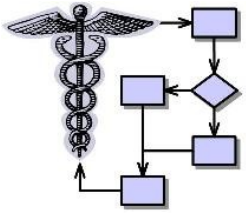


ShoeBar Associates

Lean-Agile Partners

When It Just *Has* to Work:

Agile Development in Safety-Critical Environments

Brian Shoemaker

ShoeBar Associates

Nancy Van Schooenderwoert

Lean-Agile Partners Inc.

Nancy's Background

- 15 years safety-critical systems experience
- 10 years agile team coaching
- 3 years agile enterprise coaching
- Industries: Aerospace, Medical Devices, Sonar Weaponry, Scientific Instruments, Financial Services
- Electrical Engineering and Software Engineering, embedded systems

ShoeBar Associates Lean-Agile Partners

Brian's Background

- Originally an analytical chemist
- 15 y in clinical diagnostics (immunoassay):
  analytical support → assay development → instrument software validation
- 6 y as SW quality manager (5 in clinical trial related SW)
- 4 y as independent validation consultant to FDA-regulated companies – mostly medical device
- Active in: software validation, Part 11 evaluation, software quality systems, auditing, training

ShoeBar Associates Lean-Agile Partners

When it just ***has*** to work: Agile Development in Safety-Critical Environments

- **Software too often contributes to poor safety**
- Lean principles → new style of organization & new tools
- Risk management benefits from iteration
- Essential elements: flexibility and learning, but rigor and documentation
- Teams report positive experiences

ShoeBar Associates Lean-Agile Partners

Software Can Compromise Safety

- Chemical plants
- Power stations (esp. nuclear)
- Aviation systems (civilian & military)
- Other transportation systems
- Medical devices

ShoeBar Associates Lean-Agile Partners

Safety Critical = Regulated

- Some industries (e.g. aviation): regulations prescribe methodology
- FDA-regulated (medical devices, blood banks): use any method, but show the outputs

ShoeBar Associates Lean-Agile Partners

S/W Safety Examples Abound

- **September 16, 2008: Automated External Defibrillator recalled**
  http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfRES/res.cfm?ID=73249
  Device was configured with the incorrect software, for semi-automatic instead of fully automatic use. When needed for a cardiac arrest emergency, device will require user to press the shock button instead of automatically delivering a shock (but the shock button is covered).
- **August 2, 2008: MultiLeaf Collimator recalled**
  http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfRES/res.cfm?ID=68342
  Charged-particle radiation therapy system. ***Dose Calculation Error***: Software anomaly may result in failure of a MLC leaf to reach planned position, potentially resulting in misadministration of dose to a patient.
- **September 2006: Infusion pump**
  http://www.fda.gov/MedicalDevices/Safety/RecallsCorrectionsRemovals/ListofRecalls/ucm(

  Touch-sensitive keypad used to program the pump sometimes registers a number twice when it has been pressed only once (“key bounce”). Thus the pump would deliver more than the intended amount of medication, leading to over-infusion and serious harm or death to the patient.

ShoeBar Associates Lean-Agile Partners

S/W Safety Examples Abound

- **June 6, 2006: Ventilators recalled**
  http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfRES/res.cfm?id=46439
  Older generation software - incorrect oxygen cell calibration (without compressed air supply) - can ***disable*** all alarms
- **March 6, 2006: Dialysis device recalled**
  http://www.fda.gov/MedicalDevices/Safety/RecallsCorrectionsRemovals/ListofRecalls/ucm0

  Class I recall of dialysis device (11 injuries, 9 deaths): excessive fluid loss may result if caregiver overrides device's "incorrect weight change detected" alarm. (Device used for continuous solute and/or fluid removal in patients with acute renal failure.)
- **June 2001: Radiation treatment planning software**
  http://www-pub.iaea.org/MTCD/publications/PDF/Pub1114_scr.pdf
  Software used to calculate dose duration for radiation treatment of cancer would allow use of no more than four protective blocks (stated in the user guide). Physicians at Natl. Cancer Institute in Panama devised a way to "fool" the software into using five blocks, by entering data as if they were a single shape. If coordinates entered a specific way, the calculated dose would be as much as twice that intended. Users did not confirm calculated results; at least five patients died as a direct result of overexposure to radiation.

ShoeBar Associates Lean-Agile Partners

Right problem, wrong solution

- Software issues prompt significant number of recalls
- Many still claim solution lies in rigorous, stepwise development
- Our view is that a different lifecycle is needed
- But we arrive at the same goal

ShoeBar Associates Lean-Agile Partners

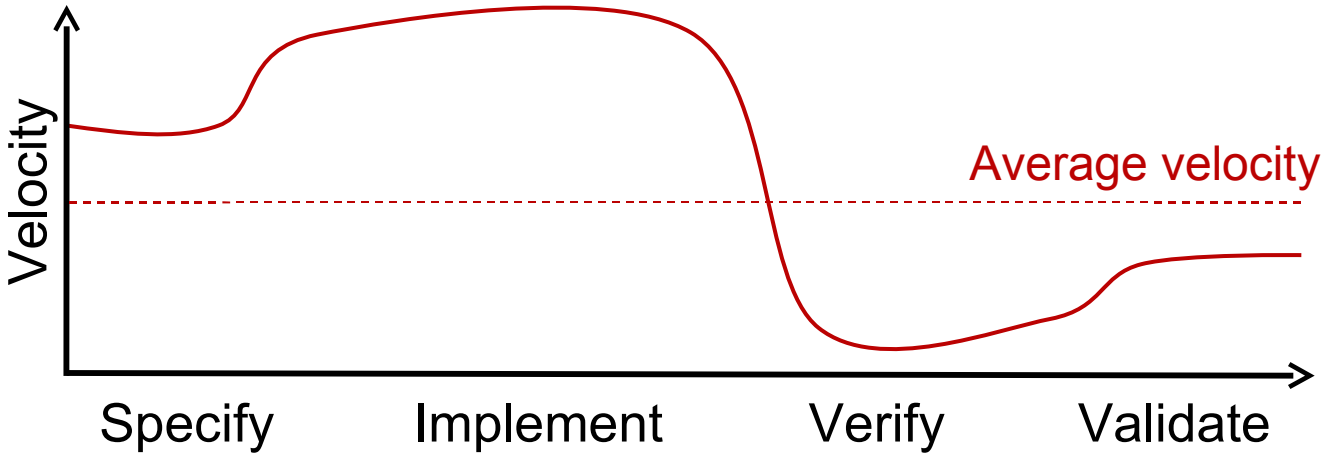
When it just ***has*** to work: Agile Development in Safety-Critical Environments

