Can RUP Be Agile?
Can RUP Be Extreme?

Orange County Rational Users' Group

January 20, 2005

By Paul Hodgetts, Agile Logic www.AgileLogic.com

Introductions

- Solutions for Delivering Your Projects:
  - Agile Process Adoption Solutions
  - Coaching, Consulting, Mentoring Services
  - Training in Agile Processes, Software Development and Enterprise Technologies
  - Turn-Key Software Development Projects
- Fullerton, CA, based
- Founded 2001 by industry veterans
- Contact info: www.agilelogic.com (866) 64-AGILE

Paul Hodgetts

- Team coach, trainer, consultant, developer
- Founder and CEO of Agile Logic
- 22 years overall, 5 years agile experience
- Certified ScrumMaster Trainer
- Innovator in Agile business and project management
- Author (Extreme Programming Perspectives)
- Presenter at conferences (ADC, XPAU, JavaOne)
- Agile Alliance Program Director
- Member of CSUF agile advisory board
- Contact info: phodgetts@agilelogic.com

Agile Logic
Agility That Delivers.

Agenda

- Development Processes
- Process Attributes
- Process Spectrum
- Unified Process
- Agile Processes
- Process Contexts
- UP/RUP and Agility

Agile Logic
Agility That Delivers.

Development Process

- Common understanding of "how we do things around here"
- Process can provide:
  - Guidance on what to do, when, by who
  - A framework for coordination
  - Instrumentation points
  - Guidance on sufficiency and completeness

Agile Logic
Agility That Delivers.

Process Improvement

- *Looking for better ways to do things*
- *Not "doing a process" for its own sake*
- *Increasing our capability to deliver software*
- *Adoption strategies – incremental to wholesale*

Agile Logic
Agility That Delivers.

Agenda

- Development Processes
- Process Attributes
- Process Spectrum
- Unified Process
- Agile Processes
- Process Contexts
- UP/RUP and Agility

Agile Logic
Agility That Delivers.

Process Attributes

- What kinds of things can we look at to better understand and discuss processes?

Agile Logic
Agility That Delivers.

Activity Sequencing

- Phased Approach
  - Gathers similar activity types together
  - Preference towards serial completion
  - Ultimate in phased approach is waterfall
- Concurrent and Parallel
  - Activities occur opportunistically
  - Activities of all types happening at same time
  - Partial completion considered the norm

Agile Logic
Agility That Delivers.

Delivery Strategy

- Defined by degree of iteration and increments
- Iterative
  - Repeatedly executing a process cycle
  - Iterations provide synchronizing points
  - Iterations provide feedback points
- Incremental
  - System is built in progressive stages
  - Iterations add features and refinements
  - Each increment has a degree of completeness

Agile Logic
Agility That Delivers.

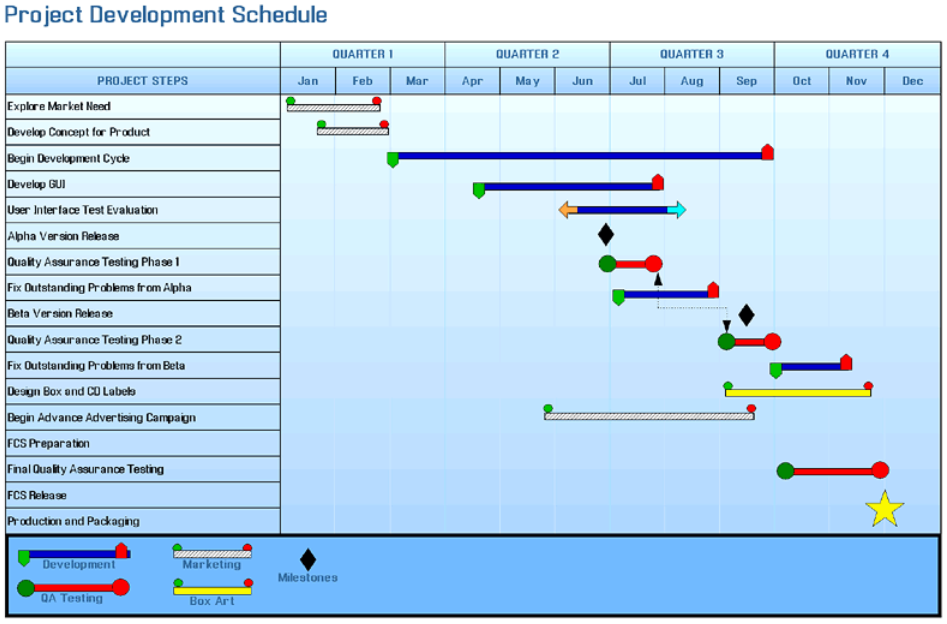
People Strategies

- Collaborative vs. Heroic
- Individual vs. Collective
- Decision Structure:
  - Hierarchical, “command and control”
  - Flattened, local empowerment
- Assigned vs. Accepted Accountability
- Degree and Range of Responsibility
- Fixed vs. Flexible Roles

