

Saarland University

Department of Computer Science

Master thesis

Development of a Learning Analytics framework for
Intelligent Tutoring Systems

submitted by

Revathy Santhanam

submitted

xx.11.2016

Supervisor

Prof. Dr. Jörg Siekmann

Advisor

Dr. Sergey Sosnovsky

Reviewers

Prof. Dr. Jörg Siekmann

Priv.-Doz. Dr. Christoph Igel

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides Statt, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Statement in Lieu of an Oath

I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis.

Einverständniserklärung

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Revathy Santhanam

Saarbrücken, xx.11.2016

---o---

to my
mother, father,
sister and my
husband
for all their
unconditional
love and moral
support

Acknowledgements

I would like to express my gratitude to my advisor, Dr. Sergey Sosnovsky, for all the time he spent with me and giving me valuable inputs, moral support and guidance. He gave me the advice and encouragement that I needed to successfully complete this thesis.

I would also like to thank Mr. Thorsten Hey from the Activemath group, who gave me the needed motivation and spent time in helping me when I had issues and the patience he had to answer all my questions. I would like to thank the entire team for supporting me all the while.

I am also grateful to Prof. Dr. Jörg Siekmann, for being my supervisor and having provided me with the opportunity and guidance to pursue this thesis.

I must also thank everyone at the Computer Science department of Saarland University who always helped me when I needed.

Last but not the least; I would like to wholeheartedly thank my family and specially my husband without the support of whom this journey would not have been possible. They have been there for me day in and out, consciously providing me with all what I needed to accomplish this milestone in my life.

Abstract

Learning Analytics provides better learning platform by finding creative ways for the students and helping in providing better opportunities for the students to overcome their learning issues. It helps in improving the level of satisfaction a student can derive from an intelligent e-learning system by providing good visualization criterion for the teachers and tutors which aids in the student's progress and helps them decide on their courses and learning content, thereby enhancing the teaching level as well.

In this Thesis, a Learning Analytics Framework is developed for Analytics Reports on Math-Bridge. This Framework is useful in visualizing the Exam Results of the students using Math-Bridge. The precursor to developing this LA Framework was to migrate all the data and structures from Derby to MySQL since the legacy system issues in Math-Bridge. This in turn increased the System's Performance comparatively.

The Math-Bridge tutoring system is made effective with the development of this Learning Analytics Framework which can be enhanced to fit the entire system and provide better and faster results to the users due to non-complicated code structure and use of Java Platform; this Framework presents at hand an efficient and effective system in place, which can take over other such e-learning systems in future.

Table of Contents

Chapter 1	18
Introduction	18
1.1 Motivation.....	18
1.2. Problem Description	23
1.3. Summary of the approach.....	23
1.4. Structure of the thesis	24
Chapter 2	25
Related technologies	25
2.1 Intelligent Tutoring Systems.....	25
2.2 Learning Analytics.....	29
2.3 Educational data storage	31
Chapter 3	36
Math-Bridge	36
3.1 Short description	36
3.1.1 Existing systems Vs Math-Bridge	36
3.1.2 Math-Bridge Content	36
3.2 Architecture.....	38
3.3 Reporting Tool	42
Chapter 4	48
Migrating Math-Bridge.....	48
4.1 Storing educational data with Derby	48
4.1.1 Derby-an Insight.....	48
4.1.2 Math-Bridge Data organization in Derby.....	50
4.2 Problems	51

4.3 Exported MySQL data base	51
4.3.1 MySQL-an Insight	51
4.3.2 Exported MySQL data.....	53
4.4 Migration procedure	53
4.5 Evaluation	62
Chapter 5	66
Learning Analytics Framework	66
5.1 Architecture.....	66
5.2 Example Report – 1	69
Chapter 6	73
Conclusion and Discussion	73
6.1 Contribution of this thesis.....	73
6.2 Limitation of this thesis	73
6.3 Future work.....	74
References	75

List of Tables

1.1: LO Types in Math-Bridge	19
---	-----------

List of Figures

1.1: procedure of content transformation	18
1.2: LA Iterative	21
2.1: shows features of the ITS	25
2.2: shows the Adaptive Learning Systems components	27
2.3: LA Process Cycle	29
2.4: Dimensions of LA	30
2.5: shows a chart of the average Error Rate	33
2.6: depicts the learning curve of a student with the error rate	34
3.1: shows the Math-Bridge content	36
3.2: Course generation in Math-Bridge	37
3.3: Math-Bridge Architecture Flow	38
3.4: Layered Architecture of Math-Bridge	41
3.5: Math-Bridge teacher interface: reporting tool	42
3.6: Math-Bridge Start Page in Spanish	43
3.7: Math-Bridge Start Page in English	44
3.8: Math-Bridge Course in German	45
3.9: Math-Bridge Course in English	46

4.1: Derby access through JVM	48
4.2: Connect to Derby from an IDE	49
4.3: MySQL storage of Data	51
4.4: JDBC connection to MySQL	51
5.1: Difference between Old and New approaches	66

Chapter 1

Introduction

1.1 Motivation

Math-Bridge is an intelligent adaptive e-Learning platform for online courses in mathematics. It was developed in order to help students with the courses taught in Universities in Europe. Math-Bridge is multi-lingual technology-based educational solution and has access to the biggest collection of semantically annotated learning objects (LOs) for mathematics. Math-Bridge provides ways to create courses that can be customized, provides support to student using student's models knowledge, thereby enabling a better means of education and effective learning techniques.

Math-Bridge also allows the tutors and teachers to help the students and track their individual progresses with the reporting tools. Based on their requirements, the students/teachers can handpick and design the courses in order to fill in the gaps of students who want to improve in a particular area/multiple areas also repeat the exercises when necessary. Math-Bridge offers a complete solution for organizing technology-enhanced learning (TEL) of mathematics on individual-, course- and/or university level.

Math-Bridge helps in addressing the problems by applying a range of techniques from the fields of Intelligent Tutoring Systems, Adaptive Hypermedia, Semantic Web and TEL. The result of these efforts is the e-Learning platform for bridging courses with a number of unique features. It brings together content from different European sources and offers it in a unified way.

Math-Bridge has works on these objectives: 1. To collect and harmonize high-quality remedial content developed by experts in bridging-level mathematics and make this content Web-accessible; 2. To enable cross-cultural and multi-lingual presentation of this content, thus,

promoting its reuse across the borders; 3. To motivate the technological reuse of the content by implementing it in a shareable format and enriching it with metadata based on open standards; 4. To offer different types of personalized access to the content, thus supporting multiple usage scenarios of the platform: from individual exploratory e-Learning to classroom-based knowledge training and testing; 5. To foster the platform adoption by increasing its usability not only for students, but also for other stakeholders, including teachers and university officials. [6]

Figure 1.1 presents the complete procedure of content transformation, step-by-step. The resulting database of remedial content is available as a collection of individual LOs, transformed into the OMDoc format for mathematical documents (Kohlhase 2006) and provided with metadata.

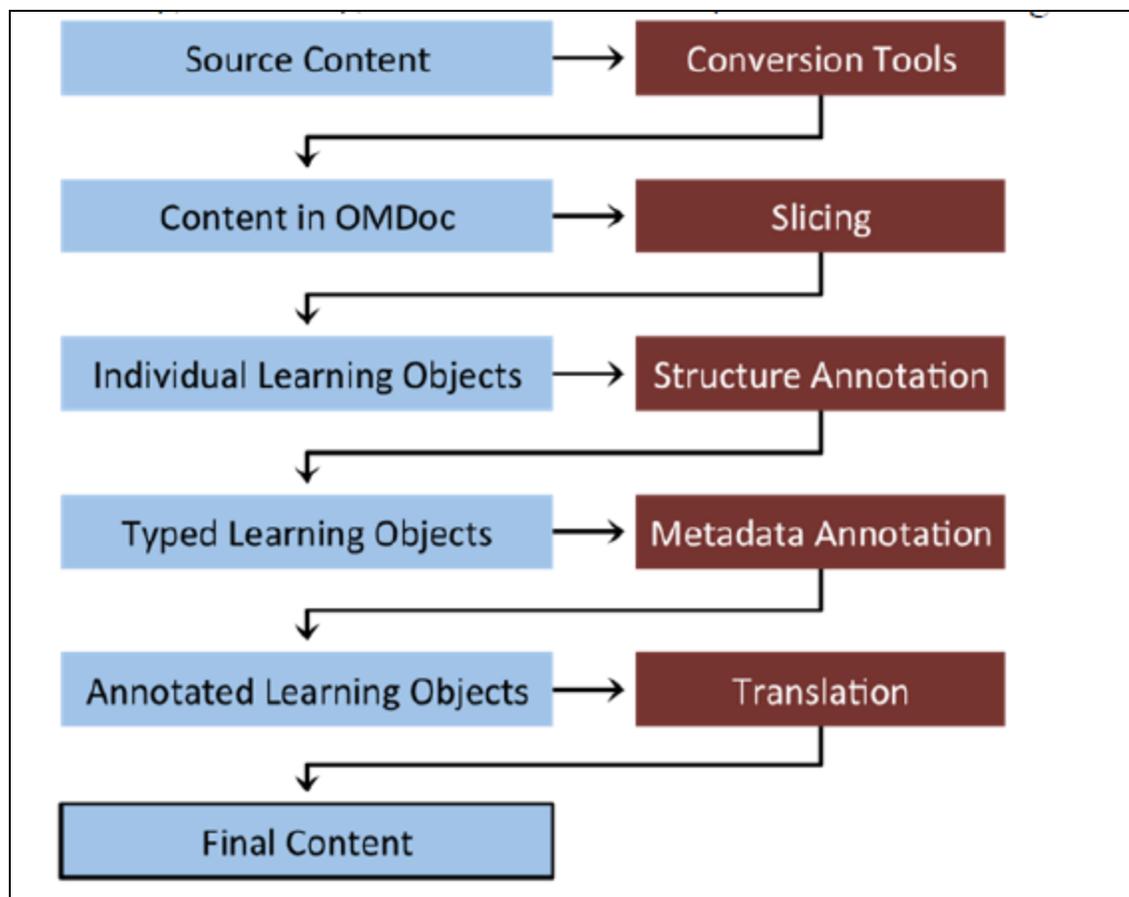


Figure 1.1: procedure of content transformation [6]

The total number of LOs in the Math-Bridge content base is around 11,000. Table 1.1 presents the details of this content base broken down by LO type.

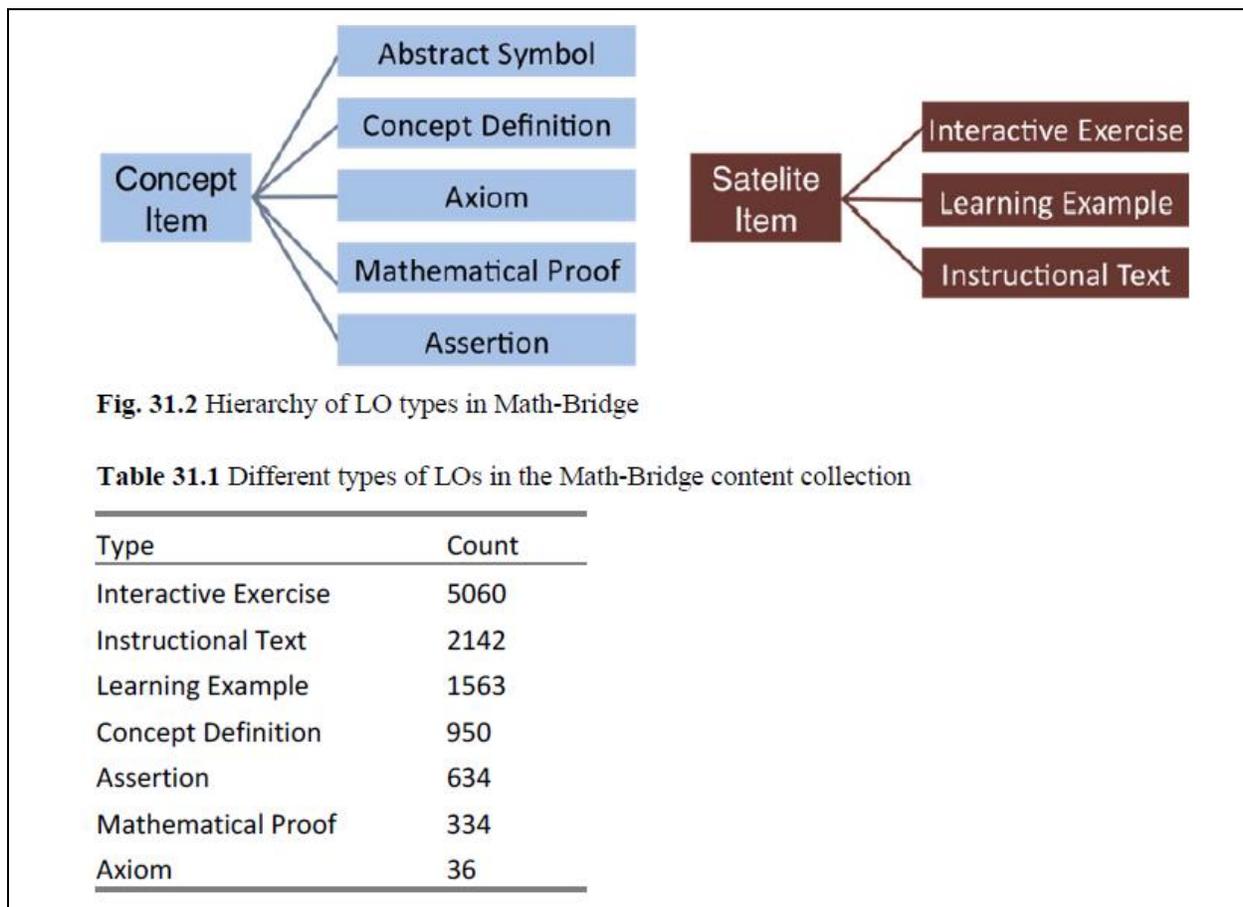


Table 1.1: LO Types in Math-Bridge

The content collections of Math-Bridge consist of multiple LOs, that link to each other by multiple relations. Both, the attributes describing LOs and the relations linking them together are specified via various metadata. This metadata elements can be:

- Descriptive metadata used for administrative, cataloguing and licensing purposes; represented mainly using the Dublin Core standard (DCMI 1999).
- Pedagogical metadata helping authors to specify multiple educational properties of LOs; adopted and extended from IEEE LOM and IMS LD standards (IEEE 2002; IMS 2003).

- Semantic metadata connecting different LOs to one another; partially relying on OMDoc and SKOS standards (Kohlhase 2006; Miles and Bechhofer 2009).

Now, speaking of Math-Bridge, the next important topic to know is Learning Analytics since Math-Bridge was implemented to help make Learning better. Learning analytics is the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs. Learning analytics (LA) is a multi-disciplinary field involving machine learning, artificial intelligence, information retrieval, statistics, and visualization. Learning analytics (LA) is a multi-disciplinary field involving machine learning, artificial intelligence, information retrieval, statistics, and visualization. LA is also a field in which several related areas of research in TEL converge. These include academic analytics, action research, educational data mining, recommender systems, and personalized adaptive learning.

The overall LA process is an iterative cycle and is generally carried out in three major steps: (1) data collection and pre-processing, (2) analytics and action, and (3) post-processing.

Data collection and pre-processing: Educational data is the foundation of the LA process. The first step in any LA effort is to collect data from various educational environments and systems. This step is critical to the successful discovery of useful patterns from the data.

Analytics and action: Based on the pre-processed data and the objective of the analytics exercise, different LA techniques can be applied to explore the data in order to discover hidden patterns that can help to provide a more effective learning experience. The analytics step does not only include the analysis and visualization of information, but also actions on that information.

Taking actions is the primary aim of the whole analytics process. These actions include monitoring, analysis, prediction, intervention, assessment, adaptation, personalization, recommendation, and reflection.

Post-processing: For the continuous improvement of the analytics exercise post-processing is fundamental. It can involve compiling new data from additional data sources, refining the data set, determining new attributes required for the new iteration, identifying new indicators/metrics, modifying the variables of analysis, or choosing a new analytics method. [13]

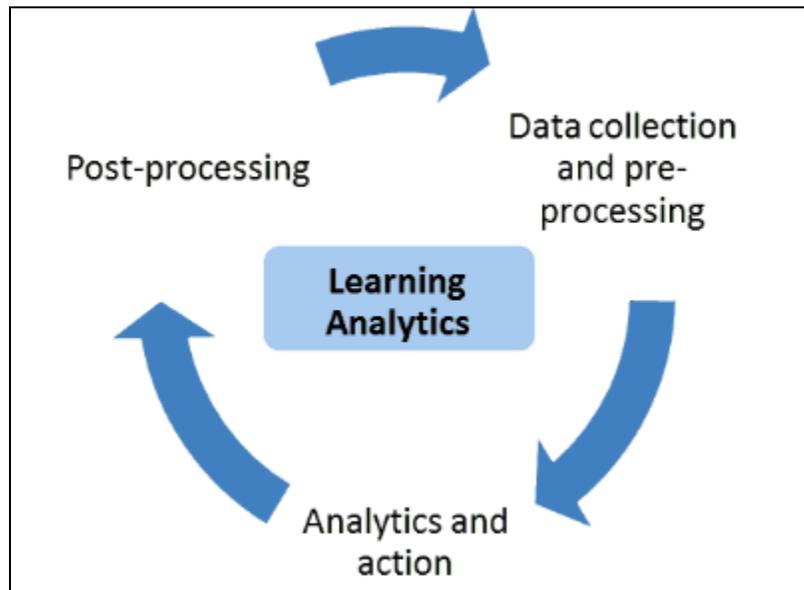


Figure 1.2: LA Iterative Process [13]

Learning Analytics development in Math-Bridge is difficult because of a legacy technology issue, which means that new developments becomes very difficult due to its old framework and the technologies it makes use of. Math-Bridge uses a time-honored database system called “Derby”. It is a relational database model built in Java and uses Embedded Framework. Math-Bridge and Derby work hand-in-hand as a single unit of operation and this thesis works towards separating Math-Bridge from Derby and give the system a makeover and have it use MySQL database to perform its operations.

This objective of this thesis came up due to these major issues with Derby database:

- a) It is very slow in nature.
- b) It does not perform as required when multiple users are accessing the system at the same time and hence inefficient.
- c) Since Derby is an ancient piece of work, there are not many resources available to treat it if any issue arises.

This is the basic motivation for my thesis topic.

1.2. Problem Description

As Explained above, there are big disadvantages if Learning Analytics Framework is to be developed upon Math-Bridge because of its legacy technologies in use. Math-Bridge uses Apache Derby for storing the system data, which uses a very old Framework and is not structured. Though easy to install and use, Apache Derby is not exactly the go-to database when it comes to handling huge data and multiple requests at the same time.

Maverick is another Java Framework that makes accessing of Math-Bridge possible by providing a platform to send input data and receive output data from and to the Math-Bridge system. It is through Maverick that the entire processing of information in Math-Bridge takes place and hence it is an integral part of the system. Maverick is essential for the Reporting tools and performing any UI related tasks on Math-Bridge.

On top of these two old technologies, there is also another legacy program called mybatis that is made use of in Math-Bridge. The usage of mybatis can be seen using an example as follows: 1. Math-Bridge user, a teacher would like to access the student's reports and see his activity, thereby clicks on the Reporting Tool. 2. The Reporting Tool invokes Maverick and sends a request asking for it to perform a specific set of instructions and get the necessary information about the student's activity. 3. Maverick then sends a command to Java Controller, which in turn calls mybatis program to get the needed data from Apache Derby using XML coding. 4. The data is then returned on a XML file from mybatis to Maverick, which is then sent back to the UI in Math-Bridge, the Reporting Tool visualizes and displays the data that is returned.

And that is where the main crunch of this Thesis comes into play.

1.3. Summary of the approach

The Thesis is carried out in this below approach:

Step 1- Data Migration from Apache Derby to MySQL: The data from Apache Derby is taken and the entire database structure is maintained and replicated into MySQL. The data is transferred into local host and then tested using a java application to check that MySQL data.

Step 2- Implementation of a Learning Analytics Framework: A new Learning Analytics Framework is built that frees Math-Bridge from unnecessary complications and calculations and XML conversions involved internally due to an additional 'mybatis' program. With this new Framework, Math-Bridge can directly contact MySQL through Java and get the necessary data for its reporting tool, thereby enhancing the capability of Math-Bridge and making it faster to produce results on the user's end.

Step 3- Piloting the Learning Analytics Framework: The Learning Analytics Framework is implemented for providing the user's activities in Math-Bridge and the exercise browsing. This LA is tested and evaluated by using both Apache Derby's and MySQL's data and that of the new version and old version of the user interface.

