Analyzing timetables using XML: A case study on the University of Twente

Marlène C. Hol
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
m.c.hol@student.utwente.nl

Jochem G.J. Verburg
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
j.g.j.verburg@student.utwente.nl

ABSTRACT

Higher educational institutions in the Netherlands are stimulated to increase the efficient use of rooms, however this makes timetabling more complex. Universities want to be rated well by students on their timetables, however do not know yet the motivations for ratings. The University of Twente has already determined some timetable Key Performance Indicators (KPI), however further analysis of the KPIs and timetables is necessary to improve timetables and facilities in the future.

This paper presents a method to analyze the University of Twente timetable data to extract KPI scores, patterns and trends (in room sizes and locations), and presents the results of this analysis. The method can be re-used for other educational institutions and institutions can compare with the results given in this paper. We observed that effectively analyzing timetable data presents some issues, both in cleaning and getting the right data.

Keywords

Timetables, XML, XQuery, University, Analysis

1. INTRODUCTION

Higher Education in the Netherlands has been stimulated to increase their efficiency the previous years. One way to do so is by increasing the efficient use of rooms. However, this leads to more problems in timetabling whereas good timetables are essential for education. If students' lectures conflict or when there is no lecture hall for a lecture, quality of education decreases drastically. The University of Twente (UT) currently struggles with getting positive feedback on their timetabling, for example in student surveys [5].

The UT is currently researching how it can improve the satisfaction of all stakeholders. Part of this research is trying to find out how it scores on its own Key Performance Indicators (KPI) to be able to analyze its improvement potential. Besides this, it also wants to find out any patterns or trends in its timetabling data that might influence the current or future results.

We analyzed the timetable data to be able to extract KPI scores, patterns and trends so timetablers can analyze the effectiveness of changes in the timetabling process. We looked at patterns and trends in the location and sizes of the rooms. This paper describes both the analysis method used, through transforming the data into XML, and the results gotten for the University of Twente.

The next section will discuss related work in the areas of timetabling and XML analysis. Then, we will describe the method used and some of the problems that had to be dealt with. Afterwards this paper presents the results and discusses these results. The paper ends with a conclusion.

2. RELATED WORK

Some related work already exists, both in the area of timetabling and the area of XML analysis. One paper discusses five main classes of constraints which timetablers have to deal with [6], which are good to keep in mind when analyzing the data. One of these classes are capacity constraints. We looked into patterns and trends regarding the capacity of the used rooms. Other papers to get an idea of what has already been researched at the UT have also been included in the references [3] [7]. One paper discusses how to make the timetables measurable (in Dutch) [3], the other is an attempt to improve the timetabling process taking into account the KPIs [7].

Another interesting paper discusses the analytical processing of XML documents and the opportunities and challenges especially in respect with regular Online Analytical Processing (OLAP) [2]. It discusses how to generate a multidimensional XML document, however we decided due to the complexity to go for the second approach the paper mentions to use XML data: store data as XML documents, query the document and then export it to another tool for data analysis (in our case, Excel). Another paper also discusses techniques to do multidimensional analysis of XML warehouses, which might be useful to transform the XML data into in the future, to be able to better analysis on the timetable data [8].

The last paper tries to mine association rules from XML, which could be useful to find correlations between e.g. room size and usage [9]. However, the first simple queries proved to be already very slow and limited by main memory. Therefore we chose to do analysis in a more traditional way by generating the results we needed using XQuery and analyzing them in Excel. This way, the university's timetablers can also very easily do analysis themselves without needing high-end computing resources.

3. MATERIALS AND METHODS

To provide timetablers with information to improve their timetables, we calculated the KPI scores and analyzed the data for trends and patterns. We mainly looked at the location and sizes of the rooms used for trends and patterns, since we believe interesting things might be found there.