Agile Logic
Agility That Delivers.

Predictive Planning and Control

- *Predict and plan expected activities*
- *Management by controlling activities per plan*
- *Change is minimized and managed via change control*



Agile Logic
Agility That Delivers.

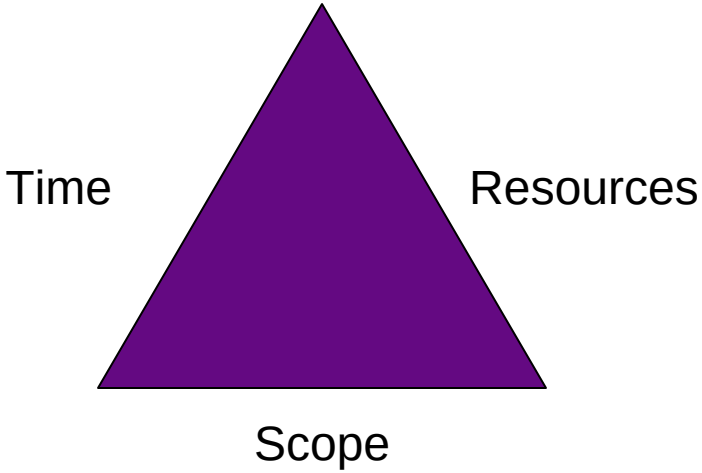
Adaptive Planning and Empirical Control

- Prioritized set of deliverables form the plan
- Opportunistic execution of activities to create deliverables
- Management via feedback and adaptation
- Empirical process control
  - *Visibility*
  - *Inspection*
  - *Adaptation*

Agile Logic
Agility That Delivers.

Project Balancing

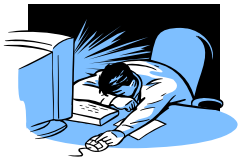
- Resources
- Time
- Scope
- Must be in balance for a healthy project



Agile Logic
Agility That Delivers.

The Resource Variable

- **Staffing is usually the least effective variable to adjust.**
  - Staffing increases have long lead times.
  - Increased intensity has diminishing returns.
  - Team culture requires some degree of stability.



- **Tools and technology can provide benefits.**
  - Effective tools provide continuing benefits.
  - Front-end costs need to be carefully amortized.
  - The wrong tools and technology increase friction.

Agile Logic
Agility That Delivers.

The Time Variable

- Can be the most painful variable to adjust
  - Early commitments are usually date-based.
  - Target dates are often the most important objective.
  - There's only so many hours in a day, and they pass by regardless of how we use them.

Agile Logic
Agility That Delivers.

The Scope Variable

- Can be the most effective variable to adjust
  - Can adjust scope breadth – what's included.
  - Can adjust scope depth – refinement.
  - Partial scope can often generate immediate returns.
  - It is often preferable to reach a date with partial scope completely finished, rather than complete scope partially finished.

Agile Logic
Agility That Delivers.

Prescription

- Prescriptive is like a cookbook:
  - What to do
  - When to do it
  - How to do it
- Creative
  - Local decisions
  - Context determines activities

Agile Logic
Agility That Delivers.

Formality and Ceremony

- Formality specifies:
  - Types and forms of work products
  - Procedures for activities
- Ceremony specifies:
  - Level of activity surrounding events
  - Degree of audit trails
  - Types and forms of communications

Agile Logic
Agility That Delivers.

Rigor

- Rigor is the precision and completeness in the execution
- Rigor is not a process attribute
- Rigor is about the way the team approaches and executes their process
- A process may provide guidance and practices to encourage rigor

Agile Logic
Agility That Delivers.

