PostGIS for Managers

pramsey@boundlessgeo.com

Boundless

Thanks for coming. This is PostGIS for Managers. FOSS4G does tend to be a pretty technical conference. The ratio of nerds to suits is very high, so I have to admit I was originally tempted to title this talk

PostGIS for Managers!?!?!?!

pramsey@boundlessgeo.com

Boundless

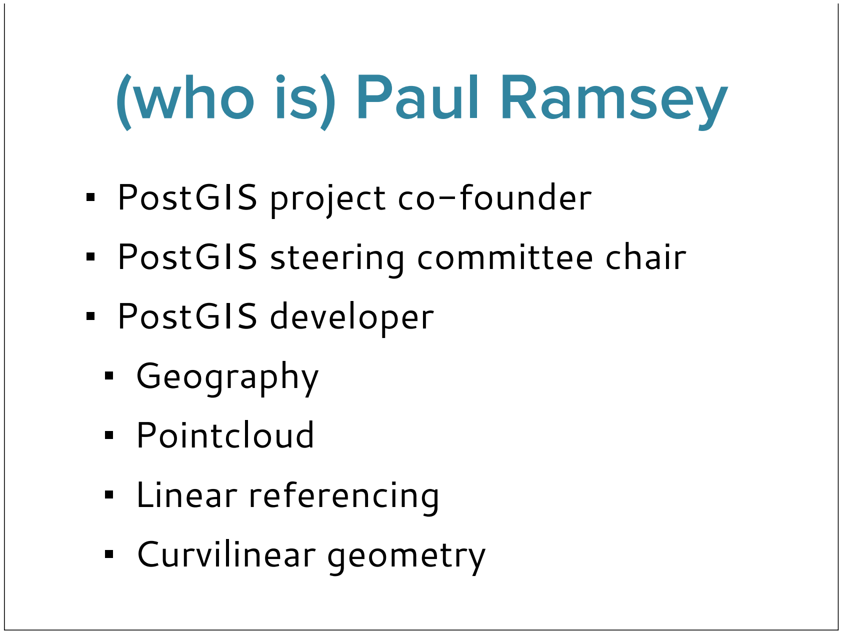
PostGIS for Managers?!?!?!?
But in fact there's a real constituency for this kind of thing, I think, amongst decision makers who are considering getting their feet wet with open source, or who have started down the road, but are still uncertain about experimenting with something as mission critical as the enterprise database.

Boundless

and I’ll let you in on a little secret, at least one of my points
will involve recommending engaging either the company I
work for, or one very much like it,
so I have decided that,
in order to avoid any appearance of conflict of interest
I am going to precede any blatantly self-serving advice

All glory to the Hypnotoad!

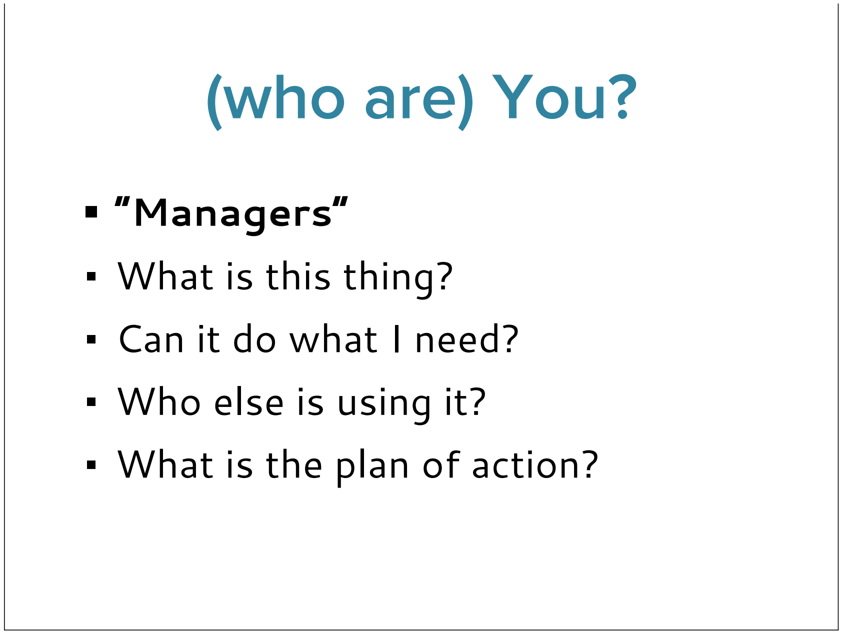
with an invocation of the hypnotoad.
All glory to the Hypnotoad.



(who is) Paul Ramsey

- PostGIS project co-founder
- PostGIS steering committee chair
- PostGIS developer
  - Geography
  - Pointcloud
  - Linear referencing
  - Curvilinear geometry

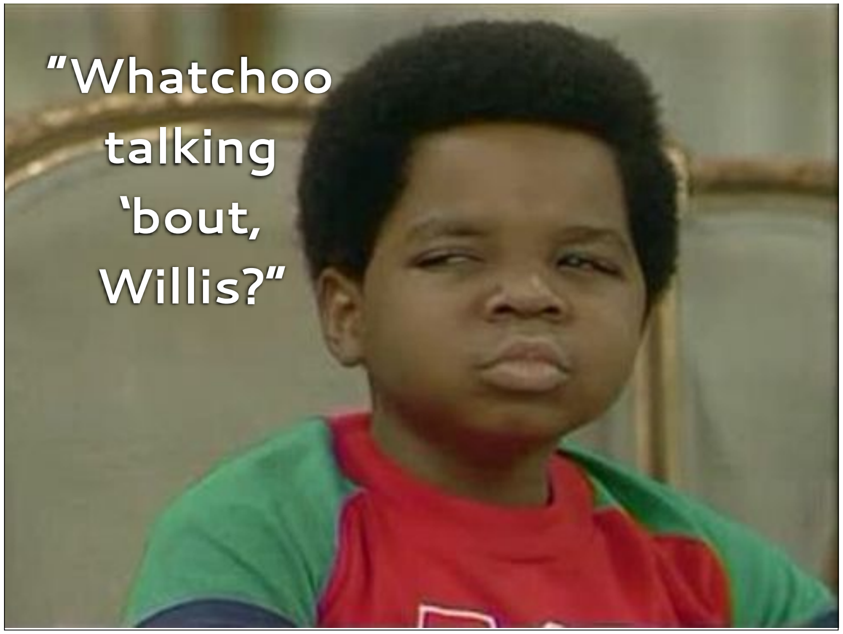
And since I've just shown the hypnotoad, I might as well cover some self-serving biographical details
...*...*...*...
and I'm also senior evangelist and developer at Boundless.
I'm here all week, try the veal.



(who are) You?

- **"Managers"**
- What is this thing?
- Can it do what I need?
- Who else is using it?
- What is the plan of action?

So, as an audience, here for "postgis for managers", what kind of folks are you.
How many of you would describe yourselves as managers?
What kinds of question of questions are important to managers?
Probably not language of implementation,
or SQL analysis tricks, or library dependencies.
More practical things, from an organizational strategy point of view,
like...
.* * * *..
So, to start off, what is this thing?

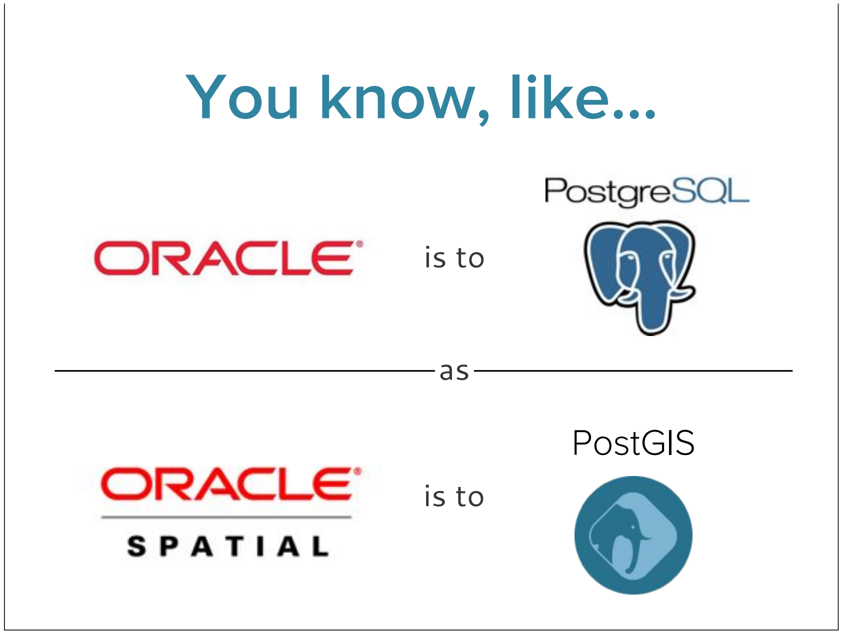


“Whatchoo talking ‘bout, Willis?”

