Some practical considerations

Impala Use Cases

- Allow us to query new types of data that do not yet have some established ETL process.
- Report generation and quick one-off analysis and prototypes.
- Monitoring & alert systems with hourly refreshed data.

Benefits

- SQL dialect, familiar to data scientists / analysts.
- **Main use**: Ad-hoc queries on data, retrieving results quickly.
- Compatible with HBase and HDFS.
- Shared metastore with Hive, easy portability.
- **Not a use case:** Stream processing (industrial environments with sensors).

Comparison with Hive

- **Hive:** SQL-based queries using MapReduce.
  - Takes time to be ready to work (cold start)
  - Materializes intermediate results, which helps for scalability and fault tolerance.
  - This makes Hive more suitable for ETL jobs or any type of process that takes hours (because you don't want to run *again* that query).

Comparison with Hive (cont.)

- **Impala:** SQL-based queries using custom execution engine.
    - Warm start. `impalad` is already running on nodes.
    - Intermediate results are streamed between executors.
    - No fault tolerance: If a node fails in the middle of the query, the query will likely have to be aborted and reissued.
    - Intermediate results are processed in memory, so one has to have enough of it (or pay the price with disk spilling).
    - Metadata needs to be refreshed (not necessary in Hive).

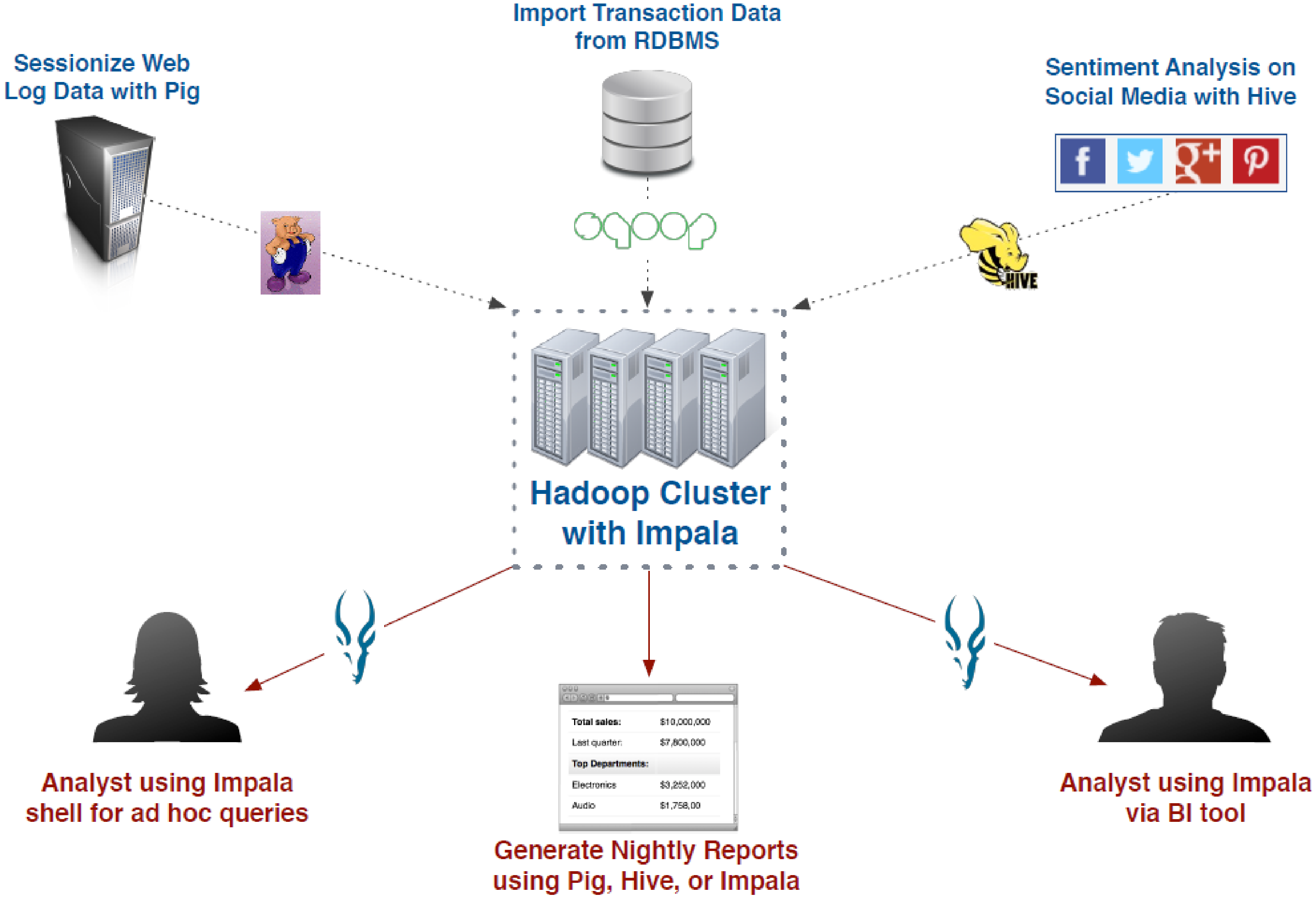
Do I need to throw away my RDBMS?

