Inventing Agile Flavor

Agile is spirituality and not a religion

Venkatesh Krishnamurthy

17 mai 2006

Objective

- Theme of my talk
- What you would get out of this talk
- Release 1 Agile Best Practices
- Release 2 Agile Best Practices and tools
- Release 3 Agile Best Practices and tools
- Spirituality Vs Religion and Agile
- Various Agile religions
- Foundation for inventions
- Values and Lean Principles
- Questions

Agile = Spirituality

Agile ≠ Religion

Spirituality Vs Religion

"The main difference between Spirituality and Religion is the freedom to choose **your own path** towards enlightenment and ultimately, recognition of our Divine selves.

There is absolutely nothing wrong with many religions and belief systems, except that they are exclusionary and limiting…… (this para with key sentence is completed in the last slide !!)

http://www.omplace.com/articles/Spirituality_vs_Religion.html

What you would learn from this presentation ?

- You will learn how to migrate your process from Traditional method to an agile way

- You will get to know some of the tools that could be used to make your software development Productive

- You would learn the values and principles on which you can invent/Derive practices

Release 3 - Where are we right now ?

- Onsite and Offshore model with a distributed team

- Completed successfully 37 Iterations of 2 weeks each

- Stress free development

- Fully Agile with "Invented"/derived set of best practices

- Highly energized team

- 90% accuracy in planning and estimation

- Attrition – Highly Controlled

- Successful 4 releases of the product with a controlled delay of 3 weeks in 2 years !

