# Agile Software Development

# ANTONIO CARLOS SILVEIRA

Senior Software Development Manager
Yahoo!

**Blog**: http://www.acarlos.com.br/blog/
**Twitter**: acarlos1000
**e-mail**: acarlos@yahoo-inc.com

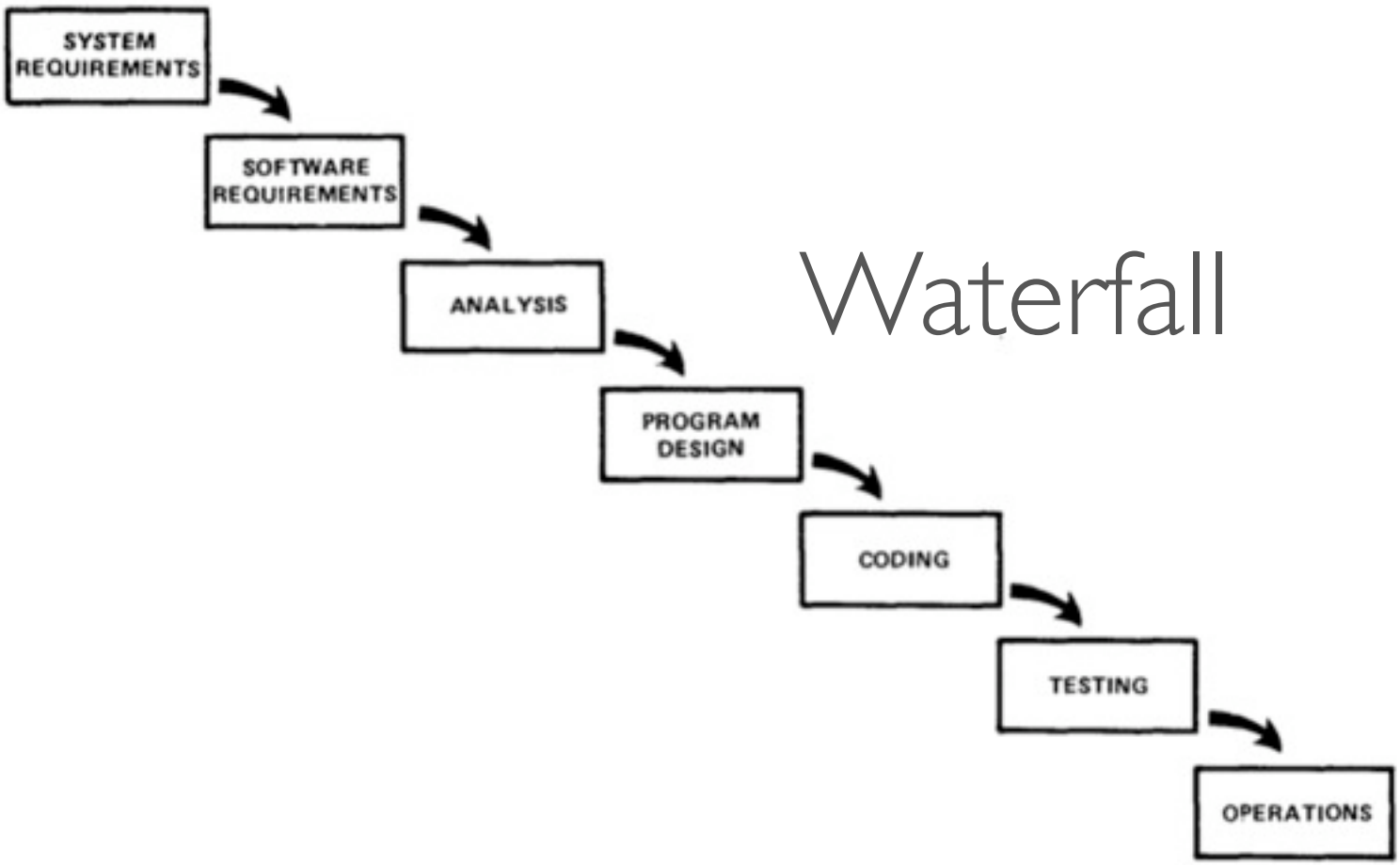
# AGENDA

1. Origins and principles
2. Roles
3. Meetings
4. Artifacts

1

# Origins and Principles

# Conventional Process




Waterfall

**Fonte: ROYCE 1970**

© Original Artist
Reproduction rights obtainable from
www.CartoonStock.com

"Somehow, it's not the same as when we were here on our honeymoon."

# Different reality

Different business needs and dynamics

OttawaCitizen
syndicom.com
THE NAZIS ARE COMING, THE NAZIS ARE COMING!!
TERROR IN THE 20TH CENTURY...
THE E-MAIL SERVER IS DOWN!!
TERROR IN THE 21ST CENTURY...

# Continuous Improvement

There is nothing new here

# The pioneers

Taylor

Ford

Sloan

Deming

Ohno

Nonaka

Takeuchi

# The Agilists

Jeff Sutherland

Ken Schwaber

Mike Cohn

Alistair Cockburn

Martin Fowler

Ron Jeffries

Kent Beck

# Agile Manifesto

| | | |
|---|---|---|
| Individuals and interactions | over | Process and tools |
| Working software | over | Comprehensive documentation |
| Customer collaboration | over | Contract negotiation |
| Responding to change | over | Following a plan |

Source: www.agilemanifesto.org

# Agile Methodologies

- ✓ Extreme Programming (XP)
- ✓ Crystal Clear
- ✓ Adaptive Software Development (ASD)
- ✓ Scrum
- ✓ Feature Driven Development (FDD)
- ✓ Dynamic Systems Dev Method (DSDM)

Friday, January 30, 2009

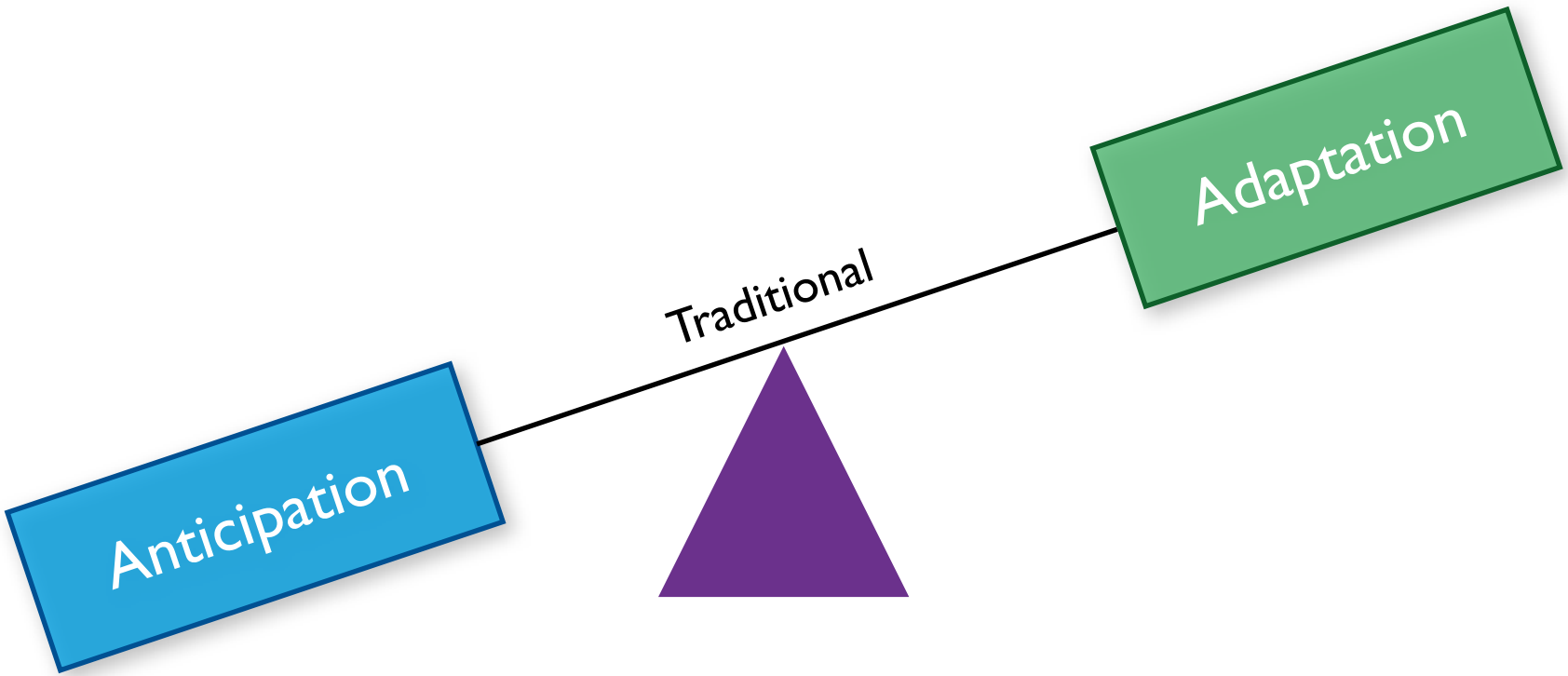
# SCRUM Who uses it?