Or as Gary Coleman might put it,
whatchoo talking bout, Ramsey?
What’s this “postgis” thing, anyhow?

PostGIS is a spatial database

It’s a great big cylinder. It’s the place you store your data, and the thing that answers questions about your data, like “what’s in this polygon” or “how far from the store?”
But perhaps it’s better to reason by analogy,



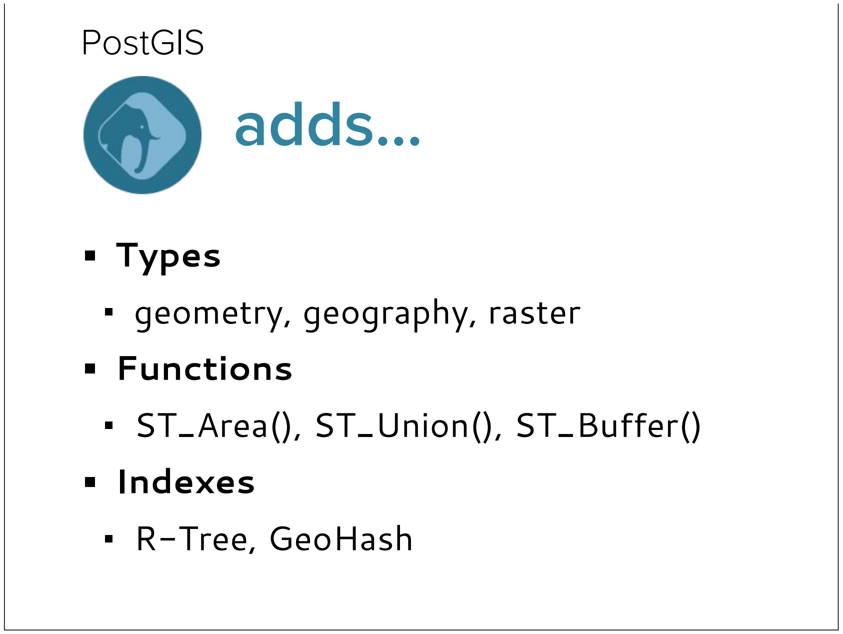
You know, like...

ORACLE is to PostgreSQL

as

ORACLE SPATIAL is to PostGIS

As Oracle is to PostgreSQL,
Oracle Spatial is to PostGIS.
A spatial database is just a database at the heart, but with some extra goodness added for dealing with spatial data and queries.

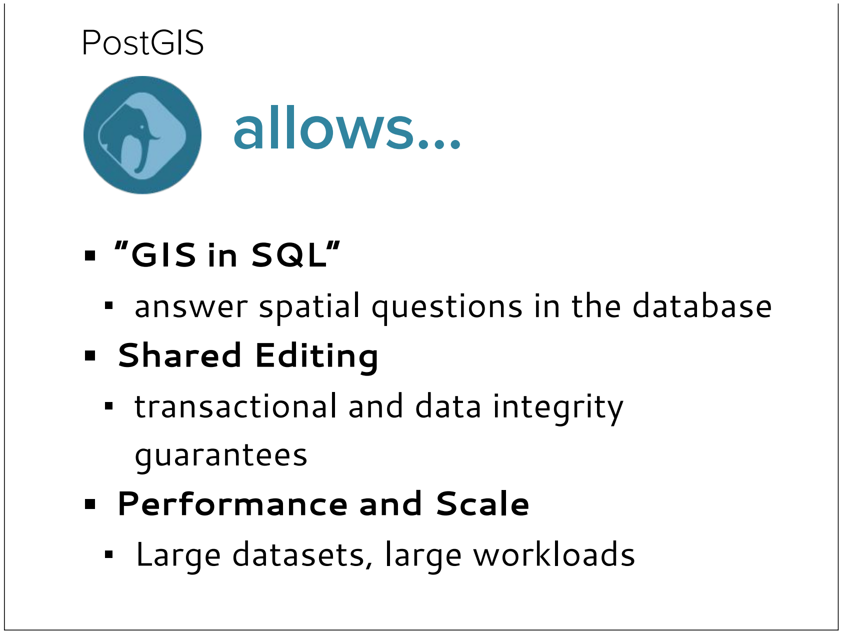


PostGIS adds...

- **Types**
  - geometry, geography, raster
- **Functions**
  - ST_Area(), ST_Union(), ST_Buffer()
- **Indexes**
  - R-Tree, GeoHash

So PostGIS takes plain vanilla PostgreSQL and adds,
* types, like...
* functions, like...
* indexes, like...

And once you have these extra pieces in your ordinary database,
these types, functions and indexes,
your database becomes less ordinary, it can do new and wonderful things

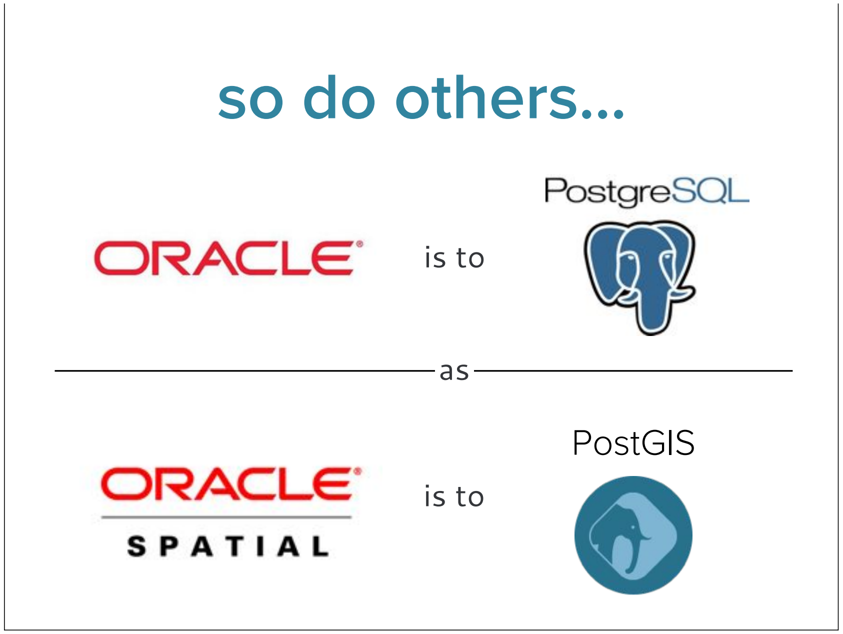


PostGIS

allows...

- **"GIS in SQL"**
  - answer spatial questions in the database
- **Shared Editing**
  - transactional and data integrity guarantees
- **Performance and Scale**
  - Large datasets, large workloads

like do GIS queries directly in the database
: generate a list of neighbours to notify, calculate the average daily drive distance of the truck, summarize parcel area by zoning code, you name it
* unlike GIS files, databases are built to maintain data integrity, under write load from multiple sources, so no passing files around, and more integration with other real-time systems
* also unlike files, databases expect to be asked to handle large datasets and large workloads with aplomb



so do others...

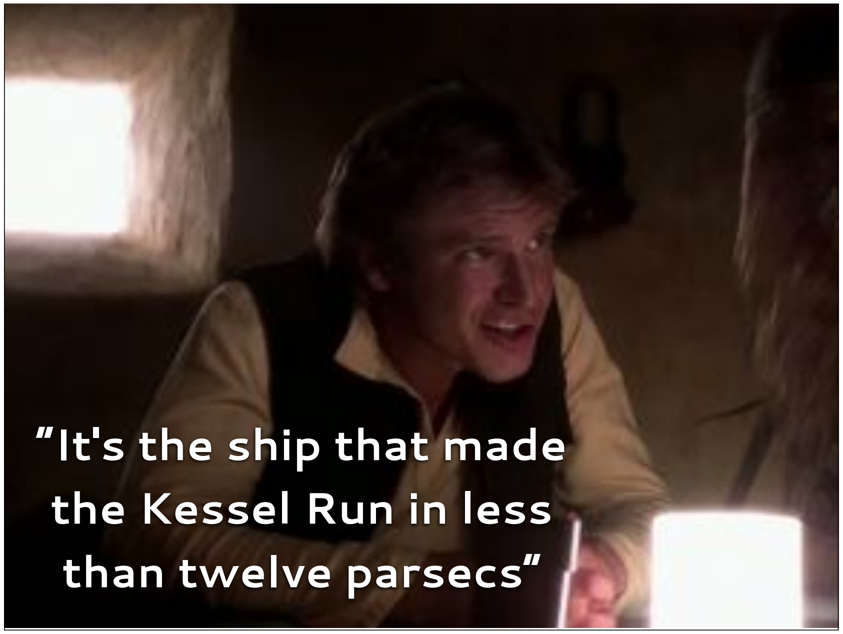
ORACLE is to PostgreSQL

as

ORACLE SPATIAL is to PostGIS

and really, all the awesome things about PostGIS are also awesome things about other spatial databases too, they all let you answer GIS questions in the database, break away from the tyranny of files and file-oriented tools, and handle big data sets efficiently.

