**Motivation**

Database vendors need access to client data for:
- Testing their engines to resolve client problems.
- Assessing impacts of a planned engine upgrade.
However, client data is often **unavailable**!
- Privacy concerns
- Transfer cost (especially at Big Data Scale)

**Objective**

Dynamically generate client’s representative data during query execution while ensuring similar performance on original and synthetic data

**Mechanism**

**Volumetric Similarity**: When a matching client plan is executed at the vendor, the number of output rows for each operator should be similar.

---

**Table: Database Summary**

<table>
<thead>
<tr>
<th>Database</th>
<th>DataSynth</th>
<th>Hydra</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size (in GB)</td>
<td>10</td>
<td>4 hours</td>
</tr>
<tr>
<td>Materialization Time</td>
<td>100</td>
<td>42 hours</td>
</tr>
<tr>
<td>Materialization Time</td>
<td>1000</td>
<td>&gt; 1 week</td>
</tr>
</tbody>
</table>

---

**Figure: Extended Workload Coverage**

- Grid Partitioning vs. Region Partitioning
- Dynamic Data Generation
- Materialization Time

---

**Figure: Experimental Evaluation**

- Workload Complexity
- LP Processing Time
- Volumetric Similarity

---

**Figure: Hydra Architecture**

- Input AQPs
- CODD (Fetch Metadata)
- Preprocess (DataSynth)
- LP Formulator
- Summary Generator

---

**Figure: Hydra Contributions**

- Workload Complexity
- Grid Partitioning vs. Region Partitioning
- Dynamic Data Generation
- Project Website: dsl.cds.iisc.ac.in/projects/HYDRA