Ryan Dorrity and Sara Kazemi
CST 363 Spring 2019
Project 1

Staff Directory System

1.0 Part 1: OLTP Design

1.1 Application Description

The Staff Directory System is designed to store staff schedule and room information about teaching staff at a public high school. This system allows a user to answer questions such as:

- Which teachers are in a particular department?
- Which teachers teach a certain class?
- Where is a teacher located during a certain class period?
- During which period does a teacher teach a certain course?
- What teachers have their preparatory period during a certain class period?

1.2 Users

There are different types of users who will interact with the Staff Directory System and each type of user has their own specific needs. School staff (teaching staff and administrators) often have the need to find when their colleagues have a free period (their preparatory period) to schedule meetings or to ask that colleague to cover a course in case of an emergency. Particular teaching staff, such as those who teach Advanced Placement courses, often recruit students from particular courses (e.g. students who are enrolled in AP Calculus AB and/or AP

Calculus BC are often recruited to take AP Computer Science A), so it is important to identify the teaching staff who teach particular courses as well as when and where they teach them. Currently enrolled students need to know where staff are at a given time of day in case they need to turn in work or ask the staff a question -- not every staff member has a permanent classroom. School visitors such as parents, district personnel, and class presenters also benefit from the Staff Directory System to locate staff with whom they are supposed to meet when they misplace or forget the exact room number.

1.3 Database Design

In designing this database, we followed the six basic steps outlined by Murach using the data we had available:

Sweetwater High School
2018-2019 SUH
Master Schedule Matrix

Teacher	Term	Period0	Period1	Period2	Period3	Period4	Period5	Period6	Period7	Period8
Alcock III, Robert	S2		94188-916 ECONOMICS Room: 902 Total: 43 End Date: 06/07/2019	94027B-1 AP US Government-B Room: 902 Total: 25 End Date: 06/07/2019	94188-913 ECONOMICS Room: 902 Total: 43 End Date: 06/07/2019	94027B-2 AP US Government-B Room: 902 Total: 20 End Date: 06/07/2019		94188-910 ECONOMICS Room: 902 Total: 30 End Date: 06/07/2019		
Amezcua, Janet	S2			91140B-11 ENGLISH 11-B Room: 301 Total: 12 End Date: 06/07/2019	91140B-531 ENGLISH 11-B Room: 301 Total: 27 End Date: 06/07/2019	91140B-2 ENGLISH 11-B Room: 301 Total: 43 End Date: 06/07/2019	RESP-4 Resource Period Room: Total: 0 End Date: 06/07/2019	91140B-801 ENGLISH 11-B Room: 301 Total: 39 End Date: 06/07/2019		
Anderson, Cynthia	S2		94076B-3 AP United States History-B Room: 404 Total: 35 End Date: 06/07/2019		95034B-2 CROSS-AGE TCH-B Room: 404 Total: 29 End Date: 06/07/2019	94076B-2 AP United States History-B Room: 404 Total: 26 End Date: 06/07/2019	94076B-5 AP United States History-B Room: 404 Total: 31 End Date: 06/07/2019	94076B-6 AP United States History-B Room: 404 Total: 28 End Date: 06/07/2019		
Apodaca, Alsacia	S2		92926B-100 INTEGRATED MATH III-B Room: 210 Total: 35 End Date: 06/07/2019	92926B-1 INTEGRATED MATH III-B Room: 210 Total: 39 End Date: 06/07/2019	92923B-917 INTEGRATED MATH I-B Room: 210 Total: 24 End Date: 06/07/2019	92926B-101 INTEGRATED MATH III-B Room: 210 Total: 36 End Date: 06/07/2019	92923B-913 INTEGRATED MATH I-B Room: 210 Total: 25 End Date: 06/07/2019			
Balsley, Craig	S2		93577B-1 PE HIGH SCHOOL COURSE II-B (10) Room: BPE Total: 45 End Date: 06/07/2019	93576B-1 PE HIGH SCHOOL COURSE I-B (9) Room: BPE Total: 45 End Date: 06/07/2019	93577B-6 PE HIGH SCHOOL COURSE II-B (10) Room: BPE Total: 52 End Date: 06/07/2019	93577B-8 PE HIGH SCHOOL COURSE II-B (10) Room: BPE Total: 52 End Date: 06/07/2019		93577B-13 PE HIGH SCHOOL COURSE II-B (10) Room: BPE Total: 49 End Date: 06/07/2019		
Bena, Alicia	S2		92923B-9 INTEGRATED MATH I-B Room: 153 Total: 24 End Date: 06/07/2019	92923B-912 INTEGRATED MATH I-B Room: 153 Total: 21 End Date: 06/07/2019		92926B-914 INTEGRATED MATH III-B Room: 153 Total: 34 End Date: 06/07/2019	92923B-914 INTEGRATED MATH I-B Room: 153 Total: 25 End Date: 06/07/2019	92926B-10 INTEGRATED MATH III-B Room: 153 Total: 28 End Date: 06/07/2019		

