

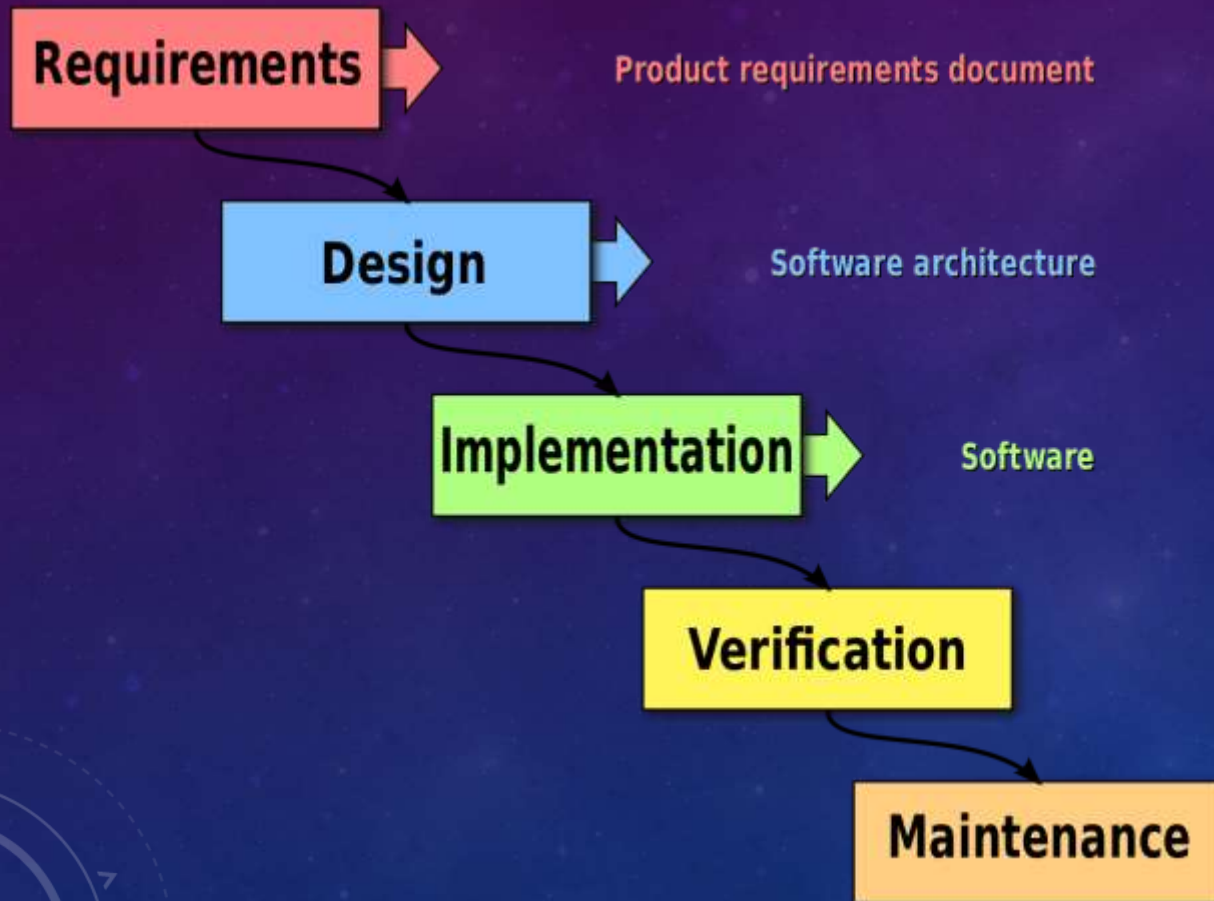
The background is a gradient of dark blue and purple, overlaid with faint, light-colored geometric patterns. These patterns include several concentric circles and arcs, some of which are marked with degree values (e.g., 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260). There are also dashed lines and arrows, suggesting a technical or scientific theme.

