

MASTER'S THESIS 2010:XX

Augmented Reality on iPhone
Full implementation of a Client-Server
Application.

JEAN-LOUIS GIORDANO

Department of Signals and Systems
Division of XXXXXXXXXXXXXXXXXXXX
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2010

Augmented Reality on iPhone
Full implementation of a Client-Server Application.
JEAN-LOUIS GIORDANO

© JEAN-LOUIS GIORDANO, 2010

Master's Thesis 2010:XX

Signals and Systems
Division of XXXXXXXXXXXXXXXXXX
Chalmers University of Technology
SE-41296 Göteborg
Sweden

Tel. +46-(0)31 772 1000

Cover:
Artwork of the application used on the AppStore, drawn by the author.

Reproservice / Department of Signals and Systems
Göteborg, Sweden 2010

Augmented Reality on iPhone
Full implementation of a Client-Server Application.
Master's Thesis in Communication Engineering
JEAN-LOUIS GIORDANO
Department of Signals and Systems
Division of XXXXXXXXXXXXXXXXXXXX
Chalmers University of Technology

Abstract

[illegible][illegible]

Keywords: iPhone, Augmented Reality, Realtime, AppStore, Ruby, Erlang, Objective-C, Mnesia, Google Map

Contents

Abstract	iii
Contents	iv
Acknowledgements	vii
1. Introduction	1
2. Basic Principles of Augmented Reality	3
2.1. What is Augmented Reality?	3
2.2. How does it work?	4
2.3. What is it for?	5
3. On the Client Side	7
3.1. The iPhone	7
3.1.1. The device	7
3.1.2. The Development Tools	7
3.2. The Application	8
3.2.1. What it does	8
3.2.2. How it works	8
3.3. Implementation of the Augmented Reality View	9
3.3.1. View Hierarchy	9
3.3.2. Positioning within 6 degrees of freedom	9
3.3.3. 3D Projection	10
4. On the Server Side	11
4.1. Architecture	11
4.2. Implementation	11
4.2.1. Ruby	11
4.2.2. Erlang	12
4.3. Results	12
5. Conclusion	15
5.1. On the AppStore	15

5.2. Further Considerations	15
5.2.1. on the Client	15
5.2.2. on the Server	15
5.3. Summary	15
References	17
A. Some Code	19

Acknowledgements

1. Introduction

Augmented Reality is a dream coming true. Of course, the concept itself is not brand new and has been used and studied for many years, but modern technologies and devices now enable Augmented Reality to be implemented relatively painlessly and be used on a day-to-day basis, in a vast range of applications from advertising to medical assistance.

At the same time, the last generation smartphones are equipped with sensors that locates its users and can interact with them in cleaver and innovative ways. The iPhone is one of those. That is why Augmented Reality on smartphones, and especially on iPhone, has been growing fast for the last couple of years.

This master thesis focuses on the design of an application of Augmented Reality on iPhone 3G-S and its implementation from both a Client and Server point of view.

At first an overview over the basic principles of Augmented Reality (chapter 2) will help understanding Augmented Reality, how it works and how it can be used. Then we will have a look at the Client side of the application (chapter 3) to see what the application is from a user's point of view, and we will go through the implementation on the iPhone to understand the underlying code structure. We will also have a look at the Server implementation (chapter 4) to get an idea of the require architecture to provide data in real-time. Finally, we will describe the results achieved with the current implementation (chapter 5), and a view on further improvements will finalize the thesis.

2. Basic Principles of Augmented Reality

2.1. What is Augmented Reality?

Augmented Reality is a term that is used to describe digital systems that display an artificial visual layer of three-dimensional or two-dimensional models upon our natural perception of reality in real-time. It aims at improving or completing our limited way of seeing the real world by presenting artificial elements that would otherwise be hidden to our senses.

