

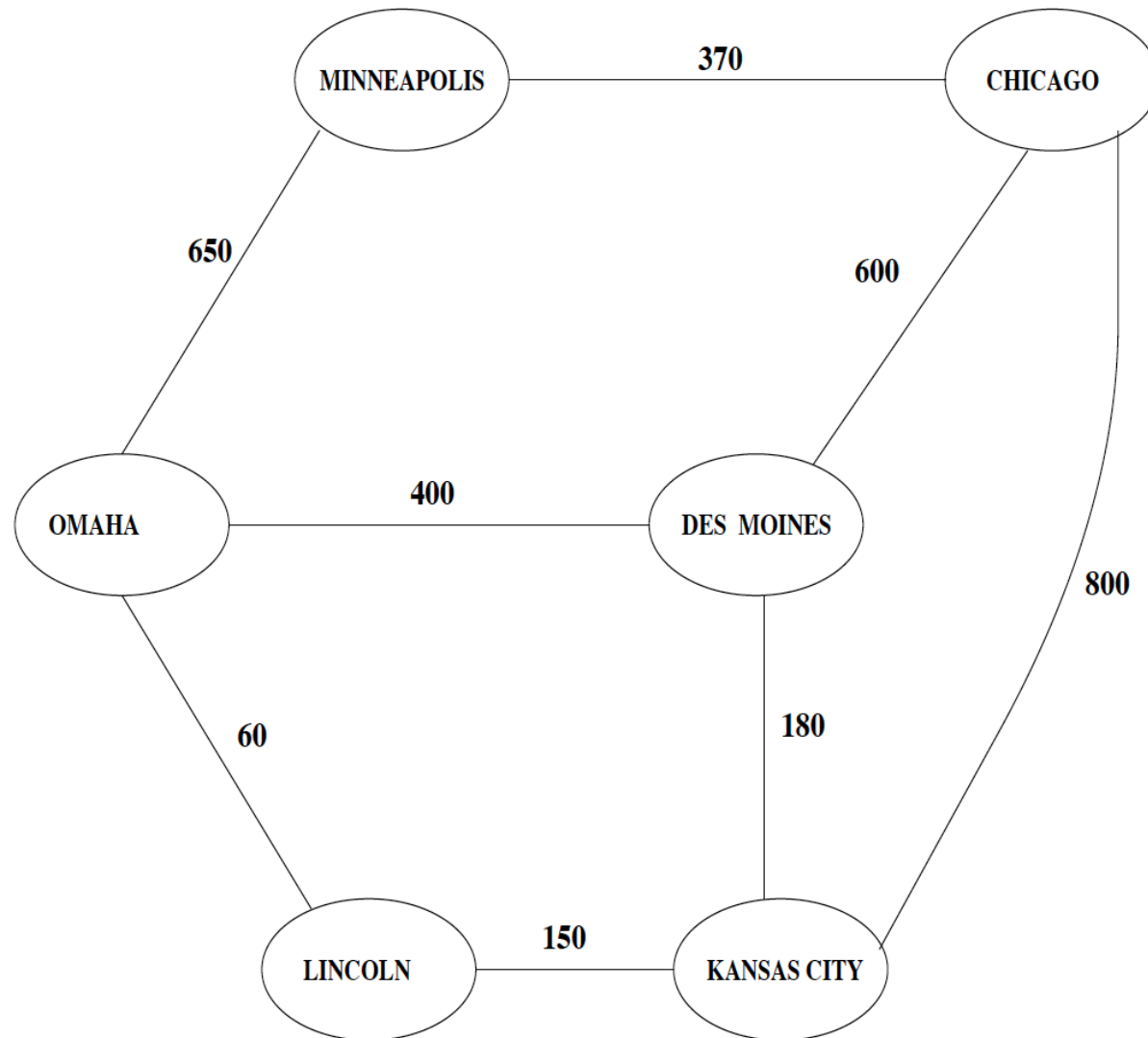
# Topological Data Models (Section 6.5)

**Peter Revesz**

**CSCE 413/813**

**Computer Science and Engineering  
University of Nebraska – Lincoln**

The following are the shortest distances in miles between some Midwestern cities.