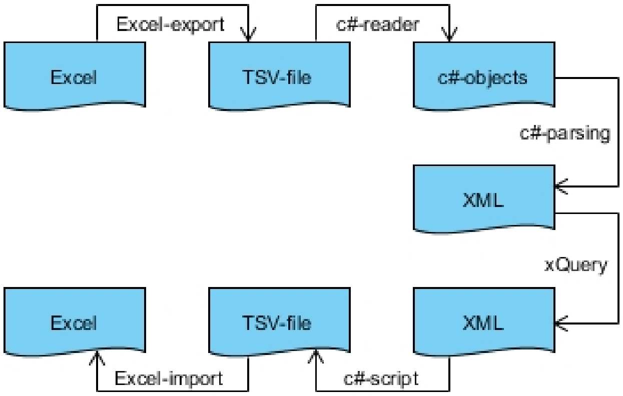

Figure 1: Conversion steps in analysis method

To analyze the data, we loaded the data from Excel into an XML database using a C# script. The C# script first transformed the data into objects, to then parse it as XML, resembling an approach using SQL/XML [4]. Then, we used XQuery to query this database for the needed information and transformed it back into Excel to be able to analyze the information. See Figure 1 for an overview, all XML files can easily be re-used and extended for further analysis. You can find the scripts in the repository[1].

There are some good reasons to transform the data into XML. One of the most important reasons is that it is really flexible and therefore can be easily extended to comply with new scheduling information. Due to this it also lends itself really well for combining data from multiple sources. This also makes it easier to adapt the methods used by other institutions. Two other reasons are that XML is very easy for humans to read, and also a format which makes it easy to distribute the data further.

The next two sections will describe some of the challenges when transforming to XML and while executing queries.

3.1 Transforming to XML

As described earlier, Figure 1 shows that the useful data, which was made available by the University of Twente, was first converted from Excel to TSV files. The excel files used were ActivitiesUT_*_v2 and UT_Overview classrooms and facilities 20150609, where * was respectively the college years 2013_2014, 2014_2015, and 2015_2016. Next, these TSV files were used as input for the C# script. We used C# instead of the original SQL/XML approach because the predefined objects in C# were easily adaptable and the entire approach was a lot faster than when using SQL. Because of these benefits, it was easy to clean the data and change the parsing when data was not correct.

The C# script consisted of two parts, the first part extracted the rooms and their capacity from the TSV file of the rooms, the second part converted the TSV files of the college years in C# objects. The rooms were saved in a list of tuples, where every tuple consisted of the abbreviation of the room (e.g. RA 1501) used in the schedules and the capacity of the rooms. The abbreviations of the room are also used in the data of the college years, therefore the capacity could be linked to this data. For the data of the college year we created a template for C# objects (ActivitiesUT) where the relevant data from the TSV files could be saved. The relevant information could be extracted by parsing the TSV files, since every line in the TSV file represented one activity. The structure chosen for the C# object was already the basis for the XML schema.

After creating the C# objects, these objects could be parsed to XML with C#. Because of the predefined C# object the information could easily adapted to XML. If the element was not included in the original Excel file, the element was also not included in the XML. If the room from the college year files was also included in the room file, the capacity of that room is also included in the XML. This transformation let to the XML of Listing 1. There are a few choices made to create this XML. First, an attribute is included to show whether this activity is part of a TOM module. This is useful for determining the KPIs. Besides that, this XML allows more than one study or quartile per activity. Some courses may be semester courses or shared with different studies and with this approach, you don't have duplicates in the final XML. In the XML we also split up the course information in different fields, whereas this was only one field in the Excel file. With this approach, also some extra information can be used. Finally, we split up the event in a start and an end, which include both a date, the hour, minute and second. With this approach, it is easier to calculate how long the activities took, which is useful for determining the KPIs. The events also contain the requested size, the room and the capacity, which was included from the other excel file. This approach makes the XML-file easily adaptable to include extra information about activities without interfering with the existing data.

