



# Agile Software Development

---

Orit Hazzan and Yael Dubinsky

Technion – Israel Institute of Technology



# Agile Software Development: Background

---

- Background
  - Software development models
  - Data
  
- Target: Improve software quality



# Agile Software Development: Main Ideas

---

- Whole team
  - development environment, roles, daily stand-up meetings
- Short releases
  - planning game, changes, gradual learning, abstraction
- Time estimations
  - resolution, who estimates
- Measures
  - visible means, management perspective



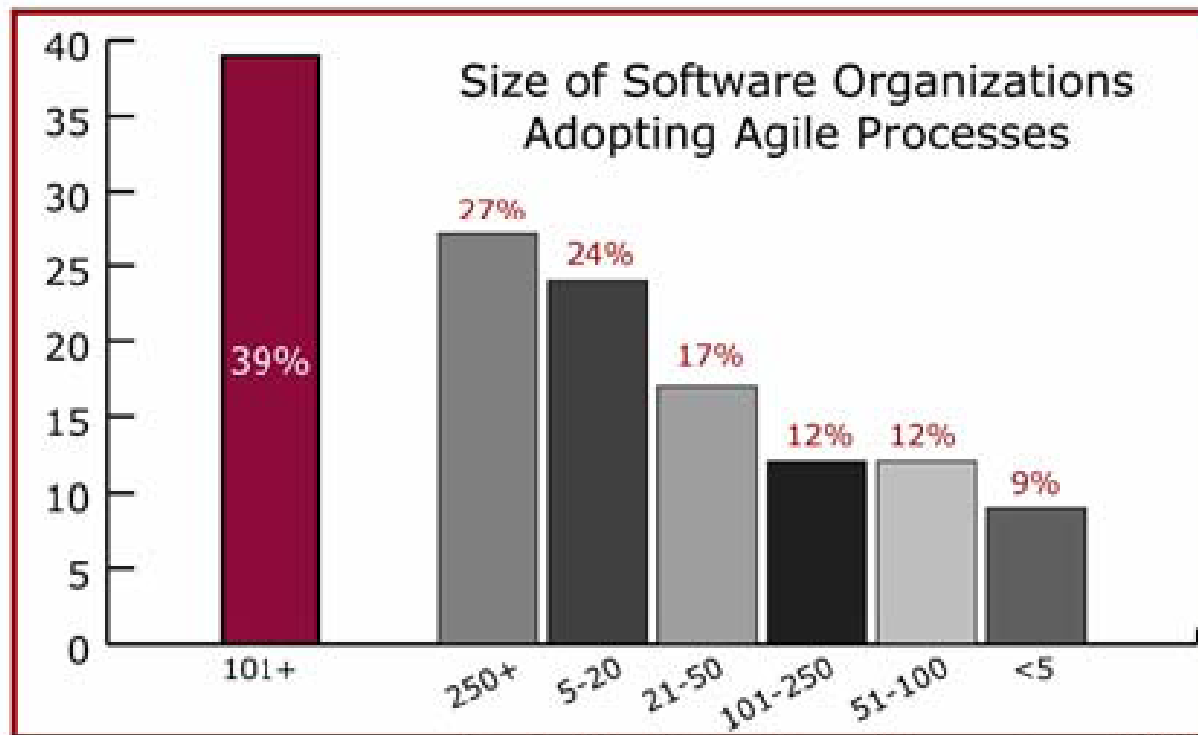
# Agile Software Development: Main Ideas

---

- Customer involvement
  - on-going, on-site
- Testing
  - unit test: first, automatic
  - acceptance tests
  - part of the development process
- Coding
  - pair programming, refactoring