# INTRODUCTION TO EXTREME PROGRAMMING (XP)

BRYAN NEHL

@KØEMT

# THE DARK TIMES



# EXTREME PROGRAMMING (XP)



- The first Extreme Programming project was started **March 6, 1996**.
- **Kent Beck – Project Leader**
- Ron Jeffries
- Don Wells
- Martin Fowler
- Chet Hendrickson

# THE AGILE MANIFESTO

**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.**

from <http://agilemanifesto.org/>

Kent Beck  
Mike Beedle  
Arie van Bennekum  
Alistair Cockburn  
Ward Cunningham  
Martin Fowler  
James Grenning  
Jim Highsmith  
Andrew Hunt  
Ron Jeffries  
Jon Kern  
Brian Marick  
Robert C. Martin  
Steve Mellor  
Ken Schwaber  
Jeff Sutherland  
Dave Thomas

© 2001, the above authors<sup>[1]</sup> this declaration may be freely copied in any form, <sup>[2]</sup> but only in its entirety through this notice



# Agile and XP

The background is a dark blue gradient with faint, light blue circular gauges and arrows. One large gauge is in the top right corner, and another is in the bottom right corner. There are also smaller circular elements in the top left and bottom left corners. The overall aesthetic is technical and modern.

# EXTREME PROGRAMMING 1E VS 2E

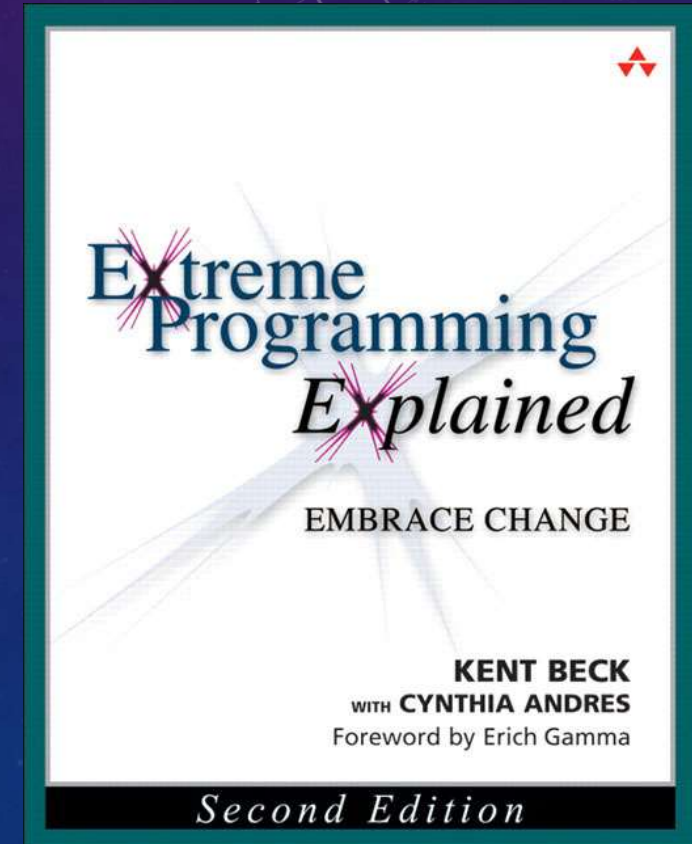
## 1e

- October 1999
- 12 practices
- Coding Standard
- Metaphor



## 2e

- November 2004
- the 5 values
- Principles
- Corollary practices



## VALUES: COMMUNICATION

- Maximize communication within team
- With customers
- Documents – easily readable & up to date
- Common Vocabulary