Listing 1: XML of an activity

```
<Activity>
  <StudyYear>20132014</StudyYear>
  <TOM>false</TOM>
  <Studies>
    <Study>IDE</Study>
  </Studies>
  <ModYear>M</ModYear>
  <Quartiles>
    <Quartile>2B</Quartile>
  </Quartiles>
  <Course>
    <Name>Product Life Cycle Management</Name>
    <CourseModCode> 192850750</CourseModCode>
    <ActivityType>OVO</ActivityType>
    <Session>OVO01/01</Session>
    <Description>Product Life Cycle Management 19285
  </Course>
  <Event>
    <Start>
      <Date>2014-4-25</Date>
      <Hour>8</Hour>
      <Minute>45</Minute>
      <Second>0</Second>
    </Start>
    <End>
      <Date>2014-4-25</Date>
      <Hour>17</Hour>
      <Minute>30</Minute>
      <Second>0</Second>
```

[1] https://github.com/jochemverburg/DataScience2016

```
        </End>
        <RequestedSize>28</RequestedSize>
        <Room>WH 215</Room>
        <Capacity>28</Capacity>
    </Event>
</Activity>
```

Data cleaning

When transforming the Excel files to XML data, we faced some data uncertainty which made it hard to correctly parse the data. First, when converting the excel files to TSV files by using Microsoft Excel 2016, some lines were split up over several lines. When parsing the TSV files, the data is split on the tab to extract several fields. However, when the line is split up over several lines, this approach is impossible. We discovered that when this happens, quotations were put at the beginning and the end of that field. Since these quotations were on two separate lines now, we were able to look at odd number of quotations and concatenate these lines. Another problem we faced was when determining the course or module code. When an activity was part of a TOM module, the fourth field contained the module code, otherwise the course code was saved in the third field. Whether the activity was part of TOM was based on the field ModYear, which contains to which module or to which phase of the study programme the course belonged. However, some activities are not part of a module or course, for example bachelor graduation, or another format for the data was used. For these cases, it was very hard to determine whether or not the activity was part of TOM, since the data structure was different. Therefore, the wrong course or module code was usedin these cases. There were also some where other data fields may be wrong. We tried to cover as much of these edge cases as possible, by looking into the files and the results of earlier queries. However, it is almost impossible to cover all these edge cases and therefore there may be still some uncertainty left.

Another problem we discovered was when parsing the rooms. Often several names or abbreviations are used for the same room or sometimes an extension was included which was not used at another line. The clearest example of this was with the rooms in the 'Vrijhof'. In the past years, some of the rooms there were renamed. Therefore, there was an entry with the name "VR 2.07 (Library)", "VR 275H (previously VR 2.07) (Library)", and "VR 275H (Library)" in the data. Of course, this is all the same room, but when parsing they were assumed to be different rooms. We tried to solve as many as possible of these cases by renaming certain rooms to the same name. Also in this case, we might have missed some of these cases and some data uncertainty is still left.

3.2 Executing queries

The queries on XML were executed using XQuery. Several problems arose when making the queries to get useful results. Some might be university-specific, some domain-specific and others were related to XQuery and the tool used to run queries, BaseX [1]. For all queries, see the repository[2].

XQuery-related issues

One of the first problems experienced when trying to analyze the KPIs, was the instability of BaseX. The simplest queries would sometimes give an 'Out of Main Memory'-error and other times run without problems. Besides this, some queries took really long to run after which BaseX sometimes crashed. To solve this issue we used C# to run some queries after using a test set in BaseX. This took a bit longer but the results could immediately be saved to a file and the used memory could also be enlarged so the 'Out of Main Memory'-error was not given. Another way to solve that queries took long to run was to make an intermediate file for e.g. the KPIs regarding students, where the dimension was already changed to the right dimension and the results were filtered. Therefore, using these intermediate files, the queries would run faster.