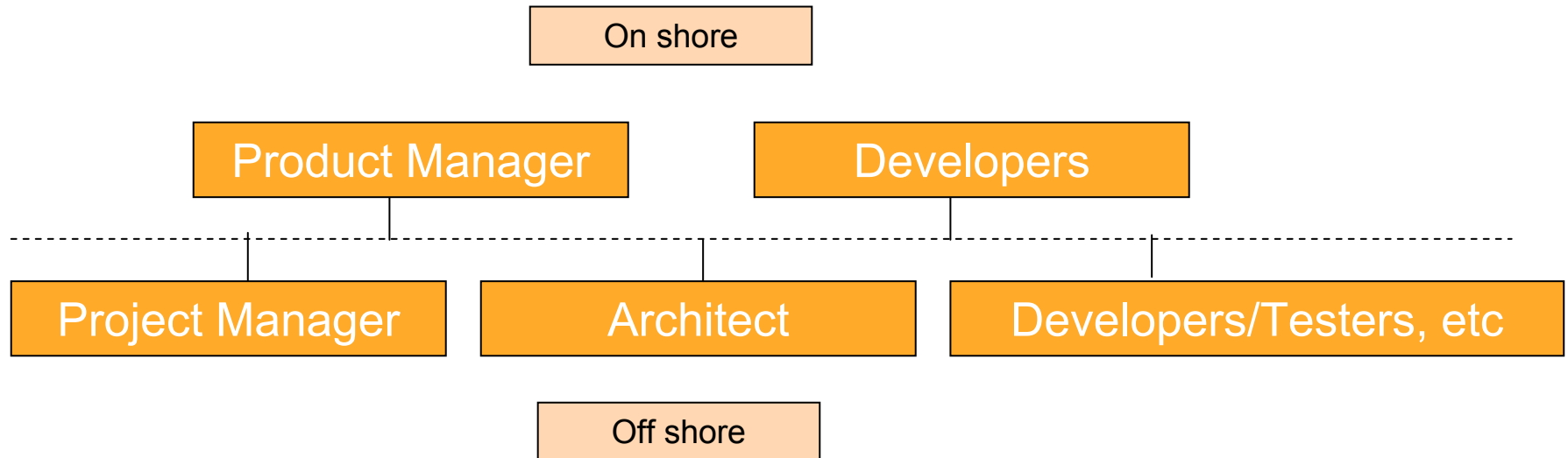# Tell me the Story from Release 1

17 mai 2006

Release 1 - Scenario

- **Project Type : Green Field**

- **Agile Experience of the team : None**

- **Total Team Size: 5 (during beginning of the project)**

- **Development Model: Onsite/Offshore**

- **Pricing: T & M**

- **Developers Background: CMM I and Traditional SDLC**
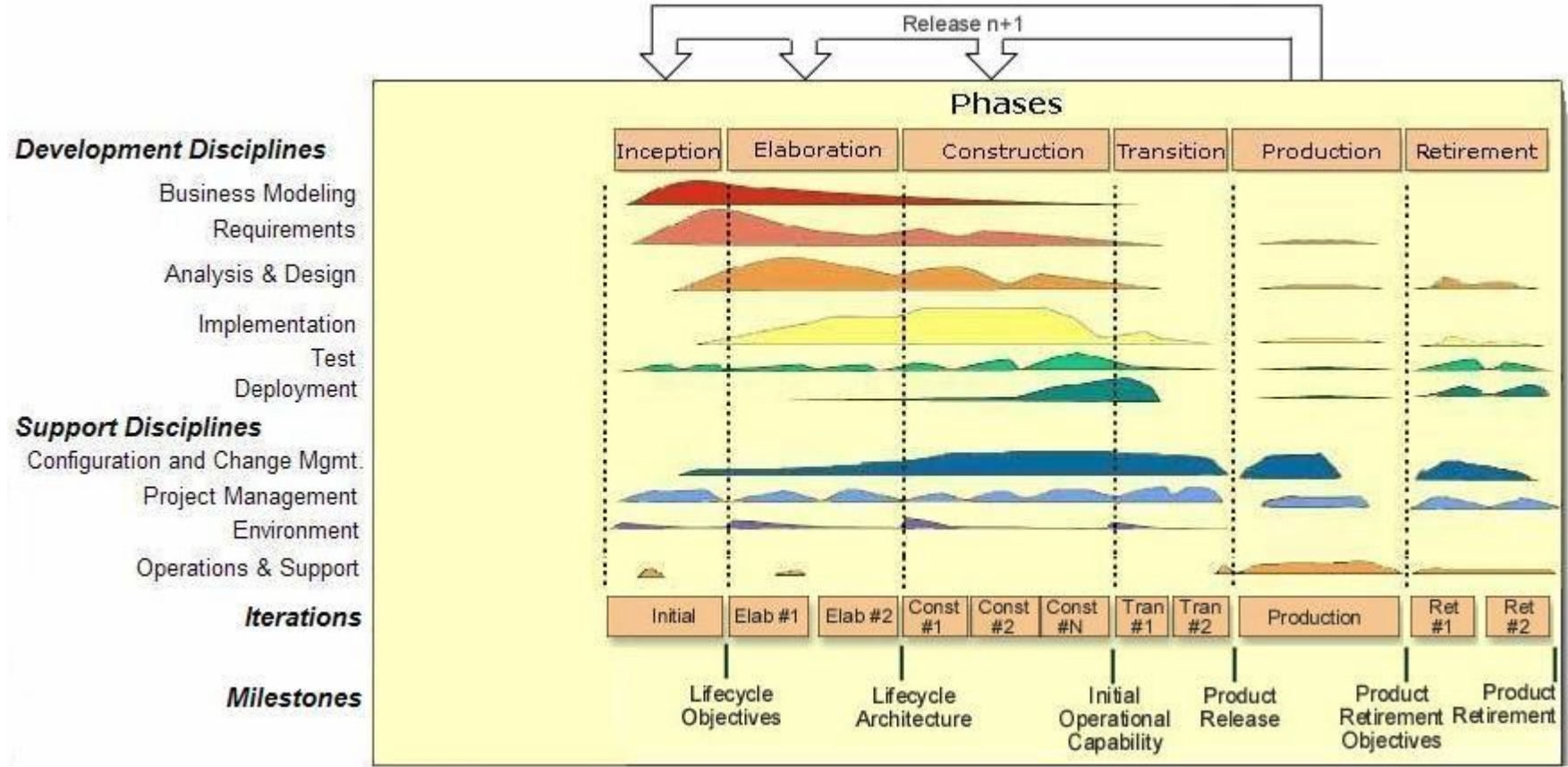
Release 1 - Team Structure

```
                          ┌──────────────┐
                          │   On shore   │
                          └──────────────┘

        ┌─────────────────────┐        ┌─────────────────────┐
        │   Product Manager   │        │     Developers      │
        └─────────────────────┘        └─────────────────────┘
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
  ┌──────────────────┐   ┌──────────────────┐   ┌──────────────────────────┐
  │ Project Manager  │   │    Architect     │   │ Developers/Testers, etc  │
  └──────────────────┘   └──────────────────┘   └──────────────────────────┘

                          ┌──────────────┐
                          │   Off shore  │
                          └──────────────┘
```

