Open Apereo 2016
100% Open for Education

UniTime: State of the Project

May 2016

Tomáš Müller
UniTime: State of the Project

- Short introduction and a few numbers
- Current release (UniTime 4.1)
- Next release (UniTime 4.2) & long term
- Walk through the new and planned features
- Course timetabling solver experiment
What is UniTime?

- Comprehensive academic scheduling solution
- Four components: course timetabling, examination timetabling, student scheduling and event management
- Open source, web-based, written in Java using modern technologies
- Using state-of-the-art optimization algorithms
- Distributed data entry and timetabling in multi-user environments
- Apereo project since March 2015
State of the Project

Achievements

• UniTime 4.1 released in March 2016
  • Following a steady schedule of one release every 15 months
• UniTime 4.2 development is well on the way
  • Planned release mid 2017
• Over 500k of lines of code (almost 600k including the CPSolver)
• About 6,000 visits of unitime.org and about 1,000 monthly downloads
• 58 institutions from 30 countries filled our voluntary registration form during the last 12 months
• Steady increase in interest and adoption from literally around the world
  • USA, Czech Republic, Pakistan, Croatia, Poland, Turkey, Peru, Kuwait, Canada, Malaysia, Spain, UAE, Palestine, Zambia, Kenya,…
… but still very little outside contributions
Current Release: UniTime 4.1

UniTime 4.1

- Released March 2016
- New room management
- Many improvements across all the components
- Integration with Ellucian Banner and Degree Works APIs
  - Student eligibility checking, enrollment synchronization, degree plans
- New RESTful APIs
- Translations can be provided using Zanata
  - English, Czech, Polish, Spanish (in progress)

See https://goo.gl/uERUxF (UniTime 4.1 Release Docs) for more details.
Room Management

• New GWT-based pages
• Ability to update rooms across academic sessions
• Additional attachments (e.g., floor plans)
• Fully customizable table of rooms
• Room Edit page
• Better room search for event management
• New RESTful API
Course Timetabling

- Class Duration Model
  - A class length can be specified in semester minutes or hours (not only minutes per week)
  - Computation can include holidays, alternative weeks, etc.

150 minutes per week
~ 1 x 150, 2 x 75, 3 x 50 regardless of the weeks

50 average weekly minutes
~ 100 mins every other week

Meeting hours/minutes (counts actual meetings)

Semester hours/minutes
(14 hours ~ 50 minutes a week)
Course Timetabling

- Cancelled Classes
  - Students can be moved away automatically or manually
  - A cancelled class can be reopen if needed (conflicts are indicated)
  - Only classes that have no students can be deleted
Course Timetabling

- **Instructional Methods**
  - Each configuration of a course can have a different instructional method (online, hybrid, traditional, etc.)
  - Students can see in their schedules

<table>
<thead>
<tr>
<th>MA 170</th>
<th>Limit</th>
<th>Time Pattern</th>
<th>Instructor</th>
<th>Time</th>
<th>Room</th>
</tr>
</thead>
<tbody>
<tr>
<td>STAT 170</td>
<td>50</td>
<td>Statistics I</td>
<td>Introductory Statistics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Configuration 1 (Traditional)</td>
<td>40</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lecture</td>
<td>40</td>
<td>1 x 50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Laboratory</td>
<td>40</td>
<td>3 x 50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recitation</td>
<td>40</td>
<td>1 x 100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lec 1</td>
<td>20</td>
<td>1 x 50</td>
<td>Newman, George</td>
<td>T 12:30p-1:20p</td>
<td>EDUC 103</td>
</tr>
<tr>
<td>Lab 1</td>
<td>10</td>
<td>3 x 50</td>
<td>Smith, John William</td>
<td>MWF 2:30p-3:20p</td>
<td>EDUC 102</td>
</tr>
<tr>
<td>Lab 2</td>
<td>10</td>
<td>3 x 50</td>
<td>Smith, John William</td>
<td>MWF 11:30a-12:20p</td>
<td>EDUC 102</td>
</tr>
<tr>
<td>Lec 2</td>
<td>20</td>
<td>1 x 50</td>
<td>Newman, George</td>
<td>T 1:30p-2:20p</td>
<td>EDUC 101</td>
</tr>
<tr>
<td>Lab 3</td>
<td>10</td>
<td>3 x 50</td>
<td>Doe, Joe</td>
<td>MWF 3:30p-4:20p</td>
<td>EDUC 102</td>
</tr>
<tr>
<td>Lab 4</td>
<td>10</td>
<td>3 x 50</td>
<td>Doe, Joe</td>
<td>MWF 1:30p-2:20p</td>
<td>EDUC 102</td>
</tr>
<tr>
<td>Rec 1</td>
<td>40</td>
<td>1 x 100</td>
<td>Newman, George</td>
<td>Th 9:30a-11:20a</td>
<td>THTR 101</td>
</tr>
<tr>
<td>Configuration 2 (Online)</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distance Learning</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dist 1</td>
<td>10</td>
<td>Arr 5 Hrs</td>
<td>Newman, George</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
UniTime 4.1: Courses

Course Timetabling

• Multiple-Major Curricula
  • Useful for dual major programs, can inherit a common part
  • These can be overridden on the multiple-major level

