

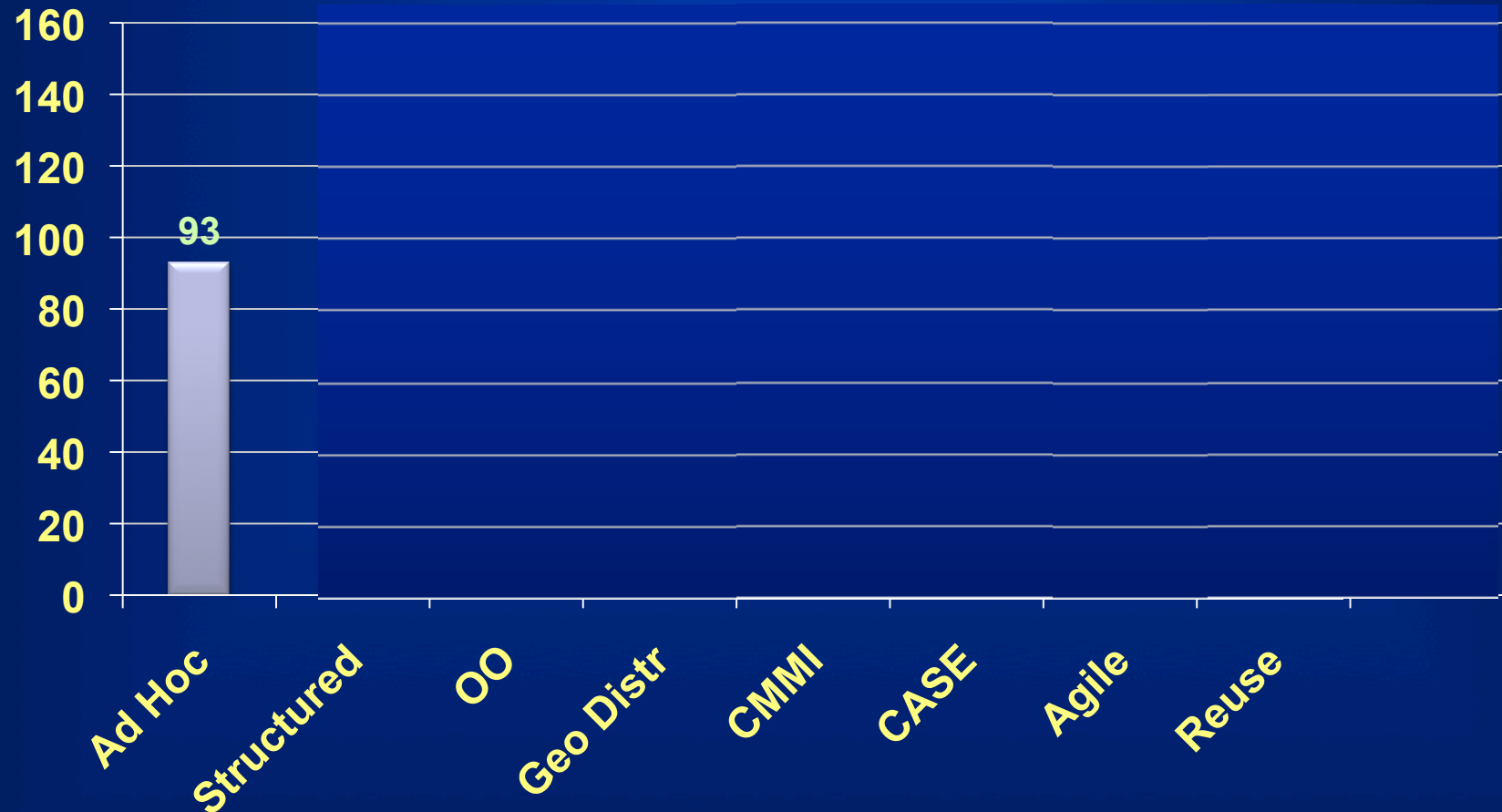
Lean, Agile, and the Human Condition

How Well Do Various Software Development Approaches Work?

- There have been many improvements to software development
- Everybody makes claims for their approach
- To compare them you must have a common measuring stick
 - Productivity, relative to structured programming
 - Quality, relative to structured programming
- With those results you can know which approaches work *best*