- *Software too often contributes to poor safety*
- **Lean principles → new style of organization & new tools**
- Risk management benefits from iteration
- Essential elements: flexibility and learning, but rigor and documentation
- Teams report positive experiences

ShoeBar Associates Lean-Agile Partners

Common scenario…

- Project velocity varies greatly
- Much slower at integration time



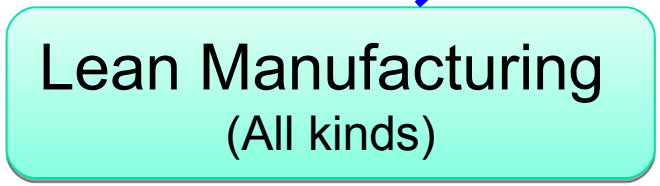
Solution: Pace yourself! It's a marathon, not a sprint

ShoeBar Associates
Lean-Agile Partners

Lean Thinking

Lean Principles:

Zero Defects
Minimize Work In Progress
Continuous Improvement



≠

Our “pain points”:

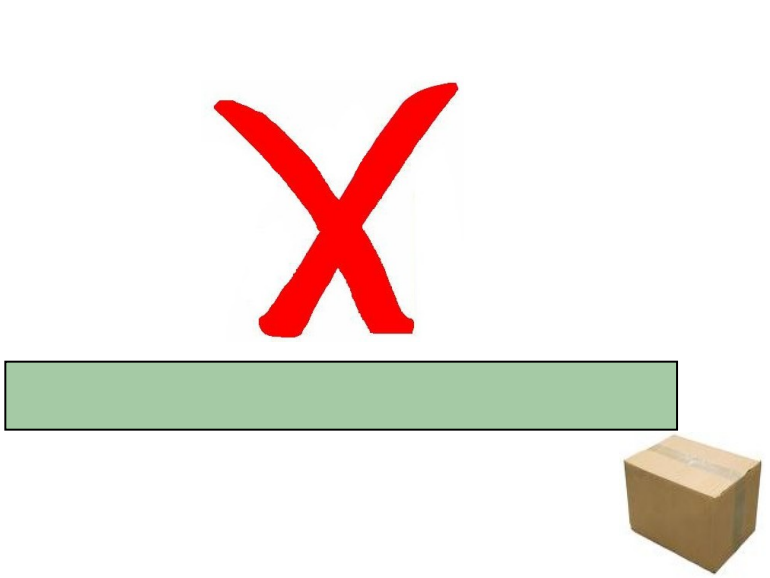
Bad news late in projects
Implementation different from spec
Documentation issues

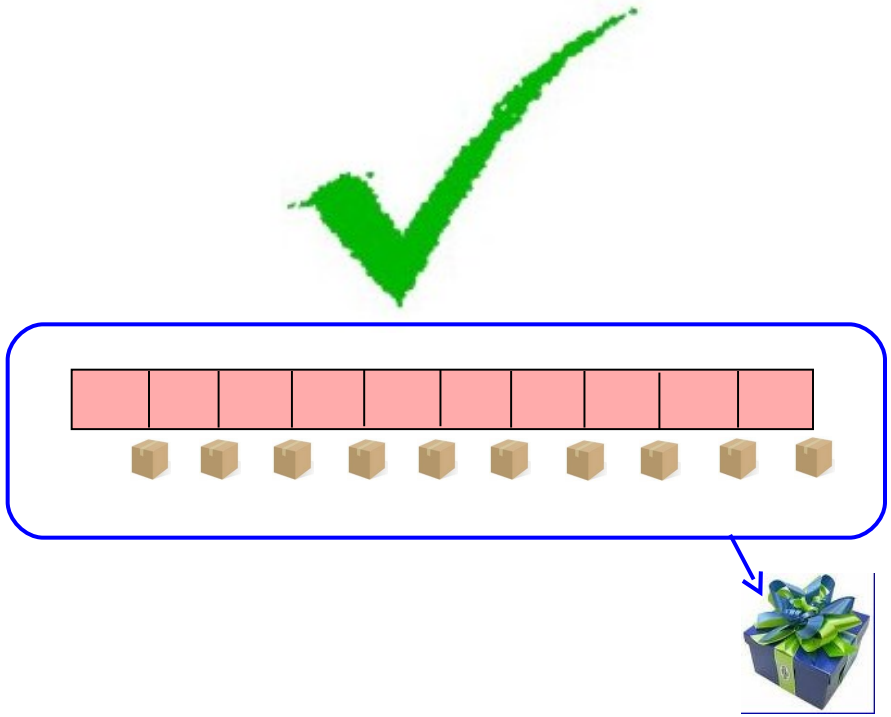
Classic “best practices”
Agile practices:
- Continuous Integration
- Automated unit tests
- Small co-located teams

