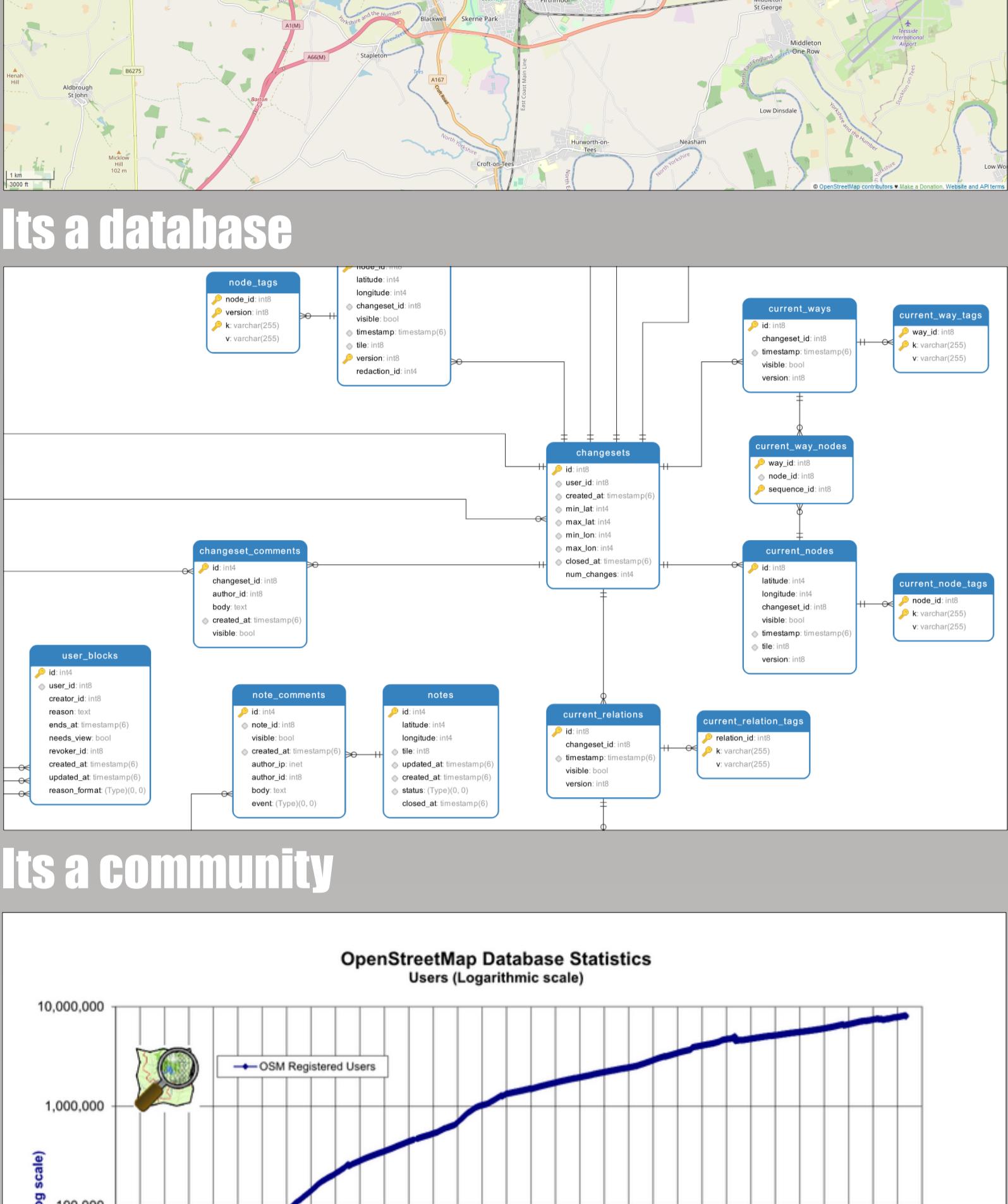


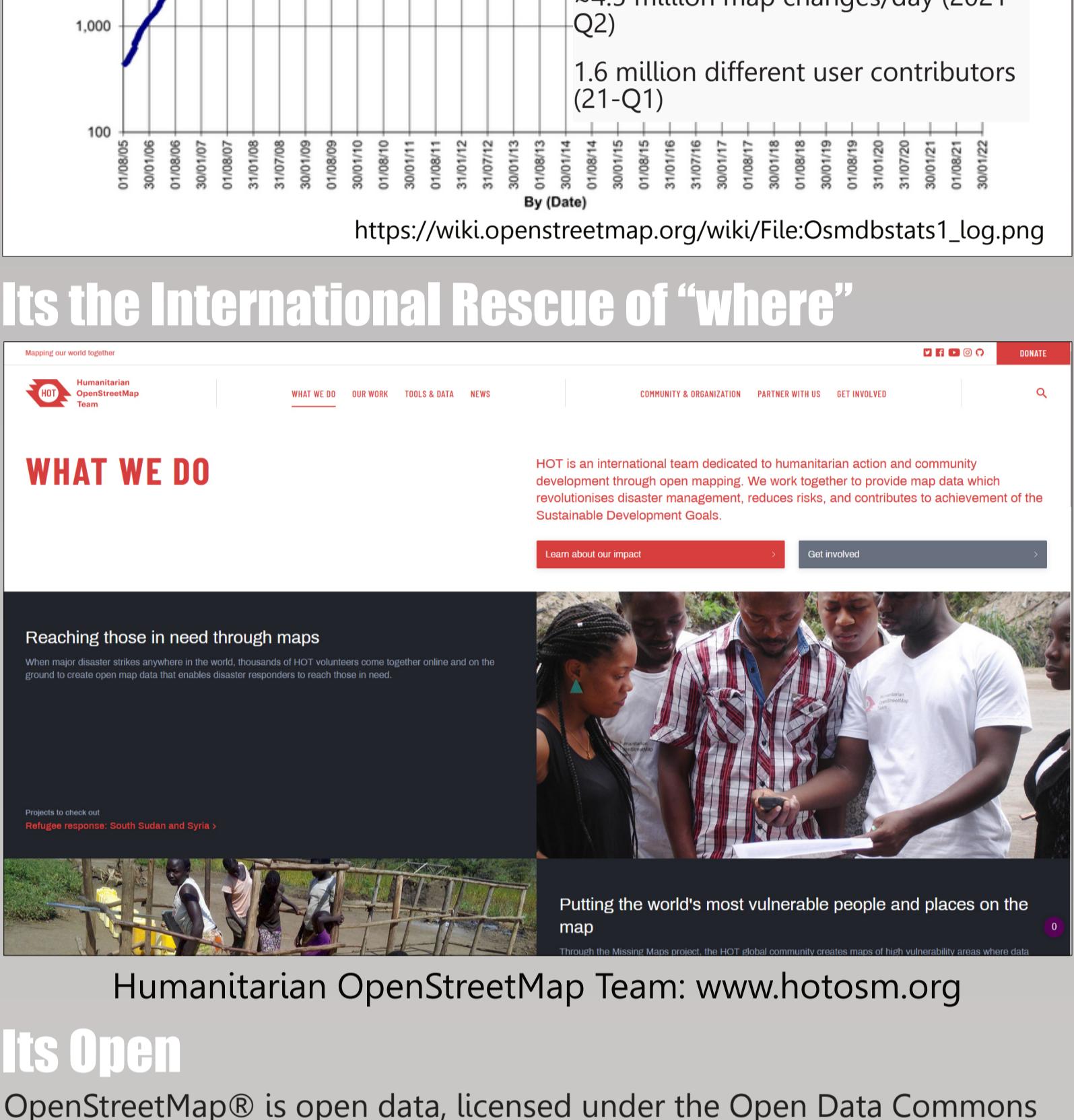
A LOVE LETTER TO OPENSTREETMAP

www.openstreetmap.org

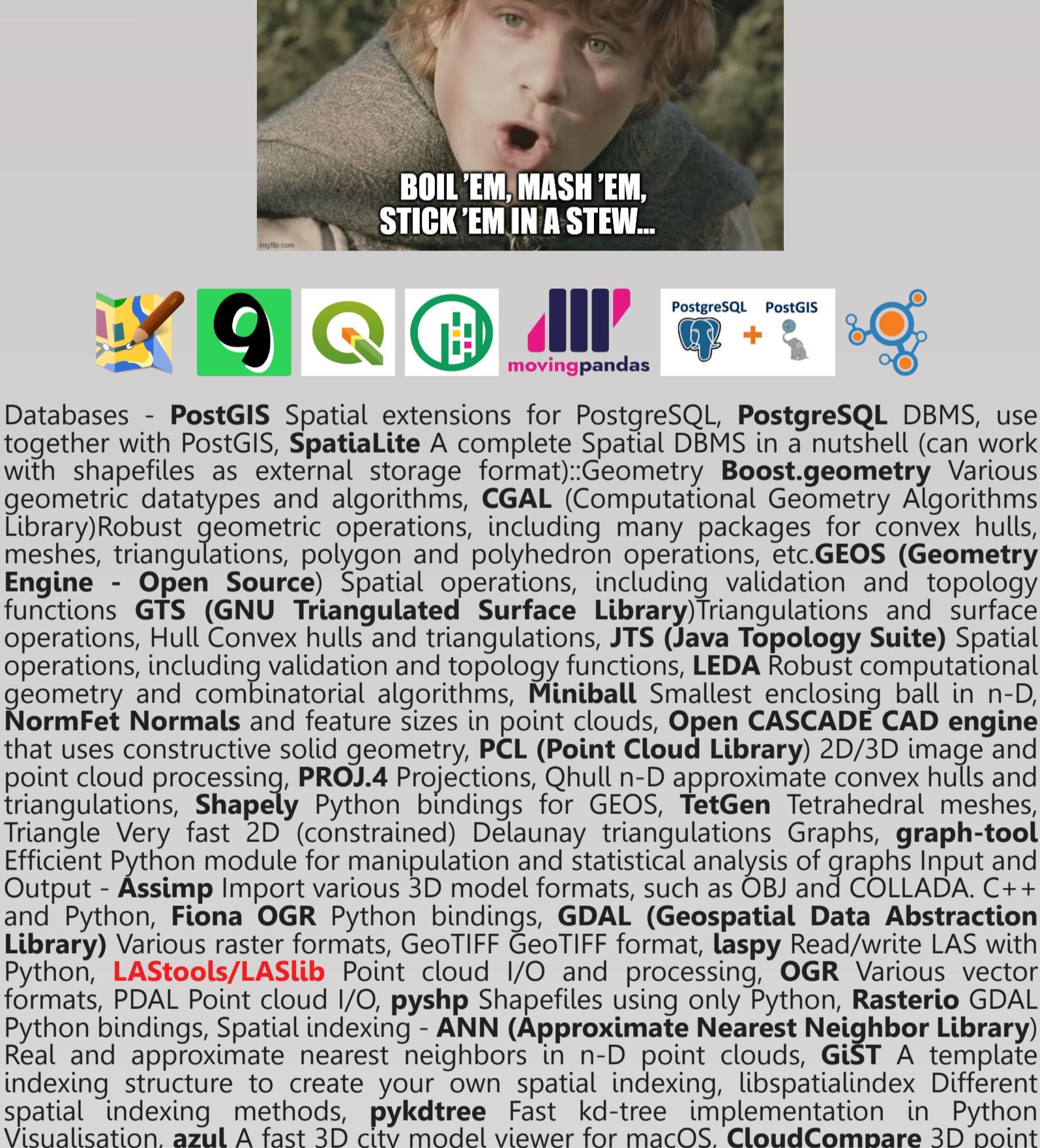
Its a map



Its a database



Its a community



Its the International Rescue of “where”



Its Open

OpenStreetMap® is open data, licensed under the Open Data Commons Open Database License (ODbL) by the OpenStreetMap Foundation (OSMF).

You are free to copy, distribute, transmit and adapt our data, as long as you credit OpenStreetMap and its contributors. If you alter or build upon our data, you may distribute the result only under the same licence. The full legal code explains your rights and responsibilities.

But, its so much more



Databases - **PostGIS** Spatial extensions for PostgreSQL, **PostgreSQL** DBMS, use together with PostGIS, **SpatialLite** A complete Spatial DBMS in a nutshell (can work with shapefiles as external storage format)::Geometry **Boost.geometry** Various geometric datatypes and algorithms, **CGAL** (Computational Geometry Algorithms Library) Robust geometric operations, including many packages for convex hulls, meshes, triangulations, polygon and polyhedron operations, etc.