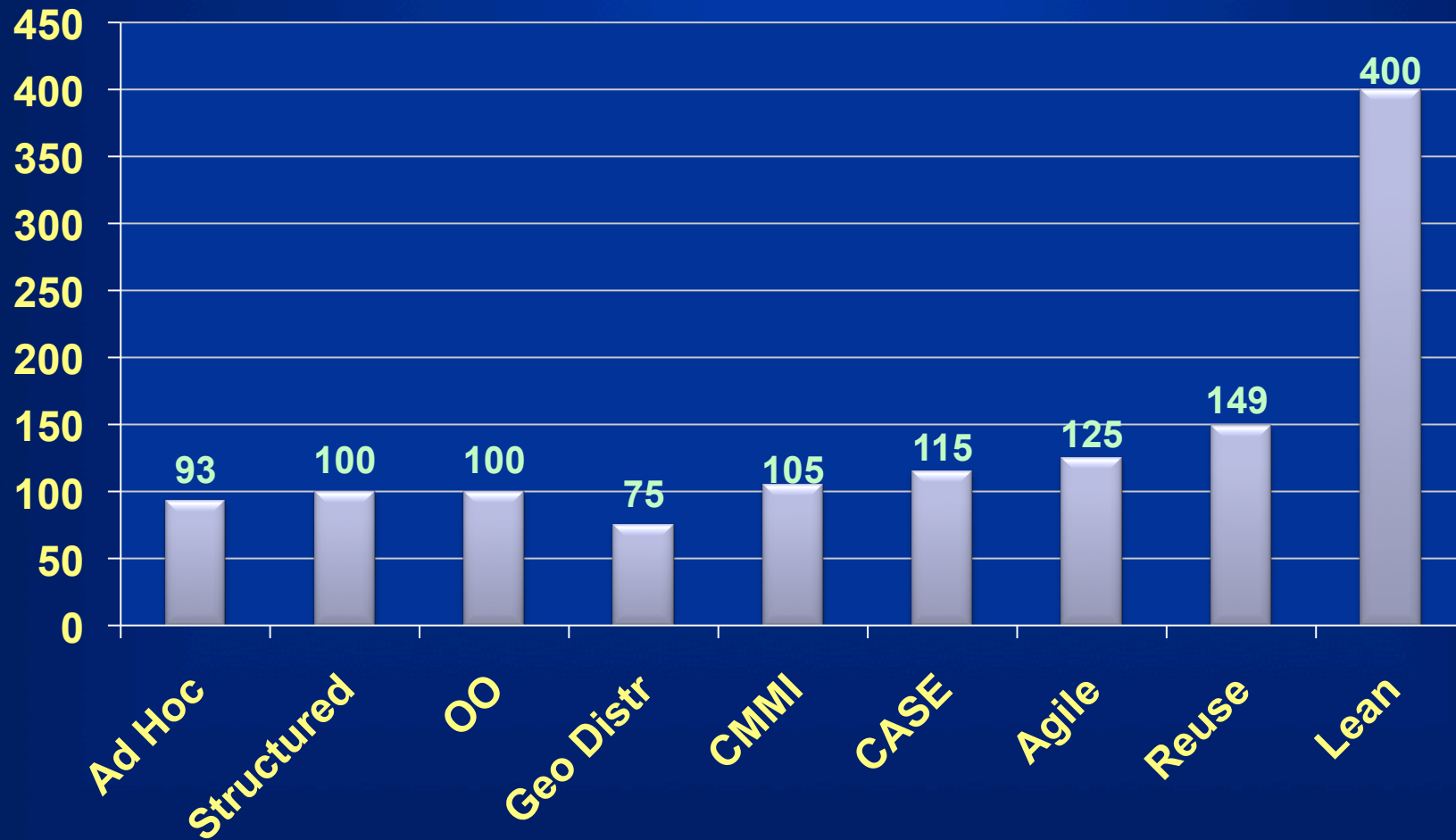


S/W Productivity (%)