ShoeBar Associates
Lean-Agile Partners

Avoid late integration

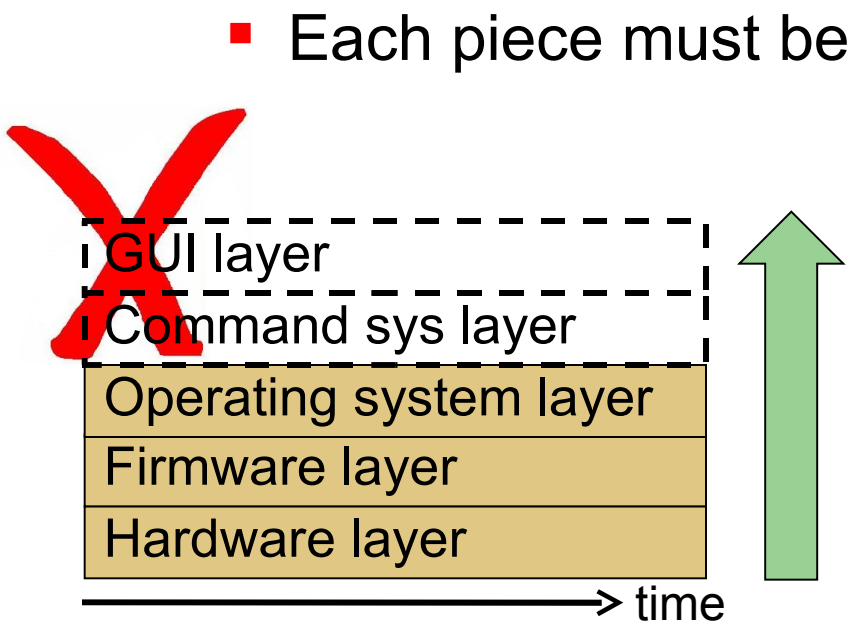
- Integrate new work as you go
- Incremental deliveries early & often

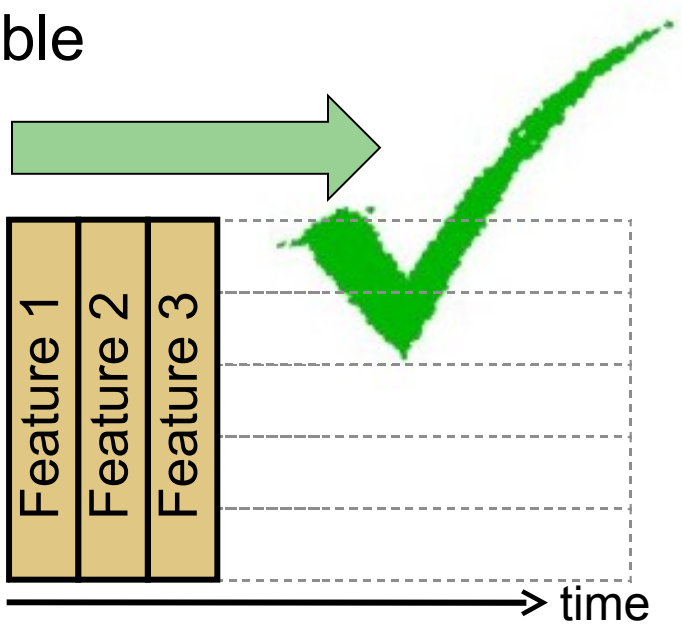




Deliver incrementally

- To deliver incrementally you must:
  - Carve the work into functional pieces
  - Each piece must be small
  - Each piece must be testable





ShoeBar Associates

Lean-Agile Partners

Work pieces: user stories

- User stories are similar to use cases
  - Written from customer view point
  - Written using words all understand
- Smaller than use cases
- Estimates are owned by the team
  - *Equally* likely to be too high or too low

ShoeBar Associates Lean-Agile Partners

Example user story

- Story – Card, Conversation, Confirmation –
  headline, narrative, test

Cards have the headline

Story

Verify Sensor Module OS runs on the new Sensor Module Radar

Narrative details captured in documents

Conditions of Satisfaction

Both In and Out values are displayed and out value should equal to 2*In value

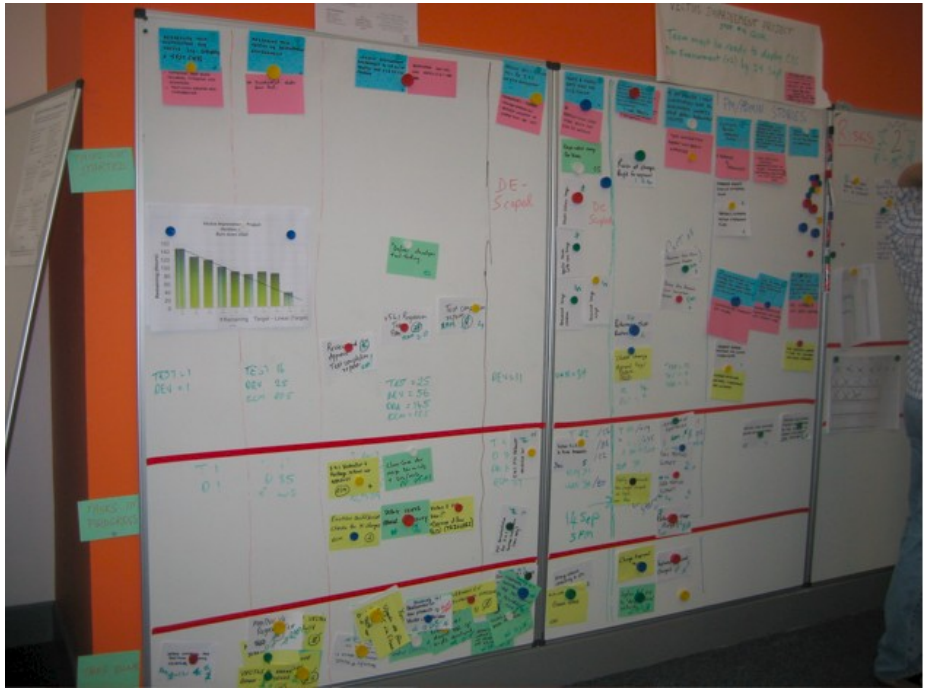
CoS becomes the root of story acceptance test

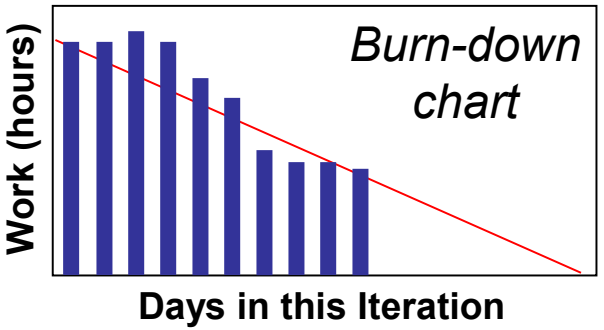
An old idea: If you have a clear goal, you are much more likely to achieve it.