An important finding during the process trying to improve both memory usage and execution time of the queries was that using a 'group by'-statement instead of nested queries improved memory usage. The disadvantage of this is that it delivers less nested XML, which often have more meaningful semantics. The next code snippet gives an example first of a nested XQuery, then an example using a 'group by'-statement. This shows that the former can add additional semantics by nesting XML, whereas the latter can only have separate combinations of $studyYear and $code. However, for the purposes of analyzing the data in Excel, the data would already have to be transformed to rows and therefore the loss of semantics was no problem.

```
let $years :=
(
for $year in
distinct-values($doc/Activities/StudyYear)
let $codes :=
(
for $code in
distinct-values
(
$doc//CourseModCode[ancestor::StudyYear=$year]
)
return
<mod code="{$code}"></mod>
)
return
<studyYear year="{$year}">{$codes}</studyYear>
)
return
<category>{$years}</category>

for $activity in $doc/root/Activity
group by
$studyYear := $activity/StudyYear,
$code := $activity/Course/CourseModCode
return
<category year="{$studyYear}" code="{$code}">
  {$activity}
</category>
```

Domain-specific issues

One of the domain-specific issues was that in case of universities it is hard to calculate KPIs regarding students, since there is a huge amount of combinations of courses possible for them. Since it was hard to extract from the data which combinations were allowed, this problem was solved in this case by using some university-specific knowledge. In gen-

[2] https://github.com/jochemverburg/DataScience2016

eral, at the UT, students who follow TOM Modules only follow lectures of that specific module and not of others. Therefore, we calculated the students' KPIs by looking at the TOM-modules. However, this might mean that schedules are optimized for TOM students and master students might feel left out.

Another problem arising when trying to calculate the KPIs for students is courses that have multiple lectures of which each student only has to attend one. E.g. if there are multiple groups for practicals. Often these are planned at the same time with multiple teachers, therefore we filtered out all multiple lectures of a course at the same time. However, it is also possible that these take place at different times, which has currently not been solved since the expected influence on the results is low and to solve this more information from the timetablers would be needed.

For calculating the occupation of rooms also some ambiguity is encountered. A lot of universities and other educational institutions will have higher occupations during the day than at night. Therefore, we made the query such that the beginning and end of the day to take into account can be set easily in a variable. Then the query will only take into account occupation during those hours.

University-specific issues

Some specific issues for the UT were also encountered, however these might also apply to other educational institutions. Not only can a course have been scheduled at multiple places at one time, it was even the case that multiple courses were planned in one lecture hall at the same time, sometimes erroneous but sometimes probably planned. These had to be filtered out when calculating occupation, otherwise the occupation would become higher than 100%.

Another often encountered issue was the lack of data. In this case, the activities were not taken into account in the XQueries if relevant data was missing (e.g. no time or no room). This could of course have led to wrong results. However, more probable is that these activities in the end never took place and therefore should also not have been taken into account.

Other university-specific issues regarding transforming data have already been discussed in Section 3.1.

4. RESULTS & DISCUSSION

Several different results were gotten. The results will be discussed in this paper, but you can also further explore the results using the Excel-files in the repository[3]. First, the KPIs of the university were calculated and analyzed. Afterwards, we will discuss the patterns and trends found regarding the sizes of the rooms that were used. Last, the location of rooms in relation to the home base of studies have been analyzed.

4.1 KPIs

First, the KPIs of the university were calculated. See Table 1 for the KPIs of the University of Twente, which were given to us.

We did not have the information to calculate the teachers' KPIs and therefore focused on the others. Besides this, it is unknown in the data available which student follows which courses. Therefore, we used some domain knowledge, which is calculating the students' KPIs for TOM modules (a type of course within the UT), and treat each TOM module as if it were followed by one student. In general namely, all students at the UT follow one TOM module per quarter. In all cases we only looked at the KPIs for days on which a course had taken place (for a module, or in case of the rooms' KPIs in a room). This causes some bias, however gives us a better idea of the KPIs then taking into account all days (e.g. a room not being used on a normal work day would not decrease the occupation, even though it should be).

Students	1. Students have a minimum of 4 contact hours on any day
	2. Students have a maximum of 6 contact hours on any day
	3. Students have a maximum of 2 free hours in 1 series on any day
	4. The timetable of students have a maximum of 11 college hours on any day. This means 8:15 clock hours, which is the time between start of the first college and the end of the last college on any day)
	5. If a student has a class at the 11th and 12th college hour, then that student has no class at the 1st and 2nd college hour the next day
	6. At Fridays there are no evening classes
Teachers	1. A teacher has a maximum of 8 contact hours per day
	2. If a teacher has a class at the 11th and 12th college hour, then that teacher has no class at the 1st and 2nd college hour the next day
Rooms	1. Rooms must have an occupation of at least 70%. Occupation is defined as follows: occupying a space (room) by the timetabling process during educational weeks.

