

# Identifying movement patterns from large scale WiFi-based location data

-A case study of TU Delft Campus

Balazs Dukai  
Simon Griffioen  
Matthijs Bon  
Martijn Vermeer  
Xander den Duijn  
Yuxuan Kang







# abstract

This document is intended to be both an example of the TU Delft  $\LaTeX$  template for reports and theses, as well as a short introduction to its use. It is not intended to be a general introduction to  $\LaTeX$  itself,<sup>1</sup> and we will assume the reader to be familiar with the basics of creating and compiling documents.

Instructions on how to use this template under Windows and Linux, and which  $\LaTeX$  packages are required, can be found in `README.txt`.

Since a report, and especially a thesis, might be a substantial document, it is convenient to break it up into smaller pieces. In this template we therefore give every chapter its own file. The chapters (and appendices) are gathered together in `report.tex`, which is the master file describing the overall structure of the document. `report.tex` starts with the line

```
\documentclass{tudelft-report}
```

which loads the TU Delft report template. The template is based on the  $\LaTeX$  book document class and stored in `tudelft-report.cls`. The document class accepts several comma-separated options. The default language is English, but this can be changed to Dutch (*e.g.*, for bachelor theses) by specifying the `dutch` option:

```
\documentclass[dutch]{tudelft-report}
```

Furthermore, hyperlinks are shown in blue, which is convenient when reading the report on a computer, but can be expensive when printing. They can be turned black with the `print` option. This will also turn the headers black instead of cyan.

If the document becomes large, it is easy to miss warnings about the layout in the  $\LaTeX$  output. In order to locate problem areas, add the `draft` option to the `\documentclass` line. This will display a vertical bar in the margins next to the paragraphs that require attention. Finally, the `nativefonts` option can be used to override the automatic font selection (see below).

This template has the option to automatically generate a cover page with the `\makecover` command. See the next section for a detailed description.

The contents of the report are included between the `\begin{document}` and `\end{document}` commands, and split into three parts by

1. `\frontmatter`, which uses Roman numerals for the page numbers and is used for the title page and the table of contents;
2. `\mainmatter`, which uses Arabic numerals for the page numbers and is the style for the chapters;
3. `\appendix`, which uses letters for the chapter numbers, starting with 'A'.

The title page is defined in a separate file, *e.g.*, `title.tex`, and included verbatim with `\input{title}`.<sup>2</sup> Additionally, it is possible to include a preface, containing, for example, the acknowledgements. An example can be found in `preface.tex`. The table of contents is generated automatically with the `\tableofcontents`

<sup>1</sup>We recommend <http://en.wikibooks.org/wiki/LaTeX> as a reference and a starting point for new users.

<sup>2</sup>Note that it is not necessary to specify the file extension.

command. Chapters are included after `\mainmatter` and appendices after `\appendix`. For example, `\input{chapter-1}` includes `chapter-1.tex`, which contains this introduction.

The bibliography, finally, is generated automatically with

```
\bibliography{report}
```

from `report.bib`. The bibliography style is specified in `tudelft-report.bst`, which is a modified version of `apsrev4-1.bst` (from REVTeX) designed to also display the titles of referenced articles. The template will automatically generate clickable hyperlinks if a URL or DOI (digital object identifier) is present for the reference. As an example, we cite the paper by Nobel laureate Andrei Geim and his pet hamster [?]. Although it is possible to manage the bibliography by hand, we recommend using EndNote (available from Blackboard) or JabRef (available from <http://jabref.sourceforge.net/>).

This template will automatically generate a cover page if you issue the `\makecover` command. There are two formats for the cover page: one with a page-filling ('bleeding') illustration, with the title(s) and author(s) in large ultrathin typeface, and the other where the illustration fills the lower half of the A4, whereas title(s), author(s) and additional text are set in the standard sans-serif font on a plain background with a color chosen by the user. The last option is selected by the optional key `split`: `\makecover[split]` yields a page with the illustration on the lower half. All illustrations are bleeding, in accordance with the TU Delft style.

Before generating the cover, you need to provide the information to put on it. This can be done with the following commands:

- `\title[Optional Color]{Title}`  
This command is used to provide the title of the document. The title title is also printed on the spine. If you use a title page (see below), this information will be used there as well. As the title, subtitle and author name are printed directly over the cover photo, it will often be necessary to adjust the print color in order to have sufficient contrast between the text and the background. The optional color argument is used for this.
- `\title[Optional Color]{Subtitle}`  
This command is used to provide a subtitle for the document. If you use a title page (see below), this information will be used there as well. It possible to adjust the print color in order to have sufficient contrast between the text and the background – the optional color argument is used for this.
- `\author{J. Random Author}`  
This command specifies the author. The default color is `tudelft-white`, but this may be adjusted in the same way as the titles.
- `\affiliation{Technische Universiteit Delft}`  
The affiliation is the text printed vertically on the front cover. It can be the affiliation, such as the university or department name, or be used for the document type (*e.g.*, Master's thesis). The default color is again `tudelft-white`, adjustable through the color option.
- `\coverimage{cover.jpg}`  
With this command you can specify the filename of the cover image. The image is stretched to fill the full width of the front cover (including the spine if a back cover is present).
- `\covertext{Cover Text}`  
If a back cover is present, the cover text is printed on the back. Internally, this text box is created using the  $\LaTeX$  `minipage` environment, so it supports line breaks.
- `\titleoffsetx{OffsetX}`, `\titleoffsety{OffsetY}` If the cover page contains a page-filling picture (*i.e.*, `split` is not specified with the `makecover` command, the best position of the title depends a lot on the picture chosen for it. The lower left corner of the minipage containing title, subtitle and author is specified by these two commands. The offsets are measured from the top left corner of the page.
- `\afiloffsetx{AfilX}`, `\afiloffsety{AfilY}` specifies the lower left corner of the text containing the affiliation, measured from the top left corner of the page.

In addition to `[split]`, the `\makecover` command accepts several additional options for customizing the layout of the cover. The most important of these is `back`. Supplying this option will generate a back cover as well as a front, including the spine. Since this requires a page size slightly larger than twice A4 (to make room for the spine), and  $\LaTeX$  does not support different page sizes within the same document, it is wise to create a separate file for the cover. `cover.tex` contains an example. The recommended page size for the full cover can be set with

```
\geometry{papersize={1226bp,851bp}}
```

after the document class and before `\begin{document}`.

The other options `\makecover` accepts are

- `nospine`  
If a back cover is generated, the title will also be printed in a black box on the spine. However, for smaller documents the spine might not be wide enough. Specifying this option disables printing the title on the spine.
- `frontbottom`  
By default the black box on the front is situated above the blue box. Specifying this option will place the black box below the blue one.
- `spinewidth`  
If a back cover is present, this option can be used to set the width of the spine. The default is `spinewidth=1cm`.
- `frontboxwidth, frontboxheight, backboxwidth, backboxheight`  
As their names suggest, these options are used to set the width and height of the front (black) and back (blue) boxes. The default widths and heights are 4.375in and 2.1875in, respectively.
- `x, y`  
The blue and black boxes touch each other in a corner. The location of this corner can be set with these options. It is defined with respect to the top left corner of the front cover. The default values are `x=0.8125in` and `y=3in`.
- `margin`  
This option sets the margin between the borders of the boxes and their text. The default value is 12pt.

For a thesis it is desirable to have a title page within the document, containing information like the thesis committee members. To give you greater flexibility over the layout of this page, it is not generated by a command like `\makecover`, but instead described in the file `title.tex`. Modify this file according to your needs. The example text is in English, but Dutch translations are provided in the comments. Note that for a thesis, the title page is subject to requirements which differ by faculty. Make sure to check these requirements before printing.

