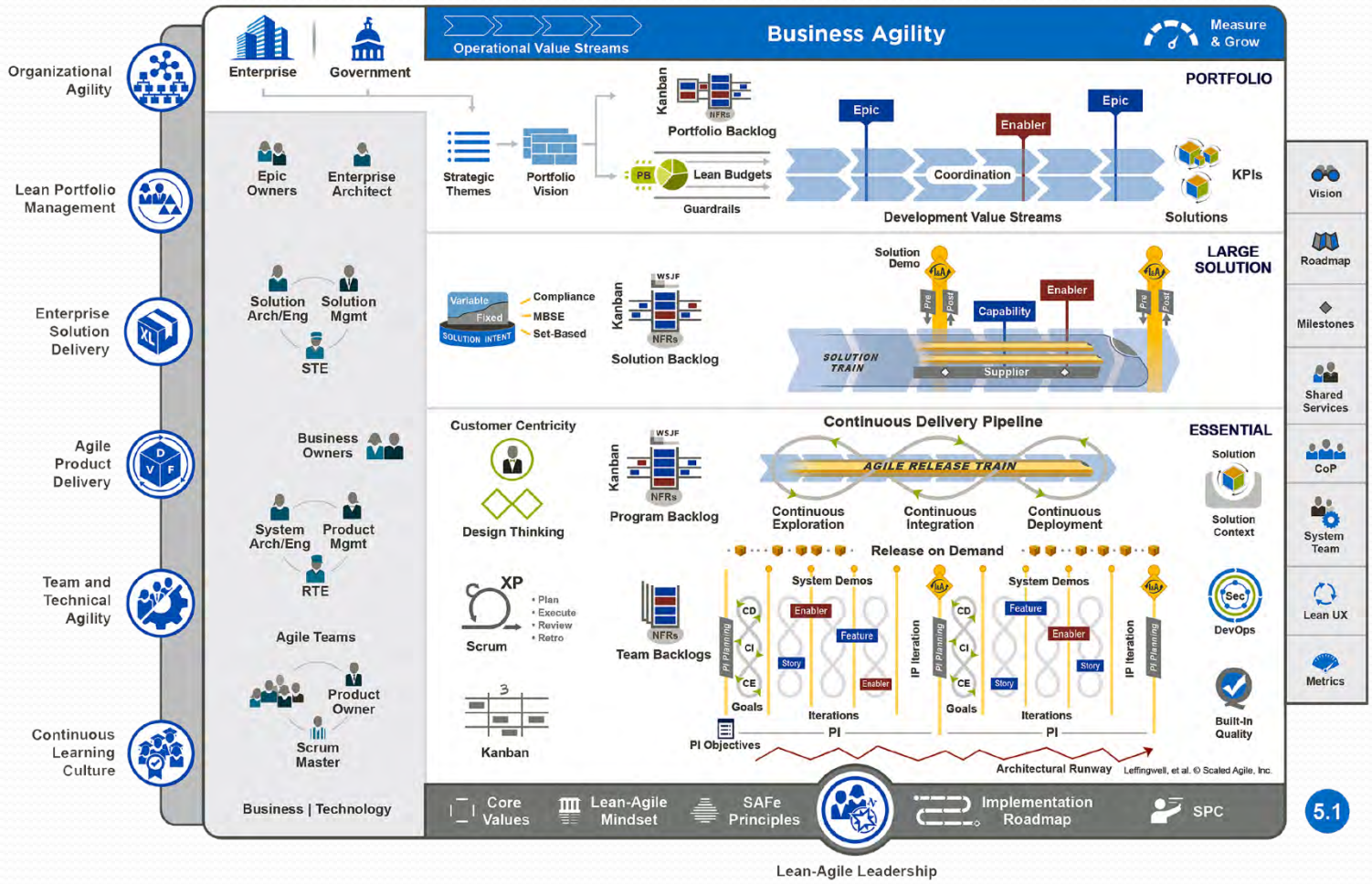
- Late big bang integration increases WIP backlog
- Agile testing early and often reduces WIP backlog
- Improves workflow and reduces WIP & lead times



Anderson, D. J. (2004). *Agile management for software engineering*. Upper Saddle River, NJ: Pearson Education.
Anderson, D. J. (2010). *Kanban*: *Successful evolutionary change for your technology business*. Sequim, WA: Blue Hole Press.

Lean Enterprise Model—SAFe 5.1

- Framework by Dean Leffingwell of Rally in 2007
- Newest version leaner, meaner, lighter, and simpler
- Lightweight framework for enterprise wide lean thinking

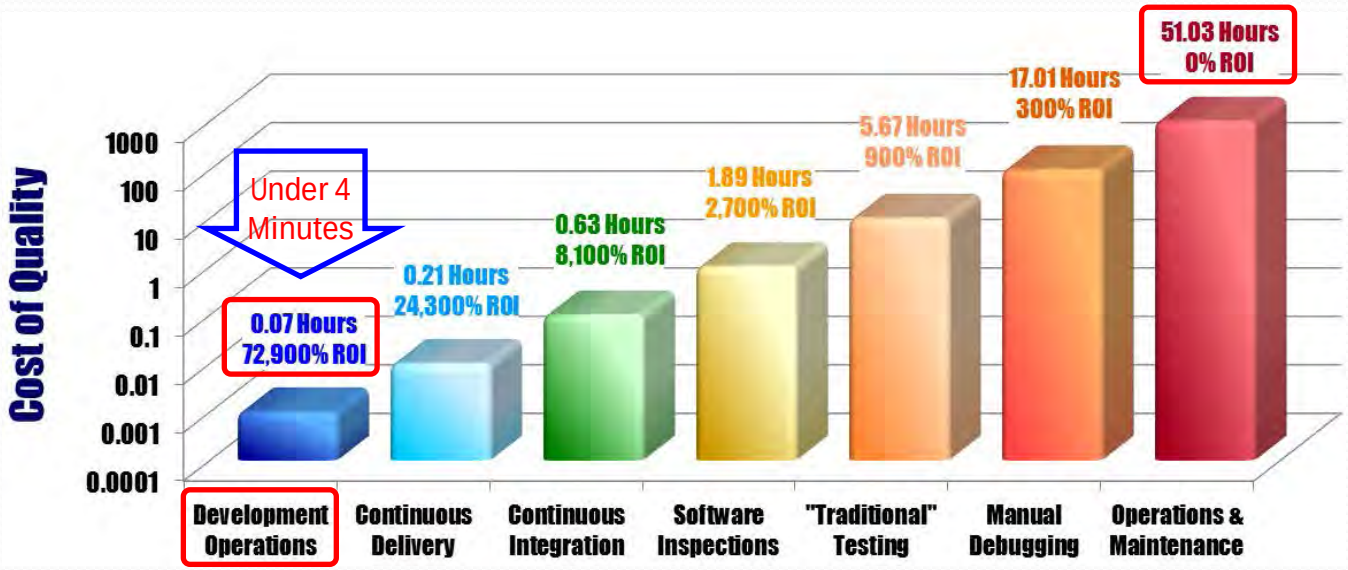


Leffingwell, D. (2007). *Scaling software agility*: *Best practices for large enterprises*. Boston, MA: Pearson Education.

