Technische Universität Berlin

Institut für Telekommunikationssysteme
Fachgebiet Architektur der Vermittlungsknoten

Fakultät IV
Franklinstrasse 28-29
10587 Berlin
http://www.av.tu-berlin.de

Diploma Thesis

Hysteretic Neural Network and Trading

Zongxiong Chen

Matriculation Number: 0390198

xx.xx.2019

Supervised by
Prof. Dr. Thomas Magedanz

Assistant Supervisor
Your second Supervisor

FOKUS Institute
Kaiserin-Augusta-Allee 31
10589 Berlin

This dissertation originated in cooperation with the Fraunhofer Institute for Open Communication Systems (FOKUS).

First of all I would like to thank Prof. Dr. Thomas Magedanz at the Fraunhofer Institute FOKUS for giving me the opportunity to carry out state of the art research in this field.

Special thanks to Mister X and Mister Y for their guidance. — Some personel words... —

Furthermore I would like to thank ...

Hereby I declare that I wrote this thesis myself with the help of no more than the mentioned literature and auxiliary means.

Berlin, 01.01.2050

. . . .
(Signature [your name])

Abstract

This template is intended to give an introduction of how to write diploma and master thesis at the chair 'Architektur der Vermittlungsknoten' of the Technische UniversitÃt Berlin. Please don't use the term 'Technical University' in your thesis because this is a proper name.

On the one hand this PDF should give a guidance to people who will soon start to write their thesis. The overall structure is explained by examples. On the other hand this text is provided as a collection of LaTeX files that can be used as a template for a new thesis. Feel free to edit the design.

It is highly recommended to write your thesis with LaTeX. I prefer to use Miktex in combination with TeXnicCenter (both freeware) but you can use any other LaTeX software as well. For managing the references I use the open-source tool jabref. For diagrams and graphs I tend to use MS Visio with PDF plugin. Images look much better when saved as vector images. For logos and 'external' images use JPG or PNG. In your thesis you should try to explain as much as possible with the help of images.

The abstract is the most important part of your thesis. Take your time to write it as good as possible. Abstract should have no more than one page. It is normal to rewrite the abstract again and again, so probaly you won't write the final abstract before the last week of due-date. Before submitting your thesis you should give at least the abstract, the introduction and the conclusion to a native english speaker. It is likely that almost no one will read your thesis as a whole but most people will read the abstract, the introduction and the conclusion.

Start with some introductionary lines, followed by some words why your topic is relevant and why your solution is needed concluding with 'what I have done'. Don't use too many buzzwords. The abstract may also be read by people who are not familiar with your topic.

Zusammenfassung

Da die meisten Leuten an der TU deutsch als Muttersprache haben, empfiehlt es sich, das Abstract zusätzlich auch in deutsch zu schreiben. Man kann es auch nur auf deutsch schreiben und anschließend einem Englisch-Muttersprachler zur Übersetzung geben.

Contents

List of Figures

List of Tables

1 Introduction

This chapter should have about 4-8 pages and at least one image, describing your topic and your concept. Usually the introduction chapter is separated into subsections like 'motivation', 'objective', 'scope' and 'outline'.

1.1 Motivation

Start describing the situation as it is today or as it has been during the last years. 'Over the last few years there has been a tendency... In recent years...'. The introduction should make people aware of the problem that you are trying to solve with your concept, respectively implementation. Don't start with 'In my thesis I will implement X'.

1.2 Objective

What kind of problem do you adress? Which issues do you try to solve? What solution do you propose? What is your goal? 'This thesis describes an approach to combining X and Y... The aim of this work is to...'

1.3 Scope

Here you should describe what you will do and also what you will not do. Explain a little more specific than in the objective section. 'I will implement X on the platforms Y and Z based on technology A and B.'

Conclude this subsection with an image describing 'the big picture'. How does your solution fit into a larger environment? You may also add another image with the overall structure of your component.

'Figure 1.1 shows Component X as part of ...'