But is PostGIS in the same league as those other databases, as the SQL Servers and the Oracles,

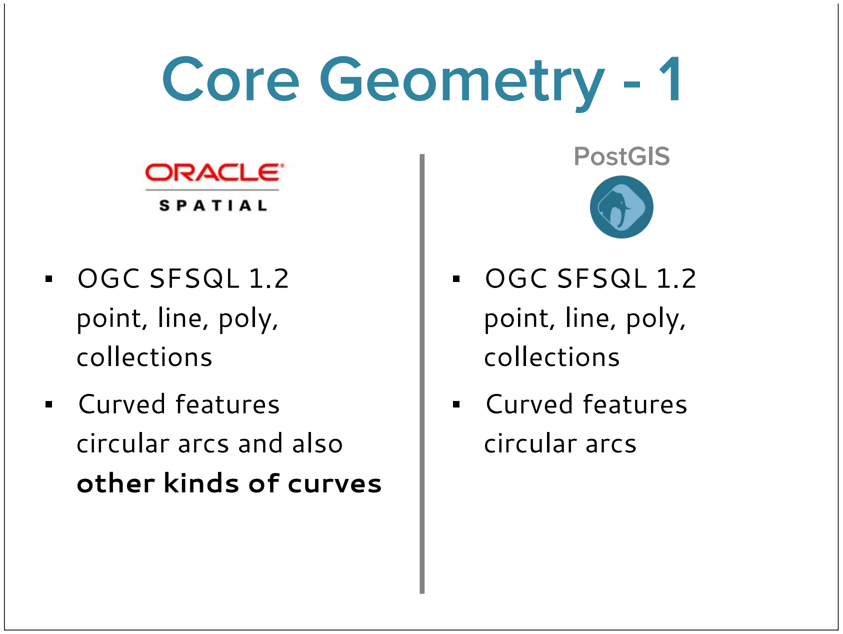


"It's the ship that made the Kessel Run in less than twelve parsecs"

Let's talk technology. Let's talk raw capability, let's get our nerdy on.

Rather than just talk about what PostGIS can do, which is a lot,
people find it more convincing when I put it next to the "leading commercial brand",
which is, let's face it, Oracle.

So, here's a quick, but thorough feature comparison,



Core Geometry - 1

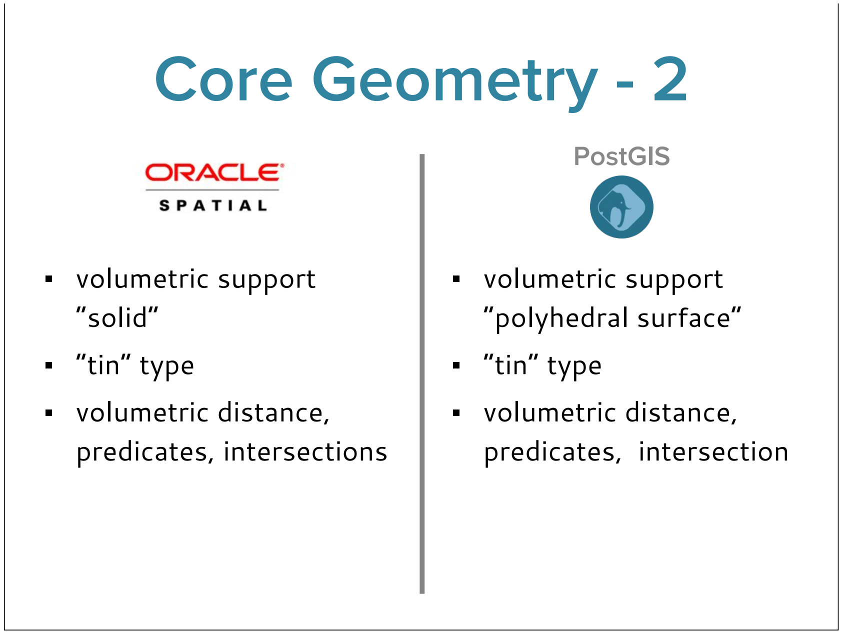
ORACLE SPATIAL

- OGC SFSQL 1.2
  point, line, poly, collections
- Curved features
  circular arcs and also **other kinds of curves**

PostGIS

- OGC SFSQL 1.2
  point, line, poly, collections
- Curved features
  circular arcs

Both databases support basic geometry types (points, lines and polygons) as defined by the Open Geospatial Consortium, and both support arcs as defined by the ISO, but Oracle supports some fancy kinds of arcs that CAD folks like.



Core Geometry - 2

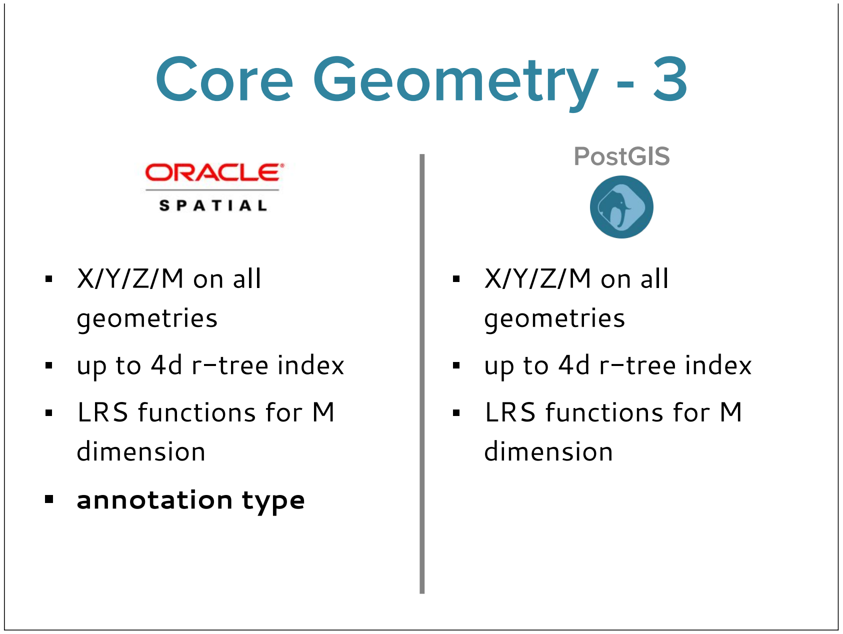
ORACLE SPATIAL

- volumetric support "solid"
- "tin" type
- volumetric distance, predicates, intersections

PostGIS

- volumetric support "polyhedral surface"
- "tin" type
- volumetric distance, predicates, intersection

Both databases have similar support for storing and retrieving 3d “volumes”, and you can see more of the PostGIS 3D support in the talk being given by Olivier Courtin on Friday at 11 about “GIS goes 3D”.



Core Geometry - 3

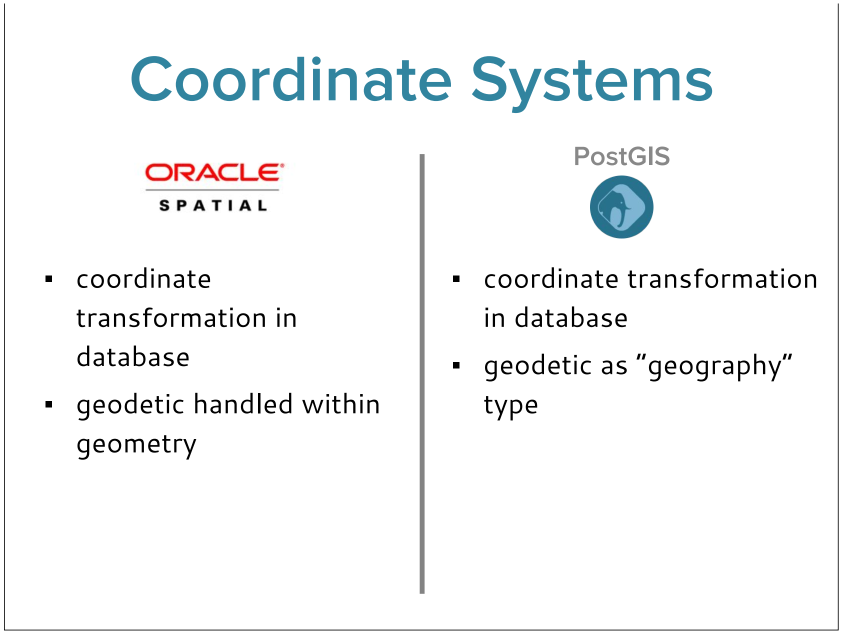
ORACLE SPATIAL

- X/Y/Z/M on all geometries
- up to 4d r-tree index
- LRS functions for M dimension
- **annotation type**

PostGIS

- X/Y/Z/M on all geometries
- up to 4d r-tree index
- LRS functions for M dimension

Both databases store geometries with more than 2 dimensions, including a Z dimension and a “measure” or “M” dimension, and both can index geometries over all four dimensions. Oracle has a quirky “annotation” type, which again is for CAD program support.