Discipline

- Discipline is about conscientiousness, courage, motivation, "doing the right thing"
- Discipline is not an attribute of a process
- Discipline is about the way the team approaches and executes their process
- A process may encourage discipline by the provided activities and criteria

Agile Logic
Agility That Delivers.

Agenda

- Development Processes
- Process Attributes
- Process Spectrum
- Unified Process
- Agile Processes
- Process Contexts
- UP/RUP and Agility

Agile Logic
Agility That Delivers.

Chaotic Processes

- *Minimal shared process*
- *Code and fix*
- *Short term decisions*
- *Can sprint very fast*
- *Does not scale*
- *Increasing debt*
  - *Quality, design, integration, knowledge*

Agile Logic
Agility That Delivers.

Bureaucratic Processes

- Targeted for all contexts
- Large and complex
- Mandated activities
- Comprehensive framework
- High overhead
- Long release cycles
- Inability to keep up with business needs

Agile Logic
Agility That Delivers.

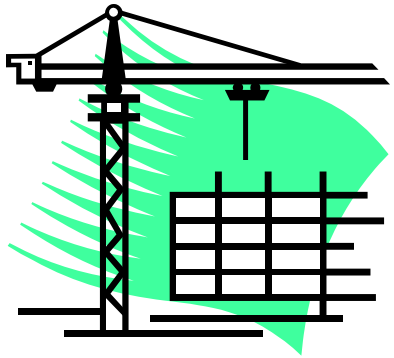
Options for Process Improvement

- *"Heroic" Approach*
  - Relies heavily on individual effort
  - Difficult to plan, results unreliable
  - High risk of failure
  - Heavy human cost

Agile Logic
Agility That Delivers.

Options for Process Improvement

- *"Formal" Methodologies*
  - Detailed, bureaucratic process
  - Engineering/construction-style planning – predictive of activities
  - Expensive, time-consuming to implement
  - Limited success, not popular with teams



Agile Logic
Agility That Delivers.

Options for Process Improvement

- *"Agile" Methodologies*
  - Just enough process
  - Adaptive rather than predictive
  - People-oriented focus to the process
  - Faster and less-costly to implement

Agile Logic
Agility That Delivers.

Agenda

- Development Processes
- Process Attributes
- Process Spectrum
- Unified Process
- Agile Processes
- Process Contexts
- UP/RUP and Agility

Agile Logic
Agility That Delivers.

UP Background

- Objectory Process (1987-1995)
  - Ivar Jacobsen at Ericsson
- Rational Objectory Process (1996-1997)
  - Rational approach (Philippe Krutchen)
  - UML
- Rational Unified Process
  - Grady Booch, Jim Rumbaugh, other sources
  - Rational tool set

Agile Logic
Agility That Delivers.

Characteristics of the UP

- A process framework, not a specific process
- Refined into a specific process instantiation
  - Tailored to a Development Case
- Very broad coverage
- All activities and work products are optional
- Encourages minimal development case

Agile Logic
Agility That Delivers.

UP Values

- Use-case driven
- Architecture-centric
- Iterative and incremental
- Attack risk early
- Deliver executable architectures (systems)
- Provoke and accommodate change early
- Baseline architecture early
- Prefer component-based designs

Agile Logic
Agility That Delivers.

UP Roles

- Large collection of roles
- Roles organized around disciplines
- Roles further specialized
- UP encourages cross-functional teams

Agile Logic
Agility That Delivers.

UP Work Products

- ~50 non-software work products
  - Vision, risk list, iteration plan, use case model, design model
- All are optional, some are recommended
- Work products are information abstractions
- Organized within disciplines
  - Requirements, design, project management

Agile Logic
Agility That Delivers.