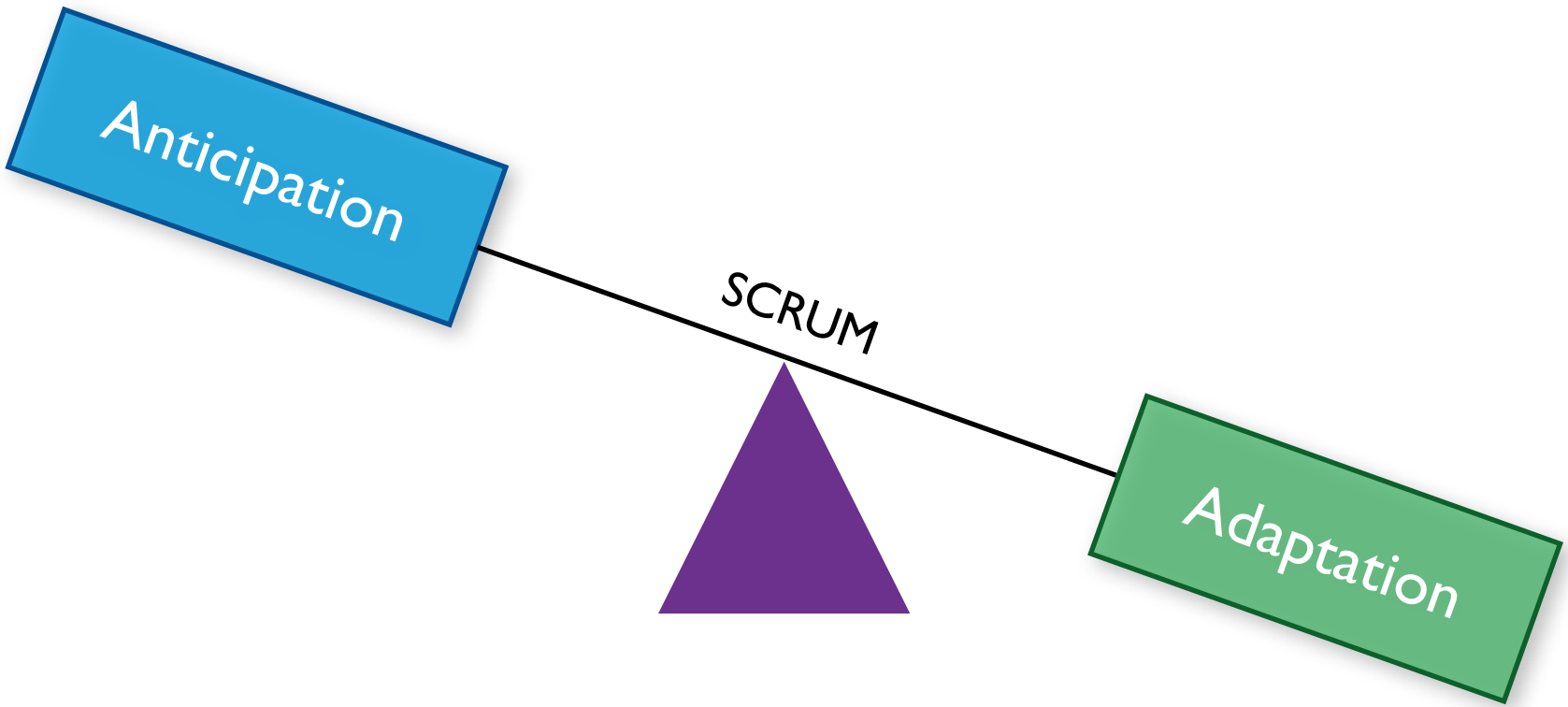
- Microsoft
- Yahoo
- Google
- Electronic Arts
- Lockheed Martin
- Philips
- Siemens
- Nokia
- BBC
- Nielsen Media
- SalesForce.com
- Oracle
- Time Warner
- Turner Broadcasting
- Toyota
- BMC Software
- CapitalOne
- Abril (Brazil)
- Globo (Brazil)
- UOL (Brazil)

- Delivery Focus
- Transparency
- Short and fixed cycles (15 days)
- Total Quality
- Team Work
- Constant Communication
- Commitment
- Self organization
- Reveal the problems

# Key Characteristics

- Self-organizing teams
- Enhanced Communication
- Commitment
- Product progresses in a series of 15 days “sprints”
- Requirements and needs go to the “product backlog”
- Working software always as the delivery
- Trust and Transparency
- Fast adaptation
- Time box
- Reveal the Problems


Adaptation
Traditional
Anticipation


Anticipation
SCRUM
Adaptation

# In the end its all about...

# Clear Understanding

# Clear Communication

EXIT
POOR
COMMUNICATION
FAIL!
STOP
Not an exit

# AND ...

Trust

Trust

# Origin and principles

2

# Roles

# The Boss

# The Boss

# with Scrum

## The focus is on **Team Work**

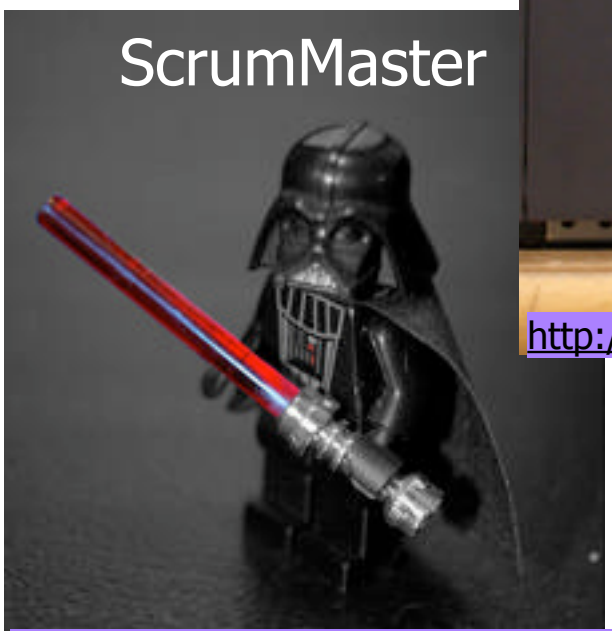
# Roles

Product Owner

http://www.flickr.com/photos/toechsner/2864422157/

http://www.flickr.com/photos/roger_g1/317671375/

ScrumMaster



http://www.flickr.com/photos/digital_stability/61466858/

# The Product Owner

- Define the features for the product
- Decide on release dates and its contents
- Responsible for the ROI, audience, etc.
- Prioritize features according to market and strategic value
- Interact with stakeholders and customers to define the Product backlog
- Adjust features and prioritize every sprint, as needed
- Accept or reject work results during the Sprint Review

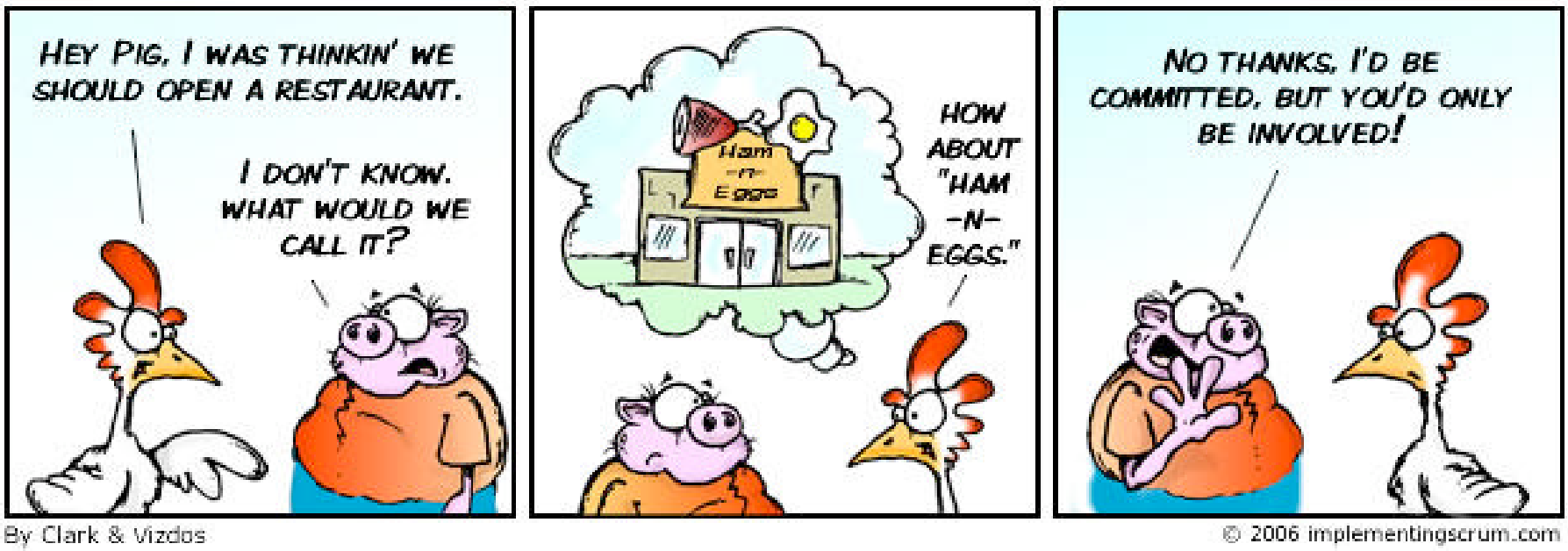
# ScrumMaster

- Protect values and principles
- Removes impediments
- Keep the team fully functional and productive
- Enable cooperation
- Shield the team from “dark forces”
- Facilitates integration
- He does **NOT** allocate tasks