Doing the product development using Agile Methodology

First Release – 3 Months Milestone

- **Started off with 3 weeks iteration**

Baseline Architecture	Incremental Design, Development, Testing and Delivery

1 Week	2 Weeks	2 Weeks	2 Weeks	2 Weeks	1 Week	1 Week

Inception	Elaboration	Construction	Transition

| Iteration 1 | Iteration 2 | ... |

| Ana-lyze | De-sign | Code+Design+Test+Integrate |

| Ana-lyze | De-sign | Code+Design+Test+Integrate |
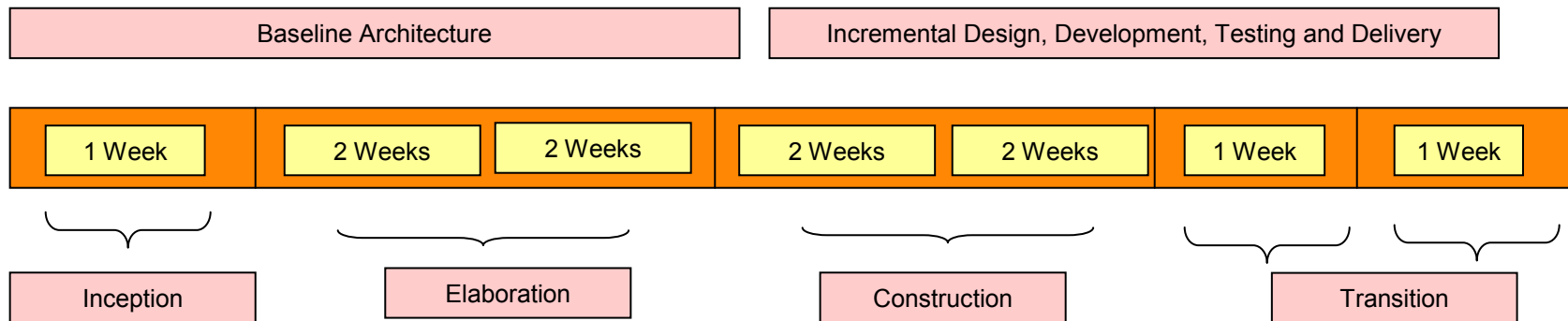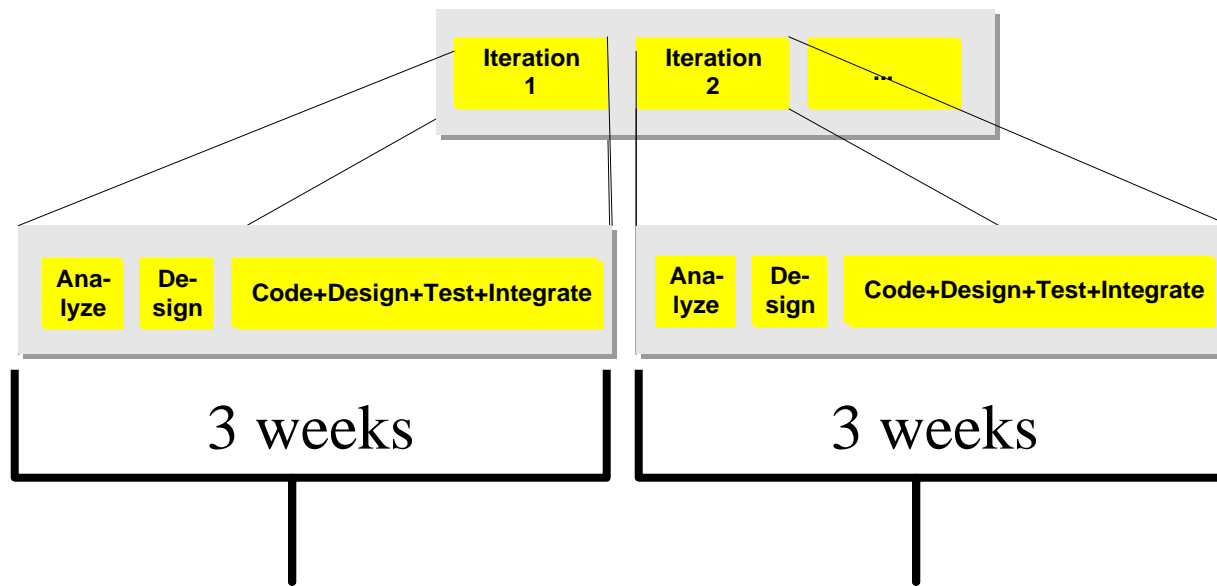
3 weeks   3 weeks

Release 1 - Detailed look at each week. Key Practices shown

valtech

Iteration 1	Iteration 2	...

