

ETHOS: A Pedagogical Design Pattern

This paper is organised as follows: First, I introduce the current format for pedagogical design patterns and some modifications considered necessary during shepherding by Paul Dyson (thanks for your helpful suggestions, Paul!). Second, the ETHOS pattern is described in full detail. Third, I give some background information regarding the pedagogical patterns movement.

1 Pattern Format

From the feedback acquired from participants at previous pedagogical patterns workshops and from those who have provided feedback in other ways, the current format has been drafted and is still open for review. Taken from the homepage of the pedagogical patterns project¹, the format contains the following sections:

Section	Contents or Purpose
Name	pattern name
Intent	what the instructor wants to teach, or avoid, or ...
Idea	how this pattern can achieve the Intent
Motivation	describes why the instructor wants to achieve the Intent with this pattern
Applicability	circumstances in which this pattern is most useful, in the opinion of the pattern author and others who have used the pattern
Contraindications	when not to use the pattern
Structure	description of the pattern's structure in time (and space), e.g. the sequence of instructions or students' activities
Consequences	what has been seen to occur when this pattern has been used
Issues to Consider	the pragmatics of using the pattern
Cultural Dependencies	issues which may make this pattern less useful for a particular culture
Resources Needed	the resources which are needed to implement the pattern
Example Instances of This Pattern	specific instances in which the pattern has been successfully used
Related Patterns	other patterns which are related to this pattern

In general, I have tried to adhere to this format. However, some modifications have been regarded necessary: First, the original pattern format seems to spread problem, context, and forces over the first six sections. What is needed here is a clear *problem statement* somewhere near the beginning of the pattern description. Subverting the original pattern format somewhat, I've inserted the problem statement in the Intent section. Second, some pattern

¹ <http://www.cs.unca.edu/~manns/oopsla.html>

sections have been split into how it affects and/or aids the instructor and the learner. These sections are subdivided into “instructor`s perspective” and “learner`s perspective”.

2 Pattern

Name

ETHOS

Intent

We teach many students that will take managerial-type positions. These positions will demand that the student has some knowledge of a wide range of subjects. Hence we need a course that covers a wide range in some detail. This is hard to do such that all topics are covered. Some areas may be overlooked, some areas may be outside the lecturer`s immediate sphere of experience.

From the instructor`s perspective: there is a need to structure a course of lectures, or the chapters of the manuscript that go with it to maintain thematic coherency.

From the learner`s perspective: there is a need to comprehend a manifold subject matter in its entirety.

Idea

To master both coherency and variety of a great many facets of a wide-ranging subject, you need a *mnemonic* and an *organisational* aid: on the one hand, to remember all the areas you should cover and on the other hand to organise your material such that these areas are covered.

Motivation

University-bred engineers will mostly take managerial-type positions where *specialism* is of secondary value. Any technology transfer, for example, is a decision-making process that requires – besides a sound technical knowledge – a broad understanding of the economic, social, and organisational implications of the new technology. For this clientele, you want to draft an introductory course about, say, object-oriented software engineering. The subject matter is to be taught in its whole spectrum (*grand tour*) avoiding pedantic and boring lectures. You choose aspects you want to elaborate upon for a longer time. In a way, you are

looking for some pedagogical "spotlights" to focus the learner's attention to the principal aspects of the teaching matter. Your motive can be outlined as follows:

You want:

- ! to add variety to your lectures;
- ! to take account of all important aspects;
- ! to arrange lectures along a line supporting cohesive learning;
- ! to widen the learner's perspective to interdisciplinary aspects.

Applicability

The subject matter should be one of engineering, e.g. a method for analysing, designing, and implementing large-scale systems. It is important that economic and technical aspects are present.

Check if the following points apply to your problem:

- ? the subject matter involves a *paradigm* [3], that is (1) a "higher principle" or way of thinking, typical of a certain discipline, but which cannot be clearly formulated and which manifests itself by examples, and (2) a "disciplinary matrix" of opinions and values holding together a scientific community;
- ? you want to hold an introductory course, i.e. you don't intent to go into many details.

The pattern also applies to structuring a textbook or manuscript about a wide-ranging subject.

Contraindications

From the instructor's perspective: by favouring "breadth before depth", ETHOS could lead into a mode whereby abstracting and generalising the subject into key concepts has the side-effect of over-simplifying a rich and deep technology full of subtleties.

From the learner's perspective: this could lead to underestimating the intellectual effort needed to master object-oriented concepts.

To avoid this risk, you should make use of Alan O'Callaghan's pedagogical pattern *Mission Impossible* (see footnote 1). It will present the learner with a problem that the basic concepts do not appear sufficient to solve, in order to provoke further questioning, exploration, and self-study.

