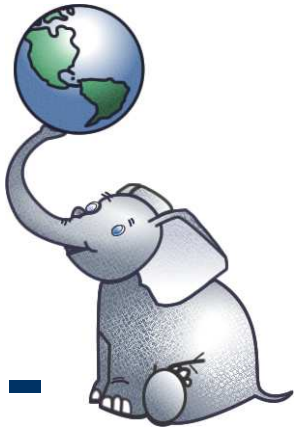


Building Location Aware Apps - Get Started with PostGIS

PART II

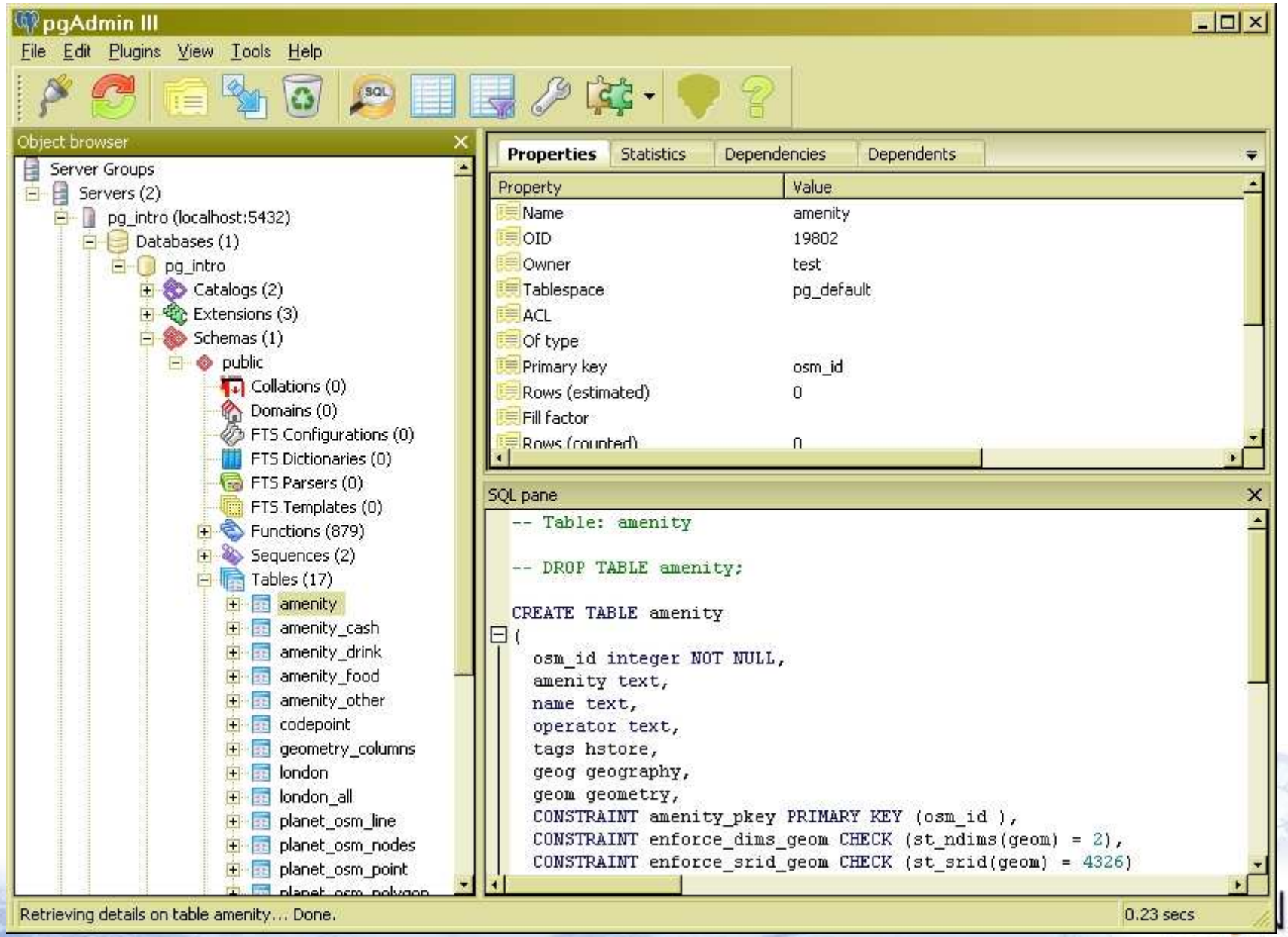
Lasma Sietinsone

l.sietinsone@ed.ac.uk

Code