Ana-lyze	De-sign	Code+Design+Test+Integrate

Ana-lyze	De-sign	Code+Design+Test+Integrate

3 weeks   3 weeks

Week1	Week2	Week3

Feedback loop

• Features Selection by the customer •Requirement Analysis • Mockups	•Design • Coding • Testing	•Coding Continued •Testing • Build • Integrate • Release for UAT
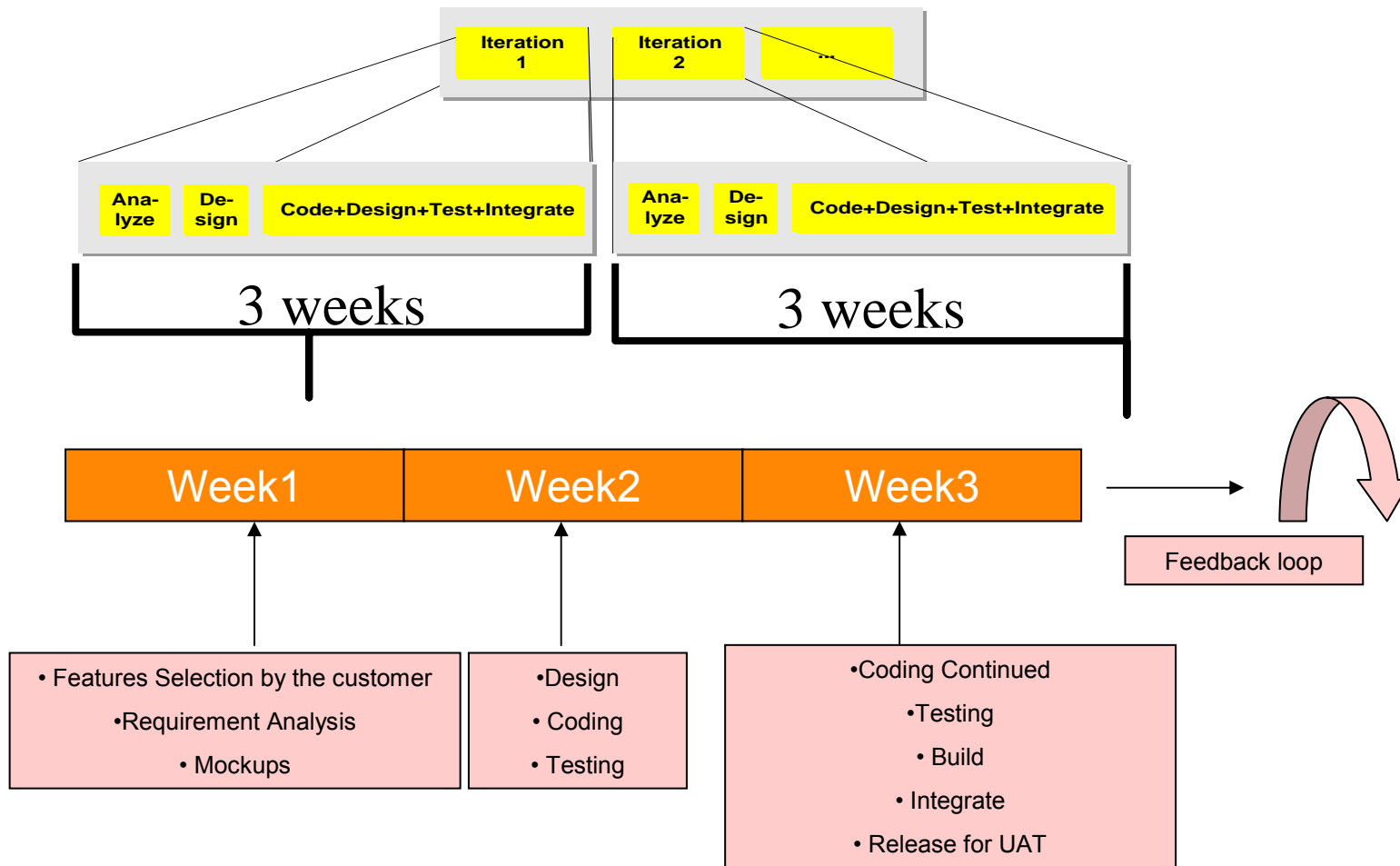
Release 1 - Development Testing Cycles

i   i+1   i+2

| Development	Test	Development	Test	Development	Test

Practices introduced in Release 1

- **Time Boxed Iteration (3 Weeks)**
- **Risk Driven and Priority Driven Development**
- **Client-Driven Development**
- **Feature Driven Development**
- **Daily 15 minutes stand up meetings**
- **Incremental Delivery**
- **Use Case Driven**

Non-Agile Practices
- **Heavy Documentation**
- **PM Controlled task allocation**
- **Bit resistance to frequent changes**
- **Microsoft Project Plan as the tool**
- **Handoff**
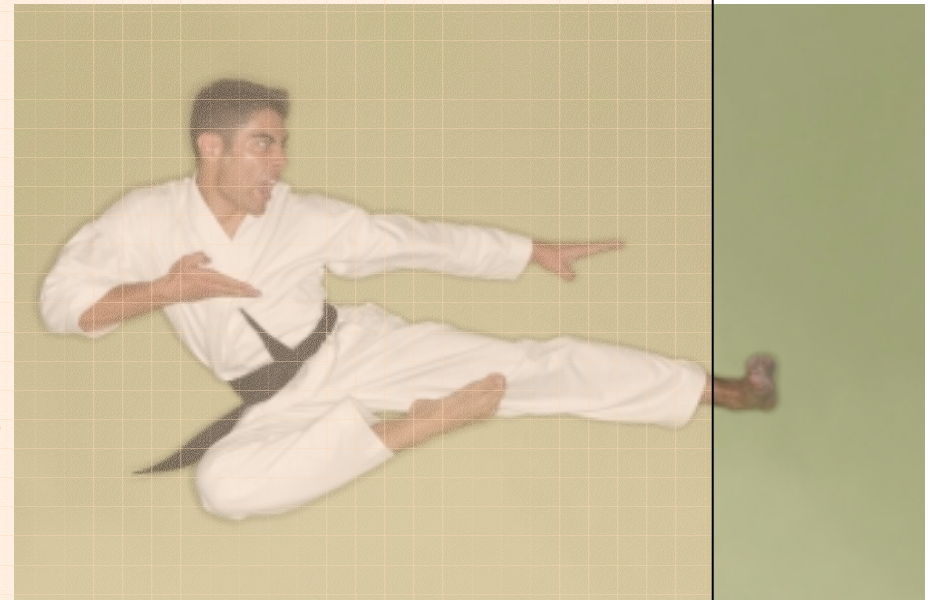
Release 2 Story