# The Team

- Typically 5-9 people
- Cross-functional:
  - Programmers, testers, UED, etc.
- Full-time allocated
- Self-organized
- Responsible for the quality
- Estimates the complexity

# Pigs X Chickens



1 Origem e princípios

2 Roles

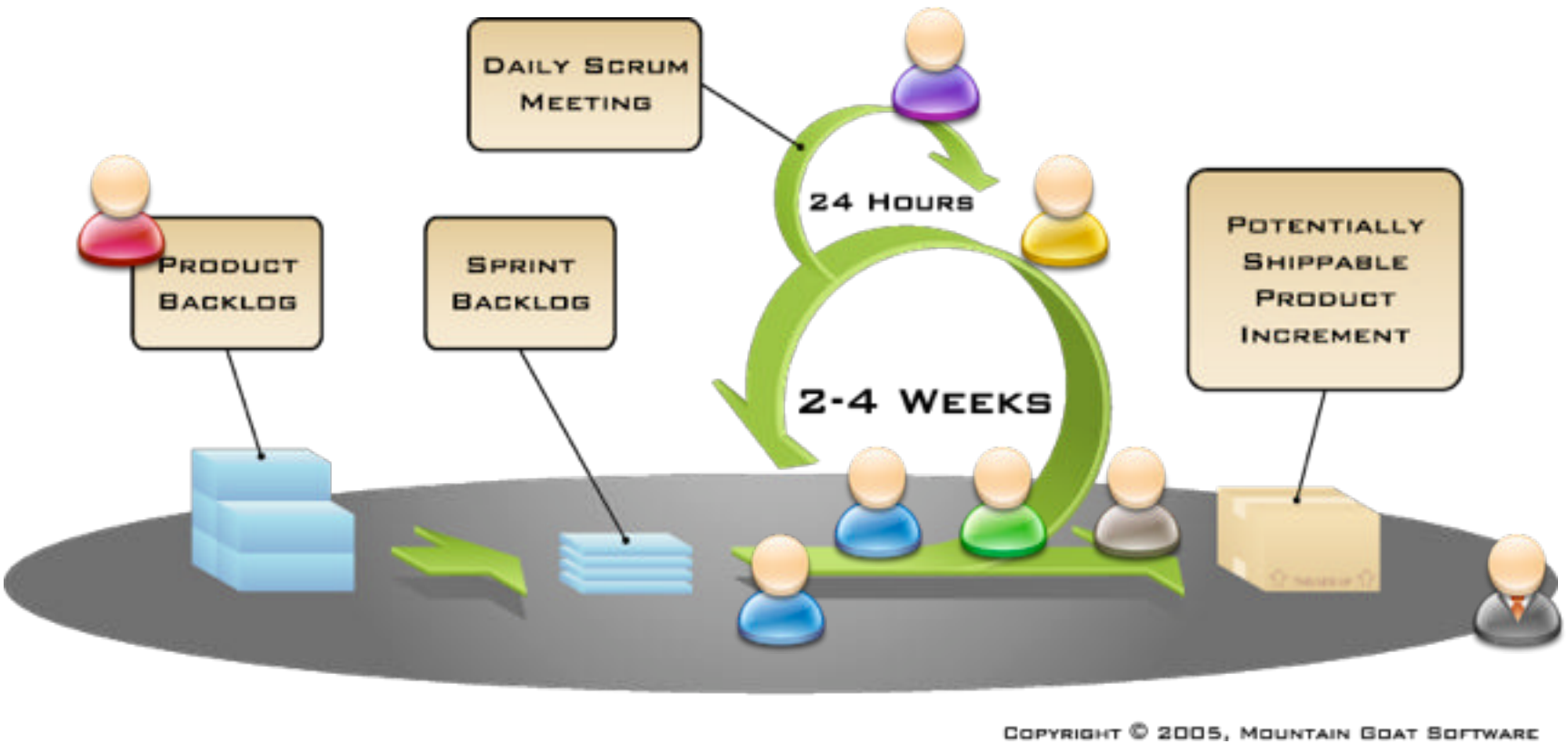
3

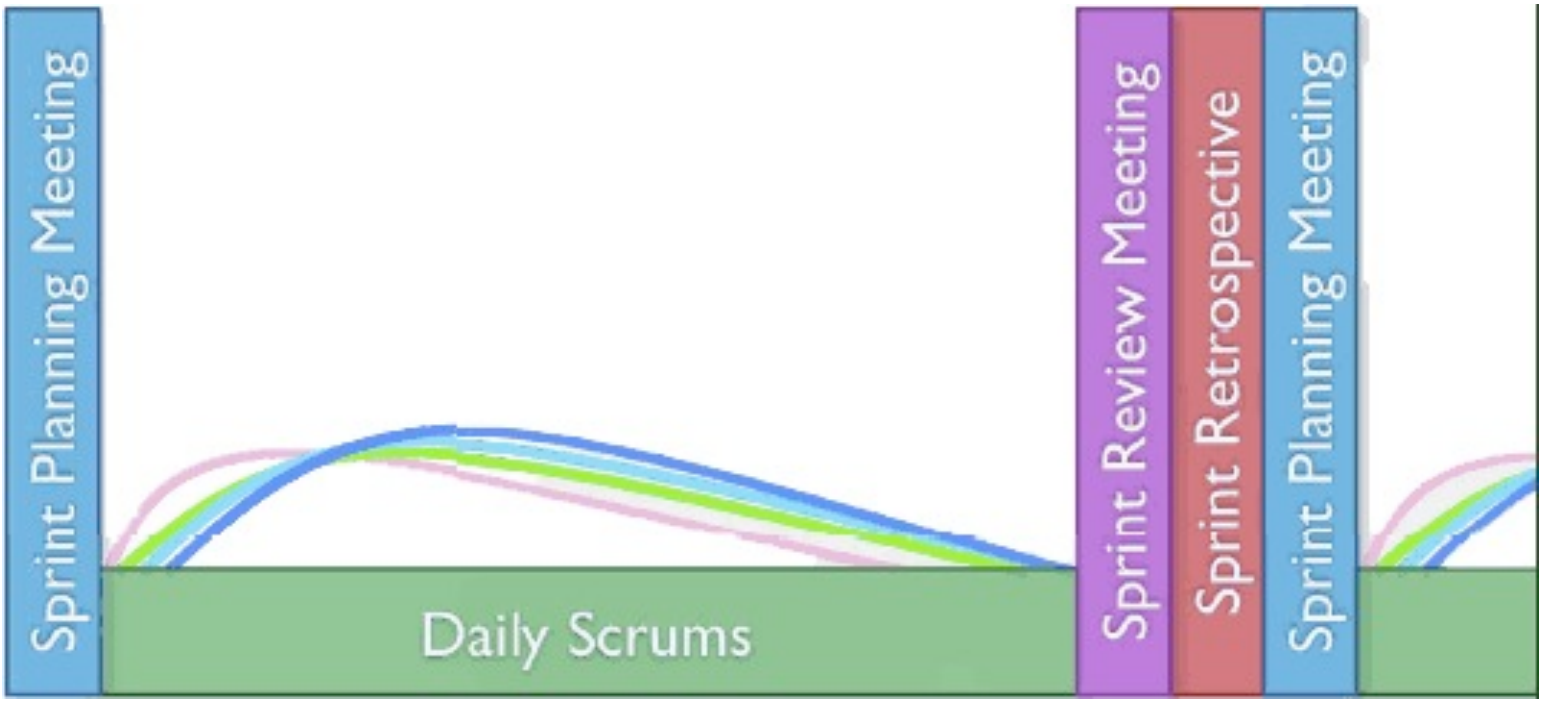
# Meetings

# Meetings

- Sprint Planning
- Sprint Review
- Sprint Retrospective
- Daily Scrums

# SCRUM Flow


Daily Scrum Meeting
24 Hours
Product Backlog
Sprint Backlog
2-4 Weeks
Potentially Shippable Product Increment
Copyright © 2005, Mountain Goat Software


Sprint Planning Meeting
Daily Scrums
Sprint Review Meeting
Sprint Retrospective
Sprint Planning Meeting

Time

# Sprint Planning

1. What are we going to build?
2. How are going to build it?

# Sprint Planning

- **A** PO explains the scope
- **B** Team estimates the complexity
- **C** Team chooses the Sprint backlog
- **D** Team writes the tasks for each story

PO

# Better communication

To avoid things like this...