Source: Agile Development: Results Delivered  
by VersionOne

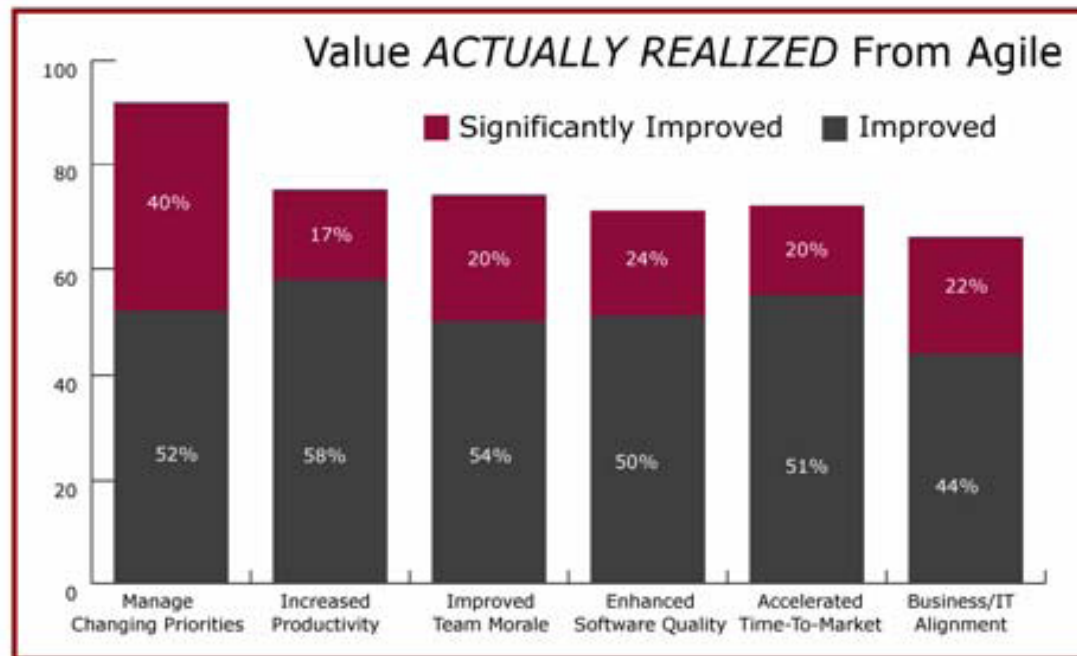
---



source: VersionOne/Agile Alliance Survey August 2006

Source: Agile Development: Results Delivered  
by VersionOne

---

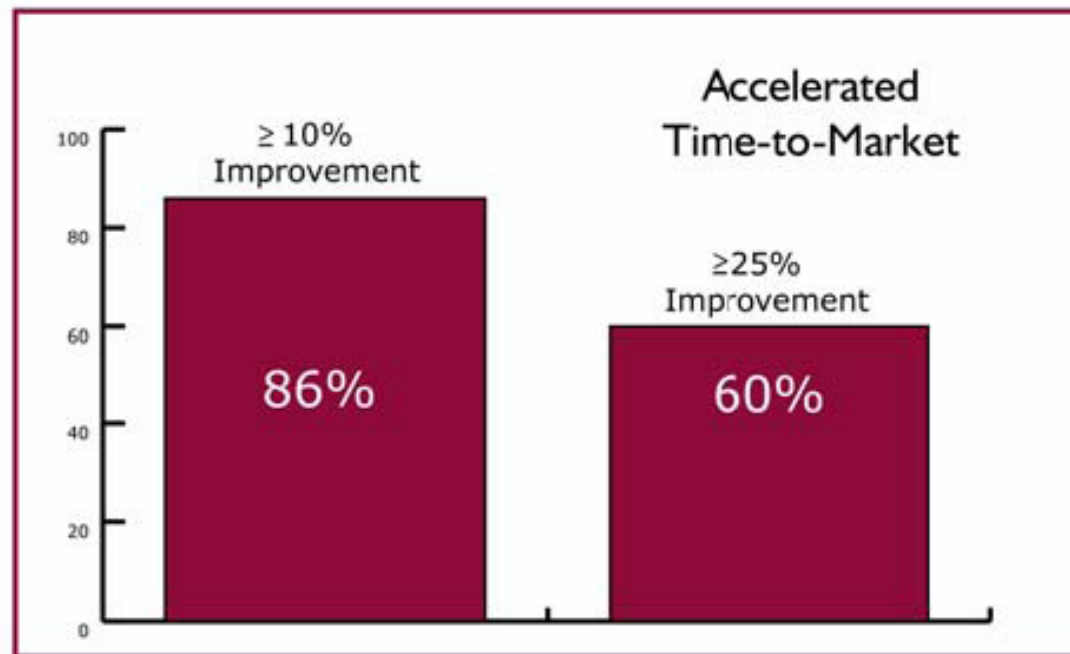


source: VersionOne/Agile Alliance Survey August 2006

---

Source: Agile Development: Results Delivered  
by VersionOne

---

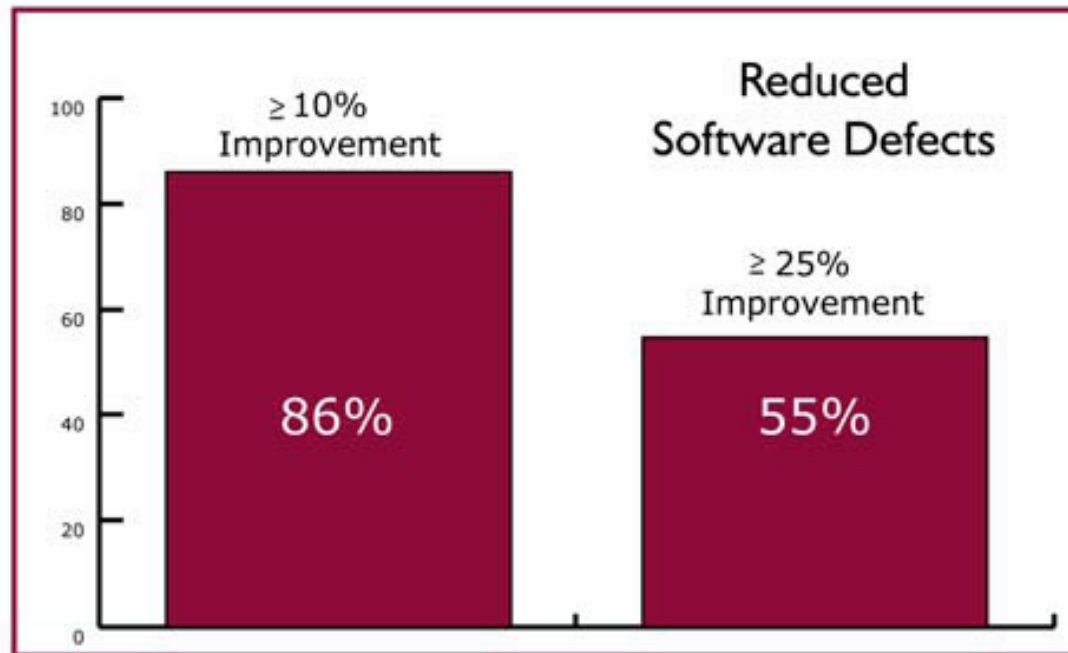


source: VersionOne/Agile Alliance Survey August 2006

---