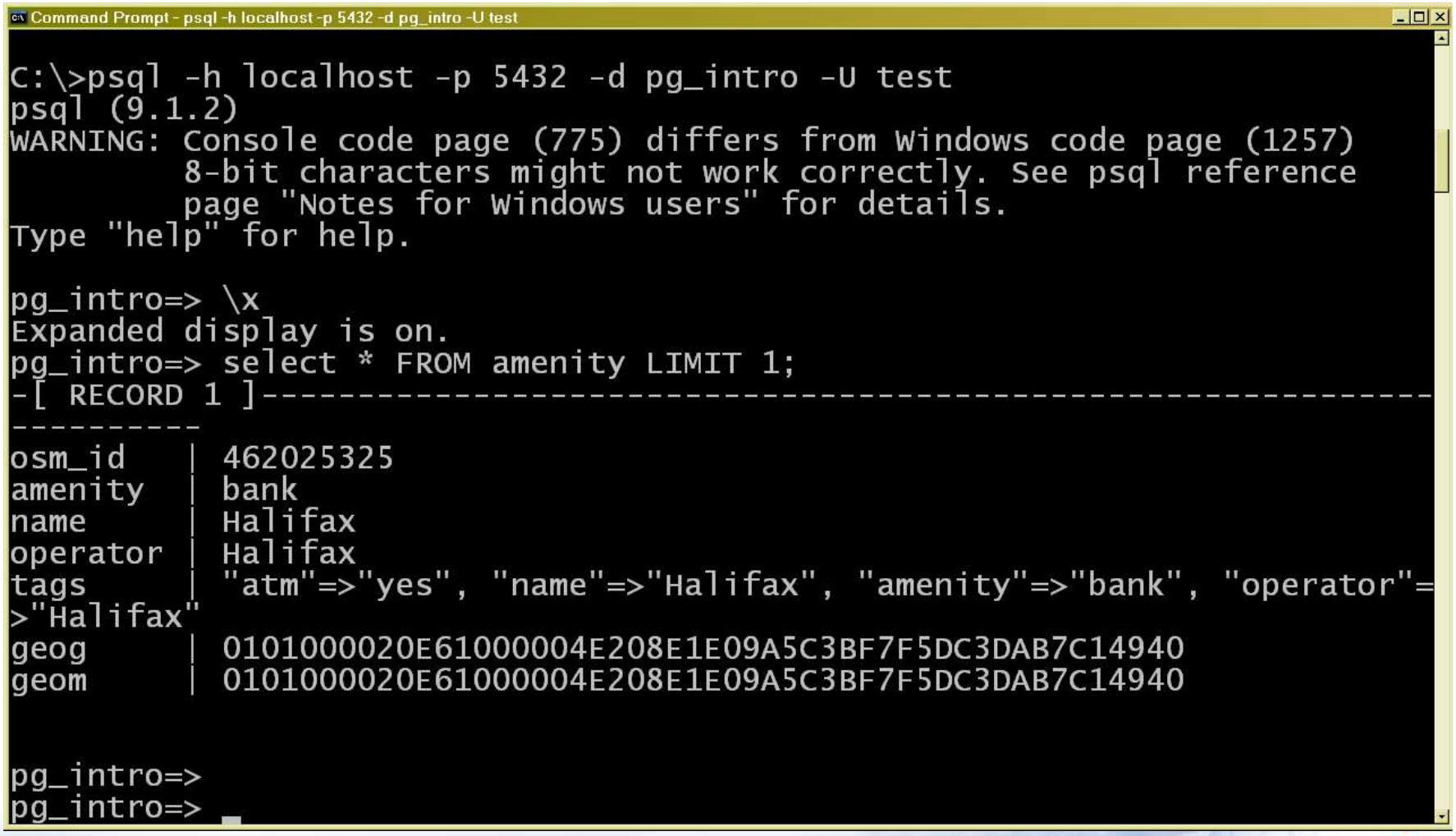
http://gismatic.com/dev8d/scripts/

pgAdmin III – GUI interface for PostgreSQL



psql - PostgreSQL interactive terminal

Powerful but need to know commands!



