

Domain Model

Objectives

This assignment has two main objectives:

- (1) Give you experience creating UML diagrams and models
- (2) Emphasize that UML can be used to model just about anything, not just software designs

Introduction

Some software is general and widely applicable (e.g., web browsers, spreadsheet programs, word processors). However, most software is written to address the needs of a specific application domain. Hospitals need hospital software. Banks need banking software. Wal-Mart needs software that is very specific to their business. Software developers must understand the ins and outs of the domain at which their software is targeted. If you don't understand how hospitals operate, you can't develop hospital software. If you don't understand how banks work, you shouldn't be developing banking software. If you find yourself in the position of having to develop software for a domain that you don't understand, your first step should be to learn about the domain. Domain knowledge can be gained in many ways: reading, attending conferences, talking to experts in the field, etc. As your knowledge of the domain grows, you should capture your knowledge by creating a document called a *domain model*. A domain model can include any form of documentation that usefully captures your newly acquired domain knowledge. Frequently a domain model will include both written language and UML diagrams that together model the concepts and processes of the domain. Not only will your domain model help you to clarify and remember what you've learned, it also will be a valuable resource during the requirements elicitation and software design phases.

Assignment

Your assignment is to select an application domain that is interesting to you, and create a domain model for it. You won't have time to actually learn about a new domain, so pick a domain that you already know about. Most domains are far too large to model in a week or two, so select some small part of your chosen domain to focus on. If you can't think of a domain that you want to model, you may select one from the following list:

- A hospital
- A bank
- A retail or grocery store
- A university
- The LDS Church
- The US Government

For example, if you model a hospital, you would want to model concepts such as: doctors, nurses, patients, wards, medical tests, medical charts, the admittance process, the checkout process, interactions between doctors, nurses, and patients, etc. For the LDS Church you would want to model the church organizational structure, interactions between the various roles in the church, church processes, etc.

Learning to model with UML is the primary goal of this assignment, so your domain model should consist primarily of UML diagrams, accompanied by whatever written explanations you find necessary to clarify and augment your UML diagrams. Your UML diagrams should be as detailed and complete as possible. They should also be non-trivial, which means you need to focus on aspects of your domain that produce interesting UML diagrams. More specifically, your domain model must include at least the following:

- 1) One large or several smaller class diagrams
- 2) Two communication diagrams
- 3) Two sequence diagrams
- 4) Two activity diagrams
- 5) Two state diagrams

In order to reduce the number of domain concepts that need to be modeled, your communication diagrams and sequence diagrams can model the same behavior.

Deliverables

One hard copy of your domain model

Grading

Your domain model will be graded on the following criteria:

- 1) Does the model focus on interesting (i.e., non-trivial) aspects of the chosen domain?
- 2) Does the model include all of the required UML diagrams?
- 3) Are the diagrams detailed and correct?
- 4) Does the model effectively communicate knowledge of the domain to the reader (this includes both UML diagrams and written explanations)?