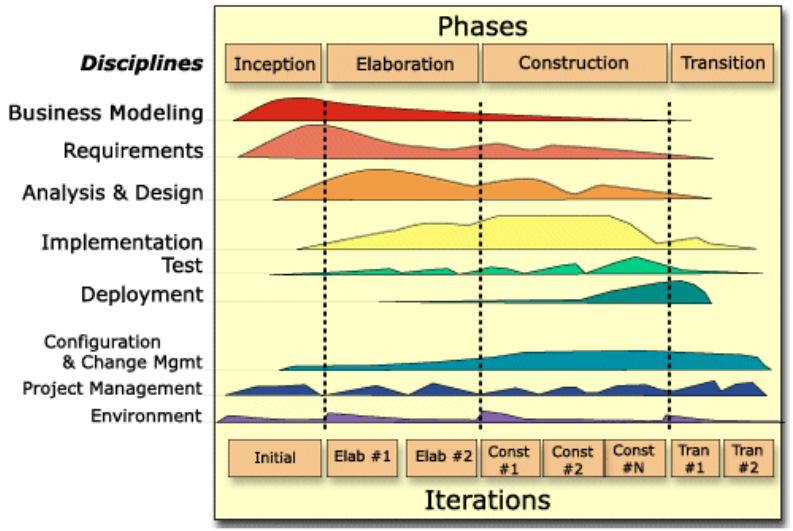
UP Activities and Workflows

- Large collection of activities to support the creation of artifacts
- Guidance provided for each activity
- Strong assignment of roles to activities
- Activities and workers grouped into workflows

Agile Logic
Agility That Delivers.

UP Activity Sequencing

- Iterative and incremental
- Iterations classified into "phases"
  - Inception, elaboration, construction, transition
- Milestone objectives define boundaries



Agile Logic
Agility That Delivers.

UP Characteristics

- Sequencing and delivery is iterative and incremental, with some phasing
- Encourages collaboration, but pretty strong individual role assignments
- Activities suggest predictive planning
- No preference for balancing strategy
- Very prescriptive, although lots of options
- Formality and ceremony optional

Agile Logic
Agility That Delivers.

UP vs. RUP

- The Unified Process
  - Broad framework
  - Many optional activities and work products
  - Tool support optional and unspecified
- The Rational Unified Process
  - A specialization of the UP, still a framework
  - A licensed product
  - Templates for work products
  - Tailored to Rational toolset

Agile Logic
Agility That Delivers.

Agenda

- Development Processes
- Process Attributes
- Process Spectrum
- Unified Process
- Agile Processes
- Process Contexts
- UP/RUP and Agility

Agile Logic
Agility That Delivers.

What Exactly Is an "Agile" Process?

- *Focus on adaptability and responsiveness*
- *Built around core strategies:*
  - Iterative and Incremental Development (IID)
  - Adaptive project management
  - Collaborative, "whole team" approach
  - Common shared vision and goals
- *Constructed from "best practices":*
  - Emphasis on simplicity, lightness, communication, self-directed teams, quality and technical excellence

Agile Logic
Agility That Delivers.

The World of Agile Processes

- *Extreme Programming (XP)*
- *Feature-Driven Development (FDD)*
- *Scrum*
- *DSDM (Dynamic System Development Method)*
- *Crystal Family of Processes, e.g. Crystal Clear*
- *Lean Software Development*
- *Adaptive Software Development (ASD)*
- Others: *MSF Agile, Agile UP/RUP, Evo, Win-Win Spiral*

Agile Logic
Agility That Delivers.

The Agile Alliance

- 2001 – representatives from agile processes meet in Snowbird, Utah.
- Agreed on a "manifesto" of values and principles:
  - 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."

Agile Logic
Agility That Delivers.

History of Extreme Programming

- Early Influences
  - Incremental, stakeholder-driven design process from Alexander
  - Programming as learning from Papert, Kay
- Kent Beck & Ward Cunningham
  - Mid-80s – Pair programming at Tektronix
  - 80s, 90s – Smalltalk culture produces refactoring, continuous integration, constant testing, close customer involvement
  - Generalized to other environments
  - Early 90s – Core values developed within patterns community, Hillside Group

Agile Logic
Agility That Delivers.

History of Extreme Programming

- 1995 – Kent summarizes in Smalltalk Best Practices
- 1996 – Ward summarizes in Episodes
- 1996 – Kent adds unit testing, metaphor at Hewitt
- 1996 – Kent takes on Chrysler C3 project
- C3 adds Ron Jeffries as coach
- Practices refined on C3, published on Wiki

Agile Logic
Agility That Delivers.

History of Extreme Programming

- Scrum practices incorporated and adapted as planning game
- 1999 – Extreme Programming Explained
- 1999 – Fowler publishes Refactoring
- 1999 – XP Immersion held, e-group formed
- 2000 – more books, first conferences
- Evolution continues through today
- 2004 Kent Beck releases EPE 2nd Edition

Agile Logic
Agility That Delivers.

What Is Extreme Programming?

- XP is a specific instantiation of an agile process
- XP combines best practices in a different way
- XP is a different approach to development
- XP provides a core process model
- XP is not intended to be a complete framework

Agile Logic
Agility That Delivers.