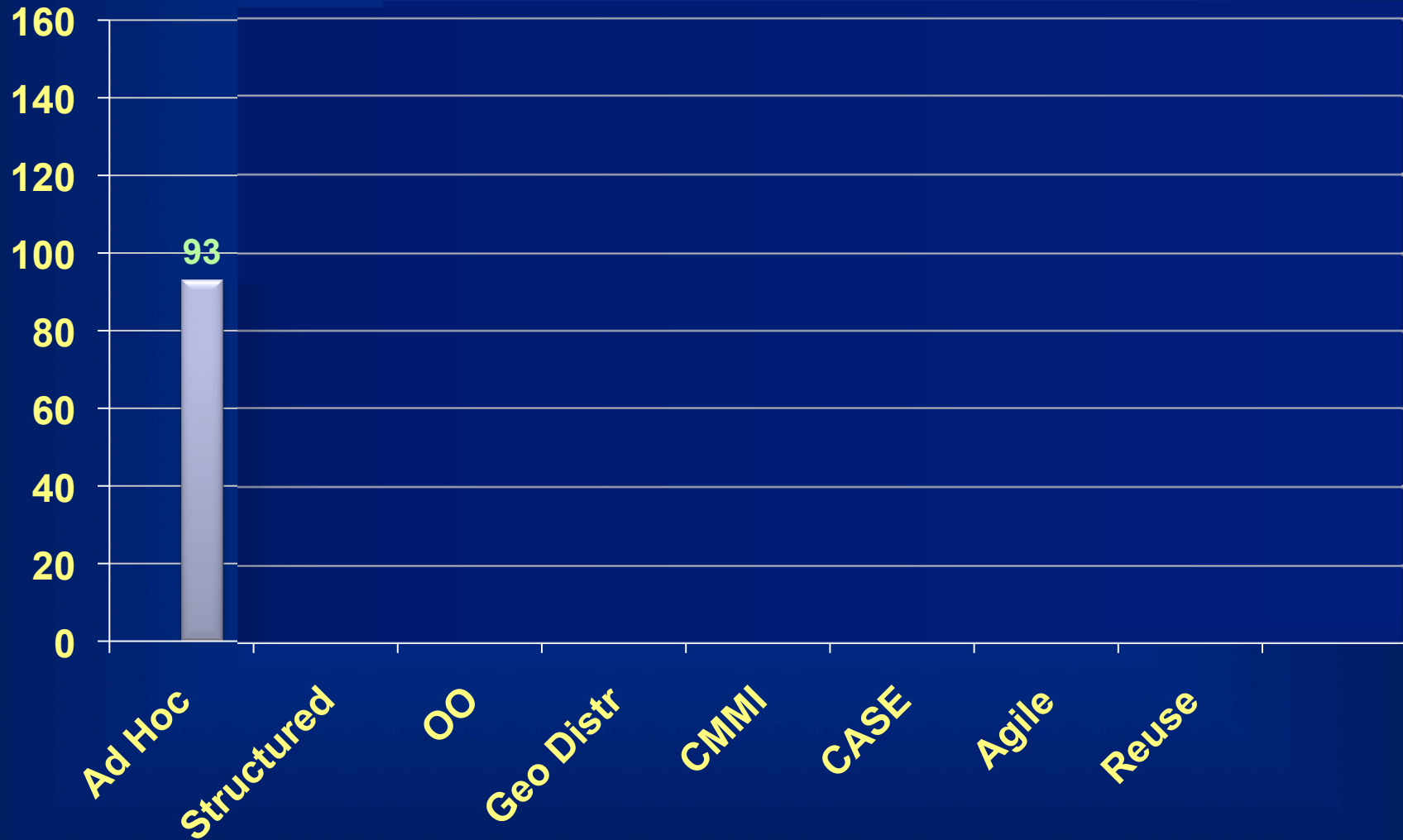


S/W Productivity (%)



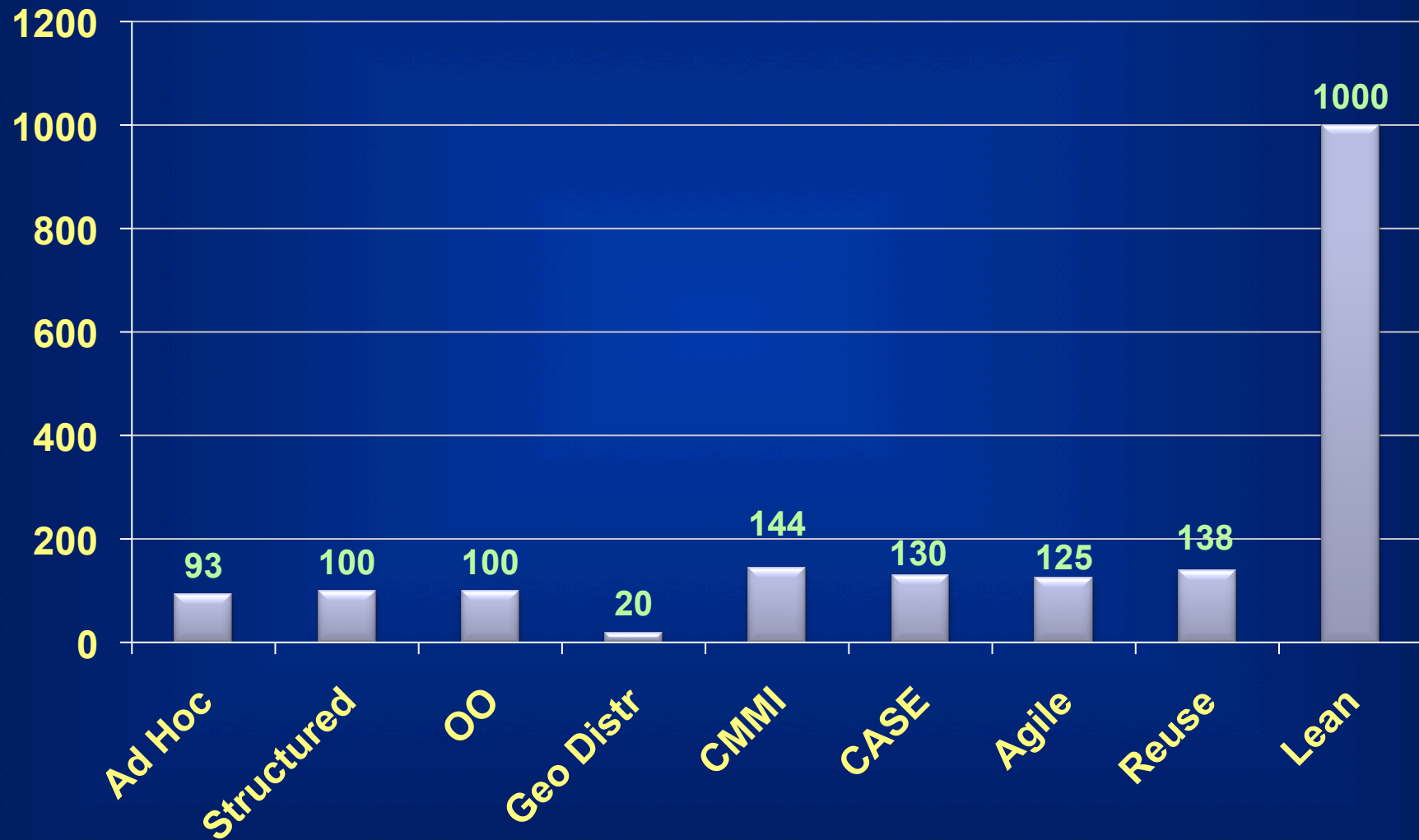


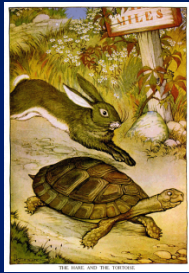
S/W Quality (%)