Table 1: The KPIs of the University of Twente

The results have been calculated per study year as can be seen in Figure 2. Of the KPIs where this was useful, the KPIs have also been calculated per month, see Figure 3 and 4. Students KPI 5 and 6 have not been shown over time since they were in general passed and therefore this wouldn't show any useful information. In all cases, one has to take into account that during the summer months some unexpected increases and decreases can take place since there are only a few lectures during those months (especially in August).

Students KPI 1 + 2

The first two KPIs were calculated separately, but also combined since they are closely related (if KPI 1 is not fulfilled, KPI2 is per definition fulfilled and the other way around). Combined, students should have between 4 and 6 contact hours per day. If students did not have any contact hours, this was not counted as a pass, neither as a fail, thus not

[3] https://github.com/jochemverburg/DataScience2016

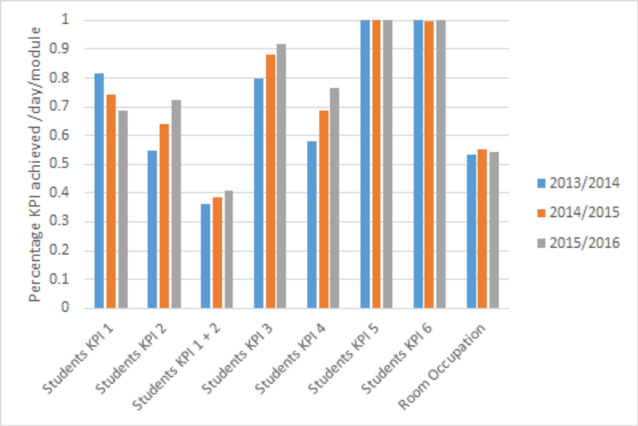

Figure 2: KPI results per study year

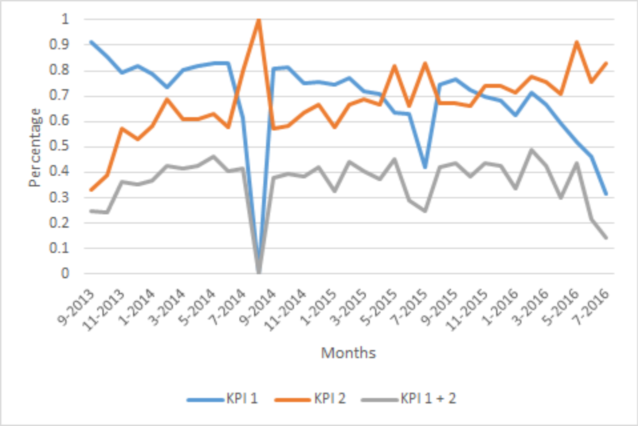

Figure 3: KPI 1 + 2 over time

influencing the percentage. Both Figures 2 and 3 show a slow decrease of KPI 1 and a slow increase of KPI 2 and the combined KPIs.

This could possibly be explained by the fact that these years also represent the years that TOM modules were introduced. In the first year, the first year of studies started having TOM modules, in the second year the second and in the third year the third. Possibly, TOM modules in higher years have less lectures and therefore fulfill KPI 2 more often and KPI 1 less often. This will have to be looked into by the timetablers themselves, since we do not have available which year the courses belong to.

Students KPI 3 + 4

Both these KPIs show an increase in the scores. The increase of KPI 4 might also be explained, like KPI 1 + 2, by the addition of second and third year courses which have less lectures. This must be verified however.

The increase of KPI 3 does not seem to have any other explanation than that the timetablers have become better at scheduling lectures for certain students together. Now, using this data they can be rest assured that actions they've undertaken to improve their schedules have paid off.