Coordinate Systems

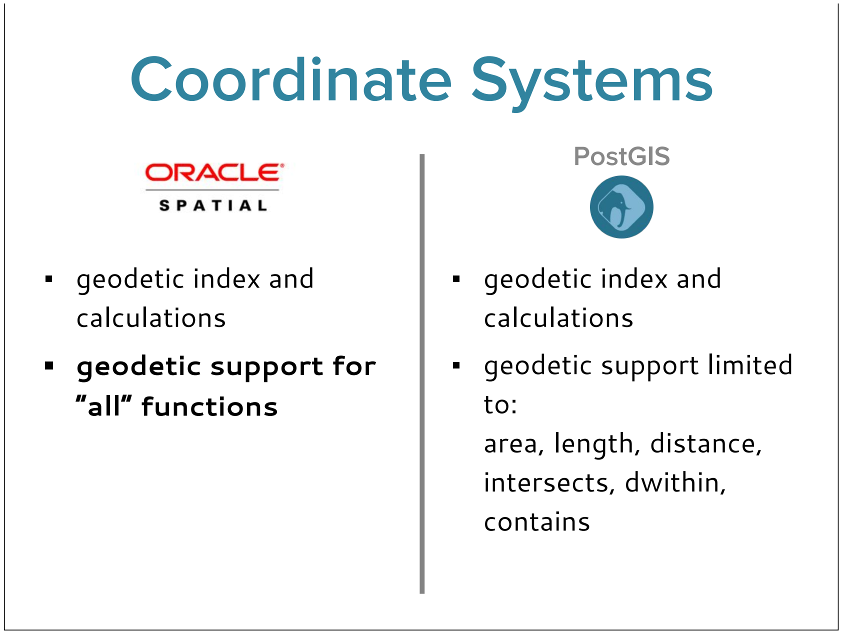
ORACLE SPATIAL

- coordinate transformation in database
- geodetic handled within geometry

PostGIS

- coordinate transformation in database
- geodetic as ”geography“ type

Both databases can handle geodetic coordinates, latitudes and longitudes, natively, though the implementation details are slightly different.



Coordinate Systems

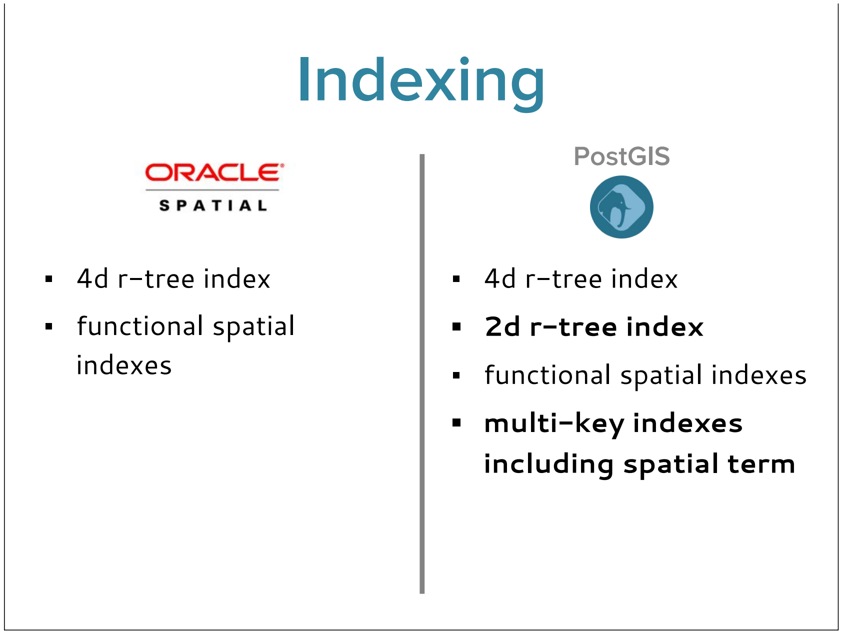
ORACLE SPATIAL

- geodetic index and calculations
- **geodetic support for ”all“ functions**

PostGIS

- geodetic index and calculations
- geodetic support limited to:
  area, length, distance, intersects, dwithin, contains

Both have indexes and functions that support geodetic coordinates, though Oracle has support for more geodetic functions.



Indexing

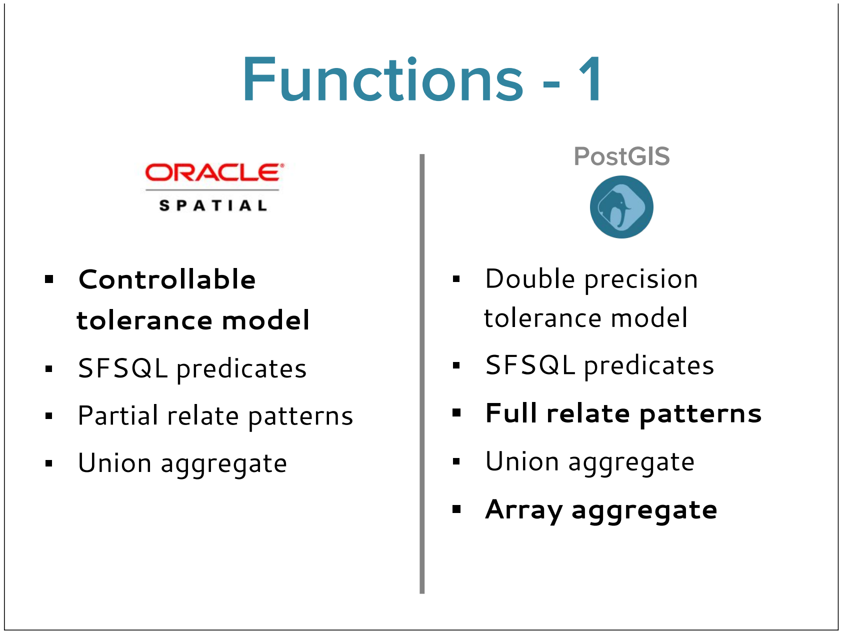
ORACLE SPATIAL

- 4d r-tree index
- functional spatial indexes

PostGIS

- 4d r-tree index
- **2d r-tree index**
- functional spatial indexes
- **multi-key indexes including spatial term**

Both have spatial indexing, though PostGIS has an extra 2d-only index for higher performance in common cases, and can do multi-key indexes against non-spatial columns, which is a result of the spatial index functionality being more generic and integrated into the database.



Functions - 1

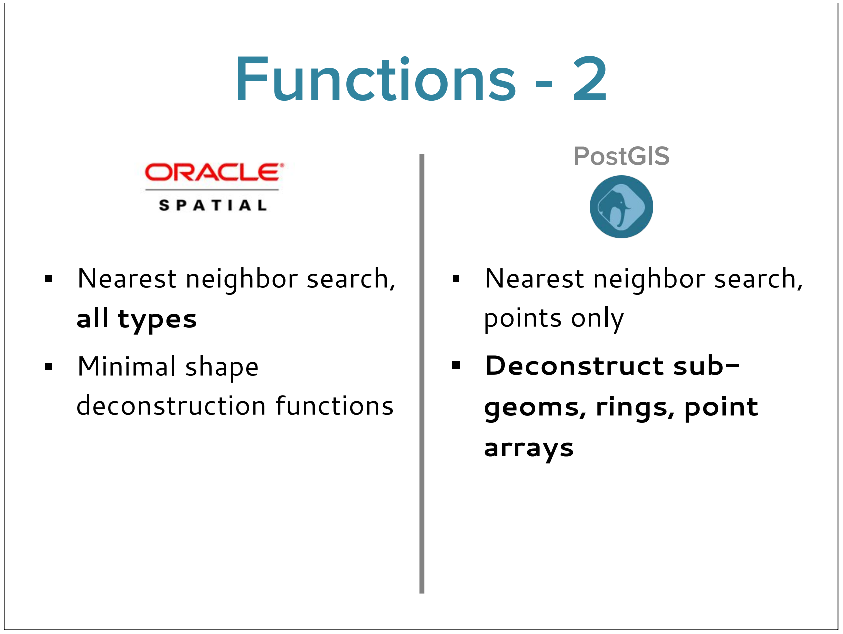
ORACLE SPATIAL

- **Controllable tolerance model**
- SFSQL predicates
- Partial relate patterns
- Union aggregate

PostGIS

- Double precision tolerance model
- SFSQL predicates
- **Full relate patterns**
- Union aggregate
- **Array aggregate**

Both databases have a large collection of functions that operate on spatial data. Oracle’s functions have a controllable tolerance model, which is nice. PostGIS has finer detail on spatial relationships, and some handy array and aggregation support Oracle lacks. In general PostGIS just has more functions than Oracle.



