



UNIVERSITY OF SOMETHING
THE GRADUATE SCHOOL
SOME DEPARTMENT

My Thesis Title

This is my subtitle

PHD THESIS

ALBERT EINSTEIN



Supervisor: George Washington
Professor

New York, February 2015

— DRAFT COPY v.2.1. —



UNIVERSITY OF SOMETHING
THE GRADUATE SCHOOL
SOME DEPARTMENT

My Thesis Title

This is my subtitle

PHD THESIS

ALBERT EINSTEIN

Supervisor: George Washington
Professor

Approved by ... November 22, 2010.

(Signature)

(Signature)

(Signature)

.....
George Washington
Professor

.....
Professor A
Professor

.....
Professor B
Professor

New York, February 2015

— DRAFT COPY v.2.1. —

— DRAFT COPY v.2.1. —



UNIVERSITY OF SOMETHING
THE GRADUATE SCHOOL
SOME DEPARTMENT

— DRAFT COPY v.2.1. —

Copyright © - All rights reserved. Albert Einstein, 2030.

Copyright statement

Another statement

Some Statement

And another one ...

(Signature)

.....
Albert Einstein

— DRAFT COPY v.2.1. —

A peer-to-peer system is a set of autonomous computing nodes (the peers) which cooperate in order to exchange data. The peers in the peer-to-peer systems that are widely used today, rely on simple keyword selection in order to search for data. The need for richer facilities in exchanging data, as well as, the evolution of the Semantic Web, led to the evolution of the schema-based peer-to-peer systems. In those systems every node uses a schema to organize the local data. So there are two ways in order for data search to be feasible. The first but not so flexible way implies that every node uses the same schema. The second way gives every node the flexibility to choose a schema according with its needs, but on the same time requires the existence of mapping rules in order for queries to be replied. This way though, doesn't offer automatic creation and dynamic renewal of the mapping rules which would be essential for peer-to-peer systems.

This diploma thesis aims to the development of a schema-based peer-to-peer system that allows a certain flexibility for schema selection and on the same time enables query transformation without the use of mapping rules. The peers use RDF schemas that are subsets (views) of a big common schema called global schema.

Keywords

Peer-to-peer, Schema-based peer-to-peer, Semantic Web, RDF/S, RQL, Jxta

— DRAFT COPY v.2.1. —

— DRAFT COPY v.2.1. —

to my parents

— DRAFT COPY v.2.1. —

Acknowledgements

I would like to thank ...

New York, May 2020

Albert Einstein

— DRAFT COPY v.2.1. —

— DRAFT COPY v.2.1. —

Abstract	1
Acknowledgements	5
Preface	17
1 Introduction	19
I Part A	21
2 Background	23
2.1 section1	23
2.1.1 sub-section 1	23
2.2 section 2	23
II Part B	25
3 Background	27
3.1 section 3.1	27
3.1.1 sub-section 3.1.1	27
3.2 section 3.2	27
III Part C	29
4 Background	31
5 Background	33
5.1 section 5.1	33
5.1.1 sub-section 5.1.1	33
5.2 section 5.2	33
Appendices	35
A Test Appendix	37
A.1 First Section	37
B Figures	39
C Tables	41
D Index Generation	43
E Illustrations	45

F Algorithms	47
Bibliography	51
Abbreviations	53
Glossary	55
Index	57

— DRAFT COPY v.2.1. —

B.1 NOR gate. 39

— DRAFT COPY v.2.1. —

— DRAFT COPY v.2.1. —

E.1 frog 45

— DRAFT COPY v.2.1. —

— DRAFT COPY v.2.1. —

C.1 My table. 41

— DRAFT COPY v.2.1. —

— DRAFT COPY v.2.1. —

List of Algorithms

2.1	How to write algorithms	23
2.2	An algorithm	24
F.1	A floating algorithm	47
F.2	An "inline" algorithm	48

— DRAFT COPY v.2.1. —

Preface goes here ...

— DRAFT COPY v.2.1. —

— DRAFT COPY v.2.1. —

This is my Introduction ... [1]

— DRAFT COPY v.2.1. —

— DRAFT COPY V

I

Part A

— DRAFT COPY v.2.1. —

Contents

2.1 section1	23
2.1.1 sub-section 1	23
2.2 section 2	23

2.1 section1

2.1.1 sub-section 1

2.2 section 2

This is another chapter ...

— DRAFT COPY v.2.1. —

ALGORITHM 2.1: How to write algorithms

Data: this text

Result: how to write algorithm with L^AT_EX2_ε

initialization **while** *not at end of this document* **do**

 read current **if** *understand* **then**

 | go to next section current section becomes this one

else

 | go back to the beginning of current section

end

end

ALGORITHM 2.2: An algorithm

```
if if condition then  
  | something if  
else if elseif condition then  
  | something elseif  
else  
  | something else  
end
```

blah blah

— DRAFT COPY V

||

Part B

— DRAFT COPY v.2.1. —

Contents

3.1 section 3.1	27
3.1.1 sub-section 3.1.1	27
3.2 section 3.2	27

3.1 section 3.1

3.1.1 sub-section 3.1.1

3.2 section 3.2

This is another chapter ...