Lean Thinking, DevOps, & Testing

- Fast testing is orders-of-magnitude more efficient
- Based on millions of automated tests run in seconds
- One-touch auto-delivery to billions of global end-users

Activity	Def	CoQ	DevOps Economics	Hours	ROI
Development Operations	100	0.001	100 Defects x 70% Efficiency x 0.001 Hours	0.070	72,900%
Continuous Delivery	30	0.01	30 Defects x 70% Efficiency x 0.01 Hours	0.210	24,300%
Continuous Integration	9	0.1	9 Defects x 70% Efficiency x 0.1 Hours	0.630	8,100%
Software Inspections	3	1	2.7 Defects x 70% Efficiency x 1 Hours	1.890	2,700%
"Traditional" Testing	0.81	10	0.81 Defects x 70% Efficiency x 10 Hours	5.670	900%
Manual Debugging	0.243	100	0.243 Defects x 70% Efficiency x 100 Hours	17.010	300%
Operations & Maintenance	0.073	1,000	0.0729 Defects x 70% Efficiency x 1,000 Hours	51.030	n/a

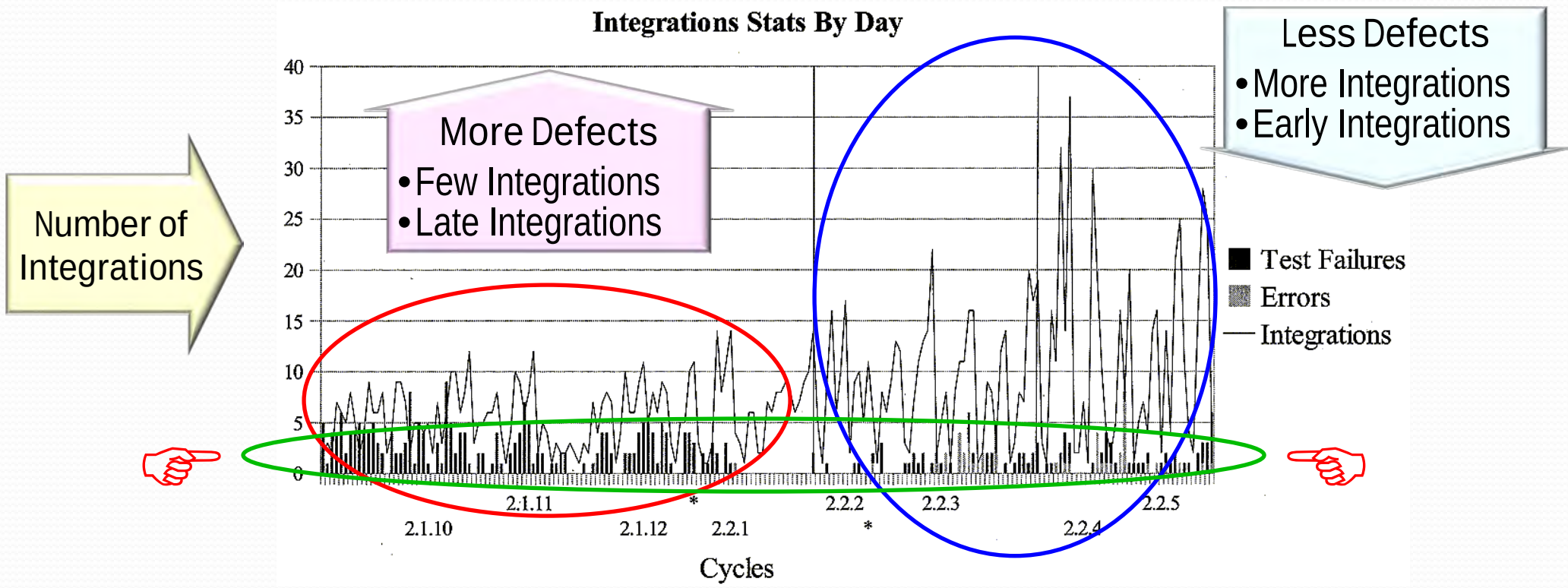


4,500 x Faster than Code Inspections

Rico, D. F. (2016). *Devops cost of quality (CoQ): Phase-based defect removal model*. Retrieved May 10, 2016, from http://davidfrico.com

Lean Thinking & Testing Speed

- Fewer integrations leave in higher bug counts
- Frequent, early integrations eliminate most defects
- Goal is to have as many early integrations as possible



Lacoste, F. J. (2009). Killing the gatekeeper: Introducing a continuous integration system. *Proceedings of the Agile 2009 Conference, Chicago, Illinois, USA*, 387-392.

Lean Thinking at HP

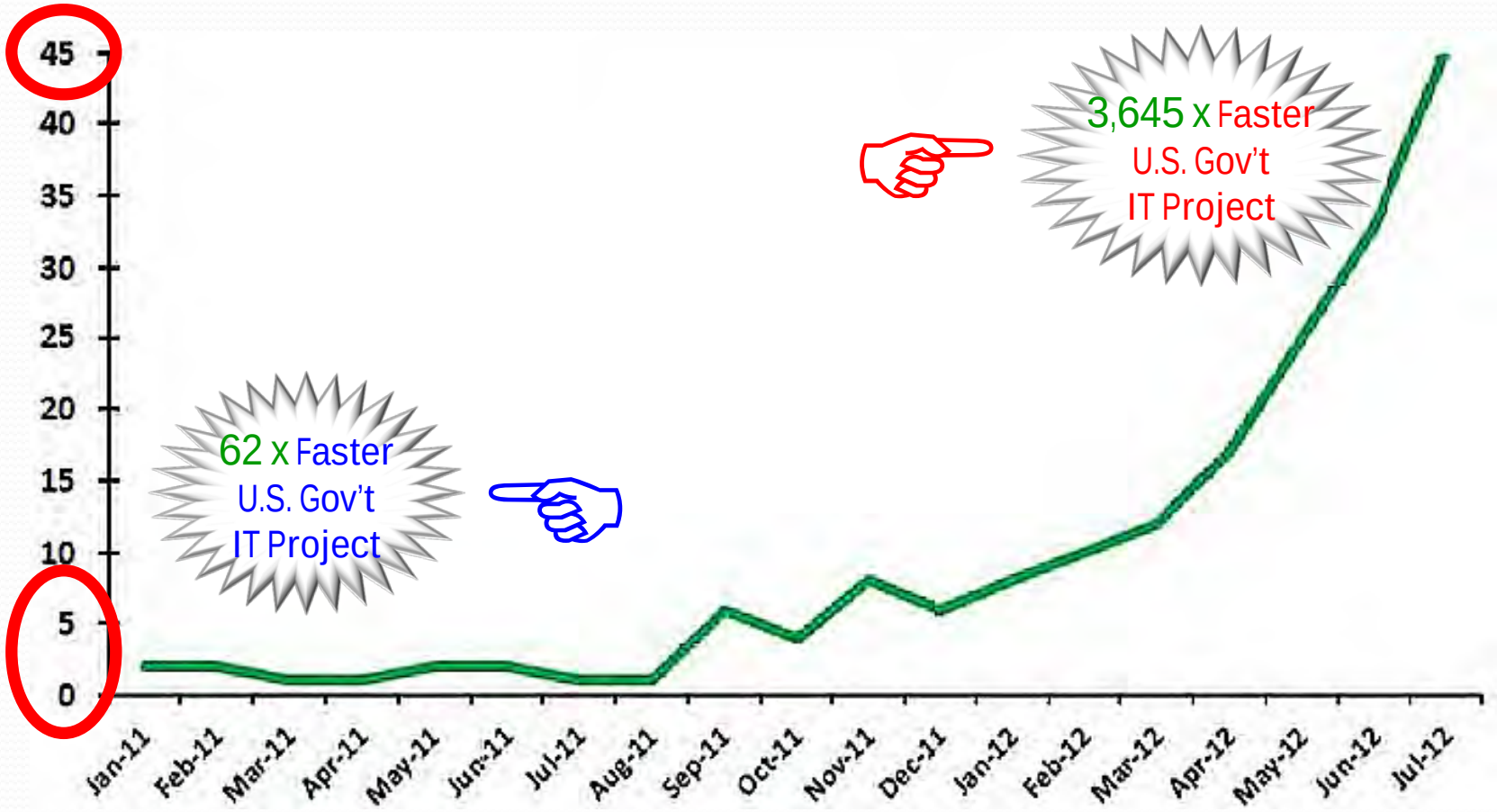
- Hewlett-Packard (HP) is major user of lean principles
- 400 engineers developed 10 million LOC in 4 years
- Major gains in testing, deployment, & innovation

Type	Metric	Manual	DevOps	Major Gains
Cycle Time Improvements	Build Time	40 Hours	3 Hours	13 x
	No. Builds	1-2 per Day	10-15 per Day	8 x
	Feedback	1 per Day	100 per Day	100 x
	Regression Testing	240 Hours	24 Hours	10 x
Development Cost Effort Distribution	Integration	10%	2%	5 x
	Planning	20%	5%	4 x
	Porting	25%	15%	2 x
	Support	25%	5%	5 x
	Testing	15%	5%	3 x
	Innovation	5%	40%	8 x

Gruver, G., Young, M. & Fulghum, P. (2013). *A practical approach to large-scale agile development*. Upper Saddle River, NJ: Pearson Education.

