SPAWAR
Systems Center
PACIFIC

ICODE-MDA Maps

Overview of Talk

- Databases
- How to setup PostgreSQL database
- Accessing the database
- ICODE-MDA Maps tool

Databases

- Organization of large amounts of data
- Store and retrieve data efficiently
- Better than using “flat files” (i.e. text file from Notepad or Microsoft Excel® spreadsheets)
- Databases:
  - Can perform complex queries, or searches over data, very efficiently
  - Can update data easily
  - Are reliable
  - Can be accessed over the network

Databases

- Popular Relational Database Management Systems (RDBMS)
  - MySQL
  - PostgreSQL
  - SQLite
  - Microsoft SQL Server
  - Microsoft Access
  - Oracle

Databases

- Databases from different DBMS not portable
- Use set standards to interface with different databases:
  - SQL
  - ODBC
  - JDBC
- Allows an application to access more than one database from different DBMS

PostgreSQL (current version: 9.2)

- Open Source RDBMS
- http://www.postgresql.org/
- Good for geospatial data
  - PostGIS extension (more later)
- Version 9.2.4 Manual:
  - http://www.postgresql.org/files/documentation/pdf/9.2/postgresql-9.2-US.pdf
- Front-end tools:
  http://wiki.postgresql.org/wiki/Community_Guide_to_PostgreSQL_GUI_Tools
  - psql – command line tool
  - pgAdminIII – GUI administration tool
  - phpPgAdmin

Methods to install PostgreSQL Database

- EnterpriseDB - One-Click Installer
  - Comes with latest version of PostgreSQL and PgAdmin III GUI
  - However, not easy to move database “data” directory
  - Not straightforward how to install PostGIS extension
- BitNami WAPP Stack
  - Complete PostgreSQL, Apache, PHP, and phpPgAdmin
  - http://bitnami.com/tag/postgresql
- Compile and install from source

https://wiki.postgresql.org/wiki/Running_%26_Installing_PostgreSQL_On_Native_Windows

Setup a local PostgreSQL Database (Linux)

- Linux installation - simplified:

```
./configure
gmake
su
gmake install
adduser postgres
mkdir /usr/local/pgsql/data
chown postgres /usr/local/pgsql/data
su - postgres
/usr/local/pgsql/bin/initdb -D /usr/local/pgsql/data
/usr/local/pgsql/bin/postgres -D /usr/local/pgsql/data
>logfile 2>&1 &
/usr/local/pgsql/bin/createdb test
/usr/local/pgsql/bin/psql test
```

PostgreSQL No-Install Method

- No-installation PostgreSQL setup instructions:
  - http://www.postgresonline.com/journal/archives/172-Starting-PostgreSQL-in-windows-without-install.html
- Great for development:
  - Can move “data” directory easily
  - Have different instances of databases running separately
  - Can move entire setup between computers easily
- However, may not be safe for productions systems
  - Security
  - Reliability

Setup a no-install PostgreSQL Database

1. Download PostgreSQL binaries:
   - http://www.enterprisedb.com/products-services-training/pgbindownload
2. Download PostGIS:
   - http://download.osgeo.org/postgis/windows/pg92/
3. Extract files to **C:/pgsql** folder:
   - postgresql-9.2.4-1-windows-x64-binaries
   - postgis-pg92-binaries-2.0.3w64
     - If prompted of duplicate folders and files, click “Yes” to overwrite
4. Create startup script – copy from:

http://www.postgresonline.com/journal/archives/172-Starting-PostgreSQL-in-windows-without-install.html

Setup a no-install PostgreSQL Database

- Create a **run_postgresql_server.bat** file in **C:/pgsql** folder:

```
set POSTGRESQL_ROOT=C:\pgsql\pgsql-9.2.4
set DATABASE_ROOT=C:\pgsql\databasefiles
PATH=%POSTGRESQL_ROOT%\bin;%PATH%
title PostgreSQL Server Running…
@ECHO ON
REM The script sets environment variables helpful for PostgreSQL
@SET PATH="%POSTGRESQL_ROOT%\bin";%PATH%
@SET PGDATA=%DATABASE_ROOT%\data
@SET PGDATABASE=postgres
@SET PGUSER=postgres
@SET PGPORT=5432
@SET PGLOCALEDIR=%POSTGRESQL_ROOT%\share\locale
REM Next line MUST be uncommented the first time to init the server, it can then be commented.
REM "%POSTGRESQL_ROOT%\bin\initdb" -U postgres -A trust
"%POSTGRESQL_ROOT%\bin\pg_ctl" -D "%DATABASE_ROOT%/data" -l %DATABASE_ROOT%/logfile start
ECHO "Click enter to stop"
pause
"%POSTGRESQL_ROOT%\bin\pg_ctl" -D "%DATABASE_ROOT%/data" stop
```

