

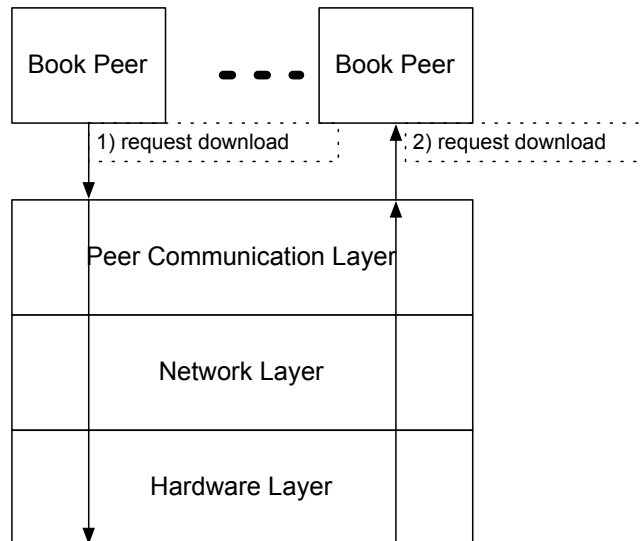
# Software Architecture in Practise — Book Swap Case

May 16, 2008

## Introduction

'Book Swap' is to be a new peer-to-peer (P2P) system for swapping books among "peers". Here peers are distributed nodes that acts as both clients and servers.

As a part of a business case for Book Swap, the software architect has outlined the following software architecture:



The main idea is that a Peer controls a set of books that other peers may download, that peers are equal, and that there is a Peer Communication Layer that abstracts away details about a specific network.

## 1 Architectural Prototyping

The architect has decided that the system should be implemented in Java and that the following interface should be observed when implementing Book Swap:

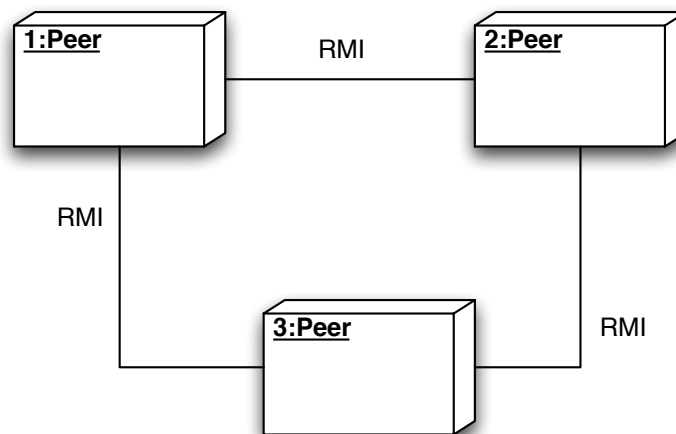
```
public interface Peer {
    /**
     * Download a Book based on a keyword search
     */
    public String download(String keyword);
    /**
     * Upload a book to this Peer
     */
    public void upload(String book);
}
```

Now, the software architect wants to apply the technique of architectural prototyping. Please consider the following:

- Outline which type(s) of architectural prototyping that is/are relevant in this setting
- The software architect is concerned about whether to use RMI or raw TCP/IP (socket) communication in the prototype. Describe how architectural prototyping may help the architect in making this decision
- The software architect is concerned about modifiability. Describe how you would create an architectural prototype in this case
- Discuss what benefits and liabilities there is of using ArchJava for architectural prototyping in this case
- The software architect decides to make a Service-Oriented Architecture. Discuss this solution

## 2 Architectural Evaluation

The software architect has designed a deployment structure of the system as shown below:



Here it has been decided to implement the system in Java and to use Java RMI for distributed communication.

The main architectural drivers for the system are modifiability and testability and the following is the main use case for the system:

The user logs on Book Swap as a Peer and offers his books for download.  
After discovering other Peers, the user downloads books from other Peers  
and allow these to upload books

The project stakeholders now want to do an evaluation of the system and decide that the ATAM method should be applied.

Please consider the following:

- Describe the steps of the ATAM method and how they would map to the case
- Point to possible examples of sensitivity points, trade-off points, and risks
- Discuss strengths and weaknesses of ATAM in relation to the case
- Discuss in which way DiscoText could be used to evaluate the architecture of a (prototype) implementation of the system
- Compare and contrast ATAM, architectural prototyping, CBAM, and aSQA as ways of evaluating a software architecture

### 3 The Architecture Process and Architecture Competence

The organization that develops the Book Swap application is a large publishing house that plans to offer its publications for peer-to-peer swapping with each swap incurring a fee for the organization.

The architect is now targetted with presenting a proposed software architecture process for project management.