Source: Agile Development: Results Delivered  
by VersionOne

---



source: VersionOne/Agile Alliance Survey August 2006



Manifesto for Agile Software Development - Windows Internet Explorer

http://agilemanifesto.org/

File Edit View Favorites Tools Help

Manifesto for Agile ...

# Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.  
Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

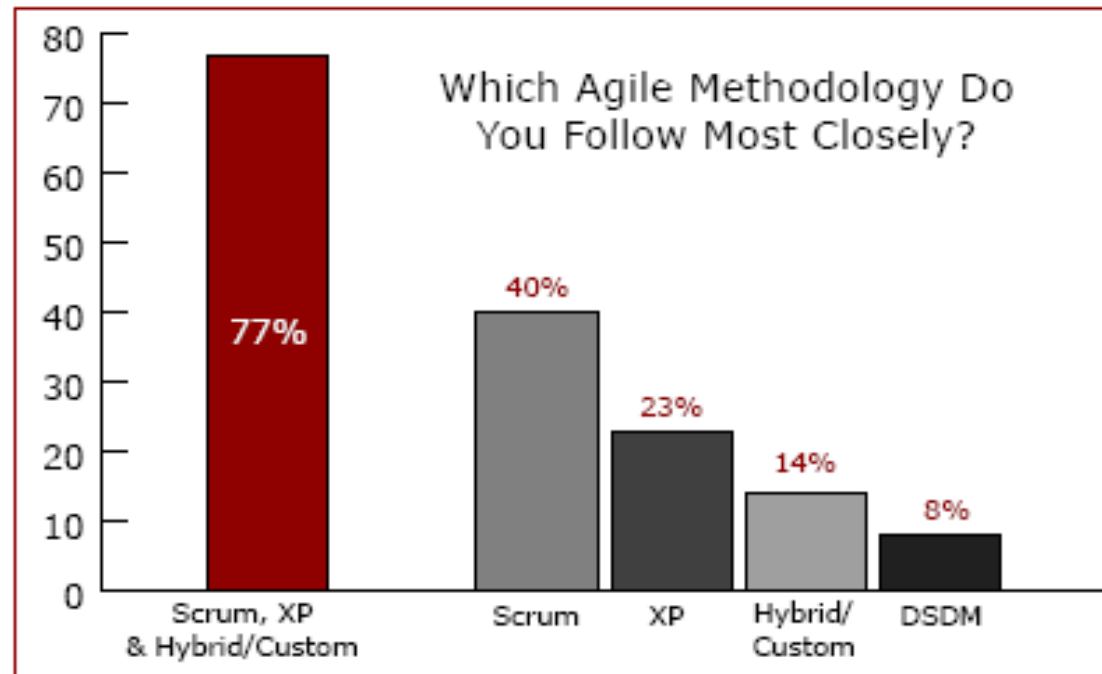
That is, while there is value in the items on the right, we value the items on the left more.