1.4 Outline

The 'structure' or 'outline' section gives a brief introduction into the main chapters of your work. Write 2-5 lines about each chapter. Usually diploma thesis are separated into 6-8 main chapters.

This example thesis is separated into 7 chapters.

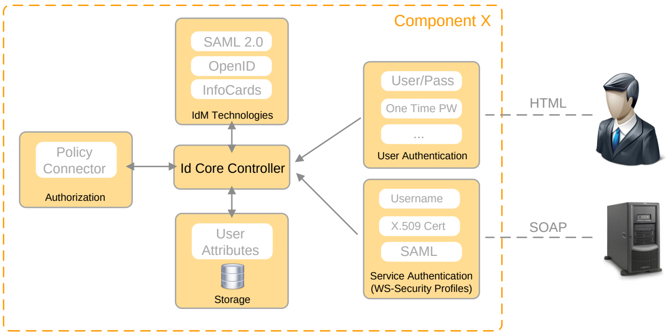

Figure 1.1: Component X

Chapter 2 is usually termed 'Related Work', 'State of the Art' or 'Fundamentals'. Here you will describe relevant technologies and standards related to your topic. What did other scientists propose regarding your topic? This chapter makes about 20-30 percent of the complete thesis.

Chapter 3 analyzes the requirements for your component. This chapter will have 5-10 pages.

Chapter 4 is usually termed 'Concept', 'Design' or 'Model'. Here you describe your approach, give a high-level description to the architectural structure and to the single components that your solution consists of. Use structured images and UML diagrams for explanation. This chapter will have a volume of 20-30 percent of your thesis.

Chapter 5 describes the implementation part of your work. Don't explain every code detail but emphasize important aspects of your implementation. This chapter will have a volume of 15-20 percent of your thesis.

Chapter 6 is usually termed 'Evaluation' or 'Validation'. How did you test it? In which environment? How does it scale? Measurements, tests, screenshots. This chapter will have a volume of 10-15 percent of your thesis.

Chapter 7 summarizes the thesis, describes the problems that occurred and gives an outlook about future work. Should have about 4-6 pages.

2 Fundamentals and Related Work

This section is intended to give an introduction about relevant terms, technologies and standards in the field of X. You do not have to explain common technologies such as HTML or XML.

2.1 Technologies

This section describes relevant technologies, starting with X followed by Y, concluding with Z.

2.1.1 Technology A

It's always a good idea to explain a technology or a system with a citation of a prominent source, such as a widely accepted technical book or a famous person or organization.

Exmple: Tim-Berners-Lee describes the "WorldWideWeb" as follows:
"The WorldWideWeb (W3) is a wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of documents." [**?**]

You can also cite different claims about the same term.
According to Bill Gates *"Windows 7 is the best operating system that has ever been released"* [**?**] (no real quote) In opposite Steve Jobs claims Leopard to be *"the one and only operating system"* [**?**]

If the topic you are talking about can be grouped into different categories you can start with a classification. Example: According to Tim Berners-Lee XYZ can be classified into three different groups, depending on foobar [**?**]:

- Mobile X
- Fixed X
- Combined X

2.1.2 Technology B

For internal references use the 'ref' tag of LaTeX. Technology B is similar to Technology A as described in section 2.1.1.

2.1.3 Comparison of Technologies

Name	Vendor	Release Year	Platform
A	Microsoft	2000	Windows
B	Yahoo!	2003	Windows, Mac OS
C	Apple	2005	Mac OS
D	Google	2005	Windows, Linux, Mac OS

Table 2.1: Comparison of technologies

2.2 Standardization

This sections outlines standardization approaches regarding X.

2.2.1 Internet Engineering Task Force

The IETF defines SIP as '...' [**?**]

2.2.2 International Telecommunication Union

Lorem Ipsum...

2.2.3 3GPP

Lorem Ipsum...

2.2.4 Open Mobile Alliance

Lorem Ipsum...

2.3 Concurrent Approaches

There are lots of people who tried to implement Component X. The most relevant are ...

3 Methodology

This section determines the requirements necessary for X. This includes the functional aspects, namely Y and Z, and the non functional aspects such as A and B.

