

# Evaluating Reachability Queries over Path Collections

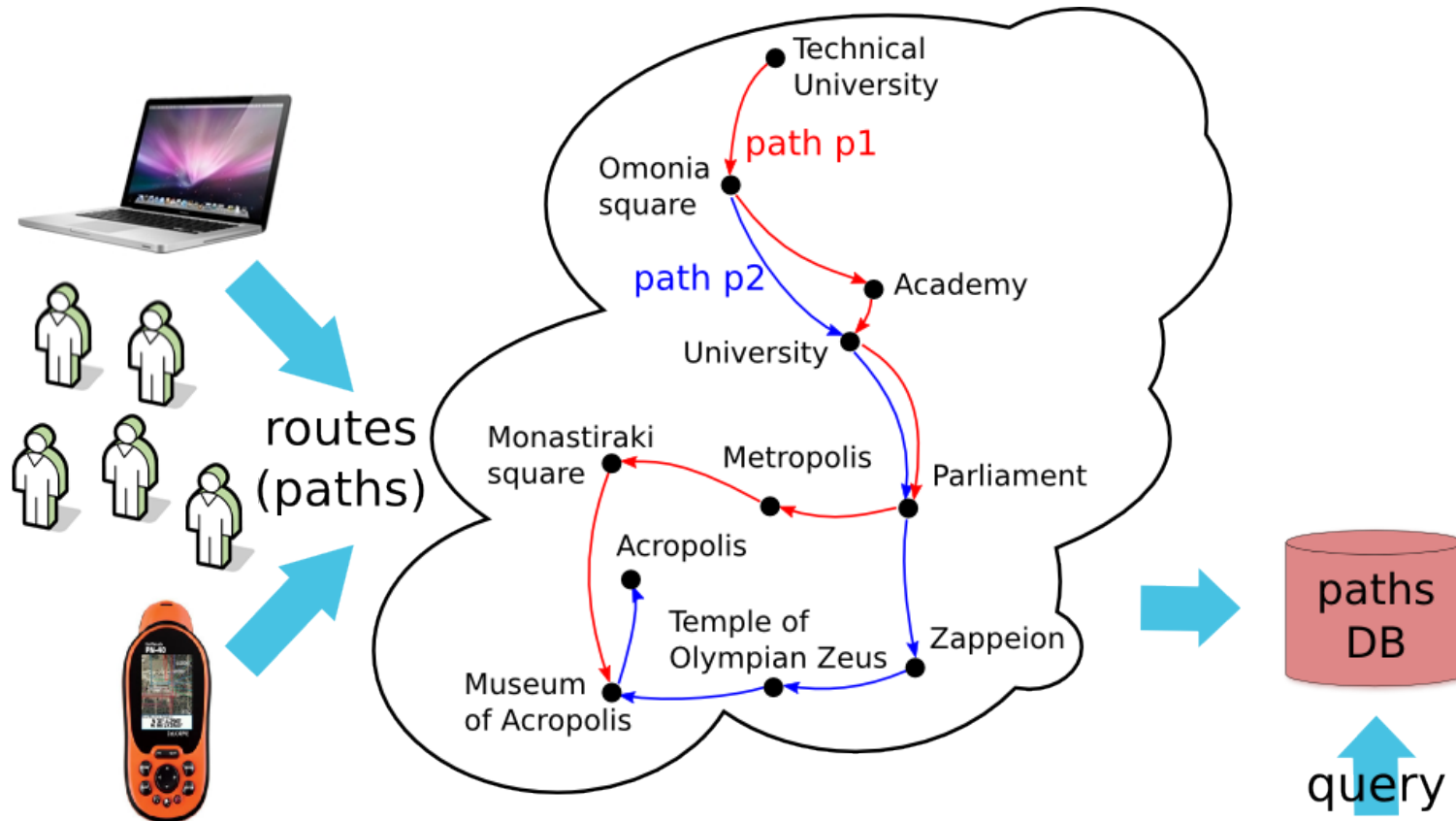
**P. Bouros<sup>1</sup>, S. Skiadopoulos<sup>2</sup>, T. Dalamagas<sup>3</sup>,  
D. Sacharidis<sup>3</sup>, T. Sellis<sup>1,3</sup>**

<sup>1</sup>National Technical University of Athens

<sup>2</sup>University of Peloponnese

<sup>3</sup>Institute for Management of Information Systems – R.C. Athena

# Motivating example



Q: How can I go from Academy to Zappeion?  
A: Academy → University → Parliament → Zappeion

# Problem definition

- Large path collections
- Frequent updates
  - New paths
- Evaluating reachability queries
  1. Is there a path from node  $F$  to  $C$ ?
  2. Find a path from node  $F$  to  $C$

- Path collection

$p_1$  ( $A, B, C, D, J$ )

$p_2$  ( $A, F, D, N, B, T$ )

$p_3$  ( $N, L, M$ )

$p_4$  ( $D, N, B, F, K$ )

$p_5$  ( $A, F, K$ )

# Solving the problem

- Using a graph
  - Merge paths to build underlying graph
  - Apply graph search algorithm for queries
- Our approach
  - Algorithm pfs
    - Visit nodes in paths in depth-first way
    - Push to dfs stack set of nodes at once
  - Index path collections, P-Index, pfsP
  - Index path connections, H-graph & H-Index, pfsH

# Thank you!