Each chapter has its own file. For example, the  $\LaTeX$  source of this chapter can be found in `chapter-1.tex`. A chapter starts with the command

```
\chapter{Chapter title}
```

This starts a new page, prints the chapter number and title and adds a link in the table of contents. If the title is very long, it may be desirable to use a shorter version in the page headers and the table of contents. This can be achieved by specifying the short title in brackets:

```
\chapter[Short title]{Very long title with many words which could not possibly  
fit on one line}
```

Unnumbered chapters, such as the preface, can be created with `\chapter*{Chapter title}`. Such a chapter will not show up in the table of contents or in the page header. To create a table of contents entry anyway, add

```
\addcontentsline{toc}{chapter}{Chapter title}
```

after the `\chapter` command. To print the chapter title in the page header, add

`\setheader{Chapter title}`

Chapters are subdivided into sections, subsections, subsubsections, and, optionally, paragraphs and subparagraphs. All can have a title, but only sections and subsections are numbered. As with chapters, the numbering can be turned off by using `\section*{...}` instead of `\section{...}`, and similarly for the subsection.

**`\paragraph{...}`** Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

The fonts used by this template depend on which version of  $\text{\LaTeX}$  you use. Regular  $\text{\LaTeX}$ , *i.e.*, if you compile your document with `latex`, `pslatex` or `pdflatex`, will use Utopia for text, Fourier for math and Latin Modern for sans-serif and monospaced text. However, if you want to adhere to the TU Delft house style, you will need to use  $\text{\XeLaTeX}$ , as it supports TrueType and OpenType fonts. Compiling with `xelatex` will use Arial for most titles and text, Courier New for monospace and Cambria for math. If you want to have a sans-serif font for the main text, while using `latex`, `pslatex` or `pdflatex`, you can use the option `noroman` in the report style: `\usepackage[... ,noroman]tudelft-report`. For document and part titles, TU Delft Ultra Light is used. For quotes, columns and text in boxes, you use Georgia. If you want to use  $\text{\XeLaTeX}$ , but do not want to use the TU Delft house style fonts, you can add the `nativefonts` option to the document class. This will still use TU Delft Ultra Light and Arial on the cover, but not for the body of the document. If you need to use these fonts for certain sections in the main text, they are available via `\tudrmfamily` (Georgia) and `\tudtitlefamily` (TU Delft Ultra Light).

You have to learn the rules of the game. And then you have to play better than anyone else.

*Albert Einstein*

The corporate colors of the TU Delft are cyan, black and white, available via `\color{tudelft-cyan}`, `\color{tudelft-black}` (which differs slightly from the default `\color{black}`) and `\color{tudelft-white}`, respectively. Apart from these three, the house style defines the basic colors `tudelft-sea-green`, `tudelft-green`, `tudelft-dark-blue`, `tudelft-purple`, `tudelft-turquoise` and `tudelft-sky-blue`, as well as the accent colors `tudelft-lavendel`, `tudelft-orange`, `tudelft-warm-purple`, `tudelft-fuchsia`, `tudelft-bright-green` and `tudelft-yellow`.

# Introduction

## 1.1. Intro

Wireless Local Area Networks (WLAN) are widely used for indoor positioning of mobile devices within this network. The use of the Wi-Fi network to estimate the location of people is an attractive approach, since Wi-Fi access points (AP) are often available in indoor environments. Furthermore, smart phones are becoming essential in daily life, making it convincing to track mobile devices. This provides a platform to track people by using Wi-Fi monitoring technology. Knowledge of people's locations and related routine activities are important for numerous activities, such as urban planning, emergency rescue and management of buildings.

To understand the human motion behaviour many studies are conducted based on data collection of GPS receivers. The Global Navigation Satellite System (GNSS) is commonly used to track people in large scale environments. However due to poor quality of received signals from satellites in urban or indoor environments, GNSS receivers are not suitable in these environments. This led to the development of alternative technologies to track people's locations, including Bluetooth, Dead Reckoning, Radio frequency identification (RFID), ultra-wideband (UWB) and WLAN (Mautz, 2012). WLAN has the advantage of widespread deployment, low cost and with the use of a smartphone as a receiver, the possibility to track a large amount of people.

In general, there are four different location tracking techniques by using the Wi-Fi network: Propagation modelling, multilateration, Fingerprinting and Cell of Origin (CoO). Many of these methods rely on Received Signal Strength Indicators (RSSI) and/or previous set of calibration measurements. In comparison, CoO is the most straightforward technique and snaps the location of the mobile device to the same coordinate position as the access point it is connected to. For this project, CoO is used to track people's movement.

At the Technical University of Delft (TU Delft) a large scale Wi-Fi network is deployed across all facilities covering the indoor space of the campus. The network is known as an international roaming service for users in educational environments and called the eduroam network. It allows students and staff members from one university to use the infrastructure throughout the campus for free. This allows for easy collection of Wi-Fi logs including individual scans of mobile devices. A continuous collection of re-locations of devices to access points for a long duration will return detailed records of people's movement. This ubiquitous and individual history location data derived from smartphones will present valuable knowledge on movement on the campus. For this reason, the project is carried out in request of the University's department of Facility Management and Real Estate (FMRE).

In this project, Wi-Fi monitoring technology is used to discover movement patterns on the campus of TU Delft. Based on the relationship between activities and places, location history can be used to discover significant places, movement patterns and hotspots. FMRE can use this information to answer questions such "what is the relation between buildings", "where do people come from" and "how regular a trajectory occurs". This project will present a method for identification of movement patterns in a large scale indoor environments and between buildings. The method uses concepts of sequential pattern mining. Previous research has been done on sequential pattern mining, such as Zhao et al. (2014) to discover people's life patterns from mobile Wi-Fi scans, Meneses & Moreira (2012) analysed place connectivity using the eduroam network and Radaelli et al. (2013) identifies indoor movement patterns by analysing a sequence of relocations. Individual movement can be identified as a sequence of relocations of a mobile device to different APs. Without any data between two subsequent re-locations, sequential analysis is a convincing way for identifying moving

patterns from wifilogs.

## 1.2. Purpose statement

Identifying movement patterns has attracted significant interest in recent years. This report will explain how movement patterns can be identified using large scale Wi-Fi based location data. This report tries to contribute with three proposes.

1. A method for identifying movement patterns by analysing individual sequences of relocations from a large scale Wi-Fi network; This includes filtering the raw data and automatically create individual trajectories over a time interval as a sequence of relocations;
2. Restructure the association rule mining algorithm to use it in a large scale tracking environment, to discover locations that are commonly associated;
3. Investigate different visualization methods for showing movement, based on a large scale Wi-Fi network.
4. A method for automatically detect what entrances are used to enter and exit a building.

The contributions can be described in one research question for this project.

- How can movement patterns be identified from large scale WiFi-based location data of the eduroam network?

In order to answer the research question well, there are some sub questions:

- What movement patterns can be identified between buildings on TU Delft campus?
- What movement patterns can be identified between large indoor regions on TU Delft campus?
- What entrances are used to enter and exit a building on TU Delft campus?

## 1.3. Methods

The Geomatics Synthesis Project (GSP) is a small research project that combines a literature study with practical research. This includes a case study of the TU Delft campus, using real-world data. Practical work includes data storing, processing, analysing, interpretation, visualization and validation. The project is carried out in a team of six students with a connection to a supervisor and stakeholders (FMRE). This involves interactive discussions between stakeholders as an important part of the research.



# 2

## Top level requirements

To keep track of the progress of the project, it is necessary to monitor to which degree the project is meeting the top level requirements and if the project is still on schedule with these requirements. In the baseline review the requirements are specified using the MoSCoW rules and killer requirements. In this chapter these previous requirements will be discussed and possible changes will be explained.

The goals that *must* be achieved is on the level of detail of the campus. It's detailed specification as stated in the baseline review is shown below.

**MUST** campus level Main goal:

1. Identify which entrances are used to enter and exit a building;
2. Identify movement patterns and connectivity between building entrances by sequential pattern mining.

Relate entrances (place) of buildings to the corresponding APs (location). Find the stay places of each individual in order of the scan time. Find individual trajectories by taking a time interval from a sequence of stay places. Find the movement patterns, by deriving a sequence of common places shared by all trajectories. Visualize the movement patterns between buildings in static maps. A *killer requirement* for this level is: Identification of APs relating to an entrance of a building

Currently the project is progressed so far that it is possible to identify building patterns between buildings. The stay places of individuals and their trajectories have been found and this has been visualized in both static and dynamic maps and bar charts. But, until now there is no accurate map with the location of all access points of the campus. There is such a map for the faculty of architecture, but it is only one building and not very clear. Until this map of the whole campus becomes available, identifying entrances will be hard to do. section 3.5 will go into greater detail about the progress that has been made so far with entrances.

