

Open Review Conference Management System

Ian Swire, Charles Taaffe

Faculty Advisor(s): Dr. Marius Silaghi, Dept. of Computer Engineering and Sciences, Florida Institute of Technology

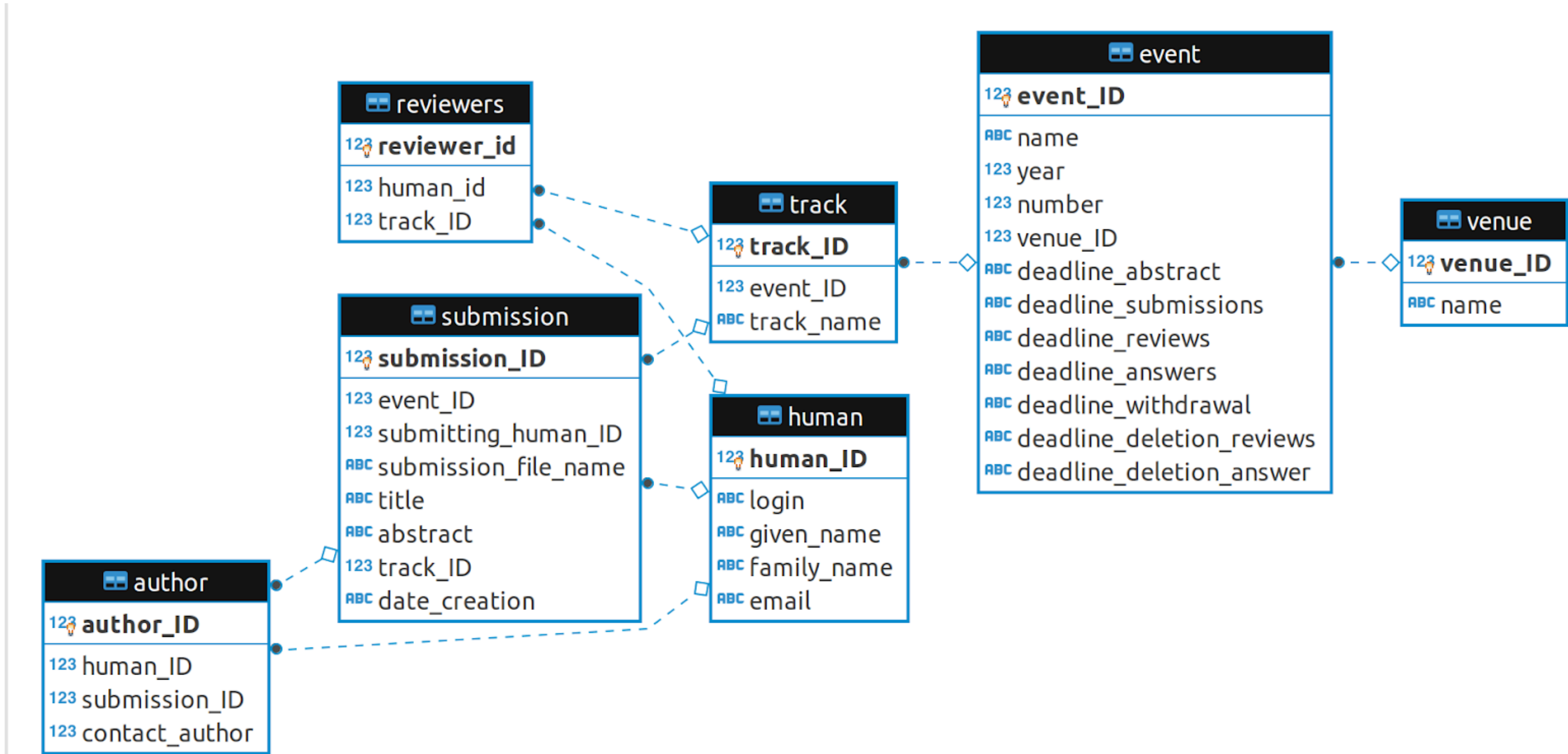


Problem statement:

Currently most if not all conferences are managed by one person or a group of people who work on assigning papers, distributing them, and creating a conference schedule after certain stages of the conference have been completed. This is a lot for people to do without the aid of automation, so we aimed to automate this process for them and make it more efficient. Allowing conference managers to spend less time on the management of the conference and more time on the amenities of the conference.

Accomplishments:

- Sped-up the process to take a matter of minutes instead of days/weeks for paper distribution as well as schedule creation
- Assigned papers with 100% success to reviewers, where success was measured by the number of papers in the reviewer's track/total number of papers per reviewer
- Organized a schedule following typical conference guidelines for sessions per day, and the total number of day s in the conference. Maximizing the breakout session time to study papers



Approach:

- We took Dr. Silaghi’s database schema that he provided us with and then modified it so that it would meet our needs as seen in the ER diagram above.
- Created Track object in code to better organize papers, reviewers, sessions, and the schedule.
- Created a stack of papers based on the number of papers per reviewer and reviewers per paper.
- Distributed said papers from the stack for each track based on their respective bid weight.
- Output the lists of papers per reviewer to a json file that is to be parsed later and create a better visual form for denoting paper distribution to the reviewers
- Schedule creation begins, going through the list of tracks and sorting the papers in that track based upon their review weight
- Sessions are then created with the best paper first, better papers last, and good papers in the middle to match the structure that is typically followed at conferences now.
- Once the schedule has been generated it is output in a json format, as seen in the figure on the left, that will be parsed to create a more visually pleasing representation of the schedule for the viewer to read.

Tools Used:

- Visual Studio/ Visual Studio Code
- DBeaver
- NUGET
- Liveshare by Microsoft
- C#
- SQLite

Day ID	Track	Session ID	Position ID	Speaker Name	Paper Name
0	Test Track 1	0	0	Test Human	Test Paper 9
0	Test Track 1	0	1	Test Human	Test Paper 1
0	Test Track 1	1	0	Test Human	Test Paper 13
0	Test Track 1	1	1	Test Human	Test Paper 5
0	Test Track 2	0	0	Test Human	Test Paper 10
0	Test Track 2	0	1	Test Human	Test Paper 2
0	Test Track 2	1	0	Test Human	Test Paper 14
0	Test Track 2	1	1	Test Human	Test Paper 6
0	Test Track 3	0	0	Test Human	Test Paper 11
0	Test Track 3	0	1	Test Human	Test Paper 3
0	Test Track 3	1	0	Test Human	Test Paper 15
0	Test Track 3	1	1	Test Human	Test Paper 7
0	Test Track 4	0	0	Test Human	Test Paper 12
0	Test Track 4	0	1	Test Human	Test Paper 4
0	Test Track 4	1	0	Test Human	Test Paper 16
0	Test Track 4	1	1	Test Human	Test Paper 8