Spatial Search, Ranking, and Interoperability

Greg Janée and James Frew
University of California, Santa Barbara

Background

- ADL: digital library architecture & reference implementation for georeferenced information
  - Distributed
  - Accommodates heteregeneous data, metadata
  - Focus on component definition, interfaces, interoperability
  - Global
  - Lightweight
    - **Accommodate small and/or non-traditional spatial providers, e.g., library catalogs**
- Provides spatial orientation to heterogeneous data sources

Spatial interoperability

- Requirements
  - Represent regions of the Earth's surface
  - Perform spatial predicates on those regions
    - **E.g., query by spatial region**
- Decision points
  - Choice of allowable region shapes
  - Allowable representations
  - Handling of projections
  - Choice of allowable predicates
- Tradeoffs
  - Flexibility vs. uniformity
  - Functionality vs. implementation simplicity

ADL approach

- A few basic shapes
  - Simple (non-intersecting, hole-free) polygons
  - Boxes
  - Polylines
  - Points
- "Spatial fallback"
  - Polygons, polylines always accompanied by bounding boxes
  - Allows simple components to fully participate in the system, albeit with lower precision
- Predicates
  - *INTERSECTS*, *WITHIN*, *CONTAINS*

Other approaches

- FGDC
  - Multiple polygons with multiple holes
- Z39.50 + "GEO" profile
  - Builds on FGDC
  - Includes predicates unsupportable by spatial engines
- OpenGIS Simple Features Specification
  - Arbitrary geometry collections
  - Full DE-9IM suite of predicates: *TOUCHES*, *CROSSES*, etc.
- GML 3.0
  - Many, many shapes, options, etc.

- All more complex than ADL's approach

Surprise!

- Even ADL’s approach has proven to be difficult to implement and insufficiently lightweight

- Three problems
  - Polygons add significant complexity
  - Geodetic continuity is necessary, but difficult to achieve
  - Most predicates are difficult to implement

Problem 1: polygons

- Polygons are desirable, but add significant complexity
  - Ramifications at all levels
  - Predicates require special software
  - Increased interface, representation complexity
  - Increased GUI complexity
    - **E.g., need controls to draw, manipulate polygons**

