

Distributed Extreme Programming: Introduction

- ◆ Distributed project teams
- ◆ Environmental constraints
- ◆ Personnel constraints

Distributed Extreme Programming: The Planning Game

- ◆ Matches well to engagement-based projects
- ◆ Ensures involvement and commitment of stakeholders
- ◆ Facilitates smaller (read: cheaper) proposals and contracts

Distributed Extreme Programming: Small Releases

- ◆ Matches well to deliverables-based contracts
- ◆ Establishes credibility of consultants
- ◆ Building enthusiasm builds commitment (and ongoing contracts)

Distributed Extreme Programming: Metaphor

- ◆ Matches well with client desire for low overhead
- ◆ Facilitates scope redefinition and contract renegotiation

Distributed Extreme Programming: Simple Design

- ◆ Matches client perspective and desire for low overhead
- ◆ Facilitates distributing workload

Distributed Extreme Programming: Testing

- ◆ Very difficult to implement true XP testing in virtual development environment
- ◆ All the usual issues
- ◆ Requires higher level of project management

Distributed Extreme Programming: Refactoring

- ◆ Even more difficult to implement than testing
- ◆ Hard to demonstrate cost benefit to client
- ◆ One solution: build contingency into bid or rates and sell quality

Distributed Extreme Programming: Pair Programming

- ◆ Impossible to implement
- ◆ What is the underlying goal?
- ◆ Implement practices towards the goal
 - Frequent reviews
 - Rotate assignments
 - Pair when possible

Distributed Extreme Programming: Collective Ownership

- ◆ All the same issues...

Distributed Extreme Programming: Continuous Integration

- ◆ All the same issues...

Distributed Extreme Programming: 40-Hour Week

- ◆ All the same issues...

Distributed Extreme Programming: On-Site Customer

- ◆ All the same issues, plus...
- ◆ Clients often believe the consultant should do it all: that's what they're paying for
- ◆ Important to contract up front

Distributed Extreme Programming: Coding Standards

◆ Of course!