**GEOS (Geometry Engine - Open Source)** Spatial operations, including validation and topology functions **GTS (GNU Triangulated Surface Library)** Triangulations and surface operations, Hull Convex hulls and triangulations, **JTS (Java Topology Suite)** Spatial operations, including validation and topology functions, **TetGen** Tetrahedral meshes, **Miniball** Smallest enclosing ball in n-D, **NormFet Normals** and feature sizes in point clouds, **Open CASCADE CAD engine** that uses constructive solid geometry, **PCL (Point Cloud Library)** 2D/3D image and point cloud processing, **PROJ.4** Projections, Qhull n-D approximate convex hulls and triangulations, **Shapely** Python bindings for GEOS, **TetGen** Tetrahedral meshes, **Triangle** Very fast 2D (constrained) Delaunay triangulations Graphs, **graph-tool** Efficient Python module for manipulation and statistical analysis of graphs Input and Output - **Assimp** Import various 3D model formats, such as OBJ and COLLADA. C++ and Python, **Fiona** **OGR** Python bindings, **GDAL (Geospatial Data Abstraction Library)** Various raster formats, GeoTIFF GeoTIFF format, **laspy** Read/write LAS with Python, **LASTools/LASlib** Point cloud I/O and processing, **OGR** Various vector formats, PDAL Point cloud I/O, **pyshp** Shapefiles using only Python, **Rasterio** GDAL Python bindings, Spatial indexing - **ANN (Approximate Nearest Neighbor Library)** Real and approximate nearest neighbors in n-D point clouds, **GIST** A template indexing structure to create your own spatial indexing, **libspatialindex** Different spatial indexing methods, **pykdtree** Fast kd-tree implementation in Python Visualisation, **azul** A fast 3D city model viewer for macOS, **D3.js** Framework to create interactive maps and more, **GeoServer** Publish web maps, **GRASS GIS** package, **LidarViewer** Interactive visualisation of massive point clouds, **mapnik Toolkit** for map applications, **MapServer** Publish web maps, **MeshLab** Process and view triangular meshes and point clouds, **ParaView** Visualiser for large data sets, **plasio** Web-based LAS/LAZ point cloud viewer, **Potree** Web-based viewer for massive point clouds, **QGIS (Quantum GIS)**, GIS package, **TileMill** Simple map creator, **VTK (Visualization Toolkit)** Software that implements a wide variety of visualisation algorithms.

To name but a few...

Not what, can you do? but what will you do?

www.hotosm.org

www.humanitarian.org

www.osmfoundation.org

www.openstreetmap.org

www.osmchina.org

www.osm.org

www.osm.org