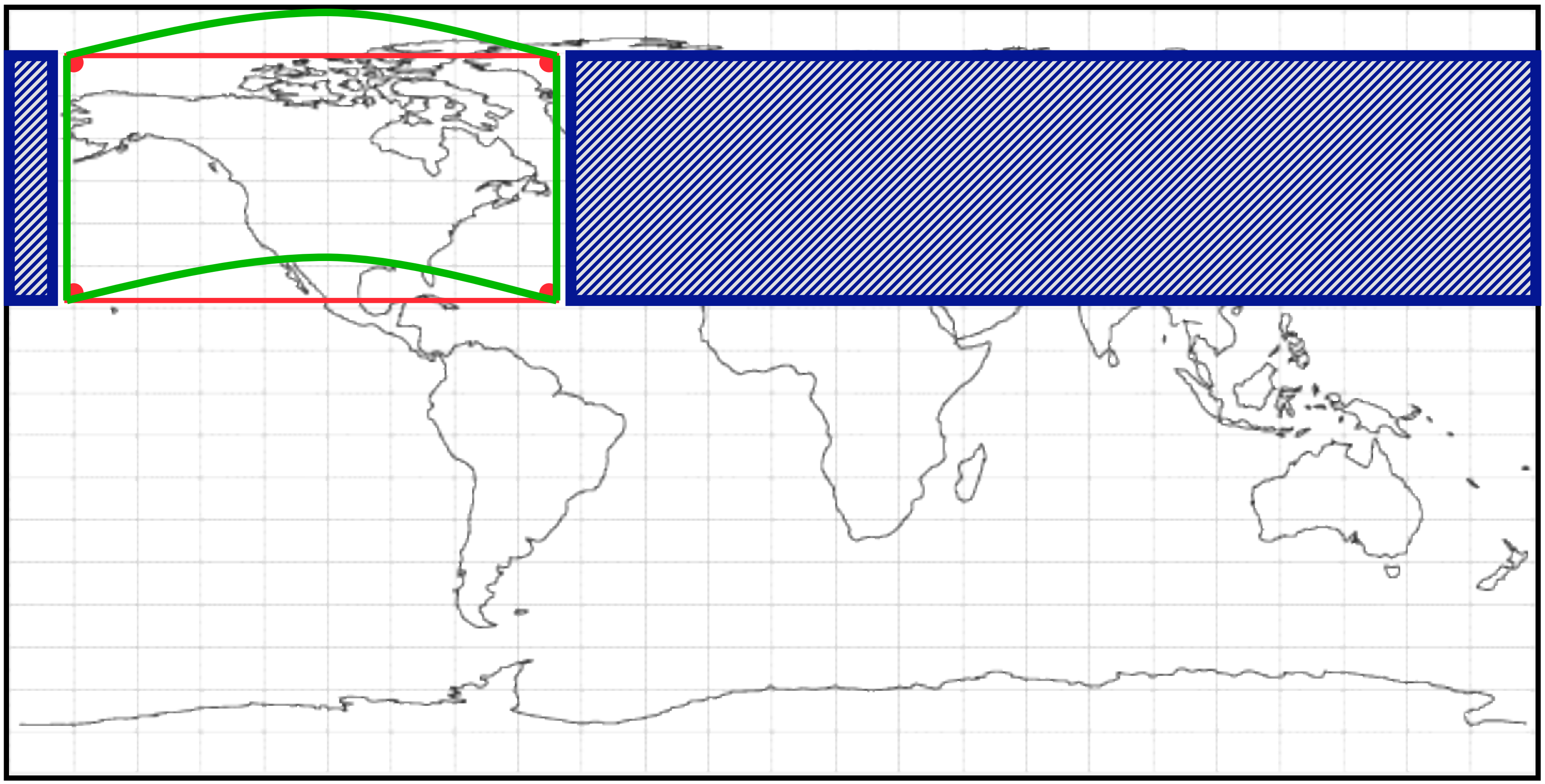
Problem 2: geodetic continuity

- Many data providers, GIS systems model (a portion of) the Earth as a plane
  - Convenient, simple
  - Plentiful algorithms
- Global view requires “geodetic continuity”
  - Recognize and accommodate the fact that the Earth’s surface is a 2-sphere
    - **E.g., polygon interiors must be specified**
  - No discontinuities
    - **E.g., must be possible to describe regions that cross the ±180 meridian**
  - Correct predicate processing
    - **E.g., meridians +179° and -179° are very close**

Rejected solutions

- Spherical regions
  - E.g., spherical polygons with great-circle edges
  - Poorly supported, both theoretically and practically
- Arbitrary, locally-appropriate planar projections
  - Burden on clients to reproject regions
    - **Nontrivial!**
  - Gives rise to geometry collections

Polygon reprojection



Proposed solution

- Geodetic box
  - Defined by N/S/E/W edges
  - By identifying east and west edges, supports geodetic continuity
  - Geodetic bounding boxes easily created from complex geometries in commonly-used cylindrical and polar projections

Problem 3: Predicates

- *INTERSECTS* is easiest; all others are hard
  - Supporting geodetic continuity is non-trivial, but manageable
- *TOUCHES*
  - Sensitive to precision, accuracy, and floating-point roundoff problems
- *CONTAINS*, *WITHIN*
  - Difficult to support over geometry collections

Predicates over collections

- Geometry collections
  - Desired by data providers; arise in reprojections
  - Not supported by PostGIS, MySQL, ...
- Faking *WITHIN*
  - Treat collection components as independent footprints
  - Given primitive:
    - $\mathbf{INTERSECTS}(Q) \equiv \{\ I \mid \exists F \in \mathbf{Footprints}(I) \mid F \cap Q \neq \emptyset\ \}$
  - Can implement *WITHIN* as:
    - $\mathbf{WITHIN}(Q) \equiv \{\ I \mid I \in \mathbf{INTERSECTS}(Q) \wedge$
      $I \notin \mathbf{INTERSECTS}(Q^C)\ \}$
- Can't fake *CONTAINS*