FAIL
failblog.org

# Sprint Review

- Team demo what was accomplished during the sprint
- Each developer demo a story
- Informal, no slides
- Whole team participates
- Anyone is invited to participate (but only Pigs can talk)

Friday, January 30, 2009

16/12
20h
STAFF

# Sprint Retrospectiva

- What was cool?
- What can we improve?
- What the team can solve?
- What the company needs to solve?
- Only for PO, SM and Team

# Daily Scrum

1. What I did since the last meeting?
2. What I will do until the next meeting?
3. Any impediments?

Friday, January 30, 2009

TODO
WIP
DONE
BLOCKS
GOAL

1 Origins and principles

2 Roles

3 **Meetings**

4

# Artifacts

# Product Backlog

http://www.flickr.com/photos/aeror/2312354037//7415

# Sprint Backlog

- The product requirements
- Everything (ideas, features, epics)
- Expressed in User Stories
- PO keeps it organized
- Always prioritized and estimated

| Backlog item | Estimate |
| --- | --- |
| Allow a guest to make a reservation | 3 |
| As a guest, I want to cancel a reservation. | 5 |
| As a guest, I want to change the dates of a reservation. | 3 |
| As a hotel employee, I can run RevPAR reports (revenue-per-available-room) | 8 |
| Improve exception handling | 8 |
| ... | 30 |
| ... | 50 |