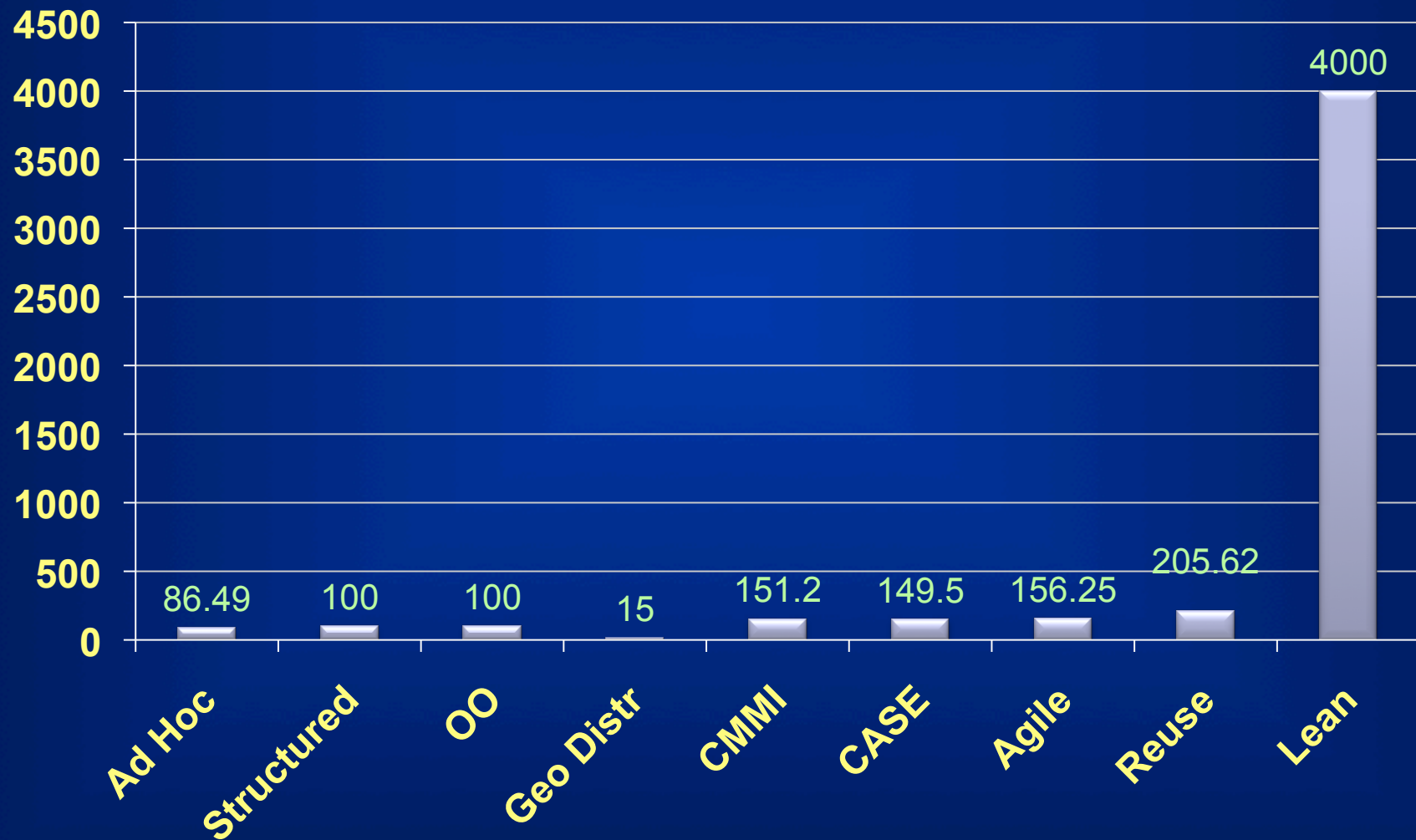


S/W Quality (%)





Productivity WITH Quality (%)



**Lean and Agile are by far the two
most effective software approaches**

Why?

Lean Production

- Term coined by Womack et al in "The Machine That Changed The World" (1990)
- **Subset of the "Toyota Production System" (TPS)**
- TPS is based on teachings of W. Edwards Deming

W. Edwards Deming (1900 – 1993)

- Everything is a system
- Put people first
- Master of quality, management, and marketing theories
- American WWII industrial success credited to his quality approach

**“Every day I think about what he meant to us.
Deming is the core of our management.”**



**Dr. Shoichiro Toyoda,
Toyota Chairman and
President 1982-1999**

**“Now more than ever, we need to
remember the teachings of Dr. Deming”**

Dr. Shoichiro Toyoda

"Deming-isms"

"There are many more things that must be managed than can be measured.
The inability to measure does not relieve us of the obligation to manage."