# Relational Database Representation

**Edge**

City1	City2	Miles
Chicago	Des_Moines	600
Chicago	Kansas_City	800
Chicago	Minneapolis	370
Des_Moines	Chicago	600
Des_Moines	Kansas_City	180
Des_Moines	Omaha	400
Kansas_City	Chicago	800
Kansas_City	Des_Moines	180
Kansas_City	Lincoln	150
Lincoln	Kansas_City	150
Lincoln	Omaha	60
Minneapolis	Chicago	370
Minneapolis	Omaha	650
Omaha	Des_Moines	400
Omaha	Minneapolis	650
Omaha	Lincoln	60

# Relational Database Representation

**Edge**

City1	City2	Miles
Chicago	Des_Moines	600
Chicago	Kansas_City	800
Chicago	Minneapolis	370
Des_Moines	Chicago	600
Des_Moines	Kansas_City	180
Des_Moines	Omaha	400
Kansas_City	Chicago	800
Kansas_City	Des_Moines	180
Kansas_City	Lincoln	150
Lincoln	Kansas_City	150
Lincoln	Omaha	60
Minneapolis	Chicago	370
Minneapolis	Omaha	650
Omaha	Des_Moines	400
Omaha	Minneapolis	650
Omaha	Lincoln	60

## Problem:

We do not always take the shortest path. There can be detours. Therefore, we need to represent that the value is the shortest distance but greater distances may be possible.

# Relational Database Representation

**Edge**

City1	City2	Miles
Chicago	Des_Moines	600
Chicago	Kansas_City	800
Chicago	Minneapolis	370
Des_Moines	Chicago	600
Des_Moines	Kansas_City	180
Des_Moines	Omaha	400
Kansas_City	Chicago	800
Kansas_City	Des_Moines	180
Kansas_City	Lincoln	150
Lincoln	Kansas_City	150
Lincoln	Omaha	60
Minneapolis	Chicago	370
Minneapolis	Omaha	650
Omaha	Des_Moines	400
Omaha	Minneapolis	650
Omaha	Lincoln	60

## Problem:

We do not always take the shortest path. There can be detours. Therefore, we need to represent that the value is the shortest distance but greater distances may be possible.

# Constraint Database Representation

## Travel

City1	Miles1	City2	Miles2	
Chicago	$d_1$	Des_Moines	$d_2$	$d_2 - d_1 \geq 600$
Chicago	$d_1$	Kansas_City	$d_2$	$d_2 - d_1 \geq 800$
Chicago	$d_1$	Minneapolis	$d_2$	$d_2 - d_1 \geq 370$
Des_Moines	$d_1$	Chicago	$d_2$	$d_2 - d_1 \geq 600$
Des_Moines	$d_1$	Kansas_City	$d_2$	$d_2 - d_1 \geq 180$
Des_Moines	$d_1$	Omaha	$d_2$	$d_2 - d_1 \geq 400$
Kansas_City	$d_1$	Chicago	$d_2$	$d_2 - d_1 \geq 800$
Kansas_City	$d_1$	Des_Moines	$d_2$	$d_2 - d_1 \geq 180$
Kansas_City	$d_1$	Lincoln	$d_2$	$d_2 - d_1 \geq 150$
Lincoln	$d_1$	Kansas_City	$d_2$	$d_2 - d_1 \geq 150$
Lincoln	$d_1$	Omaha	$d_2$	$d_2 - d_1 \geq 60$
Minneapolis	$d_1$	Chicago	$d_2$	$d_2 - d_1 \geq 370$
Minneapolis	$d_1$	Omaha	$d_2$	$d_2 - d_1 \geq 650$
Omaha	$d_1$	Des_Moines	$d_2$	$d_2 - d_1 \geq 400$
Omaha	$d_1$	Minneapolis	$d_2$	$d_2 - d_1 \geq 650$
Omaha	$d_1$	Lincoln	$d_2$	$d_2 - d_1 \geq 60$

If we start in City1 with odometer  $d_1$ , then we can arrive at City2 with odometer  $d_2$ .

# MLPQ Representation with city names encoded by id numbers

MLPQ/PReSTO - University of Nebraska-Lincoln - edge

File Edit View Window Help

SQL Dlog Apx [Icons] Max Min B X S [Icons]

[Icons]

edge

travel (16 rows)

Property

city1	d1	city2	d2	CONSTRAI
1	d1	2	d2	d1-d2<=-
1	d1	3	d2	d1-d2<=-
1	d1	5	d2	d1-d2<=-
2	d1	1	d2	d1-d2<=-
2	d1	3	d2	d1-d2<=-
2	d1	6	d2	d1-d2<=-
3	d1	1	d2	d1-d2<=-
3	d1	2	d2	d1-d2<=-
3	d1	4	d2	d1-d2<=-
4	d1	3	d2	d1-d2<=-
4	d1	6	d2	d1-d2<=-
5	d1	1	d2	d1-d2<=-
5	d1	6	d2	d1-d2<=-

Exit Export

The id encoding may be recorded in a separate relation.