1.4. Structure of the thesis

Chapter 1 provides an introduction to Math-Bridge, what the problem currently in Math-Bridge is, how is the problem is approached to be solved, and the structure of the thesis. Chapter 2 introduces several research fields relevant to the topic of this thesis, including Intelligent Tutoring Systems, Learning Analytics and Educational Data Storage. Chapter 3 talks in detail about Math-Bridge, describing the existing systems and Math-Bridge architecture and its reporting Tool. Chapter 4 gives a detailed description of the approach, data migration from Derby to MySQL, the procedure and the problems and its evaluation. Chapter 5 talks in details about the Learning Analytics Framework, its architecture, development and implementation, and an Example Report built using this Framework. Chapter 6 is the conclusion chapter which details about the contribution of this thesis, limitation of the thesis and the future works that this thesis can lead to.

Chapter 2

Related technologies

2.1 Intelligent Tutoring Systems

ITSs are computer applications that support highly interactive, personalized, tutor-like instruction. The goal of ITSs is to simulate an individual tutor who closely follows students' progress, understands their current strengths and difficulties, and provides timely feedback in the form of hints and explanations.

ITSs do that by relying on different technologies from the fields of Artificial Intelligence and Cognitive Psychology. ITSs maintain several rigorous models that represent the knowledge (or expertise) necessary for performing meaningful tutoring:

- a domain model defines a set of elementary knowledge components (e.g. concepts) that a student needs to master;
- a learner model (usually a subset - or an overlay - of the domain model) helps an ITS to keep track of what the learner knows;
- a tutoring model formalizes the necessary pedagogical principles and strategies to make intelligent decisions on how to best maintain the tutoring process.
- an interface model controls the interaction of a learner with an ITS. [14]

Intelligent tutoring systems provide individual person-to-person tutoring services and help make learning better and easier. It is proven to be effective and efficient when compared to traditional methods of learning.

Also, with such systems comes the ease for having trained professionals to do the teaching tasks at hand; thereby increasing the productivity tremendously. The professionals are only required to keep track of the progress of the students and provide help when needed.

Previously with the traditional e-learning approaches, it was all directed in order to have the topics defined even before the student gets to interact with the system. This is however not the case with Intelligent Tutoring Systems and a lot of manpower is reduced.

That being said, there is also an advantage that the course can be designed pertaining to the student's knowledge level and what topics he or she needs improvements on. That is what is called as 'Student Modeling' in ITS (Intelligent Tutoring Systems). According to Woolf [1], these systems include many features or models such as, 'Domain Knowledge Modeling', 'Student Modeling', 'Pedagogical/Instruction Modeling' and 'Problem Solving/Task Environment'. This can be seen from the below diagram:

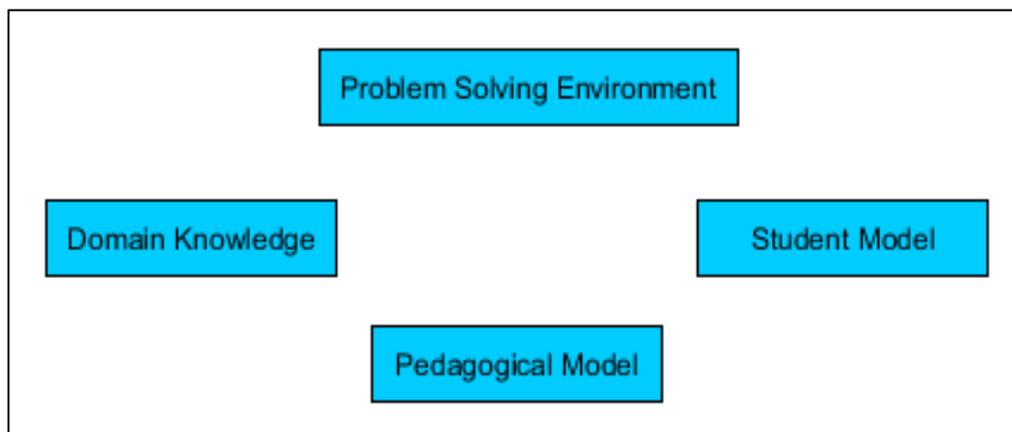


Figure 2.1: shows features of the ITS [14]

'Domain Knowledge': Since it is very well known that an Expert can be in only a particular field or a domain. This feature tries to understand and evaluate the performance Expertise in one particular Domain and models based on that expert skills and could be made use of in producing results of expert behavior.

‘Student Modeling’: It is the important type of modeling in ITS. As mentioned above, ITS revolves around creating a model suiting the needs and necessities of a student and thereby making the life of students easier and making their learning effective. This plays an important role in decision making in an ITS as well, because the tutor or a teacher is obliged to make suggestions or provide feedback based on a student model only. Self (1988)[] had stated the 6 different roles played by a student model: Corrective, Elaborative, Strategic, Diagnostic, Predictive, Evaluative. These functionalities come in handy when designing a tutor model in an ITS and in applying the student model as well.

‘Pedagogical or Instruction Modeling’: This feature is a derivative of student Modeling as it takes student model as input, wherein the Tutor Model / Teacher Model is developed based on the student model and decisions are made based on the results from the student model and confer to the student’s needs.

‘Task Environment’: This is the environment in which student users are allowed to take on varied problem solving activities. The results from this environment are then evaluated along with the domain knowledge component, which basically results in providing knowledge on expert level of problem solving. Then with this evaluation, a student model with each of the student’s knowledge is developed and kept track of. At the end, the pedagogical model provides results based on the student model evaluation and thereby the instructional actions can be taken.

However the ITS is built, based on the assumptions that Student Model is an extremely good approximate of what and how the student would interact with the system and the actions are tried to be reproduced by prediction mechanisms. The ITS, also tries to improve the student’s knowledge the same way a teach would and such an ITS can be made use of in any educational system to aid the students and also the tutors.

Adaptive Learning is the process of learning wherein the system adapts to the particular student’s choices and responses based on his/her needs. Very officially put, it is defined as follows:

“Digital learning systems are considered adaptive when they can dynamically change to better suit the learning in response to information collected during the course of learning rather than on the basis of preexisting information such as a learner’s gender, age, or achievement test score. Adaptive learning systems use information gained as the learner

works with them to vary such features as the way a concept is represented, its difficulty, the sequencing of problems or tasks, and the nature of hints and feedback provided.” [4]

An Adaptive Learning System in itself is aimed at providing better learning experience because, with the way it adapts to each of the student’s personal needs and continues to do the same and thereby making better assessments which can be useful for the tutors/teachers to understand the student’s needs and problem areas and in turn helping them achieve the intended goals.

The below diagram shows the different components of an Adaptive Learning System:

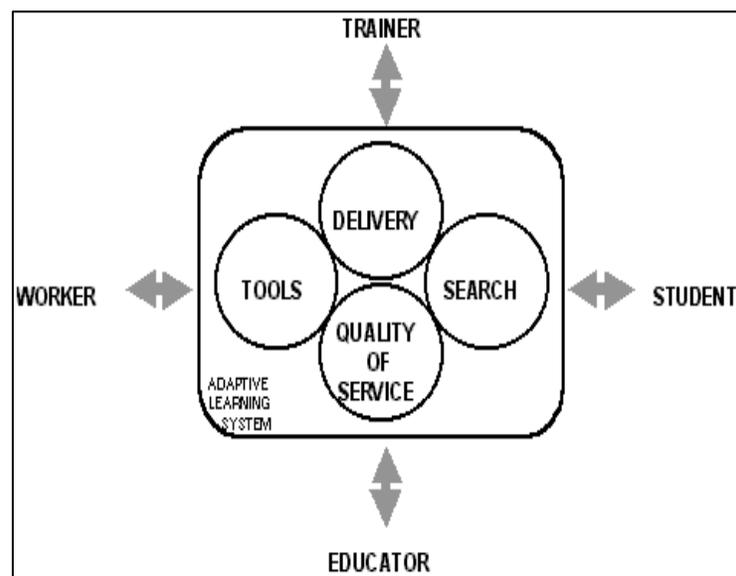


Figure 2.2 shows the Adaptive Learning Systems components, referred from [5]

The above Figure 2.2 shows the 4 components that affect the Adaptive Learning Systems according to this program are Content/Tools, Delivery, Search & Retrieval, and Quality of Service. There are numerous researches going on in these areas.

Content: talks about the varied components of the system, their interoperability and transformations.

Delivery: this concentrates of how the data is being delivered, how it’s managed or stored.

Search & Retrieval: tells how much the system is helpful to the teachers, being interactive with the students, provide intelligible outputs and assessments.

Quality of Service: tells how robust the system is, how it could be integrated or collaborated with other systems and provide a high performance. Referred from [5]

2.2 Learning Analytics

”Learning Analytics is about collecting traces that learners leave behind and using those traces to improve learning.” [2]

“the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs” [3]

Learning Analytics:

- It helps in improving the level of satisfaction a student can derive from an e-learning system and help them get prepared for future studies.
- It can help find creative ways in order to provide better learning platform for the students by forecasting the results of the students and helping in providing better opportunities for the students to overcome their learning issues.
- It provides better visualization criteria for the tutors to help with their student’s progress and help them decide on their courses and learning content, thereby enhancing the teaching level as well.

There are different ways in which learning analytics process can be visualized. Here is one such form.

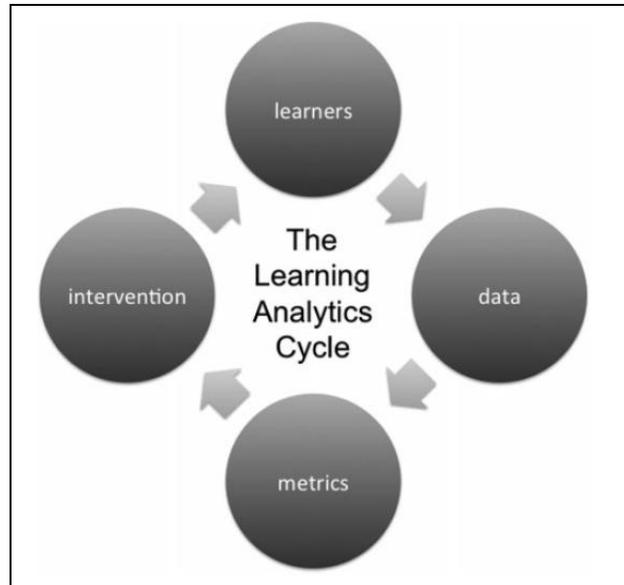


Figure 2.3: LA Process Cycle [16]

The LA cycle begins from the learners, who generate data, which is processed into metrics, which are used to inform interventions, which in turn affect learners. The learners may be students in a traditional higher education setting or in less formal contexts. The data can include demographic information, online activity, assessment data and final destination data. The metrics can be presented in a wide range of ways: from a simple indication of whether learning progress is being made to a comparison of outcomes to some benchmark, or a visual picture of activity in an online forum. The interventions again range widely, from students taking action in the light of metrics showing their activity compared to that of their peers, to teachers contacting students identified as requiring some particular extra help. [16]

The six dimensions of the proposed LA framework are (cf. Figure 2.4): stakeholders, objectives, data, instruments, external constraints, and internal limitations. [15]

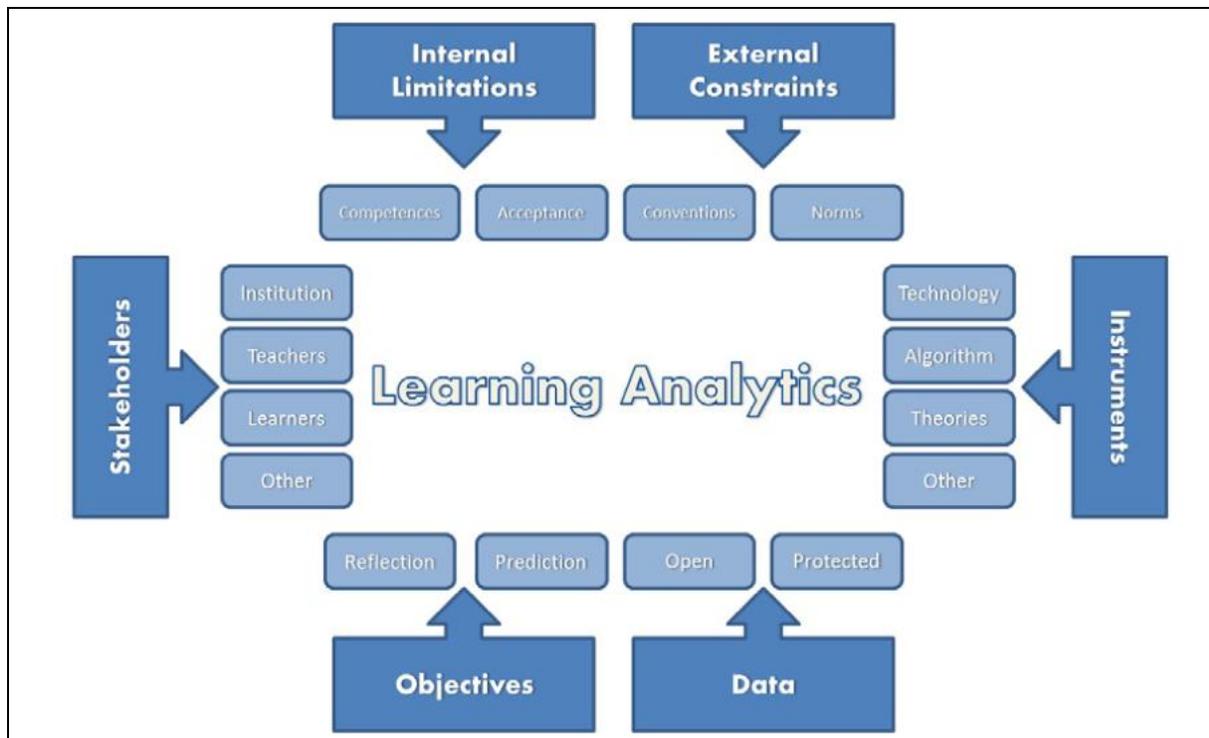


Figure 2.4: Dimensions of LA [15]

2.3 Educational data storage

The data that helps in aiding with the learning of a student, that provides information to the tutor/teacher to analyze and derive insight and help them provide better assessment and teaching to the students. This data is of different types: profile related information, predicted data based on student behavior modeling, individual results and course information of the students, his/her success/failure rate, attendance information, etc.

The above mentioned educational data is stored either locally or on an external system in order to help each student individually based on their respective data collected from multiple resources; thereby helping the students learn better. It also helps the tutors make appropriate assessments based on the respective data and hence enable them to provide effective and efficient teaching methods specific for each student.

Educational data mining (EDM) is the process in which all the data specified above and stored on a system are all collected together and are analyzed using data mining approaches and are used to help in the overall learning process of a student. EDM and LA go hand-in-hand when it comes to student data modeling. As seen previously, LA does nothing but analyzes the already collected data by applying certain methods and algorithms of data mining, certain machine learning logics and scientific measures and this is then used in predicting and aiding the student's learning experience.

EDM also is used to do the same, where the data is analyzed and some valuable insights are obtained by prediction mechanism. EDM uses multiple methods in order to perform the Prediction process like: Classification/Regression, Knowledge Tracing approaches, etc.

The difference between them can be seen by the way both EDM and LA are derived and modeled. EDM used automated and mathematical algorithms for prediction of the data whereas LA uses hypothetical and reasonable methods of prediction; thereby creating a complete understanding of the system. Thus, clearly LA leads to a better approach towards making tutor/teacher understand the student data better.

Now, speaking of Educational Data, DataShop¹ was developed by the Pittsburgh Science of Learning Center (PSLC), which is mainly used in storage of the educational data acting like a repository. This repository is used to store all the student data and the information and assessment details of all students. This data is also used for analysis purposes by means of reporting tools.

This is a free storage available for storing different types of data that are from varied sources like e-learning tools, visualization and other reporting tools and other ITS tools. DataShop can be accessed freely and by only creating a free account, one gets a chance to do data exploratory analysis among the data in the repository.

¹ <https://pslcdatashop.web.cmu.edu/>

This is what the PSLC claims that DataShop is capable of and what DataShop has been used for:

“ Toward collecting data in a uniform format, we have developed a standard XML logging format and two logging libraries (one in Flash ActionScript, the other in Java) to write this XML. Data can also be imported using a similar tab-delimited format. After importing data or logging the data to the DataShop database, the DataShop web application can help you start exploratory data analysis with tools for common learning science analyses.

You can also export data for further manipulation and analysis in other tools. Researchers have utilized DataShop to explore learning issues in a variety of educational domains. These include, but are not limited to, collaborative problem solving in Algebra (Rummel, Spada, Diziol, 2007), self-explanation in Physics (Hausmann & VanLehn , 2007), the effectiveness of worked examples and polite language in a Stoichiometry tutor (McLaren, Lim, Yaron, & Koedinger, 2007) and the optimization of knowledge component learning in Chinese (Pavlik, Presson, & Koedinger , 2007). “[7]

What makes DataShop special for Educational Data Storage- these 3Attributes says why. In DataShop, the data is 1) fine-grained: contains student data in the tiniest detail possible including all the interactions between the system, 2) longitudinal: wherein all the data from the start to the end in the entire span of the course is available, 3) extensive: which by the word itself means it consists of a huge number of data useful for the educational data mining and analysis.

Using DataShop tools, one can perform exploratory data analysis as mentioned above in the below ways:

- Datasets: DataShop provides certain datasets to work with like certain measurements, information on student performance and student numbers which can be modeled into various ways to make intelligible analysis out of it.
- Reports: DataShop also has tools that give many reports useful for analysis purposes. It is possible to get reports on anything the student does with the system like exercise answered/failed/passed, his/her responses for the questions, etc.
- Profiler for Performance analysis: as the name suggests, DataShop provides with a Profiler tool to have an eye on the student’s performance at different depths possible. Here is an example of how the Profiler looks from the system. Figure 2.5 shows a chart

of the average Error Rate with incorrect entries, hints, correct entries by a student; thereby producing the Error Rate that is valuable for assessing his/her Performance.

- Learning Curve: this does the prediction aspect in DataShop, where there are multiple visuals provided in order to track how the student learns in the system, how his/her performance varies with time, error rates as described before too. This example below shows us an example of one such a visualization provided by DataShop. Figure 2.6 depicts the learning curve of a student with the error rate with some values that were predicted. Above data were inspired and referred from [8].

