

# Object-Oriented Framework for Teaching Introductory Programming

## MASTER THESIS Project Plan

Rolf Bruderer

12th October 2004

[http://se.inf.ethz.ch/projects/rolf\\_bruderer](http://se.inf.ethz.ch/projects/rolf_bruderer)

**Project Period:** September 20th 2004 - March 19th 2005

**Student:** Rolf Bruderer ([bruderer@computerscience.ch](mailto:bruderer@computerscience.ch))

**Student-No:** 99-920-621

**Supervising Assistants:** Michela Pedroni, Till G. Bay

**Supervising Professor:** Bertrand Meyer

# 1 Project Description

## 1.1 Overview

In winter 2003/2004 a wind of change blew through the Department of Computer Science at ETH Zurich. For the first time Professor Bertrand Meyer held the course “Introduction to Programming” for first semester students. He redesigned the previous course and used the “Inverted Curriculum” approach to teach introductory programming [1, 2]. In this approach students learn step-by-step to grow from consumers to producers of a big object-oriented system instead of just starting with little boring programs like “Hello World”.

From the outset, the students are provided with an object-oriented library, called TRAFFIC [3]. It is a library for modelling a city and its public transportation system. By using TRAFFIC, students are able to produce interesting applications right from the start, just by writing a few lines of code. As they progress, the students learn to build more and more elaborate programs. In the course of the exercises they even start to understand the library from the inside. This is why this approach is also known as “progressive opening of black boxes” or as “outside-in approach”. In the end, the students should even be able to produce such libraries by themselves.

For the winter semester 2004/2005 and the second run of this course, the whole accompanying material has been improved a lot. The students will get an impressive software package, that does not only include the TRAFFIC library, but also an example program based on this library. It is a fun computer game called FLAT\_HUNT [4]. The students are expected to have a look at this example piece of software and to change and even extend it in some of the exercises. Thereby they are not only supposed to have much more fun, but also to understand and to construct real object-oriented programs right from the start.

Although the whole course material has been improved, still a lot of work remains to be done. The main goal of this thesis is to redesign the whole framework in a way that it is easier to understand, to use and to extend, even for students that have no or just little experience in programming. In the end, it should be an example of a perfectly designed object-oriented piece of software that is very easy to use. To achieve this goal we will not only have to improve the design of the whole framework. We will also need to develop the documentation and other materials provided to the students very carefully.

Furthermore we intend to improve the visualization part of the framework by using the open source and platform-independent multimedia library ESDL[5, 6]. With this library the students should be able to easily produce software that is quite impressive even for the “Nintendo generation” [7]. My work will include to continue the development of the ESDL library and to extend it to our needs.

Another part of this project is to do an evaluation of the whole approach for teaching introductory programming. The idea is to develop a test procedure to survey how successful the course is. This test should verify how good the students comprehended the basic concepts of object-oriented programming and how good they can apply them. We intend to use this test not only to evaluate the success of the course here at ETH Zurich. We also want to use it to compare with courses at other universities that are teaching with a different approach.

## 1.2 Motivation

Introducing first year students to the art of programming seems to be one of the most important topics for computer science departments. And I think, it is also the most difficult part. Many of the students have no or only little experience in software construction. A survey in the first lecture of last winter semester showed that more than 20% of the first year students had no programming experience at all [4]. On the other hand, there are a lot of students that already program quite well. It is very difficult to handle this diversity in a way that those who have no experience still have a chance to follow and the others do not get bored.

When I started studying here at ETH Zurich, the introductory programming course did not pay attention to this problem at all. Most of the examples discussed in the lectures were so hard to understand, that even

students with programming experience had difficulties to follow. An overall concept of the course was not recognizable. The exercises were just not understandable when you were a novice in computer science and the development environment of Oberon was very hard to use.

When I heard about the “Inverted Curriculum” approach [1] and the way things were changing here at ETH Zurich [2, 4], I was excited. I think, it is a very good and important thing to design a course for introductory programming very carefully. This course is the first impression students get of software engineering, and it might influence their whole life. Therefore I am very proud to contribute and I hope that a lot of students will profit from my work.