The goals that *should* be achieved, focus on the building level, where buildings are divided into regions, but since there is currently no map with the locations of the access points, this level of detail is not yet reached. However, the way that the code is setup allows for easy transformation to higher levels of detail when such a map becomes available. How this code exactly works is explained in section 3.1 and section 3.2.



# 3

## Progress

### 3.1. Data Description

The data in the table 'wifilog' contains information about the location of the Access Point (AP) in two columns. The first one is the column 'apname', which is a string with the symbolic name of the AP, for example 'A-08-G-010'. The two numbers in the second part of the string, in this case '08', represent the building number. This building number can be linked to a location in the world. The second column which contains information about location, is the column 'maploc'. This column also contains strings, which look as follows:

System Campus > [buildingid] > [specific location]'. An example of such a string is 'System Campus > 21-BTUD > 1e verdieping'. In such a string, the middle part can be linked to a building, so to a real-world location. But there are some other values for maploc, which can less clearly be linked to a real-world location. Such a value is 'Root Area', it is unclear what this value means and it contains no information about a building or area it might be in. This makes it impossible to link it to a location in the world. Then there is the value 'Unknown', a value that indicates that there was no name attached to the Access Point that user was connected to. Again in this case, it is impossible to link this value to a real-world location.

As both 'Root Area' and 'Unknown' are in the minority of records, they could be left out of the queries. But for some records, the column 'apname' did provide information about the location, while the 'maploc' column value was 'Root Area'. In most of these cases however, the building number, the second part of the string, was a number of length three. But there are no buildings on the TU Delft campus with a building number that high. When consulting Wilko Quack about this, he explained that these building numbers had an arbitrary 1 in front of the building number. So 'A-134-A-001' was not building 134, but building 34, which was an actual building number on the campus. This would mean that using the column 'apname' for getting the building number would mean a higher number of results and therefore a more realistic visualization of the movements.

Taking the substring of that column and linking it to a building with an actual location is done in two steps. First the whole string is retrieved and with a function in Python the substring is derived. Subsequently, the building id that is the result of this function can be linked to a table in the database which has for every building five columns: buildingid, name, point (as geometry), x (longitude), y (latitude) (see in subsection 3.2.3).

### 3.2. Movement between buildings

To automate the workflow of creating movement visualizations between buildings, a program is created. There is a distinction between two types of visualization:

1. Maps
2. Bar charts

For map visualization the JavaScript Leaflet.js is used, this allows for creation of an interactive user interface with a base map from Open Street Maps and visualize the buildings and movement between them. In subsection 3.2.3 the map visualization will be discussed in more detail. For the bar charts the Python module matplotlib was used. The bar plot visualizes the movement throughout the day in 24 bars. Each bar represents the movement from a selection of buildings to another selection of buildings, over a time interval of one hour. In subsection 3.2.2, the bar charts will be discussed in more detail.

### 3.2.1. Graphical User Interface

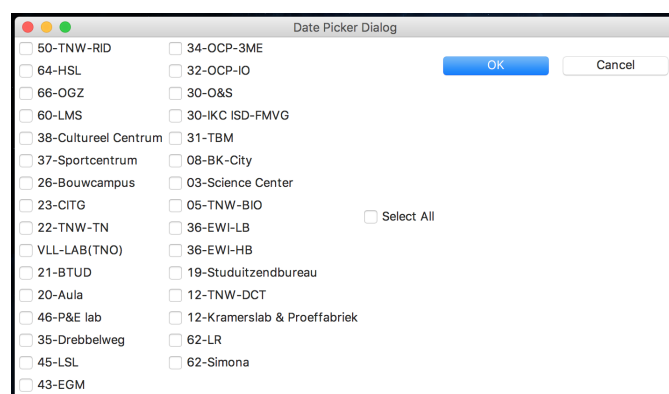
The Graphical User Interface (hereinafter referred to as GUI) for this work is a Python program that shows a Tkinter interface. When the user runs the program, it will display a main window, which is shown in Figure 3.1



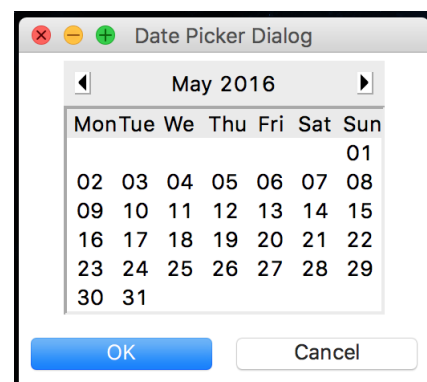
Figure 3.1: main window of GUI

To create a visualization, the user first has to select a time interval and then the buildings from and to which the movement should be visualized. The user has 2 options to select the time series for the current visualization:

1. Click on '1. Add dates' which will open the date picker dialog
2. Pick a day from the dropdown menu and click on '2. All days of week'



(a) buildings selection



(b) dates selection

Figure 3.2: GUI selection windows

Option 1 can be used to select particular days without order. Option 2 can select 'every Tuesday' or even

multiple recurring dates, such as 'every Monday to Friday'. It is also possible to combine option 1 and 2 to have for example 'every Monday and Friday the 13th of May'.

After selecting the time series, the user has to select the buildings from and to which the movement should be visualized. The 3rd and 4th button bring up the same dialog. This dialog shows checkboxes for every building. Every building that is checked will be visualized. The user also has the option to select all buildings. If the user would like to see movement from and to the same building, the user can select the same buildings twice.

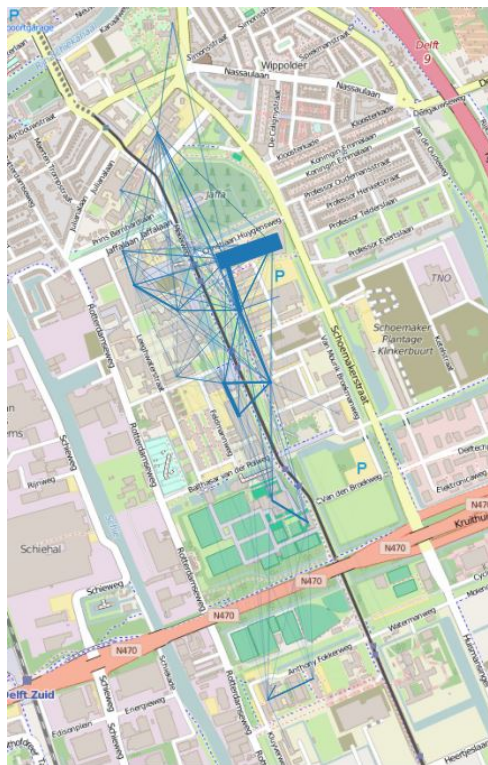
### 3.2.2. Bar chart

### 3.2.3. Maps

In order to get an overview about how people move on the campus and further more, find out movement patterns, a map visualization is essential. Map visualization consists of three parts:

1. base map: open street map is used as base map. There are many labels on open street map, providing more context of the environment, so it is more clear and readable compared to other base maps like satellite images.
2. building markers: building markers show the locations of the buildings. Google maps marker style is used since it is commonly used in many map application. Because the shape of the building is not useful in analyzing movement patterns between buildings, each building is regarded as a point instead of a polygon, thus a node in the network,
3. lines: lines are the most essential part in map visualization, they represent movements between buildings.

In the first stage of map visualization, only base map and lines are taken into consideration, building markers are not shown on the map. The line width represents the amount of movement and movements are aggregated daily regardless of the timestamp of each movement during a day. This map visualization gives an overview of the movements over a day and between which buildings there are the most movements. The following maps show the difference of the amount of movement between April 11th (weekday) and April 17th (weekend).



(a) April 11th, weekday



(b) April 17th, weekend

Figure 3.3: Static visualization

It's clear that between Aula and library, there are the most movements and the amount of movements is totally different on weekday and on weekend.

