

# Agile 2.0



# Who is Erik Drolshammer?

- Consultant Objectware AS – [erik.drolshammer@objectware.no](mailto:erik.drolshammer@objectware.no)



- Lead on Agile 2.0
- Co-founder of Objectware Open Community

# Agenda

1. Introduction
2. Mission statement
3. Structure and content overview
4. Summary

# Gartner, October 14, 2008

Quote 1:

*"The message for IT is clear; business needs and expects greater agility from IT,"* said Ms. Gomolski.

# Gartner, October 14, 2008

Quote 2:

*"The current approaches to project prioritization, resourcing, agility and governance are clearly not satisfying customer needs."*

# Post-Agile

## What is Post-Agilism?

This requires a two part answer. Post-Agilism is:

1. a growing movement of former Agilists who have moved beyond Agile methods, using a wide variety of software development tools and methodologies in their work.
2. an emerging era. Now that the Agile movement has moved to the mainstream, what's next?

Jonathan Kohl, <http://www.kohl.ca/blog/archives/000184.html>

# Post-Agile

*“Post-Agilism is simply doing what works for you.”*

Jason Gorman, <http://parlezuml.com/blog/?postid=407>

# Post-Agile

*“Post-Agilism is simply doing what works for you.”*

Jason Gorman

# Post-Agile

*“Post-Agilism is simply doing what works for you.”* Jason Gorman

- Agile 2.0 is more concrete
- Agile 2.0 is more than methodology
-

# **Manifesto for Agile Software Development**

**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.

# Mission statement

The purpose of Agile 2.0 is to create processes, techniques and technology which, in combination, will provide Cost Efficient and Easy to Adopt bootstrapping environments. The starting point is agile methodology, but the focus is to solve the challenges in the enterprise under realistic conditions.

# Mission statement continued

By this we mean extending/strengthen the agile manifesto with:

- evolvability and maintainability over project heroes

# Mission statement continued

By this we mean extending/strengthen the agile manifesto with:

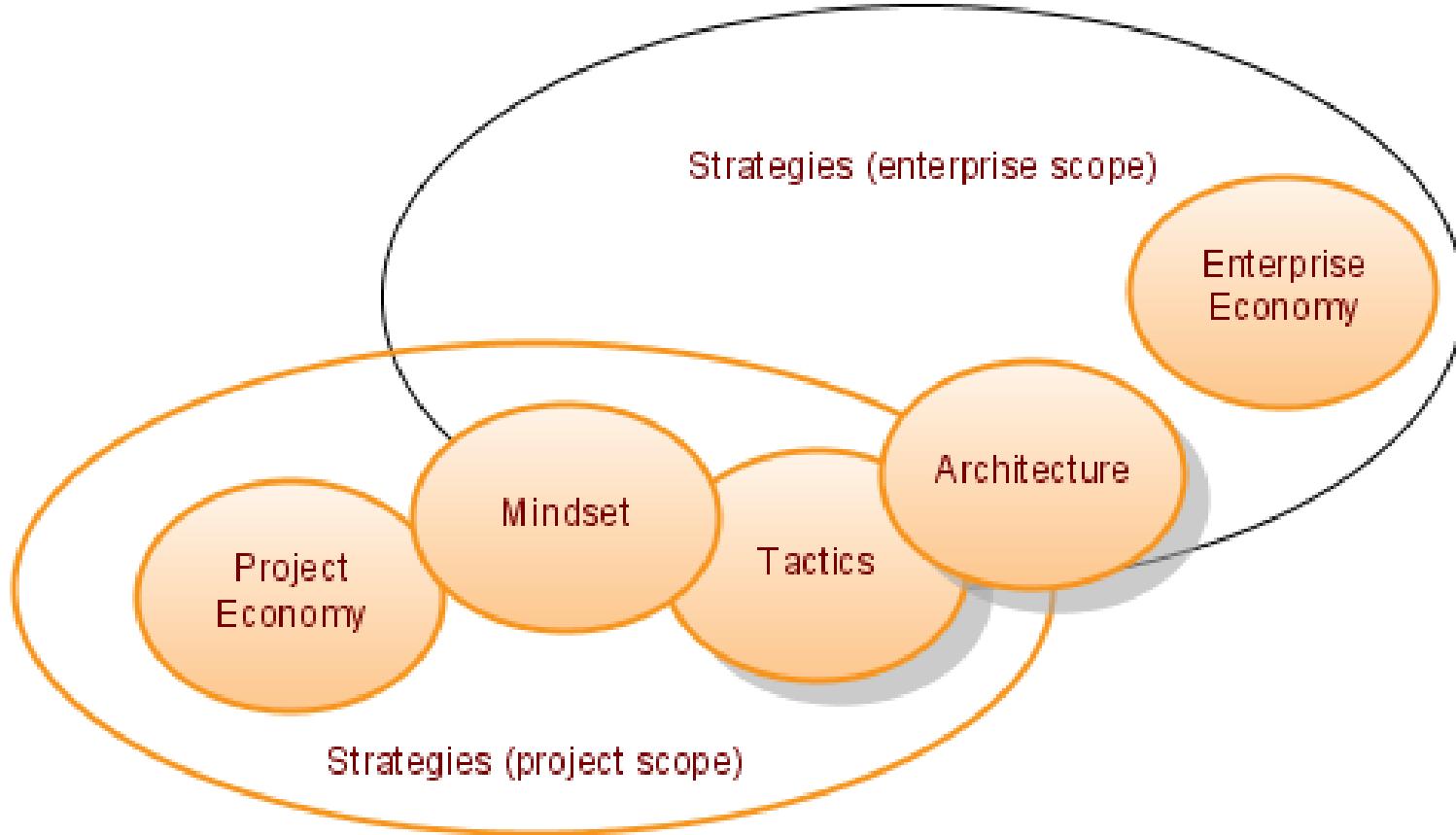
- **evolvability and maintainability** over project heroes
- **sustainability and total customer value** over glass bowl project values