# User Story

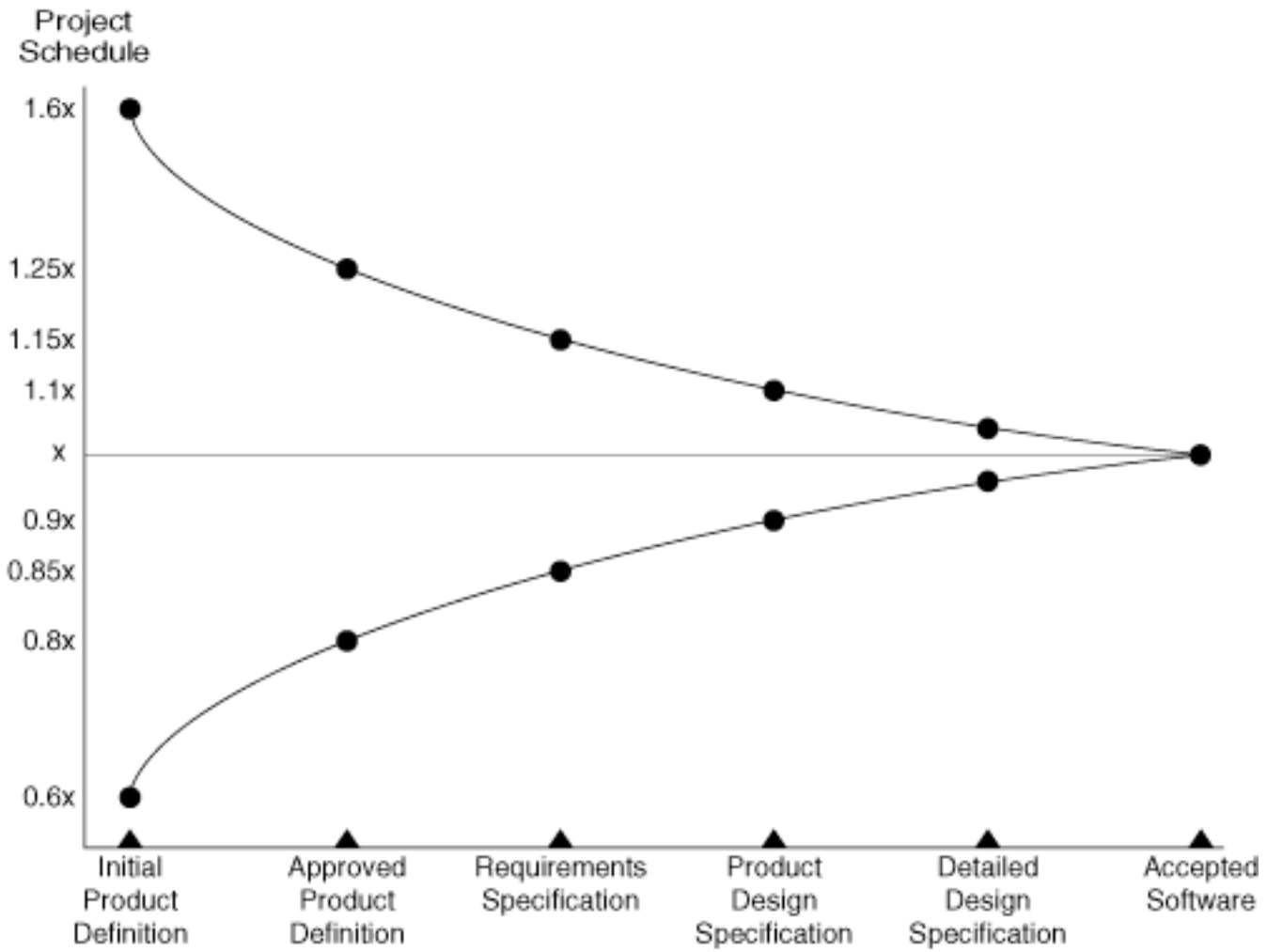
**I as a**<client>

**want** <feature>

**because** <ROI>

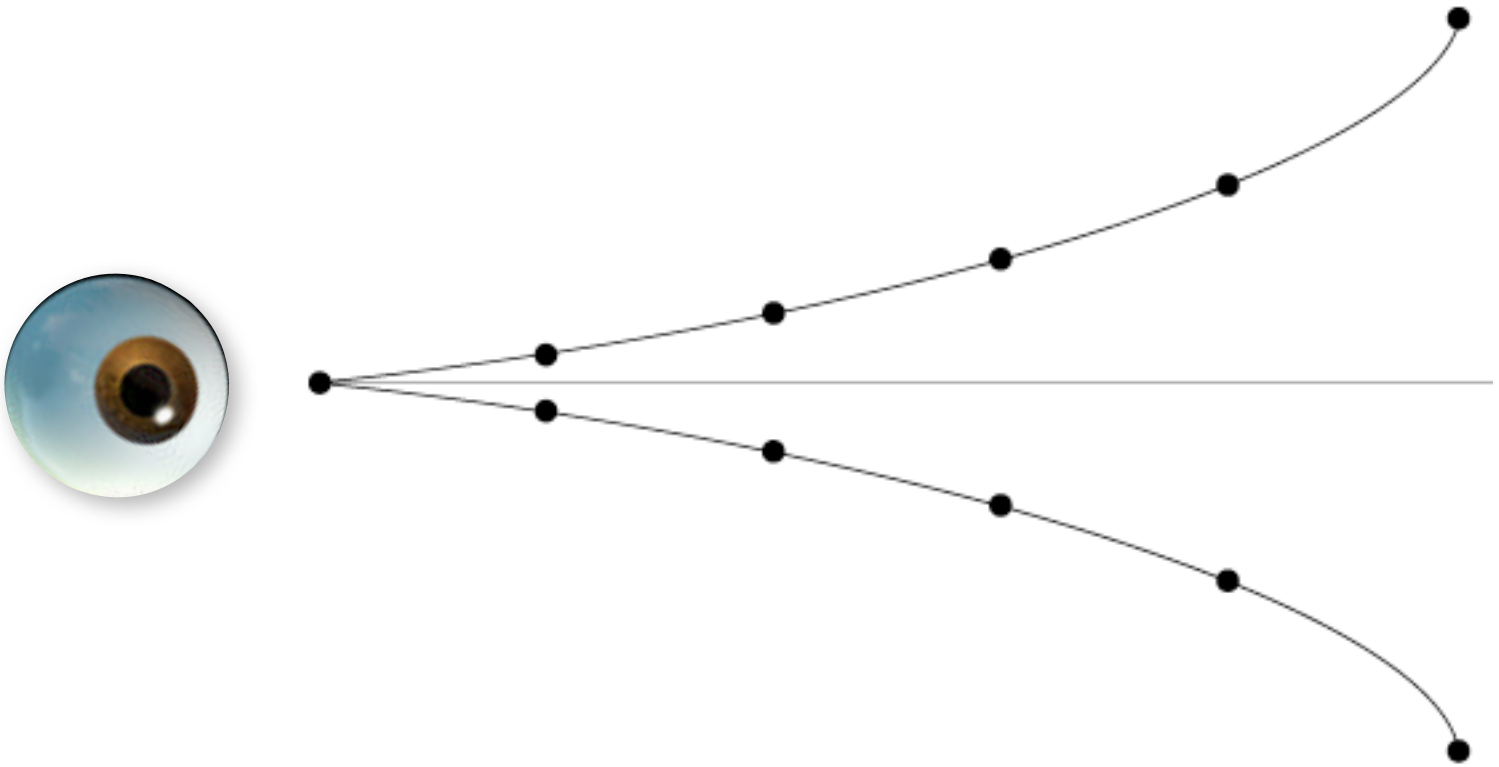
# User Story

# Uncertainty Cone

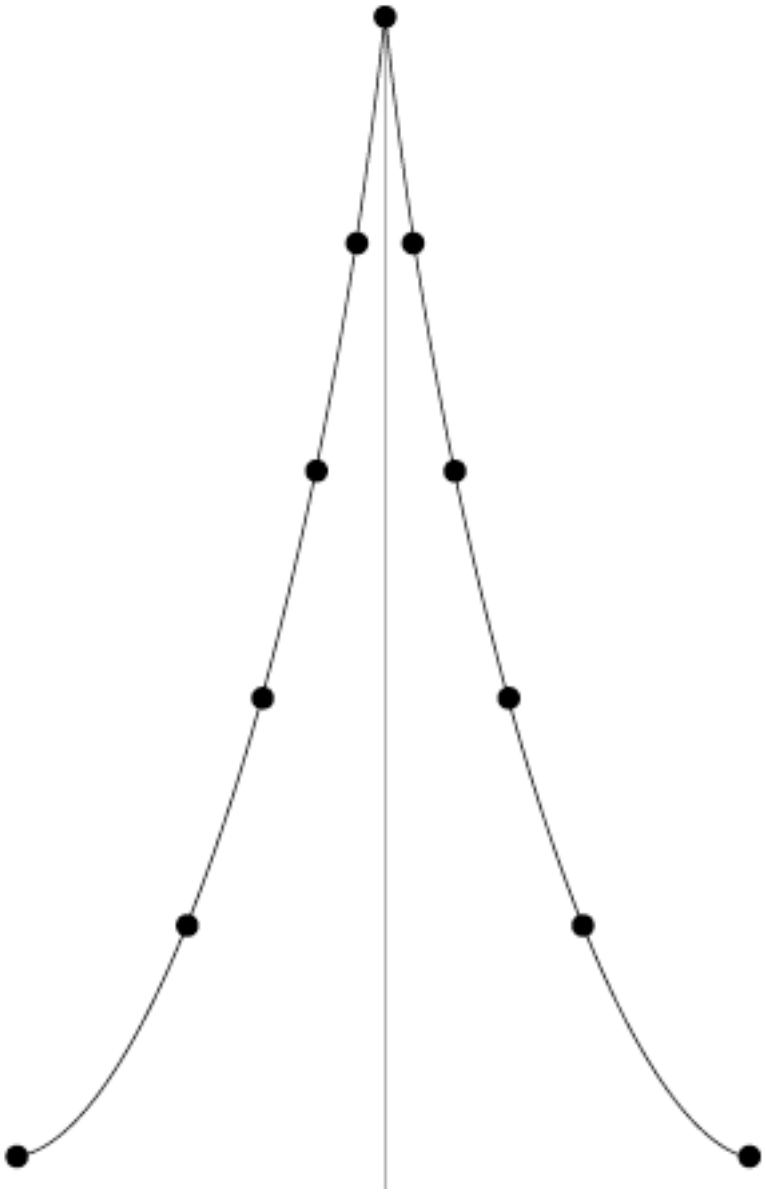



Barry Boehm

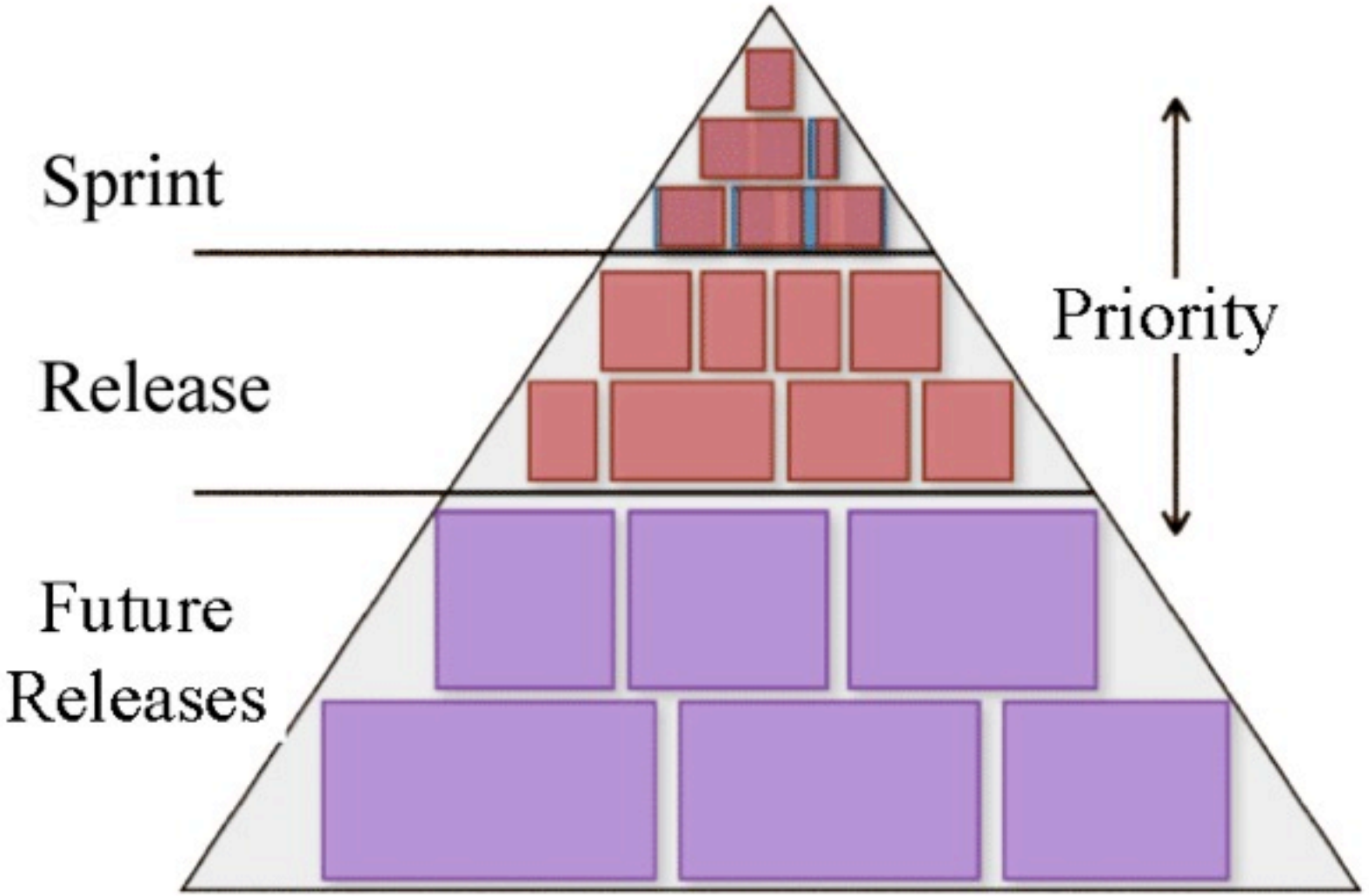
# Uncertainty Cone



# Uncertainty Cone

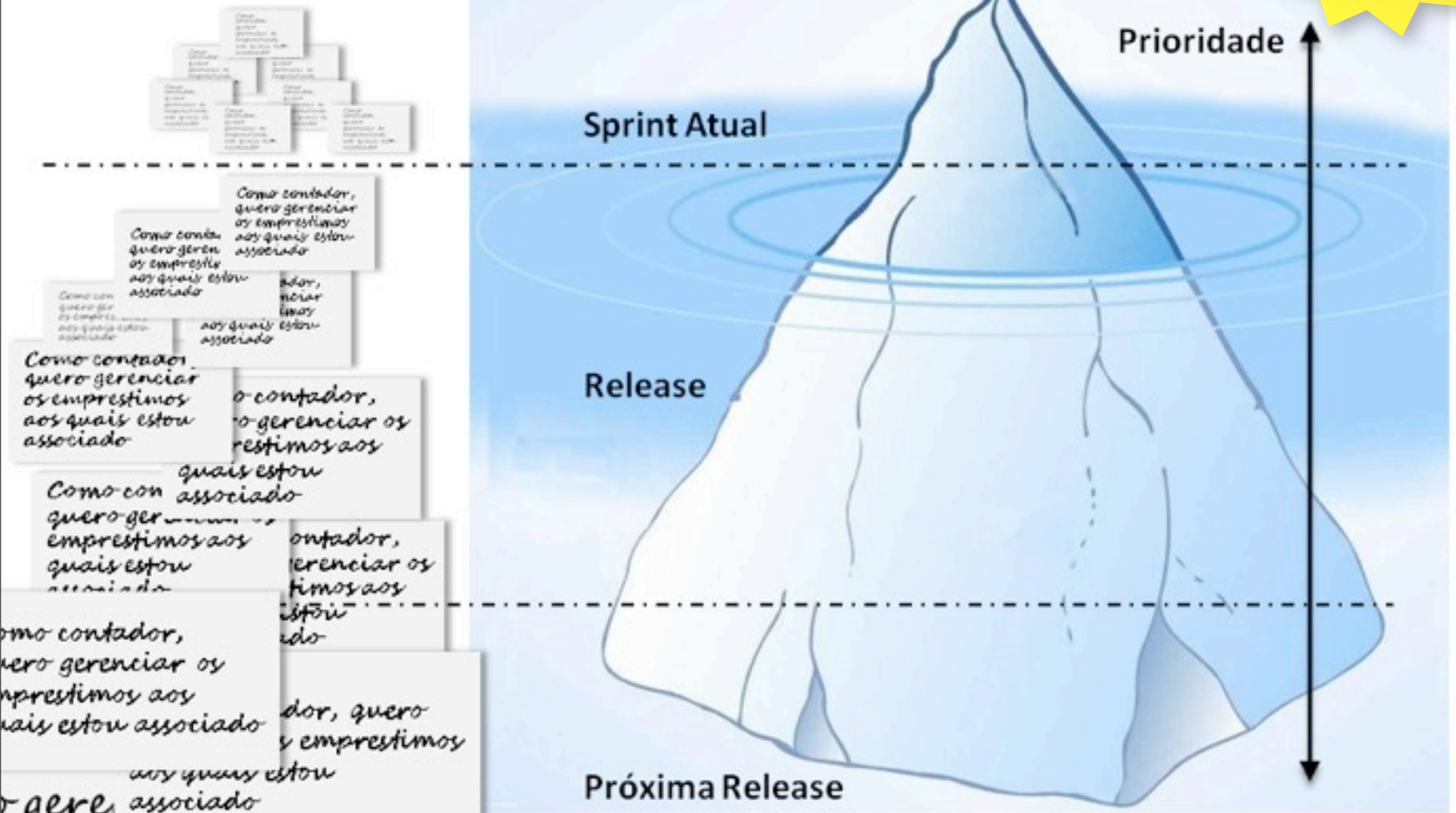


# PB Iceberg



# O iceberg do Product Backlogr

Web 2.0



# Sprint Backlog

http://www.flickr.com/photos/x180/469281131/

