Didactical Design Patterns for Mathematics: How to apply them?

Paul Libbrecht
CERMAT
PH Karlsruhe

Marc Zimmerman
PH Ludwigsburg
Didactical Design Patterns

• typical sequence of steps
• typical configurations
• \ldots good practice in math learning
• short texts
• the issues, the tension
• the solution, the rationale
• relatives
Didactical Design Patterns

• typical sequence of steps
• typical configurations
• ... good practice in math learning
• short texts
• the issues, the tension
• the solution, the rationale
• relatives

technology...
feedback...
hint...
representation...
... on demand

Name it.
Related Works

- Software Design Patterns
  - elementary, named
  - simple, illustrated, accessible
  - allows one to understand others’ code
- several didactical design patterns collections
- book upcoming, one early project, ...
Example Pattern: \textit{Help on Demand}

\textbf{Problem / Challenges / Motivation}
Students differ in their computer fluency and in their previous knowledge regarding software used in the math tutorials. All students should be enabled to use the tools for problem solving without spending too much effort in learning the tools.

\textbf{Forces}
Often the software is taught before students really need it. This may result in lengthy demonstrations of complex procedures. But normally some students already know how to use the tools. For those the demonstration is unnecessary and boring. For novices, it can be too much information in advance, and afterwards they don’t remember what to do.
Example Pattern: Help on Demand

Problem / Challenges / Motivation
Students differ in their computer fluency and in their previous knowledge regarding software used in the math tutorials. All students should be enabled to use the tools for problem solving without spending too much effort in learning the tools.

Forces
Often the software is taught before students really need it. This may result in lengthy demonstrations of complex procedures. But normally some students already know how to use the tools. For those the demonstration is unnecessary and boring. For novices, it can be too much information in advance, and afterwards they don't remember what to do.

Solution
Help on technology must be right at hand when the information is needed. In the first place, peer support should be fostered. Students can help each other when working together at the same computer. Tutors may also give hints on how to use the software. In cases where students use the software alone at home, or where all students are likely to be novices regarding a specific tool, instructions should be given where the core procedures are described. Procedures explained should be analog to those needed by the students and not step-by-step explanations of how to solve the given problem. The worked example should be a mathematically simpler example where all the essential procedures are mentioned that can be used in the current problem situation as well.
Example Pattern: Help on Demand

Problem / Challenges / Motivation

Students differ in their computer fluency and in their previous knowledge regarding software used in the math tutorials. All students should be enabled to use the tools for problem solving without spending too much effort in learning the tools.

Forces

Often the software is taught before students really need it. This may result in lengthy demonstrations of complex procedures. But normally some students already know how to use the tools. For those the demonstration is unnecessary and boring. For novices, it can be too much information in advance, and afterwards they don't remember what to do.

Solution

Help on technology must be right at hand when the information is needed. In the first place, peer support should be fostered. Students can help each other when working together at the same computer. Tutors may also give hints on how to use the software. In cases where students use the software alone at home, or where all students are likely to be novices regarding a specific tool, instructions should be given where the core procedures are described. Procedures explained should be analog to those needed by the students and not step-by-step explanations of how to solve the given problem. The worked example should be a mathematically simpler example where all the essential procedures are mentioned that can be used in the current problem situation as well.

http://www.sail-m.de/sail-m/HelpOD_en
Using Patterns: Teachers

• teachers *design* their sessions
• patterns provide guide when designing
  • typical sequence to remember
• checks when reviewing
• relationships can help deepen particular aspects
Example Usage by Teacher

• *Help on demand* can guide tool deployment:
  • introduce learning tool in class by demo
  • short, with relevant manipulation
  • leave other introductions *on-demand*
  • ... stimulate peer support
  • ... which document should be available?
Patterns applied to learners

• recognizable rythm
• can help to play together
• e.g. with Representation on Demand

• know that they may expect more representations if thinkable
Using Pattern: Software

• Objective: pattern must be applicable
• At software design time?
  • For UI functions
  • To guide and name features
• In a Handbook or video-guide?
• For user: when using the software?
Using Pattern: Software

- Objective: pattern must be applicable
- At software design time?
  - For UI functions
  - To guide and name features
- In a Handbook or video-guide?
- For user: when using the software?

think: paradigm
(e.g. copy-and-paste, hyperlink)
When it was missing...

- ActiveMath’s copy-and-paste
- not exactly CnP...
  ...drag-and-drop
- needed to explain it and demonstrate it
- unintuitive, forgotten, unused
Patterns for Software Design

- got a learning tool
- want to include some feedback
- ... at step 2?
- Feedback-on-demand says differently:
  - let students act and request feedback
  - \( \Rightarrow \) changes how the feedback is computed
  - \( \Rightarrow \) changes how the input is made
Patterns for Software Design

- got a learning tool
- want to include some feedback
- ... at step 2?
- Feedback-on-demand says differently:
  - let students act and request feedback
  - $\Rightarrow$ changes how the feedback is computed
  - $\Rightarrow$ changes how the input is made

rigid dialog... a problem
full model... probably needed

what? • others? • for teachers • for software

CERMAT
Centre for Educational Research in Mathematics and Technology
Patterns for User Interface

- e.g. for HelpOD:
  Users need to know when to look for help

- read manual before start?

- read manual when blocked?

- how to transpose help?

- help on demand answers a bit

- users would recognize the pattern
Patterns as common language

- didactical vs computer literature
- barely know each other
  - patterns an exception of understandability
- Richard Noss at Journées EducTice 2009: *need to be able to talk to each other!*
- patterns provides words
Conclusion

• inviting your reactions:
Conclusion

• inviting your reactions:

• Better ways to apply patterns?
Conclusion

• inviting your reactions:
  • Better ways to apply patterns?
  • Pedagogical theories you followed in implementing?
Conclusion

• inviting your reactions:
  • Better ways to apply patterns?
  • Pedagogical theories you followed in implementing?
  • What common-language between research communities?
Conclusion

• inviting your reactions:
• Better ways to apply patterns?
• Pedagogical theories you followed in implementing?
• What common-language between research communities?
• Other patterns known?
• further slides
Example Missing Pattern

- multilingual transfer
  - e.g. dutch student learns in English
- what are dangers?
- what are practices a learner can remember?
- what helps?