The most commonly known Augmented Reality application is the weather report system. The reporter is in a real environment in front of a blue/green screen, but the end user in front of his TV will see the reporter standing in front of an interactive map: the content of the background virtual data mapped on a real world object in real-time. That is Augmented Reality.

It is during the year 1992 that a Boeing researcher named Tom Caudell first defined "Augmented Reality" in a paper called "Augmented reality: an application of heads-up display technology to manual manufacturing processes" [Cau 92]. The term was used to describe a digital system that displays virtual graphics onto a physical reality.

The concept of Augmented Reality itself is older than its name, since the first Augmented Reality device was built and presented in a 1968 paper by Ivan Sutherland [Sut 68]. The system was designed to track the head position of the user to project before his eyes two-dimensional models in order to create an illusion of three dimensions. It relied on the idea that moving perspective images would appear three-dimensional even without stereo presentation, a principle which is called the "kinetic depth effect". In such a system, the user will perceive the object as three-dimensional only when he moves, otherwise the object will only be perceived as planar.

It has to be noted that Augmented Reality is not Virtual Reality. In Virtual Reality, there are no elements of Reality, as much as in Reality there are no elements of Virtuality by definition. In his paper called "Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum" [Mil 94], Milgram defines the Reality-Virtuality Continuum shown in Fig. 2.1.

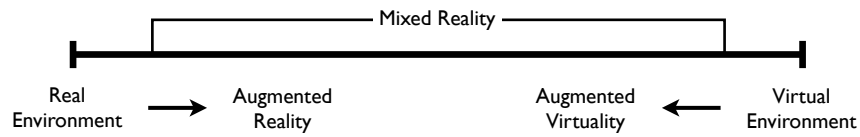


Figure 2.1.: Milgram's Reality-Virtuality Continuum [Mil 94]

As can be seen from this diagram, Augmented Reality is based on reality and tends to Virtuality in an area called "Mixed Reality". It lies near the real world end of the line, which means the predominate perception is the real world. On the symmetric position, Milgram defines Augmented Virtuality. Behind this concept lie systems where real world items added to a Virtual Environment, such as texture mapping on virtual objects in video games.

So the limits of Augmented Reality are defined by the furthest one can go at adding virtual data on a real world representation without depriving the user of its predominate perception of Reality.

2.2. How does it work?

In order to get an Augmented Reality application, one must integrate artificial object into a real scene. Of course, such an object must be rendered to scale and at the correct position and orientation. To do so, the system requires to know the position of the camera or the user from the object. This is the core principle of Augmented Reality.

To solve this problem, there are two main approaches:

- Use sensors to know the position of the object from the camera

By the mean of tags on a real object, one can get a coordinate system on which to project a virtual model. Those tags can be visual or based on any technology that can be use to locate that object, for example a set of Infra-Red diodes.

- Use sensors to know the position of the camera in a known space

If the environment is known, i.e. the application is internally aware of the position of the object, or if the object is fully virtual and does not rely on any physical counterpart,

one can evaluate the position of the camera in this environment and render the object accordingly. The position of the camera can be known by the mean of a large enough set of sensors that will be able to position it in the environment space.

Once the virtual environment is known, each virtual object must be rendered on top of an existing view by the mean of geometric projections that fits the estimated position of this object in the virtual space.

2.3. What is it for?

Applications are numerous and are invading a greater and greater number of domains: entertainment of course, with video games or virtual scavenger hunt, but it also has medical uses and various advantages for the industry.

3. On the Client Side

3.1. The iPhone

3.1.1. The device

The iPhone is a last generation smartphone developed by Apple and was revealed for the first time in 2007. It features advanced functionalities such as multimedia support, internet access, video games and many more.

Its main characteristic is that its User Interface (UI) is only based on two inputs:

- a multi-touch screen
- a 3-axis accelerometers

The last generation of iPhone, the iPhone 3G-S, also includes a GPS and a compass, and an upcoming version of its Operating System (OS) will allow the access to the video data of the camera.

Up to this day, forty-two millions iPhone have been sold across the globe.