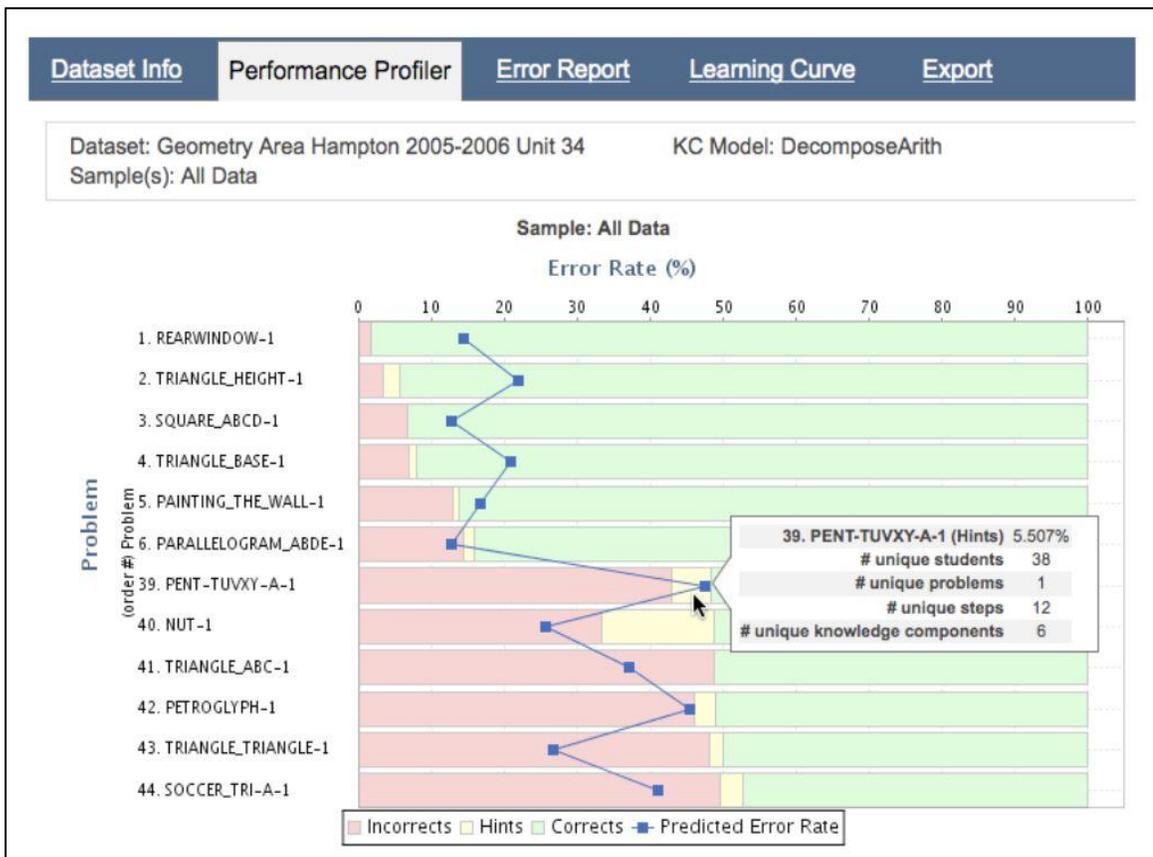


Figure 2.5 shows a chart of the average Error Rate, referred from [8]

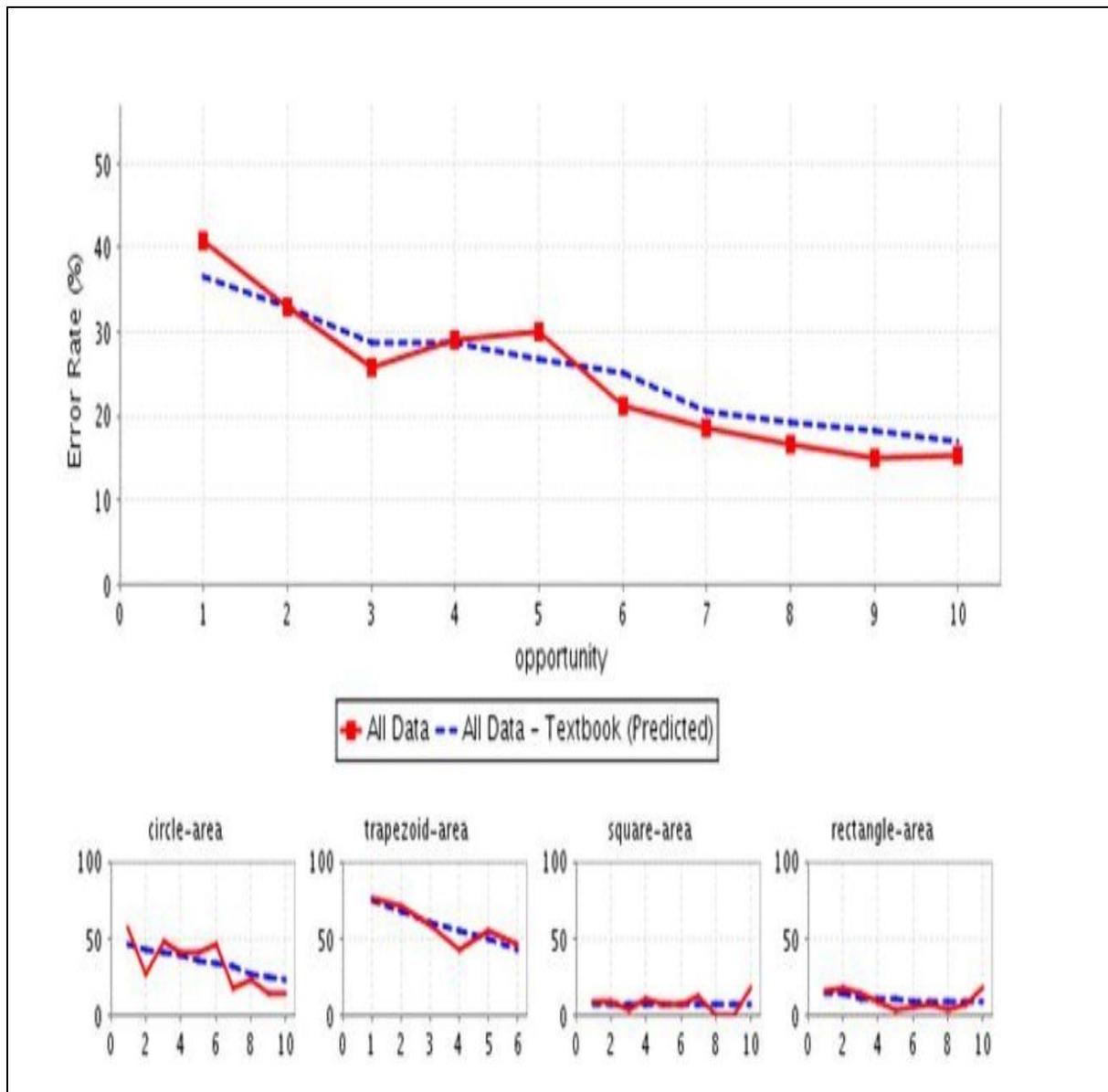


Figure 2.6 depicts the learning curve of a student with the error rate with some values that were predicted. The solid curve represents the actual values; each point is an average across all students and knowledge components for the given opportunity. The dashed curve represents the predicted curve values, based on the Learning Factor Analysis (LFA) model [9], a variant of Item Response Theory. Referred from [8].

Chapter 3

Math-Bridge

3.1 Short description

“Math-Bridge is an e-Learning platform for online courses in mathematics” [6]

According to a research it was found out that the number of students that drop out of universities was getting higher in Europe and hence Math-Bridge was developed to help student fill their gaps in studies Europe-wide, using personalized online courses and are available in multiple languages.

3.1.1 Existing systems Vs Math-Bridge

- The existing systems are not personalized and hence cannot cater to the individual student needs and area of study.
- They are not all multi-lingual and are difficult to predict new information to provide a better learning experience.
- They are easily accessible and understandable to all students and thus cannot specifically help in their improvements; thereby also making it difficult for tutors to provide a better teaching experience since they are less interactive.

3.1.2 Math-Bridge Content

All the content for the Math-Bridge tool comes from varied sources. These are as follows:

- Tampere University of Technology
- Universität Wien, mathe online: www.mathe-online.at
- Universität Kassel, VEMA: www.mathematik.uni-kassel.de/~vorkurs/
- Open Universiteit Nederland: www.voorwis.ou.nl:1025/nkbw_data/nkbw/
- DFKI: Bruchrechnung: demo.activemath.org
- Universität des Saarlandes: www.leactivemath.org

An example of the Math-Bridge content page looks like this.

The screenshot displays the Math-Bridge website interface. At the top left is the Math-Bridge logo. The main navigation bar includes a 'Table of contents' sidebar on the left, a central header for 'Differentiation according to the chain rule - Introduction to Calculus', and a 'More' section on the right with a search bar. The 'Table of contents' sidebar lists various topics, with '4. Differential Calculus' selected. The main content area is titled 'Application of the chain rule' and asks the user to compute the derivative of $f(x) = \cos(7 \cdot x^{10} - 10 \cdot x^7)$. It shows three attempts at the derivative:

- Attempt 1: $f'(x) = \cos(7 \cdot x^{10} - 10 \cdot x^7) \cdot (70 \cdot x^9 - 70 \cdot x^6)$. Feedback: 'No. Have another look at the chain rule. The first factor is $x'(f(x))$, not simply $f(x)$. So your factor $\cos(7 \cdot x^{10} - 10 \cdot x^7)$ is wrong.'
- Attempt 2: $f'(x) = \sin(7 \cdot x^{10} - 10 \cdot x^7) \cdot (70 \cdot x^9 - 70 \cdot x^6)$. Feedback: 'Not quite. Please check the signs. Also remember $\cos(x) = -\sin x$.'
- Attempt 3: $f'(x) = -\sin(7 \cdot x^{10} - 10 \cdot x^7) \cdot (70 \cdot x^9 - 70 \cdot x^6)$. Feedback: 'Yes, that's it. You'll find a similar exercise with other values here.'

 A message at the bottom states 'Your exercise is over, please close the tab.' The right sidebar contains a 'Social' section with a feedback form for the current exercise, showing 1 'Helpful' and 0 'Not helpful' votes, and options to add comments or notes. At the bottom of the sidebar is a 'Report problems' section with a link to post a problem to forums. The footer includes 'Legal Notes' and '© 2012 Math-Bridge'.

Figure 3.1 shows the Math-Bridge content [6]

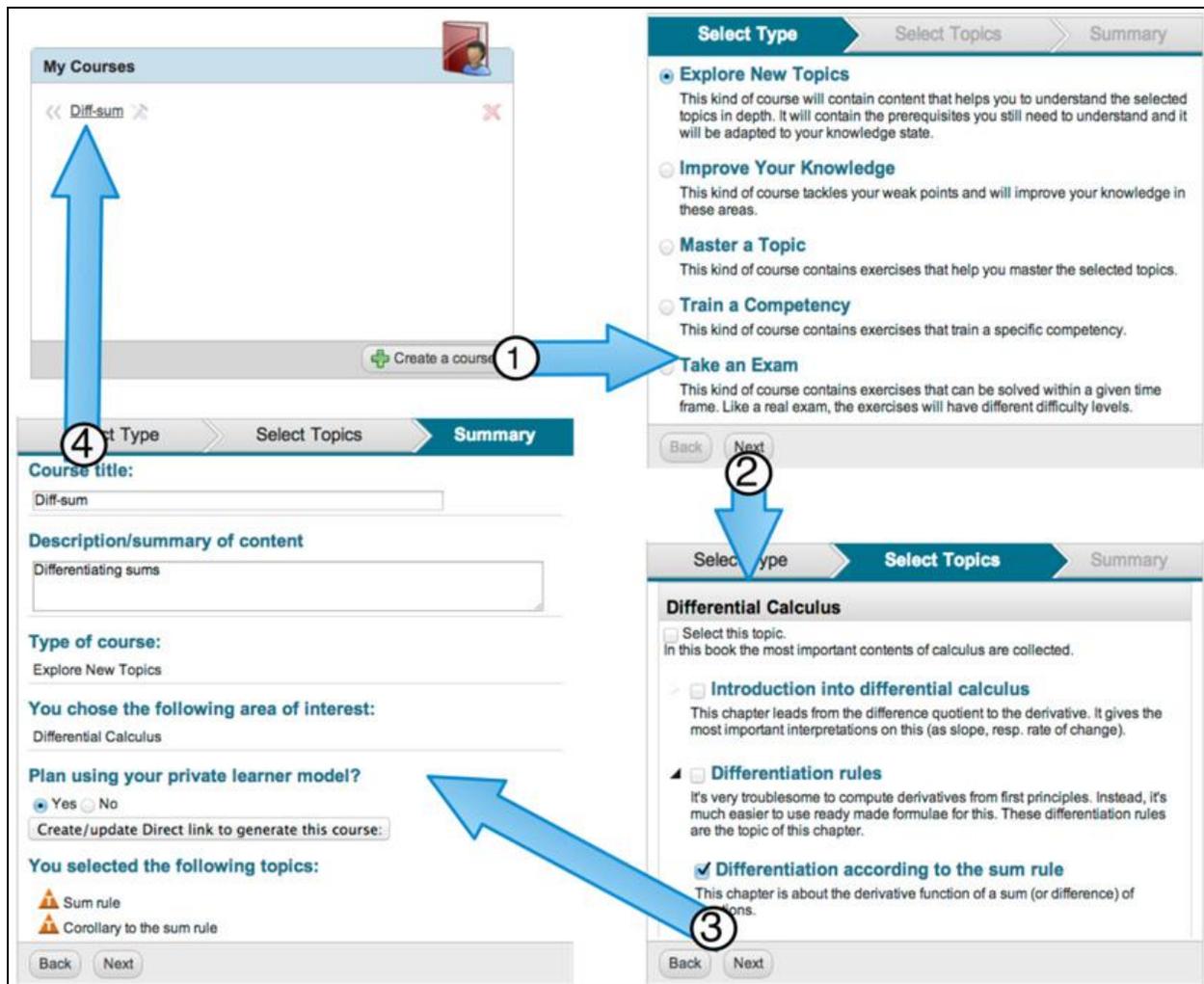


Figure 3.2: Course generation in Math-Bridge: 1) Click the button “Create a Course” in the “My Courses” widget of the Dashboard; 2) Choose one of the six course generation scenarios; 3) Select one or more target topics; 4) Name the course. [6]

3.2 Architecture

Math-Bridge is delivered as a Web Container using Apache Tomcat. Tomcat manages all the GET/POST requests from users of Math-Bridge and manages the entire load in its server.

Math-Bridge uses a platform called, ‘Maverick’ which is used for web development and uses a Model-View-Controller (MVC) type of implementation. This makes it easier for sending requests from the client to database and it returns intelligible data view to the user.

The below diagram, Figure 3.3 of the Math-Bridge Architecture shows how all the interactions are performed in the system. This was developed by the MetaMath, Mathgear group by Michael Dietrich (from DFKI).

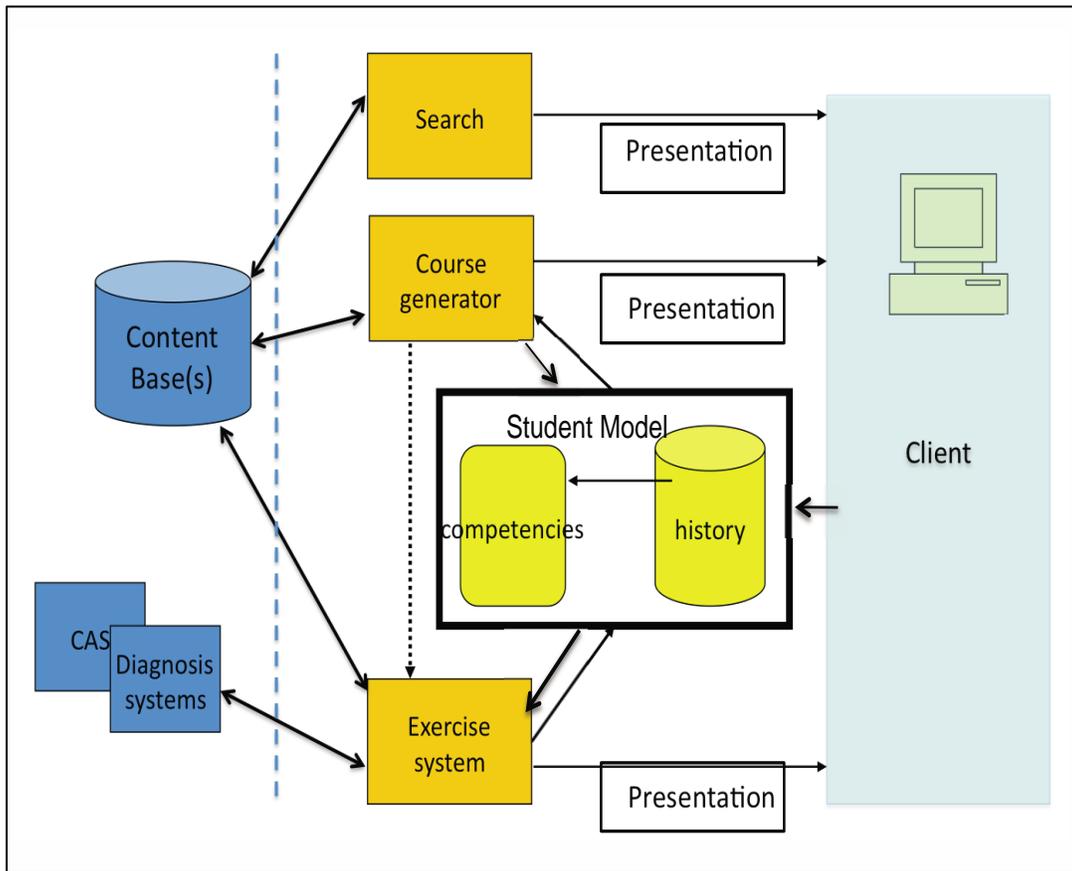


Figure 3.3: Math-Bridge Architecture Flow, Referred from [10]

In this model, the Client (a student) enters into the system his/her credentials and is able to access Math-Bridge. Once the student’s contents are verified, he/she is able to either create a new course or do an exercise. Based on this choice, either a course is generated from the Math-Bridge content-base system or an exercise is created with both content and diagnosis systems databases and is presented to the student. The student’s history is stored in history database.

The next diagram, Figure 3.4 shows the layered architecture of Math-Bridge. This is again developed by MetaMath, Mathgear group by Michael Dietrich (from DFKI). It consists of varied components as follows:

- **Web Layer:** The top-most layer of the architecture, which is certainly implements and handles the presentation elements for the user and it, is how the user interacts with Math-Bridge. It uses Apache Velocity², HTML 5/ JavaScript for user interface presentation, HTML/MathML for formula presentation.
- **Core:** handles all the necessary functions in Math-Bridge like System startup, loading of configuration, loading of contents, Building search-index, Initializing of all components. This component uses Apache Tomcat (Web Container), Java (Programming language), and Databases (Apache Derby), Core/Learner History/Notes as main technologies to perform its tasks.
- **Events:** This component's main usage lies in providing Asynchronous communication between components, Subscribe or Un-Subscribe or Publish tasks from/to Math-Bridge. It basically runs of Java and XMLRPC.
- **Content DB:** It is mainly used for Storage and deliverance of Mathematical content and for Building Search Indices in Math-Bridge and runs primarily on Java, Apache Lucerne³ to perform search operations and OMDoc⁴ sometimes for delivering Mathematical content and representing Knowledge databases.

²<http://velocity.apache.org/>

³<https://lucene.apache.org/>

⁴<http://www.omdoc.org/>

- Content: All the Java based Content are handled here by using certain operations like Load, Delete, Copy, Save. It makes use of JDOM⁵ for these operations.
- Presentation: This component uses XSL and Saxon XSLT and Xquery Processor⁶ to perform User Interface related tasks that outputs varied content in HTML from the OMDoc data available in Content DB.
- User: As the name suggests, all the user roles and data are modeled here using Java. This user includes all who enters the system like a student, tutor, teacher, course admin.
- Learner Model: This Model component uses Java to store all the user history information including the performance tracking, exercises completed or not completed as well as the learner knowledge and error rate.
- Exercises: This component does nothing but Exercise Related stuff like- completed or not completed, attempted exercises, student's responses to exercises, task accomplishments, etc. using Java.
- Tutorial Component: This particular component comes in handy for the Tutors and Teachers in the system since it has the entire information specific to a student and thereby is capable of creating personalized courses.

⁵<http://www.jdom.org>

⁶<http://saxon.sourceforge.net/>

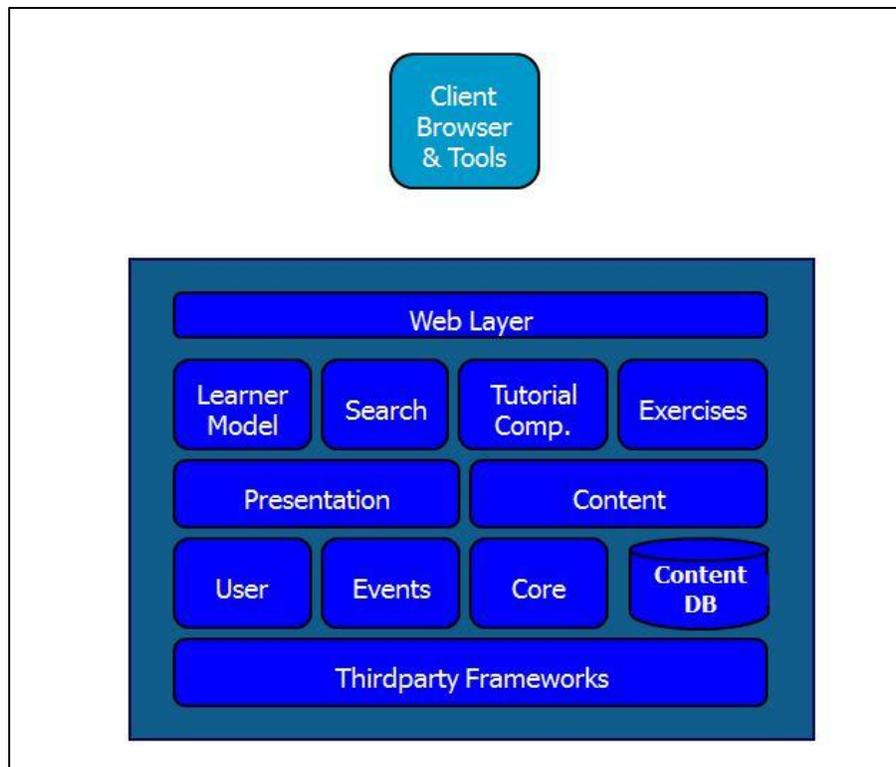


Figure 3.4: Layered Architecture of Math-Bridge, referred from [10]

3.3 Reporting Tool

The reporting tool in Math-Bridge is used specifically by the tutors/Teachers to analyze each student's activities and performance after interacting with the system. These reports are used to keep track of the student's performance and help them by providing valuable information for the student's betterment. This tool can also be used to monitor how the whole class is performing or get a specific group of student's performances. Math-Bridge system allows the teachers to work with at the least 10 different reports and analyze their students' works.