Regarding the movements are dynamic and time is also highly related to movements, a dynamic map visualization is created to display individual movement over a day with temporal information. The following screenshots of the gif file show how the movements look like at a certain time of a day:



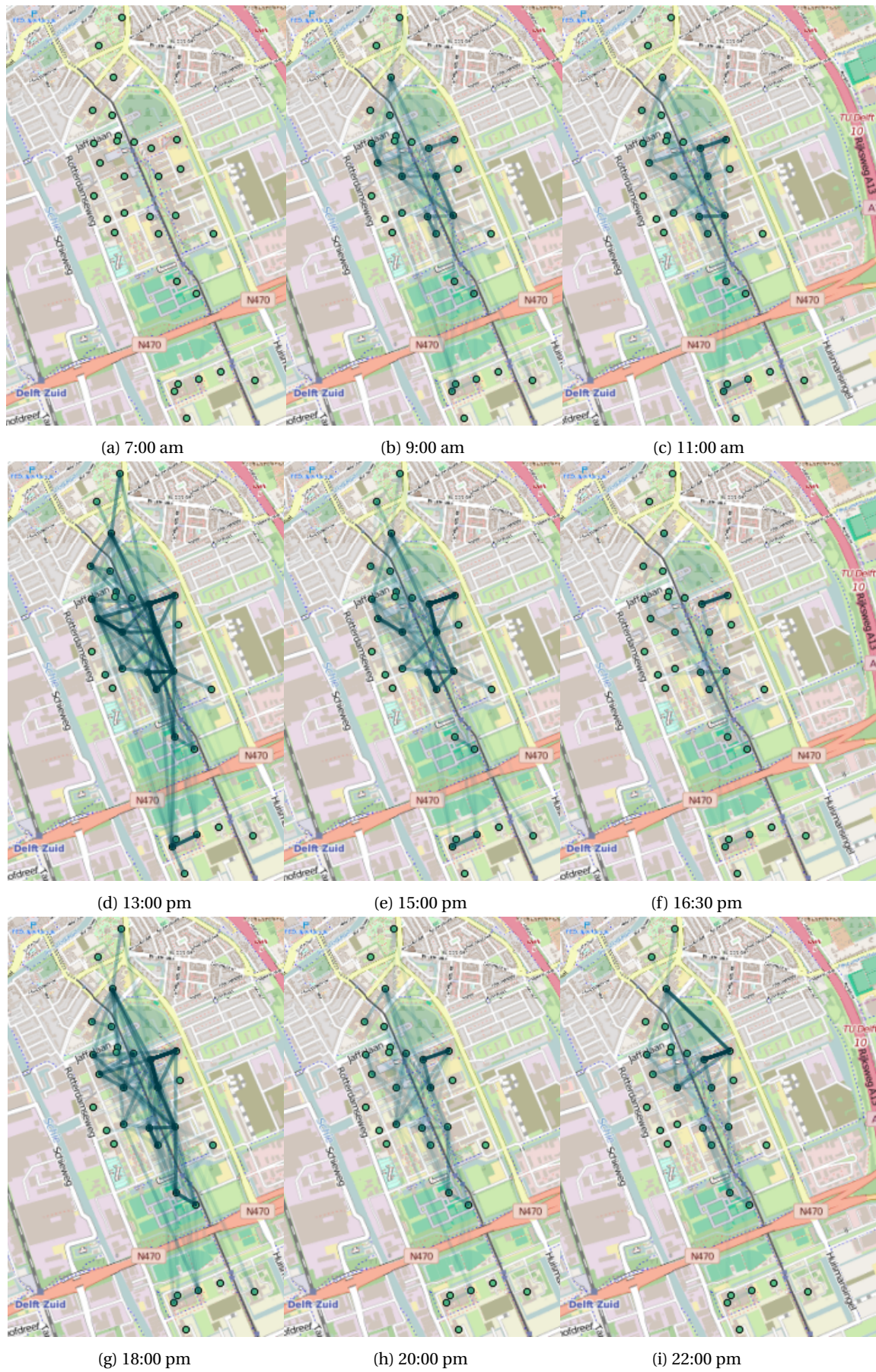


Figure 3.4: Dynamic visualization of movements, April 11th

In these pictures, the more movements there are, the less transparent the lines are. So generally speaking, from 7:00 am to 20:00 pm, there are two peaks at 13:00 pm and 18:00 pm. Hence, it is possible to get some insights about movement patterns from the animation. However, the dynamic map visualization doesn't provide detailed information to dig into but only an overview. So in order to mine on movement patterns, it is necessary to create maps containing more information, including time, direction and so forth.

Because the amount of data is big, it is more convenient to generate maps automatically so that it will fasten the progress of finding movement patterns. According to the three components of map, there is some information needed to be collected before visualizing movement on map. The locations of buildings are collected manually on Google earth based the campus map. These locations are exported as KML file and imported into QGIS. After adding geometry columns x and y, the csv file is created and imported into database. By using *ST\_MakePoint* function, a geometry column is created in database. In summary, the building locations are stored as the structure described in following table:

id	name	geometry	x	y
0	world			
3	science_center	010100000042A7..	4.36939919846287	52.0072322181367
5	tnw_bio	010100000043AE..	4.37120211221402	52.0086132164098
8	bk_city	010100000077E3..	4.37053698152436	52.0056562098059
12	tnw_dct	01010000007CA..	4.36891378927259	52.0040834950037
..	...	....	....	....

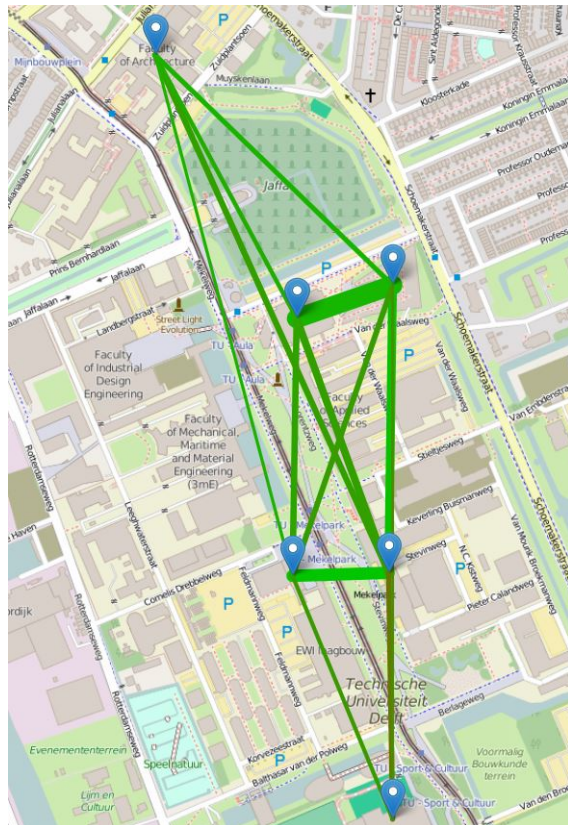
Table 3.1: Building data structure

There is a special 'building' called *world* in the database. It is not an actual location, it is a virtual location which is used if someone is not scanned on the campus in a period of time. After storing the locations of buildings in the database, these locations will be extracted automatically from database to generate maps. There are two properties of lines used to deliver information:

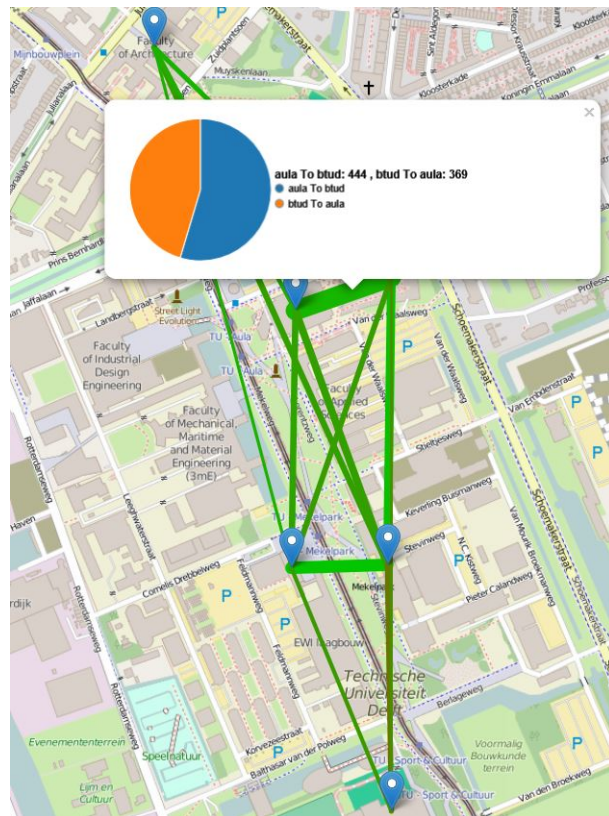
1. width: line width is used to represent the amount of movements, but the amount is aggregated for both directions.
2. color: color is gradient from red to green. Red line means the movement is not symmetric that much more people move in one direction than the other, while green line means the movement is symmetric.