17 mai 2006

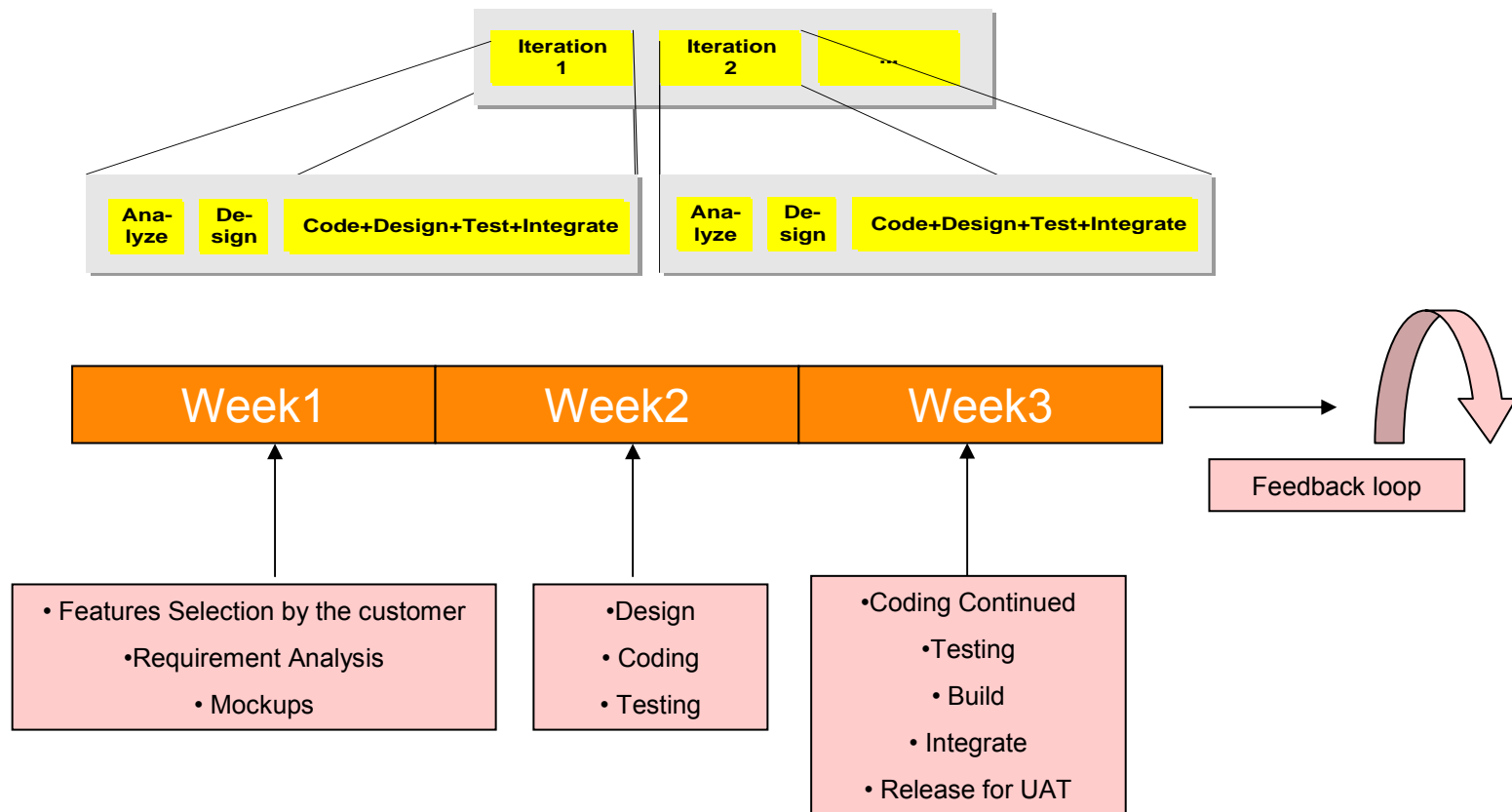- **Release review/introspection meeting**
  - Onsite
  - Offshore

- **New Practices, recommendation and research work started**

Release 2

- continued the 3 weeks Time boxed iteration model as we did in Release 1

| Iteration 1 | Iteration 2 | ... |

| Ana-lyze | De-sign | Code+Design+Test+Integrate | | Ana-lyze | De-sign | Code+Design+Test+Integrate |

| Week1 | Week2 | Week3 |

Feedback loop

Week1
- Features Selection by the customer
- Requirement Analysis
- Mockups

Week2
- Design
- Coding
- Testing

Week3
- Coding Continued
- Testing
- Build
- Integrate
- Release for UAT

Release 2 : Requirement Analysis

- Use Case based documents (UP)

• "Requirement days" through Interwise with   onsite team.

interwise Communication. Unlimited.