ShoeBar Associates Lean-Agile Partners

Total transparency

- Status reporting is not separate from team's own way of tracking their work



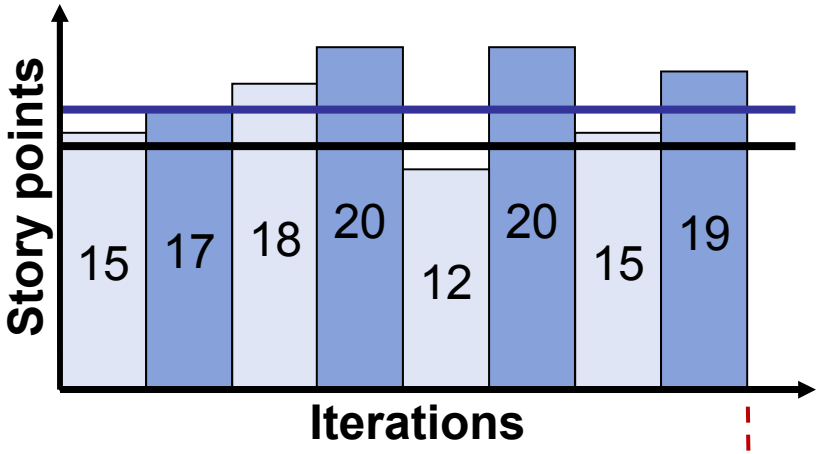


Each day:

- Team estimates hours remaining for each task
- All remaining hours are summed
- That total is today's data point on burn-down chart

ShoeBar Associates Lean-Agile Partners

Predictable project speed

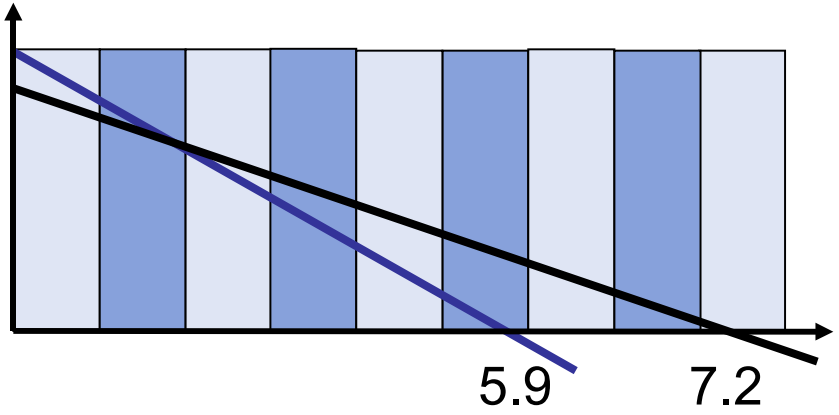


Mean (Last 8) = 17

Mean (Worst 3) = 14

Q. How long to finish project if 100 story points of work remains in product backlog?

A. If it's 14 points/iter, then it takes 7.2 iterations. If it's 17 points/iter. then it will take 5.9 iterations. You can be conservative or not, as appropriate.



ShoeBar Associates Lean-Agile Partners

4 stages of Story refinement

Story and CoS (Conditions of satisfaction) defined – from "user needs and intended uses"

Product Owner conducts the team in 'planning poker' story points estimation

At iteration planning meeting, team defines tasks & team-owned task estimates (hours)

During iteration, team pulls further detail that was not needed for estimation. PO validates stories, with Testers' help.

FDA considers software validation to be "**confirmation by examination and provision of objective evidence that software specifications conform to user needs and intended uses, and that the particular requirements implemented through software can be consistently fulfilled.**" – GPSV p. 6