3.1 Overview

In his paper about Z, Mister X outlines the following requirements for a Component X.

3.2 Hysteretic neural networks

3.2.1 Play and Prandtl-Ishlinskii networkds

Consider $K > 0$ play operators. Each of them maps an initial state $p_0^k \in \mathbb{R}$ and an input sequence $x_1, x_2, \ldots$ to an output sequence $p_1^k, p_2^k, \ldots$, i.e.,

$$p_0^k, (x_1, x_2, \ldots) \mapsto (p_1^k, p_2^k, \ldots), k = 1, \ldots, K$$

The kth play operator is given by:

$$p_n^k = G(x_n, p_{n-1}^k, w^k) := p_{n-1}^k + \Phi(w^k x_n - p_{n-1}^k), n = 1, 2, \ldots \tag{3.1}$$

where w^k are parameters and

$$\Phi(x) = \begin{cases} x - \frac{1}{2}, & x > \frac{1}{2} \\ 0, & -\frac{1}{2} <= x <= \frac{1}{2} \\ x + \frac{1}{2}, & x < \frac{1}{2} \end{cases} \tag{3.2}$$

See Fig. 3.1

It can be represented as a recurrent neural network, see Fig. 3.1. Note that in such a form the network is not feed-forward. One can unfold it to make it feed-forwaard, see Fig. **??**

definition

Definition 3.2.1 *We call this network a* play network*. If there are* m *elements in the sequence x_n, we say the unfolded network is* m-unfolded

For example, the network in Fig. **??** is 2-unfolded.

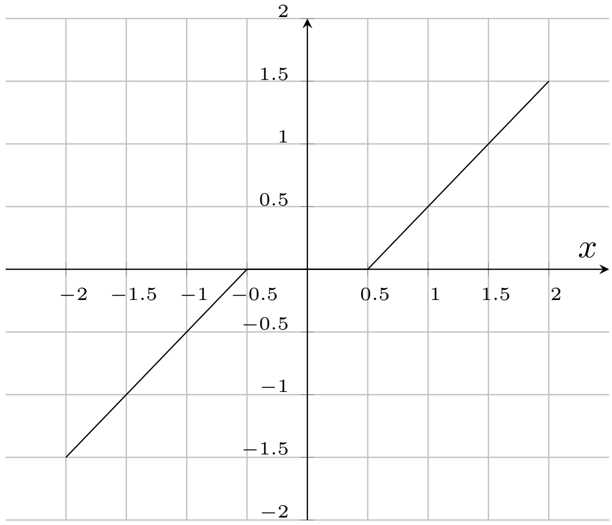

Figure 3.1: $\Phi(x)$

3.2.2 Play layers and Prandtl-Ishlinskii networkds

3.2.3 Training a PI network

Assume we are given an input sequence $x_1, x_2, \ldots, x_N$ and an output sequence $q_1, q_2, \ldots, q_N$. We perform the following steps in cycle until convergence.

1. Preparing initial states for the m-unfolded network: Fix a vector of initial states P_0 and all the weights (denoted by W). For each $k = 1, \ldots, K$, we calculate recursively $p_1^k, p_2^k, \ldots, p_N^k$ by formula **??**. We denote the corresponding (intermediate) states of the **PI** operator by

$$P_n = (p_n^1, \ldots, p_n^K), n = 1, \ldots, N.$$

2. Preparing inputs for the m-unfolded network: We fix m and group the input sequence into m-tuples:

$$\mathbf{x_1} := (x_1, \ldots, x_m), \quad \mathbf{x_2} := (x_2, \ldots, x_{m+1}), \quad \ldots,$$

which gives $M := N - m$ tuples $\mathbf{x_1}, \ldots, \mathbf{x_M}$. Next we form a new set of inputs for the m-unfolded network, attaching the vectors of intermediate states:

$$\mathbf{y_1} := (P_0, \mathbf{x_1}), \quad \mathbf{y_2} := (P_1, \mathbf{x_2}), \quad \ldots,$$