MLPQ/PReSTO - University of Nebraska-Lincoln - edge

File Edit View Window Help

SQL Dlog Apx  $\cap$   $\cup$   $-$   $\complement$   $\bowtie$   $\odot$  Max Min B X S  $\backslash$   $\square$   $\curvearrowright$

Del A

edge

travel (16 rows)  
city (6 rows)

**Property**

city

id	name	CONSTRAINTS
1	"Chicago"	
2	"Des_Moines"	
3	"Kansas_City"	
4	"Lincoln"	
5	"Minneapolis"	
6	"Omaha"	

Exit Export



Find the length of the shortest paths from  
Lincoln = 4 using the **Travel** Relation.

MLPQ/PReSTO - University of Nebraska-Lincoln - edge

File Edit View Window Help

SQL Dlog Apx [Icons] Max Min B X S [Icons]

[Icons]

edge

travel (16 rows)  
possible (10 rows)

Property

city	miles	CONSTRAINTS
4	0	
3	miles	-miles<=-150,
6	miles	-miles<=-60,
1	miles	-miles<=-950,
2	miles	-miles<=-330,
4	miles	-miles<=-300,
2	miles	-miles<=-460,
5	miles	-miles<=-710,
4	miles	-miles<=-120,
1	miles	-miles<=-930,

Exit Export

Find the length of the shortest paths from  
Lincoln = 4 using the **Travel** Relation.  
(1<sup>st</sup> step: calculate all possible distances)

MLPQ/PReSTO - University of Nebraska-Lincoln - edge

File Edit View Window Help

SQL Dlog Apx  $\cap$   $\cup$   $-$  C  $\square$   $\odot$  Max Min B X S  $\backslash$   $\square$   $\curvearrowright$

$\square$   $\square$   $\square$  Del  $\otimes$   $\oplus$   $\text{Set}$   $\square$  A  $\triangleright$   $\triangleleft$   $\triangleright\triangleright$   $\triangleleft\triangleleft$   $\triangleright\triangleleft$   $\triangleleft\triangleleft$   $\triangleright\triangleright$   $\triangleleft_R$   $\triangleright_R$   $\triangleleft_R$   $\triangleright_R$   $\triangleleft_R$   $\triangleright_R$

edge

travel (16 rows)



-800 -600

SQL Recursion

RECURSIVE View possible(city,miles)

AS (basic rule)

SELECT city, miles

FROM travel

WHERE city = 4, miles = 0

UNION (recursive rule)

SELECT s.city2, s.d2

FROM possible AS p, travel AS s

WHERE p.city = s.city1, p.miles = s.d1

$\triangleleft\triangleleft$   $\triangleleft$  0  $\triangleright$   $\triangleright\triangleright$  Add Save Del

OK Cancel Save Load Clear

It is possible to travel to  
Lincoln = 4 in 0 miles.

If it is possible to travel  
to city1 in d1 miles and  
 $\text{travel}(\text{city1}, \text{d1}, \text{city2}, \text{d2})$   
holds, then it is possible  
to travel to city2 in d2  
miles.

Find the length of the shortest paths from  
Lincoln = 4 using the **Travel** Relation.

(2<sup>nd</sup> step: find minimum distance out of all possible distances for each city)

The screenshot shows the MLPQ/PReSTO - University of Nebraska-Lincoln - edge software interface. The main window has a menu bar (File, Edit, View, Window, Help) and two toolbars. The left toolbar contains icons for SQL, Dlog, Apx, and various set operations. The right toolbar contains icons for Max, Min, B, X, S, and various navigation arrows. Below the toolbars, there is a list of tables: 'travel (16 rows)' and 'possible (10 rows)'. An 'SQL AGGREGATION' dialog box is open, showing the following fields:

- VIEW NAME: Distance(city,miles)
- SELECT: city, Min(miles)
- FROM: possible
- WHERE: (empty text area)
- GROUP BY: city
- HAVING: (empty text area)

At the bottom of the dialog box are buttons for OK, Cancel, Save, Load, and Clear.

Lincoln = 4 using the Travel Relation.

[illegible]