Figure 1-1: Master schedule used to identify data elements

Figure 1-1 shows a selection from the raw data provided by the school's administration. From this data, we identified the data elements relevant to our system: staff name, period number, course name (prep period inferred by a blank), room number, and department (inferred from course name). We ignored extraneous details that were not relevant to our needs (such as course section numbers). Staff name was broken up into smaller elements -- last name and first name.

From these data elements, we identified four main entities which would become our tables: **`staff`**, **`departments`**, and **`courses`**. We use the label "staff" as opposed to "teachers" because this leaves room to expand the directory system to non-teaching staff such as administrators and counselors. Since staff could belong to more than one department and departments could have any number of staff, a linking table (**`department_membership`**) is used to relate them. Similarly, since staff can teach several types of courses and can have more than one classroom, another linking table (**`schedules`**) is used to relate staff to courses.

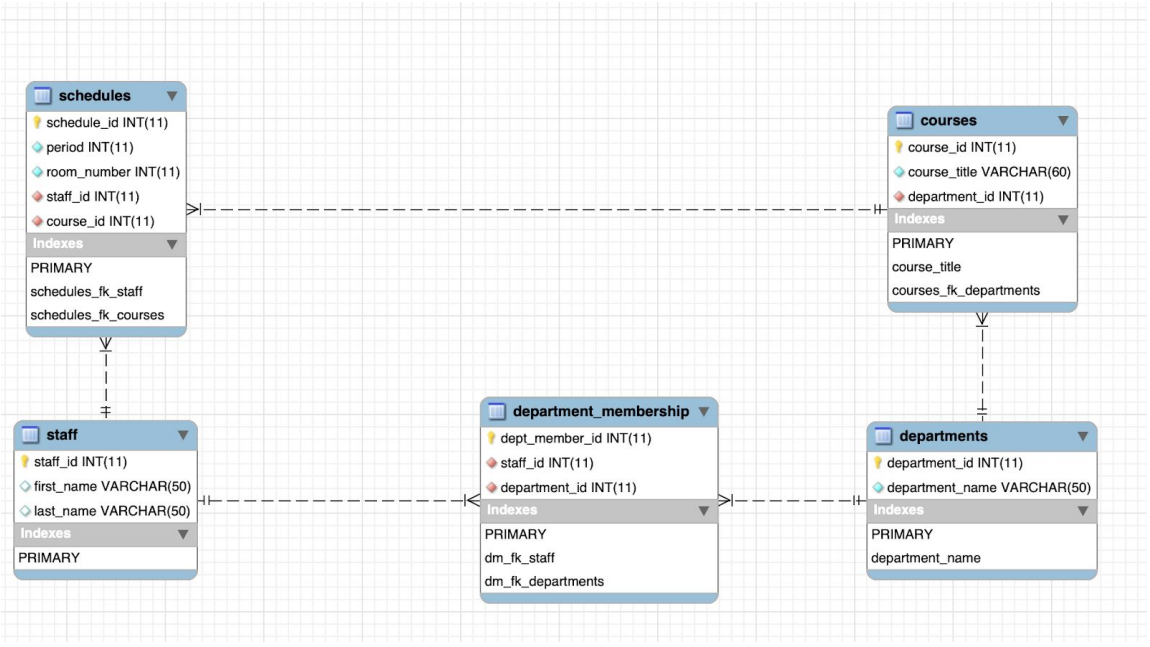


Figure 1-2: ER Diagram of the Staff Directory System

We designed our database in third normal form (3NF), meaning that every column in each table depends on the primary key and only the primary key. This increases data retrieval efficiency as each table represents a single entity, simplifying the number of columns that rely upon one another. Our staff directory (sd) database structure is illustrated in the Entity Relationship (ER) diagram in Figure 1-2.