- **Relational databases** are optimized for:
  - Relatively small amounts of data.
  - Immediate results.
  - In-place modification of data (`UPDATE` and `DELETE`).
- **Hive and Impala** are optimized for:
  - Large amounts of read-only data.
  - Extensive scalability at low-cost.
- Hive is better suited for batch processing, Impala and relational databases for interactive use.

Hive vs Impala vs RDBMS

- Hive gives you productivity but not speed.
- Relational databases give you speed but not scalability.
- Impala gives you scalability and speed, but less control.

Analytics Workflow



Sessionize Web
Log Data with Pig

Import Transaction Data
from RDBMS

Sentiment Analysis on
Social Media with Hive

Hadoop Cluster
with Impala

Analyst using Impala
shell for ad hoc queries

Total sales:	$10,000,000
Last quarter:	$7,800,000
Top Departments:	
Electronics	$3,252,000
Audio	$1,758,00

Generate Nightly Reports
using Pig, Hive, or Impala

Analyst using Impala
via BI tool

More differences between Impala and Hive

Impala lacks some functions that Hive has:

- No support for `BINARY` nor `DATE` data types.
- No support for XML functions.
- Timestamps are stored as UTC (not as locale zones).
- No implicit casting between string and numeric/boolean types.
- SQL differences between Impala and Hive.

Bridging the gaps

Hive LLAP

- LLAP (Live Long And Process) functionality in Hive 2.0.+
- Hybrid execution model relying on a daemon process that stays alive.
- Small/short queries are largely processed by this daemon directly, while any heavy lifting will be performed in standard YARN containers.

Comparison between Hive LLAP and Impala

Impala	Hive LLAP
Data mart	Enterprise data warehouse
• Good choice for interactive and ad-hoc analysis, especially with high concurrency self-service	• Good choice for long-running queries requiring heavy transformations or multiple joins • Good choice for interactive and ad-hoc analysis using features not available in Impala
• Good choice for Business Intelligence tools that allow users to quickly change queries	• Good choice for Dashboards that are pre-defined and not customizable by the viewer
• Uses Parquet as the preferred file format	• Uses ORC as the preferred file format • Does better with JSON than Impala does