Lean Thinking at Dot Coms

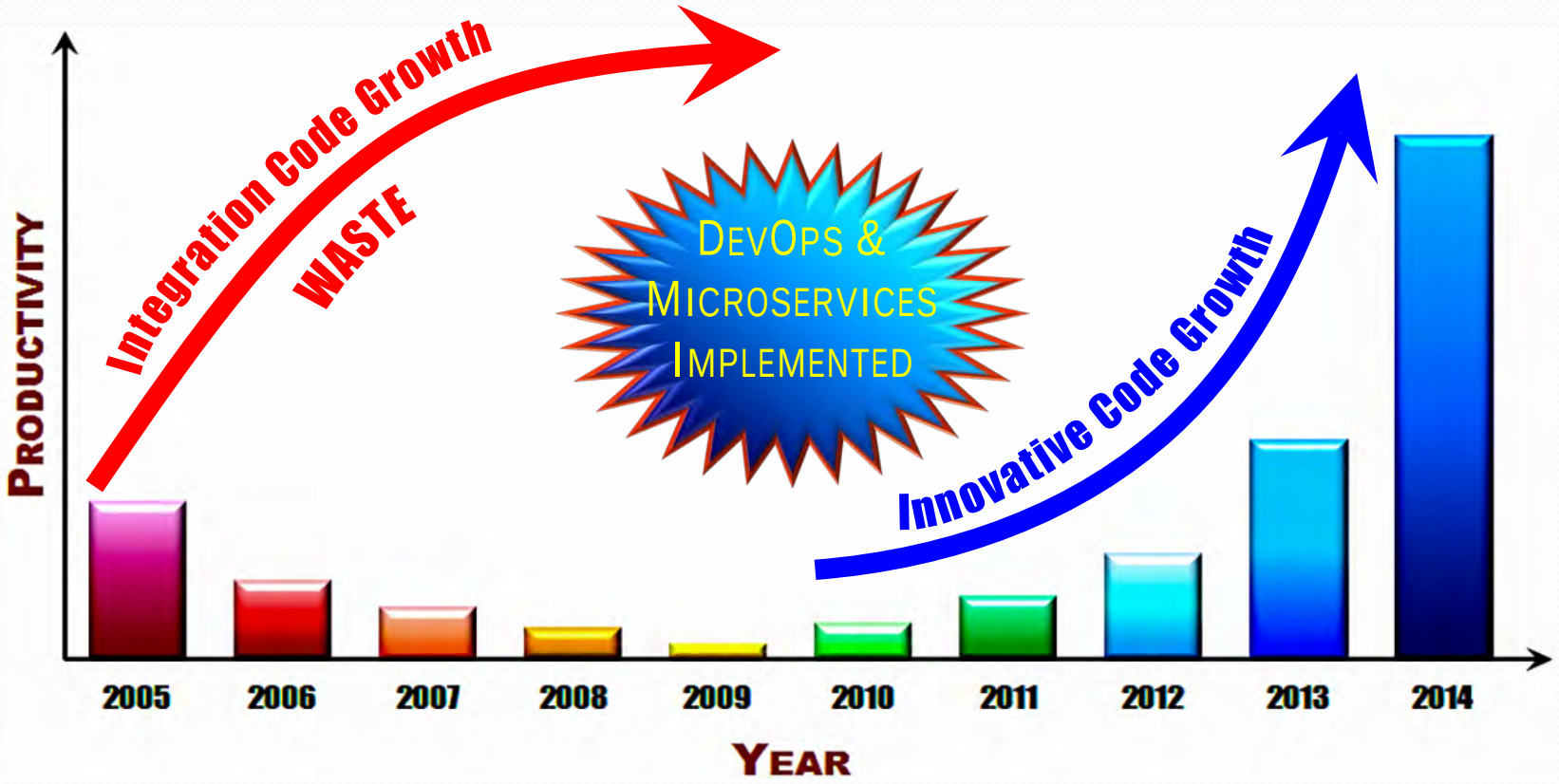
- Assembla went from 2 to 45 releases every month
- 15K Google developers run 150 million tests per day
- 30K+ Amazon developers deliver 136K releases a day



Singleton, A. (2014). *Unblock*: *A guide to the new continuous agile*. Needham, MA: Assembla, Inc.

Lean Thinking at Blackboard

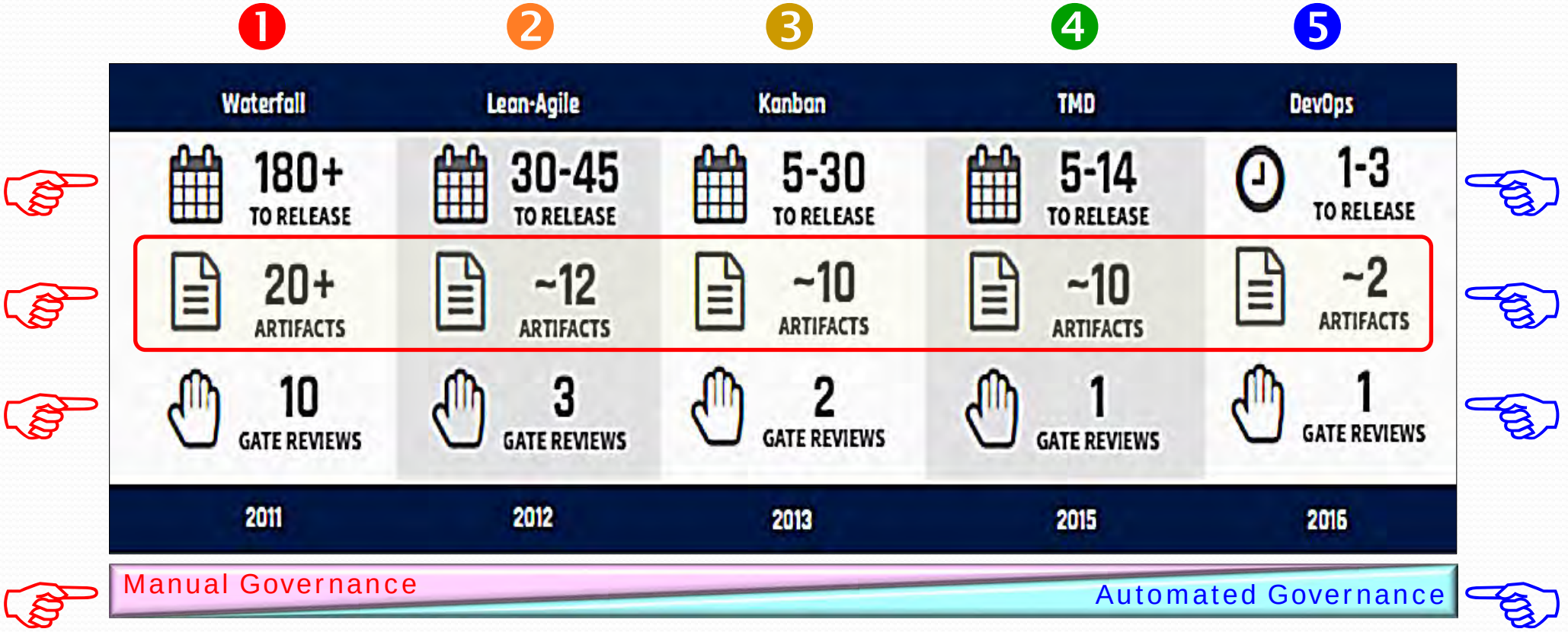
- Productivity STOPS due to excessive integration
- Implemented lean thinking principles around 2010
- Waste elimination, productivity & innovation skyrocket



Ashman, D. (2014). Blackboard: Keep your head in the clouds. *Proceedings of the 2014 Enterprise DevOps Summit, San Francisco, California, USA.*

Lean Thinking at U.S. DHS

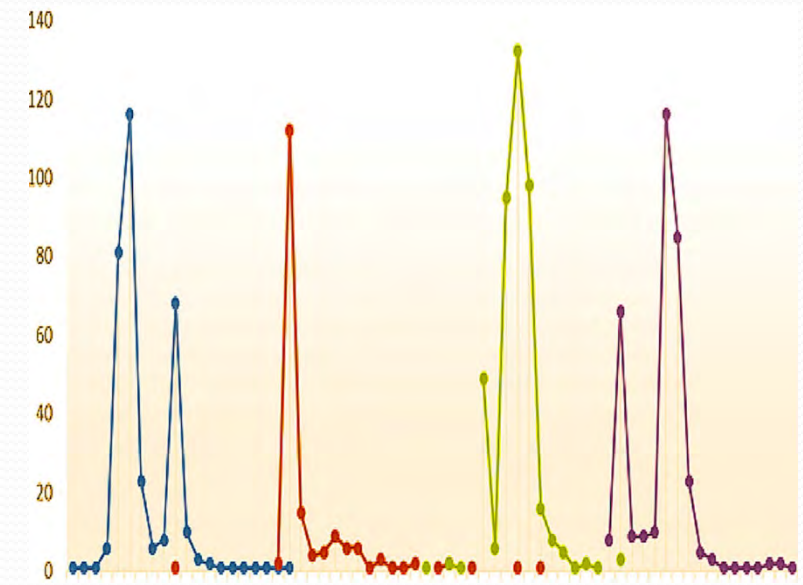
- 1st gen replete with large portfolios & governance
- 2nd-3rd gen yield minor incremental improvements
- 4th-5th gen enables big order-of-magnitude impacts