I am looking forward to a very interesting work here in Professor Meyer’s research group at the Chair of Software Engineering. I am very glad that I can do a master thesis that covers most of the topics I focused on during my studies, which are:

- Software Engineering
- Computer Graphics
- Didactics

Furthermore I will not just work all alone in the field of “Teaching Introductory Programming”. Sibylle Aregger will do a part of the redesign of the TRAFFIC library as a semester thesis. Together with her and of course with my supervisors Michela Pedroni and Till Bay, we form a little project team. I am sure that this team will be a great environment to learn a lot from each other, not only concerning technical aspects but also social skills, like teamwork. We will work together to try to achieve the goal of an almost perfect framework for teaching introductory programming.

### 1.3 Scope Of The Work

My master thesis consists of the following four parts:

- **ESDL Library:** For the new visualization part of the framework we will use the platform-independent ESDL multimedia-library. It has to be improved and it will be my work to enhance the ESDL library for our needs.
- **Object-Oriented Framework:** We will redesign the TRAFFIC library and the FLAT\_HUNT software in such a way that it is easy to understand and to use for first year students, even for those who have no programming experience. In parallel, we will improve the visualization part of the framework by using the ESDL library.
- **Documentation Material for Students:** We have to improve the documentation of the framework and to produce some additional course material that should guide the students to find their way through the whole framework.
- **Evaluation of the “Inverted Curriculum”:** We want to develop a test procedure to survey how well the students learned the basic programming skills. This test has to be independent from the “Inverted Curriculum” approach and also from the Eiffel language, because we want to apply it on first year students of different universities. Thus, we intend to compare the effects of our approach for teaching introductory programming to approaches used at other universities.

## 1.4 Intended Results

I intend to produce the following results during my master thesis:

- **Object-Oriented Framework:** An easy to use object-oriented software framework consisting of the TRAFFIC and ESDL libraries and the FLAT\_HUNT application.
- **Documentation:** Well documented class interfaces and additional material to guide the students through the programming exercises.
- **Evaluation:** A test procedure to survey how well students learned the basic programming skills. We intend to use it to evaluate the success of our object-oriented introductory programming course. If possible, the test will be applied and evaluated on first year students of different universities. It is not yet clear if this goal may be accomplished, because we do not know yet if it is possible to develop such a test and how much effort is needed for this. Thus, this result is optional. But at least a clarification about the feasibility of such a survey should be part of my report.
- **Report:** A documentation of all the work that I have done during my thesis.

## 2 Background Material

### 2.1 Reading List

- Bertrand Meyer, *Towards an Object-Oriented Curriculum*, in Journal of Object-Oriented Programming, vol. 6, no. 2, May 1993, pages 76-81.  
[www.inf.ethz.ch/~meyer/publications/tools/curriculum.pdf](http://www.inf.ethz.ch/~meyer/publications/tools/curriculum.pdf)
- Bertrand Meyer, *Object-Oriented Software Construction*, 2nd edition, Prentice Hall PTR, 1997.
- Bertrand Meyer, *Software Engineering in the Academy*, in Computer (IEEE), vol. 34, no. 5, May 2001, pages 28-35.  
<http://www.inf.ethz.ch/~meyer/publications/computer/academy.pdf>
- Bertrand Meyer, *The Outside-In Method of Teaching Introductory Programming*, 2003.  
<http://www.inf.ethz.ch/~meyer/publications/teaching/teaching-psi.pdf>
- Bertrand Meyer, *Touch of Class: Learning to Program Well - With Object Technology, Design by Contract, and Steps to Software Engineering*, to be published.  
<http://se.inf.ethz.ch/touch>
- Michela Pedroni, *Teaching Introductory Programming with the Inverted Curriculum Approach*, ETH Zurich, 2003.  
[http://se.inf.ethz.ch/projects/michela\\_pedroni](http://se.inf.ethz.ch/projects/michela_pedroni)
- Marcel Kessler, *Exercise Design for Introductory Programming: "Learn-by-doing" basic O-O-concepts using Inverted Curriculum*, ETH Zurich, 2004.  
[http://se.inf.ethz.ch/projects/marcel\\_kessler](http://se.inf.ethz.ch/projects/marcel_kessler)
- Till G. Bay, *Eiffel SDL Multimedia Library (ESDL)*, ETH Zurich, 2003.  
[http://se.inf.ethz.ch/projects/till\\_bay](http://se.inf.ethz.ch/projects/till_bay)
- Benno Baumgartner, *ESDL - Eiffel Simple Direct Media Library*, ETH Zurich, 2004.  
[http://se.inf.ethz.ch/projects/benno\\_baumgartner](http://se.inf.ethz.ch/projects/benno_baumgartner)
- Mark Guzdial, Elliot Soloway, *Teaching the Nintendo Generation to Program*, in Communications of the ACM, vol. 45, no. 4, April 2002, pages 17-21.