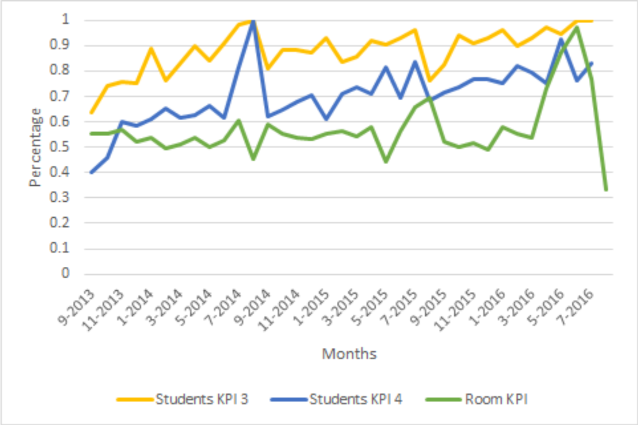

Figure 4: Other KPIs over time

Students KPI 5 + 6

The KPIs 5 and 6 for students show very good scores. This can be explained since not a lot of evening lectures are currently given in TOM modules. Students that do have evening lectures will probably not be regular students and therefore never have lectures in the morning. In total KPI 5 was failed once in year 2013/2014, and KPI 6 was failed three times in the study year 2014/2015, giving scores close to 100%.

Room occupation

As can be seen in both Figure 2 and 4, the room occupation seems to be fairly constant. The room occupation has been calculated during regular lecture hours (from 8:45 till 17:30), where each lecture hour was counted fully (so a lecture of 8:45 till 12:30 gives an occupation of 4/9). Figure 4 shows some increases in the summer periods. This can be explained by the method of calculating this KPI, only taking into account the days that a lecture was given in a room. Days without lectures in a certain room are expected not be work days, or days the room was out of order. Probably, in the summer, the same room was often used at one day, not to switch around the students too much.

4.2 Room sizes

As introduced, we tried to find out any patterns or trends when it comes to the sizes of the lecture halls. We got a file from the timetablers including for some of the rooms the sizes, however not all rooms were included (just 74 rooms with known capacity, 243 without). This might introduce a bias in the results.

In Figure 5, for all rooms the capacity was plotted against the number of bookings. It seems like the smaller rooms are used more. The average number of bookings per lecture hall was 1096. The larger rooms seem to be beneath this average more often. However, calculating the correlation using Excel, a correlation of 0.123129 was found, which is not really significant. Therefore, one can conclude that the current room capacities reflect the needs really well, and no immediate investments in rooms of different capacities are needed.

Figure 6 shows the amount of requests over time for different ranges of capacities. It is shown till March 2016, since