First time running PostgreSQL Server

1. Initialize
   - Remove REM from following line:

     ```
     REM "%POSTGRESQL_ROOT%\bin\initdb" -U postgres -A trust
     ```

   - Execute **run_postgresql_server.bat**
   - Hit any key to quit script when done

2. Run PostgreSQL server
   - Add REM back to line above
   - Execute **run_postgresql_server.bat**

First time running PostgreSQL Server

3. Set password for user postgres (admin user)
   - Open command prompt at **C:\pgsql\pgsql-9.2.4\bin**
   - Run command **psql.exe –U postgres**
   - Run command in psql prompt:

     postgres=# **alter user postgres with password ‘postgrespw’;**
4. Create a new user
   - postgres=# **create user hpcstudent with password 'hpcpassword';**
5. Create a database and grant permissions to new user
   - postgres=# **create database gisdata;**
   - postgres=# **grant all privileges on database gisdata to hpcstudent**
   - postgres=# **\q**
6. Shutdown database (stop **run_postgresql_server.bat** script)

First time running PostgreSQL Server

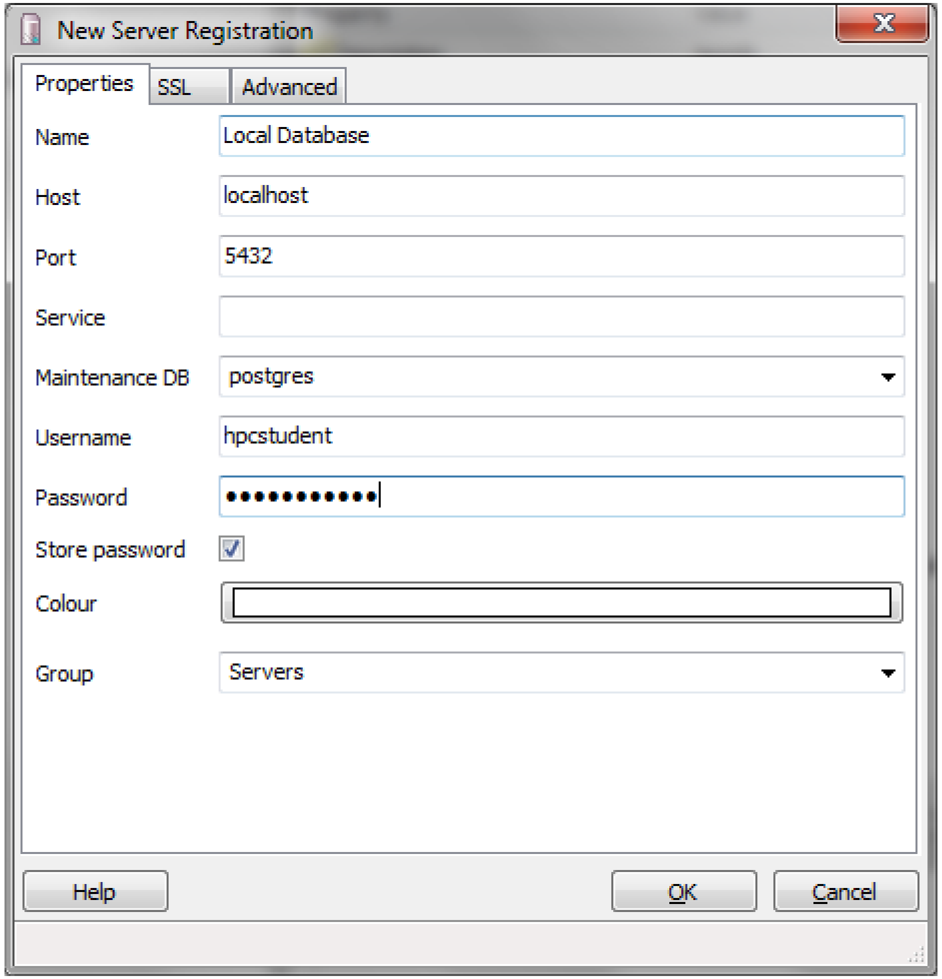
7. Update configuration to require password for login
   - Edit **pg_hba.conf** file
   - Change all instance of **trust** to **md5**
   - Save and close