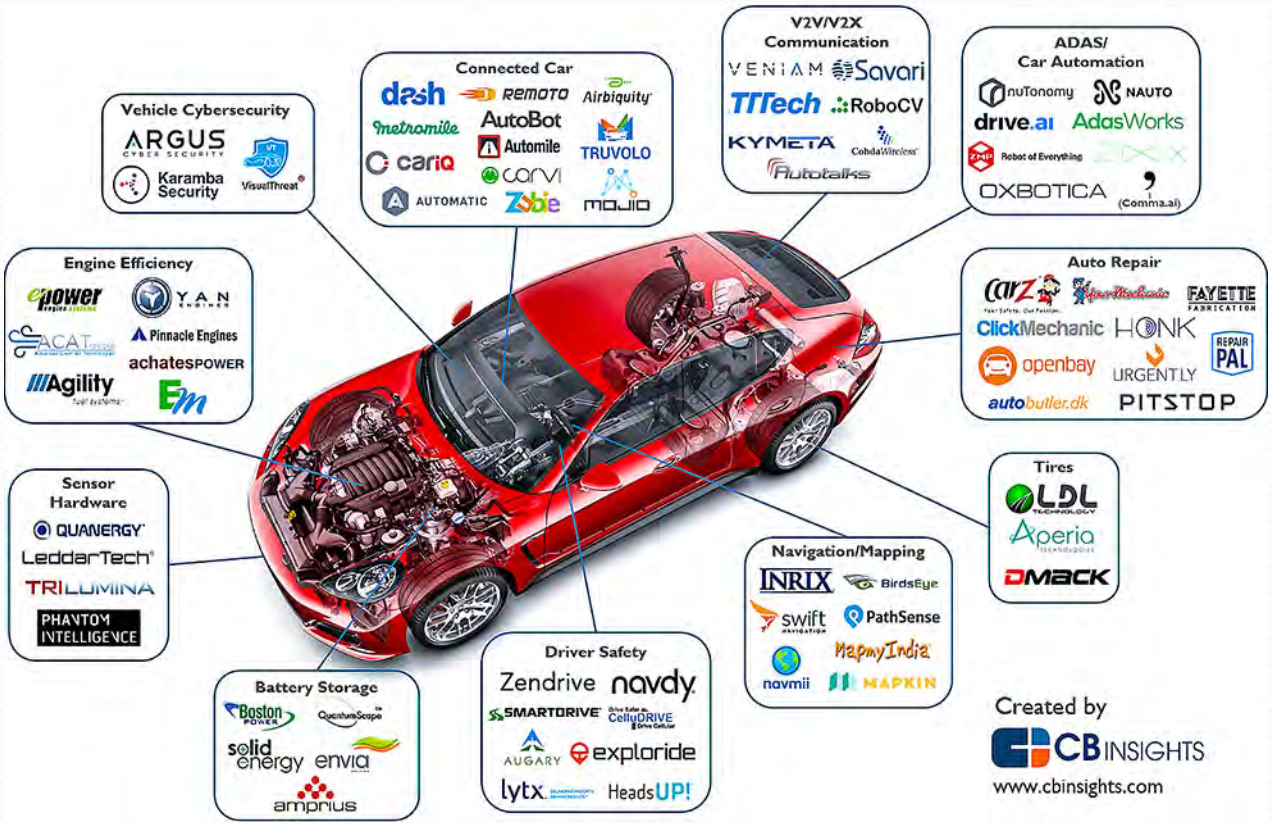


Denayer, L. (2017). *U.S. DHS citizenship and immigration services*: *USCIS agile development*. Washington, DC. iSDLC Seminar.

Lean Thinking at Tesla

- Tesla vehicle models are all electric automobiles
- Tesla autos have 100-200 million lines of code
- ☞ Tesla performs up to 130 deployments per day





Choksi, N. (2016). *How software lifecycle integration and devops are transforming car development. Goto Conference, Copenhagen, Denmark.*
Vost, S., & Wagner, S. (2016). *Towards continuous integration and continuous delivery in the automotive industry*. Ithaca, NY: Cornell University.

Lean Thinking on Big IT Portfolios

Who	Results
Google	1 code repository 40,000 commits per day 50,000 builds per day **150 million tests per day**
NETFLIX	24-day average server age 1 billion metrics per day Self-service deploys **Zero downtime**
amazon	Everything is monitored Code APIs for everything **136,000 deploys per day** Very tiny two-pizza teams
TARGET	$1 billion annual IT budget **80 deployments per week** 17 billion API calls per month Self-service DevOps Dojo training
Bing	600 developers One code branch **20,000 tests per commit** Every clean build deployed

Rix, M. (2019). Conquering the monolith: Architecting for DevOps and release on demand. *SAFe Summit Europe, Hague, Netherlands*.

ROI of Lean Thinking

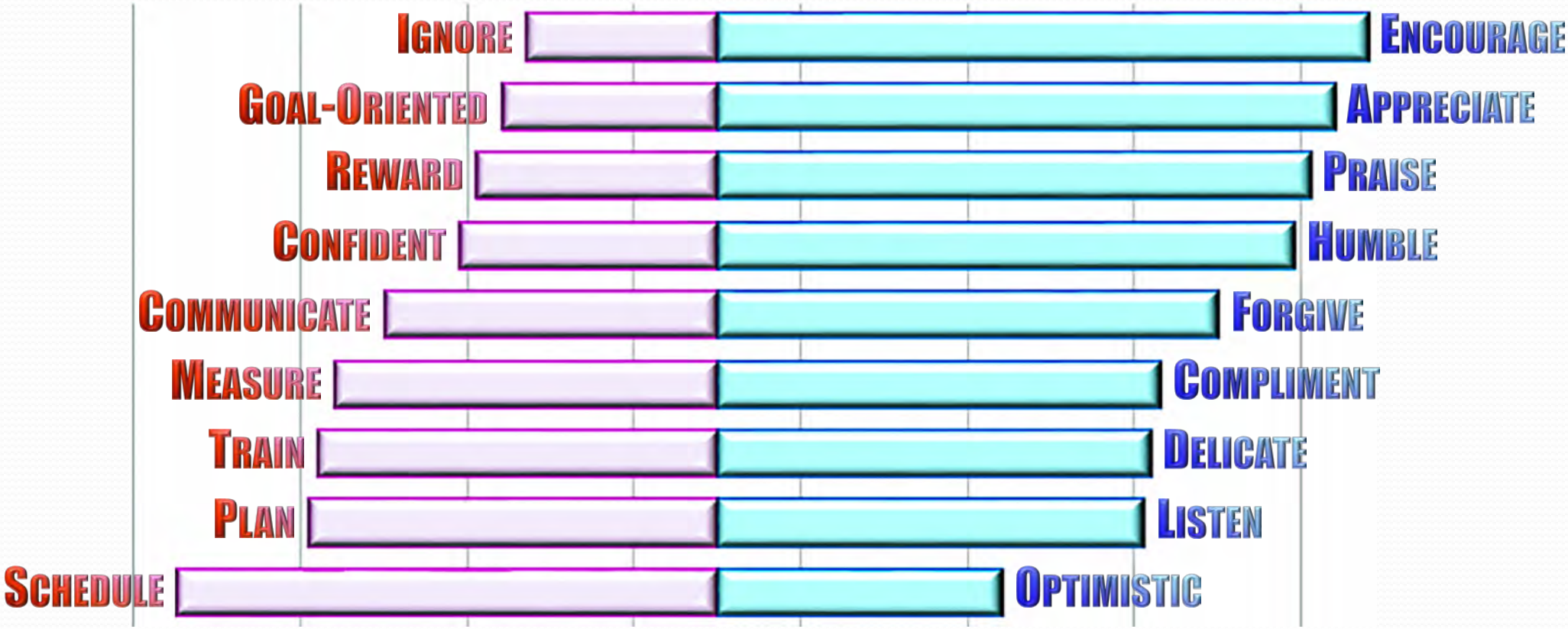
- IT lean thinking economics starting to emerge
- ROI ranges from $17M to $195M *with minor costs*
- Benefits from cost savings, revenue, and availability