# Sprint Backlog

- User Stories selected by the Team
- Will be built in next 15 days
- Fully Estimated
- Divided in daily tasks

# No changes during the Sprint

# Definition of
# DONE

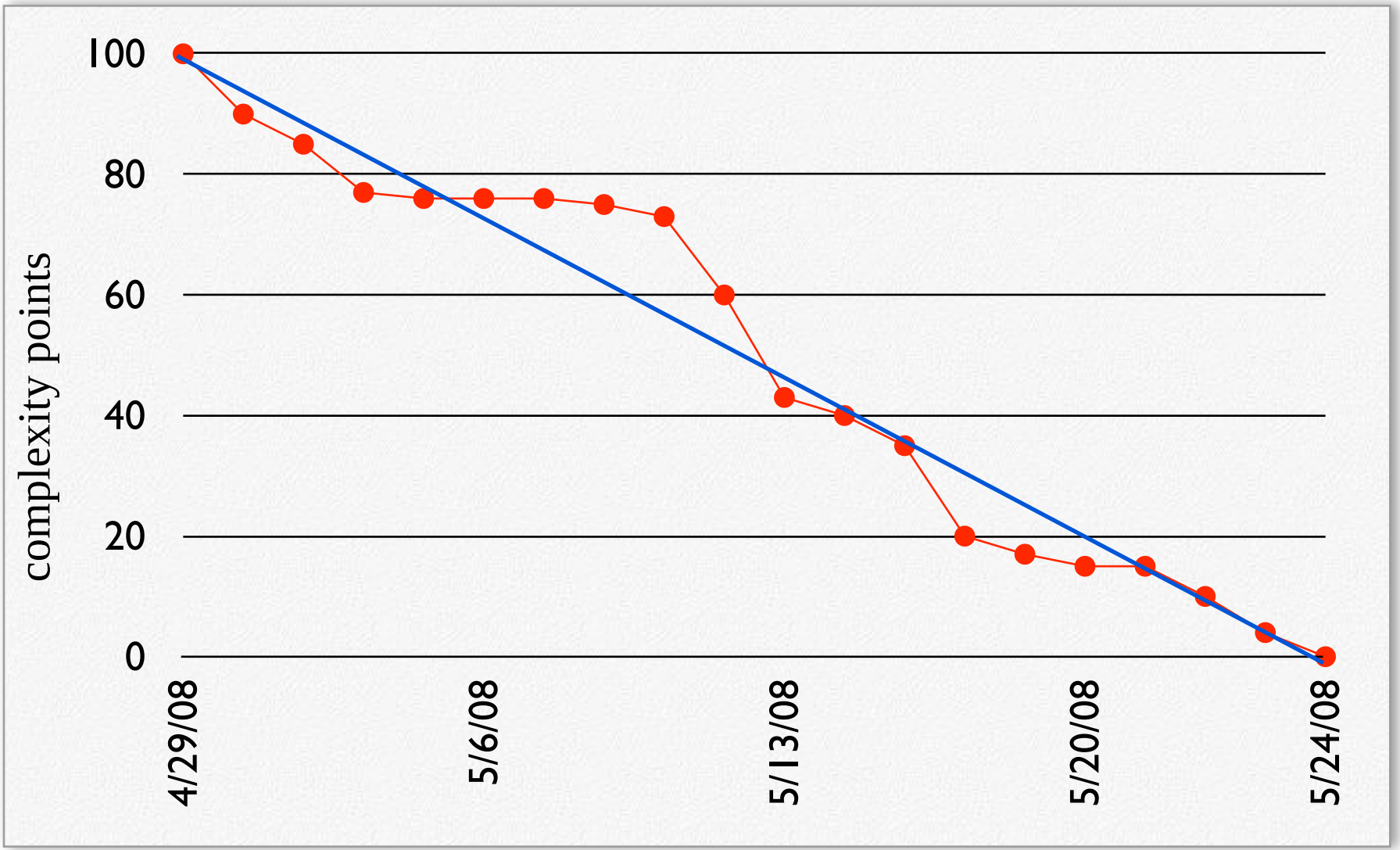
# DONE

- ✓ “DONE” or “NOT DONE”
- ✓ Avoids de 90% Syndrome
- ✓ Code produced, commented, checked in and run against current version in source control and meeting development standards
- ✓ Peer reviewed (or produced with pair programming)
- ✓ Builds without errors (CI)