3. Training the m-unfoled network: We train by stochstic gradient descent the feed-forward m-unfolded **PI** network

$$\mathbb{R}^K \times \mathbb{R}^m \ni \mathbf{y} \mapsto F_m(\mathbf{y}) \in \mathbb{R}^m$$

with the inputs $\mathbf{y_1}, \ldots, \mathbf{y_M}$ and the true targets $\mathbf{q_1}, \ldots, \mathbf{q_M}$, where

$$\mathbf{q_1} = (q_1, \ldots, {}_m), \quad \mathbf{q_2} = (q_2, \ldots, q_{m+1}), \quad \ldots$$

4. We update the initial state P_0:

$$P_0^{new} := P_0 - \nabla_{P_0}(F_m(P_0, \mathbf{x_1}) - \mathbf{q_1})^2$$

3.2.4 General hysteretic networks

A general network may consist of several play layers (and perhaps standard layers). We can such a network *hysteretic*, and denote

$$p_n = F(X_n, W)$$

where $X_n = (x_1, \ldots, x_n)$ and w is a vector of all weights of the network.

3.2.5 Demand/Supply price formation

In this model, we assume that the stock exchange accommodates two types of agents, which we call D and N and describe below. We assume that the time takes values $n = 0, 1, 2, , \ldots,$ and denote the price at time n by p_n. By defnition, p_n is the price of last transaction at time n. We will see below that demand equals supply at times n, but between any two consecutive times $n-1$ and n, this need not be the case and there may occur many transactions until demand will have become supply.

Assumption 3.2.1 *(Strategy of agents D). Agents D keep track of a trend. They buy stocks iff the price goes up and sell stocks iff the pric goes down. The total amount of stocks that are in possession of all the agents D can be described as a Prandtl-Ishlinksii operator whose input is the price p. We denote this operator by*

$$P_D(p)$$

Assumption 3.2.2 *(Strategy of agents N). The strategy of each of the agents N is characterized by a non-ideal relay with two fixed threshold $p_1 < p_2$ (different for different agents). The relay is in state 0 if the price higher than p_2 and in state 1 if the price is lower than p_1. The agent buys one stock whenever his relay switches from 0 to 1 and sells one stock whenever his relay switches frmo 1 to 0. The total amount of stocks that are in possession of all the agents N can be described as a Preisach operator whose input is the price p. We denote this operator by*

$$P_N(p)$$

The following lemma is direct consequence of the definition of the opeartors $P_D(p)$ and $P_N(p)$.

Lemma 3.2.1 1. *If p is increasing, the $P_D(p)$ is increasing (agents D buy) and $P_N(p)$ is descreasing (agents N sell).*

2. *If p is descreasing, the $P_D(p)$ is descreasing (agents D sell) and $P_N(p)$ is increasing (agents N buy).*

Remark 3.2.1 *Since $P_D(p)$ and $P_N(p)$ are not functions but operators, the monotone curves described in 3.2.1 are not defined only by the current value of p, but depend on its prehistory.*

Denote by B_n the total amount of stocks that are in possession of all the agents D and N at time n. We will see below in 3.2.4

Assumption 3.2.3 *At each moment n, the price p_n is a stable solution of equation*

$$P_D(p_n) + P_N(p_n) = B_n \tag{3.3}$$

where the stability notion is explained in Fig.??, We will say that p_n is a stabilized price.

In order to explain how a stabilized price can change stabilize to the next value, we make the next assumption.

Assumption 3.2.4 *Suppose p_{n-1} is a stabilized price as time $n-1$. In particular, it is a stable solution of equation*

$$P_D(p_{n-1}) + P_N(p_{n-1}) = B_{n-1} \tag{3.4}$$

We assue that between time moments $n-1$ and n, some external agents E buy or sell some stocks. As a result, the total amount of stocks that is in possession of agents D and N becomes B_n. Moreover, we assume that external agents do not stay at the stock exchange, i.e., the operators (thir densities) $P_D(p)$ and $P_N(p)$ do not chagne in time. If $B_n \leq B_{n-1}$, then external agents E buy $|B_n - B_{n-1}|$ stocks, which increases the price. As as result, agents D also buy. All the stocks bought by agents E and D are sold by agnets N. This leads to the increase of the price to a new value p_n^1, where p_n^1 is the smallest solution of the equation