• Newcomers to see and listen through the "recordings" to get better understanding of requirements, rather through documents.

Release 2 – Planning and Estimation

- Discarding MPP and usage of home grown agile PM tool

- Team Planning: PM Relinquishing control of assignment of tasks, and the team volunteering to get what they want to do. (XP)

- Wide Band Delphi

Release 2 - Design

- Usage of light weight tools like *ArgoUML*

- Whiteboarding with design captured through camera (didn't work out well due to low resolution, battery and memory issues)

- Do what is necessary now (DWIN) attitude. We never spent time on building complex frameworks.

•Pipe Line Testing

Development	Test

i

Development	Test

i+1

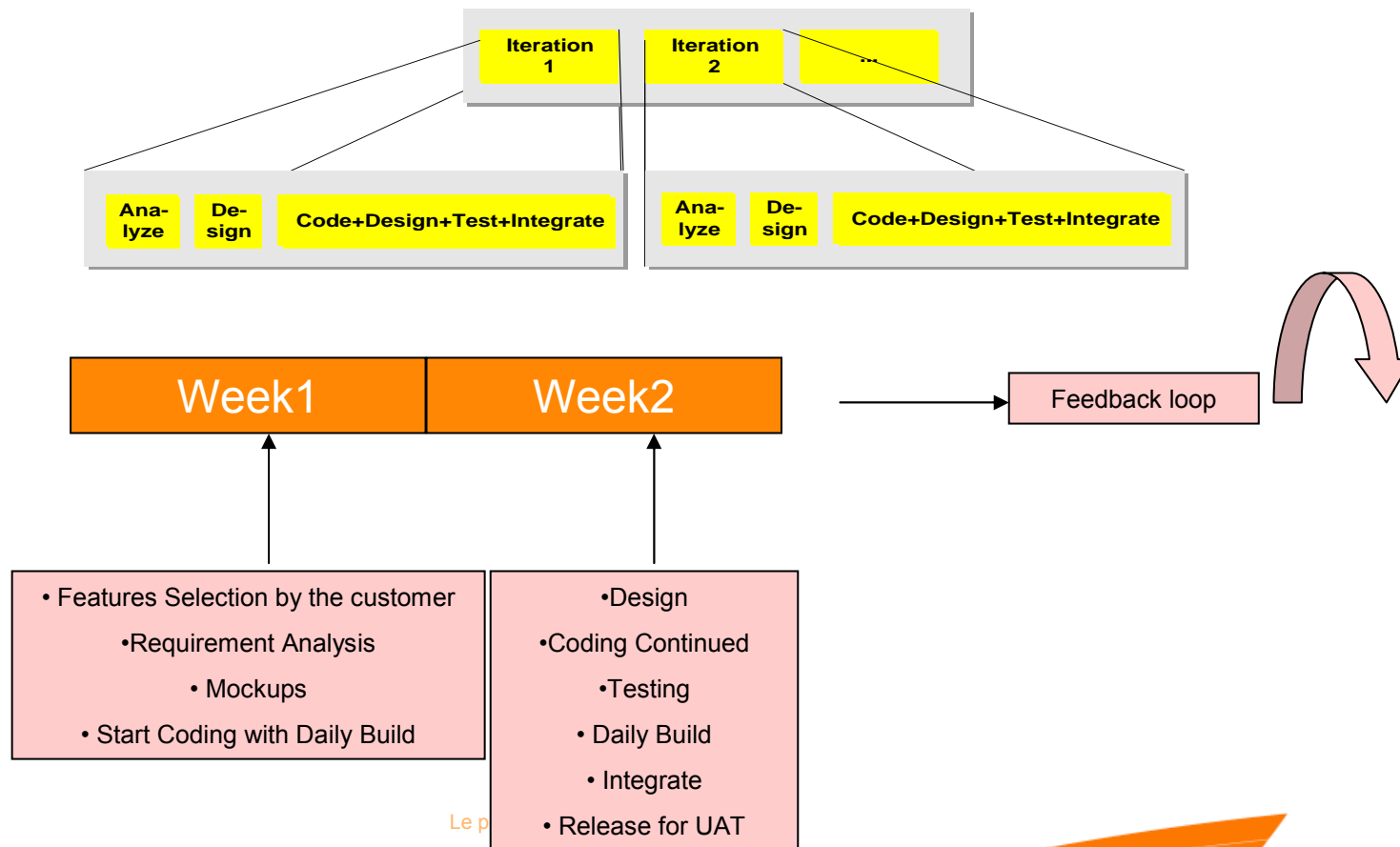| Development	Test

· · ·

i+2
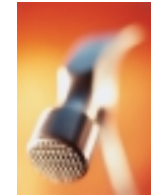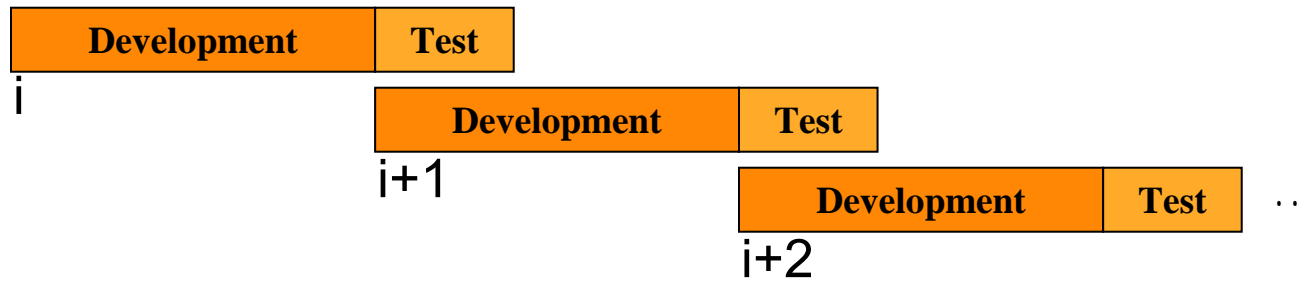
Release 2 – More best practices

- **Common Shared Repository**

- **Two Scrum meetings (SCRUM)**
    - With Local Development Team
    - With onsite team

- **Iteration retrospective (SCRUM)**

- **Release retrospective (SCRUM)**

- **Usage of IM such as Yahoo/MSN**

- **Peer Reviews**

Release 3

Team became comfortable with 2 weeks time boxed iteration

Iteration 1 | Iteration 2 | ...

Ana-lyze | De-sign | Code+Design+Test+Integrate

Ana-lyze | De-sign | Code+Design+Test+Integrate

Week1 | **Week2**

Feedback loop

- Features Selection by the customer
- Requirement Analysis
- Mockups
- Start Coding with Daily Build

- Design
- Coding Continued
- Testing