Most engineering subjects are by their very nature technically biased. So keep in mind that ETHOS is very particular to broad courses featuring *non*-technical issues on an equal footing with technical ones.

Structure

ETHOS reminds you that a solution to an engineer's problem commonly comprises Economic, Technical, Human, Organisational, and Social aspects. As an acronym, ETHOS is strictly sequential, thus a test whether all relevant aspects are taken into account. The pattern's basic structure follows its initials:

- E : economic,
- T : technical,
- H : human,
- O : organisational, and
- S : social aspects.

If several topics apply to the same aspect, subdivide the structure's individual elements by indexing, e.g. T_1, T_2, \dots, T_n (see **Example Instances of This Pattern**).

Consequences

From the instructor's perspective:

- 👉 ETHOS allows kaleidoscopic lectures, i.e. being arranged in a colourful succession, each one a self-contained unit within a wide spectrum;
- 👉 it favours breadth before depth, general knowledge before specialised knowledge;
- 👉 it provides a general framework where new topics (current developments, etc.) can easily be integrated without changing the lecture's basic structure.

From the learner's perspective:

- 👉 ETHOS supports cohesive learning;
- 👉 it permits continuous learning: even if some lectures have been missed, the student can follow the others, thus, ETHOS helps to encapsulate a lecture as a learning unit; with the manuscript being similarly structured, the congruence between lecture and manuscript will be hold, i.e. arbitrary cuts and context switches can be avoided;
- 👉 as a breadth-before-depth approach, ETHOS might lull the learner into a false sense of competence. Hands-on experiences and pragmatic subtleties of the subject could be easily underestimated (see **Contraindications**).

Issues to Consider

From the instructor's perspective: ETHOS presupposes a deep analysis of both the subject matter and its didactic; the importance of and relationship between individual ETHOS aspects have to be made explicit. Try to visualise the ETHOS pattern with the help of typography and layout. Integrate the ETHOS aspects into the general structure of your course, don't let them stand for their own. For a concrete example, topics and contents of a course on object-oriented systems design are given in **Example Instances of This Pattern**.

From the learner's perspective: the nature of ETHOS is to structure the wide spectrum of a subject in order not to lose an important aspect. Any full-scale education or training in object technology will require follow-up courses, accompanying lab-based exercises, group projects, etc.

Cultural Dependencies

None.

Resources Needed

None.

Example Instances of This Pattern

Since 1994, the ETHOS pattern has been successfully applied to a half-year lecture on "Object-Oriented Systems Design" annually given at the University of Siegen, Germany. The lecture is accompanied by lab-based exercises and a student project team. The manuscript of the lecture follows the same pattern.

Topics of lectures ²	
1.	Overview: ECBS "Engineering of Computer-Based Systems"
2.	Paradigms of Design in Computer Science
3.	Complexity of Designing
4.	Mastering Design Complexity
5.	E Industrialised Software
6.	T₁ OOX: Abstracting – Partitioning – Communicating
7.	T₂ OOAD: Foundations of Analysis and Design Methods
8.	T₃ OOP: Defining and Categorising Object-Oriented Programming Languages
9.	H₁ Cognitive Aspects: Designing as Human Problem Solving
10.	H₂ On the Tripartition of an Object: Structure – Behaviour – Constraints
11.	O Management Aspects: Technology Transfer and Project Organisation
12.	S A Science of Design
13.	Looking Back: FAQ and Course Evaluation

² <http://www.ti.et-inf.uni-siegen.de/courses/oos/oos.html>

Contents of Manuscript [6]