Figure 3.5: Math-Bridge teacher interface: reporting tool

The below figures shows the screenshots of how Math-Bridge can be used in multiple languages.

Figure 3.6, 3.7 gives an example of the Start page of Math-Bridge in 2 different languages

MATH-BRIDGE
EDUCATION SOLUTION

Panel de control

-navbar.authoring- Gestionar Mi Perfil Ayuda Salir

Sugerencias << Panel de control

Hola admin.

Introducción a Math-Bridge

Math-Bridge ...

Unirse a la comunidad Math-Bridge:
Canal en Youtube
Foros

Cursos registrados

[Indice completo de LeAM_calculus](#)

[Short Introduction to Calculus](#)

▷ **Cursos en otros idiomas:**

Cursos personales

No hay cursos disponibles.

Crear un curso

Exámenes

`$exam.getTitle($lang)`
`-exams.dashboard.duration-mins.not-set-`

Marcadores

Legal
© 2014 Math-Bridge

Figure 3.6 Math-Bridge Start Page in Spanish

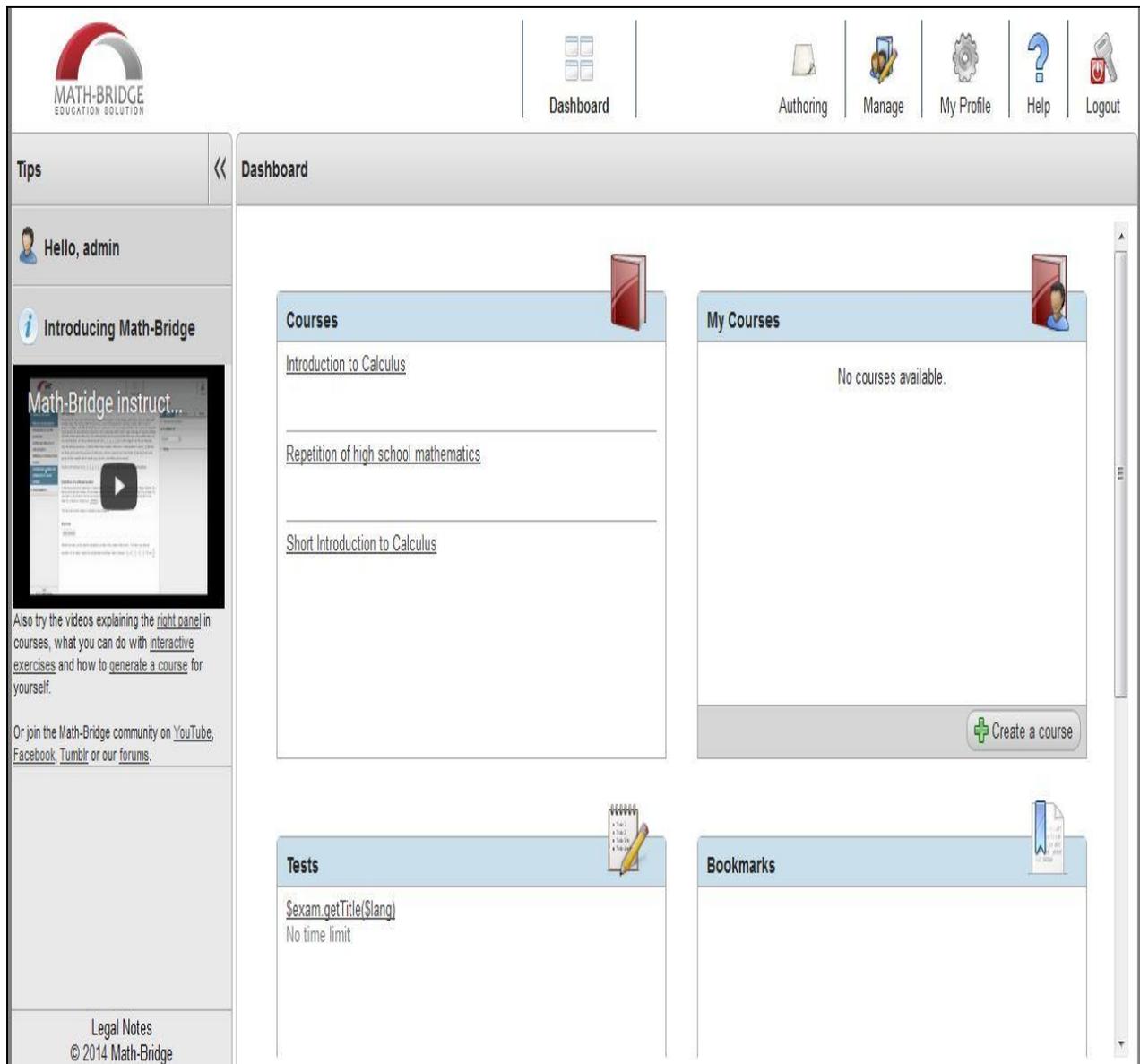


Figure 3.7 Math-Bridge Start Page in English

Figure 3.8, 3.9 gives us an example of how not just the Start Page, but also the courses can be done in multiple languages.



MATH-BRIDGE
EDUCATION SOLUTION



Hauptseite



Autoren-Tool



Verwaltung



Mein Profil



Hilfe



Abmelden

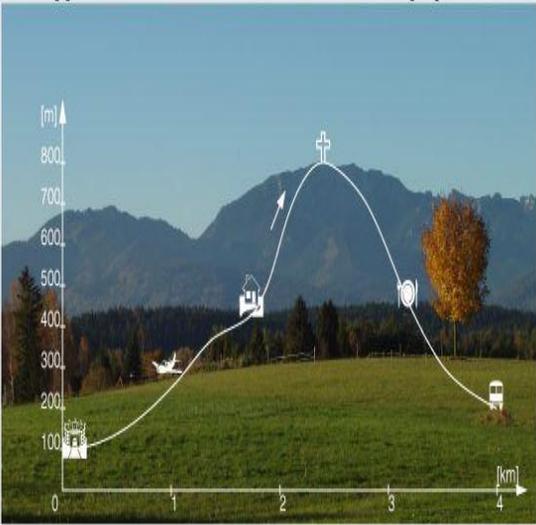
Inhaltsverzeichnis <<

- 1. Einleitung
- Eine Bergwanderung
- Bergwandern mathematisch gesehen
- 2. Im Durchschnitt?
- 3. Differenzenquotienten
- 4. Momentane Steigung
- 5. Ableitungsfunktionen
- 6. Nachtrag: Änderungen und Ä...
- 7. Ableitungen einfacher Funkti...
- 8. Ableitungsregeln

Eine Bergwanderung - Analysis Kurzeinführung <<

▲ Eine Bergwanderung

Marie und Michael haben eine Bergwanderung gemacht. In ihrem Wanderführer ist ein Profil ihrer Wanderung abgebildet.



Von ihrem Startpunkt aus (der Arena) sind sie zunächst zu einer Hütte gelaufen, dann über einen steilen Klettersteig zum Berggipfel. Von dort sind sie zuerst zu einem Bergrestaurant und dann zur Bushaltestelle im nächsten Ort, ihrem Zielpunkt, abgestiegen. Nun interessieren sie sich für die Steigung, die sie überwunden haben.

Impressum
© 2014 Math-Bridge

Figure 3.8 Math-Bridge Course in German



MATH-BRIDGE
EDUCATION SOLUTION



Dashboard



Authoring



Manage



My Profile



Help



Logout

Table of contents <<

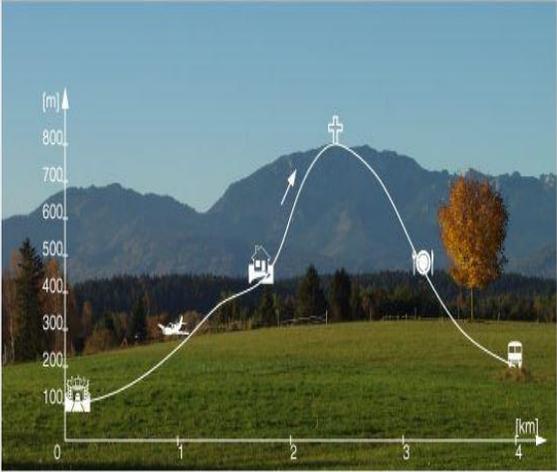
- 1. Introduction
- ▶ A hiking tour
- ↳ Let's talk math about hiking
- 2. What about the average?
- 3. Difference quotients
- 4. Actual slope
- 5. Derivative functions
- 6. Supplement: Changes and rat...
- 7. Computing the derivatives of ...
- 8. Differentiation rules

A hiking tour - Short Introduction to Calculus <<



A hiking tour

Mary and Michael were on a hiking tour. Their hiking booklet contains a profile of their tour.



From their starting point (the famous arena) they have first reached some mountain hut, then walked a steep fixed rope route to reach the mountain top. From there they descended to a restaurant and then to their final destination, the bus stop in the next village. Now they care for the slope they have overcome.

Legal Notes
© 2014 Math-Bridge

Figure 3.9 Math-Bridge Course in English

Chapter 4

Migrating Math-Bridge

4.1 Storing educational data with Derby

As seen above from Chapter 2, educational data is a necessity when it comes to an ITS like Math-Bridge. In this section is discussed how and what data is stored Derby.

4.1.1 Derby-an Insight

Apache Derby is a relational database that provides a client/server model and is completely built in Java. It can be very easily included in a Java solution using the provided embedded drivers because when it is written at one place, it can be run at any place making it very robust as it is platform-independent.

Apache Derby first was born with the name of Cloudscape by Cloudscape Inc. in 1996. It was later acquired by IBM and formed as IBM Cloudscape 5.0. Then by 2004, it became Open Source at Apache.

Derby is provided and accessible from a jar file and all the processes take place within the executable itself in the Java Virtual Machine.

The below Figure 4.1 shows how Derby accesses database with just a single JVM.

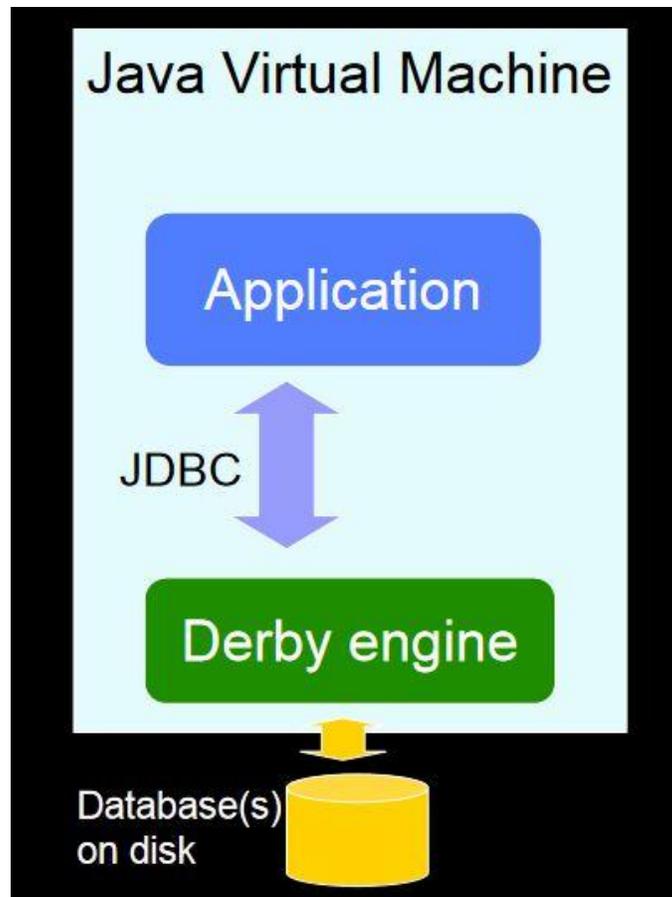


Figure 4.1: Derby access through JVM, referred from [11]

Derby is like any other databases which has the standard SQL features like rows, columns, tables, views, index, all constraints like primary-key, foreign-key and all joins, procedures, triggers, functions, etc. All the relational database features and operations are also performed and satisfied by Derby as well.

The API of Derby uses JDBC and SQL technologies and the famous developer tool used is ij, which is a scripting tool for performing SQL commands in derby using Command line.

Here is an example, Figure 4.2 of how to connect to a Derby Database from an IDE, which follows the below set of steps to establish a connection with Derby.

```
// Register the JDBC driver class
String driver="org.apache.derby.jdbc.EmbeddedDriver";
Class.forName(driver);
// set up the database name
String dbName="jdbcDemoDB";
// define the Derby connection URL to use
String connectionURL = "jdbc:derby:" + dbName +
";create=true";
// establish the connection to the database.
conn = DriverManager.getConnection(connectionURL);
```

Figure 4.2: Connect to Derby from an IDE

4.1.2 Math-Bridge Data organization in Derby

In Derby, the educational data is stored into multiple databases varied across different tables, spread between numerous rows and columns. The types of database schemas include:

derby_am_core_db, derby_learner_history_db, derby_gapstorage_db, derby_notes_db.

derby_am_core_db: contains the core information, user profile and account data and usergroups data, book data, exam data, exercise and test data.

derby_learner_history_db: contains assessment data, belief content, course data, events performed by users including when exercise was started/finished, login data, assessment and goals.

derby_gapstorage_db: contains learning gap data and tutor information.

derby_notes_db: contains notes in the system.

4.2 Problems

The main issue with all this data in Derby was that it was taking a whole lot of time for maverick to process all that information and it was less efficient when multiple users logged into the system at the same time.

The issue occurs because Maverick internally tries to connect through xml with derby and gets the raw data back and tries to fill in the gaps in the xml file and the processing is all done in the server level, thereby making it time-consuming process.

4.3 Exported MySQL data base

Educational data in this section is discussed how and what data is migrated into MySQL.

4.3.1 MySQL-an Insight

MySQL is popularly known Relational Database which is available for free. It is popular because of its capability to manage huge databases and because of how fast it can perform even with such huge amounts of Data. Also, the fact that it can be created and maintained locally makes it very interesting for use.

Like any other Relational Databases, MySQL also follows the same principles and it can be represented as below Figure 4.3 to show how the data is actually stored.

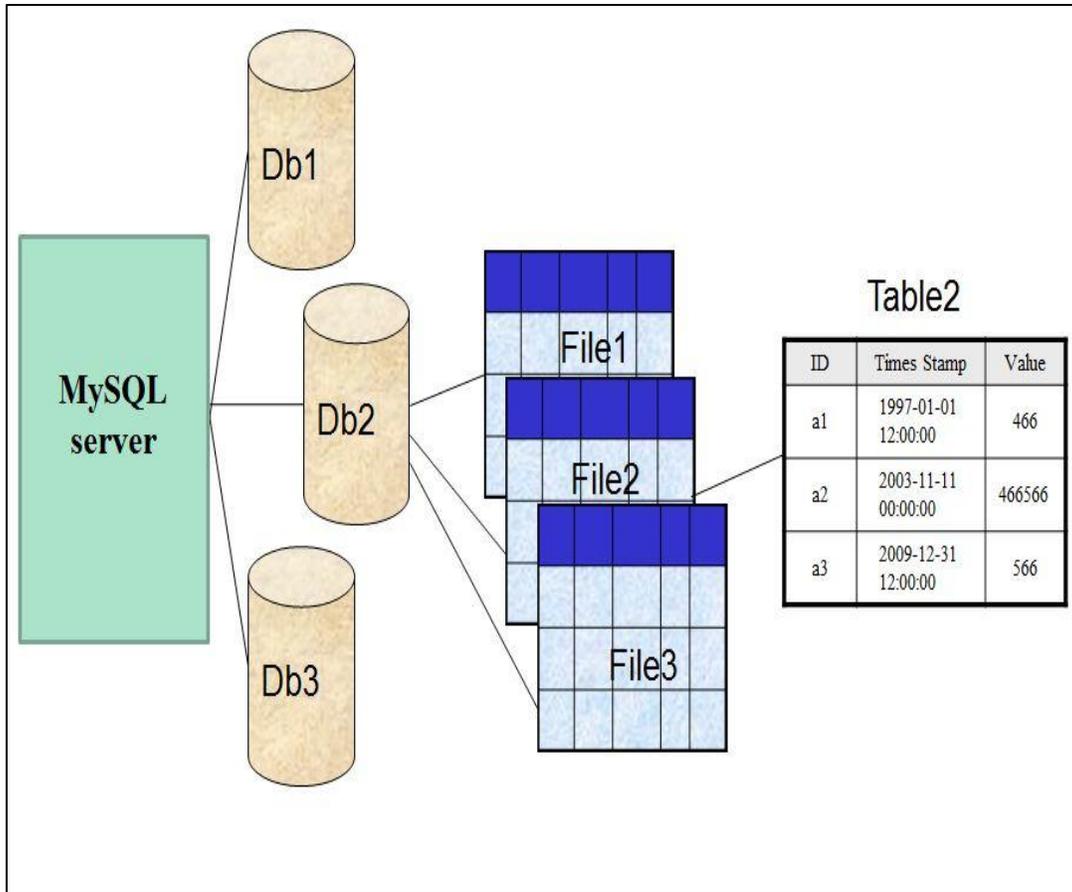


Figure 4.3: MySQL storage of Data, adapted from [12]

Just like Derby, JDBC connection to MySQL can be done as follows:

```

conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/myDB?"
+"user=root&password=password"); |