# VALUES: **SIMPLICITY**

- Simple but not simplistic
- YAGNI



# VALUES: **FEEDBACK**

- People
- Systems
- Team

# VALUES: RESPECT

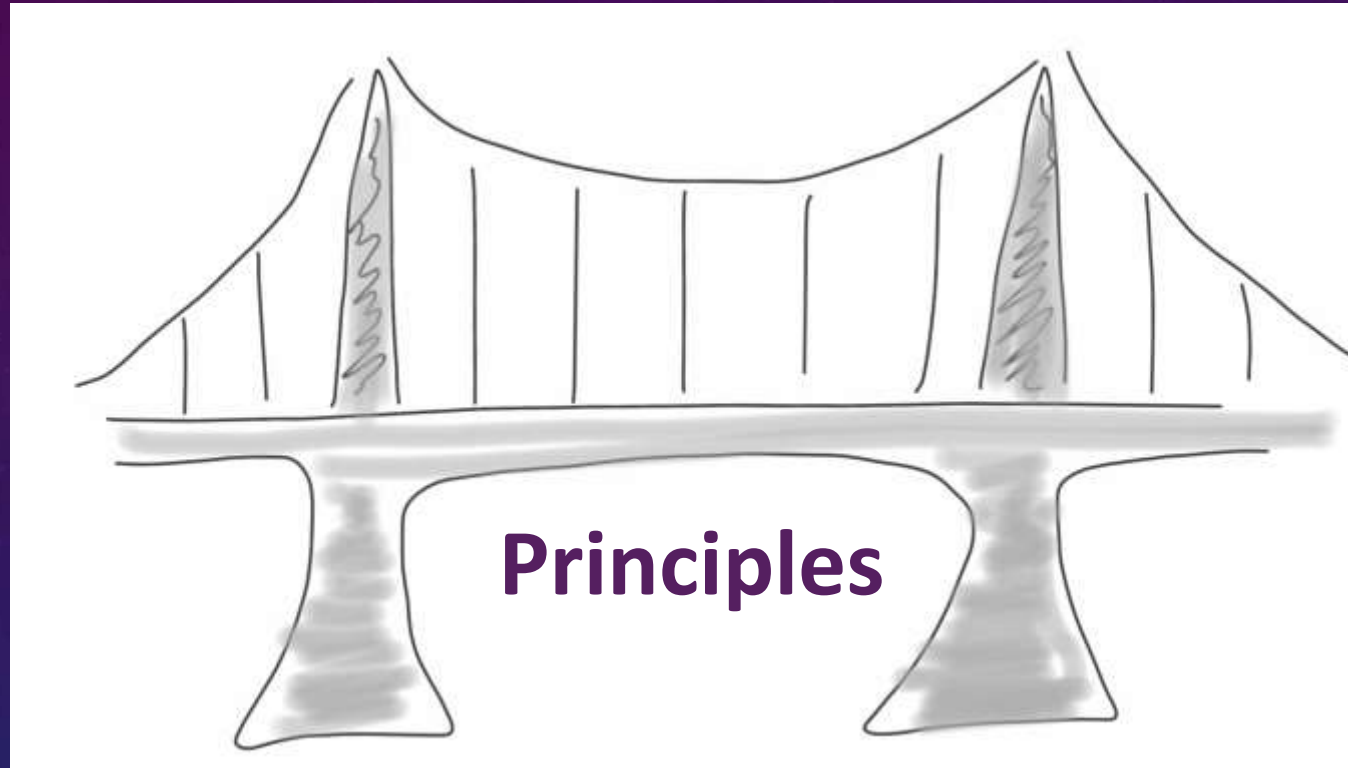
- Self
- Others
- Others work

# VALUES: COURAGE

- In combination with other values is powerful

# 14 PRINCIPLES

Values



Practices

- Humanity
- Economics
- Mutual Benefit
- Self-similarity
- Improvement
- Diversity
- Reflection
- Flow
- Opportunity
- Redundancy
- Failure
- Quality
- Baby Steps
- Accepted Responsibility



# TEAM AND HUMAN FACTORS PRACTICES

- Sit Together
- Whole Team
- Informative Workspace
- Energized Work – sustainable pace
- Pair Programming



# THE TEAM ROOM



<http://xp123.com/xplor/room-gallery/>





# PAIRING STATIONS





# PAIRING STATIONS

1B

1A

2A

2C



# PAIRING STATIONS



# REQUIREMENTS ANALYSIS AND PLANNING PRACTICES

- **Stories**
  - Cards
    - Role
    - Feature
    - Acceptance criteria
  - Sizing
  - Conversation



# DOCUMENTATION

"We're doing XP"

is ***not*** a reason to skip  
documentation.