Org	Low Perf	Med Perf	High Perf
Small - 250 -	$23M Benefits	$29M Benefits	$17M Benefits
	$0.2M Costs	$0.2M Costs	$0.2M Costs
	13,589% ROI	**17,799% ROI**	**9,932% ROI**
	3 Day Payback	*2 Day Payback*	*4 Day Payback*
Medium - 2,000 -	$42M Benefits	$66M Benefits	$36M Benefits
	$1.3M Costs	$1.3M Costs	$1.3M Costs
	3,101% ROI	**4,901% ROI**	**2,663% ROI**
	11 Day Payback	*7 Day Payback*	*13 Day Payback*
Large - 8,500 -	$114M Benefits	$195M Benefits	$76M Benefits
	$5.6M Costs	$5.6M Costs	$5.6M Costs
	1,942% ROI	**3,375% ROI**	**1,254% ROI**
	18 Day Payback	*11 Day Payback*	*27 Day Payback*

Forsgren, N., Humble, J., & Kim, G. (2017). *Forecasting the value of devops transformations*: *Measuring roi of devops*. Portland, OR: DevOps Research.
Rico, D. F. (2017). *Devops return on investment (ROI) calculator*. Retrieved August 29, 2017, from http://davidfrico.com/devops-roi.xls

Traditional vs. Lean Leaders

- Also compared traditional vs. contemporary attributes
- Older ones based on traditional project management
- Today's leaders encourage, appreciate, and praise



Henson, V. (2016). *Dale carnegie training*: *Global leadership study*. New York, NY: Dale Carnegie Training.

Lean Business Performance

76 Percent *vs* **38** Percent

High performers have twice as many successful strategic initiatives vs. low performers

31 Percent *vs* **9** Percent

High performers are three times as likely to have high organizational agility

57 Percent *vs* **28** Percent

High performers are twice as likely to have high alignment of projects to organizational strategy

Three Strategic Focuses That Drive Organizational Success

PEOPLE

Managing Talent and Change

High performers achieve significantly more successful strategic initiatives by effectively managing their project management talent. Furthermore, they demonstrate successful organizational change management, which is more effective with actively engaged sponsors.

2 times

Twice as many high performers have actively engaged sponsors compared to low performers.

PROCESSES

Project, Program and Portfolio Management Maturity

High-performing organizations fully understand the value of project management, have a project management office (PMO) and use standardized project management practices throughout the organization.

4 times

Four times more high performers have project, program and portfolio management that is highly mature compared to low performers.

OUTCOMES

Benefits Realization

Organizations need to focus on effectively establishing, measuring and communicating the intended benefits of projects and programs.

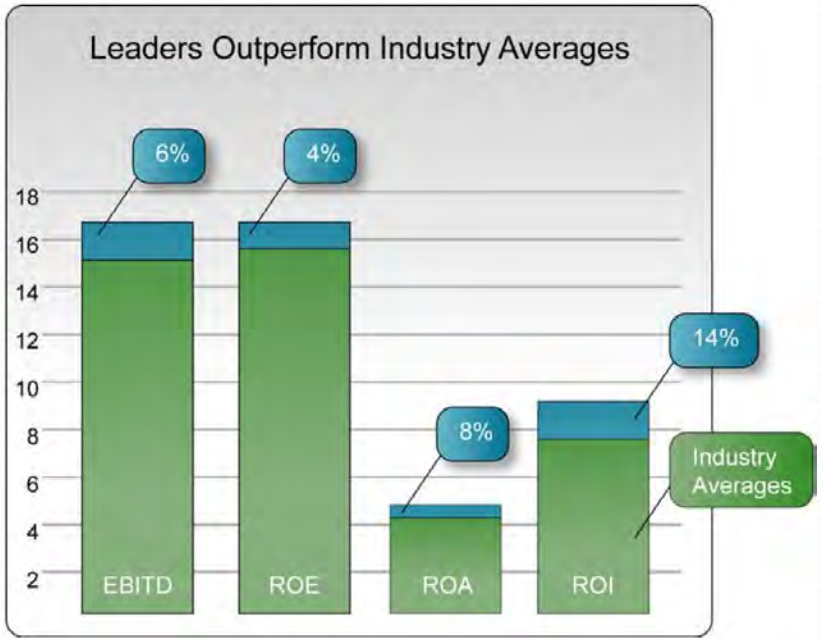
5 times

Five times more high-performing organizations have highly mature benefits realization compared to low-performing organizations.

Langley, M. A. (2014). *High-cost of low performance*: *Pulse of the profession* (*a snapshot*). Newtown Square, PA: Project Management Institute.

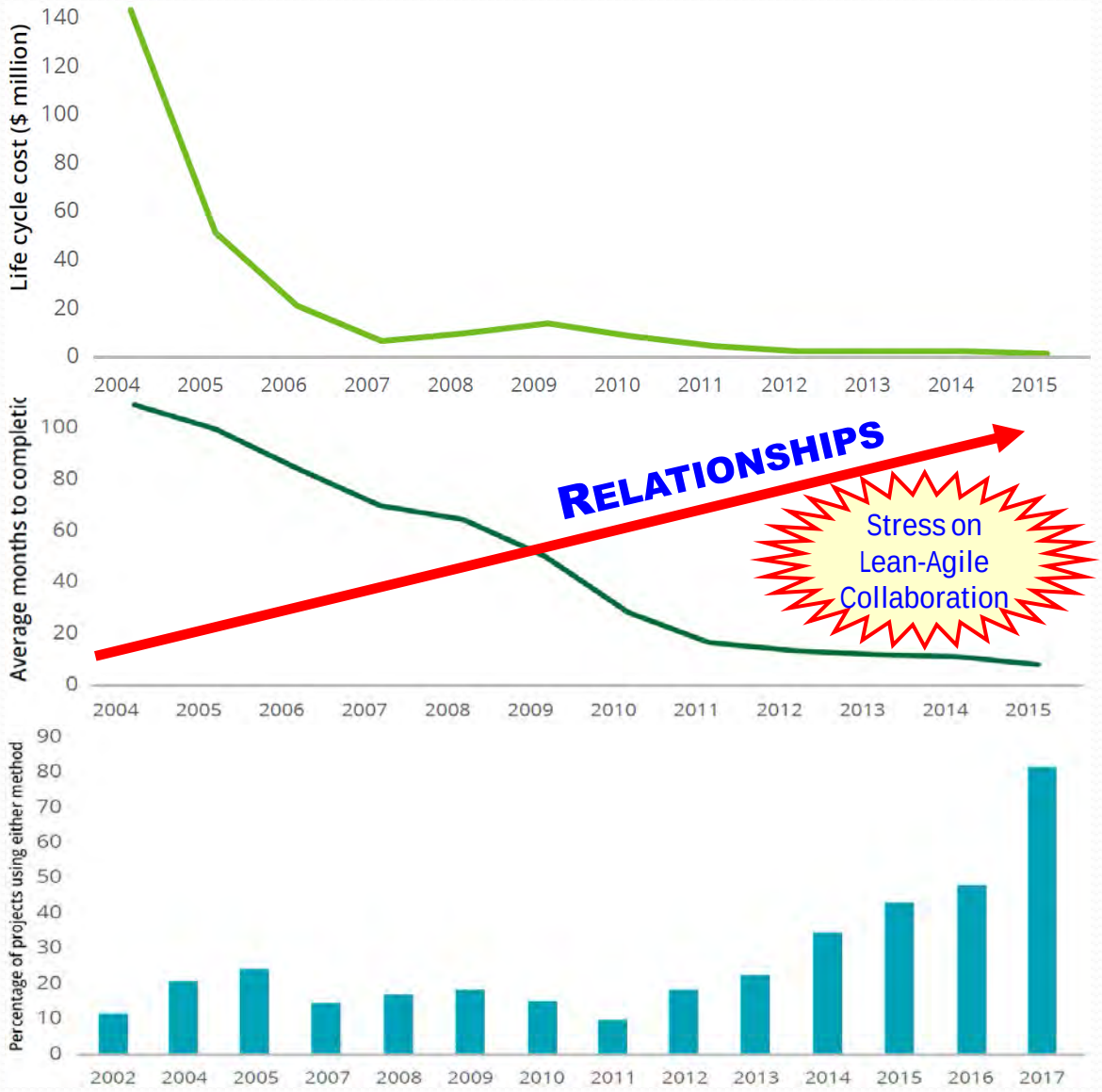
Lean Business Benefits

- Study of 15 agile vs. non-agile Fortune 500 firms
- Based on models to measure organizational agility
- Agile firms out perform non agile firms by up to 36%



Hoque, F., et al. (2007). *Business technology convergence*. The role of business technology convergence in innovation and adaptability and its effect on financial performance. Stamford, CT: BTM Institute.