```

Database name

Server and port number

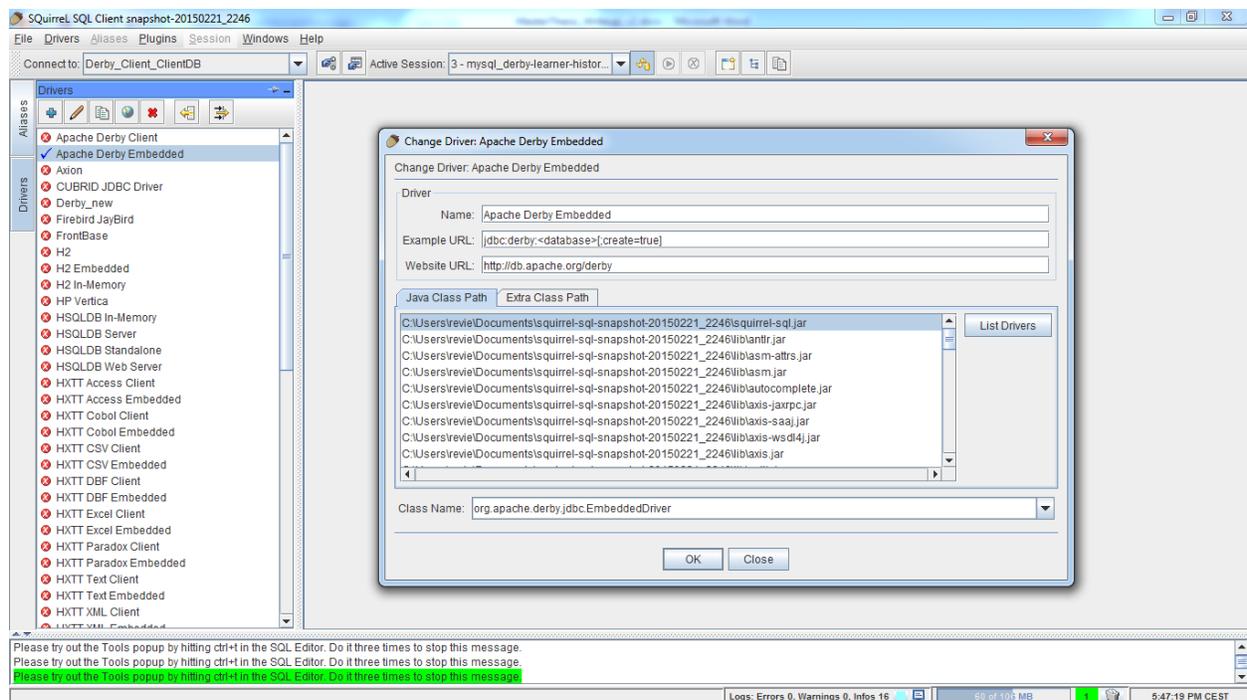
Figure 4.4: JDBC connection to MySQL

4.3.2 Exported MySQL data

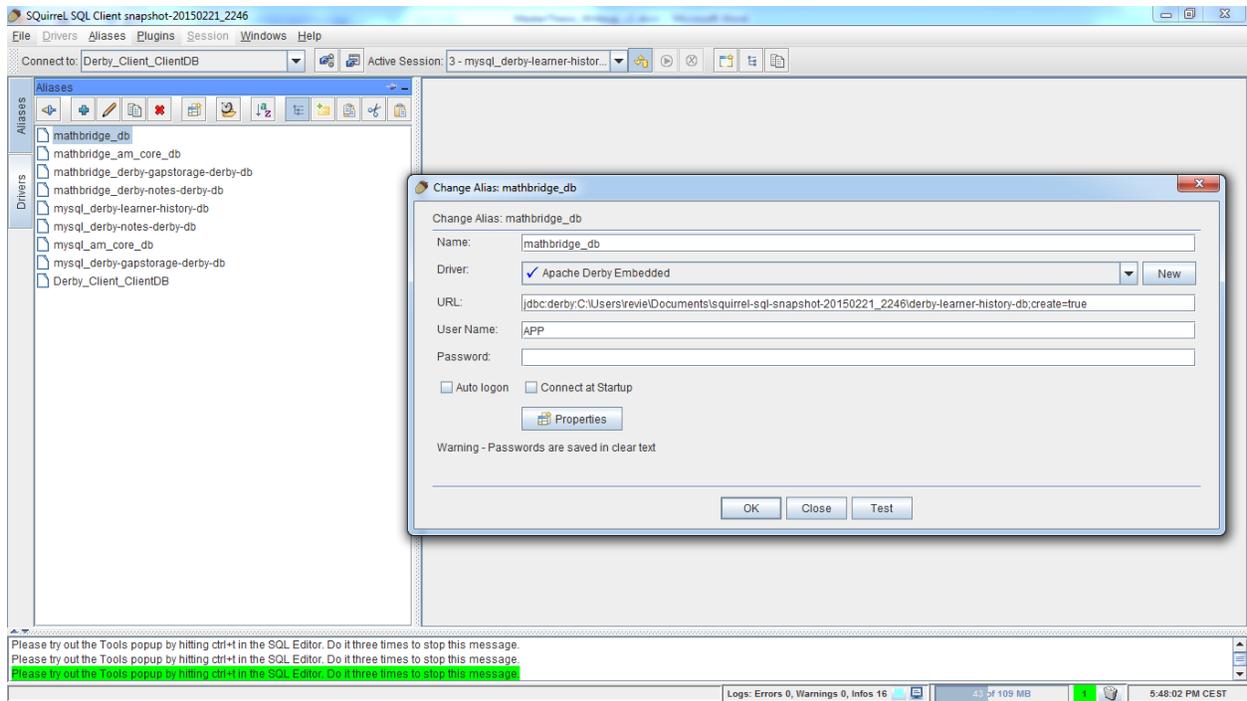
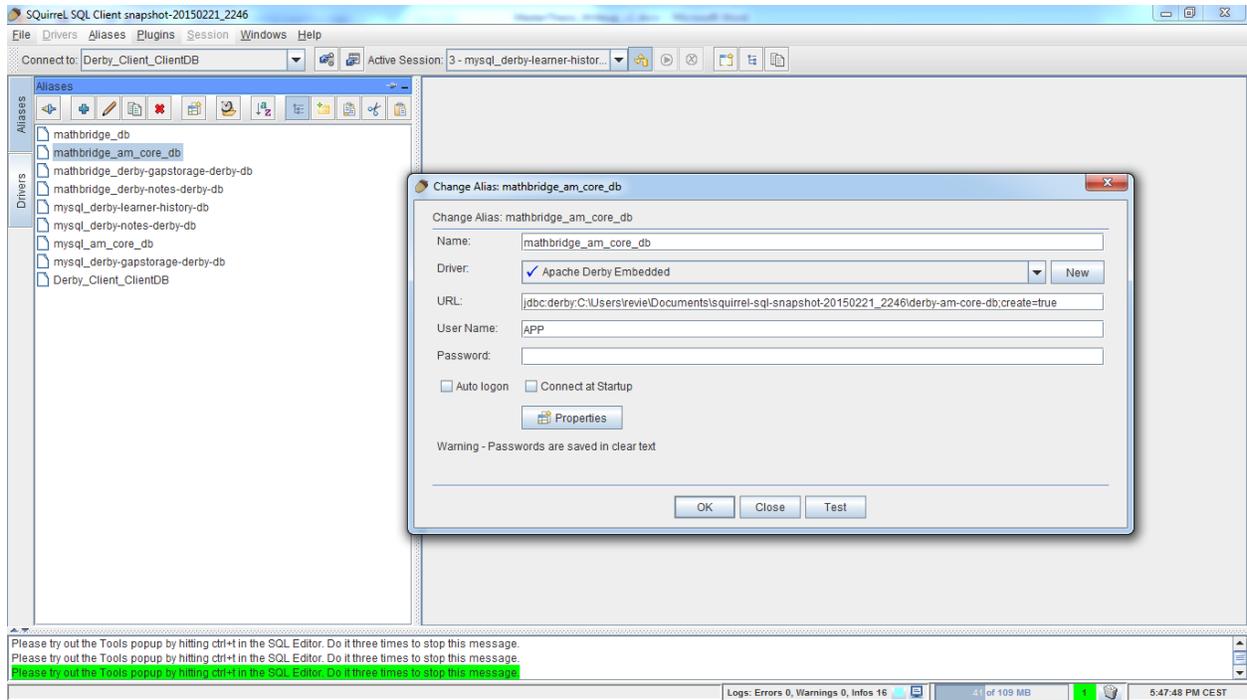
The MySQL database is created by exporting all the sample derby data manually into the localhost of the machine using a tool called, SquirrelDb. This tool enables to connect to both derby and MySQL databases at the same time and copy the databases by going over each and every table in derby. Each table from all the 4 above mentioned database schemas in derby is manually copied into MySQL with the same names and structures.

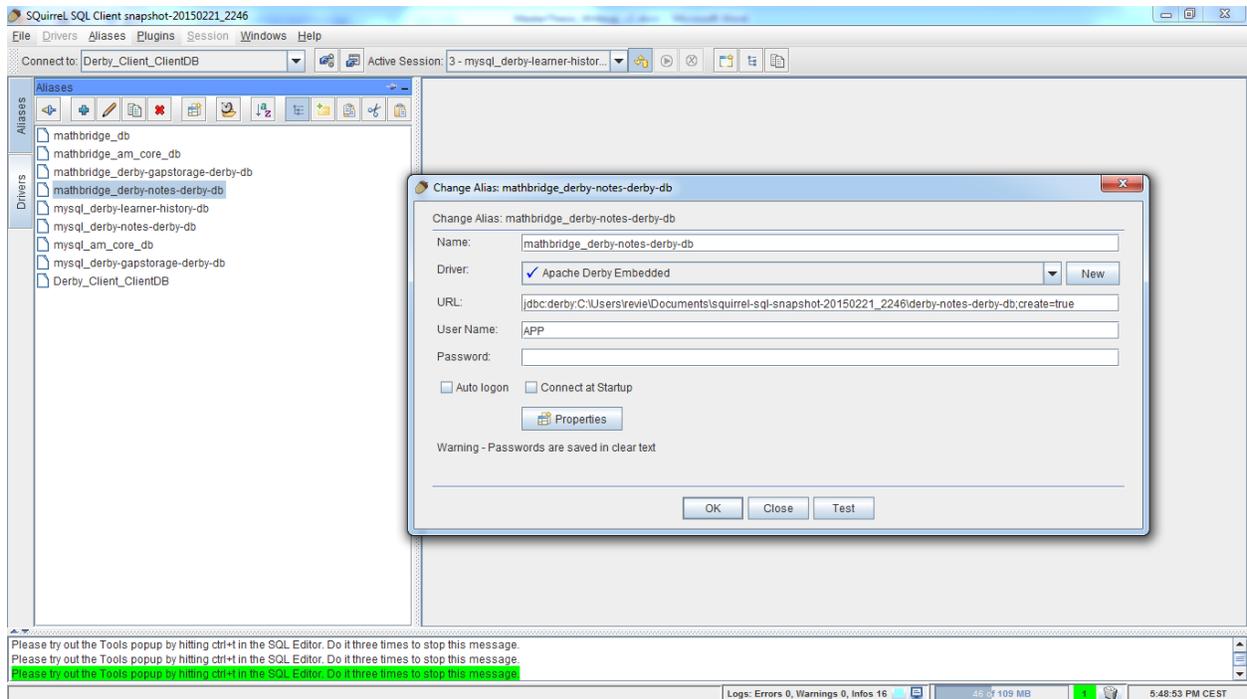
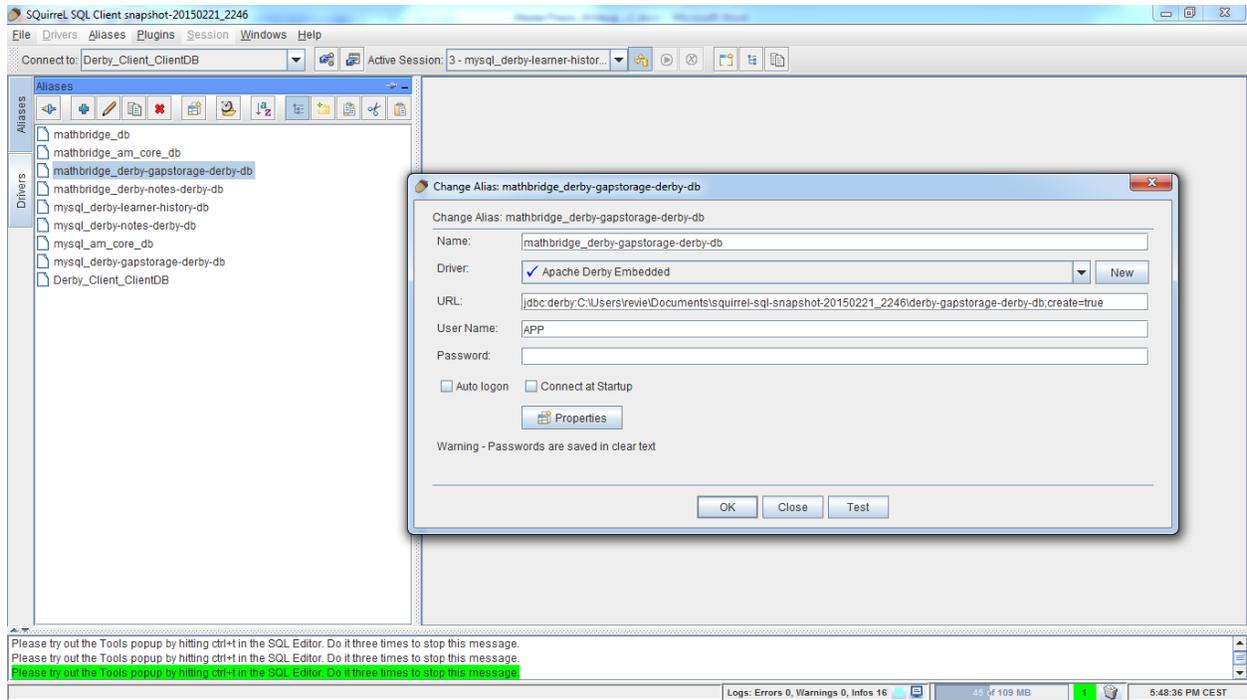
4.4 Migration procedure

The migration of data from Derby to MySQL is done using a client tool called Squirrel SQL. This tool is first opened and the required Apache Derby Driver is added as follows.



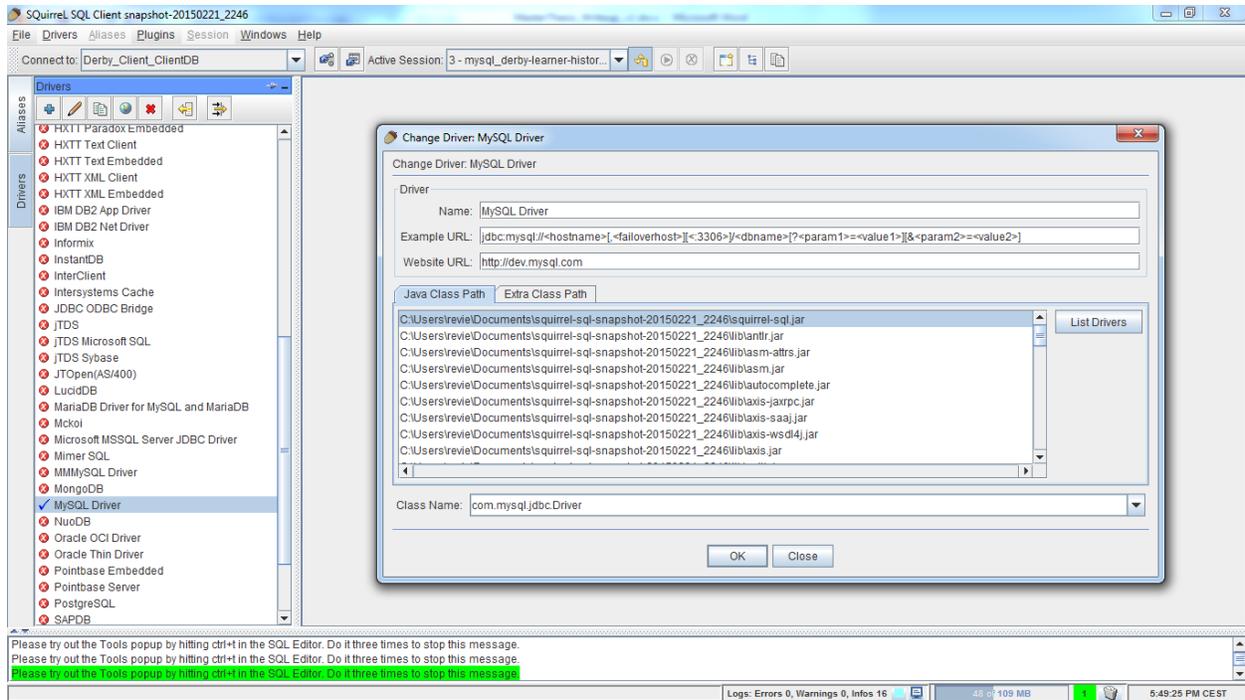
Then, the existing databases in Derby are added one by one- am_core_db, learner_history_derby_db, notes_derby_db, gapstorage_derby_db.



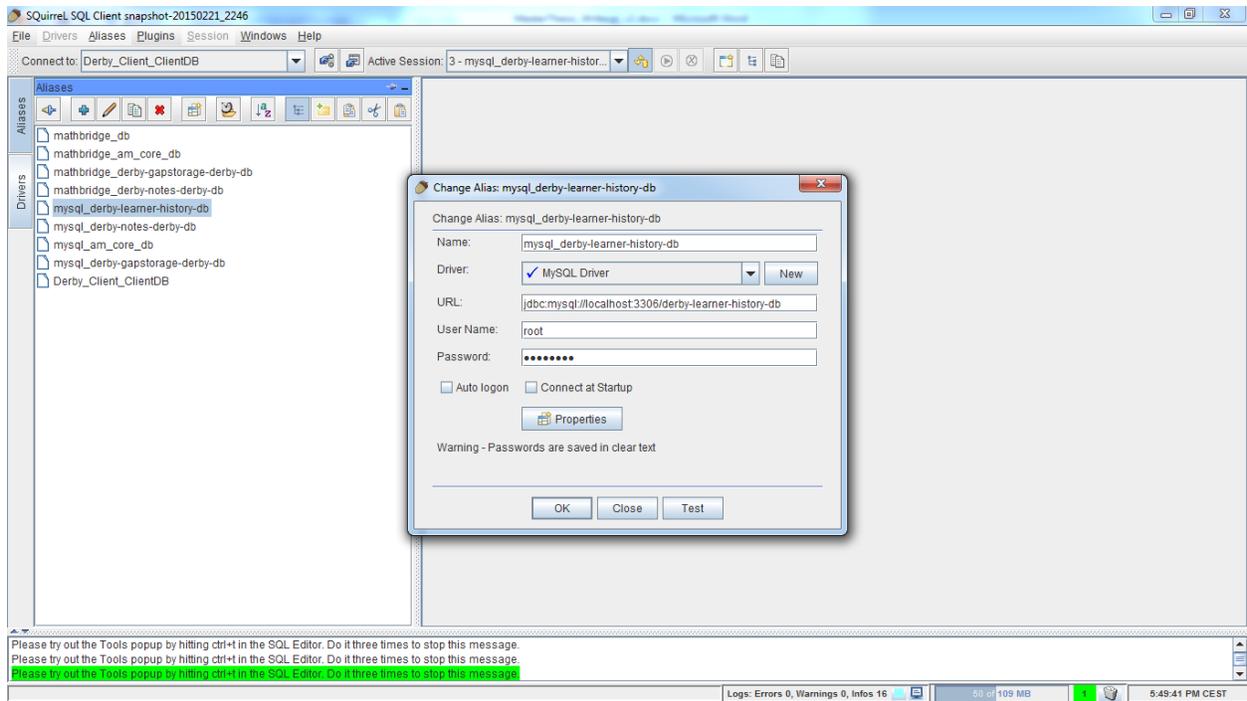
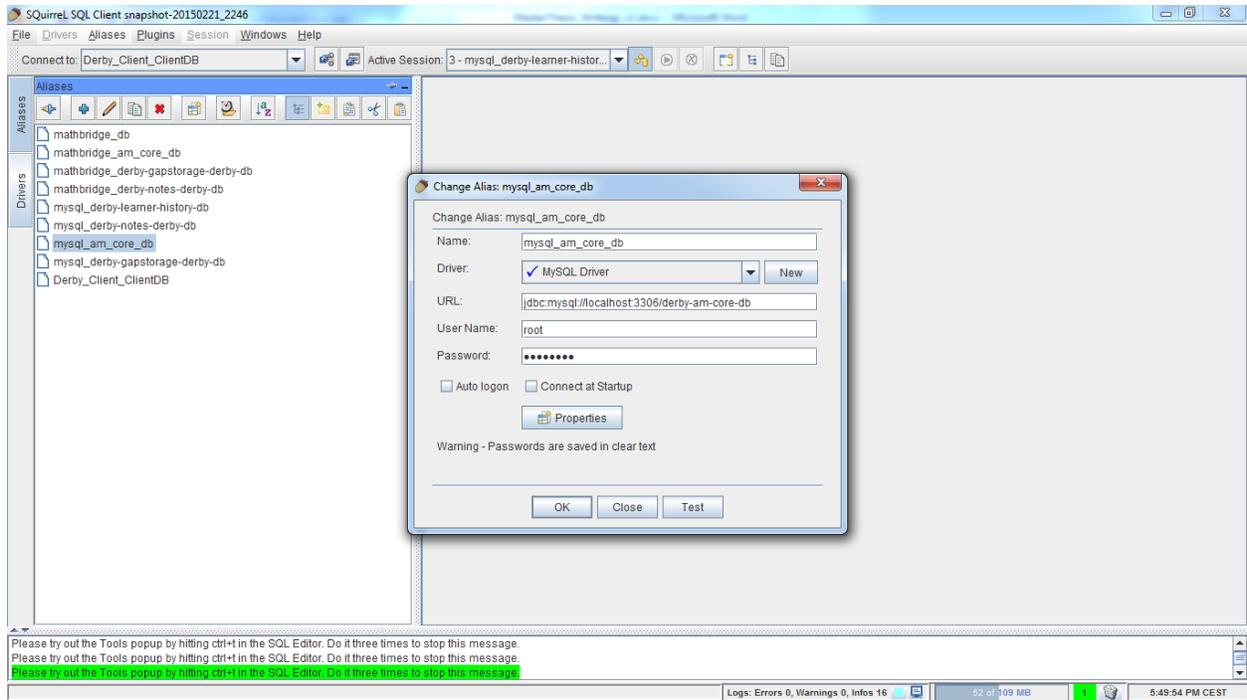