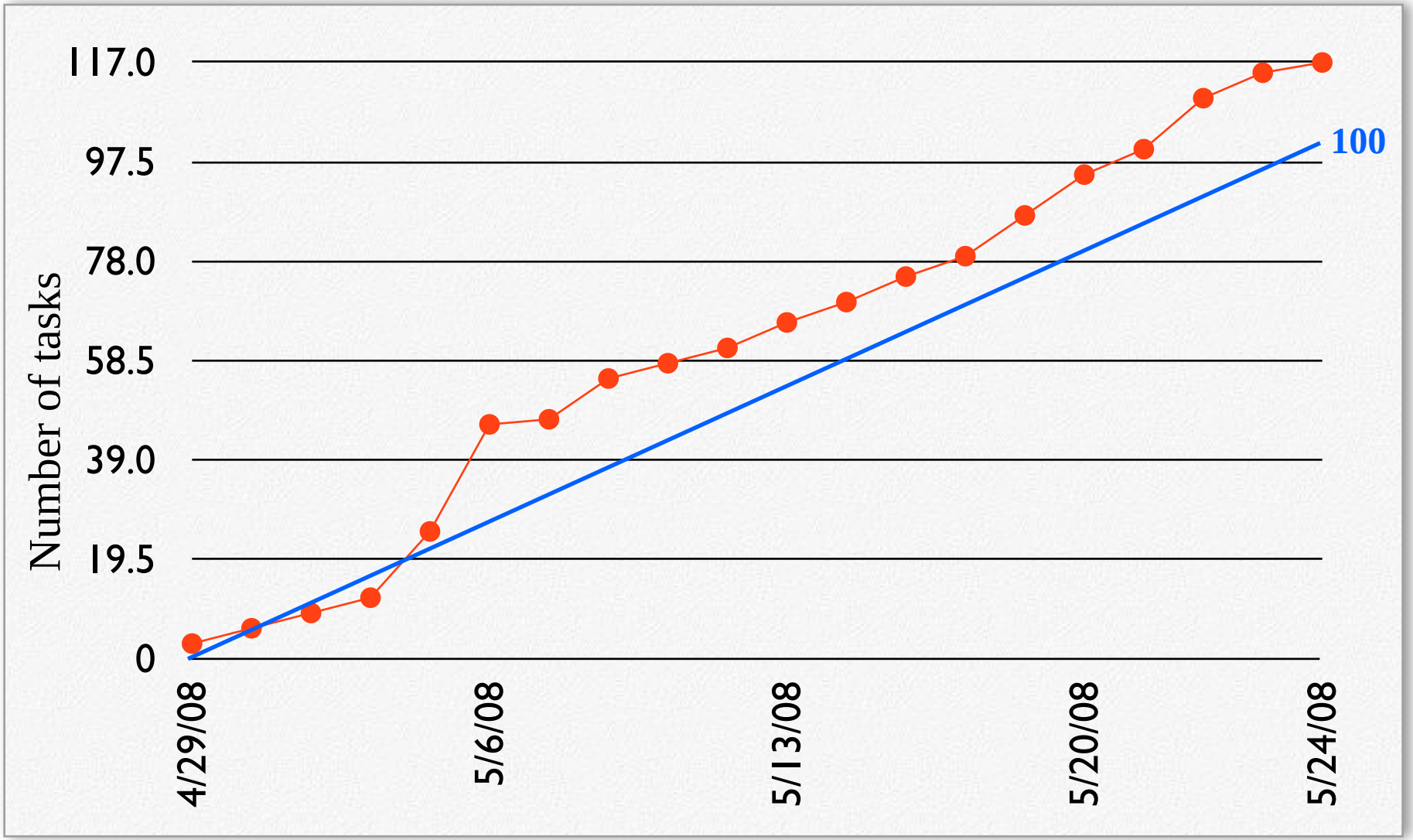
# DONE

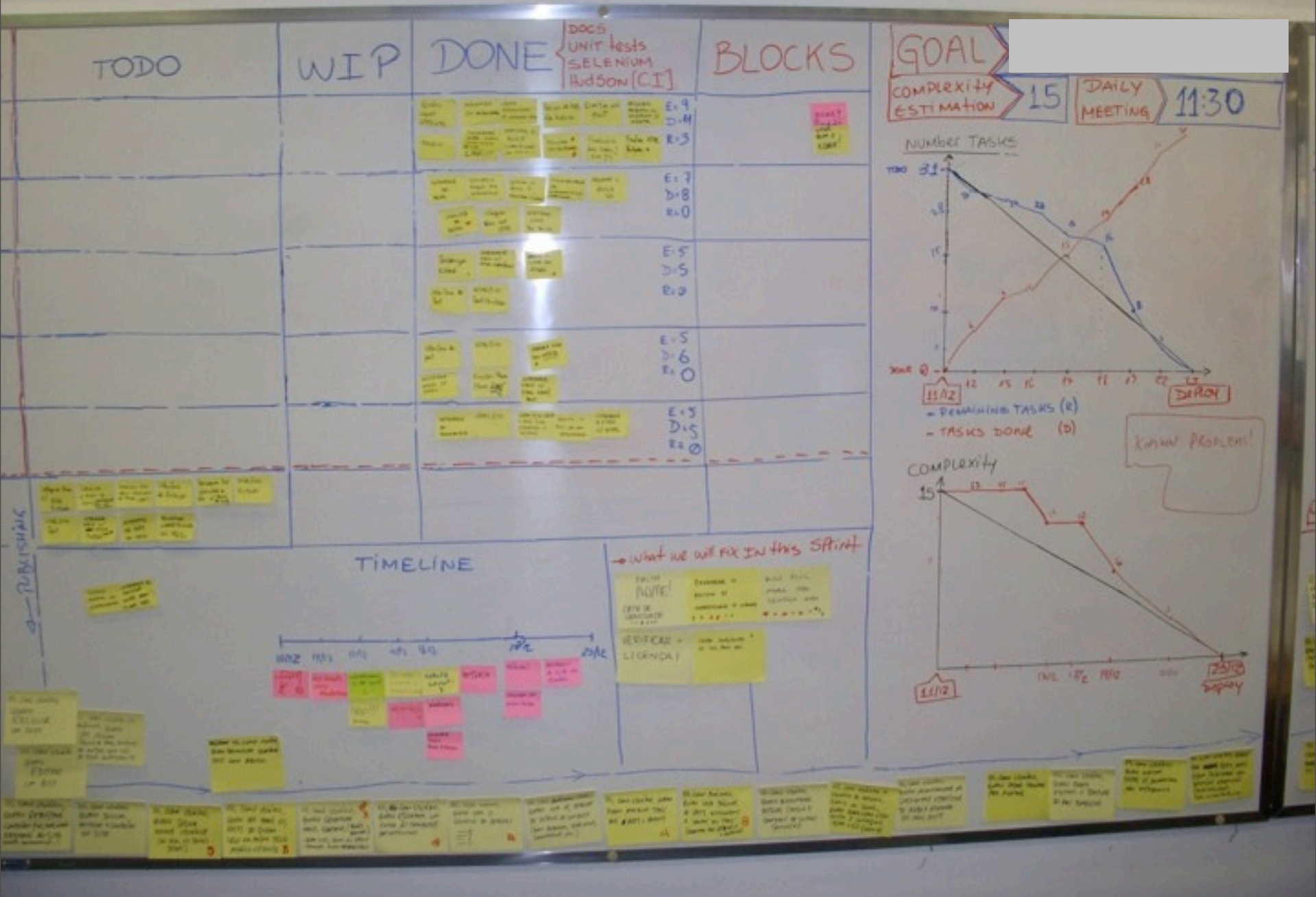
- ✓ Unit tests written and passing (TDD)
- ✓ Deployed to system test environment and passed system tests
- ✓ Passed UAT (User Acceptance Testing) (Selenium)
- ✓ Any build/deployment/configuration changes implemented/documented/communicated
- ✓ Relevant documentation/diagrams produced and/or updated
- ✓ Paranoid Compliant

# Burndown Chart

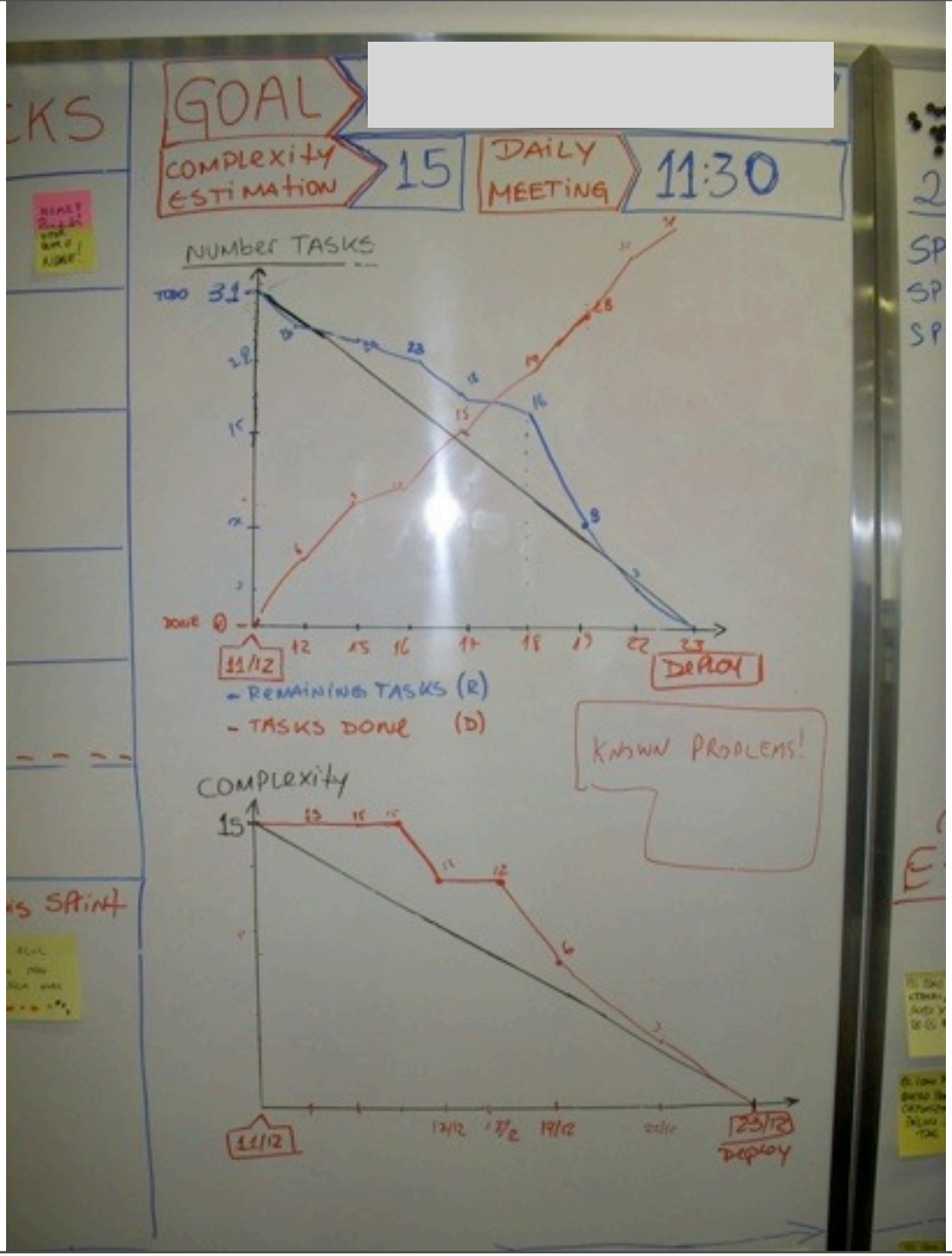


# Burn up Chart






Friday, January 30, 2009


GOAL
COMPLEXITY ESTIMATION
15
DAILY MEETING
11:30
NUMBER TASKS
TODO
DONE
11/12
DEPLOY
- REMAINING TASKS (R)
- TASKS DONE (D)
KNOWN PROBLEMS!
COMPLEXITY
15
11/12
23/12
DEPLOY

