Small, static datasets

Traditional GIS

- **Data mapping static assets**: Natural resource assets, land parcels, transportation grid, utilities

- **Human-curated**: Significant time spent precisely measuring and mapping the location of these assets

- **Nature of data limits scale**: Hand-curation means data size measured in thousands, not billions
Traditional GIS

• **Emphasis on geospatial operations:** Which light poles are within 5m of a road? How many land parcels lie in a flood zone?

• **Limited number of GIS experts using data:** Technical nature of data and problem domains in combo with specialized tools limit access to a small group of trained experts.

• **Use of dedicated GIS platforms:** Only a small set of tools (i.e. ESRI) capable of depth of GIS operations needed.
New Geoanalytic Challenges…
New Geoanalytic Challenges

- Data is primarily sensor-generated: Cell handsets, vehicle telemetry, LIDAR, satellite, social media
- Data size is unbounded, no longer limited by the rate at which humans can create it.
- Data velocity can be extremely high, with billions of sources each reporting multiple times per second.

Large, high velocity data
New Geoanalytic Challenges

• Geo is just one attribute of data: Other attributes can be equally as important

• Lower complexity of geospatial analytics: Often limited to visualization and measuring distance and spatial overlap

• New audience of analysts and data scientists: Deep GIS background often no longer required

Geolocation in context
...Require New Spatially Enriched Analytics

COMMERCIAL USES
- Weather Mapping
- General Purpose Analytics
- Network Anomalies Analytics
- Smart Meter Analytics
- Fleet Telematics
- Well Log Analytics
- Field Service Management

FEDERAL USES
- Resource Management
- Agile Geoanalytics
- Command & Control
- Intelligence & LE
- Cyber Security
- Disaster Response
- Health & Pandemics

What are some of the top use cases?
Breaking Point

General Purpose Analytics

Agile Geo-analytics

Large, High Velocity Data

Traditional GIS

Traditional BI
“There’s no way to render everything at once.”

“I downsample for a living.”

“Wait for it….”

“Latency is crushing us. These insights are too old to be useful.”

“I have to write a support ticket and wait days for a new dashboard.”
MapD for Modern Geoanalytics

Leverages massive parallelism of GPUs

Query billions of rows in milliseconds

Render & interact with your data
How We Do It

Blindingly Fast
- Parallelize SQL for GPUs
- Three-tier data caching
- LLVM Compilation Engine

Native Rendering
Vega API for declarative rendering of data in-situ on the GPU

Intuitive to Use
MapD Immerse provides drag-and-drop interactive visualization at scale
Core Density Makes a Huge Difference

CPU Processing

- 20 Cores

GPU Processing

- 39,000+ Cores

©MapD 2017
Keeping Data Close to Compute

MapD maximizes performance by optimizing memory use

**Hot Data**
- Speedup = 1500x to 5000x
- Over Cold Data

**Warm Data**
- Speedup = 35x to 120x
- Over Cold Data

**Cold Data**

**GPU RAM (L1)**
- 24GB to 256GB
- 1000-6000 GB/sec

**CPU RAM (L2)**
- 32GB to 3TB
- 70-120 GB/sec

**SSD or NVRAM STORAGE (L3)**
- 250GB to 20TB
- 1-2 GB/sec

**Data Lake/Data Warehouse/System Of Record**

©MapD 2017
Query Compilation with LLVM

Traditional DBs can be highly inefficient
- each operator in SQL treated as a separate function
- incurs tremendous overhead and prevents vectorization

MapD compiles queries w/LLVM to create one custom function
- Queries run at speeds approaching hand-written functions
- LLVM enables generic targeting of different architectures (GPUs, X86, ARM, etc).
- Code can be generated to run query on CPU and GPU simultaneously
Blogger Mark Litwintschik benchmarked MapD on a billion-row taxi data set and found it to be **orders-of-magnitude faster than the fastest CPU databases**.
Server-Side Rendering

Ultra-fast CUDA->OpenGL->PNG pipeline for declarative rendering
MapD Immerse
Using a hybrid approach to interactive visualization at scale

- Basic charts are frontend rendered using D3, leveraging fast SQL of MapD Core
- Scatterplots, pointmaps + polygons are backend rendered using the Rendering Engine on GPUs
- Geo-Viz is composited over a frontend rendered basemap
Three Ways to Get Started

**OPEN-SOURCE**
GitHub Download

**COMMUNITY**
Website Download
AWS Cloud

**ENTERPRISE**
Contact MapD Sales
AWS Cloud
About MapD

- Originated from MIT
- $37 Million in funding
- Used By 100+ Global Orgs
- Open Source Community