3.1.2. The Development Tools

To develop an application for iPhone, a specific Framework is required.

First of all, any software developed for iPhone must be programmed in Objective-C, although it is possible to call C and C++ functions from the code.

Objective-C is an Object-Oriented (OO) reflexive programming language build upon the C language. It is comparable to C++ from this point, but differs greatly in many ways, especially by its dynamic message passing system, by being weakly typed and by being able to perform dynamic loading. In its latest version, Objective-C also features a Garbage Collector which abstract the programmer of the memory management consideration. Unfortunately, this feature is not available for iPhone development.

In order to compile code for the iPhone, the use of Xcode as Integrated Development Environment is almost unavoidable. Apple has a highly proprietary approach for its

products, and programming for an Apple environment is much restrictive to this regard. Fortunately, Xcode and the set of tools provided by Apple offer a great comfort of use in many cases, especially for debugging, or for creating interfaces with the use of an Interface Builder (IB).

Once Xcode is set up, the iPhone Software Development Kit must be installed on Xcode. The iPhone SDK takes advantage of all the features of Xcode, including its set of external tool to the largest extent. After being registered to the iPhone developer's program, Xcode can also be linked directly to an actual iPhone device in order to test an application in real-time with any monitoring tool available.

This SDK contains Application Programming Interfaces (API) to all the accessible libraries of the iPhone, including Cocoa for the user interface. Unfortunately, many functionalities such as the access to raw data from the video camera are not part of a public API, so their libraries are considered as not accessible and therefore are forbidden by Apple.

3.2. The Application

3.2.1. What it does

This application has been designed for travellers in Sweden that are looking for a nearby public transport stop. Augmented Reality makes it very intuitive to find a nearby stop: the user just has to "look around" with his iPhone, and the closest stops will appear at their positions. To make it even easier for the user, a Google Map is available when the device is held horizontally if directions are required.

This makes the application very interesting for people that are easily confused when reading maps when discovering a new city.

But this application has also been designed for people that are already familiar with the city they visit. Simply by pointing at a stop they are interested in, they can get the next departures without having to go till the stop to check the timetable.

For now, the application is available for Göteborg and Stockholm, with their respective public transport companies Västtrafik and Storstockholms Lokaltrafik. But additional providers could be added later on.

3.2.2. How it works

The application takes advantage of the GPS capabilities of the iPhone to locate the user on a map. Then thanks to the compass, we are able to estimate the direction he is

looking at, and eventually the 3-axis accelerometer allows us to evaluate the angle at which he holds his phone.

Once the position of the user is known, a request is made over the internet to determine the bus stops that are close to him.

Since the positions of the bus stops are known on the server side, we can project them in the virtual space according to the user's coordinates, heading direction and phone holding angle.

Note that our application requires a GPS and a compass, therefore is only compatible with the iPhone 3G-S.

3.3. Implementation of the Augmented Reality View

3.3.1. View Hierarchy

3.3.2. Positioning within 6 degrees of freedom

In our application, we need to locate the camera of the iPhone within 6 degrees of freedom:

- Latitude
- Longitude
- Altitude
- Azimuth
- Inclination on the X-axis
- Inclination on the Z-axis

The GPS directly gives us the Altitude, Latitude and Longitude of the user whereas the compass gives us his Azimuth.

Thanks to the 3-axis accelerometer, the inclination on the X and Z axis can be computed by the mean of simple trigonometry.

3.3.3. 3D Projection

First, we project a Map on the plane defining the ground. The iPhone has an API to perform 3D projections, by the mean of layer transformations.

To apply a transform to a layer, one must provide the transform matrix corresponding to the desired projection.

In this case, the transformation will be a rotation of $\frac{\pi}{2}$ on the X-axis and a rotation corresponding to the Azimuth on the Y-Axis, followed by a translation to give depth to the view.