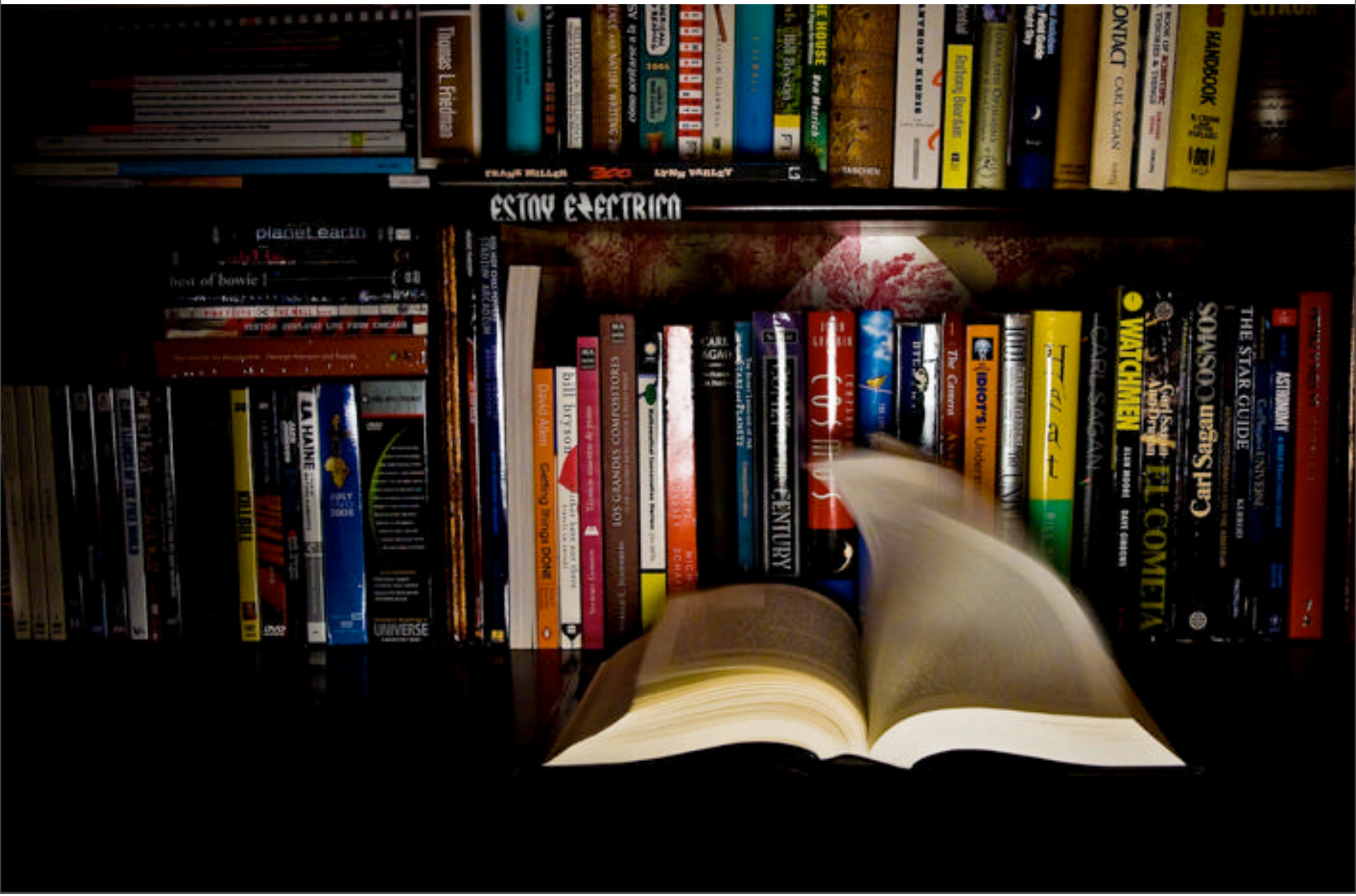
1 Origins and Principles

2 Roles

3 Meetings

4 Artifacts

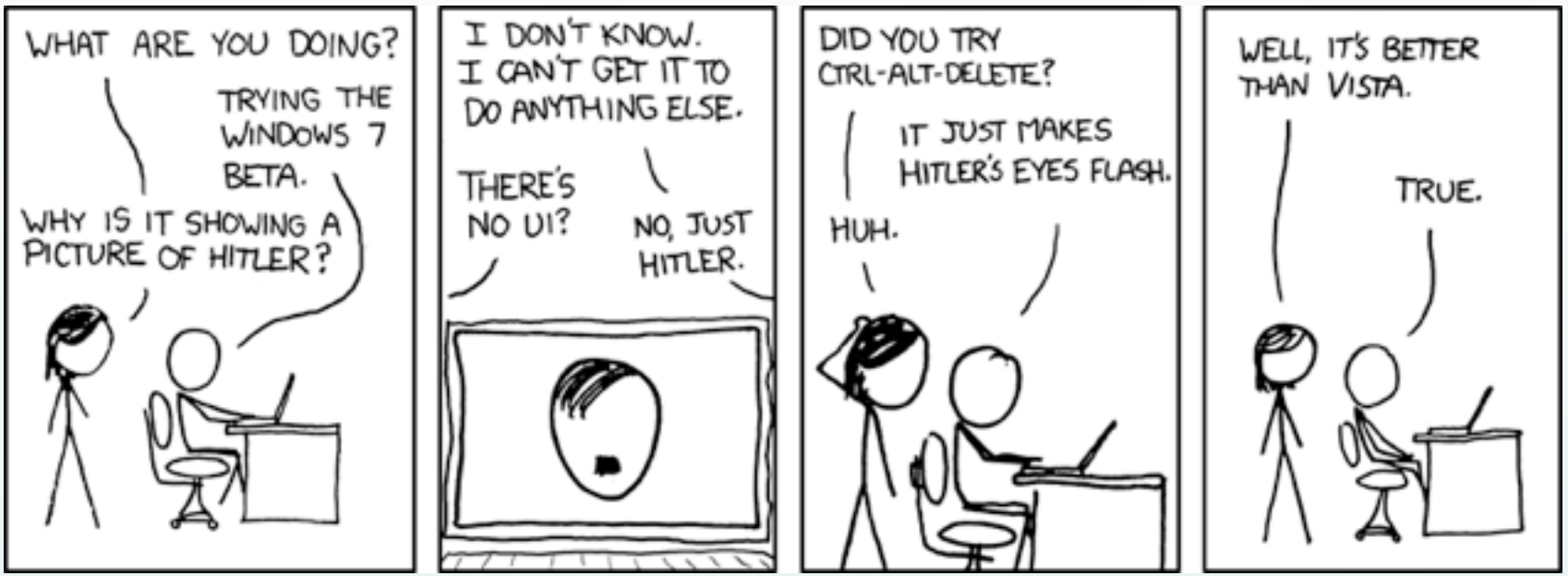
# Books


ESTOY ELECTRICO
planet earth
best of bowie
Thomas L. Friedman
CARL SAGAN
WATCHMEN
EL COMETA
Carl Sagan COSMOS
THE STAR GUIDE
ASTRONOMY
CENTURY
UNIVERSE
LA HAINE
David Allen
Getting things DONE
bill bryson
LOS GRANDES COMPOSITORES
HANDBOOK

- **Agile Software Development with Scrum** by Ken Schwaber and Mike Beedle
- **Agile Project Management with Scrum** by Ken Schwaber
- **Lean software development** by Mary Poppendieck
- **Agile and Iterative Development: A Manager's Guide** by Craig Larman
- **Agile Estimating and Planning** by Mike Cohn
- **Agile Retrospectives** by Esther Derby and Diana Larsen
- **Scrum and The Enterprise** by Ken Schwaber
- **Practices of an Agile Developer** by Venkat Subramaniam and Andy Hunt
- **Test Driven Development: By Example** by Kent Back
- **Getting Real** by 37Signals

Questions

# Thank you




**Blog**: http://www.acarlos.com.br/blog/
**Twitter**: acarlos1000
**e-mail**: acarlos@yahoo-inc.com