# REQUIREMENTS ANALYSIS AND PLANNING PRACTICES

- Weekly Cycle
  - Yesterday's weather
- Quarterly Cycle
  - Theme
  - Process improvement
  - Big picture
- Slack (the concept)
- Flow based management

# DESIGN PRACTICES

- Incremental Design
  - As you code
  - EDUF

# DESIGN PRACTICES

- **Test-First Programming**
  - What is it?
    - Simple test
    - Simple code (DTSTTCPW)
    - Refactor
    - repeat
  - Arrange-Act-Assert
  - TETTCPB



**RED – GREEN – REFACTOR**

# DESIGN PRACTICES

## Test-First Programming addresses:

- Scope creep / cowboy coding
- Coupling and cohesion -- testability
- Trust (the code & tests)
- Rhythm – focus
  - Red-Green-Refactor



# CODING AND RELEASE PRACTICES

- Continuous Integration
  - Git
- Ten Minute Build
- Continuous Deployment
  - Automation Servers



**Jenkins**

# ADOPTING XP: WHY?

- Have Production/Process critical applications?
- Knowledge silos?
- Want reduced defect rates?
- Want deployed/released software?
- Changing Requirements?
- Customer Satisfaction?
- Developer Satisfaction?

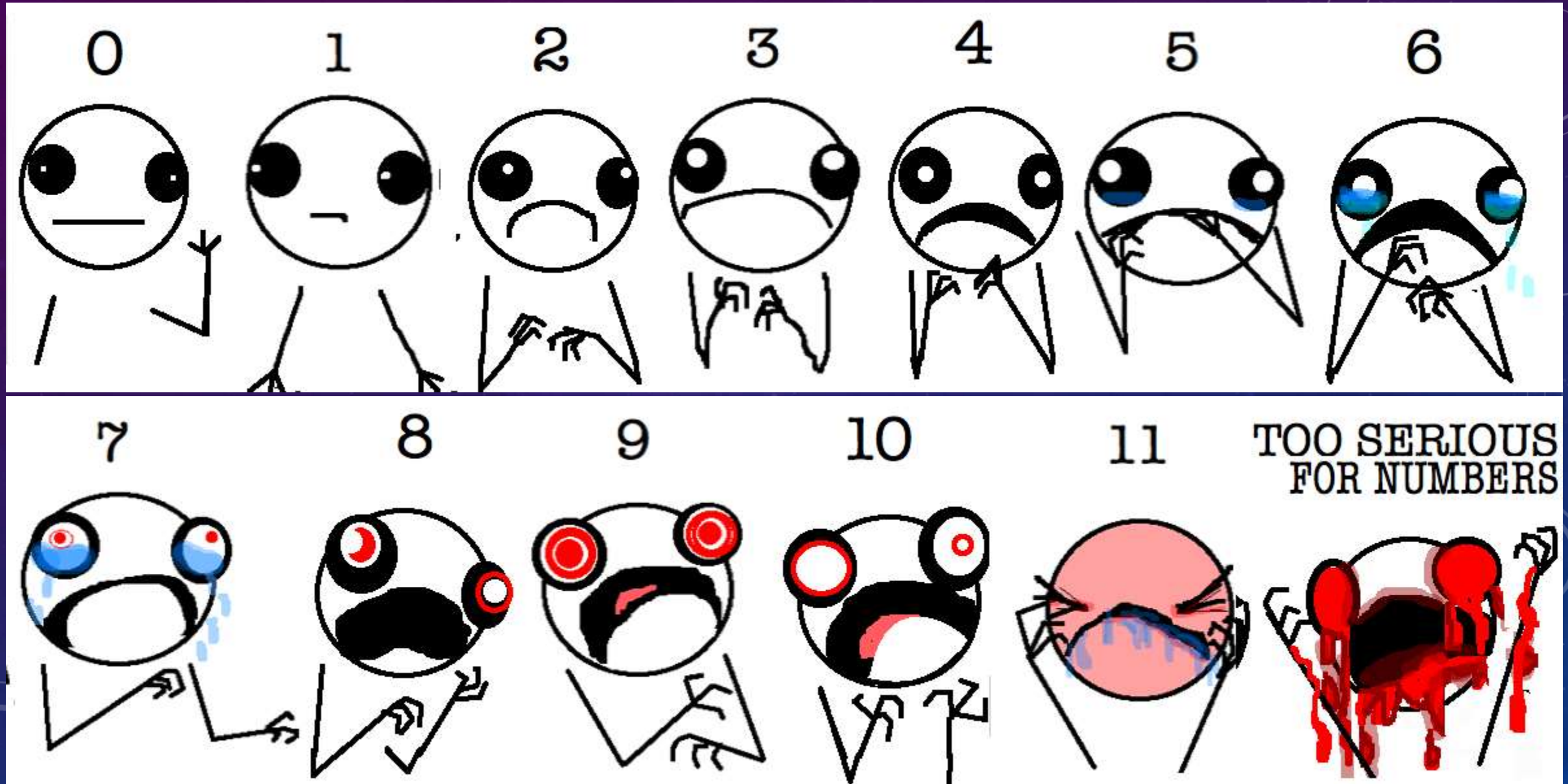


# ADOPTING XP: HOW?





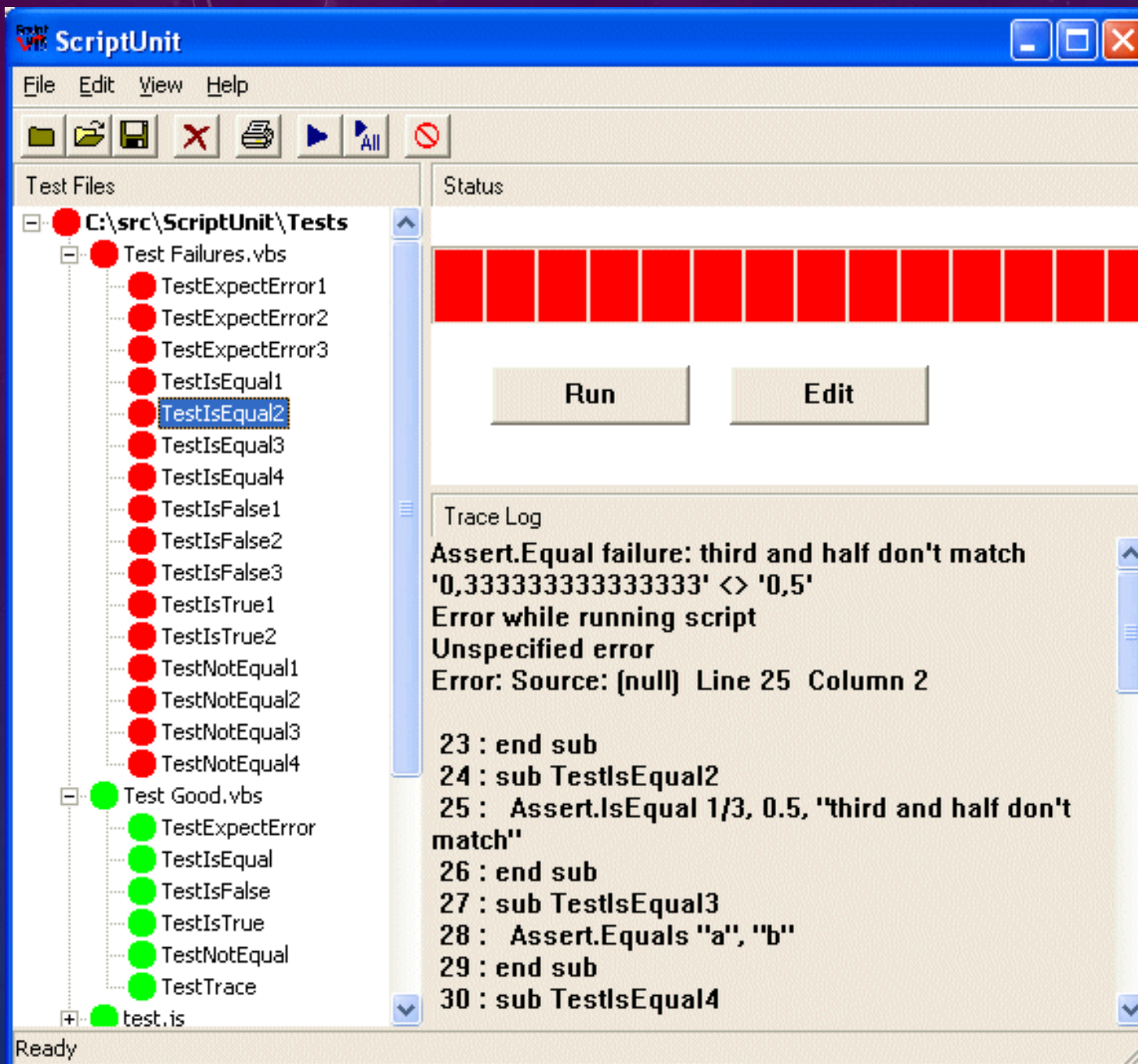
# ADOPTING XP: HOW?

