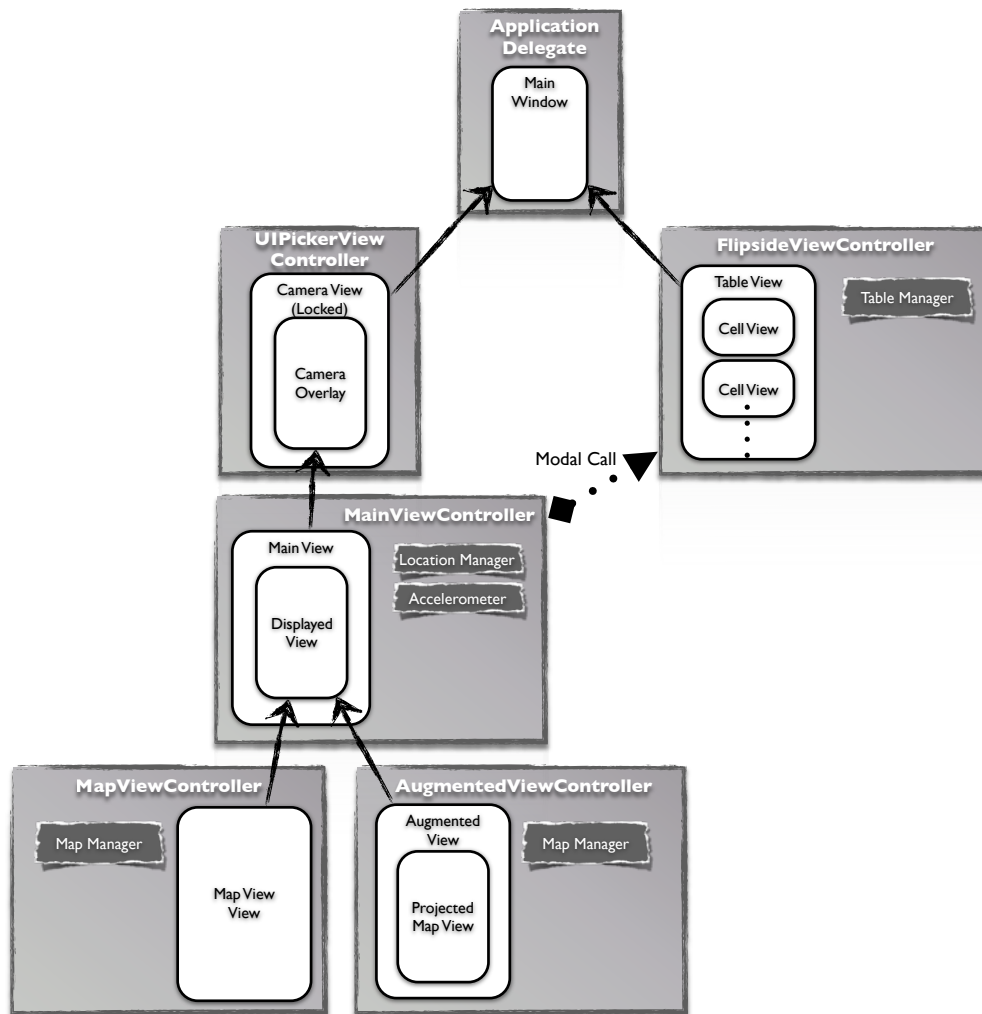


Figure 3.1.: Scheme of the View Hierarchy of the application

4. On the Server Side

4.1. Architecture

The server processes the request for nearby bus stops as follows:

1. receive a request with coordinates (Latitude and Longitude)
2. find the 10 nearest bus locations from the given coordinates in its SQLite database
3. fetch the forecast for each of these stops in an independent lightweight process
4. once each forecast is obtained, parse the result into JSON
5. reply to the client with the JSON

The application is hosted on a Debian machine (Ubuntu) with a Yaws webserver. Yaws is the Erlang alternative to Apache, and as such offers high availability and high scalability.