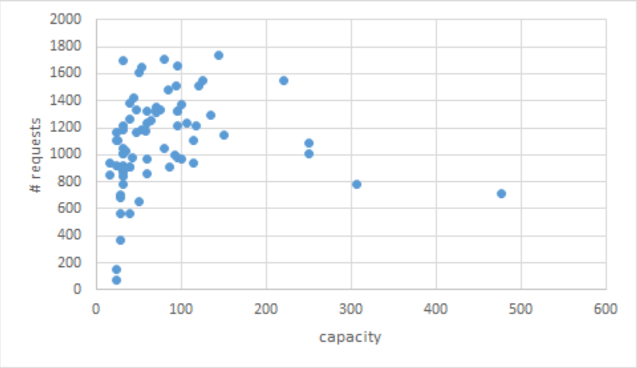

Figure 5: Usage vs capacity

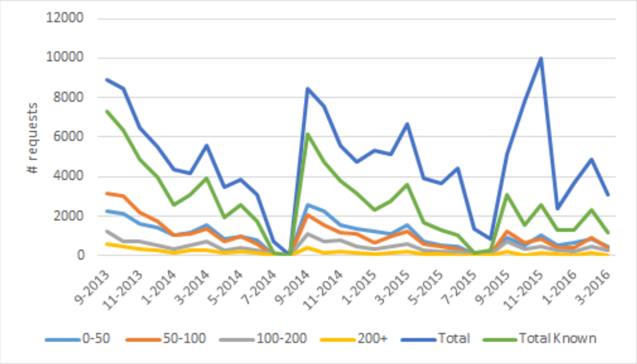

Figure 6: Usage over time divided in capacity ranges

the lectures afterwards had not been planned extensively yet at the time of analysis and are therefore not representative. The first thing one can see is that the amount of request for rooms with a known capacity has drastically decreased in the year 2015/2016, apparently other rooms have been used more in this year. Some other observations that can be made is that there's a peak every time during the first months, so in the first quarter it seems more lectures are given. Besides this, the usage of the smaller rooms (0-100) differs a lot each year, the fluctuation seems larger in these rooms. Currently, the results don't show a real change in the needed capacity. However, in future this query can be repeated to see if the capacity needs change to make decisions about what kind of rooms to build.

4.3 Room locations

Another set of data that was looked into was to see how the locations of rooms are related to certain studies. This information might be compared by timetablers with the ratings students give their timetables. It is especially interesting to see whether there's a relation between the home bases and the locations of lecture halls.

Using our knowledge of the UT studies, we made a list of all studies' study associations and where their offices are. Then we could filter in the data based on home base, to see whether there's a pattern. For the studies in the building Zilverling, we also took Ravelijn and Hal B (the neighboring buildings) as home base since there are almost no lecture halls in the Zilverling.

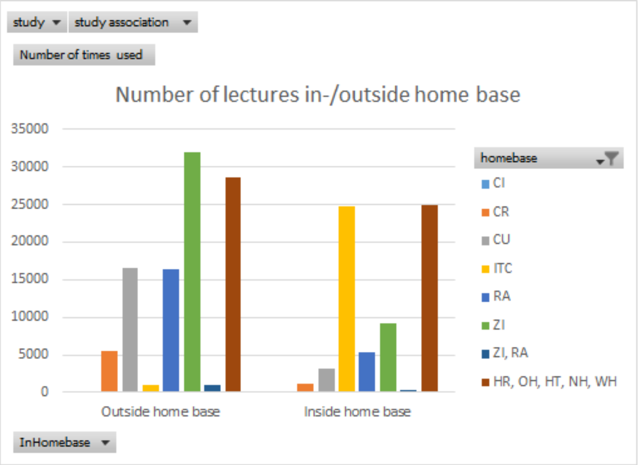

Figure 7: Number of lectures in-/outside homebase

Figure 7 shows how many times lectures took place inside and outside the home base, per home base. Overall, more lectures take place outside the home base than inside (100,824 vs. 68,715). As expected, studies in ITC almost never take place outside ITC. Besides this, it shows that the studies with home base in the Horst have almost half of their lectures in the Horst. However, one might expect this would be more. The other studies seem to have approximately the same ratios (between 3:1 and 5:1). It is therefore interesting to see whether these studies' students rate their timetables worse than the studies in ITC and Horst. Of course, there is a lot of bias in this results, since the number of rooms usable for lectures in the home bases differ a lot.

Figure 8 shows the lectures in- and outside home base over time. It makes clear that the last year more lectures have been given inside the home bases, however this has mainly been caused by a lot more lectures scheduled in ITC. Without ITC, the ratios of lectures in- and outside the home base stay approximately the same, with peaks at the same moments.

It seems that there is still a lot to improve when it comes to putting lectures in the home base of studies. If possible timetablers can for example try to put all Horst-studies in the Horst as much as possible, using the space in other buildings to accommodate the other studies. This would probably increase the satisfaction of students with the facilities and timetables.

5. CONCLUSION & FUTURE WORK

Analyzing timetable data can give useful insights in how to improve the timetables. However, even though one would think timetable data is pretty standardized, as in a lot of other data one has to do a lot of data cleaning. Besides this, it is a challenge to get all data to effectively analyze timetables (such as the capacity of all rooms). The paper shows how one can overcome these problems to effectively analyze the data.