- PostgreSQL database is now ready for use, with a database named **gisdata** for use

- To run PostgreSQL server normally:
  - Run **run_postgresql_server.bat** script

pgAdmin III: GUI Administration Tool for PostgreSQL

- Open pgAdmin III
- Add new server connection with values shown to the right →



New Server Registration

Name: Local Database

Host: localhost

Port: 5432

Service:

Maintenance DB: postgres

Username: hpcstudent

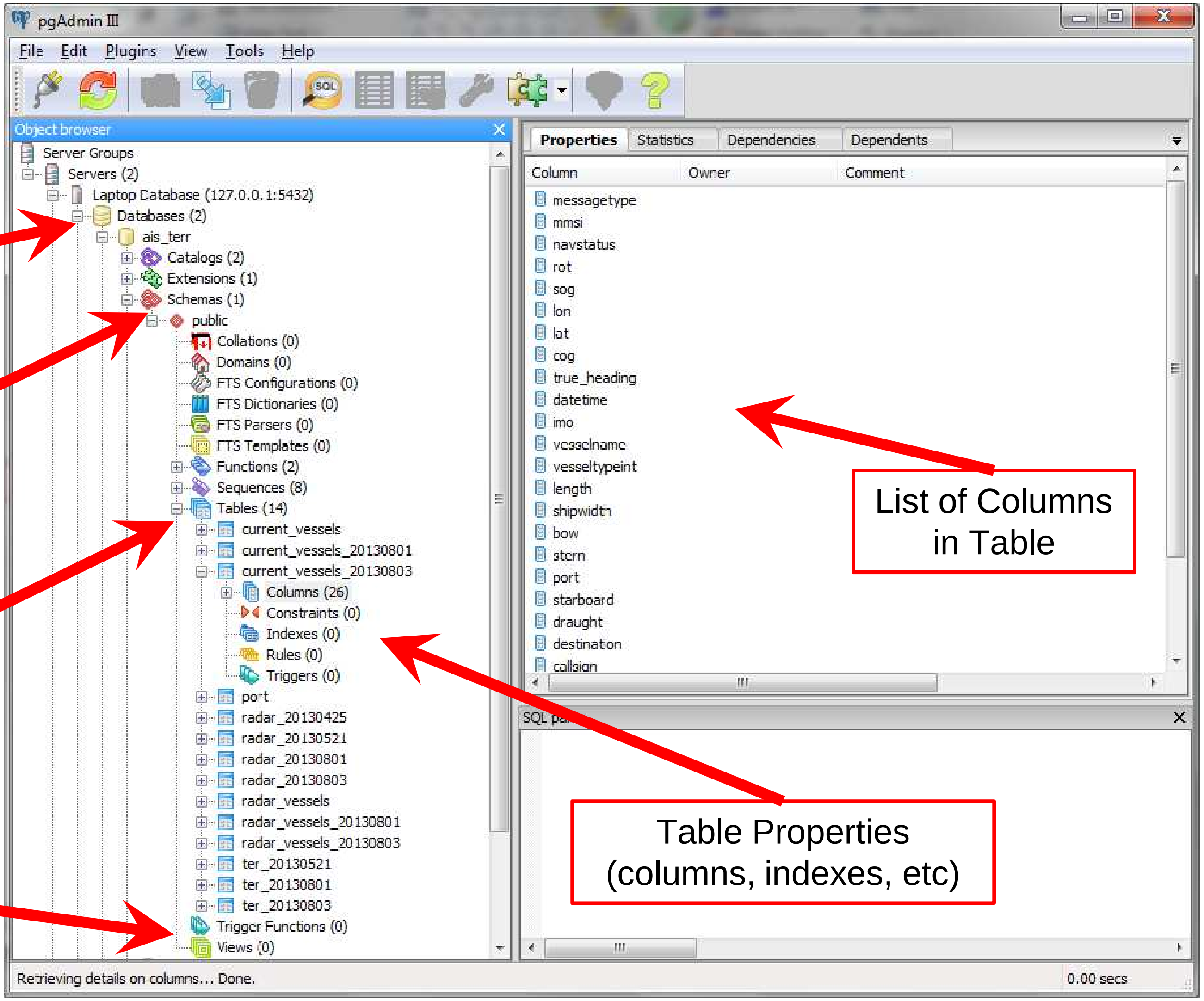
Password: ●●●●●●●●●●●●

Store password: ☑

Colour:

Group: Servers

pgAdmin III: GUI Administration Tool for PostgreSQL


pgAdmin III
File Edit Plugins View Tools Help
Object browser
Properties Statistics Dependencies Dependents
SQL pane

- Server Groups
  - Servers (2)
    - Laptop Database (127.0.0.1:5432)
      - Databases (2)
        - ais_terr
          - Catalogs (2)
          - Extensions (1)
          - Schemas (1)
            - public
              - Collations (0)
              - Domains (0)
              - FTS Configurations (0)
              - FTS Dictionaries (0)
              - FTS Parsers (0)
              - FTS Templates (0)
              - Functions (2)
              - Sequences (8)
              - Tables (14)
                - current_vessels
                - current_vessels_20130801
                - current_vessels_20130803
                  - Columns (26)
                  - Constraints (0)
                  - Indexes (0)
                  - Rules (0)
                  - Triggers (0)
                - port
                - radar_20130425
                - radar_20130521
                - radar_20130801
                - radar_20130803
                - radar_vessels
                - radar_vessels_20130801
                - radar_vessels_20130803
                - ter_20130521
                - ter_20130801
                - ter_20130803
              - Trigger Functions (0)
              - Views (0)

Column	Owner	Comment
messagetype		
mmsi		
navstatus		
rot		
sog		
lon		
lat		
cog		
true_heading		
datetime		
imo		
vesselname		
vesseltypeint		
length		
shipwidth		
bow		
stern		
port		
starboard		
draught		
destination		
callsign		

Retrieving details on columns... Done. 0.00 secs

List of Databases

List of Schemas

List of Tables

Trigger functions and View

List of Columns in Table

Table Properties (columns, indexes, etc)

pgAdmin III:
GUI Administration Tool for PostgreSQL

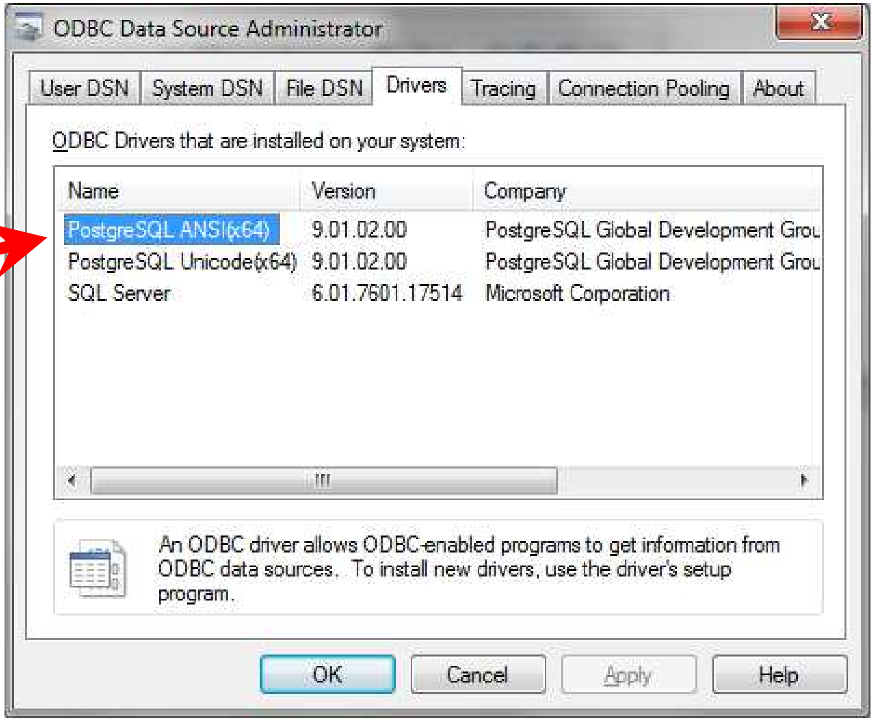
- **public** Schema will be used by default
- Recommend to create new schema for each data type
    - For example: ais_data, coastal_radar_data, etc.
    - Multiple users to not interfere with each other
    - Better organization of different datasets
    - For objects of same name to not conflict
- Create tables in each schema to store data
    - Can be performed through commands using psql.exe or pgAdmin III