- 1 Paradigm Shifts in Computer Science
Paradigm Shifts in the Large: Kuhn's Thesis.
Paradigm Shifts in the Small: A Time without a Method •
The Art of Programming vs. Software Engineering • The Human Factor • SA/SD vs. OOx.
Object-Oriented World Models: Executable Models • Scenario of Object-Oriented Designing.
- 2 The Problem: Mastering Design Complexity
"No Silver Bullet": The Descriptive Nature of Complexity • Complexity and its Dimensions •
Design Complexity.
"Hopes for the Silver": The Magical Number Seven • The Architecture of Complexity •
Divide and Conquer.
ETHOS Aspects of the Object Paradigm
- 3 **E for Economic**
On the Way to Industrialised Software: The Principle of Locality • Software Reuse •
Standard Class Libraries.
Competitive Pressure: Productive Software Development • Software Quality.
- 4 **T for Technical**
Object-Oriented Concepts: Abstracting • Partitioning • Communicating.
Object-Oriented Applications: Analysis and Design • Programming Languages.
- 5 **H for Human**
On the Psychology of Object-Oriented Concepts: Designing as Human Problem Solving •
Cognitive Structures • Scheme and Correction • The Contribution of the Object Paradigm.
On the Philosophy of Object-Oriented Concepts: The World of Ontology •
An Ontological Object Model.
- 6 **O for Organizational**
Technology Transfer: Questions about the Technology •
Questions about the Interface between Client and Designer •
Questions about the Design Process •
Questions about Project Control • Questions about Staff Management.
Aspects of Management: Homomorphism between Process and Product • Lean Management •
Object Management.
- 7 **S for Social**
"The Science of Design": Creating the Artificial • Curriculum of a Science of Design •
The Contribution of the Object Paradigm.
Architectural Designing: Ideals of a Generalised Discipline of Design •
Good Designing from an Architectural Perspective • The Contribution of The Object Paradigm.
An Ontology of Design: Deep Structure: States – Events – Laws •
Good Designing from an Ontological Perspective • The Contribution of the Object Paradigm.
- A Excursions: Imagery • Classification • Inheritance vs. Encapsulation • "The Treaty of Orlando" •
The Terminology of the Object Management Group.
- B The Object-Oriented Method by Example: Analysis • Design • Programming.
- C Tables: Literature • Persons • Glossary • Abbreviations • Index.

The ETHOS pattern has been recommended in a textbook on presentation techniques [8]. It was also used to condense the range of benefits of design patterns in an introductory article for a scientific journal [5].

Related Patterns

In principle, any instance of the Alexanderian pattern form, i.e. Problem – Context – Forces – Solution, is a likely candidate for structuring a course or manuscript. Experiences of applying the pattern form in this way are yet to be published.

3 Background

Design patterns and pattern languages are celebrated as a new kind of literature in the software engineering community: they help to capture, communicate, and reuse design experience in a clear and concise way. It is this *economy of expression* that have inspired instructors to use the pattern form to condense the body of knowledge in educating and training people in object technology [1, 2, 4]. Like software engineering, pedagogy, too, is regarded as a *design* discipline: courses, lectures, exercises, and labs have all to be designed. As most instructors of object technology haven't had a specific pedagogical education or training, they look for an efficient vehicle to transfer teaching experience. Pedagogical design patterns will help here. A first internet-based project has started and about 50 patterns have already been collected (see footnote 1). However, none of them has yet been reviewed in a PLoP-styled Writer's Workshop.

Since 1996, a number of pedagogical patterns workshops have been run: at ECOOP '96 in Linz, Austria, at TOOLS USA '96 in Santa Barbara, California, at OOPSLA '96 in San Jose, California. In addition, potential patterns were collected during a fourth workshop, held at the OT '97 conference in Oxford, England. There was also a large amount of interest generated after a short Educators' Symposium presentation and Birds-of-a-Feather session held at the OOPSLA '97 conference. What is still missing is the creative feedback of a *Writer's Workshop* as practised at PLoP conferences. The author of this paper is willing to incite a similar kind of workshop series for instructors of object technology in academia and industry called (Euro)PLoT: Pattern Languages of Teaching [7]. Most people attending PLoP conferences are instructors or trainers of object technology. So, EuroPLoP '98 would be a good start to submit a pedagogical design pattern to the spirit of the new writing culture.

References

- [1] Anthony, Dana L. G.: Patterns for Classroom Education. Pattern Languages of Program Design 2. Reading: Addison-Wesley 1996.
- [2] Brito e Abreu, Fernando: Pedagogical Patterns: Picking Up the Design Patterns Approach. Object Expert, March-April 1997, pp. 37-41.
- [3] Kuhn, Thomas S.: The Structure of Scientific Revolutions. University of Chicago 1970.
- [4] Lilly, Susan: Patterns for Pedagogy. Object Magazine, January 1996, pp. 93-96.
- [5] Quibeldey-Cirkel, Klaus: Hot Topic: Design Patterns. Informatik-Spektrum 19 (1996), pp. 326-327. (in German)
- [6] ... The Object Paradigm in Computer Science. Stuttgart: Teubner 1994. (in German)
- [7] ... Transferring Teaching Experience with Patterns: A Contribution to a Pedagogy of Computer Science. Proc. of "GI-Fachtagung: Informatik und Ausbildung", Stuttgart: Springer 1998, pp. 33-42.
- [8] Thiele, Albert: Presenting Successfully: Presentation Techniques for Managers. Düsseldorf: VDI 1991. (in German)