Functions - 2

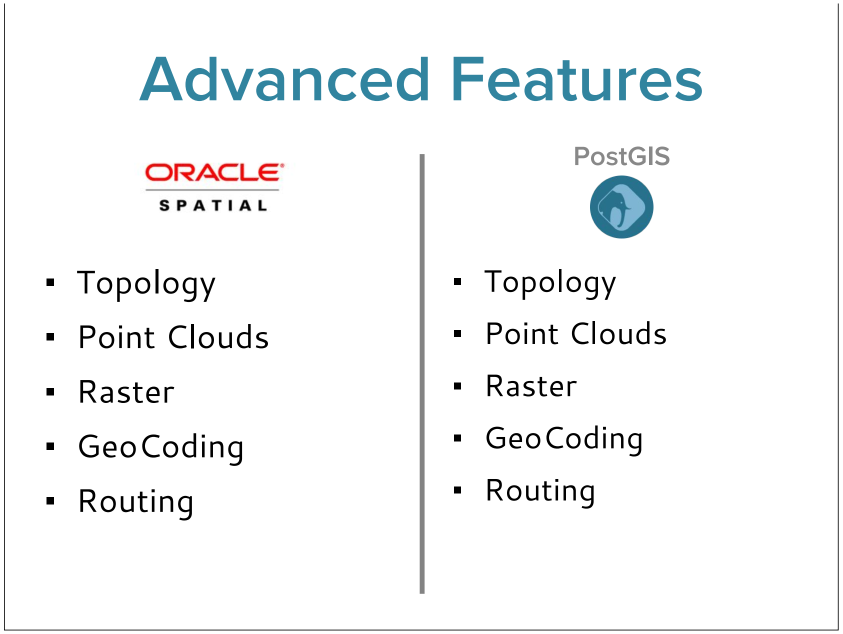
ORACLE SPATIAL

- Nearest neighbor search, **all types**
- Minimal shape deconstruction functions

PostGIS

- Nearest neighbor search, points only
- **Deconstruct sub-geoms, rings, point arrays**

Both support nearest-neighbour searches, though Oracle's support is more complete. PostGIS has better utility functions around geometry manipulation.



Advanced Features

ORACLE SPATIAL

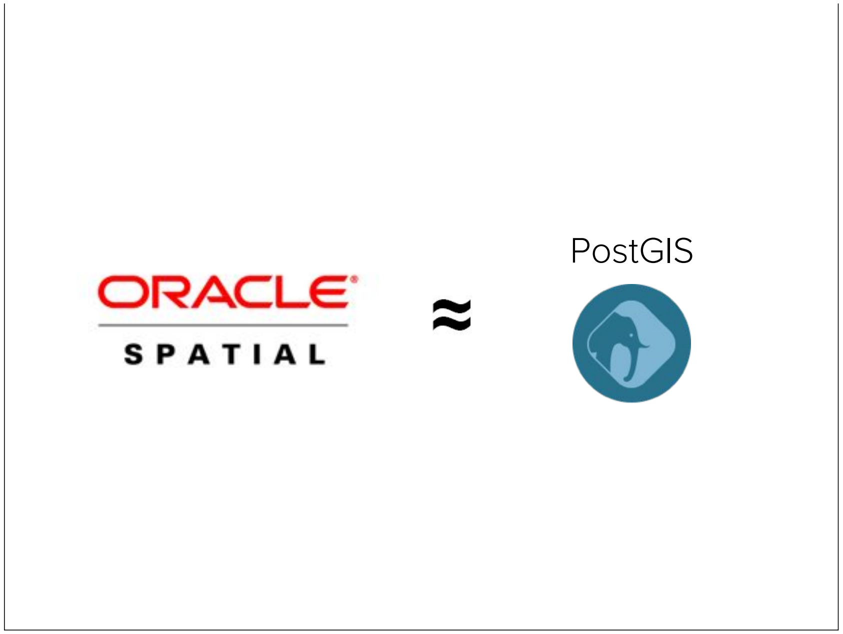
- Topology
- Point Clouds
- Raster
- GeoCoding
- Routing

PostGIS

- Topology
- Point Clouds
- Raster
- GeoCoding
- Routing

And both have a big collection of features that are related to, or built on top of, or often used with, the core spatial functionality, like
topology handling vector coverages
point clouds for lidar data
raster support (for raw storage in Oracle's case and for analytics in PostGIS's case)
geocoding and routing for location based services use cases.

Both databases do an awful lot, and when you add it all up,
the bit where Oracle does a little more
and the bits where PostGIS does a little more, you have to conclude that, functionally



ORACLE SPATIAL ≈ PostGIS

They really aren’t all that different.
They have much the same functionality, but for some small differences.
OK, so PostGIS can do what you need, next question...

“You know the way, right?”

Who else is using PostGIS?
It’s ordinary human nature to feel more comfortable if a decision has been validated by other folks, so let’s take a look at other organizations using PostGIS.

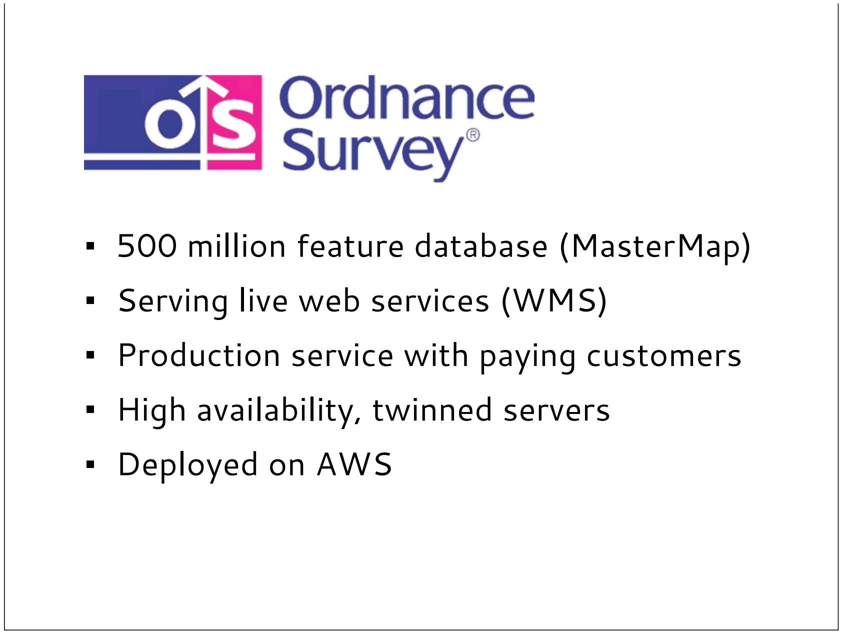
PostgreSQL Operating at Scale

First, Skype!
Actually I don’t know that Skype uses PostGIS at all, but I always like to include them as an example of an organization using PostgreSQL at scale. The data back-end for Skype is PostgreSQL, every call is logged, all billing tracked using PostgreSQL.
Instagram is another example of a PostgreSQL cluster running at scale. So, size isn’t something to worry about, it does scale.

Google™
DIGITALGLOBE®

- Huge catalogs of images in PostGIS
- Where / what / when queries
- Images stored separately, database holds metadata

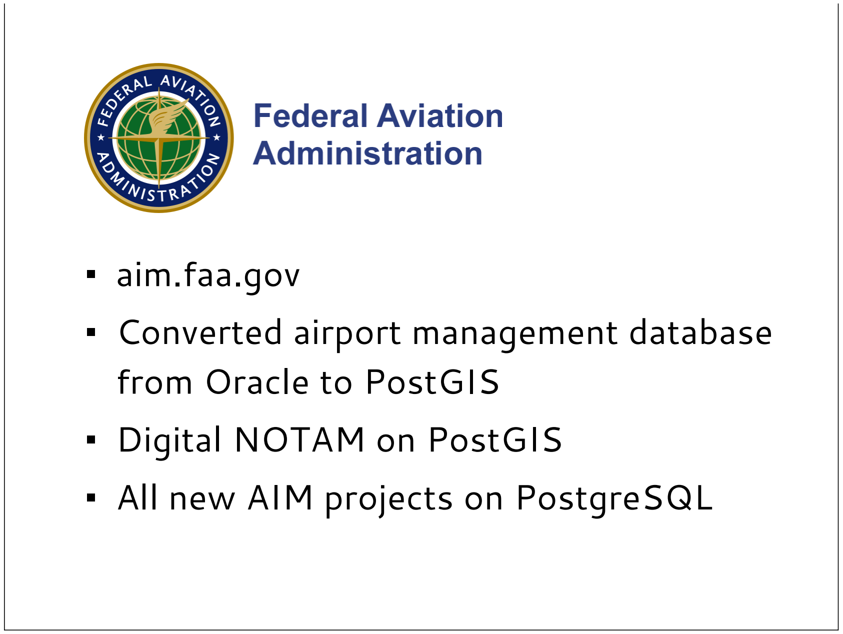
Google and Digital globe are both big organizations that use PostGIS, for similar purposes, to manage their collection of imagery. Not the images themselves, but the image metadata, which allows them to control image preparation automation, find the “best” images for any particular area, find historical data, and so on.