Lean Government Benefits



Cost

Length

Agility

Viechnicki, P., & Kelkar, M. (2017). *Agile by the numbers*: *A data analysis of agile development in the US federal government*. Washington, DC: Deloitte, LLC.

Lean Business Performance

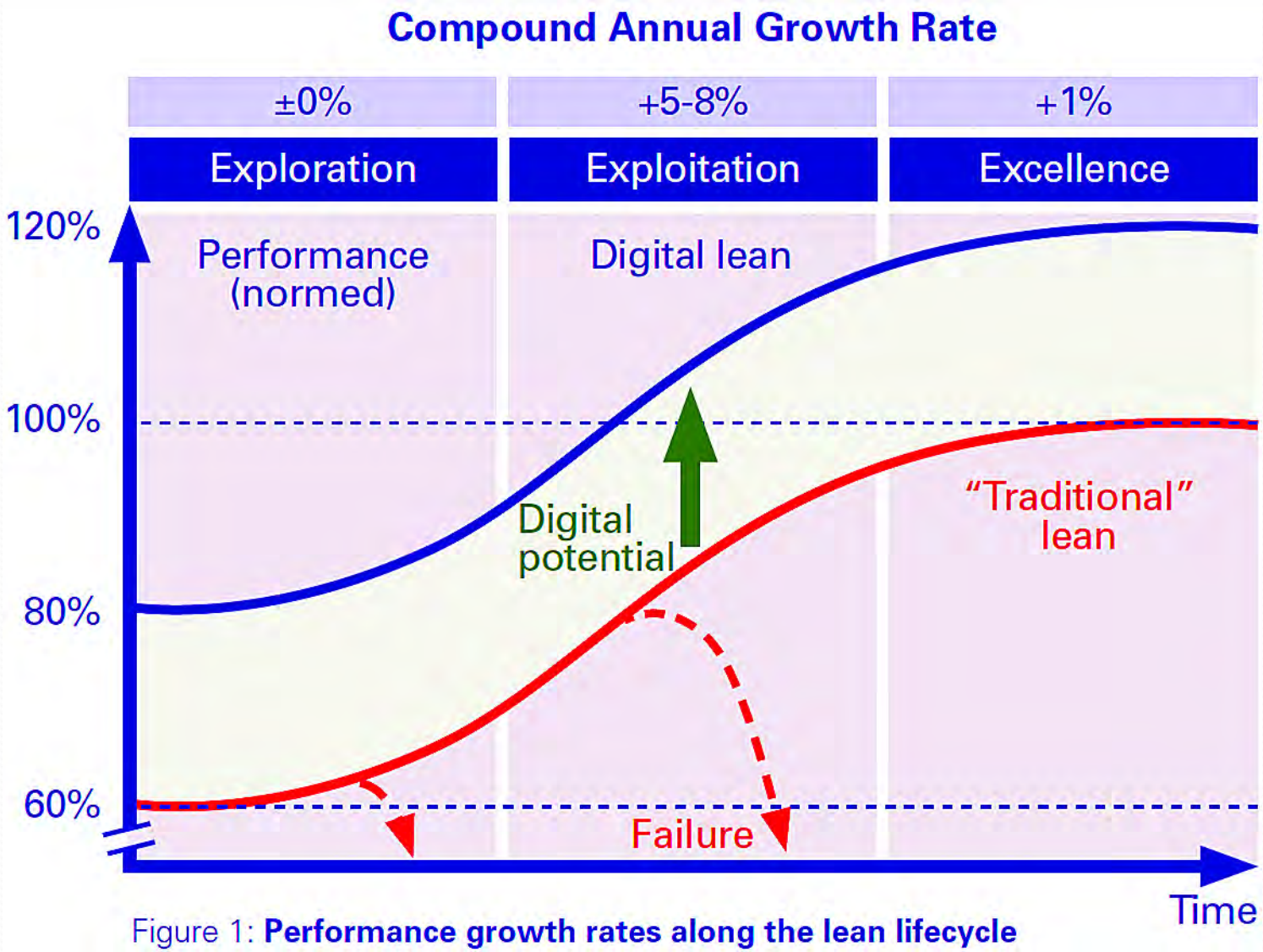
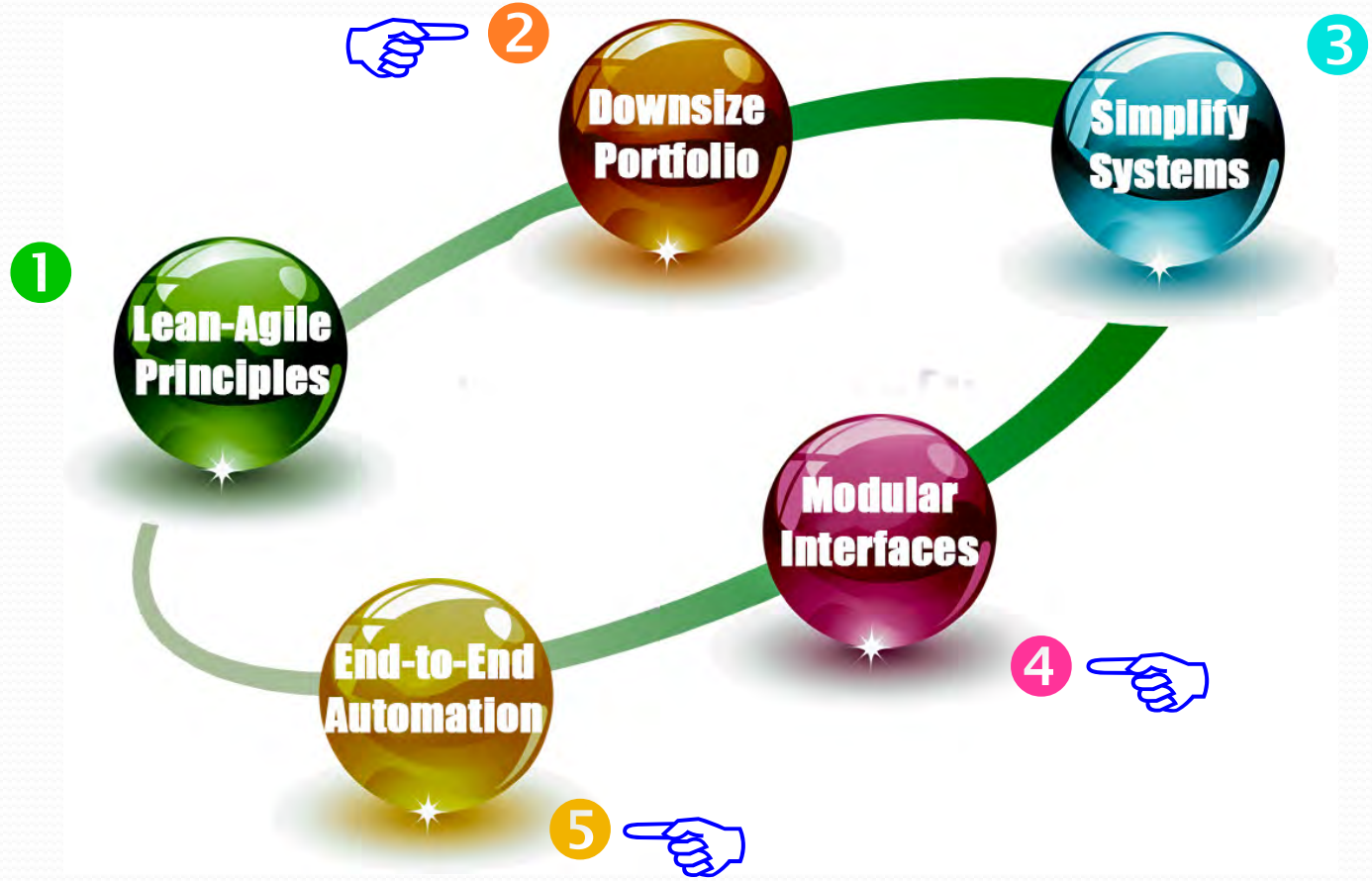


Figure 1: **Performance growth rates along the lean lifecycle**

Schreiber, B., Romanus, W., & Lee, Y. (2017). *Integrating lean principles into digital transformation.* Frankfurt, Germany: Arthur D. Little.