- Daily Build
- Integrate
- Release for UAT

Release 3 – Planning, Estimation

- Velocity based estimation

• Maturity increased with the team, and a very good rhythm of iterations being established.

• Usage of IEH during estimation and planning

• Wiki as the knowledge and information sharing medium

• Scrum notes being shared via Wiki with onsite team.

• Rotation of scrum master role (Invented)

Time: 0.00 hrs.

Distance (miles)

WIKIPEDIA
The Free Encyclopedia
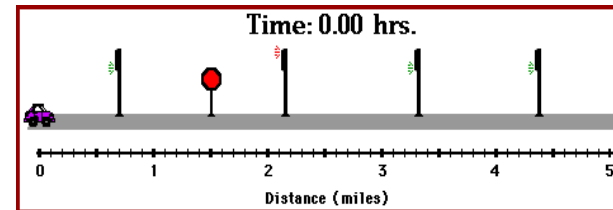
Release 3

- Continuous Integration, with Cruise control sending emails for build failures.

- Integration of Checkstyle/PMD/Junit with Cruise Control for quality control

- Big Visible graphs (XP)

- Daily Build

- Both Dev Team and Testing team being part of CI and Daily build activities

- Practicing TDD

- Back log management

- Feature teams

Release 3

- Daily 20 minutes refactoring cycle (Invented)

- Introducing Skype as another mode of communication in addition to interwise and other IMs, Wiki.

- Making effective use of graphs like Release Burn Down, Iteration Burn down, Discovery Rate to effectively plan, estimate the iterations

- Code Freeze used to happen for iterations every Wednesday, and incomplete things to move to backlog.
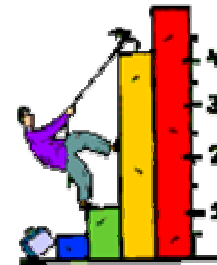
interwise Communication. Unlimited.

Spirituality Vs Religion

"The main difference between Spirituality and Religion is the freedom to choose **your own path** towards enlightenment and ultimately, recognition of our Divine selves.