Ordnance Survey

- 500 million feature database (MasterMap)
- Serving live web services (WMS)
- Production service with paying customers
- High availability, twinned servers
- Deployed on AWS

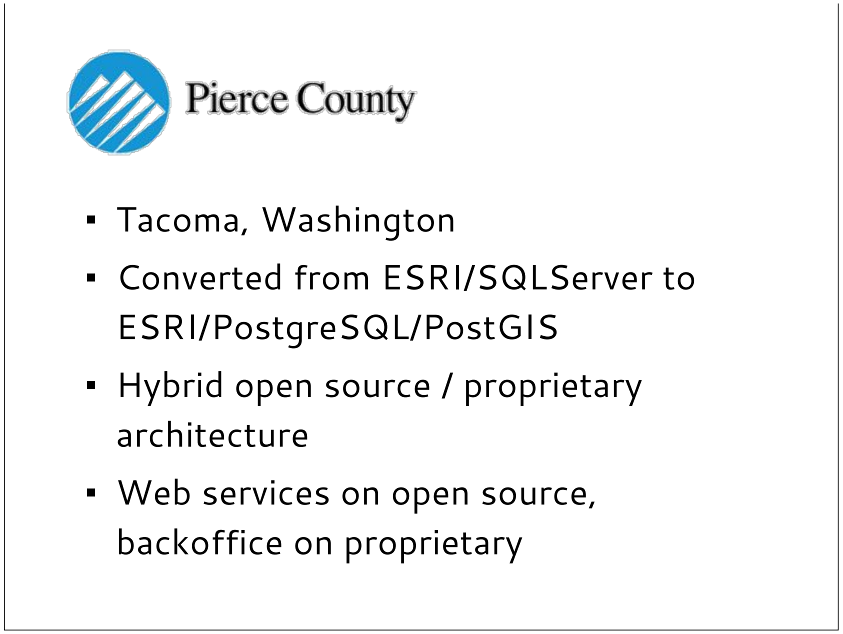
When people ask me what the biggest single dataset being stored in PostGIS is, I usually cite the UK Ordnance Survey, whose MasterMap database is over 500M records in size. There are probably larger datasets now, things like LIDAR collections maybe, but as a genuine, classic GIS dataset, MasterMap is pretty good and very big.
Ordnance Survey uses PostGIS on their web tier, supporting WMS live rendering and deployed on Amazon Web Services.



Federal Aviation Administration

- aim.faa.gov
- Converted airport management database from Oracle to PostGIS
- Digital NOTAM on PostGIS
- All new AIM projects on PostgreSQL

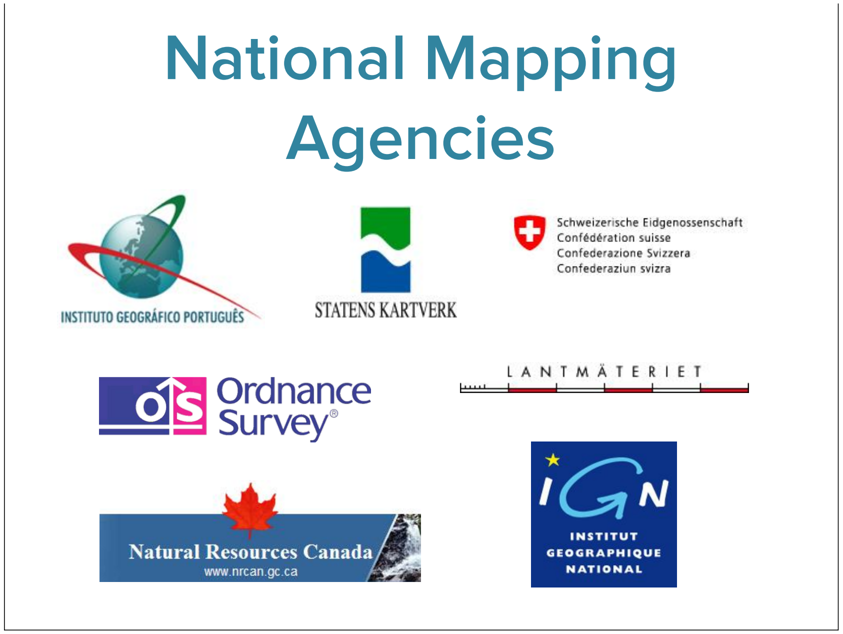
The FAA has moved their airport management data and the notice to aviators databases to PostGIS, and found the results so good that they set up a policy that all future systems migrations are to move away from Oracle to PostgreSQL.



Pierce County

- Tacoma, Washington
- Converted from ESRI/SQLServer to ESRI/PostgreSQL/PostGIS
- Hybrid open source / proprietary architecture
- Web services on open source, backoffice on proprietary

Those have all been big organization examples, but even smaller organizations like Pierce County in Washington State have converted and enjoyed good results. They converted from SQL Server to PostGIS, and maintain a hybrid architecture, using ArcGIS desktops for their traditional GIS management and analysis, but with PostGIS to support automated analysis and web publishing tasks.

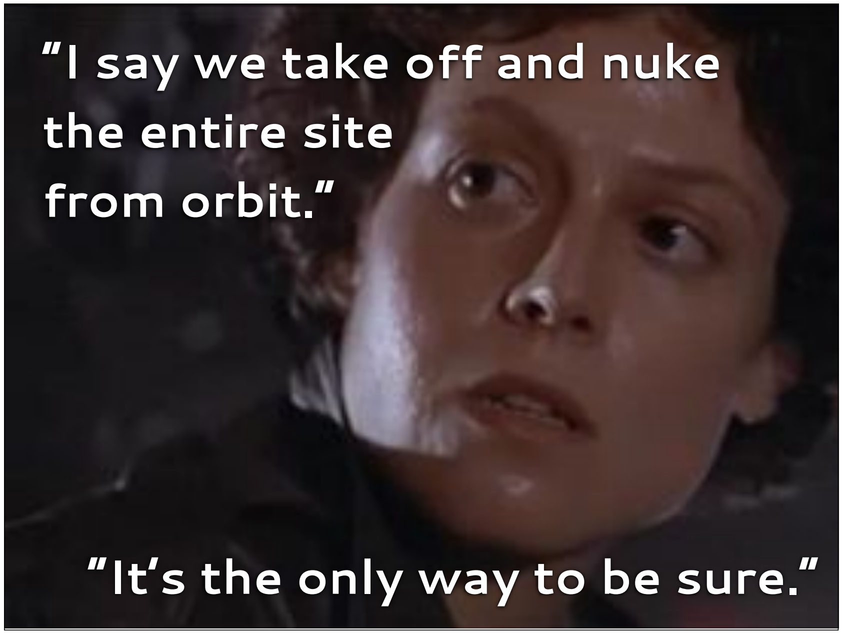


National Mapping Agencies

INSTITUTO GEOGRÁFICO PORTUGUÊS
STATENS KARTVERK
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
Ordnance Survey
LANTMÄTERIET
Natural Resources Canada
www.nrcan.gc.ca
IGN
INSTITUT GEOGRAPHIQUE NATIONAL

I already mentioned UK Ordnance Survey, but national mapping agencies in general have moved to PostGIS in a big way, particular in the web publishing space. Generally open source adoption seems to work in from the edges, starting with web publishing workloads and then taking over existing workloads like corporate databases once it has proven itself over a few years of operations.

So lots of organizations have been able to make a change, from proprietary infrastructures to open source infrastructures, from Oracle and SQL Server to PostGIS. What about an organization just getting started, how do they deal with their legacy of proprietary infrastructure?

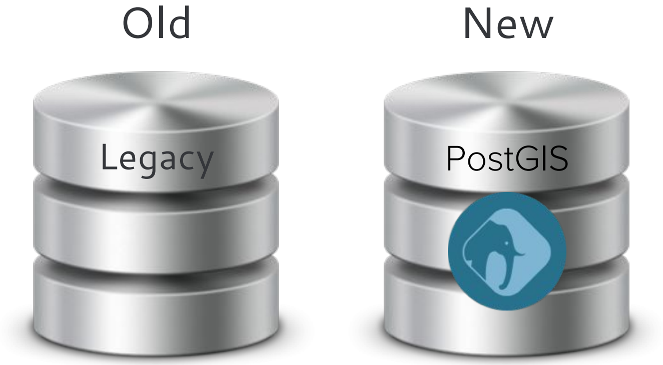


“I say we take off and nuke the entire site from orbit.”

“It’s the only way to be sure.”

One way is to nuke the whole thing from orbit.
It’s an emotionally satisfying plan, but it’s not realistic. An organization cannot cut over all its systems on one day. Day to day operations have to go on, so most staff time will still be devoted to keeping the wheels turning, and only a fraction can be devoted to creating the systems of tomorrow. What’s the solution?

It takes two...



One thing to get comfortable with right away is that during transition there’s going to be two databases running. This situation could easily go on for a number of years. It’s not permanent, but it’s not entirely temporary either: it’s a phase.