$$P_D(p_n^1) + P_N(p_n^1) = B_n \tag{3.5}$$

satisfying the inequality

$$p_n^1 \geq p_{n-1} \tag{3.6}$$

Note that small values $|B_n - B_{n-1}|$ may correspond to small change of the price as in Fig.?? or large changes as in Fig.??. In the latter case, the price jumps up due to the fold bifurcation.

If p_n^1 is a stable of equation 3.5, the, by definition, it coincides with the new stabilized price:

$$p_n := p_n^1 \tag{3.7}$$

*Otherwise, the price makes several jumps taking semi-stable values $p_n^2, \dots, p_n^{m-1}$ and a stable value p_n^m (**hopefully with a finite** m) such that*

$$missingnow \tag{3.8}$$

By definition, we set

$$p_n := p_n^m \tag{3.9}$$

Analogously, the price will leave the stable value p_{n-1} if $B_n > B_{n-1}$. In this case external agents E sell $B_n - B_{n-1}$ stocks, which decreases the price. As a result, agents D also sell. All the stocks sold by agents E and D are bought by agent N.

The last assumption concerns the strategy of external agents E.

Assumption 3.2.5 *B_n is a Markov chain. For example, $B_n \sim \mathcal{N}(B_{n-1} + \mu, \tau^{-1})$ with some mean μ and precision $\tau > 0$*

Remark 3.2.2 *Set*

$$G(p) := P_D(p) + P_N(p) \tag{3.10}$$

Then, formally, the relationship between the price p_n and the noise B_n is the same as Dima's model. **??** Though in our second model, there is a number of further restrictions on admissible vlaues of p_n due to Assumption 3.2.4

3.2.6 Gradient of networks

First we only consider **one play**

$$G(P_n, w^1) = \sum_{i=1}^{S} \tilde{\theta_i} \tanh(\theta_i P_n + \theta_{i0}) + \tilde{\theta_0} \tag{3.11}$$

Where $P_n = [p_1, p_2, \dots, p_n]$, $G(P_n, w^1) = [y_1, y_2, \dots, y_n]$, $\forall i \in [1, ..., S]$, $\theta_i P_n = (\theta_i p_1, \theta_i p_2, \dots, \theta_i p_n)$,

So $\tanh(\theta_i P_n + \theta_{i0}) = [\tanh(\theta_i p_1 + \theta_{i0}), \tanh(\theta_i p_2 + \theta_{i0}), \dots, \tanh(\theta_i p_n + \theta_{i0})]$

So $\sum_{i=1}^{S} \tilde{\theta_i} \tanh(\theta_i P_n + \theta_{i0}) + \tilde{\theta_0} = [\sum_{i=1}^{S} \tilde{\theta_i} \tanh(\theta_i p_1 + \theta_{i0}) + \tilde{\theta_0}, \sum_{i=1}^{S} \tilde{\theta_i} \tanh(\theta_i p_2 + \theta_{i0}) + \tilde{\theta_0}, \dots, \sum_{i=1}^{S} \tilde{\theta_i} \tanh(\theta_i p_n + \theta_{i0}) + \tilde{\theta_0}] = [y_1, y_2, ..., y_n]$.