Five Keys to Lean Success

- Everything begins with lean thinking principles
- Next step is smaller portfolios & simpler designs
- Final step is modular interfaces & E2E automation



Kim, G., Debois, P., Willis, J., & Humble, J. *The devops handbook*: *How to create world-class agility, reliability, and security in technology organizations*. Portland, OR: IT Revolution Press.

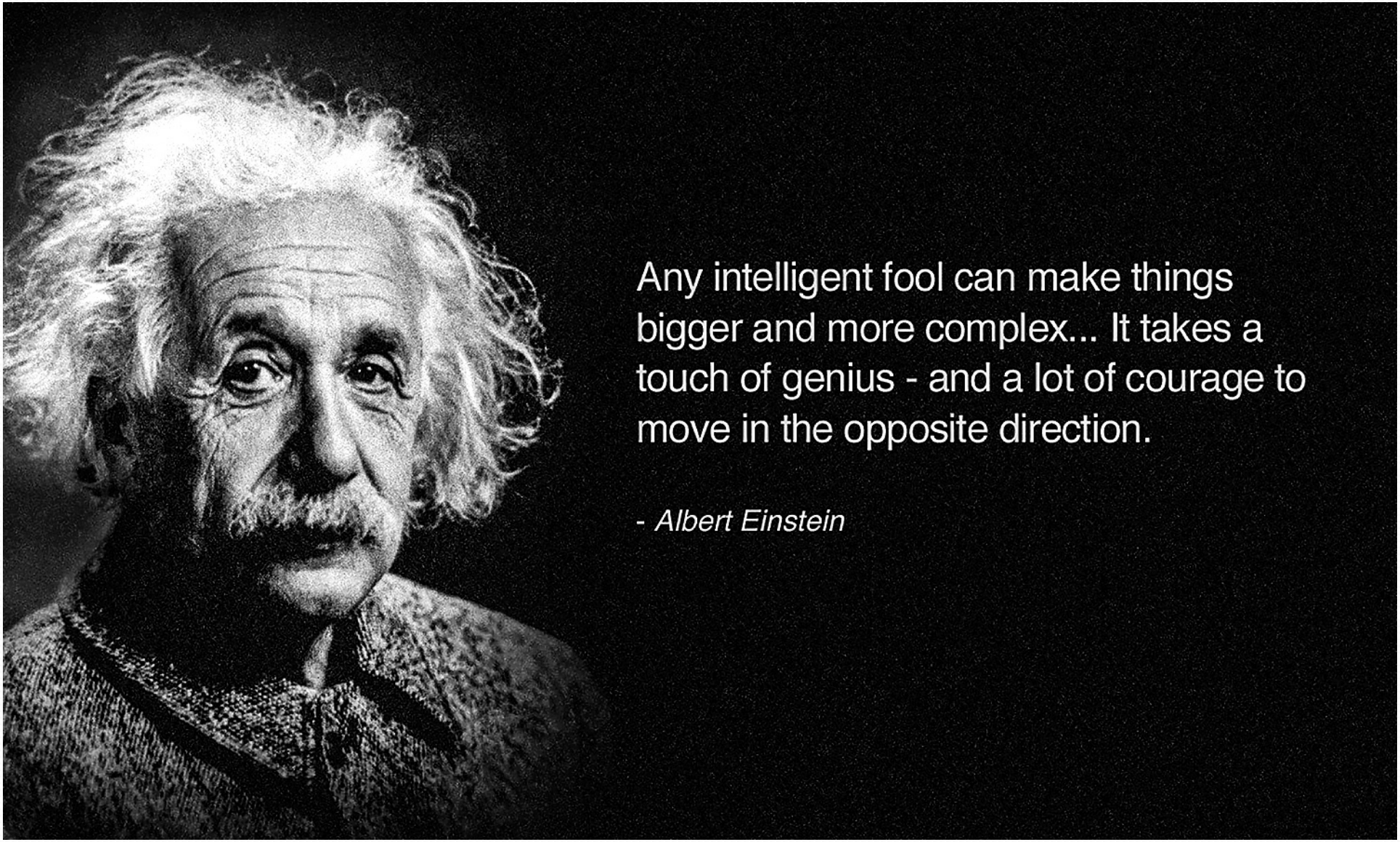
Lean Thinking Summary

- Lean DOES NOT mean deliver it now and fixing it later
- Lightweight, yet disciplined approach to development
- Reduced cost, risk, & waste while improving quality

What	How	Result
Flexibility	Use lightweight, yet disciplined processes and artifacts	Low work-in-process
Customer	**Involve customers early and often throughout development**	Early feedback
Prioritize	**Identify highest-priority, value-adding business needs**	Focus resources
Descope	**Descope complex programs by an order of magnitude**	Simplify problem
Decompose	**Divide the remaining scope into smaller batches**	Manageable pieces
Iterate	Implement pieces one at a time over long periods of time	Diffuse risk
Leanness	Architect and design the system one iteration at a time	JIT waste-free design
Swarm	**Implement each component in small cross-functional teams**	Knowledge transfer
Collaborate	**Use frequent informal communications as often as possible**	Efficient data transfer
Test Early	**Incrementally test each component as it is developed**	Early verification
Test Often	**Perform system-level regression testing every few minutes**	Early validation
Adapt	Frequently identify optimal process and product solutions	Improve performance

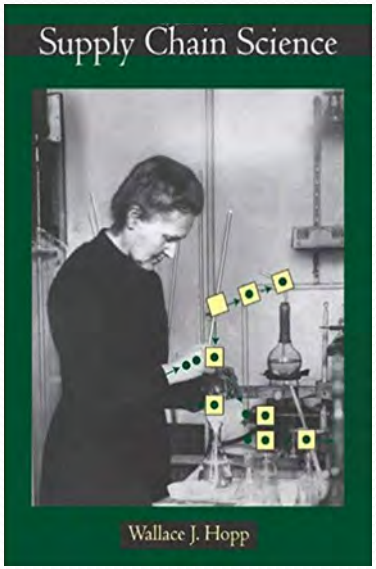
Rico, D. F. (2012). *What's really happening in agile methods: Its principles revisited*? Retrieved June 6, 2012, from http://davidfrico.com/agile-principles.pdf
Rico, D. F. (2012). *The promises and pitfalls of agile methods*. Retrieved February 6, 2013 from, http://davidfrico.com/agile-pros-cons.pdf
Rico, D. F. (2012). *How do lean & agile intersect*? Retrieved February 6, 2013, from http://davidfrico.com/agile-concept-model-3.pdf