"Management Activities/Interventions will not improve unstable systems"

"We are going to learn about the prevailing system of management.
How to take joy out of life."

"It takes more than engineering knowledge to design...it takes Profound Knowledge"

"Those who only give us best efforts—let them stay home. Without the benefit
of Profound Knowledge they are counter-productive."

“Profound Knowledge”

“...for transformation from the present style of Western management to one of optimization.”

Deming, “Out Of The Crisis”

Facets:

- 1) “Appreciation for a system” (cooperation)
- 2) “Knowledge about variation” (problem recognition)
- 3) “Theory of knowledge” (predictive problem solving)
- 4) “Psychology” (aligning work on human nature)

“The System of Profound Knowledge is a paradigm shift. It’s not like “One-Minute Manager” or “Good to Great” where you can just pick up an isolated idea and run with it. You need to understand the whole system, and then do all of it. That’s what makes winning the Deming Prize so impressive: It takes five years, and must be top-down from senior management.”

Richard Zultner, Deming assistant from 1985 to 1993

What Is A "System?" (Deming view)

"A system is a set of inter-connected elements or parts that integrate the thoughts and actions of people [to] achieve a collectively agreed-upon purpose."

David Cochran

"Collective System Design: Designing Sustainable Systems"

"Every organization is a system that takes inputs from suppliers and transforms them using processes."

Steve Horn

"Deming's System of Profound Knowledge"

"The East is learning scientific thinking faster than the West is learning systems thinking."

Dr. Russell Ackoff

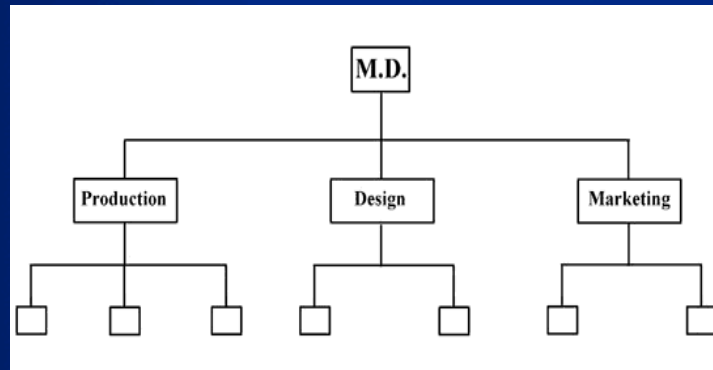
Professor Emeritus, Management Science, Wharton School

System-Thinking: The Key to Big Gains

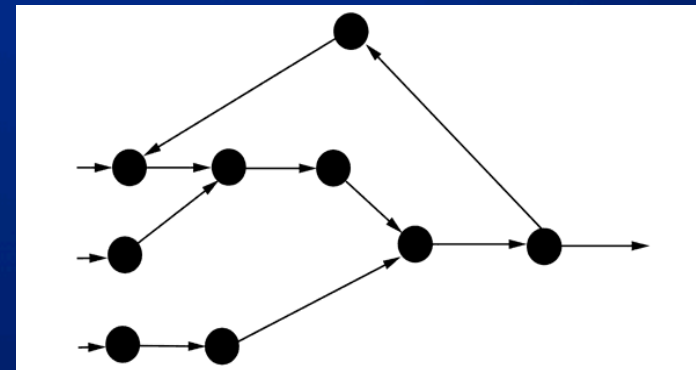
“The nub of 'system thinking' is that the big gains come from improving the way the system works and increasing co-operation between departments. There is relatively little to be gained by motivating people to just work a bit harder and make fewer mistakes.”

Steve Horn

“Deming's System of Profound Knowledge”