Take y_j, where $j \in [1, ..., n]$ for example

Let $z_j = \theta_i p_j + \theta_{i0}$ and $f(z_j) = \tanh(\theta_i p_j + \theta_{i0})$, we obtain

$$y_j = \sum_{i=1}^{S} \tilde{\theta_i} \tanh(\theta_i p_j + \theta_{i0}) + \tilde{\theta_0} = \sum_{i=1}^{S} \tilde{\theta_i} f(z_j) + \tilde{\theta_{i0}} \tag{3.12}$$

Calculate derivation for y_j,

$$\frac{\partial y_j}{\partial p_j} = \sum_{i=1}^{S} \tilde{\theta_i} \theta_i \frac{\partial f(z_j)}{\partial z_j} \tag{3.13}$$

Now let's consider the mapping between p_j and x_j. let $\sigma_j = w^1 x_j - p_{j-1}$

$$p_j = \Phi(\sigma_j) + p_{j-1} \tag{3.14}$$

and

$$\Phi(x) = \begin{cases} x - 1/2, & x > 1/2 \\ 0, & -1/2 < x < -1/2 \\ x + 1/2, & x < -1/2 \end{cases} \tag{3.15}$$

Using chain rule, we obtain

$$\frac{\partial y_j}{\partial x_j} = \frac{\partial y_j}{\partial p_j}\frac{\partial p_j}{\partial x_j} = \sum_{i=1}^{S} \tilde{\theta}_i \theta_i w^1 \frac{\partial f(z_j)}{\partial z_j}\frac{\partial \Phi(\sigma_j)}{\partial \sigma_j} \tag{3.16}$$

To consider **multiple plays** case, we reformulate the derivation as following:

$$\frac{\partial y_j^1}{\partial x_j} = \frac{\partial y_j^1}{\partial p_j^1}\frac{\partial {p_j}^1}{\partial x_j} = \sum_{i=1}^{S} \tilde{\theta}_i^1 \theta_i^1 w^1 \frac{\partial f(z_j^1)}{\partial z_j^1}\frac{\partial \Phi(\sigma_j^1)}{\partial \sigma_j^1} \tag{3.17}$$

Now from the architecture, we know that if we have P plays,

$$F = \frac{1}{P}\sum_{k=1}^{P} G^k \tag{3.18}$$

Where $F = [f_1, f_2, ..., f_n]$, and

$$f_j = \frac{1}{P}\sum_{k=1}^{P} y_j^k \tag{3.19}$$

our derivation is:

$$\frac{\partial f_j}{\partial x_j} = \frac{1}{P}\sum_{k=1}^{P}\frac{\partial y_j^k}{\partial x_j} = \frac{1}{P}\sum_{k=1}^{P}\frac{\partial y_j^k}{\partial p_j^k}\frac{\partial p_j^k}{\partial x_j} = \frac{1}{P}\sum_{k=1}^{P}\sum_{i=1}^{S}\tilde{\theta}_i^k \theta_i^k w^k \frac{\partial f(z_j^k)}{\partial z_j^k}\frac{\partial \Phi(\sigma_j^k)}{\partial \sigma_j^k} \tag{3.20}$$

4 Concept

This chapter introduces the architectural design of Component X. The component consists of subcomponent A, B and C.

In the end of this chapter you should write a specification for your solution, including interfaces, protocols and parameters.

4.1 Sub-component A

The concept chapter provides a high-level explanation of your solution. Try to explain the overall structure with a picture. You can also use UML sequence diagrams for explanation.

Figure 4.1 illustrates the situation between Alice and Bob. (sequence diagram from www.websequencediagrams.com)

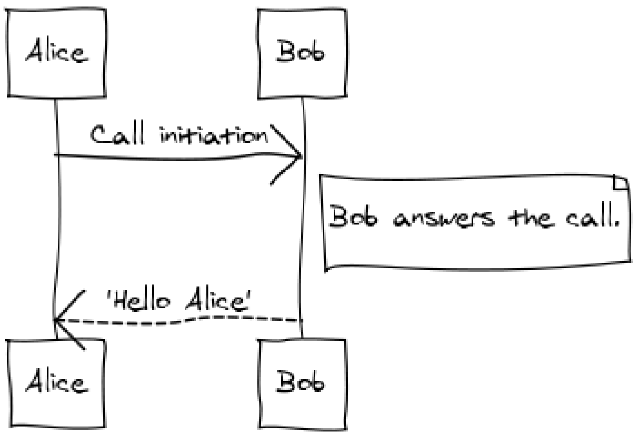

Figure 4.1: Alice and Bob

4.2 Sub-component B

Lorem Ipsum...