Done

Start | J.. h.. M. E.. C.. A. J.. ה. V.. M. C.. h.. EN | 07:59

Source: The State of Agile Development  
by VersionOne

---



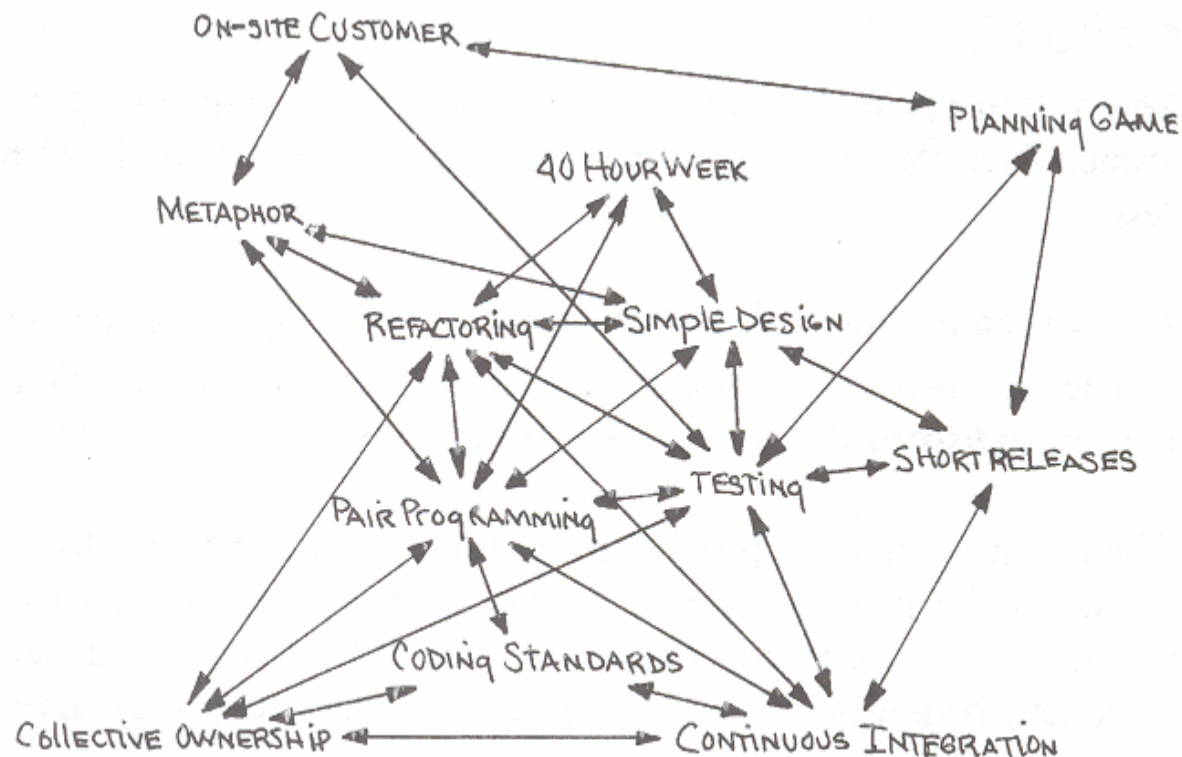


# Extreme Programming

---

- developed by software practitioners - Kent Beck
- values:
  - feedback, simplicity, communication, courage, respect
- practices
- why 'extreme'

# Extreme Programming practices



**Note:**  
nothing is new;  
gathering the  
practices  
together is XP  
uniqueness

FIGURE 4. The practices support each other

Beck, K. (2000). *eXtreme Programming explained*, Addison Wesley.



# Agile in Practice: Conceptual Changes

---

- Agile software development introduces new organizational culture:
  - Cooperation and knowledge sharing (*vs.* knowledge-is-power)
  - Simplicity (*vs.* habit-to-increase-complexity)
  - Change in work habits
  - Change in how the customers are conceived
  - Change in customers-developers relationships



# Summary

---

- The nature of software development
- Agile development
- Extreme Programming
- Cultural change