Autonomous actors

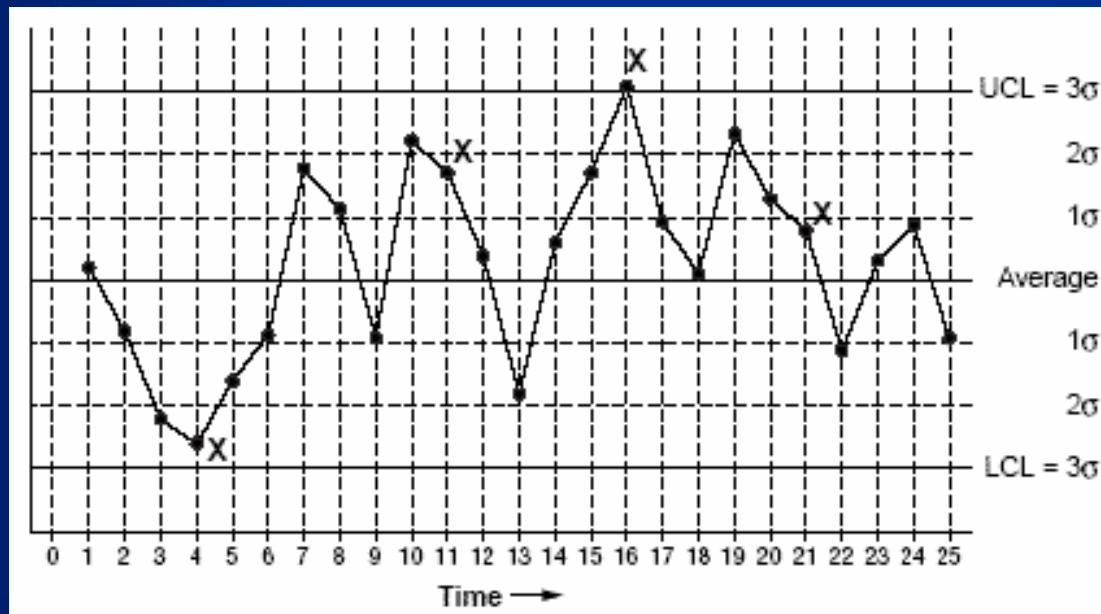


Interdependent actors

Tools Without A System View Are Ineffective

“These days you can see Shewhart control charts in many manufacturing operations in Europe and America. But most of these companies get very little benefit from the charts because they do not know how to act as part of a system.”

Steve Horn
Ibid.



(chart from
www.asq.org)

The Goal: Stable, Sustainable Systems

A sustainable system:

- a) knows its purpose, [Appreciation For A System]
- b) constantly seeks to achieve its purpose, [Knowledge About Variation]
- c) adapts quickly when [its] purpose is not being met. [Theory of Knowledge]

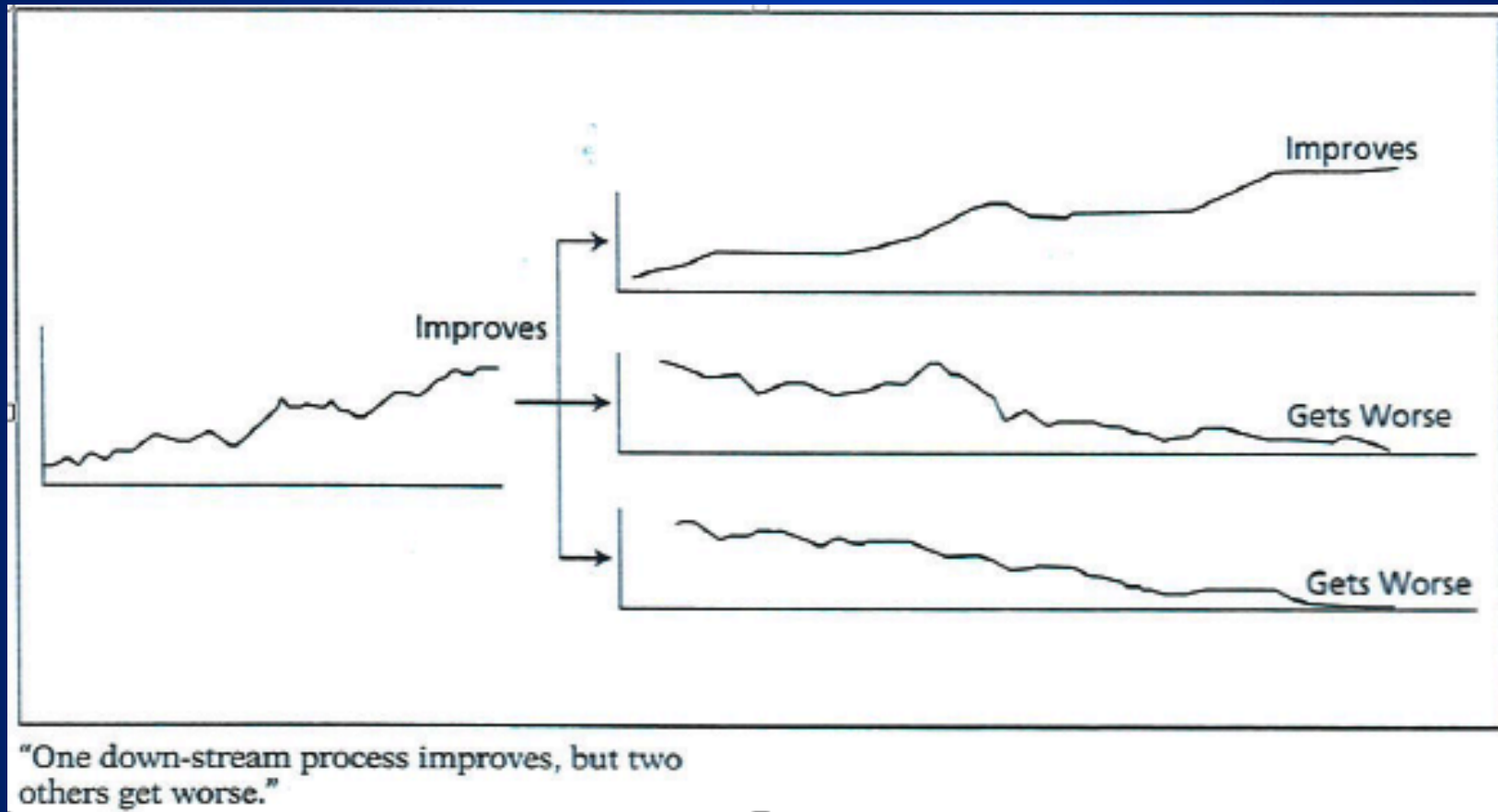
David Cochran
“Collective System Design: Designing Sustainable Systems”

All Four Facets of PK Are Essential

“The various segments of the system of profound knowledge proposed here can not be separated. They interact with each other.”