Based on this map visualization, users can choose certain dates and certain buildings to generate maps automatically. It makes it easier to find out movement patterns. Since not all buildings are chosen, the map will only display the movements between several buildings, which makes the map more readable:





(a) Amount of movements on April 25th



(b) Amount of movements in pie chart

As shown in the map, the lines are in different colors, which shows the symmetry of the movements. If the user is willing to know more about the movement, it is also possible to click on the line to check the amount of the movements for each direction in detail, and there will be a pie chart showing how symmetric the movements are. With the map visualization, it is easy to focus on movements which are special or interesting.

### 3.3. Sequences

#### 3.3.1. introduction

The following section describes how movement patterns were derived on campus level, without considering the direction of the movement. An association rule mining algorithm (Agrawal et al., 1993) was used to identify groups of buildings that frequently visited in combination with each other. Firstly the algorithm is described briefly, then the results are presented.

#### 3.3.2. Association rules mining

Association rule mining is a technique to analyse what variables or items are commonly associated with each other in large databases. Probably the one of the main application is to analyse which items are commonly bought together by customers of a supermarket. As an example for this use case is an association rule of an itemset {bread, butter}, tells that in 80% of those transactions including {bread, butter}, also {milk} was present. In other words, 80% of the people who buy bread and butter also buy milk (Agrawal et al., 1993). Compared to sequence mining, association rule mining does not consider the order of items neither within, nor across transactions.

Thus every rule is composed by two itemsets, the *antecedent* {bread,butter} on the left-hand side, and the *consequent* {milk} on the right-hand side. The rule is denoted as {bread, butter} => {milk}.

When a trajectory is simplified into a set of distinct buildings that the person visited, association rules for buildings can be derived. In this case the rule describes the set of buildings, or buildingset, that are commonly visited in combination. For example the rule {BK\_City, Aula} => {Library} tells that a group of people who visited the buildings BK\_City and Aula also visited the Library.

As association rule mining does not consider the order of buildings, nor the time spent in a building, it is important that these variables are appropriately handled and noise is filtered out prior running the algorithm.

In the first version the buildingsets were stored in a table as below, where the field *mac* contains the mac-address of a device and each remaining field represents a building. Value 1 is given if the device was recorded in a building, otherwise no value is given. This binary encoding is rather simplistic as it does not consider the amount of time spent in a building and therefore it does not allow to differentiate between occasional or regular visits.

mac	aula	bk_city	bouwcampus	btud	ctig	...
A	1	1			1	
B			1	1		
C		1			1	
D	1					
E	1		1			

Table 3.2: My caption

Therefore in the second version a distinction between *occasional*, *regular* and *frequent* stays was added to the buildingsets. The division between the categories is based on the 40 hour workweek and 1.5 hour lecture durations (Table X).

Category	hours/week	ID
occasional	$\leq 0.5$	1
regular	$> 0.5, \leq 5$	2
frequent	$> 5$	3

Table 3.3: My caption

The trajectories of approximately 14,000 devices were used to create the first set of association rules with categorized stay duration. At this stage only the noise was filtered from the data but not the stationary devices, and people carrying two devices were not accounted for. The time range of trajectories spanned from 31.03.2016 to 02.05.2016, approximately one month.

Although there are several measures to evaluate the interestingness of an association rule (Zhang et al., 2009), only *support* and *confidence* were used for testing purposes.

**Support** “The support for a rule is defined to be the fraction of transaction in the dataset that satisfy the union of items in the consequent and antecedent of the rule.” (Agrawal et al., 1993). In case of the rule {BK\_City, Aula} => {Library}, the support is the percentage of the total dataset that includes BK\_City, Aula and Library.

**Confidence** Confidence measures the strength of the rule, and is considered as a conditional probability. In case of the rule {BK\_City, Aula} => {Library}, the confidence is the probability that Library is in the trajectory if both BK\_City and Aula are in the trajectory (Agrawal et al., 1993; Anbukkarasy & Sairam, 2013).

The most interesting rules are displayed in Figure 3.6:

Supp ^	Conf	Covr	Strg	Lift	Levr	Antecedent	Consequent
0.02	0.86	0.02	6.92	6.51	0.01	drebbelweg=2, ewi_lb=2 →	ewi_hb=2
0.01	0.74	0.01	24.80	3.38	0.00	btud=2, drebbelweg=2, tbn=2 →	citg=2
0.01	0.70	0.01	30.21	3.21	0.00	aula=2, lr=2, ocp_io=2 →	citg=2
0.01	0.70	0.01	29.22	2.44	0.00	aula=2, lr=2, ogz=2 →	btud=2
0.01	0.72	0.02	6.60	5.49	0.01	btud=2, ewi_lb=2 →	ewi_hb=2
0.01	0.73	0.01	15.17	5.52	0.01	aula=2, btud=2, ewi_lb=2 →	ewi_hb=2
0.01	0.72	0.01	16.24	5.44	0.00	btud=2, citg=2, ewi_lb=2 →	ewi_hb=2
0.01	0.90	0.01	17.31	6.81	0.01	btud=2, drebbelweg=2, ewi_lb=2 →	ewi_hb=2
0.01	0.85	0.01	20.16	6.43	0.00	citg=2, drebbelweg=2, ewi_lb=2 →	ewi_hb=2
0.01	0.74	0.01	10.08	5.59	0.01	ewi_lb=2, tnw_tn=2 →	ewi_hb=2
0.01	0.86	0.01	21.85	6.51	0.00	drebbelweg=2, ewi_lb=2, tnw_tn=2 →	ewi_hb=2
0.01	0.72	0.01	25.66	2.52	0.00	aula=1, tbn=2 →	btud=2

Figure 3.6: Amount of movements in pie chart

In the buildingset of approx. 14,000 devices 2% was recorded in all of the buildings *Drebbelweg*, *EWI-LB*, *EWI-HB* (Support = 0.02). There is an 86% chance that if a device is recorded in the buildings *Drebbelweg*, *EWI-LB*, then it is also recorded in *EWI-HB* (Confidence = 0.86). And they spent on average between half hour to five hours a week in each building (*drebbelweg*=2, *ewi\_lb*=2, *ewi\_hb*=2).

### 3.4. Pre-Processing

Each chapter has its own file. For example, the  $\LaTeX$  source of this chapter can be found in `chapter-1.tex`. A chapter starts with the command

```
\chapter{Chapter title}
```

This starts a new page, prints the chapter number and title and adds a link in the table of contents. If the title is very long, it may be desirable to use a shorter version in the page headers and the table of contents. This can be achieved by specifying the short title in brackets:

```
\chapter[Short title]{Very long title with many words which could not possibly  
fit on one line}
```

Unnumbered chapters, such as the preface, can be created with `\chapter*{Chapter title}`. Such a chapter will not show up in the table of contents or in the page header. To create a table of contents entry anyway, add

```
\addcontentsline{toc}{chapter}{Chapter title}
```

after the `\chapter` command. To print the chapter title in the page header, add

```
\setheader{Chapter title}
```

Chapters are subdivided into sections, subsections, subsubsections, and, optionally, paragraphs and subparagraphs. All can have a title, but only sections and subsections are numbered. As with chapters, the numbering can be turned off by using `\section*{...}` instead of `\section{...}`, and similarly for the subsection.

