

Learning Objectives

- After this segment, students will be able to
 - Describe need for SQL extensions
 - List two ways to extend query languages

Data Models of Spatial Networks

1. Conceptual Model: Entity Relationship Diagrams, Graphs
2. Logical Data Model & Query Languages
 - Abstract Data types
 - Custom Statements in SQL
3. Physical Data Model: Storage-Structures, Algorithms

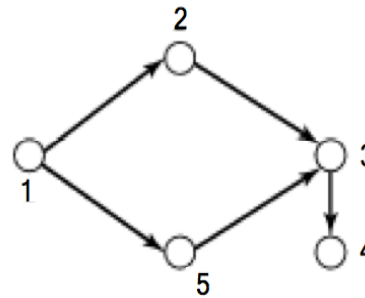


Transitive Closure

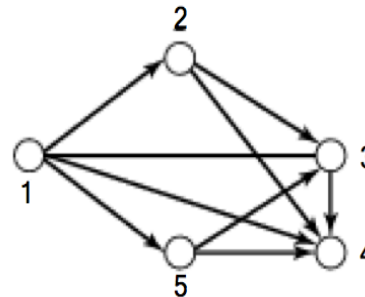
- Consider a graph $G = (V, E)$
- Transitive closure(G) = $G^* = (V^*, E^*)$, where
 - $V^* = V$
 - (A, B) in E^* if and only if there is a path from A to B in G .

Transitive Closure - Example

- Example
 - G has 5 nodes and 5 edges
 - G^* has 5 nodes and 9 edges
 - Note edge $(1,4)$ in G^* for
 - path $(1, 2, 3, 4)$ in G .



(a) Graph G



(c) Transitive closure (G) = Graph G

R

SOURCE	DEST
1	2
1	5
2	3
3	4
5	3

(b) Relation form

X

SOURCE	DEST
1	2
1	5
2	3
3	4
5	3
1	3
2	4
5	4
1	4

(d) Transitive closure in relation form



Limitations of Original SQL

- Recall Relation algebra based languages
 - Ex. Original SQL
 - Can not compute transitive closure, e.g., shortest path

Supporting Graphs in SQL

- Abstract Data Type (user defined)
 - SQL3
 - May include shortest path operation!
- Custom Statements
 - SQL2 - CONNECT clause in SELECT statement
 - For directed acyclic graphs, e.g. hierarchies
 - SQL3 - WITH RECURSIVE statement
 - Transitive closure on general graphs