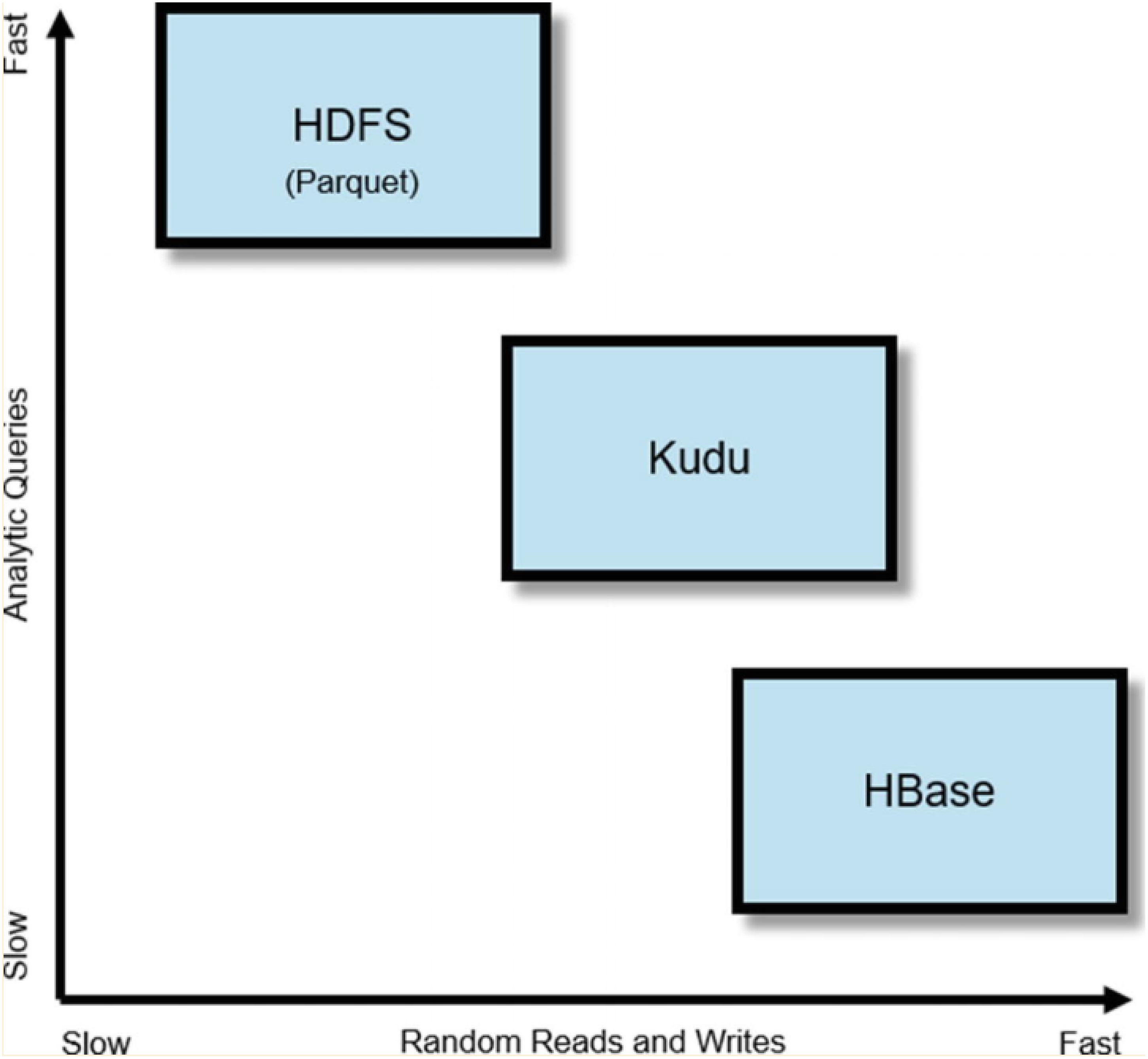
Query performance: Hive LLAP and Impala

- Cloudera Blog: Query performance comparison
- Hive Testbench

Apache Kudu

- Columnar storage engine for structured data.
- Used in combination with Impala (since Kudu has no querying engine) for relational data management and analytics.
- **Use case:** IoT and time series applications, with real time data ingestion, visualization and complex event processing.

Apache Kudu (cont.)



References

- Cloudera Blog: Hive LLAP vs Impala
- Getting Started with Impala
- Next-Generation Big Data

The End

NobleProg Evaluation Survey