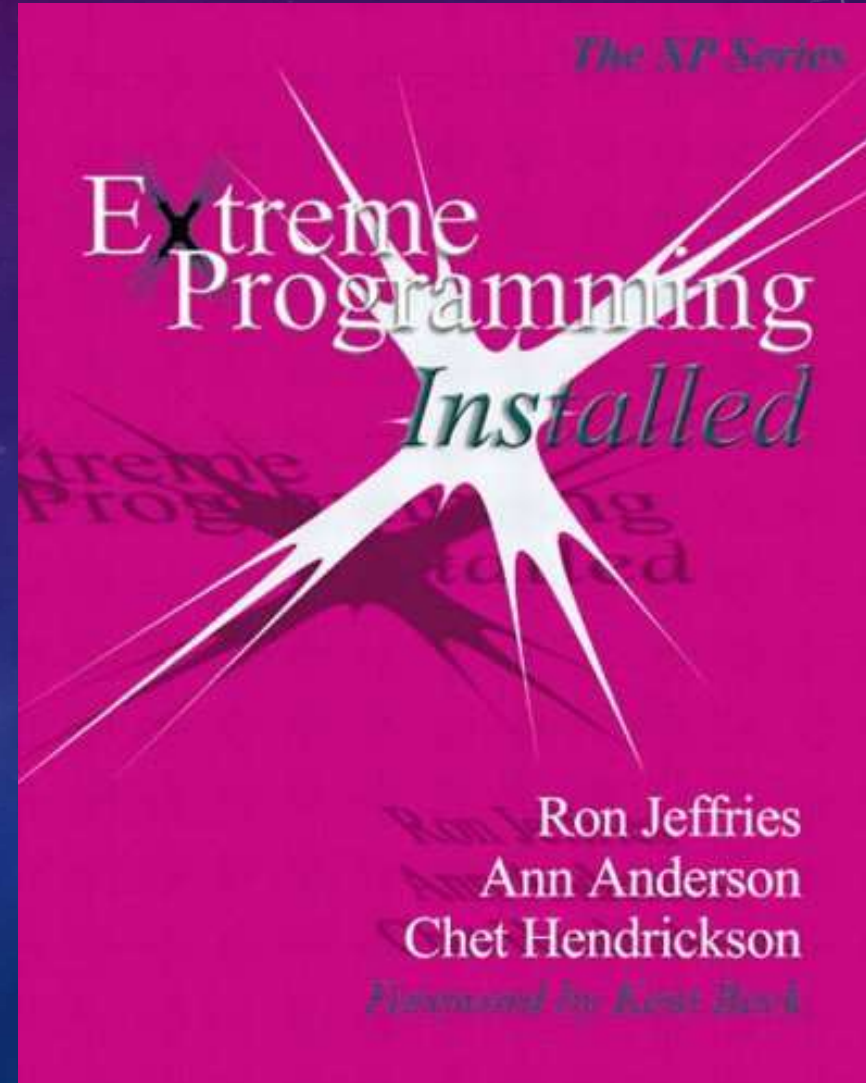
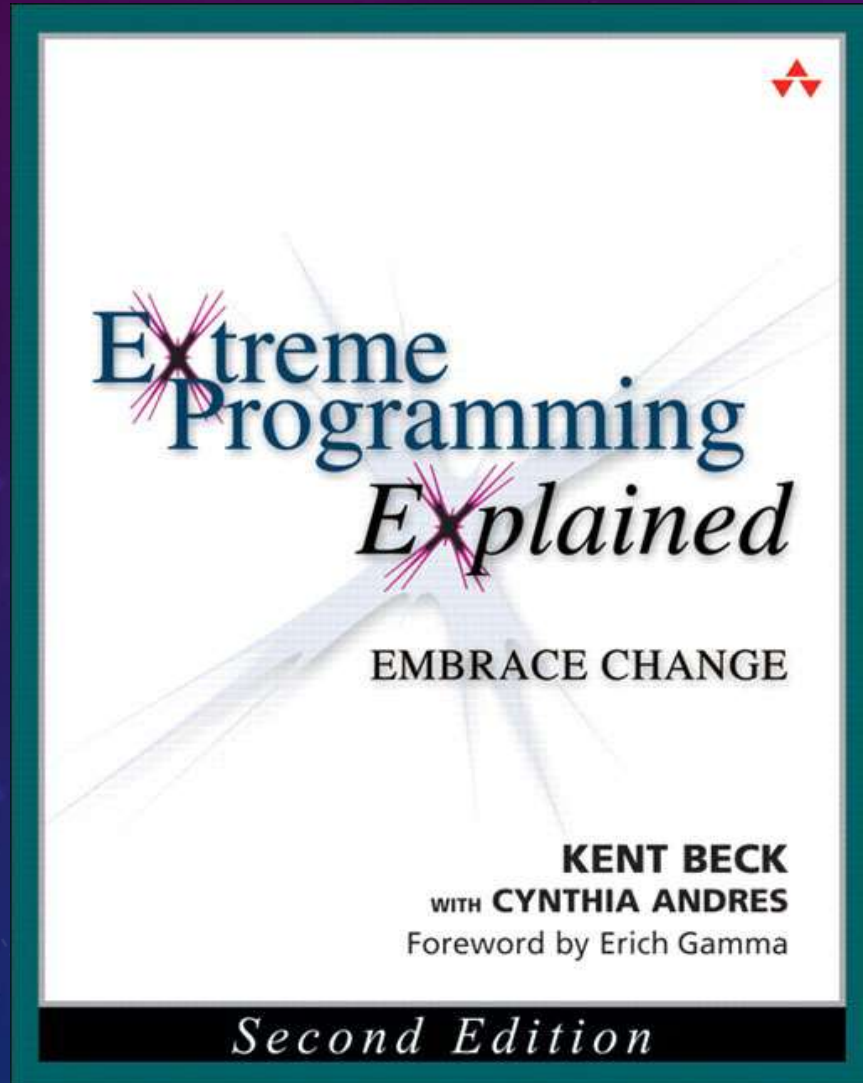








# WHERE DO I GO FROM HERE?





## WEB RESOURCES

1. <http://www.extremeprogramming.org/>
2. <http://www.extremeprogramming.org/more.html>
3. <http://c2.com/cgi/wiki?ExtremeProgramming>
4. <http://ronjeffries.com/xprog/what-is-extreme-programming/>
5. <http://xp123.com/xplor/>
6. <http://www.xp123.com/xplor/xp0202/xp-one-page.PDF>



# finished



**Bryan Nehl** – **@k0emt** – dbBear.com

<https://github.com/k0emt/Presentations>