

# 6. Usability of Design Patterns

1

Prof. Dr. U. Aßmann  
Chair for Software Engineering  
Faculty of Computer Science  
Technische Universität Dresden  
13-0.1, 11/16/13

- 1) Using and writing patterns in companies
- 2) Successes of patterns

Design Patterns and Frameworks, © Prof. Uwe Aßmann



## 6.1 Using and Writing Patterns in Companies

3

Design Patterns and Frameworks, © Prof. Uwe Aßmann



## Literature (To Be Read)

2

- ▶ K. Beck, J. Coplien, R. Crocker, L. Dominick, G. Meszaros, F. Paulisch, J. Vlissides. Industrial Experience with Design Patterns Int. Conference on Software Engineering (ICSE) 1996
  - Beck First Class Software (consultancy)
  - G. Meszaros, BNR/NorTel (telecom)
  - Paulish & Dominick, Siemens
  - Crocker Motorola
  - Coplien ATT
  - Vlissides IBM

Prof. Uwe Aßmann, Design Patterns and Frameworks



## Patterns May Be Domain-Specific

4

- ▶ Telecom domain (Coplien, Meszaros)
  - Process and organizational patterns are very useful in larger teams
- ▶ Business domain
  - Banking
  - Administrative systems
- ▶ Problem domain vs solution domain
  - Patterns can be written for both of them
- ▶ How to come to these domain-specific patterns?
  - Solution: **Experience Factory**
    - Write them yourself, for your own company
    - Building a catalogue of domain-specific or company-specific design patterns
    - And record them in an Experience Factory

Prof. Uwe Aßmann, Design Patterns and Frameworks



## Pattern Writing is Hard

- 5
- ▶ Mesczeros identified three groups of pattern users in his company
    - People who are able to describe them (**pattern gurus**)
    - People who can recognize but not describe them (**pattern users**)
    - People who are oblivious about patterns (**pattern ignorants**)
  - ▶ He observed that only a small percentage of people can write patterns
    - The distinction may arise from people focussing on different things:
    - On similarities as opposed to differences between things

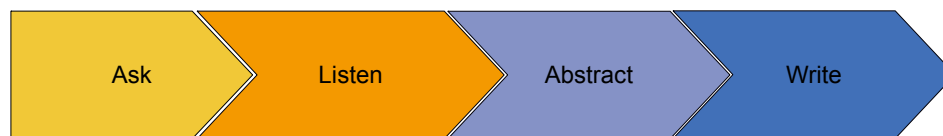


## Pattern Mining

- 6
- ▶ Patterns should be mined in interviews of domain specialists (Paulisch, Coplien)
  - ▶ The **pattern miner** (pattern writer) should refine and polish the pattern
    - And then go out to the domain experts again
    - About 3 interviews are necessary (Paulisch)
  - ▶ Sanity check:
    - The prototypical patterns should be presented to other groups that have not been involved in the process
  - ▶ Paulisch used hypertext to publish the pattern catalogue

## Pattern Miners are Internal Consultants

- 7
- ▶ It can be quite useful if pattern miners are not involved in the product groups,
    - since unconscious knowledge might exist in the group that can better be reflected from an outsider
    - They often do not *know* what they have done or cannot explain it
    - They often contradict each other in their assertions
    - Product group members are often so busy with their day-to-day work, that pattern mining does not work for them
    - They also need to learn how to write patterns
  - ▶ Often, they reengineer the design decisions and their rationale, because they have been forgotten



## Good Questions for Pattern Mining (Vlissides)

- 8
- ▶ Why did you design this way? [Rationale, Motivation]
  - ▶ Is what seems to be complexity here really worthwhile? [Consequences]
  - ▶ What are your assumptions? [Rationale]
  - ▶ Why are your assumptions realistic? [Forces]
  - ▶ What happens 6 months from now when I need new feature F? [extensibility, variability]
  - ▶ Hint: Ask these questions yourself, if you write a pattern

## Success Criteria for Pattern Catalogues

- 9
- ▶ Domain-specific pattern catalogues seem to be successful if
    - they consist of a small catalogue of low-level patterns (may be idioms)
      - If the catalogue has more than 30 entries, tool support is desired
      - (The GOF catalogue has about 30 entries!)
    - there is a single architectural (coarse-grain) pattern that describes the structure of the products
      - Such as pipe-and-filter style
  - ▶ A pattern catalogue needs a *pattern mentor*
    - That promotes the patterns within an organization
    - A master of the Experience Factory
    - This helps the organization to accept patterns



## 6.2 Successes of Design Patterns

11



## Success Criteria for Single Patterns

- 10
- ▶ They must be compact
  - ▶ They must be mined from working designs
  - ▶ They must be mined from “best practice”
  - ▶ They need not be object-oriented



## Design Patterns Improve Communication

- 12
- ▶ .... everyone experiences....



## ... But Measuring Impact is Hard

- 13
- ▶ Communication is clearly simplified
  - ▶ Programmers can master more complex designs
  - ▶ Vlissides reports that he started to require that consulted groups read GOF
    - After this, the groups had a much better understanding of what was going on
  - ▶ Engineers forget that they talk in patterns, after they have learned about them
    - Write me an email in 2 years from now, if you remember this statement
    - I will put you up on the courses home page
    - ... and may be invite you as a guest lecturer



## The End

15



## Mining Forgotten Requirements

- 14
- ▶ A pattern can help to discover *forgotten requirements*
    - A design has a rationale from the requirements
    - If a pattern can be matched only partially in a design, this may indicate that some requirements were uncovered
    - Hence, reiterate on the requirements document
  - ▶ Example:
    - A Mediator is discovered in a design
    - Usually, a Mediator enables dynamic reconfiguration of communication
    - If this requirement has not been fixed, discuss with the client whether he needs it
    - If not, you may simplify the design
    - If yes, you argue for more money :-)