The results in this paper can both be used by timetablers of the UT and of other educational institutions. It shows how to clean data and how one can analyze the data, even

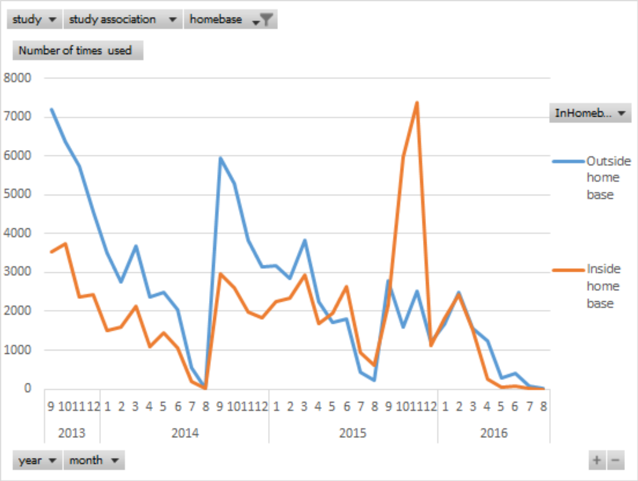

Figure 8: Number of lectures in-/outside homebase over time

though not all information might be available (for example by using TOM modules as a substitute of students). Besides this, the results give the opportunity to track the effectiveness of changes in the timetabling process (e.g. analyzing upcoming timetables even before they're published), however one needs to take into account other factors that might influence the results (e.g. the improvement of KPI 2 and 4 might be explained by also having second and third year TOM modules).

The results first of all show that the UT in general is doing a good job in improving their KPIs, the method presented in this paper can help them to continuously keep track of the scores. Secondly, the results regarding room sizes show that currently there is no urgent need for more rooms of certain sizes. However, it is interesting for all educational institutions to keep track of the correlation and trends to be able to anticipate early on when new smaller or larger rooms are needed. Last, the results show that students at the UT still have to go outside of their home base a lot. To increase satisfaction it might be good to cluster lectures of students around their preferred location (probably, their home base).

In the future, the UT should try to standardize the data more and make sure the data can be made available in a cleaner way, to further improve analysis. Besides this, it would be good if data can be added so that the XML-files can be extended. Furthermore, it might be interesting to invest in a system that will run analysis for the timetablers, without disturbing their work, especially since the used tools had high memory usage and long query times. For other timetablers, the system can be reused to analyze their own timetables and the results can be used to compare their own timetables to.

For future work, it is interesting to see how the results compare to the timetables' ratings of students. Besides this, it might be good to analyze which relevant data should be collected to perform effective analysis. Other research should focus more on the technology, how to improve the performance of the queries and transformations. If the performance is improved, it will become easier to do a lot of analysis also when making new timetables.

6. REFERENCES

[1] BaseX. Basex. the xml database. http://basex.org/, 2015. Accessed: 2016-03-08.

[2] R. Bordawekar and C. Lang. Analytical processing of xml documents: Opportunities and challenges. *SIGMOD Record*, 34(2):27–32, 2005.

[3] A. Dijksterhuis. Kwaliteit roosters universiteit twente-onderzoek naar het meetbaar maken van de prestaties van het rooster op universiteit twente. *University of Twente*, 2014.

[4] A. Eisenberg and J. Melton. Sql/xml is making good progress. *SIGMOD Rec.*, 31(2):101–108, June 2002.

[5] Elsevier. Elseviers beste studies. http://onderzoek.elsevier.nl/onderzoek/beste-studies-2015/17/overzicht, 2015. Accessed: 2016-03-26.

[6] R. Lewis. A survey of metaheuristic-based techniques for university timetabling problems. *OR Spectrum*, 30(1):167–190, 2008.

[7] F. Meijer Cluwen. Dynamic room allocation-adaptive planning of teaching facilities at the university of twente. *University of Twente*, May 2015.

[8] B.-K. Park, H. Han, and I.-Y. Song. Xml-olap: A multidimensional analysis framework for xml warehouses. *Lecture Notes in Computer Science*, 3589:32–42, 2005.

[9] J. Wan and G. Dobbie. Extracting association rules from xml documents using xquery. pages 94–97, 2003. Conference of WIDM 2003: Proceedings of the Fifth ACM International Workshop on Web Information and Data Management.