Automated agile tests

GPSV = General Principles of Software Validation

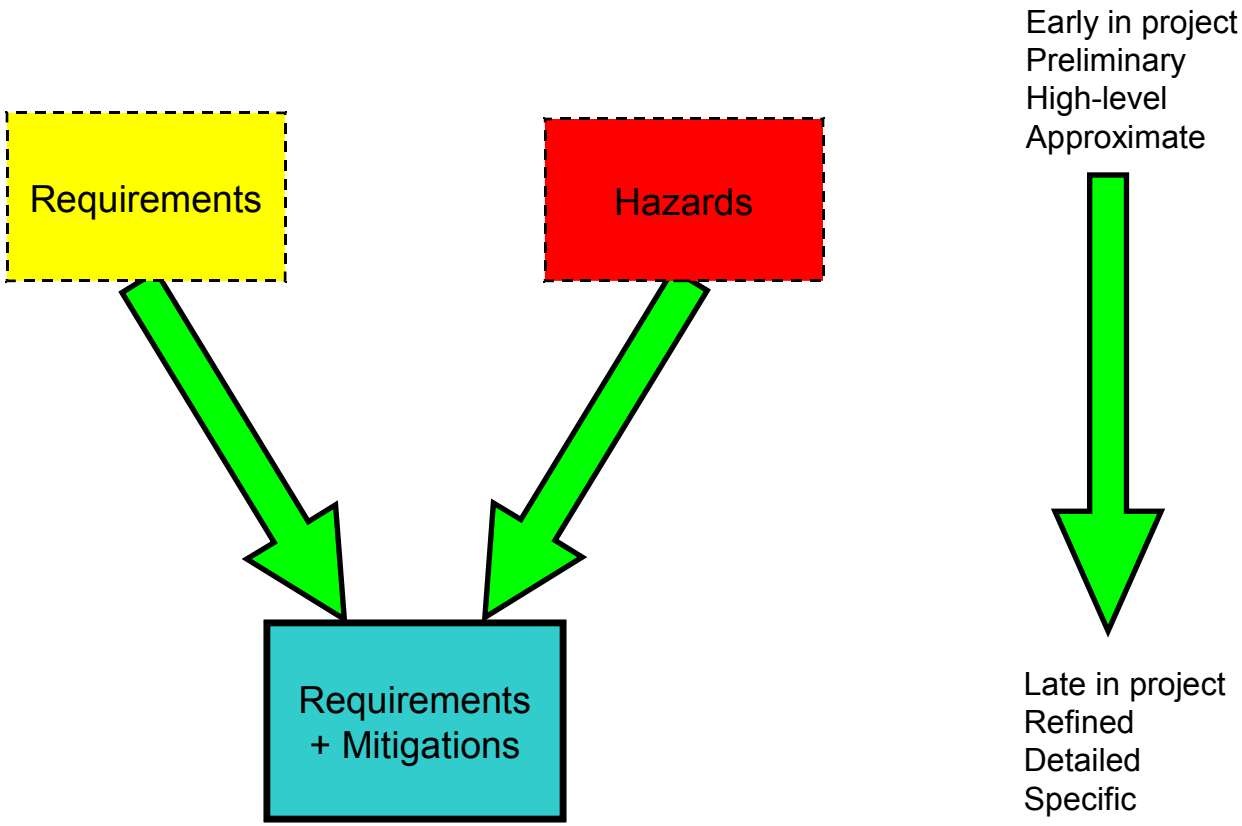
ShoeBar Associates Lean-Agile Partners

When it just ***has*** to work: Agile Development in Safety-Critical Environments

- *Software too often contributes to poor safety*
- *Lean principles → new style of organization & new tools*
- **Risk management benefits from iteration**
- Essential elements: flexibility and learning, but rigor and documentation
- Teams report positive experiences

ShoeBar Associates Lean-Agile Partners

Requirements / Hazards: Converging Analyses



ShoeBar Associates Lean-Agile Partners

Classical Risk Ranking

	Probability:			
Severity:	High	Occasional	Low	Remote
Major	U	U	U	A
Moderate	U	A	A	N
Minor	A	A	N	N

Acceptability is ranked as follows:

U = unacceptable – mitigation required

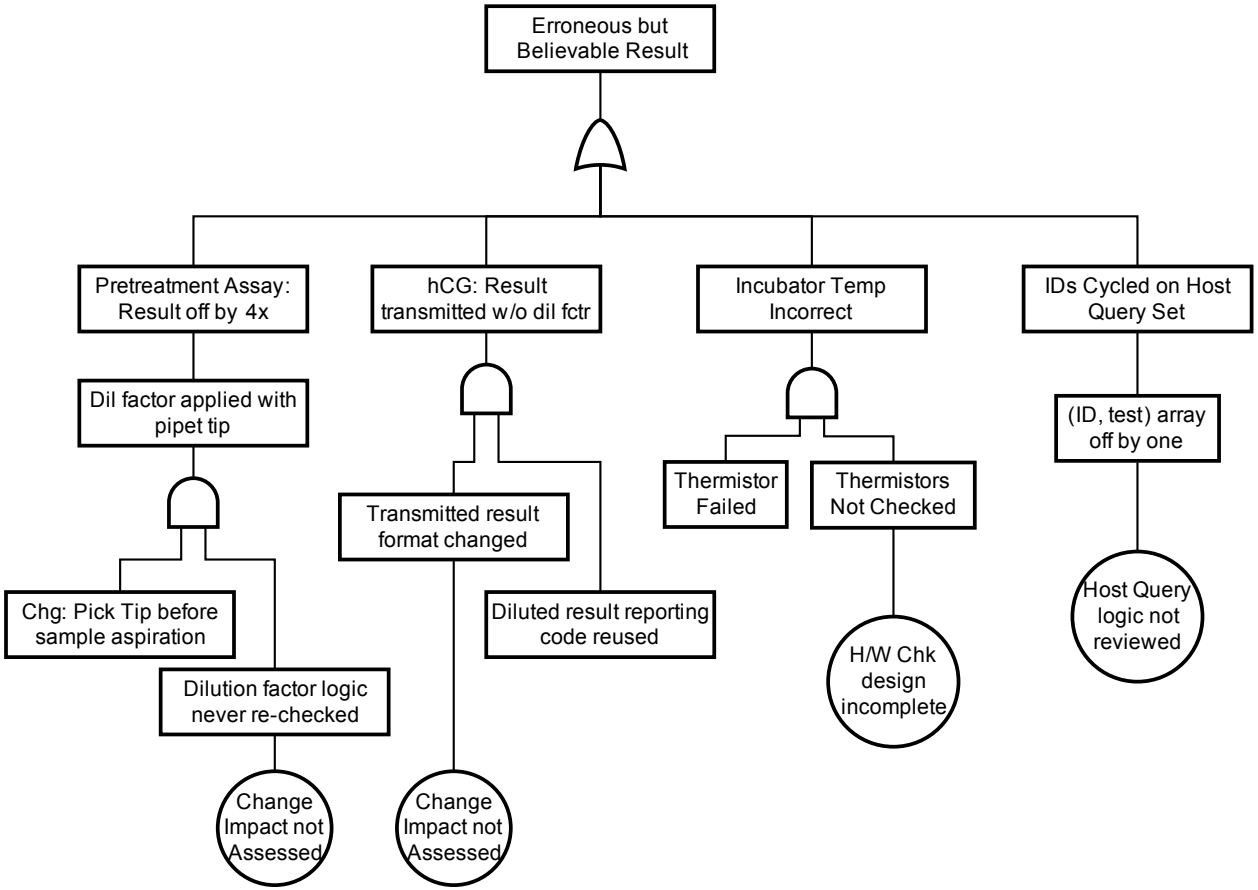
A = ALARP (as low as reasonably practicable) – mitigate as reasonable; risk decision must be documented and reviewed