- Karl Frey, Angela Frey-Eiling, *Allgemeine Didaktik*, ETH Zurich, 2001.
- Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley Professional, 1995.

## 3 Project Management

### 3.1 Objectives and Priorities

The main objectives of this work ordered by priorities are as follows:

1. Develop material for guiding students through the programming exercises with the existing framework.
2. Redesign the object-oriented framework such that it is easy to understand and to use for first year students.
3. Improve the visualization part of the framework by using the ESDL library.
4. Enhance the ESDL library to our needs.
5. Develop a test procedure to survey the success of object-oriented introductory programming courses or at least clarify how to do such a survey.
6. Evaluate and compare several introductory programming courses of different universities using our test procedure.
7. Improve and adopt the existing documentation material and exercises to the new framework.

Priority 1 is to develop additional documentation to guide the students. This has the highest priority because we want to use this material already in the coming semester that starts on October 19th 2004.

The following priorities 2, 3, and 4 will be done in parallel, because the redesign of the framework and the improvement of the visualization part go hand in hand. Furthermore, we have to enhance the ESDL library for the visualization part.

The priorities 5, 6, and 7 may be done after the whole redesign of the framework. However, I intend to start at least with priority 5 in parallel with the redesign of the framework.

It might be that the time of my master thesis does not suffice to accomplish the last priorities perfectly. But there will still be enough time before the third run of the course in winter semester 2005/2006 for someone else to fulfill this.

### 3.2 Criteria for Success

#### 3.2.1 Design of the Framework

The most important point is that the students are happy with the new framework. To achieve this and to provide them with a good example of an object-oriented software, the framework has to be:

- easy to understand
- easy to use
- easy to remember
- sound
- exemplary object-oriented (reusable, extendable, generic, ...)

### 3.2.2 Quality of Code

Since the framework is also meant to be an exemplary piece of software for the students, the quality of the code is even more important than in any usual system. Therefore the following points are not only important but absolutely essential:

- style guidelines
- contracts (preconditions, postconditions, invariants)
- documentation of implementation and interfaces

### 3.2.3 Documentation

During this thesis some material to guide the students through their programming exercises will be developed. It is important that this documentation meets the following requirements:

- easy to understand
- complete
- useful

The same criteria will be applied to my final report.

## 3.3 Method of Work

As already mentioned in the motivation, I am not the only person working on this topic during my master thesis. Together with Sibylle Aregger doing a semester thesis and with our supervisors Michela Pedroni and Till Bay, we will work in a team. We will have weekly team meetings to discuss important design decisions.

During our work, we will continuously document our program code carefully. Our supervisors will give us helpful feedback about the code from time to time to ensure a good quality of code.

For the implementation, we will use *ISE EiffelStudio 5.5* and *Gobo Eiffel Library*. The documentation will be written with *LyX*.

## 4 Plan With Milestones

### 4.1 Project Steps

- Project Start: September 20 2004
- Existing framework studied and project plan written: October 8 2004
- Existing framework prepared for the second run of the course including additional material to guide the students: October 20 2004
- Background material read and decisions on framework design: November 3 2004
- Test procedure to survey the success of an introductory programming course developed (optional): January 12 2005
- Framework implemented: February 24 2005
- Report completed: March 18 2005
- Deadline: March 19 2005