There is absolutely nothing wrong with many religions and belief systems, except that they are exclusionary and limiting. The main goals and ideals of most world religions are very much the same, but in most we are taught that as followers of a particular belief system, we are right, we will be saved and the others, well, poor unenlightened ones, they will be left behind. … "

In Para 2, replace the word "religions" with "agile flavors" and you will see a new dimension of agile world.

http://www.omplace.com/articles/Spirituality_vs_Religion.html

Various Agile Religions

approximately 60 key practices from following agile religions

1. Extreme Programming (XP)
2. Scrum Development
3. Crystal Methodologies
4. Feature Driven Development (FDD)
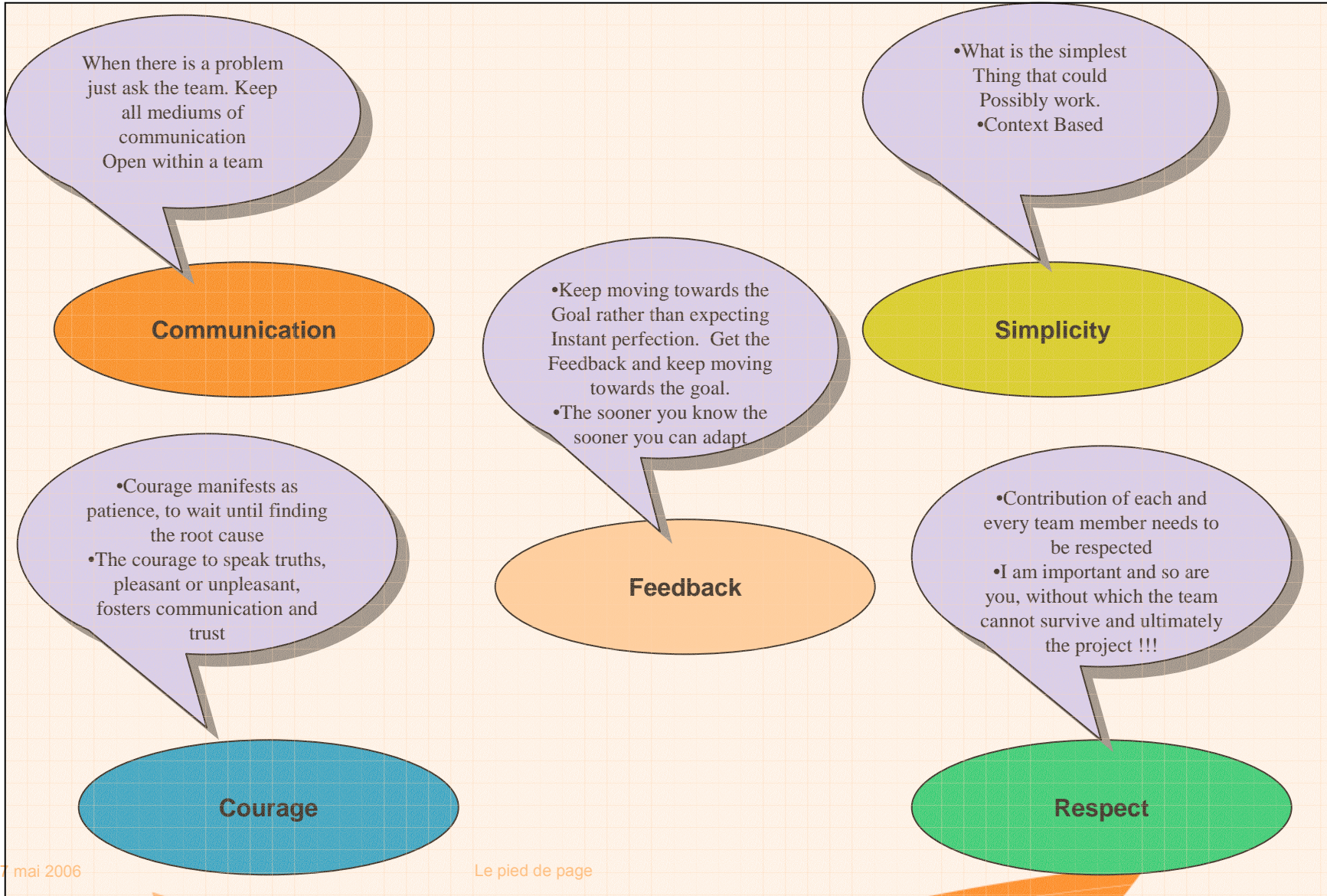5. Dynamic Systems Development Method (DSDM)

All Inventions to be based

- ▪ **Values**
- ▪ **Principles**

Values

When there is a problem just ask the team. Keep all mediums of communication Open within a team

•What is the simplest Thing that could Possibly work.
•Context Based

Communication

•Keep moving towards the Goal rather than expecting Instant perfection. Get the Feedback and keep moving towards the goal.
•The sooner you know the sooner you can adapt

Simplicity

•Courage manifests as patience, to wait until finding the root cause
•The courage to speak truths, pleasant or unpleasant, fosters communication and trust

Feedback

•Contribution of each and every team member needs to be respected
•I am important and so are you, without which the team cannot survive and ultimately the project !!!

Courage

Respect

Lean Principles

1. Specify what creates *value* from the customers perspective

2. Identify all steps across the whole *value stream*

3. Make those actions that create value *flow*

4. Only make what is *pulled* by the customer just-in-time

5. Strive for *perfection* by continually removing successive layers of waste

Thank you

www.valtech.com