Erlang is a functional programming language especially designed to handle high concurrency and distributed systems. A large part of the application is therefore implemented in Erlang in order to take advantage of this.

But to make it easy to maintain and update, the more complex tasks are implemented in Ruby.

Ruby is a modern dynamic scripted language, and has been designed to make the developers happy. As such, it is a powerful tool to implement complex algorithms painlessly.

4.2. Implementation

4.2.1. Ruby

Ruby is used to implement the SQL request to the database and to parse data. It is also used to maintain the database.

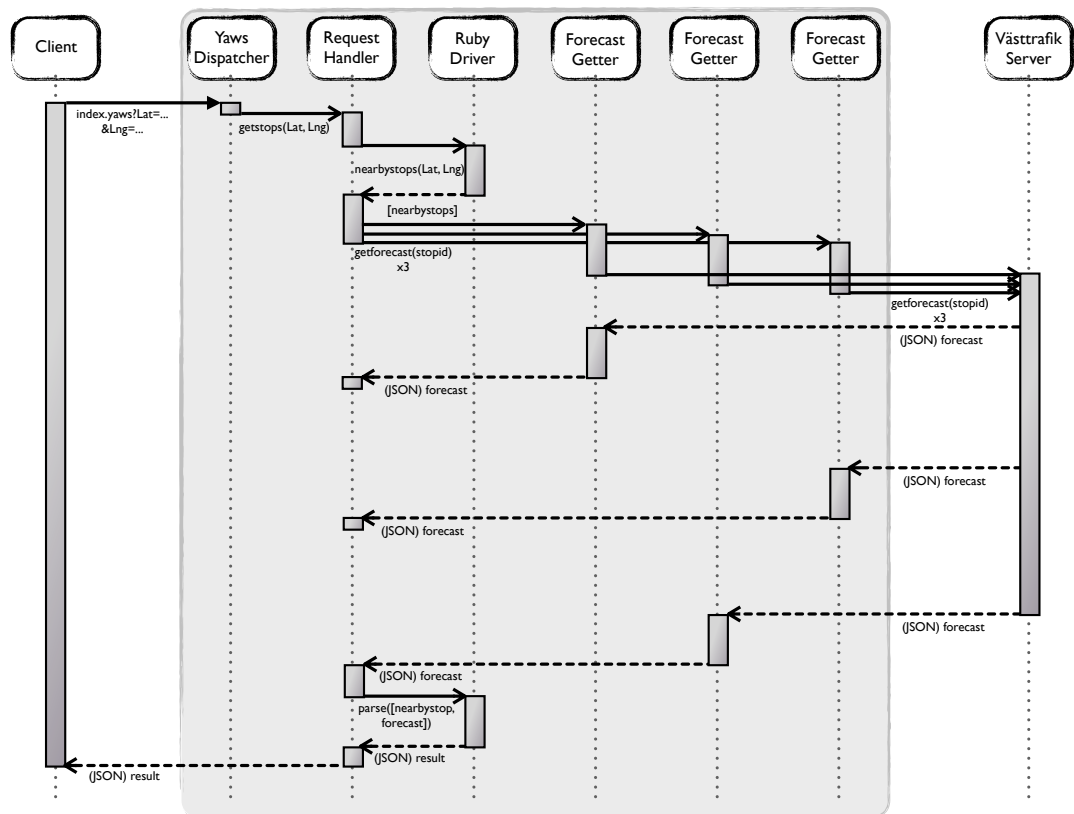


Figure 4.1.: Simplified UML Sequential Diagram of a request

4.2.2. Erlang

Erlang is used to dispatch the lightweight processes, to perform the distant requests to Västtrafik and Storstockholms Locaktrafik, and also to take care of data caching by the mean of a Mnesia database.