Accessing a PostgreSQL Database

- PHP, Python, Matlab, Perl, C++ (libpq) has built-in functions to access PostgreSQL database:
  - Each has a different syntax to connect to different DBMS
- Can also use JDBC driver
- This course: Use PostgreSQL **ODBC drivers**
  - Standard middleware API, independent of OS and DBMS
  - http://www.postgresql.org/ftp/odbc/versions/
  - MSI driver versions for PostgreSQL 9.2
    - 32-bit - psqlodbc_09_02_0100.zip
    - 64-bit - psqlodbc_09_02_0100-x64.zip

Accessing a PostgreSQL Database

- Verify driver installation of drivers by running:
  - ODBC Data Source Administrator
    - Start -> search for “odbc” -> Data Source (ODBC)
    - Orígenes de datos ODBC



ODBC Drivers that are installed on your system:

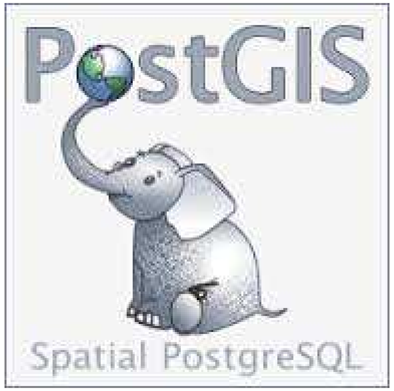
Name	Version	Company
PostgreSQL ANSI(x64)	9.01.02.00	PostgreSQL Global Development Grou
PostgreSQL Unicode(x64)	9.01.02.00	PostgreSQL Global Development Grou
SQL Server	6.01.7601.17514	Microsoft Corporation

An ODBC driver allows ODBC-enabled programs to get information from ODBC data sources. To install new drivers, use the driver's setup program.

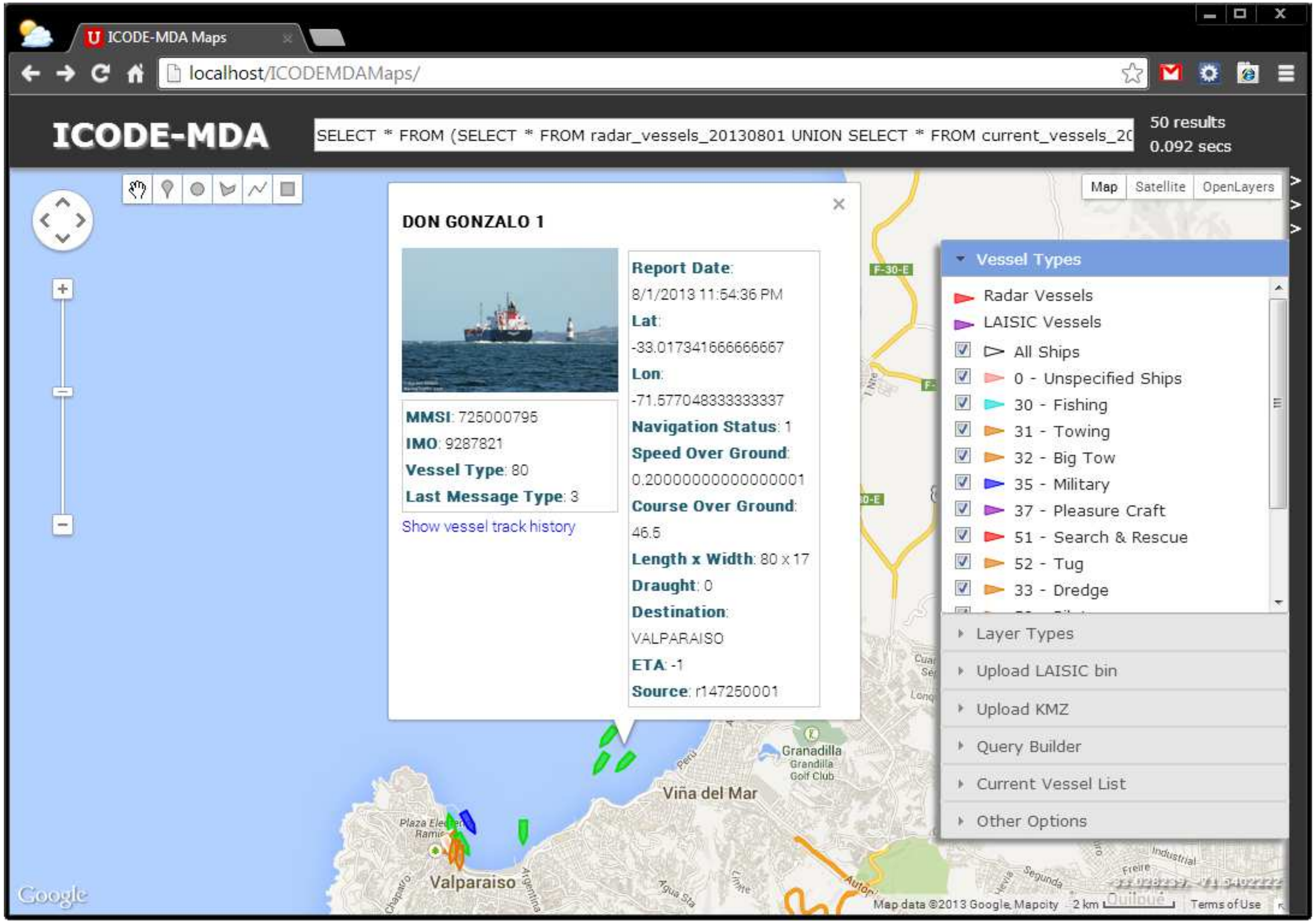
Find PostgreSQL drivers here OK!