4.3 Proposed API

Lorem Ipsum...

4.4 Layer X

Lorem Ipsum...

4.5 Interworking of X and Y

Lorem Ipsum...

4.6 Interface Specification

Lorem Ipsum...

5 Implementation

This chapter describes the implementation of component X. Three systems were chosen as reference implementations: a desktop version for Windows and Linux PCs, a Windows Mobile version for Pocket PCs and a mobile version based on Android.

5.1 Environment

The following software, respectively operating systems, were used for the implementation:

- Windows XP and Ubuntu 6
- Java Development Kit (JDK) 6 Update 10
- Eclipse Ganymede 3.4
- Standard Widget Toolkit 3.4

5.2 Project Structure

The implementation is seperated into 2 distinguished eclipse projects as depicted in figure 5.1.

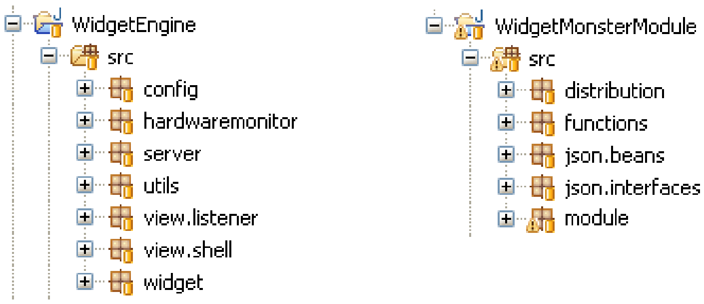

Figure 5.1: Project Structure

The following listing briefly describes the single packages of both projects in alphabetical order to give an overview of the implementation:

config
Lorem Ipsum...

server
Lorem Ipsum...

utils
Lorem Ipsum...

5.3 Important Implementation Aspects

Do not explain every class in detail. Give a short introduction about the modules or the eclipse projects. If you want to explain relevant code snippets use the 'lstlisting' tag of LaTeX. Put only short snippets into your thesis. Long listing should be part of the annex.

Listing 5.1: JSON String Code Snippet


```
{
        id: 1,
        method: "myInstance.getGroup",
        params: ["Teammates", 2, true]
}

{
        id: 2,
        result: [
                  "groupDesc":"These are my teammates",
                  {
                          "javaClass":"src.package.MemberClass",
                          "memberName": "Bob",
                  }
                ]
}
```


You can also compare different approaches. Example: Since the implementation based on X failed I choosed to implement the same aspect based on Y. The new approach resulted in a much faster ...

5.4 Graphical User Interface

Lorem Ipsum...

5.5 Documentation

Lorem Ipsum...

6 Evaluation

In this chapter the implementation of Component X is evaluated. An example instance was created for every service. The following chapter validates the component implemented in the previous chapter against the requirements.

Put some screenshots in this section! Map the requirements with your proposed solution. Compare it with related work. Why is your solution better than a concurrent approach from another organization?

6.1 Test Environment

Fraunhofer Institute FOKUS' Open IMS Playground was used as a test environment for the telecommunication services. The IMS Playground ...

6.2 Scalability

Lorem Ipsum

6.3 Usability

Lorem Ipsum

6.4 Performance Measurements

Lorem Ipsum

7 Conclusion

The final chapter summarizes the thesis. The first subsection outlines the main ideas behind Component X and recapitulates the work steps. Issues that remained unsolved are then described. Finally the potential of the proposed solution and future work is surveyed in an outlook.

7.1 Summary

Explain what you did during the last 6 month on 1 or 2 pages!

The work done can be summarized into the following work steps

- Analysis of available technologies
- Selection of 3 relevant services for implementation
- Design and implementation of X on Windows
- Design and implementation of X on mobile devices
- Documentation based on X
- Evaluation of the proposed solution

7.2 Dissemination

Who uses your component or who will use it? Industry projects, EU projects, open source...? Is it integrated into a larger environment? Did you publish any papers?

7.3 Problems Encountered

Summarize the main problems. How did you solve them? Why didn't you solve them?