Emergence

- XP provides values and principles to guide team behavior
- Team expected to self-organize
- XP provides specific core practices
- Each practice is simple and self-complete
- Combination of practices produces more complex emergent behavior
- Synergy of practices still not fully understood

Agile Logic
Agility That Delivers.

Why Is It Called "Extreme?"

- Selected the minimal set of effective practices
- "Turned the knob up to 10" on each practice
  - Very short cycles (planning game)
  - Continuous code reviews (pair programming)
  - Extensive testing (unit testing, acceptance testing)
  - Continuous integration
  - Constant design improvement (refactoring)
  - Continuous architecture refinement (metaphor)
  - Etc…

Agile Logic
Agility That Delivers.

XP Values

- Communication
- Simplicity
- Feedback
- Courage

Agile Logic
Agility That Delivers.

XP Principles

- Rapid Feedback
- Assume Simplicity
- Incremental Change
- Embracing Change
- Quality Work
- Teach Learning
- Small Initial Investment
- Play to Win
- Concrete Experiments
- Open Honest Communication
- Work With Instincts
- Accepted Responsibility
- Local Adaptation
- Travel Light
- Honest Measurement

Agile Logic
Agility That Delivers.

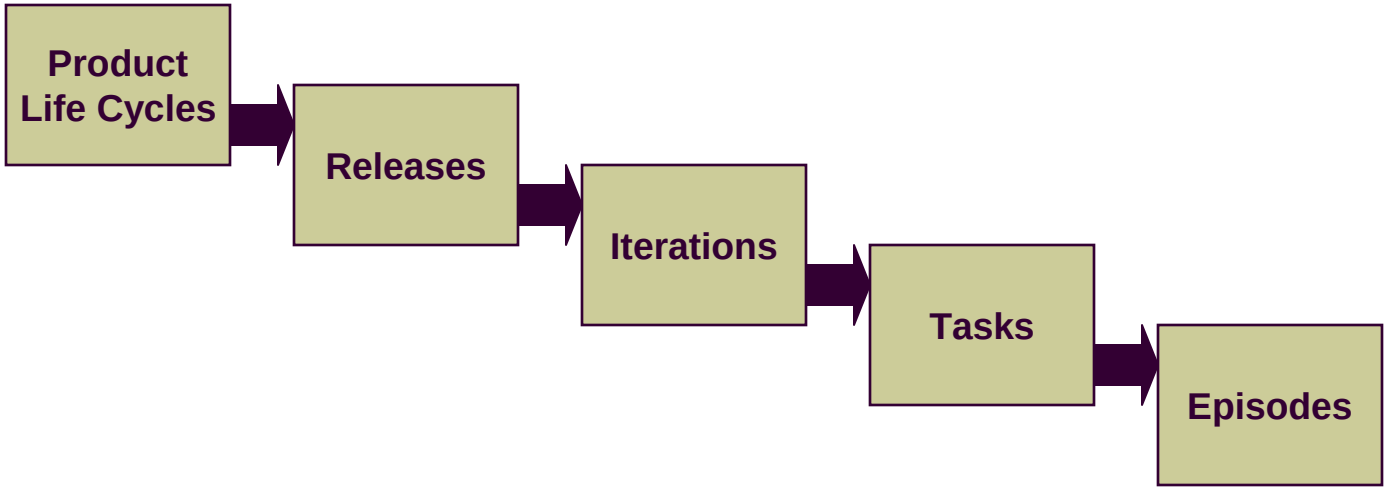
XP Project Community

- Emphasis on the "Whole Team"
- Collaboration and colocation
- Three general roles
  - Customer
  - Developer
  - Manager
- Roles define areas of accountability
- Specific job functions neither specified nor excluded (e.g., QA, PM, operations, etc.)

Agile Logic
Agility That Delivers.