4.3. Results

Everything works fine.
(give numbers)

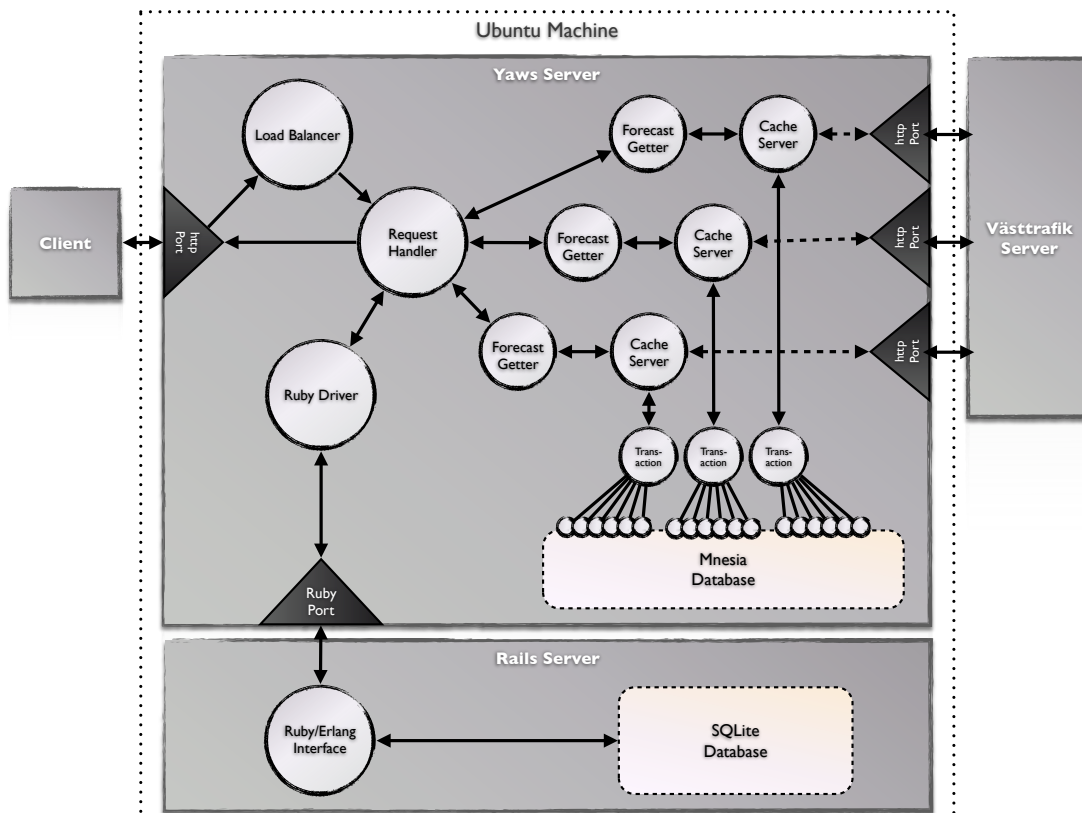


Figure 4.2.: Scheme of the Server Architecture

5. Conclusion

5.1. On the AppStore

The first release of the application in its revision number 0.9 was uploaded under the name "Hållplats Väst" on the AppStore on March the 8th and was ready for sale on March the 13th.

After one month on the store, more than 400 units were downloaded.

5.2. Further Considerations

5.2.1. on the Client

Implement image recognition.

5.2.2. on the Server

Extend to more cities in Sweden or even Europe.

5.3. Summary

References

- [Cau 92] Caudell, T.P. and Mizell, D.W: *Augmented reality: an application of heads-up display technology to manual manufacturing processes*, Res. & Technol., Boeing Comput. Services, Seattle, January 1992

- [Sut 68] Sutherland, I. E. , *A head-mounted three dimensional display*, Proceedings of the December 9-11, 1968, fall joint computer conference, part I, San Francisco, California, December 1968

- [Mil 94] Milgram, P., H. Takemura, et al., *Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum*. SPIE Proceedings: Telemanipulator and Telepresence Technologies . H. Das, SPIE. 2351 : 282-292, 1994

A. Some Code