### 3.5. Entrances

`\paragraph{...}` Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

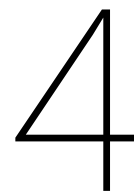
### 3.6. Static and mobile devices

The fonts used by this template depend on which version of  $\LaTeX$  you use. Regular  $\LaTeX$ , *i.e.*, if you compile your document with `latex`, `pslatex` or `pdflatex`, will use Utopia for text, Fourier for math and Latin Modern for sans-serif and monospaced text. However, if you want to adhere to the TU Delft house style, you will need to use  $\XeLaTeX$ , as it supports TrueType and OpenType fonts. Compiling with `xelatex` will use Arial for most titles and text, Courier New for monospace and Cambria for math. If you want to have a sans-serif font for the main text, while using `latex`, `pslatex` or `pdflatex`, you can use the option `noroman` in the report style: `\usepackage[... ,noroman]tudelft-report`. For document and part titles, TU Delft Ultra Light is used. For quotes, columns and text in boxes, you use Georgia. If you want to use  $\XeLaTeX$ , but do not want to use the TU Delft house style fonts, you can add the `nativefonts` option to the document class. This will still use TU Delft Ultra Light and Arial on the cover, but not for the body of the document. If you need to use these fonts for certain sections in the main text, they are available via `\tudrmfamily` (Georgia) and `\tudtitlefamily` (TU Delft Ultra Light).

You have to learn the rules of the game. And then you have to play better than anyone else.

*Albert Einstein*

The corporate colors of the TU Delft are cyan, black and white, available via `\color{tudelft-cyan}`, `\color{tudelft-black}` (which differs slightly from the default `\color{black}`) and `\color{tudelft-white}`, respectively. Apart from these three, the house style defines the basic colors `tudelft-sea-green`, `tudelft-green`, `tudelft-dark-blue`, `tudelft-purple`, `tudelft-turquoise` and `tudelft-sky-blue`, as well as the accent colors `tudelft-lavendel`, `tudelft-orange`, `tudelft-warm-purple`, `tudelft-fuchsia`, `tudelft-bright-green` and `tudelft-yellow`.



# Conclusion

This document is intended to be both an example of the TU Delft  $\LaTeX$  template for reports and theses, as well as a short introduction to its use. It is not intended to be a general introduction to  $\LaTeX$  itself,<sup>1</sup> and we will assume the reader to be familiar with the basics of creating and compiling documents.

Instructions on how to use this template under Windows and Linux, and which  $\LaTeX$  packages are required, can be found in `README.txt`.

## 4.1. Document Structure

Since a report, and especially a thesis, might be a substantial document, it is convenient to break it up into smaller pieces. In this template we therefore give every chapter its own file. The chapters (and appendices) are gathered together in `report.tex`, which is the master file describing the overall structure of the document. `report.tex` starts with the line

```
\documentclass{tudelft-report}
```

which loads the TU Delft report template. The template is based on the  $\LaTeX$  book document class and stored in `tudelft-report.cls`. The document class accepts several comma-separated options. The default language is English, but this can be changed to Dutch (*e.g.*, for bachelor theses) by specifying the `dutch` option:

```
\documentclass[dutch]{tudelft-report}
```

Furthermore, hyperlinks are shown in blue, which is convenient when reading the report on a computer, but can be expensive when printing. They can be turned black with the `print` option. This will also turn the headers black instead of cyan.

If the document becomes large, it is easy to miss warnings about the layout in the  $\LaTeX$  output. In order to locate problem areas, add the `draft` option to the `\documentclass` line. This will display a vertical bar in the margins next to the paragraphs that require attention. Finally, the `nativefonts` option can be used to override the automatic font selection (see below).

This template has the option to automatically generate a cover page with the `\makecover` command. See the next section for a detailed description.

The contents of the report are included between the `\begin{document}` and `\end{document}` commands, and split into three parts by

1. `\frontmatter`, which uses Roman numerals for the page numbers and is used for the title page and the table of contents;
2. `\mainmatter`, which uses Arabic numerals for the page numbers and is the style for the chapters;
3. `\appendix`, which uses letters for the chapter numbers, starting with 'A'.

---

<sup>1</sup>We recommend <http://en.wikibooks.org/wiki/LaTeX> as a reference and a starting point for new users.

The title page is defined in a separate file, *e.g.*, `title.tex`, and included verbatim with `\input{title}`.<sup>2</sup> Additionally, it is possible to include a preface, containing, for example, the acknowledgements. An example can be found in `preface.tex`. The table of contents is generated automatically with the `\tableofcontents` command. Chapters are included after `\mainmatter` and appendices after `\appendix`. For example, `\input{chapter-1}` includes `chapter-1.tex`, which contains this introduction.

The bibliography, finally, is generated automatically with

```
\bibliography{report}
```

from `report.bib`. The bibliography style is specified in `tudelft-report.bst`, which is a modified version of `apsrev4-1.bst` (from REVTeX) designed to also display the titles of referenced articles. The template will automatically generate clickable hyperlinks if a URL or DOI (digital object identifier) is present for the reference. As an example, we cite the paper by Nobel laureate Andrei Geim and his pet hamster [?]. Although it is possible to manage the bibliography by hand, we recommend using EndNote (available from Blackboard) or JabRef (available from <http://jabref.sourceforge.net/>).

## 4.2. Cover and Title Page

This template will automatically generate a cover page if you issue the `\makecover` command. There are two formats for the cover page: one with a page-filling ('bleeding') illustration, with the title(s) and author(s) in large ultrathin typeface, and the other where the illustration fills the lower half of the A4, whereas title(s), author(s) and additional text are set in the standard sans-serif font on a plain background with a color chosen by the user. The last option is selected by the optional key `split`: `\makecover[split]` yields a page with the illustration on the lower half. All illustrations are bleeding, in accordance with the TU Delft style.

Before generating the cover, you need to provide the information to put on it. This can be done with the following commands:

- `\title[Optional Color]{Title}`  
This command is used to provide the title of the document. The title title is also printed on the spine. If you use a title page (see below), this information will be used there as well. As the title, subtitle and author name are printed directly over the cover photo, it will often be necessary to adjust the print color in order to have sufficient contrast between the text and the background. The optional color argument is used for this.
- `\title[Optional Color]{Subtitle}`  
This command is used to provide a subtitle for the document. If you use a title page (see below), this information will be used there as well. It possible to adjust the print color in order to have sufficient contrast between the text and the background – the optional color argument is used for this.
- `\author{J. Random Author}`  
This command specifies the author. The default color is `tudelft-white`, but this may be adjusted in the same way as the titles.
- `\affiliation{Technische Universiteit Delft}`  
The affiliation is the text printed vertically on the front cover. It can be the affiliation, such as the university or department name, or be used for the document type (*e.g.*, Master's thesis). The default color is again `tudelft-white`, adjustable through the color option.
- `\coverimage{cover.jpg}`  
With this command you can specify the filename of the cover image. The image is stretched to fill the full width of the front cover (including the spine if a back cover is present).
- `\covertext{Cover Text}`  
If a back cover is present, the cover text is printed on the back. Internally, this text box is created using the  $\text{\LaTeX}$  minipage environment, so it supports line breaks.
- `\titleoffsetx{OffsetX}, \titleoffsety{OffsetY}` If the cover page contains a page-filling picture (*i.e.*, `split` is not specified with the `makecover` command, the best position of the title depends

<sup>2</sup>Note that it is not necessary to specify the file extension.

a lot on the picture chosen for it. The lower left corner of the minipage containing title, subtitle and author is specified by these two commands. The offsets are measured from the top left corner of the page.

- `\afiloffsetx{AfilX}`, `\afiloffsety{AfilY}` specifies the lower left corner of the text containing the affiliation, measured from the top left corner of the page.

In addition to `[split]`, the `\makecover` command accepts several additional options for customizing the layout of the cover. The most important of these is `back`. Supplying this option will generate a back cover as well as a front, including the spine. Since this requires a page size slightly larger than twice A4 (to make room for the spine), and  $\text{\LaTeX}$  does not support different page sizes within the same document, it is wise to create a separate file for the cover. `cover.tex` contains an example. The recommended page size for the full cover can be set with

```
\geometry{papersize={1226bp,851bp}}
```

after the document class and before `\begin{document}`.