— DRAFT COPY v.2.1. —

— DRAFT COPY v

III

Part C

— DRAFT COPY v.2.1. —

This is another chapter ...

— DRAFT COPY v.2.1. —

Contents

5.1 section 5.1	33
5.1.1 sub-section 5.1.1	33
5.2 section 5.2	33

5.1 section 5.1

5.1.1 sub-section 5.1.1

5.2 section 5.2

This is another chapter ...

— DRAFT COPY v.2.1. —

Appendices

— DRAFT COPY v.2.1. —

A.1 First Section

...

— DRAFT COPY v.2.1. —

```
\begin{figure}[!ht]
  \centering
  \includegraphics{figures/2.png}
  \caption{NOR gate.}
  \label{figureB.1}
\end{figure}
```

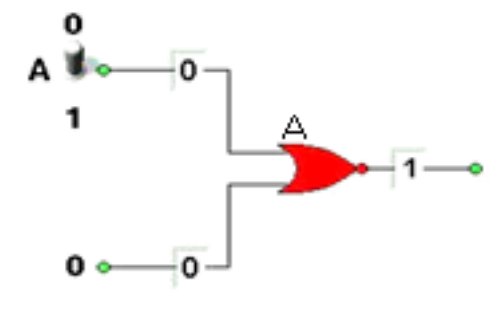


Figure B.1: *NOR gate.*

— DRAFT COPY v.2.1. —

```
\begin{table}[h!]  
  \centering  
  \caption{My table.}  
  \begin{tabular}{|c|c|}  
    \hline  
    \textbf{A} & \textbf{B} \\ \hline  
    a11 & a12 \\ \hline  
    a21 & a22 \\ \hline  
  \end{tabular}  
  \label{table01}  
\end{table}
```

Table C.1: *My table.*

A	B
a11	a12
a21	a22

— DRAFT COPY v.2.1. —


```
\index{xerox} \index{babel} \index{anna} \index{babylon}
```

— DRAFT COPY v.2.1. —

```
\begin{Illustration}[!h]  
  \centering  
  \includegraphics[width=0.5\textwidth]{figures/frog.jpg}  
  \caption{frog}  
  \label{frog_image}  
\end{Illustration}
```



Illustration E.1: *frog*

— DRAFT COPY v.2.1. —

Algorithms are typeset using the `algorithm2e` package.

Code for floating algorithms:

```
\begin{algorithm}[tb]
  \caption{A floating algorithm}
  \begin{algorithm2e}[H]
    \uIf{if condition}{
      something if \;
      solved \;
    }
    \uElseIf{elseif condition}{
      something elseif \;
    }
    \Else{
      something else \;
    }
  \end{algorithm2e}
\end{algorithm}
```

ALGORITHM F.1: A floating algorithm

```
if if condition then
  | something if ;
  | solved ;
else if elseif condition then
  | something elseif ;
else
  | something else ;
end
```

For inline algorithms use the "H" option for both `algorithm` and `algorithm2e` environments:

```
\begin{algorithm}[H]
  \caption{An "inline" algorithm}
  \begin{algorithm2e}[H]
    \uIf{if condition}{
      something if \;
      solved \;
    }
    \uElseIf{elseif condition}{
      something elseif \;
    }
    \Else{
      something else \;
    }
  \end{algorithm2e}
\end{algorithm}
```

ALGORITHM F.2: An "inline" algorithm

```
if if condition then
| something if ;
| solved ;
else if elseif condition then
| something elseif ;
else
| something else ;
end
```

(and the paragraph continues here ...)

Of course one might prefer a plain inline presentation:

```
\begin{algorithm2e}[H]
  \uIf{if condition}{
    something if \;
    solved \;
  }
  \uElseIf{elseif condition}{
    something elseif \;
  }
  \Else{
    something else \;
  }
\end{algorithm2e}
```

Text starts here ...

```
if if condition then
| something if ;
| solved ;
else if elseif condition then
| something elseif ;
else
| something else ;
end
```

... and continues here.

— DRAFT COPY v.2.1. —

- [1] S. R. Nassif, "Design for variability in DSM technologies," in *Proc. IEEE 1st Int. Symp. Quality Electron. Des. (ISQED)*, San Jose, CA, USA, Mar. 2000, pp. 451–454.

— DRAFT COPY v.2.1. —

BPF Band Pass Filter

— DRAFT COPY v.2.1. —

— DRAFT COPY v.2.1. —

<i>term</i>	<i>description</i>
female	girl
male	boy

— DRAFT COPY v.2.1. —

anna, [43](#)

babel, [43](#)

babylon, [43](#)

xerox, [43](#)

— DRAFT COPY v.2.1. —



— DRAFT COPY v.2.1. —



University of Something
The Graduate School.
Some Department

PhD Thesis

**This is my PhD
Thesis Title**

My Name

New York
October 2014



University of Something
The Graduate School
Some Department

PhD Thesis

**This is my PhD
Thesis Title**

My Name

New York
October 2014

— DRAFT COPY v.2.1. —