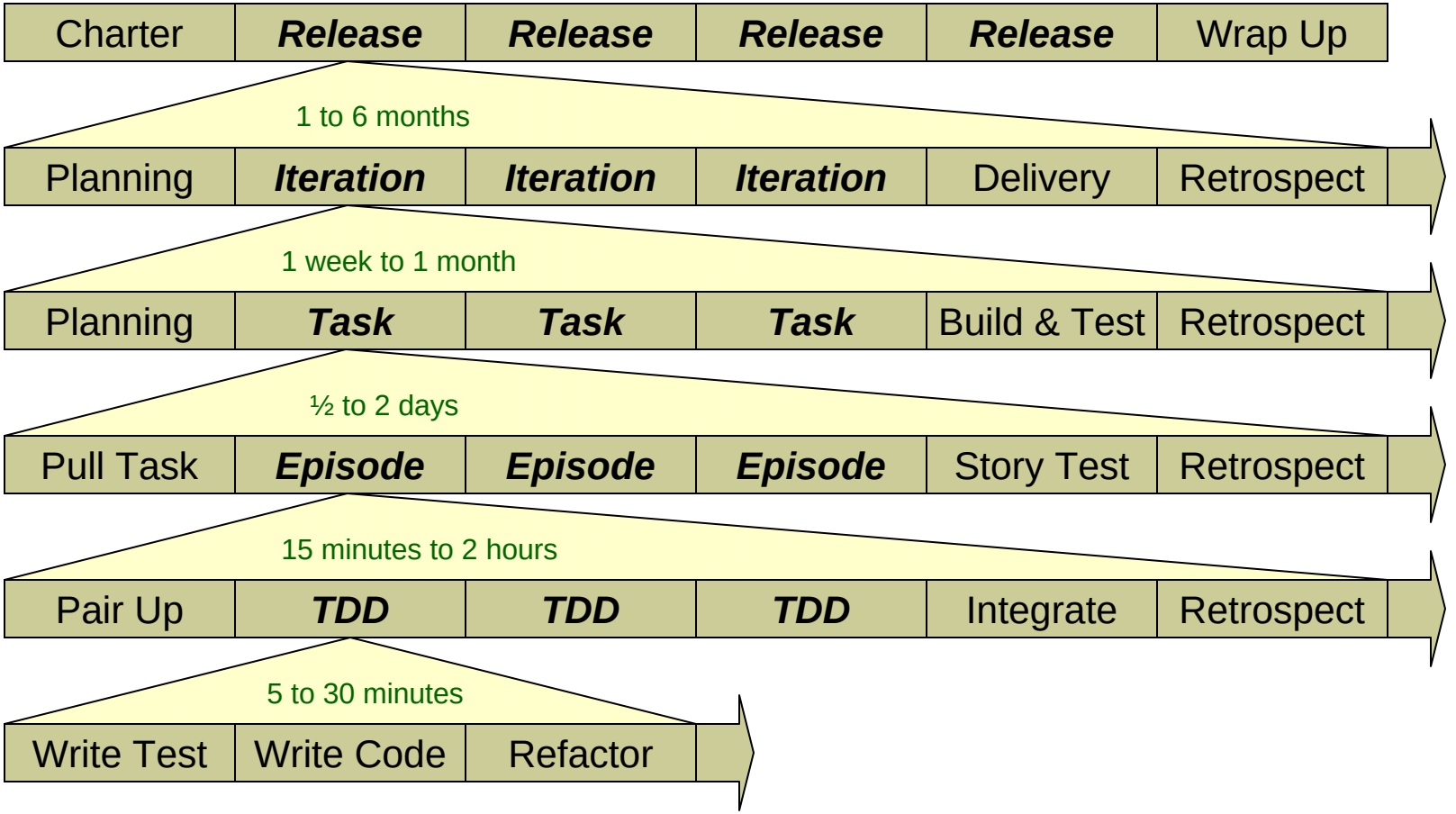
XP Process Cycle

- XP is iterative and incremental
- XP is driven by time-boxed cycles
- The rhythm of the XP process is crucial



Agile Logic
Agility That Delivers.

XP Process Cycle



Agile Logic
Agility That Delivers.

The Original 12 XP Practices

- On-Site Customer
- The Planning Game
- Small Releases
- Testing
- Simple Design
- Pair Programming
- Refactoring
- Continuous Integration
- Collective Ownership
- Coding Standards
- Metaphor
- 40-Hour Week

Agile Logic
Agility That Delivers.

Evolving Practices

- On-Site Customer
  - Whole Team
- The Planning Game
  - Release Planning
  - Iteration Planning
- Testing
  - Acceptance Testing
  - Unit Testing
  - Test-Driven Development
- Refactoring
  - Design Improvement
- 40-Hour Week
  - Sustainable Pace

Agile Logic
Agility That Delivers.

