

# Product and Release Planning Practices for Extreme Programming

**Gert van Valkenhoef** (1,2), Tommi Tervonen (1)  
Bert de Brock (1), Douwe Postmus (2)

- (1) Faculty of Economics and Business, University of Groningen (NL)  
(2) Department of Epidemiology, University Medical Center Groningen (NL)

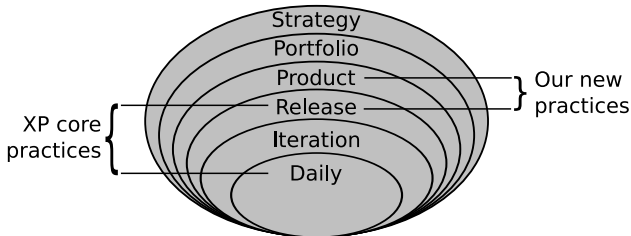
11th Int. Conf. on Agile Software Development (XP2010)  
Trondheim (Norway), 3 June 2010

# Customer Problems in XP

XP is nice for developers, but not the customer:

- **Lack of management context (1):**
  - XP does not address product planning or long-term goals
  - Lack of practices for the customer role
- **User story overload (2):**
  - Customer needs to consider too many alternatives
- **Prioritization stress (3):**
  - Responsibility of prioritizing stressful
  - Consequences of plan difficult to oversee
  - Business value of constantly managing priorities?

# New Practices



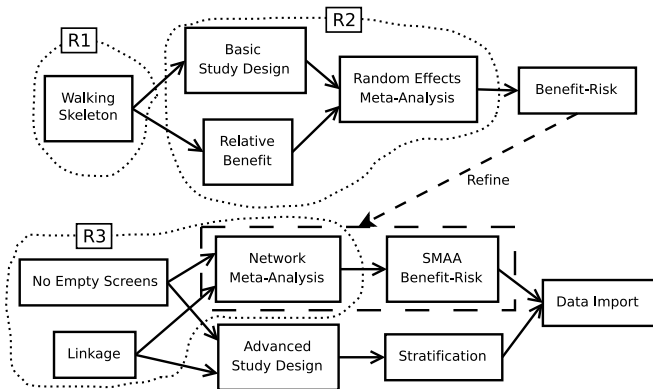
To address the problems, we propose two new practices

- Rolling Forecast for Product Planning
  - Addresses problem (1)
- Assisted Release Planning
  - Addresses problems (2) and (3)

# Practice 1: Rolling Forecast

- Replaces failed 'system metaphor' for expectation management
- Product goals:
  - By customer and manager, at inception
  - Functional, but vague; not requirements, but vision
  - Should (as everything) be reconsidered periodically
- Theme forecast:
  - By customer, manager and developer(s), every release
  - A set of themes and their likely order
  - Prediction: themes for the next 2-3 releases
  - The basis for story elicitation and release planning
- Evaluating old forecasts and updating new ones creates a *rolling forecast*

# Rolling Forecast Example



## Practice 2: Assisted Release Planning

We assist release planning with an optimization model:

- Stories, complexity, business value and themes elicited
- Model generates suggested release plan
- Optimal business value given velocity
- When velocity estimate changes dramatically, same data can be used to re-plan
- We implemented the model and made a prototype story tracking system for use in our own project

# Release Planning Model

Model optimizes business value given budget:

- For every story:
  - Story complexity: {1, 2, 3, 5, 8}
  - Business value: {1, 2, 3, 4, 5}
  - (Precedence constraints:  $x$  should be completed before  $y$ )
- Stories can be grouped into themes – synergy effects
- We fix a budget for the release (estimated velocity)
- Implemented as knapsack problem, solved by integer programming

# Conclusions

- XP lacks practices for the customer and product planning
- **Rolling forecast** helps manage expectations, provides context for user stories
- **Assisted release planning** reduces customer workload by generating suggested plan given complexity, business value and themes.



# Discussion

- There can be a lot of uncertainty about velocity, and this can make our planning model less suitable in some situations.
- Eliciting values for themes is challenging, since the model needs the values to be on the same scale as story values.
- Further work addresses both: “Quantitative release planning for Extreme Programming”, submitted to IEEE TSE.
- Practices need to be validated and improved through other real-life applications.