Example from College of Education, Masaryk University

• Each high school teacher has two approbations
• There are many possible combinations
• Curricula are created as a combination of
  • Common part (each student must have)
  • Specific courses for each major
• There are exceptions
Examination Timetabling

- Examination Status: Each examination problem can be timetabled and published at a different time
- Ability to associate examination managers with a particular problem
UniTime 4.1: Students

## Batch Student Scheduling

- Minimal Perturbation Mode: minimize changes to existing students
- Request Groups: keep students of the same group together
- Interactive Changes: manual changes after the solver has been run
- Student Filter: ability to run the solver only for a certain group of students

See the Student Scheduling in UniTime presentation on Wednesday, 11:45am in KC 912
Online Student Scheduling

- Many UI improvements in the Scheduling Assistant
- Quick Add/Drop: easy way to add/drop a single course
- Degree planning integration
  - Provides ability to retrieve degree plan from an external system
  - Build a schedule in just a few clicks
  - Initial implementation using Ellucian DegreeWorks

See the Student Scheduling in UniTime presentation on Wednesday, 11:45am in KC 912
UniTime 4.1 APIs

- RESTful JSON APIs
- Using HTTP-simple authentication or an API token
- Retrieve instructor schedule, class information, enrollments, events
- Room management
- Online student scheduling
- Data Exchange XMLs
- GWT RPCs

GET api/instructor-schedule?term=Fall2016&id=<id>
GET api/class-info?classId=<id>
GET api/events?type=ROOM&r:text=EDUC+101&term=Fall2016
GET api/events?type=PERSON&term=Fall2016&ext=<id>
GET api/enrollments?classId=<id> (or courseld, examld, eventld)

See https://goo.gl/ikdOix (UniTime 4.1 API Specs) for more details.
Next Release: UniTime 4.2

UniTime 4.2 (in development)

- Planned release mid 2017
- Instructor Scheduling (TA Assignment)
- Student Group Scheduling (Learning Communities)
- Student Scheduling (XE/DegreeWorks API, Instructional Mode)
- UniTime Mobile (Responsive Design)
- Many additional improvements across all the components (e.g., consensus date reporting)

See https://goo.gl/Bv1qQw (UniTime 4.2 Specs) for more details.
Instructor Scheduling

• Assignment of instructors to classes, respecting various constraints (availability, maximal load, required skills, same course, etc.)
• Modeled as a new optimization problem
• Instructors are to be assigned AFTER the course timetabling is done
• Typical use case: assignment of teaching assistants
• A lot of data already exist in UniTime

• On an instructor: teaching preference, list of attributes, maximal teaching load, course and time preferences
• On a course: indicate that instructor assignment needed, teaching load, attribute and instructor requirements and preferences
• UniTime will offer similar features as in course or exam timetabling (conflict statistics, interactive changes, reporting, …)
Student Scheduling

• Ellucian Banner XE interface (additional features have been recently implemented in the interface like course overrides or back-dating, asynchronous calls for automated waitlisting)

• Extension of the course request model to allow for the student to further specify which classes he/she wants to take
  • Instructional method (online, hybrid, traditional)
  • Preferred instructor
  • …

See the Student Scheduling in UniTime presentation on Wednesday, 11:45am in KC 912
UniTime 4.2: Groups

Student Group Scheduling

- UniTime allows for many ways how student conflicts can be minimized
- However, all these deal with individual student enrollments at the end
  - The solver can shuffle students around freely
- There are some universities that want to keep students together
  - For historic reasons
  - Or as a way of making students feel like there is a community
- Can be modeled with student groups, each with a list of students and courses that the group needs to take
- This information can be used in course timetabling as well as in student scheduling
  - E.g., to reserve a particular class of a course to a student group, while avoiding time conflicts within the group
UniTime Mobile

• Useful for pages that are accessible by students and instructors
  • Online student scheduling (Scheduling Assistant)
  • Event management
• We have some support since UniTime 3.5 (using MGWT)
  • There can be a different permutation of the client code for each platform (desktop, tablet, phone)
• Would like to revisit the current approach and, e.g., use the responsive design instead
UniTime Solver Evolution

It is not just the user interface that keeps evolving

• A lot of changes has been done in the solver engine as well
• To demonstrate, we have taken 2007 benchmark data from Purdue, and run them through the various solver builds since the paper (March 2008) till the one released with UniTime 4.1 (Dec 2015)
• There was 50% improvement in the solution quality since UniTime 3.1
  • 33% less student conflicts
  • 15% improvement in time preferences
  • 40% in room preferences
  • 80% in distribution preferences
• Besides of these, there have been a lot of new constraints and other features added into the solver over the years.

For Purdue C8-2007 benchmark data sets see http://www.unitime.org/uct_datasets.php
See the PATAT 2016 research paper for more details (to be published).
Conclusion

Long Term

• Constraint Solver: instructor and student group scheduling
• UI: moving from Struts to GWT, localization, documentation, mobile
• Interfaces: IMS Course Planning & Scheduling, more APIs and XMLs

For more details, please see us at the conference

• UniTime 101 (Sunday, 9 am - 12 pm in GC 261)
• UniTime: State of the Project (Tuesday, 3pm - 3:45pm in KC 912)
• Student Scheduling in UniTime (Wednesday, 11:45am - 12:30pm in KC 912)
• Or visit www.unitime.org

An online demo is available at https://demo.unitime.org