

# Redesign of the TRAFFIC Library

## SEMESTER THESIS Project Plan

Sibylle Aregger

12th November 2004

[http://se.inf.ethz.ch/projects/sibylle\\_aregger](http://se.inf.ethz.ch/projects/sibylle_aregger)

**Project Period:** Semester start - semester end

**Student:** Sibylle Aregger ([siaregge@student.ethz.ch](mailto:siaregge@student.ethz.ch))

**Student-No:** 99-905-564

**Supervising Assistant:** Michela Pedroni

**Supervising Professor:** Bertrand Meyer

# 1 Project Description

## 1.1 Overview

In winter 2003/2004, Professor Bertrand Meyer redesigned the course “Introduction to Programming” for first semester students at the Department of Computer Science at ETH Zurich. He introduced the approach called “Inverted Curriculum” [1, 2]. In this approach students start as consumers of a big object-oriented system to become producers. In contrary, the standard way of teaching introductory programming starts with small producer programs like “Hello World”. Here the focus lies on syntax and other basic language constructs like how a program file looks like and how to use/include libraries.

In the “Introduction to Programming” course, the students work on an object-oriented library called TRAFFIC. TRAFFIC models a city and its public transportation system. The combination of the Inverted Curriculum and the TRAFFIC library allows the students to write interesting programs right from the start. As Bertrand Meyer puts it:

Teaching is better than preaching, and if something is better than teaching it must be the demonstration, carried out by the students themselves, of the principles at work, producing “Wow” results.

The TRAFFIC library is supposed to build up on familiar constructs, containing interesting algorithms and data structure examples, support graphics and multimedia. The combination of the “Inverted Curriculum” and the TRAFFIC library will make learning how program not only efficient but fun for the students.

For the current semester, winter 2004/2005, the accompanying material of the course has been improved a lot. In addition to the TRAFFIC library itself the students will get an example program based on the TRAFFIC library. This program is called FLAT\_HUNT [3]. FLAT\_HUNT is a game similar to Scotland Yard. With FLAT\_HUNT, the students will get the opportunity to study and even change and extend this game application in some exercises.

Despite all these improvements there still remains a lot of work to do. The scope of this semester thesis will be to understand the implementation and design choices of the TRAFFIC library as it stands, and in addition to redesign it with the intention of making the whole system easy to understand, easy to use and easy to extend. All this work and analysis will be done while keeping in mind that this is a library used for introductory programming course where many students have no or little experience with programming. So the overall goal is to make this library exemplary object-oriented and easy to use. Therefore the documentation of the interface and the development are crucial.

## 1.2 Motivation

When I found out about the possibility of working on TRAFFIC as a semester thesis, I was excited[1, 2]. From my own experience I know how difficult introductory programming courses are for novice programmers. These problems involved difficulties handling the environment, huge exercise programs from the start and the syntax of the programming language. Many of these problems disappear with the new approach for introductory programming. There is no need to care about what files to include in order to be able to run the application. Or the common programming mistakes introduced by forgetting to put a “;” at the end of an instruction can’t happen because Eiffel[4], the chosen programming language for this course, has a very simple syntax. In addition if something goes wrong the programmer, in our case the novice programmer, will get a meaningful error message as where and what went wrong. By doing a semester thesis based on TRAFFIC I am able to do interesting and useful work. My hope is that the outcome of this semester thesis will make learning to program interesting and, due to the new teaching approach called “Inverted Curriculum” [1], more reasonable. Another reason for choosing this topic was that it offers a good mix of different areas such as programming tasks and design abilities. In addition the usability issue is an important aspect of the

expected result, which I find very interesting, mainly due to my minor subject that introduced this aspect of software to me.

Additionally, I will not work by myself but in a small team together with Rolf Bruderer, who is doing a master thesis on the redesign of the EDSL library and will also improve the graphics for TRAFFIC, and our supervisors Michela Pedroni and Till Bay. The team work is another reason for my high motivation and expectations about this semester thesis.

### 1.3 Scope Of The Work

My semester thesis consists of the following parts:

- **Redesign of TRAFFIC:** I will redesign the TRAFFIC library in order to make it easy to use and easy to understand. A major focus will lie on the user group of the target which are first year students. Another aspect will be to use a reasonable amount of design patterns.
- **Documentation:** I will produce a new documentation of the framework and possibly some additional course material to guide students and developers to find their way through the framework. Documentation, especially of the interfaces is crucial.
- **Report:** All taken design decisions and the experience with the system will be described in the final report.

### 1.4 Intended Results

I intend to achieve the following results at the end of my semester thesis:

- **Redesign of TRAFFIC:** An easy to use, understandable and extendable object-oriented software framework of the TRAFFIC system.
- **Documentation:** Good documentation material to guide the students and further developers through the framework.
- **Report:** A documentation of all the work and decisions I have made as well as the experiences I have had during my semester thesis.

## 2 Background Material

### 2.1 Reading List

- 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)
- 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, *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>

- Bertrand Meyer, *Object-Oriented Software Construction*, 2nd edition, Prentice Hall PTR, 1997.
- Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley Professional, 1995.
- Karine Arnout, *From Patterns to Components*, ETH Zurich, 2004  
<http://se.inf.ethz.ch/people/arnout/patterns/>
- 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>
- 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)

## 3 Project Management

### 3.1 Objectives and Priorities

Ordered by priorities, the main objectives of this work are as follows:

1. Study the existing framework.
2. Get an overview about possible design patterns, their use, advantages and disadvantages as well as their suitability to be used for the redesign of the existing TRAFFIC framework.
3. Redesign the existing framework with respect to ease of use, understandability and reuseability, with a special focus on first year students needs.
4. Implementation of the redesigned framework.

In parallel I will write my report.

### 3.2 Criteria for Success

#### 3.2.1 Design of the Framework

As the TRAFFIC framework is used in introductory teaching of programming, a good example of an object-oriented software is crucial. Therefore the following points will have to be met:

- easy to understand
- easy to use

- easy to remember
- sound
- object-oriented design (reusable, extendable, generic, ...)
- reasonable use of design patterns

### 3.2.2 Quality of Code

Since the framework is also meant to be an exemplary piece of software for students the following points are essential:

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

### 3.2.3 Documentation

During my semester thesis some material to guide students and further developers through the framework will be produced. This documentation has to be:

- easy to understand
- complete
- useful

The same criteria will apply to my final report.

## 3.3 Method of Work

As already mentioned in the motivation, I am not working by myself but in a small team. Therefore regular team meetings to discuss important design decisions will be held.

During my work, I will continuously document my program code carefully. My supervisors will give me helpful feedback about the code from time to time to ensure good quality of code.

For the implementation, I will use *ISE EiffelStudio 5.5* and *Gobo Eiffel Library*. For the development on the Mac only *ISE EiffelStudio 5.4* is available at the moment of this writing. Documentation will be written with *LyX*.

## 4 Plan With Milestones

### 4.1 Project Steps

- Project Start: October 2004
- Projectplan: October 2004
- Report witting: continous
- Analysis TRAFFIC: November 2004
- Redesign TRAFFIC: Decembe 2004
- Project End: February 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] 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>.
- [3] 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).
- [4] Eiffel Software. Eiffel, retrieved May 2003.  
<http://www.eiffel.com>.