The other options `\makecover` accepts are

- `nospine`  
If a back cover is generated, the title will also be printed in a black box on the spine. However, for smaller documents the spine might not be wide enough. Specifying this option disables printing the title on the spine.
- `frontbottom`  
By default the black box on the front is situated above the blue box. Specifying this option will place the black box below the blue one.
- `spinewidth`  
If a back cover is present, this option can be used to set the width of the spine. The default is `spinewidth=1cm`.
- `frontboxwidth`, `frontboxheight`, `backboxwidth`, `backboxheight`  
As their names suggest, these options are used to set the width and height of the front (black) and back (blue) boxes. The default widths and heights are `4.375in` and `2.1875in`, respectively.
- `x`, `y`  
The blue and black boxes touch each other in a corner. The location of this corner can be set with these options. It is defined with respect to the top left corner of the front cover. The default values are `x=0.8125in` and `y=3in`.
- `margin`  
This option sets the margin between the borders of the boxes and their text. The default value is `12pt`.

For a thesis it is desirable to have a title page within the document, containing information like the thesis committee members. To give you greater flexibility over the layout of this page, it is not generated by a command like `\makecover`, but instead described in the file `title.tex`. Modify this file according to your needs. The example text is in English, but Dutch translations are provided in the comments. Note that for a thesis, the title page is subject to requirements which differ by faculty. Make sure to check these requirements before printing.

## 4.3. Chapters

Each chapter has its own file. For example, the  $\text{\LaTeX}$  source of this chapter can be found in `chapter-1.tex`. A chapter starts with the command

```
\chapter{Chapter title}
```

This starts a new page, prints the chapter number and title and adds a link in the table of contents. If the title is very long, it may be desirable to use a shorter version in the page headers and the table of contents. This can be achieved by specifying the short title in brackets:

```
\chapter[Short title]{Very long title with many words which could not possibly
fit on one line}
```

Unnumbered chapters, such as the preface, can be created with `\chapter*{Chapter title}`. Such a chapter will not show up in the table of contents or in the page header. To create a table of contents entry anyway, add

```
\addcontentsline{toc}{chapter}{Chapter title}
```

after the `\chapter` command. To print the chapter title in the page header, add

```
\setheader{Chapter title}
```

Chapters are subdivided into sections, subsections, subsubsections, and, optionally, paragraphs and subparagraphs. All can have a title, but only sections and subsections are numbered. As with chapters, the numbering can be turned off by using `\section*{...}` instead of `\section{...}`, and similarly for the subsection.

## 4.4. `\section{...}`

### 4.4.1. `\subsection{...}`

```
\subsubsection{...}
```

**\paragraph{...}** Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

## 4.5. Fonts and Colors

The fonts used by this template depend on which version of  $\TeX$  you use. Regular  $\TeX$ , *i.e.*, if you compile your document with `latex`, `pslatex` or `pdflatex`, will use Utopia for text, Fourier for math and Latin Modern for sans-serif and monospaced text. However, if you want to adhere to the TU Delft house style, you will need to use  $\XeTeX$ , as it supports TrueType and OpenType fonts. Compiling with `xelatex` will use Arial for most titles and text, Courier New for monospace and Cambria for math. If you want to have a sans-serif font for the main text, while using `latex`, `pslatex` or `pdflatex`, you can use the option `noroman` in the report style: `\usepackage[... ,noroman]{tudelft-report}`. For document and part titles, TU Delft Ultra Light is used. For quotes, columns and text in boxes, you use Georgia. If you want to use  $\XeTeX$ , but do not want to use the TU Delft house style fonts, you can add the `nativefonts` option to the document class. This will still use TU Delft Ultra Light and Arial on the cover, but not for the body of the document. If you need to use these fonts for certain sections in the main text, they are available via `\tudrmfamily` (Georgia) and `\tudtitlefamily` (TU Delft Ultra Light).

You have to learn the rules of the game. And then you have to play better than anyone else.

*Albert Einstein*

The corporate colors of the TU Delft are cyan, black and white, available via `\color{tudelft-cyan}`, `\color{tudelft-black}` (which differs slightly from the default `\color{black}`) and `\color{tudelft-white}`, respectively. Apart from these three, the house style defines the basic colors `tudelft-sea-green`, `tudelft-green`, `tudelft-dark-blue`, `tudelft-purple`, `tudelft-turquoise` and `tudelft-sky-blue`, as well as the accent colors `tudelft-lavendel`, `tudelft-orange`, `tudelft-warm-purple`, `tudelft-fuchsia`, `tudelft-bright-green` and `tudelft-yellow`.



# 5

## Recommendations

This document is intended to be both an example of the TU Delft  $\LaTeX$  template for reports and theses, as well as a short introduction to its use. It is not intended to be a general introduction to  $\LaTeX$  itself,<sup>1</sup> and we will assume the reader to be familiar with the basics of creating and compiling documents.

Instructions on how to use this template under Windows and Linux, and which  $\LaTeX$  packages are required, can be found in `README.txt`.

### 5.1. Document Structure

Since a report, and especially a thesis, might be a substantial document, it is convenient to break it up into smaller pieces. In this template we therefore give every chapter its own file. The chapters (and appendices) are gathered together in `report.tex`, which is the master file describing the overall structure of the document. `report.tex` starts with the line

```
\documentclass{tudelft-report}
```

which loads the TU Delft report template. The template is based on the  $\LaTeX$  book document class and stored in `tudelft-report.cls`. The document class accepts several comma-separated options. The default language is English, but this can be changed to Dutch (e.g., for bachelor theses) by specifying the `dutch` option:

```
\documentclass[dutch]{tudelft-report}
```

Furthermore, hyperlinks are shown in blue, which is convenient when reading the report on a computer, but can be expensive when printing. They can be turned black with the `print` option. This will also turn the headers black instead of cyan.

If the document becomes large, it is easy to miss warnings about the layout in the  $\LaTeX$  output. In order to locate problem areas, add the `draft` option to the `\documentclass` line. This will display a vertical bar in the margins next to the paragraphs that require attention. Finally, the `nativefonts` option can be used to override the automatic font selection (see below).

This template has the option to automatically generate a cover page with the `\makecover` command. See the next section for a detailed description.

The contents of the report are included between the `\begin{document}` and `\end{document}` commands, and split into three parts by

1. `\frontmatter`, which uses Roman numerals for the page numbers and is used for the title page and the table of contents;
2. `\mainmatter`, which uses Arabic numerals for the page numbers and is the style for the chapters;
3. `\appendix`, which uses letters for the chapter numbers, starting with 'A'.

---

<sup>1</sup>We recommend <http://en.wikibooks.org/wiki/LaTeX> as a reference and a starting point for new users.

The title page is defined in a separate file, *e.g.*, `title.tex`, and included verbatim with `\input{title}`.<sup>2</sup> Additionally, it is possible to include a preface, containing, for example, the acknowledgements. An example can be found in `preface.tex`. The table of contents is generated automatically with the `\tableofcontents` command. Chapters are included after `\mainmatter` and appendices after `\appendix`. For example, `\input{chapter-1}` includes `chapter-1.tex`, which contains this introduction.

The bibliography, finally, is generated automatically with

```
\bibliography{report}
```

from `report.bib`. The bibliography style is specified in `tudelft-report.bst`, which is a modified version of `apsrev4-1.bst` (from REVTeX) designed to also display the titles of referenced articles. The template will automatically generate clickable hyperlinks if a URL or DOI (digital object identifier) is present for the reference. As an example, we cite the paper by Nobel laureate Andrei Geim and his pet hamster [?]. Although it is possible to manage the bibliography by hand, we recommend using EndNote (available from Blackboard) or JabRef (available from <http://jabref.sourceforge.net/>).

## 5.2. Cover and Title Page

This template will automatically generate a cover page if you issue the `\makecover` command. There are two formats for the cover page: one with a page-filling ('bleeding') illustration, with the title(s) and author(s) in large ultrathin typeface, and the other where the illustration fills the lower half of the A4, whereas title(s), author(s) and additional text are set in the standard sans-serif font on a plain background with a color chosen by the user. The last option is selected by the optional key `split`: `\makecover[split]` yields a page with the illustration on the lower half. All illustrations are bleeding, in accordance with the TU Delft style.