7.4 Outlook

Future work will enhance Component X with new services and features that can be used ...

List of Acronyms

3GPP	3rd Generation Partnership Project
AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
AS	Application Server
CSCF	Call Session Control Function
CSS	Cascading Stylesheets
DHTML	Dynamic HTML
DOM	Document Object Model
FOKUS	Fraunhofer Institut fuer offene Kommunikationssysteme
GUI	Graphical User Interface
GPS	Global Positioning System
GSM	Global System for Mobile Communication
HTML	Hypertext Markup Language
HSS	Home Subscriber Server
HTTP	Hypertext Transfer Protocol
I-CSCF	Interrogating-Call Session Control Function
IETF	Internet Engineering Task Force
IM	Instant Messaging
IMS	IP Multimedia Subsystem
IP	Internet Protocol
J2ME	Java Micro Edition
JDK	Java Developer Kit
JRE	Java Runtime Environment
JSON	JavaScript Object Notation
JSR	Java Specification Request
JVM	Java Virtual Machine
NGN	Next Generation Network
OMA	Open Mobile Alliance
P-CSCF	Proxy-Call Session Control Function
PDA	Personal Digital Assistant
PEEM	Policy Evaluation, Enforcement and Management
QoS	Quality of Service
S-CSCF	Serving-Call Session Control Function
SDK	Software Developer Kit
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SMS	Short Message Service

SMSC	Short Message Service Center
SOAP	Simple Object Access Protocol
SWF	Shockwave Flash
SWT	Standard Widget Toolkit
TCP	Transmission Control Protocol
Telco API	Telecommunication API
TLS	Transport Layer Security
UMTS	Universal Mobile Telecommunication System
URI	Uniform Resource Identifier
VoIP	Voice over Internet Protocol
W3C	World Wide Web Consortium
WSDL	Web Service Description Language
XCAP	XML Configuration Access Protocol
XDMS	XML Document Management Server
XML	Extensible Markup Language

Annex

```xml
<?xml version="1.0" encoding="UTF-8"?>
<widget>
	<debug>off</debug>
	<window name="myWindow" title="Hello Widget" visible="true">
		<height>120</height>
		<width>320</width>
		<image src="Resources/orangebg.png">
			<name>orangebg</name>
			<hOffset>0</hOffset>
			<vOffset>0</vOffset>
		</image>
		<text>
			<name>myText</name>
			<data>Hello Widget</data>
			<color>#000000</color>
			<size>20</size>
			<vOffset>50</vOffset>
			<hOffset>120</hOffset>
		</text>
	</window>
</widget>
```

Listing 1: Sourcecode Listing

```
INVITE sip:bob@network.org SIP/2.0
Via: SIP/2.0/UDP 100.101.102.103:5060;branch=z9hG4bKmp17a
Max-Forwards: 70
To: Bob <sip:bob@network.org>
From: Alice <sip:alice@ims-network.org>;tag=42
Call-ID: 10@100.101.102.103
CSeq: 1 INVITE
Subject: How are you?
Contact: <sip:xyz@network.org>
Content-Type: application/sdp
Content-Length: 159
v=0
o=alice 2890844526 2890844526 IN IP4 100.101.102.103
s=Phone Call
t=0 0
c=IN IP4 100.101.102.103
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000

SIP/2.0 200 OK
Via: SIP/2.0/UDP proxy.network.org:5060;branch=z9hG4bK83842.1
;received=100.101.102.105
Via: SIP/2.0/UDP 100.101.102.103:5060;branch=z9hG4bKmp17a
To: Bob <sip:bob@network.org>;tag=314159
From: Alice <sip:alice@network.org>;tag=42
Call-ID: 10@100.101.102.103
CSeq: 1 INVITE
Contact: <sip:foo@network.org>
Content-Type: application/sdp
Content-Length: 159
v=0
o=bob 2890844526 2890844526 IN IP4 200.201.202.203
s=Phone Call
c=IN IP4 200.201.202.203
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Listing 2: SIP request and response packet[**?**]