Please consider the following:

- Outline which architectural design activities are relevant in this case and what they should contain
- The architect decides to use a backlog in his architecture work. Describe how the backlog may be used
- Consider which types of duties, skills, and knowledge are most relevant in this case
- Give examples of how Organizational Coordination and Organizational Learning could ideally take place in this organization
- Give examples of what would make the organization architecturally competent. Give examples of what would make the organization architecturally incompetent.

## 4 Architectural Decisions

For the development of the Book Swap application, it has been decided to 1) use Java and RMI as a distributed platform, 2) to use the software architecture methods of Bass et al. (2003), 3) to always program towards an interface, and 4) there cannot be any node that is a server or a client.

Please consider the following:

- What kind of design decisions are the four decisions?
- Using one of the decisions as an illustration, use the template of Tyree and Akerman (2005) to document it (as far as possible)
- Discuss how representations of design decisions could be used in architectural evaluation
- Explain how design decisions could fit into the IEEE 1471 framework

## 5 Architectural Reconstruction

Consider the following (part of a) prototype of Book Swap that the architect of Book Swap has built (where `Peer` extends `Remote`):

```
package dk.saip
public class Peer {
    private Collection<String> books = new LinkedList<String>();
    private Collection<Peer> peers = new LinkedList<Peer>();
    private Network network = new Network();

    public int download(String keyword, Peer to) {
        for (String book: books) {
            if (book.contains(keyword)) {
                return network.send(book, to);
            }
        }
        return network.send(null, to);
    }

    public void upload(String book, Peer from) {
        books.add(book);
    }

    ...
}
```

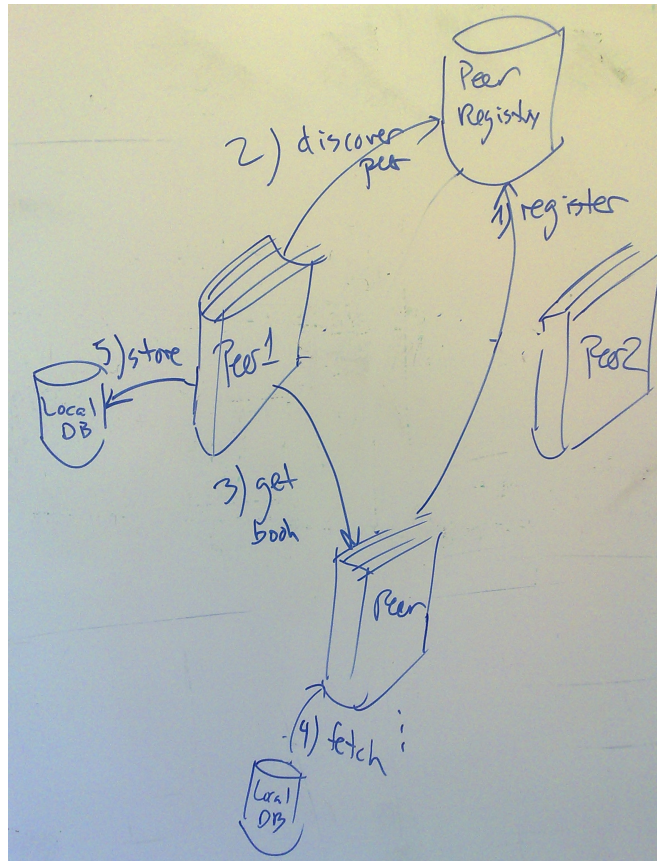
He has decided to implement the network communication in C++ allowing for higher speed.

Please consider the following:

- Explain the Symphony process and how it would apply to this case. No formal architectural documentation exists currently
- What kinds of source views, target view, and hypothetical views would there be in this case? How could they be obtained?
- What would a typical UML tool reverse engineer the prototype into? What would be missing?
- What information would be needed to reconstruct requirements or design decisions?

## Connectors

The architect of Book Swap has drawn the following rich picture of his intended architecture:



Please consider the following

- The architect wants to describe his design using connector types. Outline which types he has chosen
- Which other connector types are/are not applicable to Book Swap?
- The architect decides to implement the system in ArchJava. Discuss the benefits and liabilities of this choice
- Sketch the code of Book Swap in ArchJava (it does not have to compile)
- What consequences does the choice of using ArchJava have for architecture reconstruction?

## Architectural Reflection

The customer of the Book Swap system requires that the system becomes self-managed. One specific requirement is that for the mobile version of Book Swap, the system should always choose the best available network connection, trading off cost and bandwidth of the connection. Another requirements is that if one Book Peer fails, it should be discovered and if possible repaired.

Please consider the following:

- Where does Book Swap (with the added requirements) belong on the Autonomic Computing axis?
- Illustrate how the three-layer architecture of Kramer and Magee (2007) could map to Book Swap
- The architect decides to using DiscoTect in the lower layer of the three-layer architecture. Which kinds of probes would he need? Give examples (verbally) of rules that he could write to transform system events into architectural events