



Database Concepts

BMI 773: Clinical Research Informatics

Yuriy Sverchkov

University of Wisconsin–Madison

February 10, 2020



Transactional databases vs data warehouses



	Transactional database	Data warehouse
Purpose	Support day-to-day operations	Collect and aggregate data for analysis
Update frequency	In real-time	Periodically (a few times per day or less)
Optimized for	Fast updates, fast look-up of individual patients	Analytics, searches of collections
Epic e.g.	Chronicles	Clarity and Caboodle



Table Relation Relvar

first_name	last_name	gender	birth_date
Lauren	Johnson	F	1982-03-12
Peter	Jurasik	M	1950-04-25
Richard	Biggs	M	1960-03-18
Lauren	Johnson	F	1974-06-01



Table
Relation
Relvar

Column
Attribute
Field

first_name last_name gender birth_date

Lauren	Johnson	F	1982-03-12
Peter	Jurasik	M	1950-04-25
Richard	Biggs	M	1960-03-18
Lauren	Johnson	F	1974-06-01



Table
Relation
Relvar

Column
Attribute
Field

	first_name	last_name	gender	birth_date
Row Tuple Record	Lauren	Johnson	F	1982-03-12
Peter	Jurasik	M	1950-04-25	
Richard	Biggs	M	1960-03-18	
Lauren	Johnson	F	1974-06-01	

Why have more than one table?

First Name	Last Name	Birth date
Lauren	Johnson	1982-03-12
Alice	Smith	1979-12-04
James	White	1964-09-01

- ▶ Consider a database of patients

Why have more than one table?

First Name	Last Name	Birth date	Drug	Prescription date
Lauren	Johnson	1982-03-12	Acetaminophen	2010-01-06
Alice	Smith	1979-12-04	Amoxicillin	1998-01-29
James	White	1964-09-01	Albuterol	1990-12-15

- ▶ Consider a database of patients and their prescriptions
- ▶ How can we track multiple prescriptions for a patient?

Why have more than one table?

First Name	Last Name	Birth date	Drug	Prescription date	Drug 2	Prescription date 2
Lauren	Johnson	1982-03-12	Acetaminophen	2010-01-06	—	—
Alice	Smith	1979-12-04	Amoxicillin	1998-01-29	Lisinopril	2019-05-26
James	White	1964-09-01	Albuterol	1990-12-15	—	—

- ▶ Consider a database of patients and their prescriptions
- ▶ How can we track multiple prescriptions for a patient?
 - ▶ Add columns?

Why have more than one table?

First Name	Last Name	Birth date	Drug	Prescription date
Lauren	Johnson	1982-03-12	Acetaminophen	2010-01-06
Alice	Smith	1979-12-04	Amoxicillin	1998-01-29
Alice	Smith	1979-12-04	Lisinopril	2019-05-26
James	White	1964-09-01	Albuterol	1990-12-15

- ▶ Consider a database of patients and their prescriptions
- ▶ How can we track multiple prescriptions for a patient?
 - ▶ Add columns?
 - ▶ One row per prescription?

Patients

Patient ID	First Name	Last Name	Birth date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01

Prescriptions

Prescription ID	Patient ID	Drug	Prescription date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26



Patients

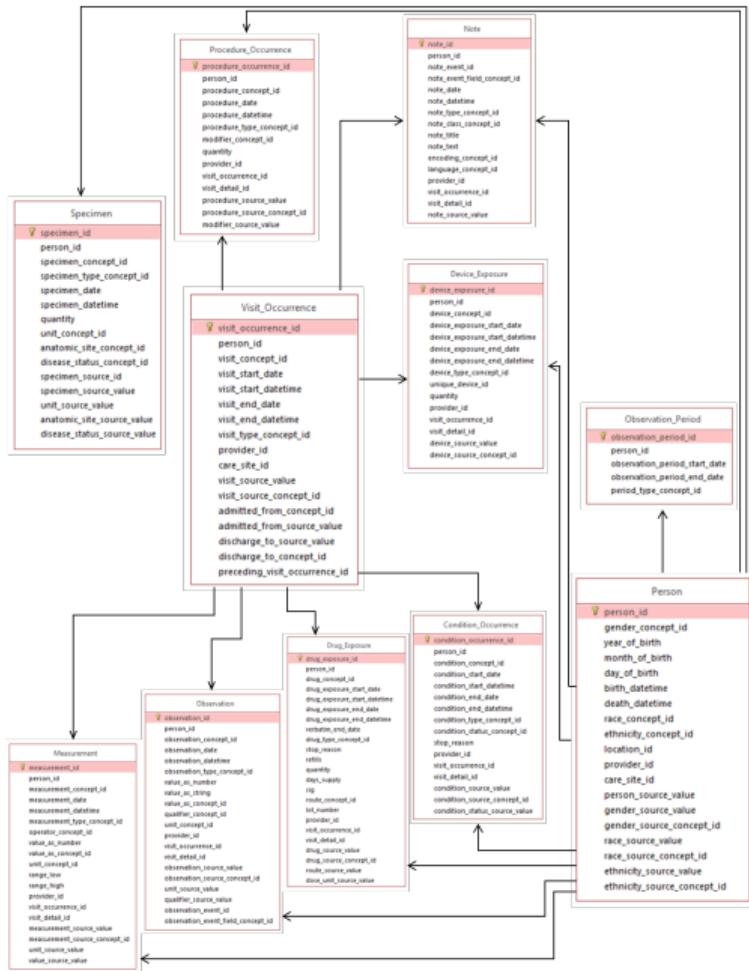
Patient ID	First Name	Last Name	Birth date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01