Deming
“The New Economics”

PK: The Cure For Sub-Optimization



"Without Profound Knowledge we end up optimizing one part of the system, but lose out elsewhere. One area may be greatly improved, but this can cause two others to decline. Without Profound Knowledge a manager would never know"

Deming, in "Four Days with Dr. Deming"

What Is Lean?

Lean is the application of Profound Knowledge to the system of business, including both product development and production, to yield results optimized for the greatest benefit of everyone involved: producers, customers, and society.

Lean has ethics at its heart:

- It seeks win/win outcomes for human beings
- It is more effective at generating wealth than the prior systems it replaced
- It incorporates other kinds of needs than wealth

Lean is:

- **A philosophy of work AND**
- **A map for implementation**

The Principles of Lean

- Value: what do the stakeholders need?
- Value Stream: minimize entropy of design
- Flow: remove whitespace and backflows
- Pull: work only based on customer need
- Perfection: right from the start, or fix fast

These allow us to learn lessons from other industries

Not All "Lean" Is Lean

Much of modern "Lean" takes pieces of Deming's work and implements it without changing the system; e.g., "Six Sigma," "kanban" software processes,...

This is "Lean," not Lean

What Is Agile?

Agile software development is a group of software development methodologies that are based on similar principles. Agile methodologies generally promote a project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages teamwork, self-organization and accountability, a set of engineering best practices that allow for rapid delivery of high-quality software, and a business approach that aligns development with customer needs and company goals.

Wikipedia
“Agile software development”

Agile Principles (Agile Manifesto)

- Satisfy the customer through early and continuous delivery of valuable software.
- **Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.**
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- **Business people and developers must work together daily throughout the project.**
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- **The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.**

Agile Principles (Continued)

- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity--the art of maximizing the amount of work not done--is essential.
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Lean/Agile Commonalities

- Principles-based
- Aligning to customer needs ("values")
 - Involving all stakeholders
 - Regular, timely delivery of product
- Effectiveness
 - Cyclical, adaptive, learning, continuous improvement
 - Technical excellence
 - Encouraging teamwork
 - Sustainable
- "Human-friendly"

Lean/Agile Differences

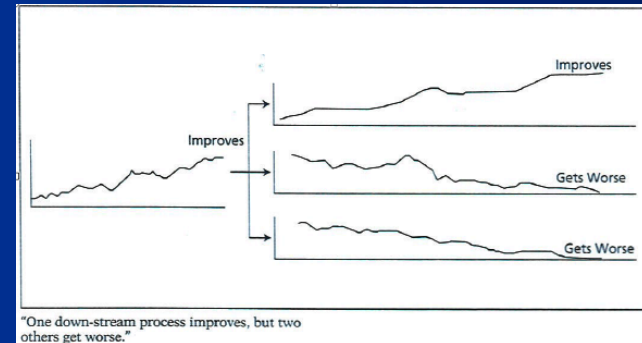
- Optimization
- **Level of Focus**
- Verification/Validation
- **Variation**
- Prediction

Optimization

- **Lean:** You must actively build up **Profound Knowledge** to obtain an optimized system

“It takes knowledge to optimize. Without Profound Knowledge we end up optimizing one part of the system, but lose out elsewhere in the system.”

Deming



Domain Design
Change-Driven Design

- **Agile:** Optimization will happen naturally, through **Emergent Behavior**

“Instead of committing to market a piece of software that hasn’t even been written yet, agile empowers teams to optimize their release as it’s developed”

Refactoring

agilemethodology.org

Level of Focus

- **Lean: Enterprise. System of Systems**

"The aim, the values and beliefs of the organization, as set forth by top management, are important. Without an aim, there is no system. The performance of any component is to be judged in terms of its contribution to the aim of the system."

Deming

Hoshin Kanri
QFD

- **Agile: Project. System.**

Extreme Modeling
Scrum Teams
Scrum of Scrums
Business participation

Verification/Validation

- **Lean: Do it right the first time AND evolve**
 - **Identify and eliminate classes of defects up front**
 - **Minimize testing and inspection (use sampling)**
 - **Corresponds to auto manufacture**
 - **Corresponds to chip manufacture**

Poka Yoke
Jidoka
Automation
Cleanroom
Formal methods

- **Agile: Evolve-to-right**
 - **Tests are the starting point, later run them all**

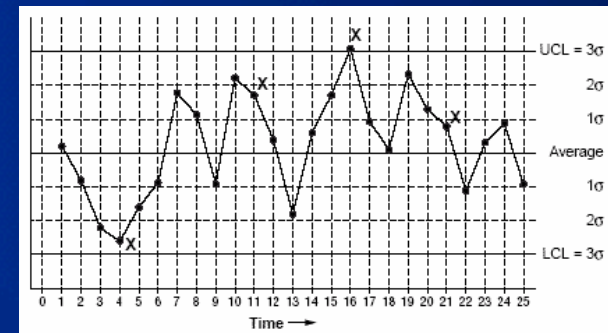
Test-driven development
Refactoring

Variation

- **Lean:** Treat business as a process and identify common ("natural") and special ("unnatural") causes of variation

"It is the manager's job to know the difference [between common and special variation]. Without this basic knowledge, any management action will be merely tampering. Managers working hard, yet making things worse."