Additional Practices

- Stand-Up Meetings
- Tracking & Metrics
- Retrospectives
- Big Visible Charts
- Team Culture
- Consensus
- Skunk Works, War Room
- Version & Configuration Management, Automated Builds, Build Promotion

Agile Logic
Agility That Delivers.

XP Characteristics

- XP is iterative and incremental, there are no phases
- Each iteration intended to be shippable
- XP is highly collaborative
  - Collective control and ownership
  - Self-organizing teams
  - Basic role structure, assumes flexibility
- XP utilizes adaptive planning
- Preferred balancing strategy is via scope
- XP is creative overall, prescriptive at the practice level
- XP discourages unnecessary formality and ceremony

Agile Logic
Agility That Delivers.

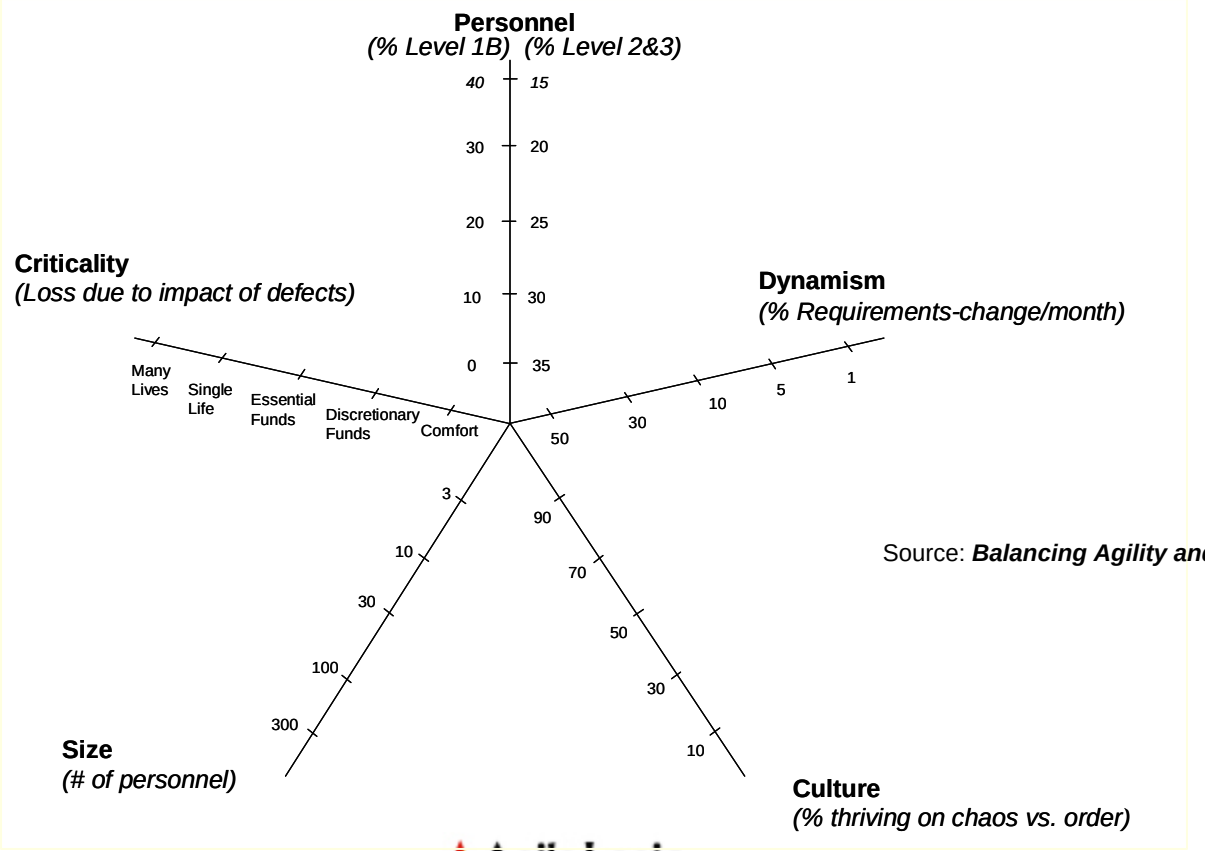
Agenda

- Development Processes
- Process Attributes
- Process Spectrum
- Unified Process
- Agile Processes
- Process Contexts
- UP/RUP and Agility

Agile Logic
Agility That Delivers.