The **`staff`** table has `staff_id` as an automatically incrementing primary key, which also serves as an index for the table. The **`departments`** table has department_id is an automatically incrementing primary key, which also serves as an index for the table. As a linking table between **`staff`** and **`departments`**, **`department_membership`** contains both `staff_id` and `department_id` as foreign keys. The **`courses`** table has an automatically incrementing primary key (and index) named `course_id` and also has `department_id` as a foreign key. Last, the **`schedules`** table contains an automatically incremented primary key (and index) named `schedule_id`. This table has `staff_id` and `course_id` as foreign keys. Originally, we had planned on creating a separate **`rooms`** table, that would contain a column of all rooms in the entire school. However, because this was the only column that would go into this table, we chose to add this column to the **`schedules`** table instead. For this reason, **`schedules`** may be the only table in our database that may be considered second normal form (2NF).

2.0 OLAP Design

2.1 The Data Warehouse

Our database warehouse uses the following star schema to represent the dimensional attributes of the course master schedule at Sweetwater High.

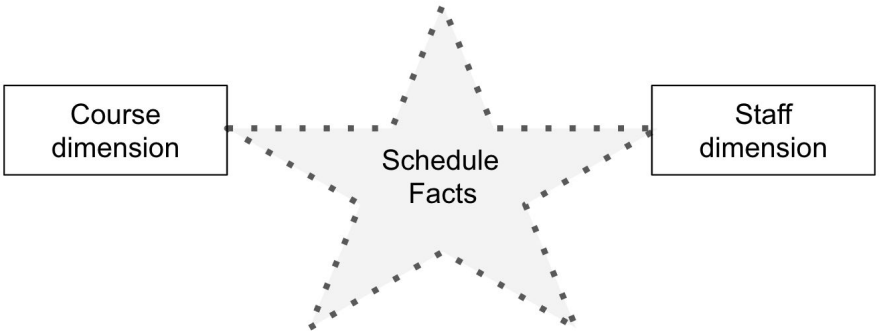


Figure 2-1:Star schema representing the dimensional attributes out Sweetwater High's master schedule

In the early stages of design, we considered creating two additional dimensional tables for `room` and `period`, as represented in Figure 2-2 by the location and time dimensions..

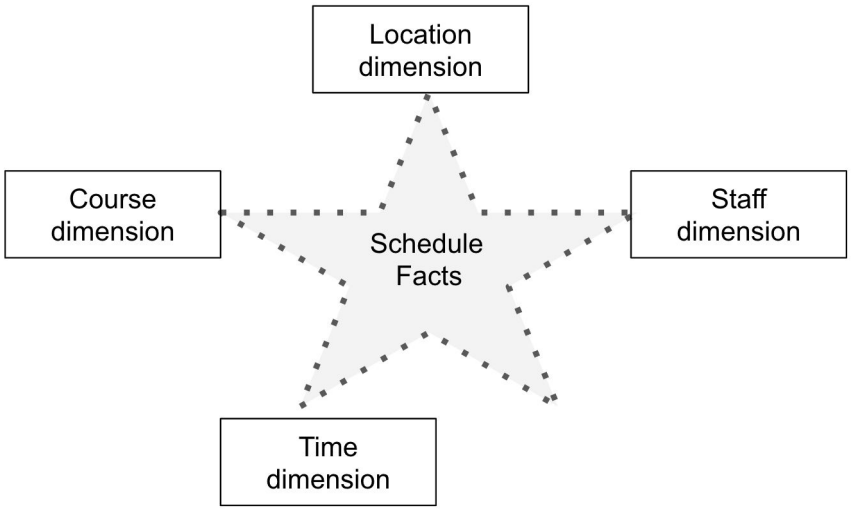
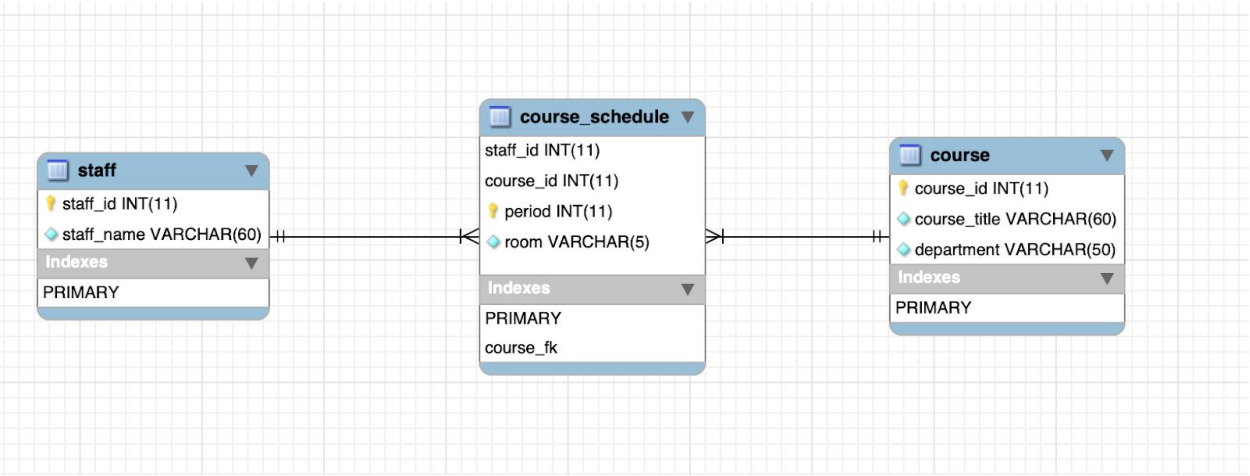


