# OO – Project Management Experiences, 4 steps


mueller & friends ™
NEUE REALITÄTEN SCHAFFEN

A sucessful realization of projects using OO-technology brings new opportunities/challenges. It requires not only OO methodology/skills, but also a new working/cooperation style: the 90° Shift

**1** 1990
OO Discovered,
Start Pioneering

From **procedural programming**   To **object orientation**

Conflicts   Needed help   **2**

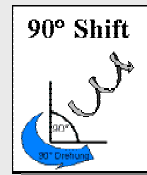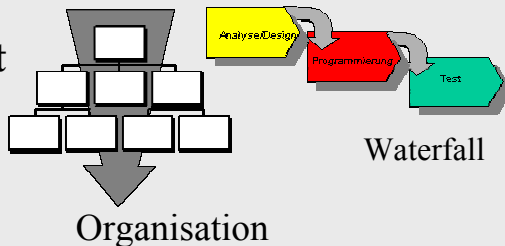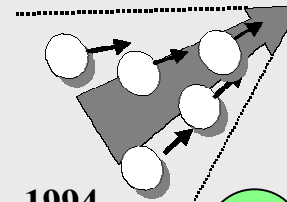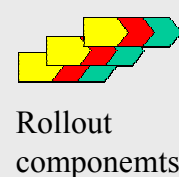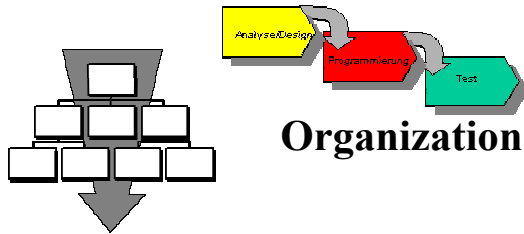| | New Challenges, new and more opportunities | |
|---|---|---|
| 1. Vertical Information | New Challenges, new and more opportunities | 1. Focus on big picture (horizontal information) |
| 2. Working in Cascades | Conflict because: Cascade separation became obsolete | 2. Working in parallel |
| 3. Interaction on interface documents only | New procedures and working-styles were needed | 3. Flowteam cooperation |
| 4. No responsibility for customer/added value | Customer driven development became possible | 4. Personal responsibility for rollout components/services |

**From
linear Project
Management
(-paradigm)**

Organisation   Waterfall

90° Shift

Rollout componemts

1994
Flow Team

**To
OO Project
Management
(-paradigm)**

**3**   **4**

# OO – Project Management Experiences

TM

**mueller & friends**
NEUE REALITÄTEN SCHAFFEN

> A sucessful realization of projects using OO-technology brings new opportunities/challenges. It requires not only OO methodology/skills, but also a new working/cooperation style: the 90° Shift
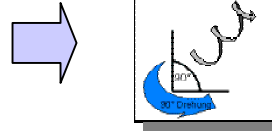
**linear Project Management Paradigm**
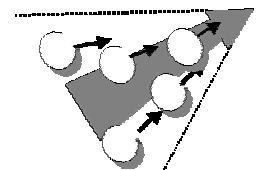
**OO discovered pioneering 1990**
**- From sequential to OO**

**OO Project Management (-paradigm)**

Analyse/Design
Programmierung
Test

**Organization**
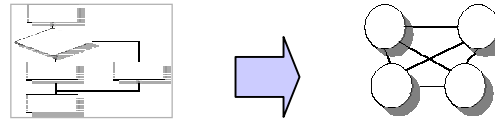
**- Traditional working style was not sufficient**

90° Shift

**Flow Team**

| linear Project Management | New Challenges | OO Project Management |
|---|---|---|
| 1. Vertical Information | New Challenges, new and more opportunities | 1. Focus on big picture (horizontal information) |
| 2. Working in Cascades | Conflict because: Cascade separation became obsolete | 2. Working in parallel |
| 3. Interaction on interface documents only | New procedures and working-styles were needed | 3. Flowteam cooperation |
| 4. No responsibility for customer/added value | Customer driven development became possible | 4. Personal responsibility for rollout components/services |