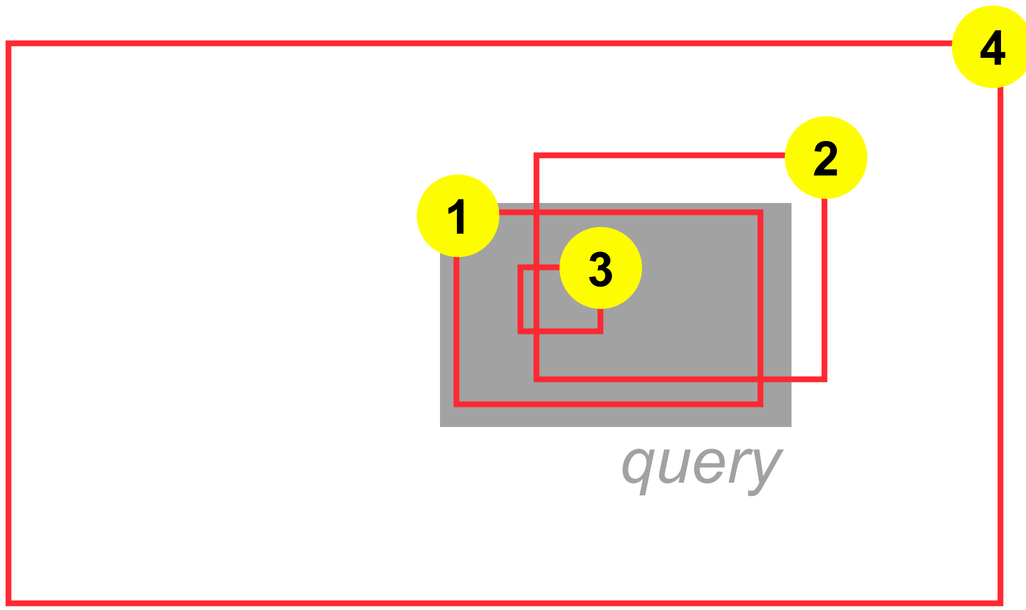
Conclusion

- To achieve spatial interoperability that is global and lightweight:
  - Region shapes must be limited to only the simplest, discontinuity-free shape: geodetic boxes
    - **Easy to operate on**
    - **Easy to create from complex geometries in commonly-used cylindrical and polar projections**
  - Predicates must be limited to *INTERSECTS*
    - **Relatively easy to implement supporting geodetic continuity**

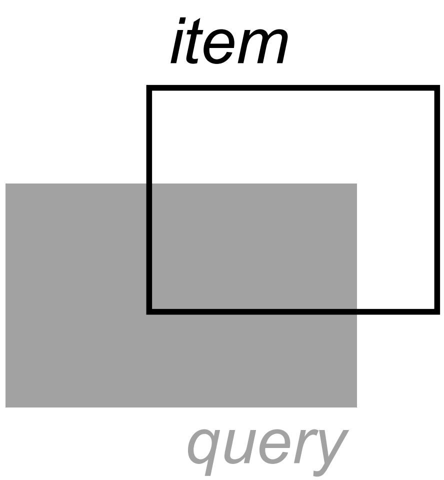
Spatial ranking

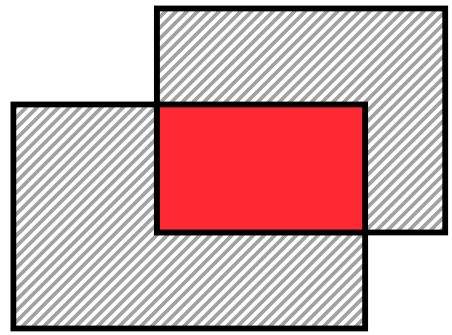
- Observed phenomenon:
  - World Map is first result of every query

- Idea: rank by spatial similarity to query region

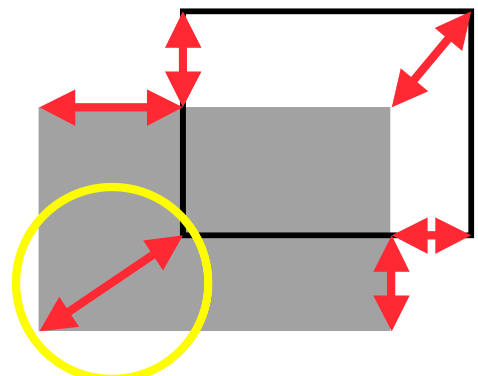


Spatial similarity functions





Intersection/
union ratio



Hausdorff
distance

Take-away points

- DL spatial interoperability is distinct from GIS and distributed GIS problems
  - Global view ("geodetic continuity")
  - Lightweight
- Specific recommendation for spatial interoperability
  - Geodetic boxes
  - *INTERSECTS* predicate
- Spatial ranking is necessary