Figure 2-2 Original proposed star schema, in contrast with our final design in 2-1.

However, with the current OLTP design, the location and time dimensions would have only had one column each. In the future, if we were to, for example, include more information about a class' location such as school site or building, this would justify the creation of an additional location dimension table. Likewise, if further information about time were provided, such as term or period start and period end, this would justify and additional time dimension table. However, since the only location information we currently store in our OLTP database is the `room_number`, and the only time information we store is the `period`, we have included these two attributes as columns in the fact table.

Figure 2-3 contains an E-R diagram of our data warehouse, which follows the design of the star schema in Figure 2-1.



At the center is our fact table, **`course_schedule`**, which represents all the courses currently being scheduled throughout the day at Sweetwater High. It has a composite primary key made up of `staff_id`, `course_id`, and `period`. Both `staff_id` and `course_id` are foreign keys from the **`staff`** and **`course`** dimension tables. To the left, is the **`staff`** dimension table, which contains information about the staff teaching the courses (`staff_id` and `staff_name`). To the right, is the **`course`** dimension table, which contains information about the each course (`course_id`, `title`, and `department`).

2.1 The ETL System

Data is extracted from the **sd** database to fill the tables in the database warehouse. The course_schedules fact table is populated with data from a `SELECT` query to the **sd.schedules** table. The **staff** dimension table is populated with data from a `SELECT` query to the **sd.staff** table. Since staff names are broken up into last and first names in the **sd.staff** table, it is necessary to concatenate the values in these two columns and store the result in the staff_name column of the **staff** dimension table. We use the format `CONCAT(last_name, ', ', first_name`. Finally, the course dimension table is populated with data from a `SELECT` query to the **sd.courses** table that joins on the **sd.department** table in order to include the department name that a course is associated with.

2.2 Typical Queries

We previously identified common questions users of this system (students, staff, and school visitors) ask in designing the queries to our OLAP system:

- Which teachers are in a particular department?
- Which teachers teach a certain class?
- Where is a teacher located during a certain class period?
- During which period does a teacher teach a certain course?
- What teachers have their preparatory period during a certain class period?

We refined these to the following queries (an asterisk indicates that views were created from that query):

- Which teachers are in the Mathematics department?
- Which teachers teach Integrated Math III?

- Where is Ms. Kazemi during Period 4?
- Which period(s) is AP Computer Science A taught and where?
- Which teachers have their preparatory periods during period 5?
- How many preps (distinct courses) does each teacher have?
- How many preps on average do Sweetwater teachers have?

2.3 Final Comments

Because no critiques were offered in the assessment of part one of our project, we did not refine the design of the original OLTP database. However, we did add some rows of data to create more interesting results. In the design of the data warehouse using OLAP principles, it became obvious how other types of data from other databases (e.g. student enrollment and student grades) could be integrated into the data warehouse to facilitate different types of queries that could be used by administration in making important master schedule decisions. For example, with an additional database about student enrollment and grades, we could create, for example, a query that analyzes students' average grades in a particular class based on the number of preps a teacher has, possibly finding a link between student success and how much time a teacher has to focus on the planning and assessment stages for each of his/her preps. We look forward to integrating these features in the future to build a more robust DW/BI system.