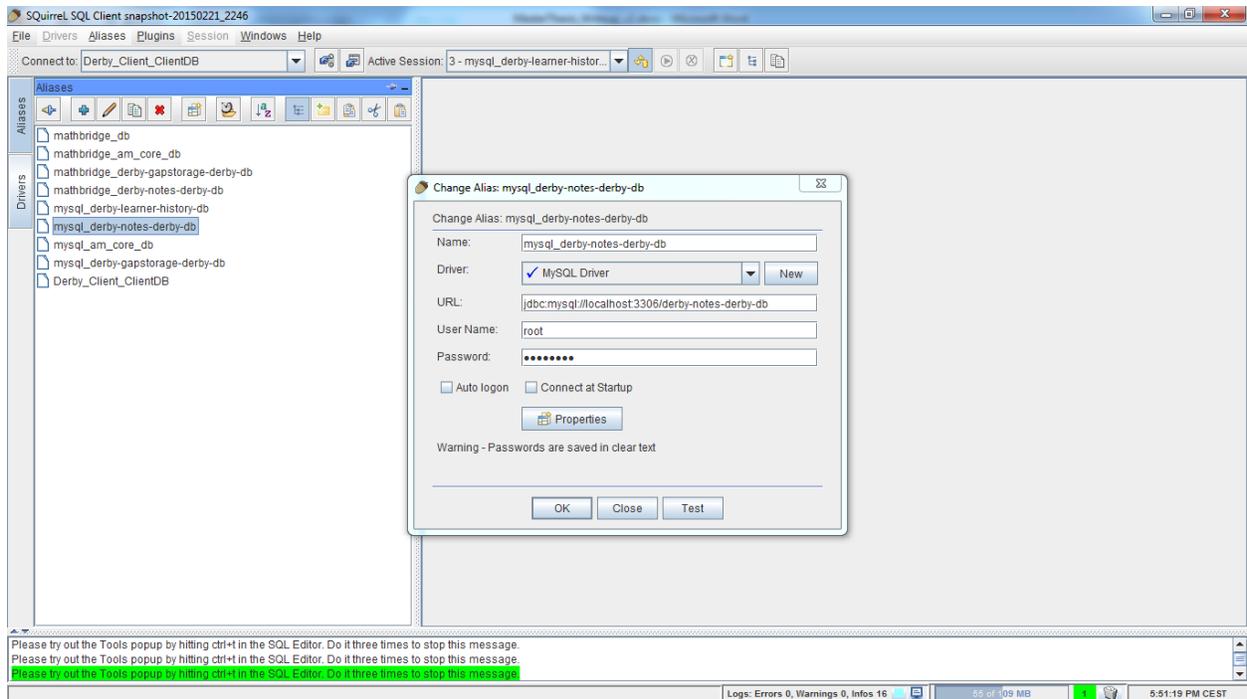
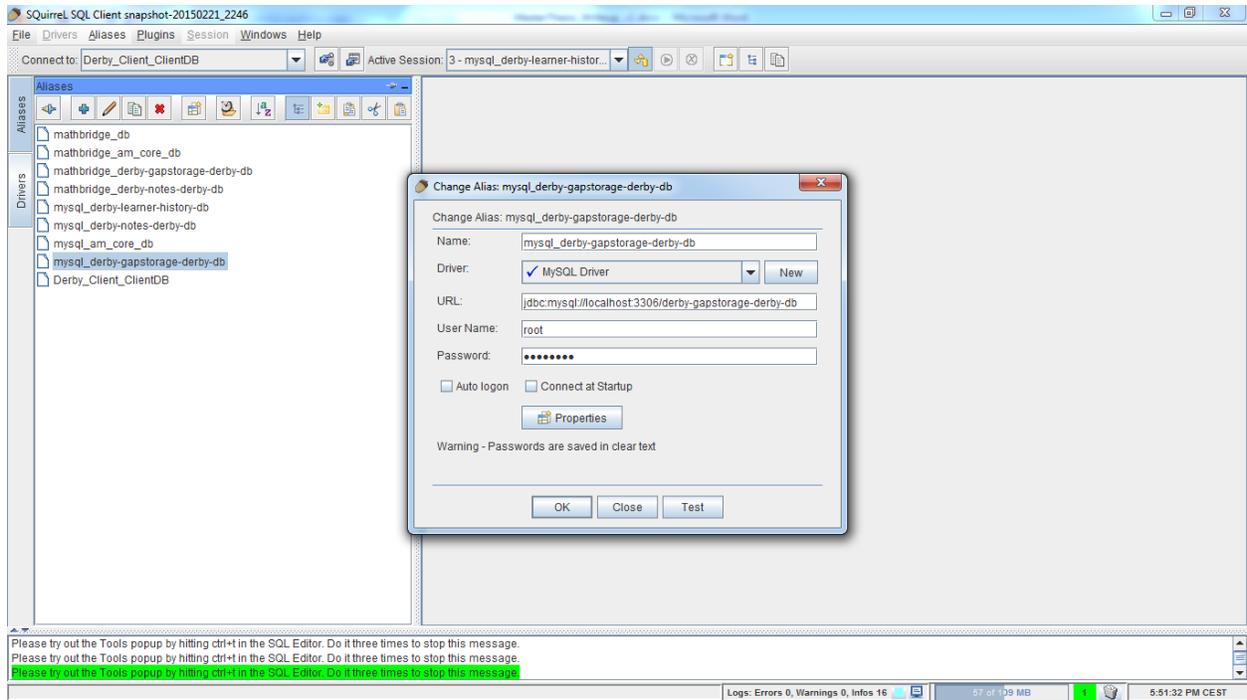
Now, similarly for the migration process the MySQL databases needed be created in localhost using MySQL Server Client. Then these empty database schemas are filled up with the data from their respective Derby database counterparts.

First, the MySQL driver is added as follows.



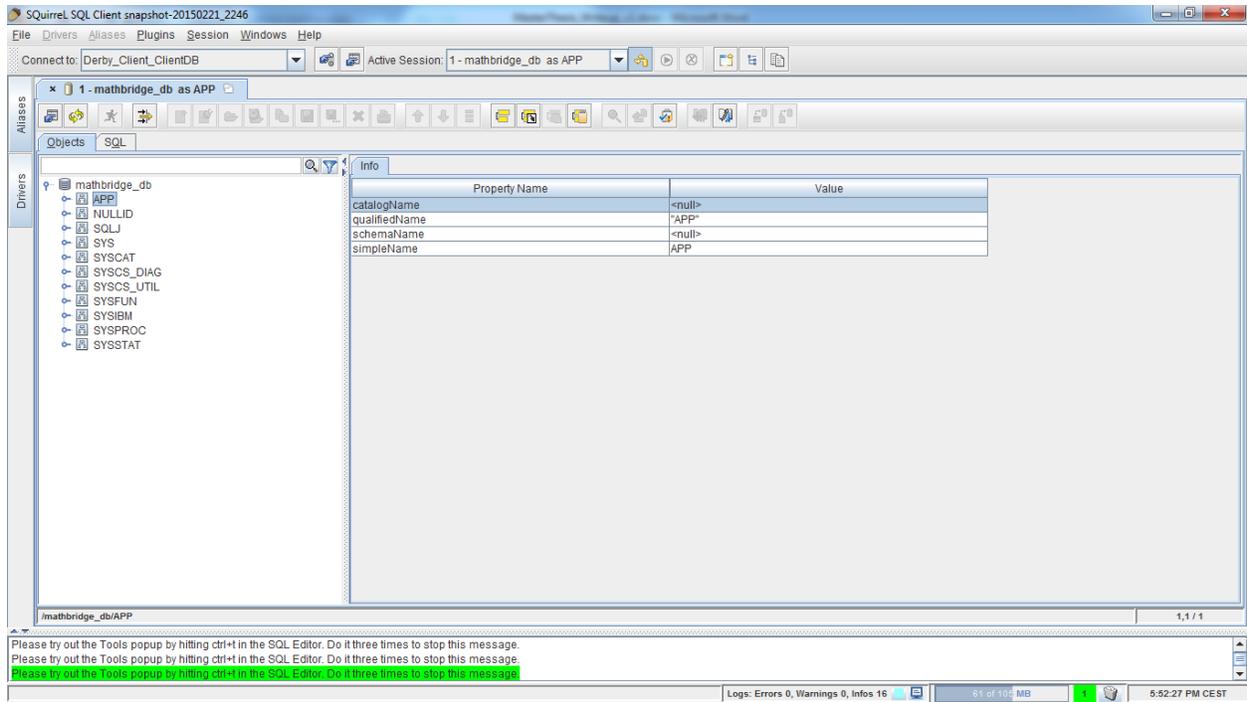
Then, the empty MySQL databases created before are connected with the MySQL Driver. This is done for all the 4 databases as before- am_core_db, learner_history_derby_db, notes_derby_db, gapstorage_derby_db.





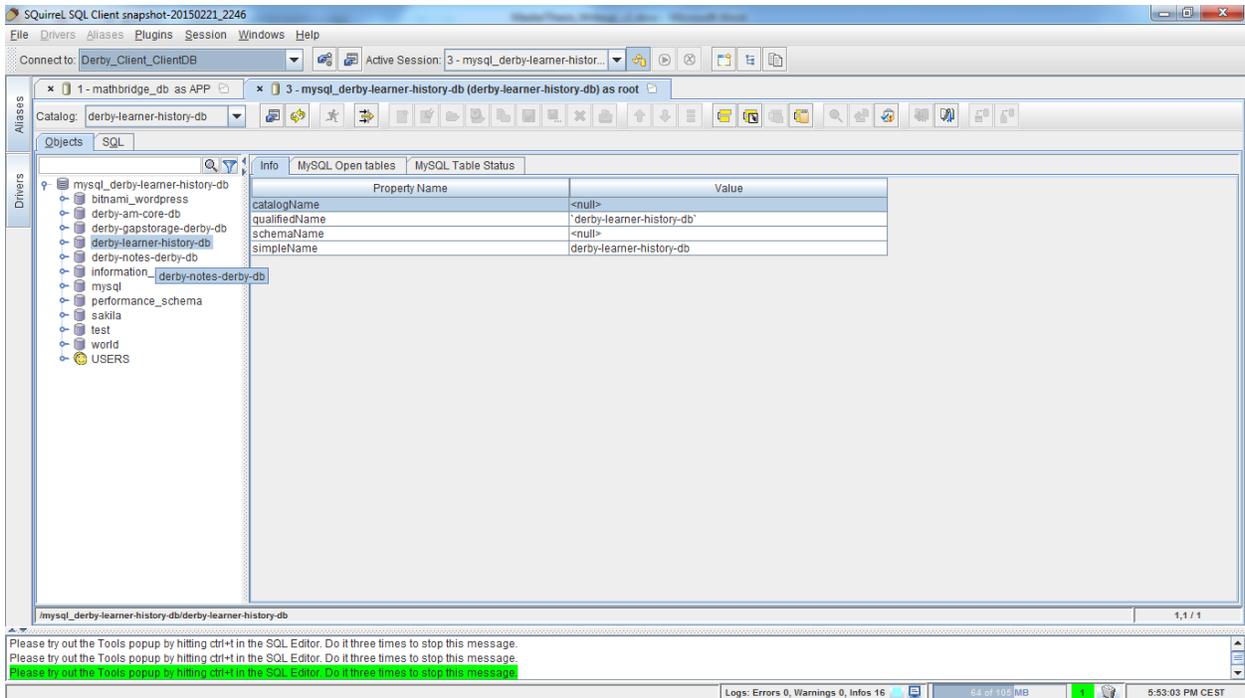
Migration takes place one by one from one database to another, from one table to another.

First, the connection to the Derby database is made and the tables are loaded.

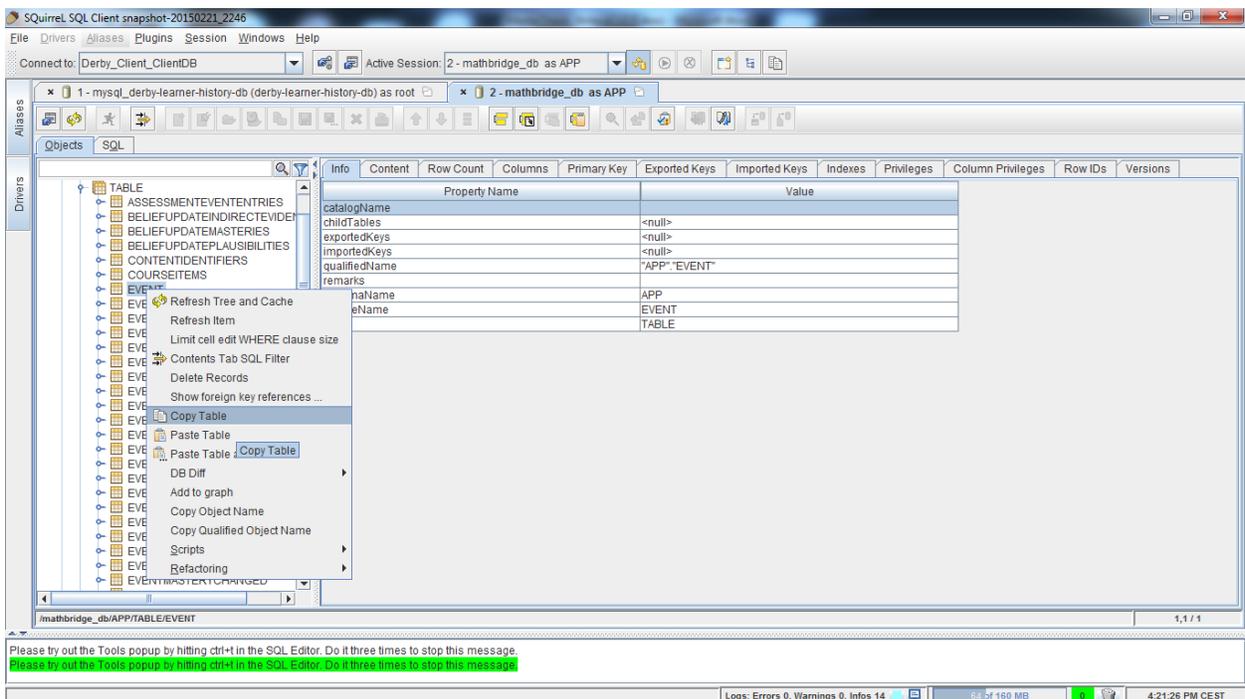


Then, connections to the MySQL databases are made.

Here, in this example it is showed how a table from am_core_db is migrated from Derby to MySQL.

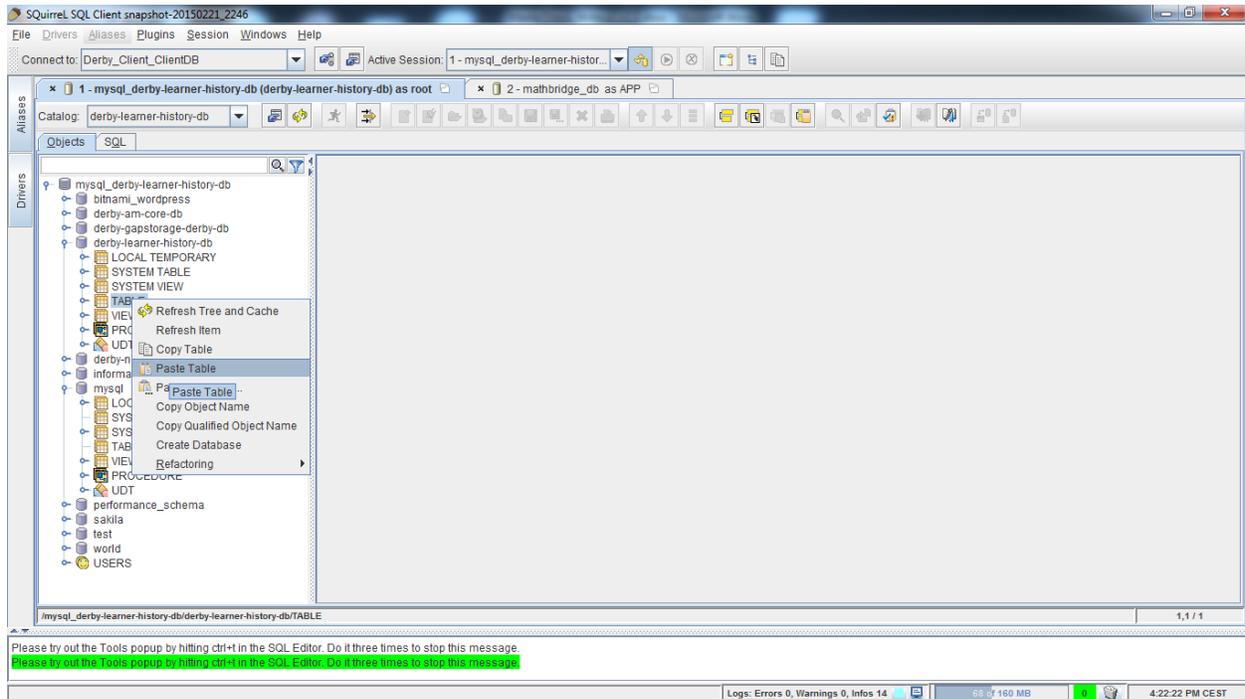


The required table is first 'right-clicked' upon and then 'Copy Table' option is selected.

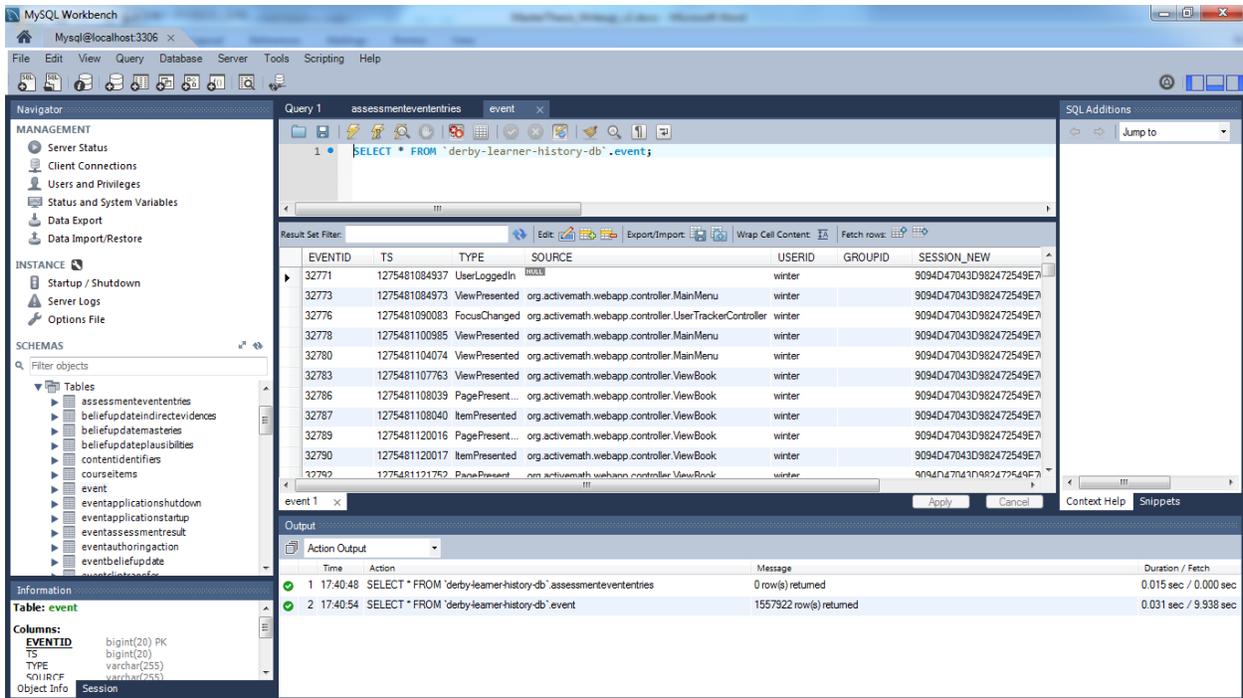


The copied table is then pasted in MySQL by choosing 'Paste Table' option as below.

This copies the Table along with all its Table structure, schema names and key constraints and is migrated into MySQL.