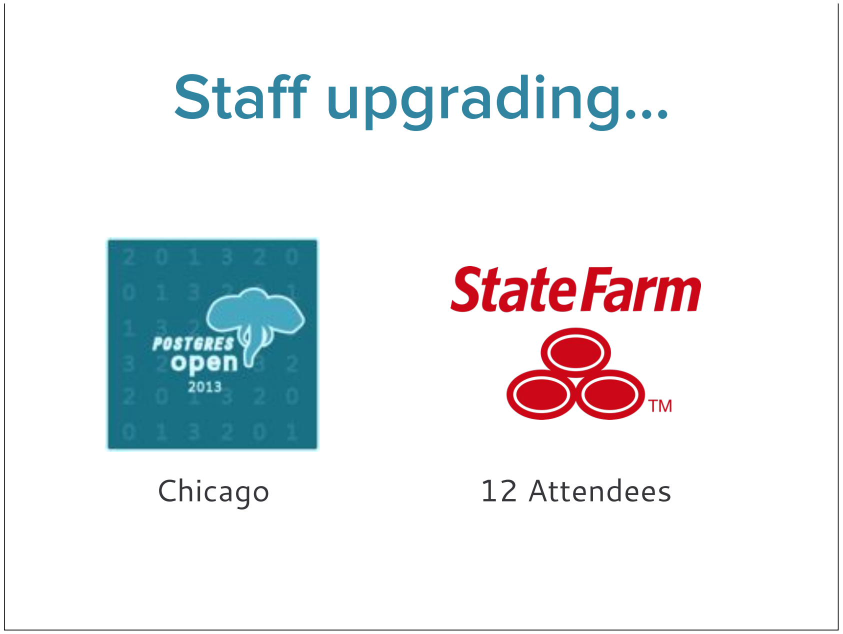
Great place to...

- Sense of psychological reality
- Test out existing tools / interoperability
- Build staff expertise
- Build prototype applications
- Migrate first production applications

Standing up a working PostgreSQL/PostGIS database for your organization is the first step.
It gives you an existence proof. This thing actually runs, under your roof.
It gives you a place to see how well your existing tools work with it.
It gives your staff the hands-on experience they are going to need as you migrate bigger and more complex applications.
It gives you a place to try test migrations of applications.
It gives you a place for migrated applications to live in production.

How to go from...

So, how do you go about turning an Oracle person into a PostgreSQL person.
Is it mission impossible?
Not at all.



Staff upgrading...

Chicago

12 Attendees

Last year I got to go to the PostgresOpen conference in Chicago, and one of the things I noticed was all the folks I met from State Farm Insurance. They are based just down-state in Bloomington Illinois, and they are migrating from Oracle to PostgreSQL. So their migration team, Oracle specialists all, was at PostgresOpen, learning new things.

That's some staff upgrading, and it doesn't take much of it.
- most staff recognize that learning new skills increases their value, it also makes work more interesting
- the transition from one database to another doesn't require much new learning, the core concepts are the same, just terminology and details differ

Staff augmentation...

salesforce
Tom Lane Rejected My Patch and
All I Got Was This Stupid T-Shirt
In short: -1 from me.
regards, tom lane

Oracle Shop **???**

Tom Lane
PostgreSQL Guru

A more extreme example, and one that doesn't apply to most organizations, is that of SalesForce.com, which is an Oracle shop. One day, seemingly out of the blue, they hired Tom Lane, one of the top PostgreSQL developers. Turns out, there were tooling up to migrate their core infrastructure to PostgreSQL. As a software-as-a-service business with a mission critical reliance on their core database, having the #1 expert in their employ was a big risk reducer.

Staff augmentation...

salesforce

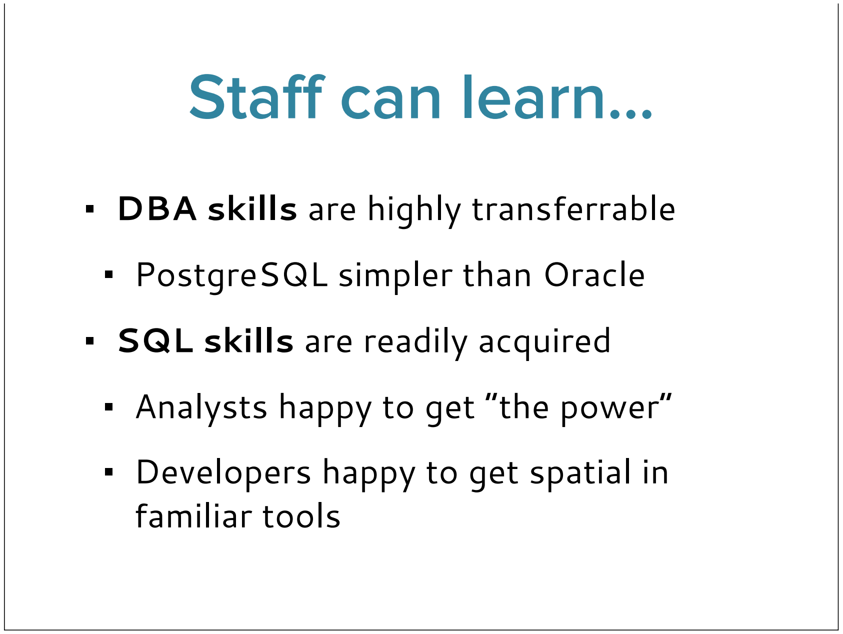
purchased

heroku

Tom Lane
PostgreSQL Guru

As it turned out, SalesForce didn’t end up migrating, they used the threat of migration to dramatically improve their licensing deal with Oracle. But since in the same period they also purchased Heroku, a platform-as-a-service company with a big reliance on PostgreSQL, they’ve kept Tom on. It probably also helps future proof their deal with Oracle.

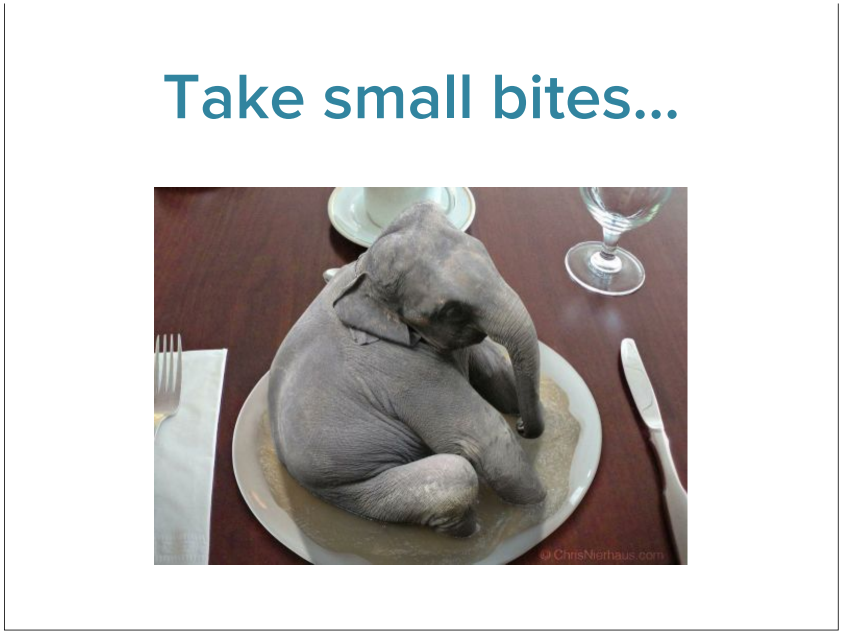
The T-shirt Tom is modelling, BTW, reads “Tom Lane rejected my patch and all I got was this stupid t-shirt. In short, -1 from me, regards Tom Lane.”



Staff can learn...

- **DBA skills** are highly transferrable
  - PostgreSQL simpler than Oracle
- **SQL skills** are readily acquired
  - Analysts happy to get “the power”
  - Developers happy to get spatial in familiar tools

So, staff transition is really not a problem.
* For the DBAs, the road is very smooth. Their core skills are the same, in many ways PostgreSQL is a much easier beast to manage than Oracle.
* For analysts and developers, moving more workflow to the database gives them access to the power of SQL, which are also readily acquired, with a few days of course work. It give the analysts a new power tool, and it gives developers access to spatial functionality without the overhead of learning a new gestalt like GIS.



Take small bites...

So, how to transition? how to eat an elephant?
Take small bits.