Deming



Control Chart

- **Agile:** No concept for different causes of variation.

Prediction

- **Lean: Predict, systematically and by hypothesis**

“Management is prediction” - W. Edwards Deming

“Write down the key outcome that is desired. Then list the main drivers that impact that outcome. Then list design changes for each outcome to be tested with the PDSA cycle. When PDSA does not involve...a theory it becomes really just trial and error (try one thing then another then another).”

PDSA
TRIZ
Change-Driven Design

Tom Nolan

- **Agile: The future is unknowable. Do what you can with what you know now, and evolve as you learn**

“Much of what is “known” is not so.”

John Hunter
Management Consultant

Emergent Behavior
Short development cycles
Customer involvement

Lean/Agile Comparison

- **Agile: Effective. Simpler, less change to implement, less powerful. System-level. Great human values**
- **Lean: Effective. Harder, more change required, more powerful. System and SoS-level. Great human values**

“Agile is a really elegant solution for many organizations. It is local and team-focused. It is based on a number of simplifying assumptions, and is therefore easy to implement...much more so than Lean. In my talks with Agile advocates, they agree that each of the Lean tools is more powerful than its corresponding Agile tool. It also usually leads to sub-optimized solutions, which are difficult to recover from. However, difficulty in working effectively with Lean tools can make Agile the best solution for a group.”

Richard Zultner
Deming Assistant 1985-1993

“Get Prepared” vs. “Get Started”

Lean (Get Prepared)

George S. Patton

Ralph Waldo Emerson

Abraham Lincoln

Agile (Get Started)

Francis Ford Coppolla
and
George Lucas

School teachers
at the airport

Lean, Agile, and the Human Condition: "Put People First"

- People in general, customers in particular.
(Put the interests of others above your own)
- Be unflinchingly honest in your dealings. People notice, and remember.
- Contribute to society, not just to self
- Take some “beach time” for yourself each day (and also dedicated time for your spouse)
- Do business “by handshake”: if the other isn’t honest, no contract will make him act so
- Accept human “messiness”; don’t dehumanize the workplace

Lean, Agile, and the Human Condition: "Listen, Always"

- Be interested rather than trying to be interesting
 - come at any situation with an interested and curious mind -- life becomes much more interesting for everyone around you too
 - “Be interested in the people you meet -- their lives, their history, their story. Where are they from? How did they get here? What have they learned? put yourself into their frame of reference
- Listen to your "gut"...even if it contradicts those around you (don't just automatically accept it though)

Lean, Agile, and the Human Condition:

"Set A Worthy Vision and Goals"

- Something you love, and that, if others are involved, at least has elements they will love too
- A cause inspiring loyalty amongst your team
- Always serving others (must primarily benefit them)
- Consider timing when picking goals (observe trends up to the present, and extrapolate)
- Give people something of lasting value to them
- Fill a real void: Find everyday problems that nag at everyday people
 - Identify the customer's actual needs, not what you or they assume they need
 - No shortcuts in what you offer them or how

Lean, Agile, and the Human Condition: "Strategize"

- It's the rebels who truly make a difference
 - treat all non-absolutes as hypotheses to be disproven
 - Do not look to competitors to emulate them
 - Study competitors to identify the holes in what they offer
- **Conventional wisdom is always wrong**
- Involve others
 - Share the pie (makes a bigger pie for everyone)
 - Surround yourself with smarter people and guide them
 - be merciful to people's failures; give "second chances" (but speak the "truth in love")
 - Build loyalty between the team members
- **Anticipate what can go wrong, and head it off**

Lean, Agile, and the Human Condition: "Act"

- Be a doer, not just a talker [Agile!]
 - Ideas don't accomplish anything: results-oriented initiatives and strategies do. No results? Change your initiatives and strategies!
 - Many see needs and sit around waiting: Be the one who acts on it
 - Don't link your doing too tightly to your being compensated
 - Work hard, with high standards: it opens doors
 - Communicate sufficiently but without excess, and always personally (speak above write; no emails) [Agile!]
- Do whatever it takes to please the customer
- Treat everyone with respect and honesty
- Win/win only
 - In every deal, everyone walks away from the table feeling good
 - Never pretend to be someone else
- Continually renew innovation

Lean, Agile, and the Human Condition: "Measure Your Results"

- Based on accomplishment, not on time spent at job
- Choose your metrics based on your goals, not aimlessly or based on tradition
- Document what you learn from it

Redux

- Lean and Agile are the two most-effective software development approaches ever
- **Lean (not "*Lean*") is based on W. Edwards Deming**
- Deming emphasized cooperation, problem recognition, prediction, and people orientation
- **Agile is principled, simplified, and also people oriented**
- Great combination is Lean ("prepare and predict") with Agile ("get going and improve")

If you're interested in learning more about lean software development...



(Google “Crosstalk Lean Software”)