Prescriptions

Prescription ID	Patient ID	Drug	Prescription date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26

Condition occurrences

Cond. Occ. ID	Patient ID	Condition	Date
1	3	Asthma	1990-11-30
2	2	Bacterial ear infection	1998-01-29
3	1	Dysmenorrhea	2010-01-06
4	2	Hypertension	2018-11-01



- ▶ A **database schema** describes the structure of a relational database
 - ▶ Tables
 - ▶ Columns
 - ▶ Primary and foreign keys
- ▶ There are a few formal standards for graphically representing a schema graphically: object-role models (ORM) and multiple standards for Entity-relationship (ER) diagrams.
- ▶ Example: OHDSI Common Data Model Clinical Tables Schema

Structured Query Language (SQL)



- ▶ A language for managing data in relational database management systems (RDBMS)

Structured Query Language (SQL)



- ▶ A language for managing data in relational database management systems (RDBMS)
- ▶ There are differences in dialects and implementation between different RDBMS providers

Structured Query Language (SQL)

- ▶ A language for managing data in relational database management systems (RDBMS)
- ▶ There are differences in dialects and implementation between different RDBMS providers
- ▶ SQL has functionality for data query, data manipulation (adding/deleting/modifying records), data definition (defining a schema) and data access control (managing user access to the database).

Structured Query Language (SQL)

- ▶ A language for managing data in relational database management systems (RDBMS)
- ▶ There are differences in dialects and implementation between different RDBMS providers
- ▶ SQL has functionality for data query, data manipulation (adding/deleting/modifying records), data definition (defining a schema) and data access control (managing user access to the database).
- ▶ We will focus on data query

SQL SELECT

```
SELECT * FROM Patients;
```

SQL SELECT

```
SELECT * FROM Patients;
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

SQL SELECT

```
SELECT * FROM Patients;
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

```
SELECT first_name FROM Patients;
```

SQL SELECT

```
SELECT * FROM Patients;
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

```
SELECT first_name FROM Patients;
```

first_name
Lauren
Alice
James
Lauren

SQL SELECT

```
SELECT * FROM Patients;
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

```
SELECT DISTINCT first_name  
FROM Patients;
```

```
SELECT first_name FROM Patients;
```

first_name
Lauren
Alice
James
Lauren

SQL SELECT

```
SELECT * FROM Patients;
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

```
SELECT DISTINCT first_name  
FROM Patients;
```

first_name
Lauren
Alice
James

```
SELECT first_name FROM Patients;
```

first_name
Lauren
Alice
James
Lauren

SQL SELECT

```
SELECT * FROM Patients;
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

```
SELECT DISTINCT first_name  
FROM Patients;
```

first_name
Lauren
Alice
James

```
SELECT first_name FROM Patients;
```

first_name
Lauren
Alice
James
Lauren

```
SELECT first_name, last_name  
FROM Patients  
WHERE birth_date > '1975';
```

SQL SELECT

```
SELECT * FROM Patients;
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

```
SELECT DISTINCT first_name  
FROM Patients;
```

first_name
Lauren
Alice
James

```
SELECT first_name FROM Patients;
```

first_name
Lauren
Alice
James
Lauren

```
SELECT first_name, last_name  
FROM Patients  
WHERE birth_date > '1975';
```

first_name	last_name
Lauren	Johnson
Alice	Smith

SQL SELECT ... WHERE ... AND ... OR ...

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

```
SELECT * FROM Patients
  WHERE birth_date > '1975'
    AND last_name == 'Johnson';
```

SQL SELECT ... WHERE ... AND ... OR ...

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

```
SELECT * FROM Patients
WHERE birth_date > '1975'
AND last_name == 'Johnson';
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12

```
SELECT * FROM Patients
WHERE birth_date > '1975'
OR last_name = 'Johnson';
```

SQL SELECT ... WHERE ... AND ... OR ...