EDINA®

PostGIS – quick show around

- Functions start with ST_*
- Metadata tables:
  - geometry_columns
  - spatial_ref_sys
- Utilities:
  - Shp2pgsql (import)
  - Pgsql2shp (export)

EXERCISE 1 - import ESRI Shapefile

- File: greater_london_const_region.shp
- *shp2pgsql, shp2pgsql-gui:* load SHP, DBF files

```
$ shp2pgsql  --help
$ shp2pgsql [options] [shapefile schema.table  | psql
    –h [host] –p [port] –U [user] [database]
```

- Derive centroids:
  - Place labels on the centre point of my geometries
- Derive bounding boxes:
  - Get all Flickr photos that fall within each bounding box
- Dissolve/union shapes
- View data with QGIS

EDINA®

Desktop GIS Tools – Fun!

- View database contents (QGIS, OpenJUMP)
- Create on-the-fly maps (OpenJump)

QGIS – Quantum GIS
http://www.qgis.org/

OPENJump
http://www.openjump.org/

EDINA®

EXERCISE 2 - import CSV file with postcodes

- File: wc.csv
- Standard formats: CSV, TEXT

```
$ psql --help
$ psql –h [host] –p [port] –U [user] –c “COPY [tablename]
  FROM [filename] WITH CSV DELIMITER ‘,’” [database]
```

- Make geometry from latitude & longitude (northing & easting)
- Set spatial reference system
- Add spatial index
- Cluster data

EDINA®

EXERCISE 3 - import OSM

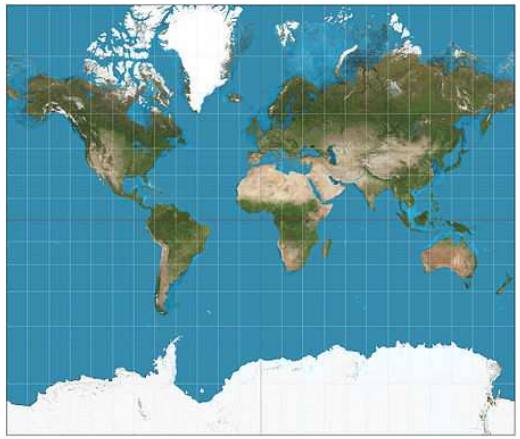
- *osm2pgsql*: load OSM data

```
$ osm2pgsql  --help
$ osm2pgsql [options] planet.osm
$ osm2pgsql [options] planet.osm.{gz,bz2}
$ osm2pgsql [options] file1.osm file2.osm file3.osm
```

- Query tags
- Geometry in text, geojson and other formats
- Align data with different SRS – projections & transformations!

Map projections (SRS, SRID)

- **Conformal** – preserve angle
  - WGS84 (EPSG:4326) – Google Earth
  - OSGB 1936 (EPSG:27700) - British National Grid
  - *Google Projection ("EPSG:900913) – Google Maps*
- **Equal-area** – preserve area
- **Equidistance** – preserve distance for some standard line



http://spatialreference.org

EDINA®

EXERCISE 4 – backend for "POIs" app

OSM based nearest POI (Point of Intereset) queries for location enabled mobile device

e.g.