Before generating the cover, you need to provide the information to put on it. This can be done with the following commands:

- `\title[Optional Color]{Title}`  
This command is used to provide the title of the document. The title title is also printed on the spine. If you use a title page (see below), this information will be used there as well. As the title, subtitle and author name are printed directly over the cover photo, it will often be necessary to adjust the print color in order to have sufficient contrast between the text and the background. The optional color argument is used for this.
- `\title[Optional Color]{Subtitle}`  
This command is used to provide a subtitle for the document. If you use a title page (see below), this information will be used there as well. It possible to adjust the print color in order to have sufficient contrast between the text and the background – the optional color argument is used for this.
- `\author{J. Random Author}`  
This command specifies the author. The default color is `tudelft-white`, but this may be adjusted in the same way as the titles.
- `\affiliation{Technische Universiteit Delft}`  
The affiliation is the text printed vertically on the front cover. It can be the affiliation, such as the university or department name, or be used for the document type (*e.g.*, Master's thesis). The default color is again `tudelft-white`, adjustable through the color option.
- `\coverimage{cover.jpg}`  
With this command you can specify the filename of the cover image. The image is stretched to fill the full width of the front cover (including the spine if a back cover is present).
- `\covertext{Cover Text}`  
If a back cover is present, the cover text is printed on the back. Internally, this text box is created using the L<sup>A</sup>T<sub>E</sub>X minipage environment, so it supports line breaks.
- `\titleoffsetx{OffsetX}, \titleoffsety{OffsetY}` If the cover page contains a page-filling picture (*i.e.*, `split` is not specified with the `makecover` command, the best position of the title depends

<sup>2</sup>Note that it is not necessary to specify the file extension.

a lot on the picture chosen for it. The lower left corner of the minipage containing title, subtitle and author is specified by these two commands. The offsets are measured from the top left corner of the page.

- `\afiloffsetx{AfilX}`, `\afiloffsety{AfilY}` specifies the lower left corner of the text containing the affiliation, measured from the top left corner of the page.

In addition to `[split]`, the `\makecover` command accepts several additional options for customizing the layout of the cover. The most important of these is `back`. Supplying this option will generate a back cover as well as a front, including the spine. Since this requires a page size slightly larger than twice A4 (to make room for the spine), and  $\text{\LaTeX}$  does not support different page sizes within the same document, it is wise to create a separate file for the cover. `cover.tex` contains an example. The recommended page size for the full cover can be set with

```
\geometry{papersize={1226bp,851bp}}
```

after the document class and before `\begin{document}`.

The other options `\makecover` accepts are

- `nospine`  
If a back cover is generated, the title will also be printed in a black box on the spine. However, for smaller documents the spine might not be wide enough. Specifying this option disables printing the title on the spine.
- `frontbottom`  
By default the black box on the front is situated above the blue box. Specifying this option will place the black box below the blue one.
- `spinewidth`  
If a back cover is present, this option can be used to set the width of the spine. The default is `spinewidth=1cm`.
- `frontboxwidth`, `frontboxheight`, `backboxwidth`, `backboxheight`  
As their names suggest, these options are used to set the width and height of the front (black) and back (blue) boxes. The default widths and heights are `4.375in` and `2.1875in`, respectively.
- `x`, `y`  
The blue and black boxes touch each other in a corner. The location of this corner can be set with these options. It is defined with respect to the top left corner of the front cover. The default values are `x=0.8125in` and `y=3in`.
- `margin`  
This option sets the margin between the borders of the boxes and their text. The default value is `12pt`.

For a thesis it is desirable to have a title page within the document, containing information like the thesis committee members. To give you greater flexibility over the layout of this page, it is not generated by a command like `\makecover`, but instead described in the file `title.tex`. Modify this file according to your needs. The example text is in English, but Dutch translations are provided in the comments. Note that for a thesis, the title page is subject to requirements which differ by faculty. Make sure to check these requirements before printing.

## 5.3. Chapters

Each chapter has its own file. For example, the  $\text{\LaTeX}$  source of this chapter can be found in `chapter-1.tex`. A chapter starts with the command

```
\chapter{Chapter title}
```

This starts a new page, prints the chapter number and title and adds a link in the table of contents. If the title is very long, it may be desirable to use a shorter version in the page headers and the table of contents. This can be achieved by specifying the short title in brackets:

```
\chapter[Short title]{Very long title with many words which could not possibly
fit on one line}
```

Unnumbered chapters, such as the preface, can be created with `\chapter*{Chapter title}`. Such a chapter will not show up in the table of contents or in the page header. To create a table of contents entry anyway, add

```
\addcontentsline{toc}{chapter}{Chapter title}
```

after the `\chapter` command. To print the chapter title in the page header, add

```
\setheader{Chapter title}
```

Chapters are subdivided into sections, subsections, subsubsections, and, optionally, paragraphs and subparagraphs. All can have a title, but only sections and subsections are numbered. As with chapters, the numbering can be turned off by using `\section*{...}` instead of `\section{...}`, and similarly for the subsection.

## 5.4. `\section{...}`

### 5.4.1. `\subsection{...}`

```
\subsubsection{...}
```

**\paragraph{...}** Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

## 5.5. Fonts and Colors

The fonts used by this template depend on which version of  $\TeX$  you use. Regular  $\TeX$ , *i.e.*, if you compile your document with `latex`, `pslatex` or `pdflatex`, will use Utopia for text, Fourier for math and Latin Modern for sans-serif and monospaced text. However, if you want to adhere to the TU Delft house style, you will need to use  $\XeTeX$ , as it supports TrueType and OpenType fonts. Compiling with `xelatex` will use Arial for most titles and text, Courier New for monospace and Cambria for math. If you want to have a sans-serif font for the main text, while using `latex`, `pslatex` or `pdflatex`, you can use the option `noroman` in the report style: `\usepackage[... ,noroman]tudelft-report`. For document and part titles, TU Delft Ultra Light is used. For quotes, columns and text in boxes, you use Georgia. If you want to use  $\XeTeX$ , but do not want to use the TU Delft house style fonts, you can add the `nativefonts` option to the document class. This will still use TU Delft Ultra Light and Arial on the cover, but not for the body of the document. If you need to use these fonts for certain sections in the main text, they are available via `\tudrmfamily` (Georgia) and `\tudtitlefamily` (TU Delft Ultra Light).

You have to learn the rules of the game. And then you have to play better than anyone else.

*Albert Einstein*

The corporate colors of the TU Delft are cyan, black and white, available via `\color{tudelft-cyan}`, `\color{tudelft-black}` (which differs slightly from the default `\color{black}`) and `\color{tudelft-white}`, respectively. Apart from these three, the house style defines the basic colors `tudelft-sea-green`, `tudelft-green`, `tudelft-dark-blue`, `tudelft-purple`, `tudelft-turquoise` and `tudelft-sky-blue`, as well as the accent colors `tudelft-lavendel`, `tudelft-orange`, `tudelft-warm-purple`, `tudelft-fuchsia`, `tudelft-bright-green` and `tudelft-yellow`.

# Bibliography

- Agrawal, Rakesh, Tomasz Imieliński, and Arun Swami (1993). "Mining association rules between sets of items in large databases". In: *ACM SIGMOD Record*. Vol. 22. ACM, pp. 207–216. URL: <http://dl.acm.org/citation.cfm?id=170072> (visited on 04/09/2015).
- Agrawal, Rakesh and Ramakrishnan Srikant (1994). "Fast algorithms for mining association rules". In: *Proc. 20th int. conf. very large data bases, VLDB*. Vol. 1215, pp. 487–499. URL: [https://www.it.uu.se/edu/course/homepage/infoutv/ht08/vldb94\\_rj.pdf](https://www.it.uu.se/edu/course/homepage/infoutv/ht08/vldb94_rj.pdf) (visited on 04/09/2015).
- Anbukkarasy, G. and N. Sairam (2013). "Interesting Metrics Based Adaptive Prediction Technique for Knowledge Discovery". In: *International Journal of Engineering and Technology* 5.3, pp. 2069–2076. URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.411.6332&rep=rep1&type=pdf> (visited on 05/16/2016).
- Zhang, Yuejin et al. (2009). "A Survey of Interestingness Measures for Association Rules". In: *International Conference on Business Intelligence and Financial Engineering, 2009. BIFE '09*. International Conference on Business Intelligence and Financial Engineering, 2009. BIFE '09. IEEE, pp. 460–463. DOI: 10.1109/BIFE.2009.110.