Process Context

- Factors to consider when choosing process



Source: ***Balancing Agility and Discipline***, Boehm & Turner

Agile Logic
Agility That Delivers.

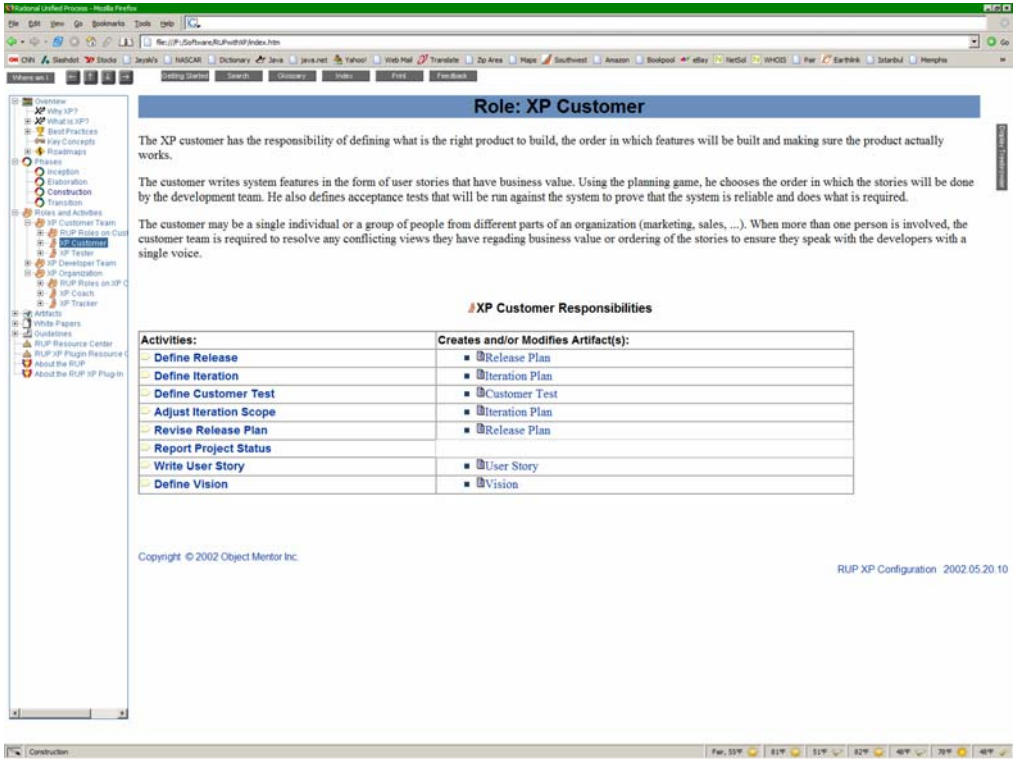
Agenda

- Development Processes
- Process Attributes
- Process Spectrum
- Unified Process
- Agile Processes
- Process Contexts
- UP/RUP and Agility

Agile Logic
Agility That Delivers.

XP and RUP

- XP plug-in for RUP from IBM/Rational
- Available from http://www-106.ibm.com/developerworks/rational/library/4156.html



Agile Logic
Agility That Delivers.

Can RUP Be Agile?

- RUP is a flexible framework
- RUP is often over-implemented
  - Subtractive vs. additive process design
- Where RUP pushes against agility
  - Prescriptive nature of the framework
  - Tendency towards predictive planning
- RUP can and has been implemented in an agile way

Agile Logic
Agility That Delivers.

References and Resources

- *Extreme Programming Explained (2nd edition)*
  - By Kent Beck
- *Software Development for Small Teams*
  - By Gary Pollice, et al
- *Balancing Agility and Discipline*
  - *By Barry Boehm & Richard Turner*
- *Lots and lots of other XP and RUP books*
- Ron Jeffries's XP Site
  - www.xprogramming.com
- IBM Rational's RUP Site
  - www-306.ibm.com/software/awdtools/rup/index.html
- XP Discussion List
  - groups.yahoo.com/group/extremeprogramming/
  - (Lots of other great Yahoo! groups.)
- The Agile Alliance Site
  - www.AgileAlliance.org
- Agile Logic's Resources Site
  - www.AgileLogic.com/resources.html
- So. Cal. Agile / XP User Group
  - groups.yahoo.com/group/xpsocal/

Agile Logic
Agility That Delivers.