# Mission statement continued

By this we mean extending/strengthen the agile manifesto with:

- **evolvability and maintainability** over project heroes
- **sustainability and total customer value** over glass bowl project values
- **facts and statistics** over religion and preaching

# Structure



# Structure

- Tactics
- Evaluation of tactics according to organizational drivers
  - Enterprise economy versus Project economy
- Agile and Software Architecture(s)
- Mindset and methodology
- Strategies

# Tactics

- Make it easy to adopt advanced concepts
- Smallest pieces of the puzzle
- Concrete, technical
- Similar to eXtreme Programming's *practices*

# Tactics – some examples

- *Enterprise Maven Infrastructure*
- *JigZaw – Agile Testing done Right*
- Deploy Java application as daemon
  - RPM, Java Service Wrapper, init.d-scripts

# Evaluation of tactics according to organizational drivers

- Enterprise economy versus Project economy

# Evaluation of tactics according to organizational drivers

- Team and organization maturity
- Time to market
- Is tactic A suited in this context?
- Applying tactic B makes tactic C and D very expensive.
  - How to choose?
  - Who makes the decision (project or enterprise)?

...

# Agile and Software Architecture(s)

- How to ensure a sound architecture when starting a new project?
- How to prevent the architecture from corrupting over time?
  - Technical and architectural **debt**
- How to avoid sub-optimization?
  - Which design/architecture decisions can a single programmer (or a pair) make by themselves?
  - How to make developers aware of that their decisions might have more far-reaching effects than their single, small component?
  - See Enterprise economy versus Project economy

# **Agile and Software Architecture(s)**

## **- some examples**

- Helper code as separate projects
- Use levels to group components according to functionality

# Mindset and methodology

- Which agile techniques have we found most valuable?
- When? In which context?
- Modifications and additions to the basic concepts/techniques

# Strategies

Tactics

- + Methodology
- + Architecture
- + Drivers

**= Strategy**

# Best documented pieces

- *Enterprise Maven Infrastructure*
- *JigZaw – Agile Testing done Right*
- Deploy Java application as daemon
  - RPM, Java Service Wrapper, init.d-scripts
- Helper code as separate projects
- Maven FAQ

# Summary: Agile 2.0

- ...use Agile Manifesto as basis
- ... is Post-Agile “compliant”
- ... is about **solving challenges** in the enterprise

# Resources

[http://wiki.community.objectware.no/display/smjidtonull/  
Agile+2.0+Community+Home](http://wiki.community.objectware.no/display/smjidtonull/Agile+2.0+Community+Home)

[http://wiki.community.objectware.no/display/smjidtonull/  
Agile+2.0+and+Post-Agilism](http://wiki.community.objectware.no/display/smjidtonull/Agile+2.0+and+Post-Agilism)