N = negligible – acceptable without review

Risks: Analyze Early and Often

- Hazards identified through systematic methods (FMEA, FMECA, or FTA)
- Requirements: become more refined as design evolves
- Hazards: evaluate repeatedly throughout project
- Systematic analysis: best if we know the design
- Hazard mitigation: changing or adding to requirements

FTA: Work Back from Potential Hazards



ShoeBar Associates Lean-Agile Partners

FMEA: Build Up from Component Failures

Failure Mode	Effect	Causes	S1	Mitigation	S2
Sample ID / results array off by one	Wrong results reported	Inconsistent array logic; incorrect initialization	5	Optional – operator approve results before saving	2
Initialization fails to warm up lamp	Can't perform analyses	Startup logic can be set to skip steps and left that way	4	Reset all startup parameters on initialization	1
Dilution factor associated with pipet tip, picked in advance	Wrong result reported (dilution applied to wrong sample)	Counting logic not rechecked when pick-in-advance process introduced	5	(a) Track dilution and pipet tip separately; (b) show dilution with reported result	1

S1 = Severity rating before mitigation; S2 = severity rating after mitigation
Severity (sample values only): 5 = critical; 1 = nuisance
Note this analysis does not include two of the standard engineering estimates: occurrence (probability) or detection.

ShoeBar Associates Lean-Agile Partners

Hazards: Often Caught in Context

- **Direct failure**
  Software flaw in normal, correct use of system causes or permits incorrect dosage or energy to be delivered to patient.
- **Permitted misuse**
  Software does not reject or prevent entry of data in a way that (a) is incorrect according to user instructions, and (b) can result in incorrect calculation or logic, and consequent life-threatening or damaging therapeutic action.
- **User Complacency**
  Although software or system clearly notes that users must verify results, common use leads to over-reliance on software output and failure to cross-check calculations or results.

ShoeBar Associates Lean-Agile Partners

Hazards: Often Caught in Context

- **User Interface confusion**
  Software instructions, prompts, input labels, or other information is frequently confusing or misleading, and can result in incorrect user actions with potentially harmful or fatal outcome.
- **Security vulnerability**
  Attack by malicious code causes device to transmit incorrect information, control therapy incorrectly, or cease operating. No examples in medical-device software known at this time, but experience in personal computers and "smart" cellular phones suggests this is a serious possibility.

Lean-Agile adapts well to hazard mitigation

- Early analysis not static – review & revise as iterations proceed
- Users / product owner have multiple chances to uncover hazard situations
- Hazards can be simulated via “mock objects” in test suite
- Flexible, adaptive method can react to hazards learned during development (considered “negative user stories”)

When it just ***has*** to work: Agile Development in Safety-Critical Environments

- *Software too often contributes to poor safety*
- *Lean principles → new style of organization & new tools*
- *Risk management benefits from iteration*
- **Essential elements: flexibility and learning, but rigor and documentation**
- Teams report positive experiences

ShoeBar Associates Lean-Agile Partners

Know the objections & benefits

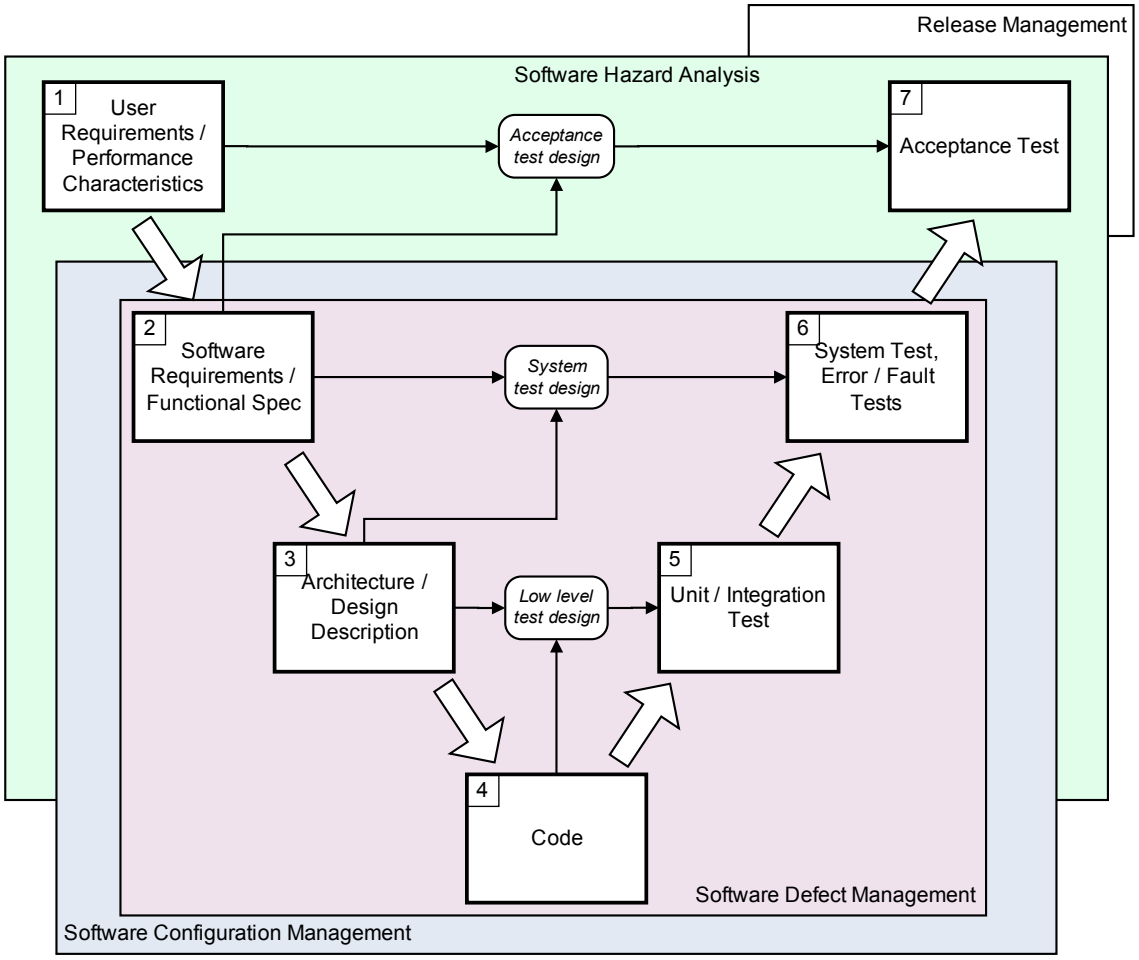
Points to counter:

- Lack of defined requirements
- Lack of structured review/release cycles
- Lack of documentation

Advantages to offer:

- Ability to resolve incomplete / conflicting requirements
- Ability to reprioritize requirements (mitigations) as system takes shape
- Many chances to identify hazards (controls not frozen too soon)

Know your SW lifecycle

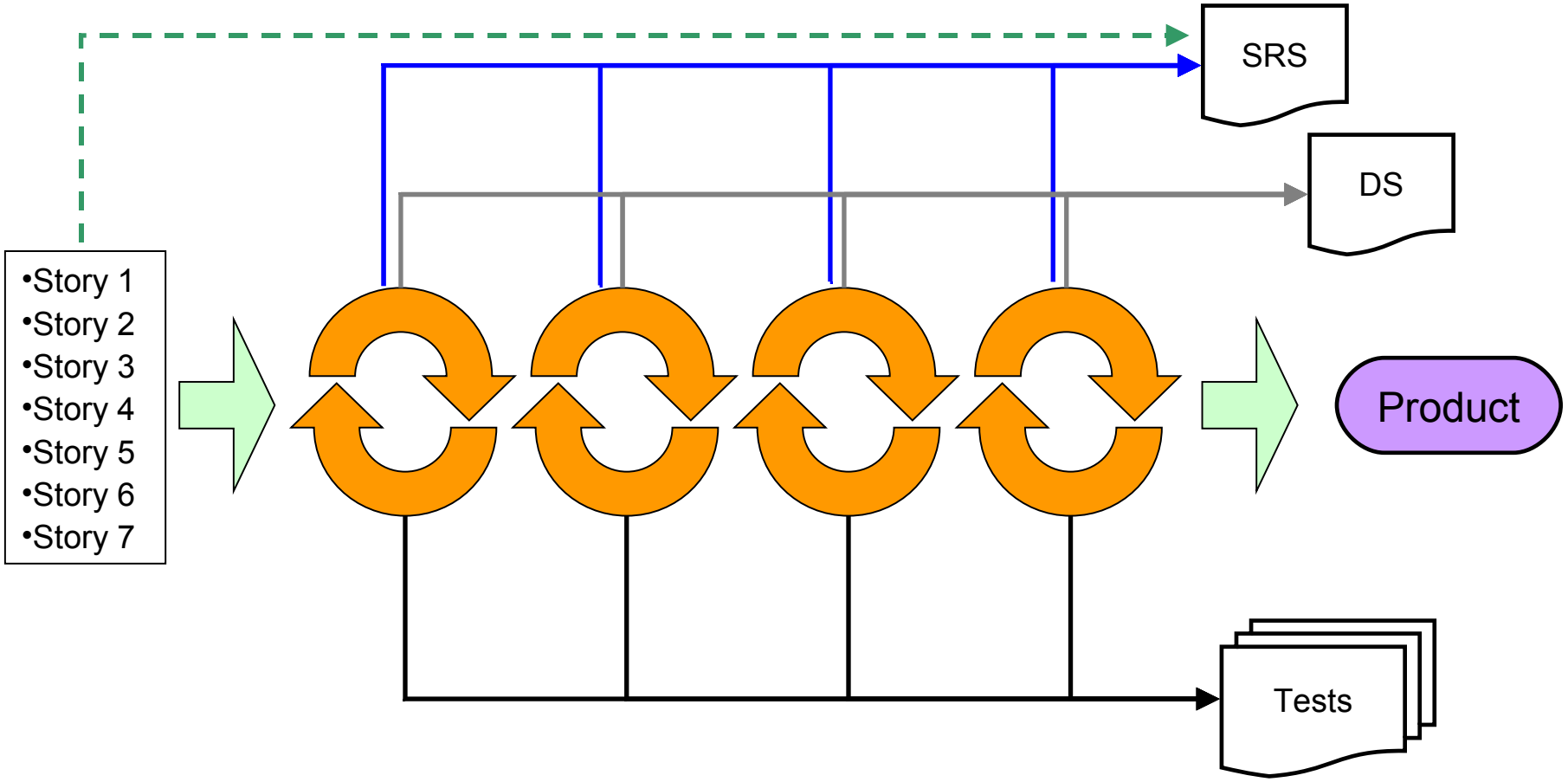


ShoeBar Associates Lean-Agile Partners

Document Effectively but Flexibly

- SOPs: focus on deliverables
  - Cover all required areas
  - Specify outputs, not strict order of completion
- Development outputs: focus on information
  - Requirements, architecture/design, hazard analysis
  - View as deliverables rather than process support
  - Consider nontraditional form if this makes information capture easier or more automatic