## 4.2 Tentative Schedule

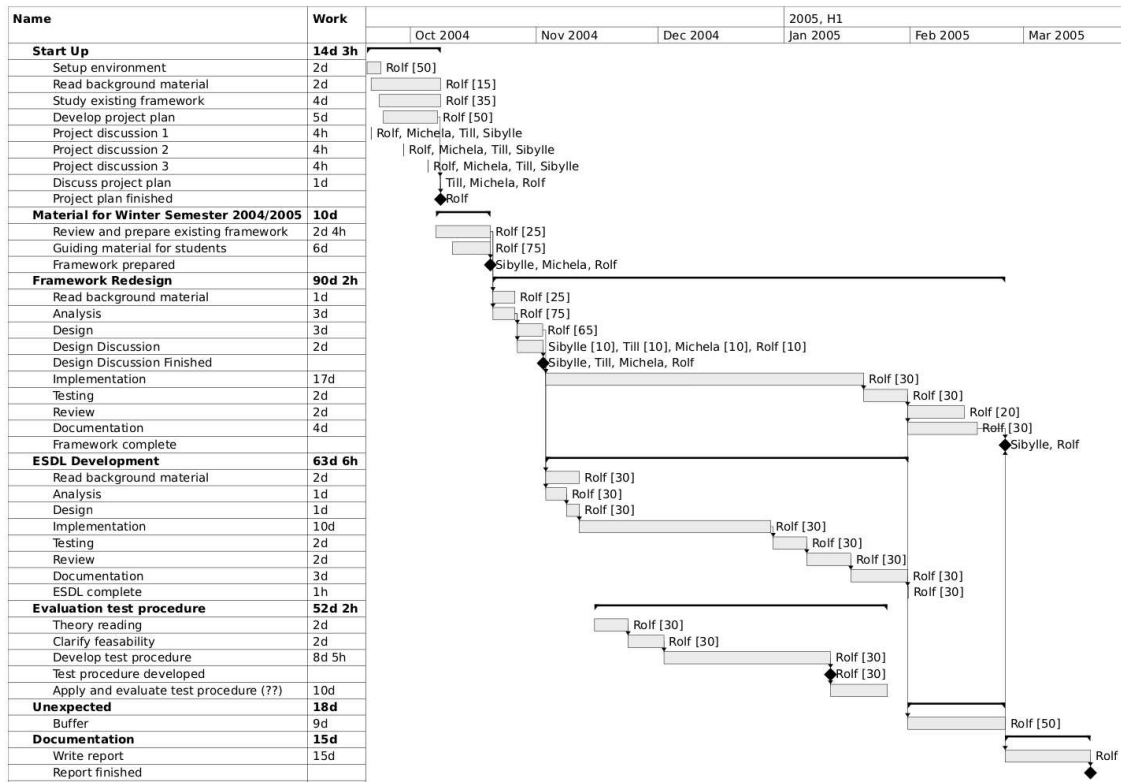


Figure 1: Tentative Schedule

## References

- [1] Bertrand Meyer. *The Outside-In Method of Teaching Introductory Programming*, 2003.  
<http://www.inf.ethz.ch/~meyer/publications/teaching/teaching-psi.pdf>.
- [2] Michela Pedroni. *Teaching Introductory Programming with the Inverted Curriculum Approach*. ETH Zurich, 2003.  
[http://se.inf.ethz.ch/projects/michela\\_pedroni](http://se.inf.ethz.ch/projects/michela_pedroni).
- [3] Bertrand Meyer. *Touch of Class: Learning to Program Well - With Object Technology, Design by Contract, and Steps to Software Engineering*. To be published.  
<http://se.inf.ethz.ch/touch>.
- [4] Marcel Kessler. *Exercise Design for Introductory Programming: "Learn-by-doing" basic O-O-Concepts using Inverted Curriculum*. ETH Zurich, 2004.  
[http://se.inf.ethz.ch/projects/marcel\\_kessler](http://se.inf.ethz.ch/projects/marcel_kessler).
- [5] Till G. Bay. *Eiffel SDL Multimedia Library (ESDL)*. ETH Zurich, 2003.  
[http://se.inf.ethz.ch/projects/till\\_bay](http://se.inf.ethz.ch/projects/till_bay).
- [6] Benno Baumgartner. *ESDL - Eiffel Simple Direct Media Library*. ETH Zurich, 2004.  
[http://se.inf.ethz.ch/projects/benno\\_baumgartner](http://se.inf.ethz.ch/projects/benno_baumgartner).
- [7] Mark Guzdial, Elliot Soloway. *Teaching the Nintendo Generation to Program*. *Communications of the ACM*, 45(4):17–21.