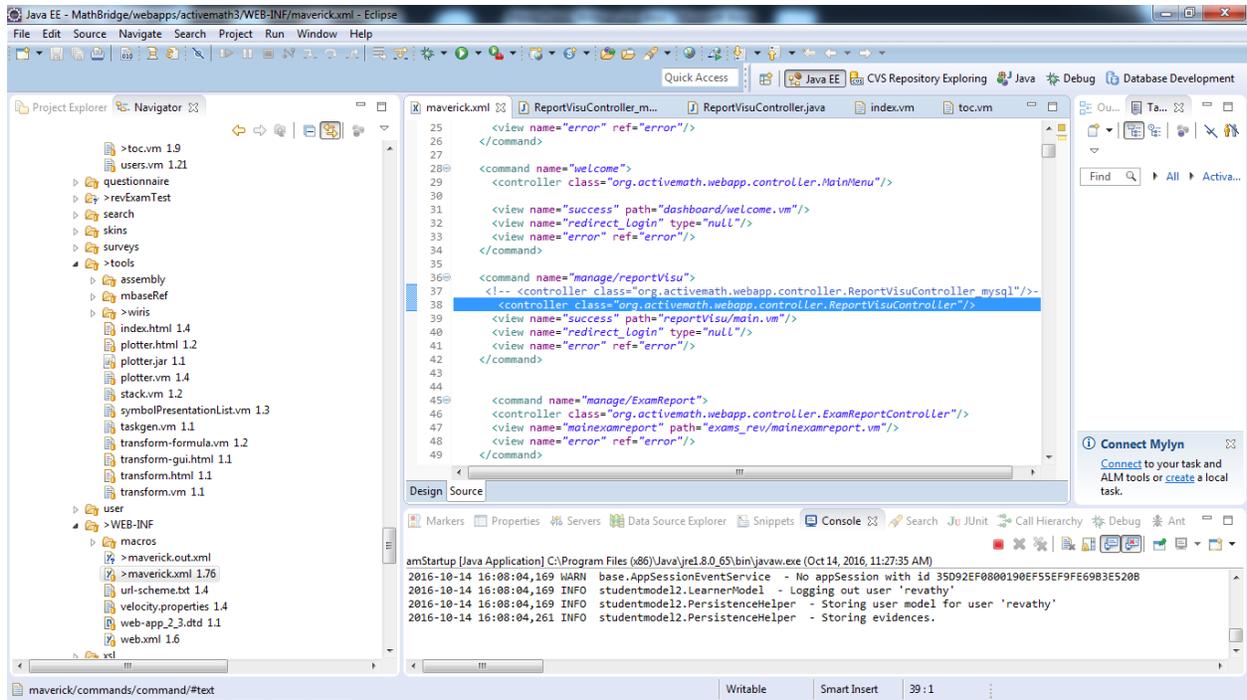
Repeating the same process for all the tables from all 4 databases in Derby migrates all the data into MySQL. This can be verified by refreshing the MySQL data using MySQL Server Client tool as below.



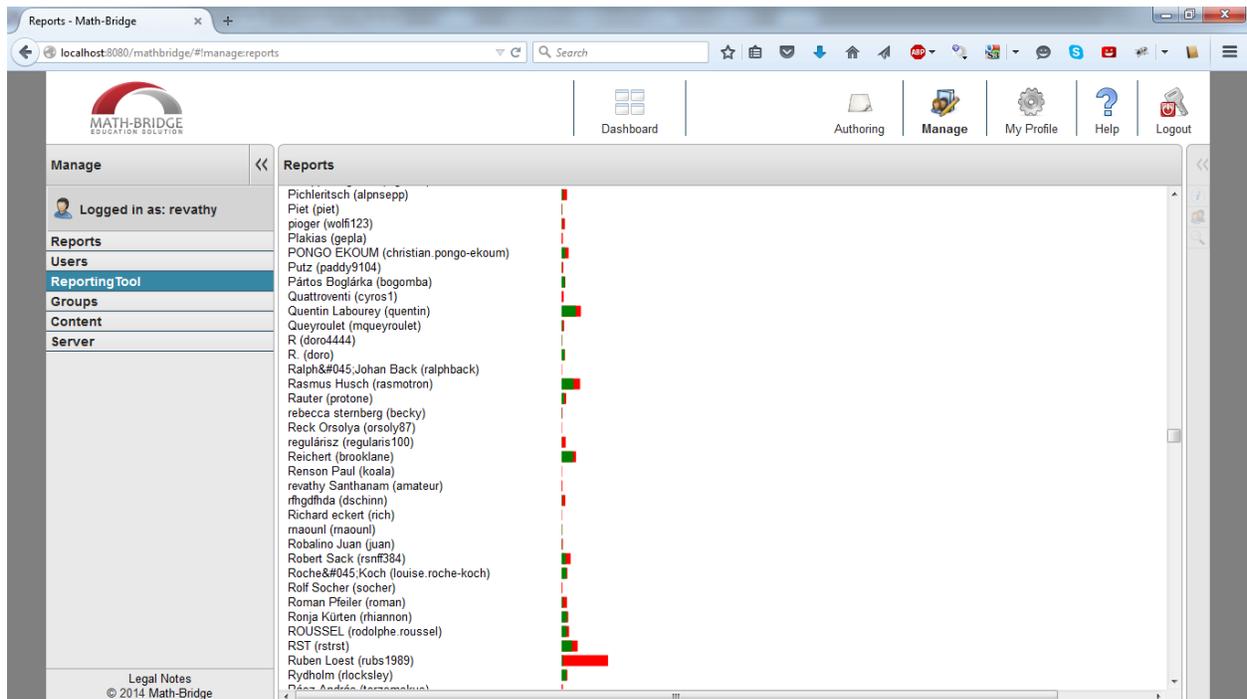
4.5 Evaluation

This section provides a comparison between the time taken by Derby Database and MySQL Database, to produce the same report from both separately, thereby showing the necessity to move data from Derby to MySQL. This is an Example Visualization Report using MySQL data that successfully connects with MySQL database and reports back JSON data which are then visualized using D3 Charts.

Firstly, connecting the Maverick command to existing report to fetch data from Derby, we get the following result.



It took 0:15,19 (mm:ss) to get the data from Derby. The user revathy santhanam (amateur) is the new test user created in Math-Bridge with 2 exercices to test that the activity details in Derby are from the server.



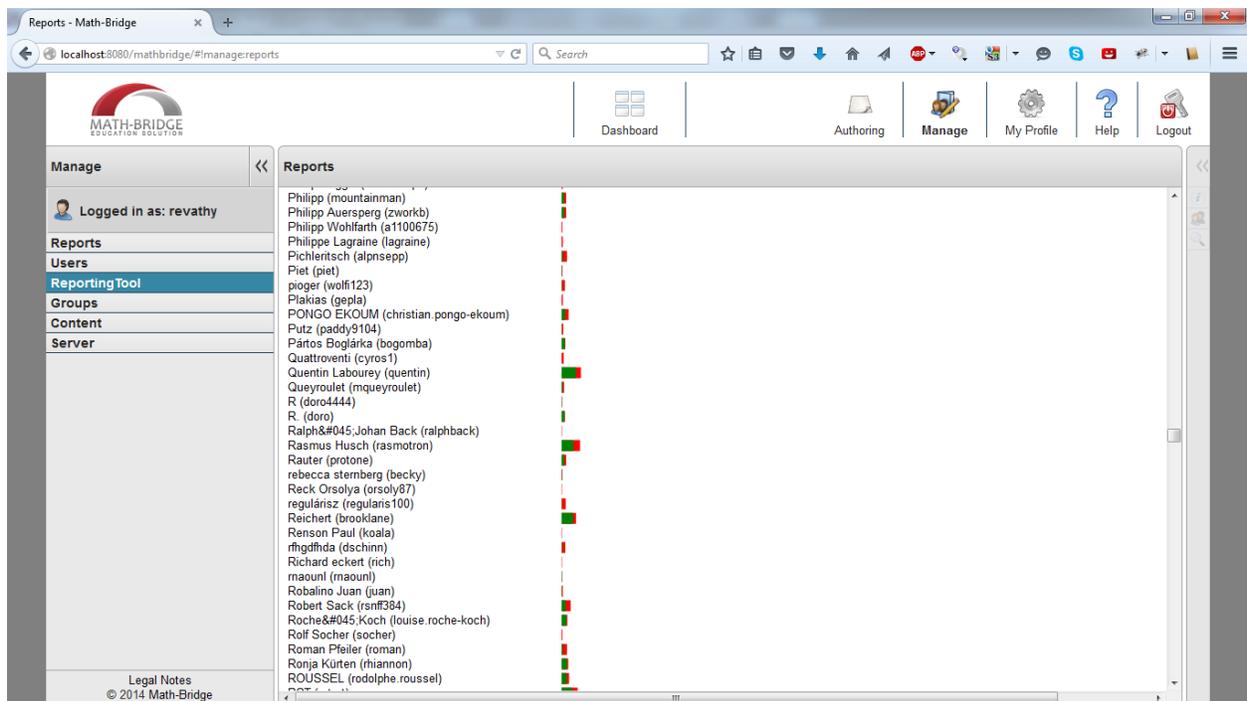
Then, connecting the Maverick command to the report to fetch data from MySQL, we get the following result.

```

103 System.out.println("newAction() has been triggered inside ReportVisuController.java");
104 return "success";
105 }
106 private Connection getConnectionFromPool(String db) throws SQLException{
107 long currentTime = Calendar.getInstance().getTime().getTime();
108 long offset = currentTime - this.lastConRefresh;
109 if(this.lastConRefresh > 0){
110 if(((offset / (1000*60*60)))>2){
111 this.connectionAMCore.close();
112 try {
113 Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/" + db, "root", "password");
114 Log.info("Reconnecting to Database");
115 } catch (Exception e) {
116 e.printStackTrace();
117 }
118 this.lastConRefresh = Calendar.getInstance().getTime().getTime();
119 return connectionAMCore;
120 }else{
121 return DriverManager.getConnection("jdbc:mysql://localhost:3306/" + db, "root", "password");
122 }
123 }
124 }
125 }
126 protected String action() throws Exception
127 {
128 System.out.println("Now inside action() of ReportVisuController.java");
129 model = this.getModel();
130 //String action = request.getParameter("action");
131 //System.out.println("Context path: "+request.getContextPath());
132 try
133 {
134 Statement stmtUserFullNames;
135
136 ResultSet resultUserFullNames;
137 Class.forName("com.mysql.jdbc.Driver");
138 connectionAMCore= getConnectionFromPool("derby-am-core-db");
139 //
140 // stmtUserFullNames= connectionAMCore.createStatement();
141 //
142 //
143 //
144 //
145 //
146 //
147 //
148 //
149 //
150 //
151 //
152 //
153 //
154 //
155 //
156 //
157 //
158 //
159 //
160 //
161 //
162 //
163 //
164 //
165 //
166 //
167 //
168 //
169 //
170 //
171 //
172 //
173 //
174 //
175 //
176 //
177 //
178 //
179 //
180 //
181 //
182 //
183 //
184 //
185 //
186 //
187 //
188 //
189 //
190 //
191 //
192 //
193 //
194 //
195 //
196 //
197 //
198 //
199 //
200 //
201 //
202 //
203 //
204 //
205 //
206 //
207 //
208 //
209 //
210 //
211 //
212 //
213 //
214 //
215 //
216 //
217 //
218 //
219 //
220 //
221 //
222 //
223 //
224 //
225 //
226 //
227 //
228 //
229 //
230 //
231 //
232 //
233 //
234 //
235 //
236 //
237 //
238 //
239 //
240 //
241 //
242 //
243 //
244 //
245 //
246 //
247 //
248 //
249 //
250 //
251 //
252 //
253 //
254 //
255 //
256 //
257 //
258 //
259 //
260 //
261 //
262 //
263 //
264 //
265 //
266 //
267 //
268 //
269 //
270 //
271 //
272 //
273 //
274 //
275 //
276 //
277 //
278 //
279 //
280 //
281 //
282 //
283 //
284 //
285 //
286 //
287 //
288 //
289 //
290 //
291 //
292 //
293 //
294 //
295 //
296 //
297 //
298 //
299 //
300 //
301 //
302 //
303 //
304 //
305 //
306 //
307 //
308 //
309 //
310 //
311 //
312 //
313 //
314 //
315 //
316 //
317 //
318 //
319 //
320 //
321 //
322 //
323 //
324 //
325 //
326 //
327 //
328 //
329 //
330 //
331 //
332 //
333 //
334 //
335 //
336 //
337 //
338 //
339 //
340 //
341 //
342 //
343 //
344 //
345 //
346 //
347 //
348 //
349 //
350 //
351 //
352 //
353 //
354 //
355 //
356 //
357 //
358 //
359 //
360 //
361 //
362 //
363 //
364 //
365 //
366 //
367 //
368 //
369 //
370 //
371 //
372 //
373 //
374 //
375 //
376 //
377 //
378 //
379 //
380 //
381 //
382 //
383 //
384 //
385 //
386 //
387 //
388 //
389 //
390 //
391 //
392 //
393 //
394 //
395 //
396 //
397 //
398 //
399 //
400 //
401 //
402 //
403 //
404 //
405 //
406 //
407 //
408 //
409 //
410 //
411 //
412 //
413 //
414 //
415 //
416 //
417 //
418 //
419 //
420 //
421 //
422 //
423 //
424 //
425 //
426 //
427 //
428 //
429 //
430 //
431 //
432 //
433 //
434 //
435 //
436 //
437 //
438 //
439 //
440 //
441 //
442 //
443 //
444 //
445 //
446 //
447 //
448 //
449 //
450 //
451 //
452 //
453 //
454 //
455 //
456 //
457 //
458 //
459 //
460 //
461 //
462 //
463 //
464 //
465 //
466 //
467 //
468 //
469 //
470 //
471 //
472 //
473 //
474 //
475 //
476 //
477 //
478 //
479 //
480 //
481 //
482 //
483 //
484 //
485 //
486 //
487 //
488 //
489 //
490 //
491 //
492 //
493 //
494 //
495 //
496 //
497 //
498 //
499 //
500 //
501 //
502 //
503 //
504 //
505 //
506 //
507 //
508 //
509 //
510 //
511 //
512 //
513 //
514 //
515 //
516 //
517 //
518 //
519 //
520 //
521 //
522 //
523 //
524 //
525 //
526 //
527 //
528 //
529 //
530 //
531 //
532 //
533 //
534 //
535 //
536 //
537 //
538 //
539 //
540 //
541 //
542 //
543 //
544 //
545 //
546 //
547 //
548 //
549 //
550 //
551 //
552 //
553 //
554 //
555 //
556 //
557 //
558 //
559 //
560 //
561 //
562 //
563 //
564 //
565 //
566 //
567 //
568 //
569 //
570 //
571 //
572 //
573 //
574 //
575 //
576 //
577 //
578 //
579 //
580 //
581 //
582 //
583 //
584 //
585 //
586 //
587 //
588 //
589 //
590 //
591 //
592 //
593 //
594 //
595 //
596 //
597 //
598 //
599 //
600 //
601 //
602 //
603 //
604 //
605 //
606 //
607 //
608 //
609 //
610 //
611 //
612 //
613 //
614 //
615 //
616 //
617 //
618 //
619 //
620 //
621 //
622 //
623 //
624 //
625 //
626 //
627 //
628 //
629 //
630 //
631 //
632 //
633 //
634 //
635 //
636 //
637 //
638 //
639 //
640 //
641 //
642 //
643 //
644 //
645 //
646 //
647 //
648 //
649 //
650 //
651 //
652 //
653 //
654 //
655 //
656 //
657 //
658 //
659 //
660 //
661 //
662 //
663 //
664 //
665 //
666 //
667 //
668 //
669 //
670 //
671 //
672 //
673 //
674 //
675 //
676 //
677 //
678 //
679 //
680 //
681 //
682 //
683 //
684 //
685 //
686 //
687 //
688 //
689 //
690 //
691 //
692 //
693 //
694 //
695 //
696 //
697 //
698 //
699 //
700 //
701 //
702 //
703 //
704 //
705 //
706 //
707 //
708 //
709 //
710 //
711 //
712 //
713 //
714 //
715 //
716 //
717 //
718 //
719 //
720 //
721 //
722 //
723 //
724 //
725 //
726 //
727 //
728 //
729 //
730 //
731 //
732 //
733 //
734 //
735 //
736 //
737 //
738 //
739 //
740 //
741 //
742 //
743 //
744 //
745 //
746 //
747 //
748 //
749 //
750 //
751 //
752 //
753 //
754 //
755 //
756 //
757 //
758 //
759 //
760 //
761 //
762 //
763 //
764 //
765 //
766 //
767 //
768 //
769 //
770 //
771 //
772 //
773 //
774 //
775 //
776 //
777 //
778 //
779 //
780 //
781 //
782 //
783 //
784 //
785 //
786 //
787 //
788 //
789 //
790 //
791 //
792 //
793 //
794 //
795 //
796 //
797 //
798 //
799 //
800 //
801 //
802 //
803 //
804 //
805 //
806 //
807 //
808 //
809 //
810 //
811 //
812 //
813 //
814 //
815 //
816 //
817 //
818 //
819 //
820 //
821 //
822 //
823 //
824 //
825 //
826 //
827 //
828 //
829 //
830 //
831 //
832 //
833 //
834 //
835 //
836 //
837 //
838 //
839 //
840 //
841 //
842 //
843 //
844 //
845 //
846 //
847 //
848 //
849 //
850 //
851 //
852 //
853 //
854 //
855 //
856 //
857 //
858 //
859 //
860 //
861 //
862 //
863 //
864 //
865 //
866 //
867 //
868 //
869 //
870 //
871 //
872 //
873 //
874 //
875 //
876 //
877 //
878 //
879 //
880 //
881 //
882 //
883 //
884 //
885 //
886 //
887 //
888 //
889 //
890 //
891 //
892 //
893 //
894 //
895 //
896 //
897 //
898 //
899 //
900 //
901 //
902 //
903 //
904 //
905 //
906 //
907 //
908 //
909 //
910 //
911 //
912 //
913 //
914 //
915 //
916 //
917 //
918 //
919 //
920 //
921 //
922 //
923 //
924 //
925 //
926 //
927 //
928 //
929 //
930 //
931 //
932 //
933 //
934 //
935 //
936 //
937 //
938 //
939 //
940 //
941 //
942 //
943 //
944 //
945 //
946 //
947 //
948 //
949 //
950 //
951 //
952 //
953 //
954 //
955 //
956 //
957 //
958 //
959 //
960 //
961 //
962 //
963 //
964 //
965 //
966 //
967 //
968 //
969 //
970 //
971 //
972 //
973 //
974 //
975 //
976 //
977 //
978 //
979 //
980 //
981 //
982 //
983 //
984 //
985 //
986 //
987 //
988 //
989 //
990 //
991 //
992 //
993 //
994 //
995 //
996 //
997 //
998 //
999 //
1000 //

```

The resulting Report data is as follows. This does not have the user revathy santhanam (amateur) since the below data is from MySQL and above was from Derby, where the user's data is.



It took 0:09,20 (mm:ss) to get the data from MySQL. This proves that data migration to MySQL would be of a great benefit for Math-Bridge when multiple users are accessing the system at a time and when huge amount of data is involved.

Chapter 5

Learning Analytics Framework

5.1 Architecture

As discussed in Chapter1, this thesis creates a new Framework for the operation of Math-Bridge without the use of an additional Program called ‘mybatis’; thereby making it easier for data access and faster operations. This also provides no surprises in the end, since it is a straightforward approach.

Originally, any reports run on Math-Bridge server gets results through the ‘Maverick’ Framework, with request to the ‘mybatis’ which then provides the resultant data as XML content from Derby database and returns it to Maverick which is then produced on screen for the user requesting the report. In this approach, all the internal processes as to how Maverick manipulates with the XML is unknown and how the data is displayed is a mystery on its own. All the database related statements are written in XML file and there is no proper Java Implementations involved.

In this Thesis, A whole new Approach is devised which provides proper Java Implementations to all the interfaces and a proper format to get data from MySQL is followed all throughout the exam database. There is no more ‘Magic!’ but rather coding. Now Maverick requests data from the Java Controller which then directly runs SQL queries on the MySQL database and the results are returned to the Controller and displayed at user-end and the example HTML file which runs an Ajax query provides a success notification if the right data comes back.

This framework can be made use of to do reporting on Math-Bridge with exam related information and provide faster results compared to its ‘mybatis’ counterpart because of No-XML

approach, the data is retrieved faster using just Java Coding on the client end in comparison to performing huge operations at the Server-end. Thus, the server load is reduced.

The whole issue with the old way of doing things and the new way is depicted below in the flow chart diagram. In the diagram, it is seen that previously a 'mybatis' program was made use of to perform the database operations. The mybatis worked together with maverick to provide the results from the database onto an xml-file that was read out on the screen. But, now mybatis is completely removed and maverick gets commands directly from the MySQL.