Capture knowledge as work proceeds



ShoeBar Associates Lean-Agile Partners

Use appropriate tools

- Initial user stories – may simply be index cards
- Requirements managed as they're elaborated
- Unit test harness
- Code-comment document extraction
- User-focused functional / system test engine – best if tied to requirements

ShoeBar Associates Lean-Agile Partners

Don’t forget communication

- With customer / product owner – for input and ongoing feedback
- Among team members – frequent but brief, to build team dynamics
- With management – to show progress and build trust

When it just ***has*** to work: Agile Development in Safety-Critical Environments

- *Software too often contributes to poor safety*
- *Lean principles → new style of organization & new tools*
- *Risk management benefits from iteration*
- *Essential elements: flexibility and learning, but rigor and documentation*
- **Teams report positive experiences**

ShoeBar Associates Lean-Agile Partners

What Do Defect Outcomes Suggest?

Team	Defects/Function Point	
Follett Software[1]	0.0128	agile
BMC Software[1]	0.048	agile
GMS[2]	0.22	agile
Industry Best[3]	2.0	traditional
Industry average[3]	4.5	traditional

1 Computed from data reported in Cutter IT Journal, Vol. 9, No. 9 (Sept 2008), page 10
2 “Newbies” paper presented at Agile 2006. See last slide for full reference.
3 Capers Jones presentation for Boston SPIN, Oct., 2002

ShoeBar Associates Lean-Agile Partners

Case: Device Software

- Authors compared one Agile and one non-Agile project:
  found that Agile gave lower cost, shorter development time, better accommodation of change, better test cases, and higher quality
- Used FDA's concept of "least burdensome approach" as part of their justification for using the Agile method
- Considered risk as integral part of development
- Iterative approach helped manage scope and limit feature creep

Source: Jenks & Rasmussen, Agile 2009.

Case: Comments

Developer:
"Control what you know, don't let it control you."

Client:
"At time of commercial launch, a number of features, once thought to be essential, were not included. Some were deferred as long as three years. Nonetheless, the product was considered highly successful and trading off nice to have features for three years of sales is an easy choice."

Source: Jenks & Rasmussen, Agile 2009.

Case: Reported Results

- High visibility – few surprises, able to manage and control
- Cost / duration: Agile project required 20-30% smaller team ***and*** shorter time, saved 35-50% cost, vs. non-Agile project
- Agile project gave higher quality – fewer overall defects, especially at end of project
- Agile project involved far better work-life balance and team morale (issues surfaced and managed in course of project, not saved for the end)

Source: Jenks & Rasmussen, Agile 2009.

ShoeBar Associates Lean-Agile Partners

Quote for the Day

"It is not the strongest of the species that survive, not the most intelligent, but the one most responsive to change."

- Charles Darwin

ShoeBar Associates Lean-Agile Partners

Recommended Reading

- *Implementing Lean Software Development* by Mary & Tom Poppendieck
- *Agile Estimating & Planning* by Mike Cohn
- *The Elegant Solution* by Matthew May
- *The Goal* by Eliyahu Goldratt
- *Release It!* by Michael Nygard
- *Safeware* by Nancy Leveson

References

- Cutter article by Michael Mah (on Follett, BMC Software), available by emailing him at michael.mah@qsma.com
- Papers by Nancy V. available no-charge, at http://www.leanagilepartners.com/publications.html
  - The Four Pillars of Agile Adoption
  - Embedded Agile Project by the Numbers with Newbies (Gives statistics reported for GMS team), presented at Agile 2006
- Weyrauch, Kelly, “Safety-Critical. XP Rules.”, *Better Software*, July/August 2004.
- EduQuest, Inc., "FDA Auditing of Computerized Systems and Part 11," notes from course given July 2005.

Standards – Software Safety

- AAMI TIR32:2004 Medical device software risk management
- IEC 60812:2006 (2nd ed) Analysis techniques for system reliability – Procedure for failure mode and effects analysis (FMEA)
- IEC 60601-1: 2005 (3rd ed) Medical electrical equipment – Part 1: General requirements for basic safety and essential performance *(60601-1-4 "Programmable Electrical Medical Systems" is available standalone, but will not be in the future)*
- IEC 62304:2006 Medical Device Software – Software Life Cycle Processes
- ISO 13485:2003 (2nd ed) Medical devices – Quality management systems – Requirements for regulatory purposes
- ISO 14971:2007 (2nd ed) Medical devices – Application of risk management to medical devices

References – FDA Documents

Design Control Guidance For Medical Device Manufacturers (March 11, 1997), http://www.fda.gov/cdrh/comp/designgd.html

General Principles of Software Validation (January 11, 2002), http://www.fda.gov/cdrh/comp/guidance/938.html

Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices (May 11, 2005), http://www.fda.gov/cdrh/ode/guidance/337.html

Off-The-Shelf Software Use in Medical Devices (Sep. 9, 1999), http://www.fda.gov/cdrh/ode/guidance/585.html

Cybersecurity for Networked Medical Devices Containing Off-the-Shelf (OTS) Software (Jan. 14, 2005), http://www.fda.gov/cdrh/comp/guidance/1553.html

ShoeBar Associates Lean-Agile Partners

Contact Information

Nancy Van Schooenderwoert
Lean-Agile Partners, Inc.
162 Marrett Rd., Lexington, MA 02421
781-860-0212
NancyV@leanagilepartners.com
http://www.leanagilepartners.com

Brian Shoemaker, Ph.D.
Principal Consultant, ShoeBar Associates
199 Needham St, Dedham MA 02026
781-929-5927
bshoemaker@shoebarassoc.com
http://www.shoebarassoc.com

ShoeBar Associates Lean-Agile Partners