- Stand up a parallel server
- Migrate some small applications
- Upgrade staff skills and exercise them on migrations

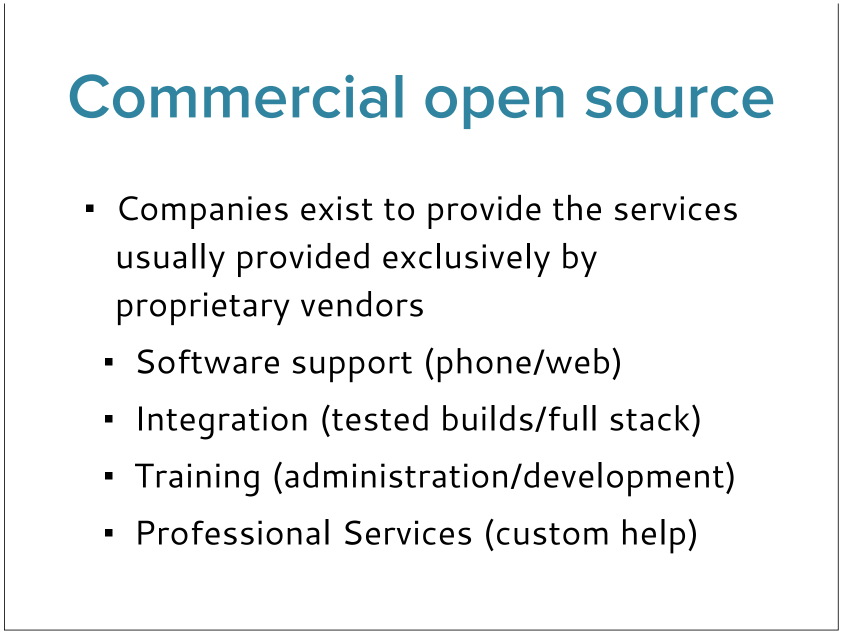
Keep on moving; once you get momentum, make shutting down the legacy server the goal to motivate continuing migrations.

But what about...

But what about support????

All glory to the Hypnotoad!

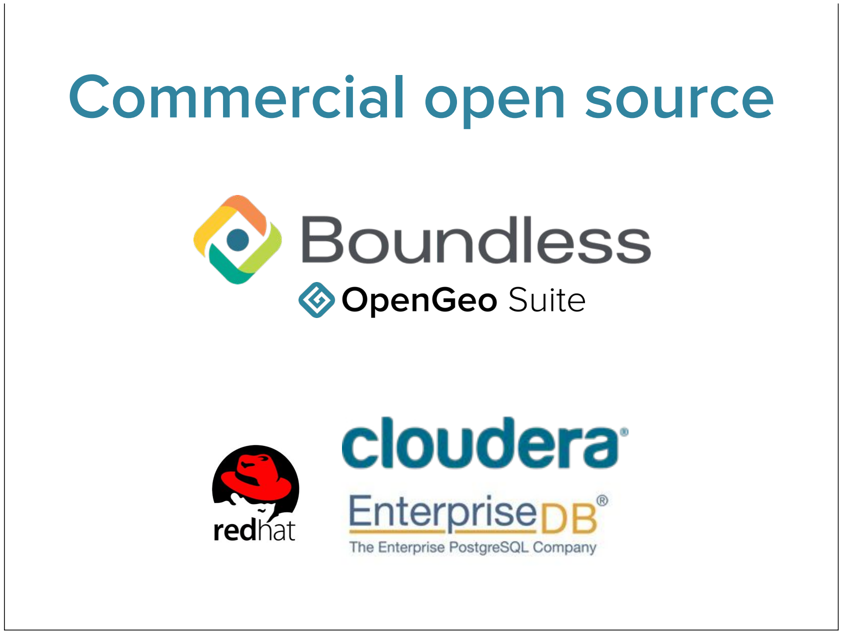
I thought you would never ask...



Commercial open source

- Companies exist to provide the services usually provided exclusively by proprietary vendors
  - Software support (phone/web)
  - Integration (tested builds/full stack)
  - Training (administration/development)
  - Professional Services (custom help)

Good news, there are actually companies that exist to provide all the things usually provided by proprietary vendors,
* the support functions
* testing and integration and building of the source code
* training services
* and professional services
Companies that provide the assurance that the software will be professionally maintained and built for you as a customer, and the insurance that if things DO go wrong, there will be someone to call who can help.



Commercial open source

Boundless
OpenGeo Suite

redhat

cloudera

EnterpriseDB®
The Enterprise PostgreSQL Company

Coincidentally, I happen to work for one of those companies. Boundless provides support subscriptions for OpenGeoSuite, which is a full stack of technology for web mapping, including PostGIS, GeoServer and OpenLayers.
* But we're hardly unique, Cloudera fills that role for Hadoop, and Red Hat for Linux, and EnterpriseDB for plain vanilla PostgreSQL.

So, we've covered all of my managerial questions about PostGIS:

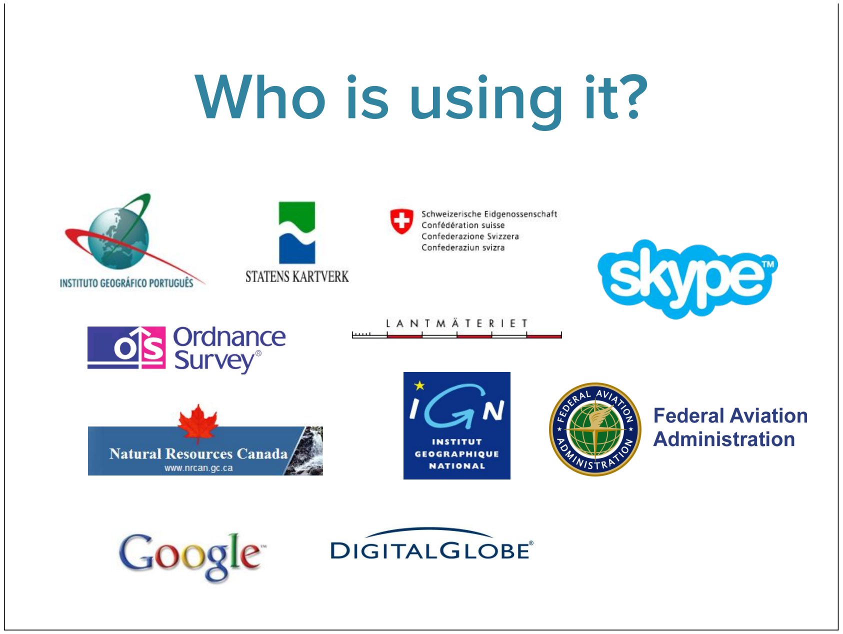
What is this thing?

PostGIS is a spatial database, it allows you to store and query spatial data just like any other data, using SQL as the query language.

Can it do what I need?

ORACLE
SPATIAL
≈
PostGIS

It can do basically everything Oracle Spatial can do.



Who is using it?

INSTITUTO GEOGRÁFICO PORTUGUÊS
STATENS KARTVERK
Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra
skype
Ordnance Survey
LANTMÄTERIET
Natural Resources Canada
www.nrcan.gc.ca
IGN
INSTITUT GEOGRAPHIQUE NATIONAL
FEDERAL AVIATION ADMINISTRATION
Federal Aviation Administration
Google
DIGITALGLOBE

Lots of big serious organizations, both public and private are using PostGIS for real-world work.

What is the plan?

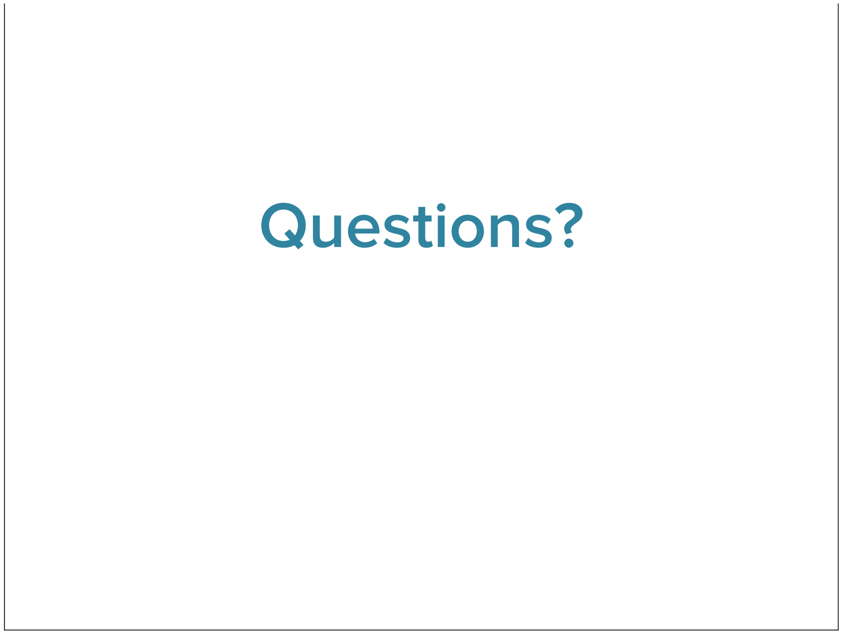
- Run a second database for a while (it's OK)
- Train up your staff, they already have core skills
- Migrate in increments to learn and gain confidence

And getting your organization into PostGIS is not a big left:
- get a development environment up and running
- give your staff some training and learning time
- and migrate your applications a little bit at a time, to learn and grow your skills

What is the alternative?

First Law of Holes

Because the alternative to getting into PostGIS runs afoul of the First Law of Holes, which is
"If you find yourself in a hole, the first thing to do is stop digging"



Questions?

Thanks.

PostGIS for Managers

pramsey@boundlessgeo.com

Boundless