PostGIS – GIS Extension to PostgreSQL

- Adds GIS-specific functionality to PostgreSQL databases
- http://www.postgis.net/
- Run the following SQL scripts to install functionality:
  - postgis.sql
  - spatial_ref_sys.sql
  - postgis_comments.sql
- Additional installation instructions:
  - http://postgis.refractions.net/documentation/manual-1.4/ch02.html#PGInstall



ICODE-MDA Maps – Developer's Tool



ICODE-MDA Maps

localhost/ICODEMDAMaps/

ICODE-MDA

SELECT * FROM (SELECT * FROM radar_vessels_20130801 UNION SELECT * FROM current_vessels_20

50 results
0.092 secs

DON GONZALO 1

MMSI: 725000795
IMO: 9287821
Vessel Type: 80
Last Message Type: 3

Show vessel track history

Report Date:
8/1/2013 11:54:36 PM
Lat:
-33.017341666666667
Lon:
-71.577048333333337
Navigation Status: 1
Speed Over Ground:
0.20000000000000001
Course Over Ground:
46.5
Length x Width: 80 x 17
Draught: 0
Destination:
VALPARAISO
ETA: -1
Source: r147250001

Vessel Types

- ▶ Radar Vessels
- ▶ LAISIC Vessels
- ☑ ▷ All Ships
- ☑ ▶ 0 - Unspecified Ships
- ☑ ▶ 30 - Fishing
- ☑ ▶ 31 - Towing
- ☑ ▶ 32 - Big Tow
- ☑ ▶ 35 - Military
- ☑ ▶ 37 - Pleasure Craft
- ☑ ▶ 51 - Search & Rescue
- ☑ ▶ 52 - Tug
- ☑ ▶ 33 - Dredge

ICODE-MDA Maps – Setup

- Obtain code from ICODE-MDA Google Code trunk:
  - https://icode-mda.googlecode.com/svn/trunk/googleMapsIcode/PostgreSQLMaps
- Edit **phpsql_dbinfo.php** with your database credentials
- Place code into UniServer's **www** folder
  - UniServer is a WAMP tool that includes Apache HTTP server, MySQL, and PHP for Windows (LAMP for Linux)
    - Can use BitNami's WAPP that includes PostgreSQL instead of MySQL
  - Use UniServer's HTTP server to host map

title

- body