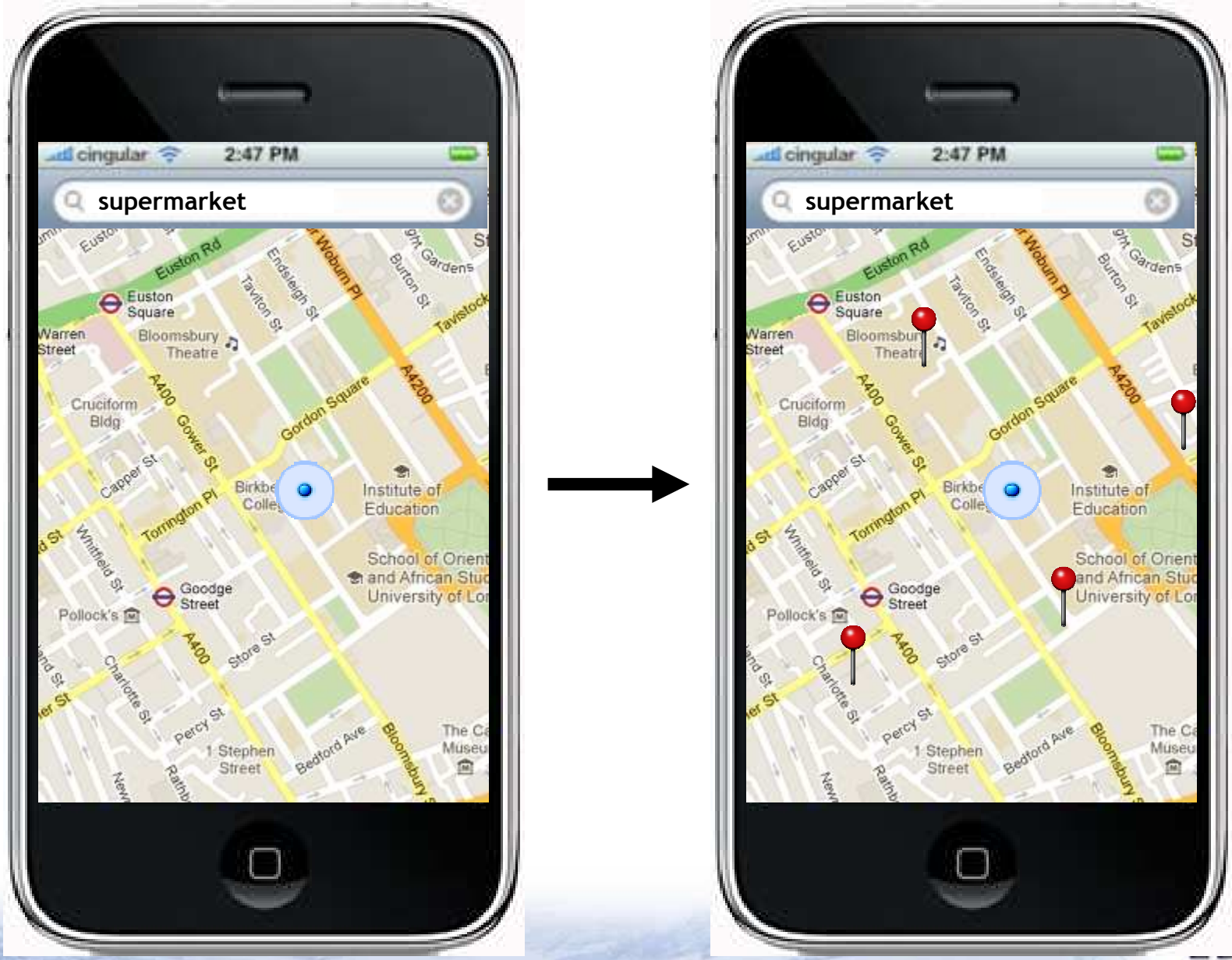
pub? (-> food?) -> cash? -> food?

supermarket??

hotel?

EDINA®

EXERCISE 4 – backend for "POIs" app (cont.)



EXERCISE 4 – backend for "POIs" app (cont.)

- Create user tables for OSM POIs:
  - 08_demo_init_poi_tables.sql
- Create stored procedure for location queries:
  - 09_demo_stored_procedure.sql

Thank you!

Questions?

EDINA®