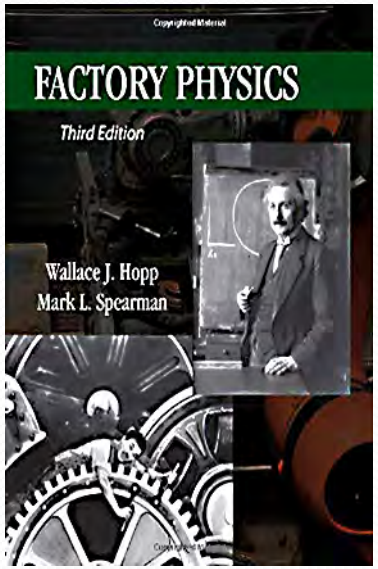
Albert Einstein on Lean Thinking

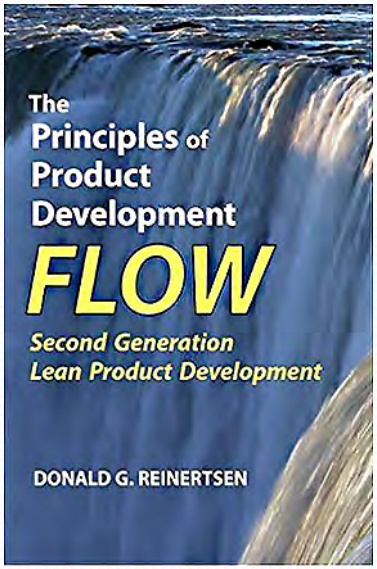


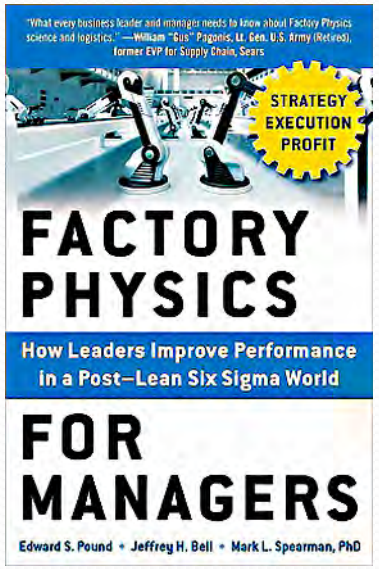
Lean Thinking Resources

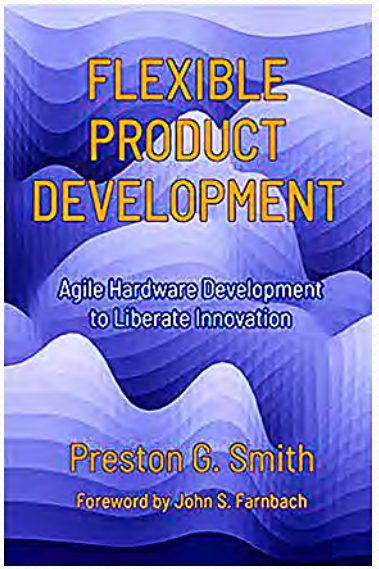
- Guides to lean economics, science, and thinking
- Illustrate key principles of just-in-time supply chains
- Keys to apply lean-thinking at strategic-tactical levels









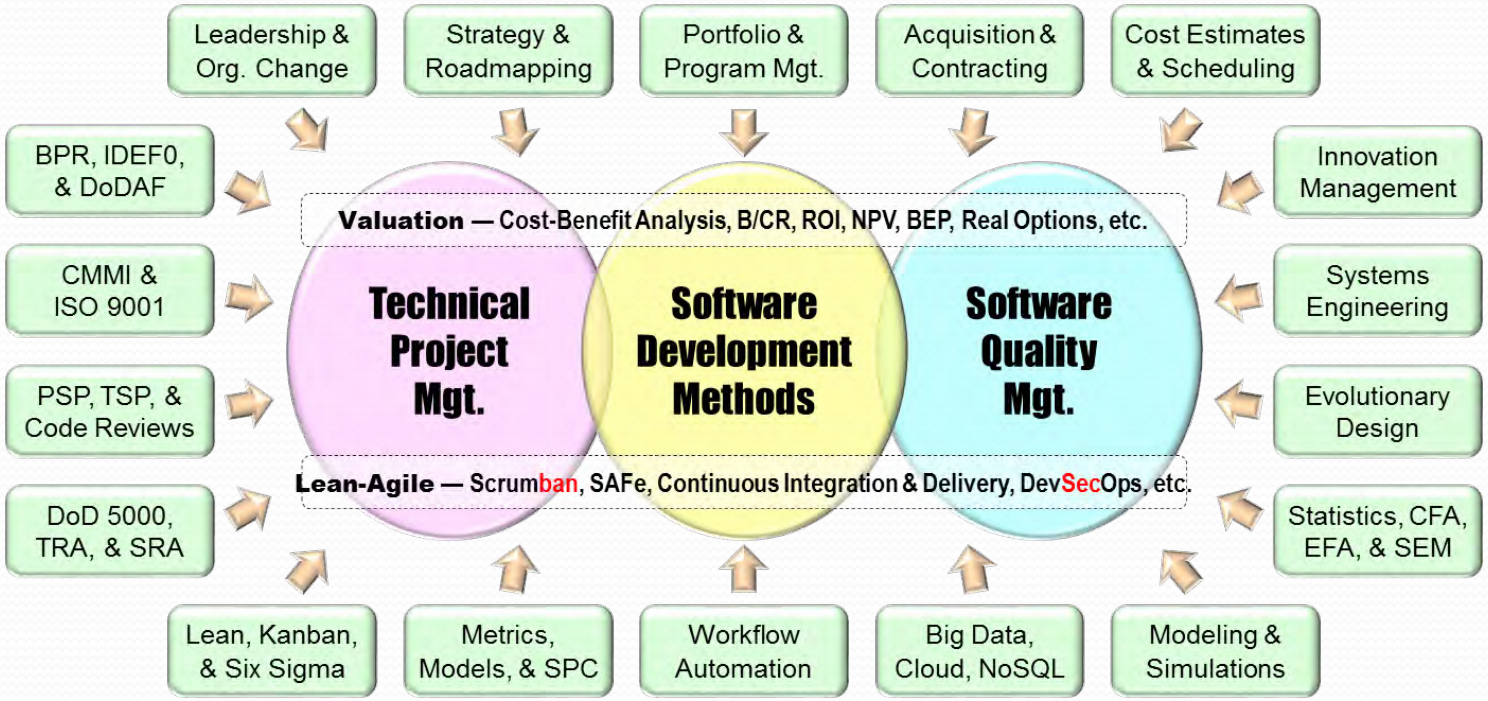


Lean Thinking Videos

- http://davidfrico.com/lean-startup.htm
- http://davidfrico.com/design-sprints.htm
- http://davidfrico.com/top-lean-videos.htm

Hopp, W. J. (2008). *Supply chain science*. Long Grove, IL: Waveland Press.
Hopp, W. J., & Spearman, M. L. (2008). *Factory physics*. Long Grove, IL: Waveland Press.
Reinertsen, D. G. (2009). *The principles of product development flow*: *Second generation lean product development*. New York, NY: Celeritas.
Pound, E. S., Bell, J. H., Spearman, M. L. (2014). *Factory physics*: *How leaders improve performance in a post-lean six sigma world*. New York, NY: McGraw-Hill Education.
Smith, P. G. (2018). *Flexible product development*: *Agile hardware development to liberate innovation*. San Francisco, CA: Jossey-Bass.

Dave's Professional Capabilities



Economic Value of Agile Businesses, Enterprises & Organizations - http://davidfrico.com/value-of-business-agility.pdf

STRENGTHS – Lean & Agile Thinking • Enterprise Transformation & Roadmapping • 360 Leadership Assessments • Executive & Agile Coaching • Enterprise Business Agility • Agile Acquisition Contracts • Scaled Agile Framework (SAFe) • Development Security Operations (DevSecOps) • Cloud Computing & Amazon Web Services (AWS) • Portfolio, Program, & Project Mgt. • Lean-Agile Product Management & Design Thinking • 5x5x5 Innovation & Marketing Sprints • Annual & Quarterly Strategic Planning • Technology & Product Roadmapping • Program Increment & Big Room Planning • Emergent & Evolutionary Microservices • Exploratory MVP, MVA, & MMF Experiments • Scrumban, Kanban & Lean-Agile Assessments • Performance Metrics, Measures & Dashboards • Agile lifecycle management (ALM) workflow tools ...

39+ YEARS IN IT INDUSTRY

DR. DAVID F. RICO, PMP, CSEP, EBAS, BAF, ACP, CSM, SAFE, DEVOPS, AWS
LEAN-AGILE · CI · CD · DEVSECOPS · CLOUD COMPUTING

email • dave1@davidfrico.com
website • http://www.davidfrico.com
youtube • http://davidfrico.com/daves-videos.htm

twitter • http://www.twitter.com/dr_david_f_rico • follow
linkedin • http://www.linkedin.com/in/davidfrico • connect

• http://www.davidfrico.com/daves-background.pdf • background
• http://www.davidfrico.com/daves-capabilities.pdf • capabilities
• http://www.davidfrico.com/daves-timeline.pdf • agile timeline
• http://www.davidfrico.com/daves-portfolio.pdf • agile portfolio
• http://www.davidfrico.com/daves-skillsets.pdf • agile skills

PMP, CSEP, EBAS, BAF, FCP, FCT, ACP, CSM, DEVOPS, AWS & SAFE