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

```
SELECT * FROM Patients
WHERE birth_date > '1975'
AND last_name == 'Johnson';
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12

```
SELECT * FROM Patients
WHERE birth_date > '1975'
OR last_name = 'Johnson';
```

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
4	Lauren	Johnson	1974-05-27

SQL JOIN



Patients

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

Prescriptions

prescription_id	patient_id	drug	date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26
5	5	Warfarin	1961-08-01

SQL JOIN

Patients

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

Prescriptions

prescription_id	patient_id	drug	date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26
5	5	Warfarin	1961-08-01

```
SELECT Patients.first_name, Patients.last_name, Prescriptions.drug
FROM Patients INNER JOIN Prescriptions
ON Patients.patient_id = Prescriptions.patient_id;
```

SQL JOIN



Patients

patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

Prescriptions

prescription_id	patient_id	drug	date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26
5	5	Warfarin	1961-08-01

```
SELECT Patients.first_name, Patients.last_name, Prescriptions.drug
FROM Patients INNER JOIN Prescriptions
ON Patients.patient_id = Prescriptions.patient_id;
```

first_name	last_name	drug
Lauren	Johnson	Acetaminophen
Alice	Smith	Amoxicillin
Alice	Smith	Lisinopril
James	White	Albuterol

SQL INNER/LEFT/RIGHT/FULL JOIN



patient_id	first_name	last_name	birth_date
1	Lauren	Johnson	1982-03-12
2	Alice	Smith	1979-12-04
3	James	White	1964-09-01
4	Lauren	Johnson	1974-05-27

prescription_id	patient_id	drug	date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26
5	5	Warfarin	1961-08-01

INNER JOIN

first_name	last_name	drug
Lauren	Johnson	Acetaminophen
Alice	Smith	Amoxicillin
Alice	Smith	Lisinopril
James	White	Albuterol

LEFT JOIN

first_name	last_name	drug
Lauren	Johnson	Acetaminophen
Alice	Smith	Amoxicillin
Alice	Smith	Lisinopril
James	White	Albuterol
Lauren	Johnson	—

OUTER JOIN

first_name	last_name	drug
Lauren	Johnson	Acetaminophen
Alice	Smith	Amoxicillin
Alice	Smith	Lisinopril
James	White	Albuterol
Lauren	Johnson	—
—	—	Warfarin

RIGHT JOIN

first_name	last_name	drug
Lauren	Johnson	Acetaminophen
Alice	Smith	Amoxicillin
Alice	Smith	Lisinopril
James	White	Albuterol
—	—	Warfarin

SQL COUNT and GROUP BY

prescription_id	patient_id	drug	date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26
5	5	Warfarin	1961-08-01

```
SELECT COUNT(prescription_id)  
FROM Prescriptions;
```

SQL COUNT and GROUP BY

prescription_id	patient_id	drug	date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26
5	5	Warfarin	1961-08-01

```
SELECT COUNT(prescription_id)  
FROM Prescriptions;
```

```
COUNT(prescription_id)  
5
```

SQL COUNT and GROUP BY

prescription_id	patient_id	drug	date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26
5	5	Warfarin	1961-08-01

```
SELECT COUNT(prescription_id)  
FROM Prescriptions;
```

```
COUNT(prescription_id)  
5
```

```
SELECT patient_id, COUNT(prescription_id)  
FROM Prescriptions  
GROUP BY patient_id;
```

SQL COUNT and GROUP BY

prescription_id	patient_id	drug	date
1	3	Albuterol	1990-12-15
2	2	Amoxicillin	1998-01-29
3	1	Acetaminophen	2010-01-06
4	2	Lisinopril	2019-05-26
5	5	Warfarin	1961-08-01

```
SELECT COUNT(prescription_id)  
FROM Prescriptions;
```

```
COUNT(prescription_id)  
5
```

```
SELECT patient_id, COUNT(prescription_id)  
FROM Prescriptions  
GROUP BY patient_id;
```

patient_id	COUNT(prescription_id)
1	1
2	2
3	1
5	1