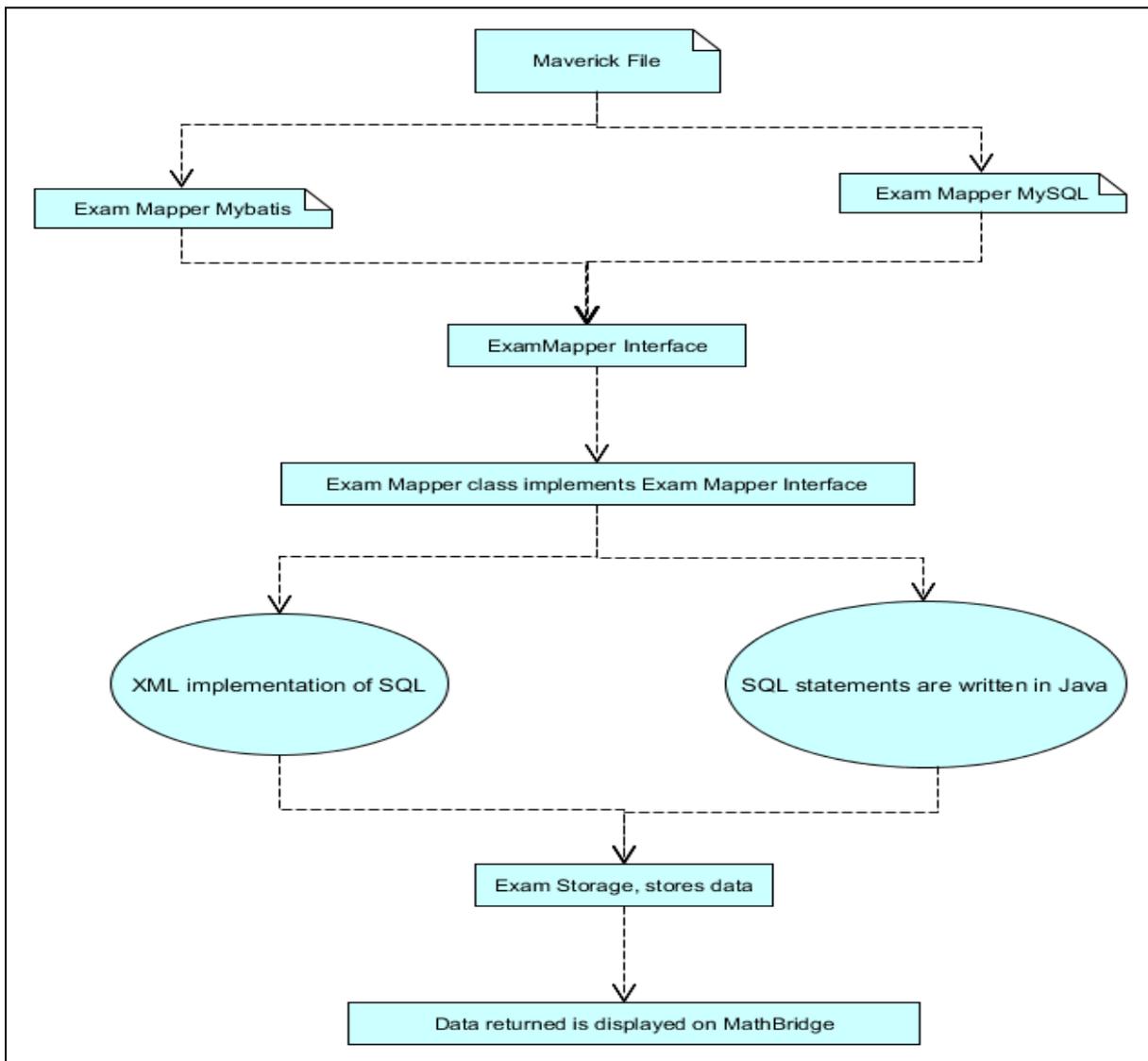


Figure 5.1: Difference between Old and New approaches

The existing system uses the below format of XML coding to query the Derby Database. It uses XML to Delete, Select, Insert or Update the SQL data. This was completely removed in the new Framework.

```
9      </select>
10     <delete id="deleteExam" parameterType="String">
11         DELETE FROM Exams WHERE id = #{id}
12     </delete>
13
14     <select id="listExams" resultMap="examResultMap">
15         SELECT * FROM Exams
16     </select>
17     <resultMap id="examResultMap" type="Exam">
18         <constructor>
19             <i>idArg column="id" javaType="String"/>
20         </constructor>
21         <collection property="pointsFromList" select="listPointEntries" column="id" />
22     </resultMap>
23
24     <insert id="insertExam" parameterType="Exam">
25         INSERT INTO Exams
26             (id, bookId, pointsMax, startTime, endTime, duration, isExam, isCompulsory, creator, title)
27         VALUES
28             (#{id}, #{bookId}, #{pointsMax}, #{startTime}, #{endTime}, #{durationSecs}, #{isExam}, #{isCompulsory}, #{creator}, #{title})
29     </insert>
30     <update id="updateExam" parameterType="Exam">
31         UPDATE Exams SET
32             bookId = #{bookId},
33             pointsMax = #{pointsMax},
34             startTime = #{startTime},
35             endTime = #{endTime},
36             duration = #{durationSecs},
37             isExam = #{isExam},
38             isCompulsory = #{isCompulsory},
39             creator = #{creator},
40             title = #{title}
41         WHERE id = #{id}
42     </update>
43
44     <insert id="insertPointEntry" parameterType="ExamPointEntry" useGeneratedKeys="true" keyProperty="id">
45         INSERT INTO examPoints
46             (examId, exerciseId, points)
```

With only Java, SQL queries can be easily transformed without using XML. This is shown below and it is much simpler than before to get the required data or insert any data into MySQL. No use of extra conversions needed by the server and it is free to do other works instead of converting XML data.

For example, the different ways of writing updateExam procedure in both old and new ways are shown above and below.

```
110 }
111 }
112 @Override
113 public void updateExam(Exam exam) {
114     // TODO Auto-generated method stub
115     query="UPDATE Exams SET bookId = #{bookId}, pointsMax = #{pointsMax}, startTime = #{startTime}, endTime = #{endTime}, duration = #{durationSecs}, isExam = #{isExam},
116
117     try {
118         rs= stm.executeQuery(query);
119     } catch (SQLException e) {
120         // TODO Auto-generated catch block
121         e.printStackTrace();
122     }
123     finally {
124         // try { if (stm != null) stm.close(); } catch (Exception e) {};
125         //try { if (con != null) con.close(); } catch (Exception e) {};
126     }
127 }
128
129 @Override
130 public boolean existsExam(Exam exam) {
131     // TODO Auto-generated method stub
132     query="SELECT COUNT(*) FROM Exams WHERE id = " + exam.getId()+"";
133     try {
134         rs= stm.executeQuery(query);
135     } catch (SQLException e) {
136         // TODO Auto-generated catch block
137         e.printStackTrace();
138     }
139     finally {
140         // try { if (stm != null) stm.close(); } catch (Exception e) {};
141         // try { if (con != null) con.close(); } catch (Exception e) {};
142     }
143     return true;
144 }
145 }
146
147 @SuppressWarnings("unchecked")
148 @Override
```

5.2 Example Report – 1

Using the above Framework, an example Report is developed in Java. This Report is a Java Controller that uses Ajax to query the MySQL. First, Maverick runs the controller and this controller creates an Ajax call to the highest level Exam Interface implemented. This interface in turn using Java SQL Statements queries the MySQL Database and gets back the Exam entries and returns to the Controller, which then displays the returned row and a success notification.

Below it is shown how the new command is given to Maverick to get the data. A new html file is created to use AJAX query and it is given as input to maverick. When it succeeded to load that html file, it is asked to check the MySQL database for exam id =1, if it is found then the message is displayed that the controller returned successfully.

```
Java EE - MathBridge/webapps/activemath3/WEB-INF/maverick.xml - Eclipse
File Edit Source Navigate Search Project Run Window Help
Quick Access Java EE CVS Repository Exploring Java Debug Database Development
maverick.xml ReportVisuController_mysqljava ReportVisuController.java index.vm tocvm ExamTest.html mainexamreport.vm
24 <view name="redirect_login" type="null"/>
25 <view name="error" ref="error"/>
26 </command>
27
28 <command name="welcome">
29 <controller class="org.activemath.webapp.controller.MainMenu"/>
30
31 <view name="success" path="dashboard/welcome.vm"/>
32 <view name="redirect_login" type="null"/>
33 <view name="error" ref="error"/>
34 </command>
35
36 <command name="manage/reportVisu">
37 <controller class="org.activemath.webapp.controller.ReportVisuController_mysql"/>
38 <!-- <controller class="org.activemath.webapp.controller.ReportVisuController"/> -->
39 <view name="success" path="reportVisu/main.vm"/>
40 <view name="redirect_login" type="null"/>
41 <view name="error" ref="error"/>
42 </command>
43
44
45 <command name="manage/ExamReport">
46 <controller class="org.activemath.webapp.controller.ExamReportController"/>
47 <view name="mainexamreport" path="exams_rev/mainexamreport.vm"/>
48 <view name="error" ref="error"/>
49 </command>
50
51
52
53 <command name="dashboard">
54 <controller class="org.activemath.webapp.controller.MainMenu"/>
55
56 <view name="success" path="dashboard/dashboard.vm"/>
57 <view name="redirect_login" type="null"/>
58 <view name="error" ref="error"/>
59 </command>
60 <command name="help">
61 <controller class="org.activemath.webapp.controller.HelpController"/>

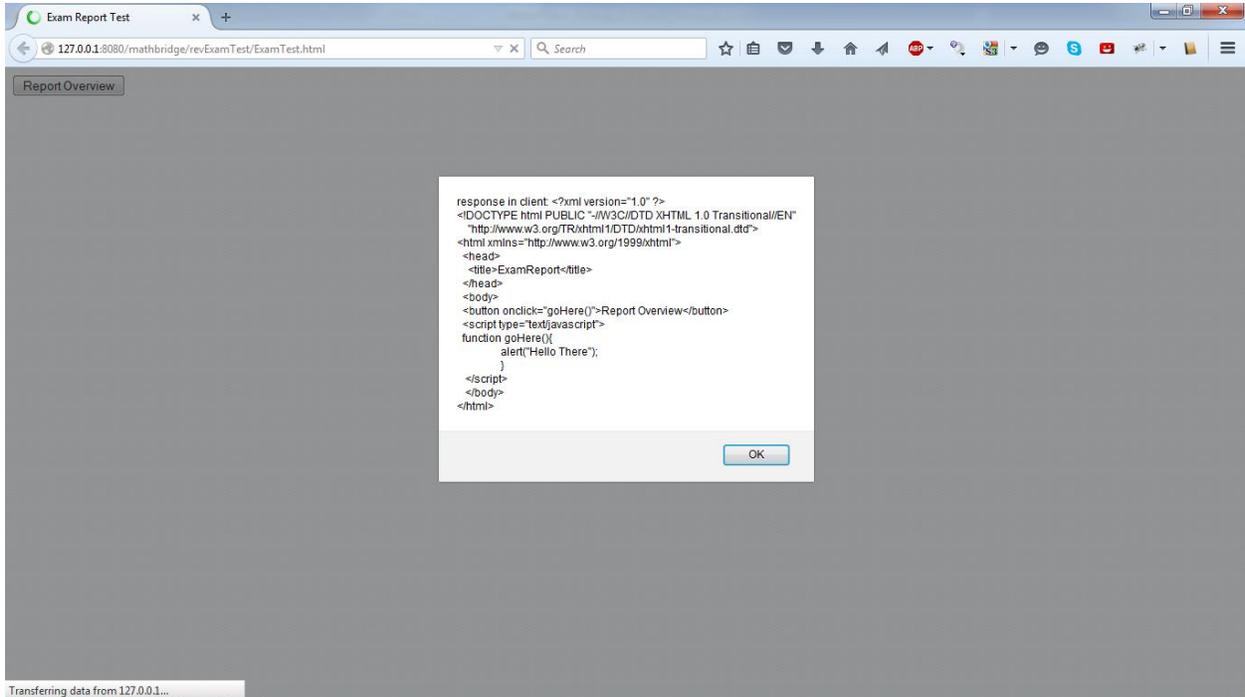
```

The ajax call returns the response from the vm file, if it returns successfully else an error response is displayed. At the same time, if maverick was not able to contact the controller itself, then an error is returned.

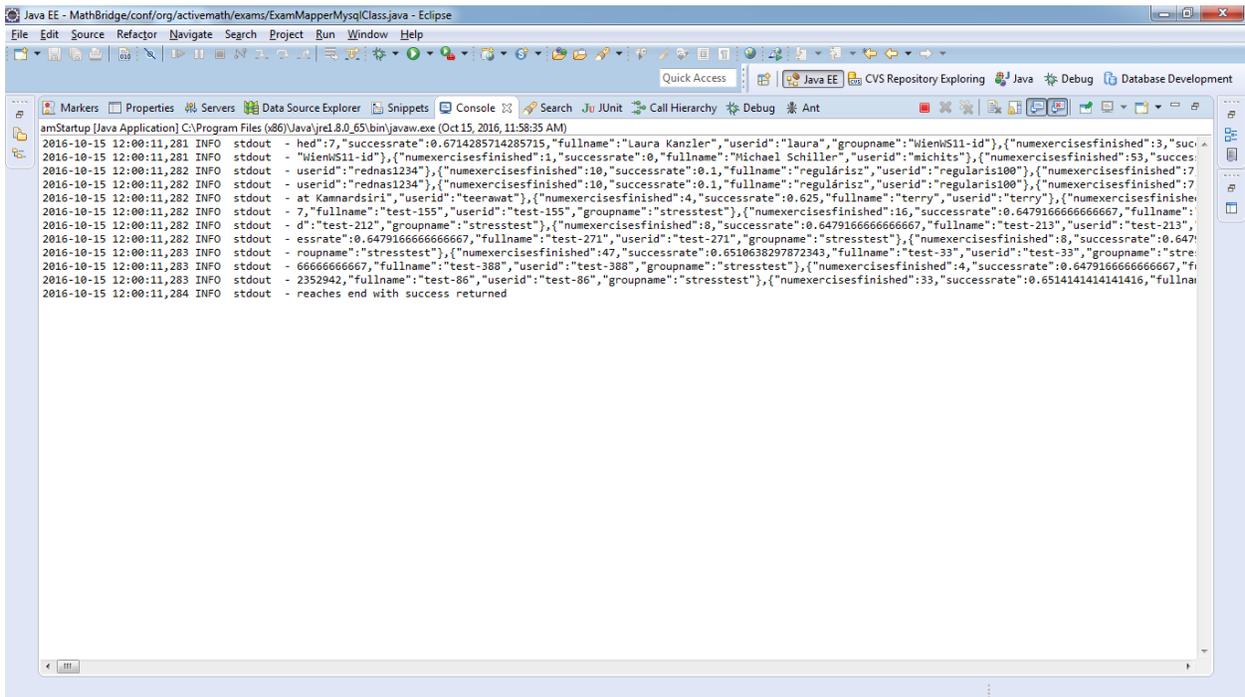
```
Java EE - MathBridge/webapps/activemath3/revExamTest/ExamTest.html - Eclipse
File Edit Source Navigate Search Project Run Window Help
Quick Access Java EE CVS Repository Exploring Java Debug Database Development
maverick.xml ReportVisuController_mysqljava ReportVisuController.java index.vm tocvm ExamTest.html
1 |
2 <html>
3 <head>
4
5 <title>Exam Report Test</title>
6
7 </head>
8 <body>
9 <div id="out">
10 <script src="/mathbridge/Lib/jquery-1.6.4.js"></script>
11 <script>
12 jQuery.ajax({
13     type: "GET",
14     url: "/mathbridge/manage/ExamReport.cmd",
15     async: false,
16     success: (function(resp){
17         $("#out").append(resp)
18         if(resp != "invalid"){
19             alert("response in client: "+resp);
20         }
21         else
22             alert("ERROR response from ExamReportController 'invalid'")
23     })),
24     error: (function(err){
25         console.log(err)
26         $("#out").append(err.responseText)
27         alert("ERROR contacting ExamReportController");
28     })
29 });
30 </script>
31 </body>
32 </html>

```

The response from the client is displayed on the ajax html page as below, since it succeeded to get the exam id back.



The below resultset shows the data returned from MySQL in the java console for the report.



The test data are the ones in MySQL, that are brought back to be displayed on the report. The same data is shown in MySQL as well.

The screenshot shows the 'Reports' page in the Math-Bridge application. The left sidebar contains navigation options: Manage, Reports, Users, Reporting Tool (highlighted), Groups, Content, and Server. The main area displays a list of test items from 'test-67' to 'test-98'. Each item has a vertical progress bar to its right, with green and red segments indicating completion status. The interface includes a search bar, a dashboard icon, and user navigation links like 'Authoring', 'Manage', 'My Profile', 'Help', and 'Logout'.

The screenshot shows MySQL Workbench with a query executed: `SELECT * FROM `derby-am-core-db`.amuser;`. The 'Navigator' pane on the left shows the database structure, including the 'amuser' table. The 'Output' pane at the bottom shows the execution results:

Time	Action	Message	Duration / Fetch
1 12:22:07	SELECT * FROM `derby-learner-history-db`.userdata	0 row(s) returned	0.016 sec / 0.000 sec
2 12:22:20	SELECT * FROM `derby-am-core-db`.amuser	1964 row(s) returned	0.047 sec / 0.000 sec

The 'Information' pane shows details for the 'amuser' table, including its columns: `ID` (varchar(255) PK).

Chapter 6

Conclusion and Discussion

6.1 Contribution of this thesis

This Thesis migrates all the data from Derby Database into MySQL Server successfully. In this thesis, a new Framework for Math-Bridge is provided wherein the Exam data can be easily accessed directly from MySQL without any extra platforms. This makes reporting easier and faster. This Platform also enables several connections handling mechanism, in order to aid multiple user(s) requests at the same time.

This thesis also provides an Example Visualization Report using MySQL data that successfully connects with MySQL database and reports back JSON data which are then visualized using D3 Charts. There is also a comparison provided of the time taken to produce the same report from both Derby Database and MySQL Database separately, thereby showing the necessity to move data from Derby to MySQL.

6.2 Limitation of this thesis

This Thesis provides a Framework only for the Exam Information and can be used for reporting out of only Exam Related information. The provided Visualization Report is visualized using D3 Charts and provides only the information for user-related information, like user names, user activities, group names, etc.

6.3 Future work

The work done in this thesis can be further adapted to generate many other forms of reports using various data available from Math-Bridge like Assessment Analysis, Performance Analysis, Student capacity, Efficiency Calculation, etc. The same D3 charts can be used to get varied reports from all the available data.

Similar to the Framework provided in this thesis for Exam Information, the entire System can be re-modelled to completely make use of MySQL, thereby overthrowing usage of 'mybatis' and 'Derby'. This will improve the performance of the system on the whole and not just the Reporting Tools. It will also be powerful enough to handle multiple requests at the same time from multiple users at the system level without overloading the Server.

References

- [1] Woolf, B. 1992. AI in Education. [*Encyclopedia of Artificial Intelligence*](#), Shapiro, S., ed., John Wiley & Sons, Inc., New York, pp. 434-444.
- [2] Learning Analytics, Eric Duval, Department of Computer Science, KU Leuven, Leuven, Belgium
- [3] Penetrating the Fog: Analytics in Learning and Education, Long & Siemens, 2011
- [4] U.S. Department of Education, Office of Educational Technology, 2013
- [5] Adaptive Learning Systems Focused Program, Harris Liebergot, <http://www.atp.nist.gov/atp/97wp-lt.htm>
- [6] Chapter 31 Math-Bridge: Closing Gaps in European Remedial Mathematics with Technology-Enhanced Learning; Sergey Sosnovsky, Michael Dietrich, Eric Andrès, George Gogvadze, Stefan Winterstein, Paul Libbrecht, Jörg Siekmann, Erica Melis
- [7] <http://pslcdatashop.org/about/faq.html>
- [8] A Data Repository for the EDM community: The PSLC DataShop, Kenneth R. Koedinger, Ryan S.J.d. Baker, Kyle Cunningham, Alida Skogsholm, Brett Leber, John Stamper; Human-Computer Interaction Institute, Carnegie Mellon University, Department of Social Science and Policy Studies, Worcester Polytechnic Institute
- [9] Learning Factors Analysis - A general method for cognitive model evaluation and improvement. Proceedings of the 8th International Conference on Intelligent Tutoring Systems, Cen, H., Koedinger, K., and Junker, B. 2006.
- [10] MetaMath, Mathgear group by Michael Dietrich (from DFKI)
- [11] Dan Debrunner (djd@debrunners.com) , ApacheCon US 2004 - Las Vegas, NV
- [12] Scott Halvorson, WDTC/DPG, ATEC Forecasters Conference , Boulder Colorado 25 July 2006
- [13] A Reference Model for Learning Analytics M.A. Chatti, A.L. Dyckhoff, U. Schroeder, and H. Thüs, International Journal of Technology Enhanced Learning(IJTEL) – Special Issue on “State-of-the-Art in TEL”

- [14] Corbett, A.T., Koedinger, K.R., Anderson, J.R.: Chapter 37 Intelligent Tutoring Systems. Handbook of Human-Computer Interaction. Elsevier Science B. V. (1997)
- [15] Translating Learning into Numbers: A Generic Framework for Learning Analytics, Wolfgang Greller and Hendrik Drachsler, Journal of Educational Technology & Society Vol. 15, No. 3, Learning and Knowledge Analytics (July 2012), pp. 42-57; Published by: International Forum of Educational Technology & Society
- [16] An overview of learning analytics, Doug Clow, Teaching in Higher Education Volume 18, 2013 - Issue 6