Data Virtualization Technology and its Application to DB2 on z/OS

Tim Willging, Rocket Software
Anthony Ciabattoni, Rocket Software
What is Data Virtualization?

Mainframe resident data virtualization solution for real time, universal access to data, regardless of location or format.
Data Virtualization Use Cases

- Makes data more accessible
  - Lowers barriers for customers to access data

- Provides modern access to legacy data
  - SQL or NOSQL access to data…in place

- Allows data to be integrated easily into other tools
  - JDBC access to virtualized data
  - Integrate into data movement tools
  - BI Tools
  - SQL Editors

- Useful for:
  - Dashboards
  - Analytics (BI Tools)
  - Websites
  - Data Consolidation
  - Mobile applications
Data Virtualization Components

- **Data Access Layer**
  - Provides access to supported data sources
  - Specialized access to mainframe data sources
    - VSAM, Flat Files, IMS, ADABAS, SMF, etc
  - Link to relational data sources

- **Data Catalog**
  - Repository of data sources
  - Information required to access each data source
  - Virtual Tables/Virtual Views

- **SQL Subsystem / Engine**
  - Processes standard SQL
  - Redirects SQL to underlying data sources
  - Calls Data Catalog and Data Access Layer
  - Ability to join data from different data sources
  - Returns result set in standard SQLDA format
How We Optimize Mainframe TCO

- Mainframes have multiple processors
- DV has patented technology that allows it to run 99% of its own processing in the zIIP engine
Architecture built to leverage zIIP
Extract, Virtualize, and Load
Optimizations for Data Virtualization

- zEDC
- SMT2
- SIMD
- Metal C optimized Server specific to z System version
- Pageable Large Frames
- SMC-R
- zIIP
- Parallelism – Intra-SQL and Intra-Partition
- Shared Memory Object based JNI interface
More Optimizations!

- Exploits AT-TLS, because we use it, we inherit all of the new crypto/encryption advances
- Flash Express exploitation for reduced Dynamic Address Translation (DAT) overhead. Exploited for all above the bar Private and Shared Memory Objects (buffer pools, Metal C heap, data areas)
- VSAM – new SRB mode support (not ICI based)
- Log Streams - Future
- Exploitation of 64 bit storage, Shared Memory Objects and more importantly z Flash Express to take advantage of Pageable Large Pages (reducing DAT code path)
- MapReduce (reading different sections of the same dataset in parallel then aggregating the Virtualization engine
- Parallel I/O, only keeping a file open for milliseconds, network and file I/O is done in parallel
- Full intra SQL and intra-partition parallelism
How does it work????
Virtualization / Mapping Process

- Specify location and name of “copy book”
- Identify data source attributes
  - Data Set Name
  - IMS DBD, PSB
  - ADABAS Attributes
  - Relation Data Source (DRDA attributes, Table name, View name)
- Enter name for “Virtual Table”
- Fields of copy book end up being column names
- Virtualization provides SQL (or NOSQL) access to Virtual Table
- Graphical interface provided for mapping process
- Saved in metadata repository
DB2 Analytics Accelerator Loader for z/OS

Extend analytic capabilities by bringing non-DB2 data to IBM Analytics Accelerator and z Systems

- **In-memory transformation (ETL) of non-DB2 data.** Significant cost and time reduction by eliminating manual processes.

- **Insight into more data types such as IMS, VSAM, sequential files, Oracle, Adabas, SMF, etc.**

- Assists in **data availability** by loading to multiple accelerators in parallel

- **Direct load** of Accelerator-only tables (AOTs) for greater **savings**

- **Load Resume** process adds data – avoid reloading entire table

- **Support for consolidating changes made to replicated table(s)**
IBM DB2 Analytics Accelerator Loader V2.1

- Extends capability of Accelerator beyond DB2 data
- Addresses challenges identified in loading non DB2 data
  - Manual – Labor intensive Extract and Transform
  - Slow – due to staging data to disk

**Additional Features**

- Embedded capability to “virtualize” and load many types of data to accelerator
- Views to load 100s of SMF
- Ability to load Syslog data
- Load Resume
- Historical data load
DB2 Analytics Accelerator Loader

Automates loading of non-DB2 data

Entire process is automated in one job:

- Creates DB2 table
- Table added to Accelerator
- Source data extracted
- In-memory conversion to DB2 format
- Data loaded to Accelerator
- Table is enabled for acceleration
In Memory Data Transformation

IBM DB2 Analytics Accelerator

Batch DSNUTILB
Accelerator Loader
Source SQL Statement

SQL Result Set
Accelerator Loader Server

IBM z
Shared memory objects
Data I/O
Hybrid SRB/TCB

Virtual tables
Mappings

Relational and Non-relational data

DRDA Sources (Oracle)
IMS DB
Sequential
VSAM
Remote DB2
DB2 Analytics Accelerator Loader Infrastructure

Inter Server Communication

- JDBC
- DRDA

APAR - PI65565

Oracle

Federation Server

DB2 LUW

Solutions
Loading non-DB2 Data Sources into Accelerator

Solution Comparison

- Native Solution without Tooling
  - Multi-step manual solution

- Accelerator Loader V1.1
  - Partially automated solution

- Accelerator Loader V2.1—Automates entire process:
  - User builds a select statement from data source(s) (IMS, VSAM, Oracle, …)
  - Automatically creates the DB2 table
  - Automatically adds table to Accelerator
  - Automatically extracts specified source data
  - Automatically converts data to necessary DB2 format (in memory)
  - Automatically loads data to Accelerator
  - Automatically enables table for acceleration

Single batch job!
Accelerator Loader High Availability Load

- Loads to multiple Accelerators are performed in parallel!
- High Availability Load One Single Load Operation

1. Reduced Elapsed Time
2. Reduced CPU
3. Simplified Operations
4. Batch ALT syntax
5. Additional options
Accelerator Loader Benefits

- LOAD RESUME support
  - Avoid the need for full LOAD replace
- Accelerator-only tables (AOTs) Support
- Support for tables in replication mode when loading from a file
- Usability and control features
  - Optionally performance Accelerator table functions
  - Parallelism options
- Performance enhancements
  - Improved:
    - Elapsed Time
    - CPU Time
    - zIIP Offload
- Improved ISPF interface
Lab Test Results Loading VSAM Data

**VSAM KSDS 200 GB Data**

---

**The Old vs The New**

200GB VSAM KSDS
z13 LPAR: 2 CPs, 6 zILPs w/SMT
ACCEL_LOAD_TASKS = 16

---

**Minutes**

<table>
<thead>
<tr>
<th></th>
<th>Traditional Approach</th>
<th>Loader V1.1 function</th>
<th>Loader V2.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extract + DB2 Load +</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accelerator Load</td>
<td>Elapsed Time</td>
<td>GCP Time</td>
<td>zIIP time</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>21</td>
<td>11</td>
</tr>
<tr>
<td>Extract + Loader</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>External Load</td>
<td>189</td>
<td>128</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>21</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>15</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>11</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>

---

**Simpler & more easily maintainable**
- Leverage SQL to extract the non-relational source
- Extraction, Conversion, Load in one step. Data never lands

**Significant reduction in GCP requirements**
- Reduces software pricing MSUs. Frees up costly GCP capacity.

**Major reduction in time. 3+ hrs => 11 mins**
- Reduction in ETL / Batch windows.
- Increases time/processing available for more advanced analytics

**Excellent exploitation of z Systems’ cost-effective technologies**
- zILP-eligibility: We witnessed up to 90%
- z13 SMT: Witnessed up to 50% additional zIIP capacity

---

*Extract: A bare bones COBOL program was written/utilized for the base Extract*
Expanding the Analytic Accelerator Opportunities

New Accelerator Loader V2.1

Facilitates loading data to the Accelerator
Bring your analytics to your data – save data replication costs
Operational Analytics

- SMF Records
- RMF Data
- Syslog
- Audit Data
- Custom Metrics
- DB2 Logs
- Websphere Logs
- CICS Logs
- SQL Performance
- Transactional Performance
Operational Analytics

- User/Usage Analysis
- Peak Time Analysis
- Forecasting & Planning
- Trend Analysis
- Security & Forensics
- Capacity Planning
- Proactive Alerting
- Root Cause Analysis
- Detailed Historical Analysis
- Root Cause Analysis
- DB2 Analytics Accelerator for z/OS
Large Financial Data Virtualization Use Case

- **IDAA placed via a Solution**
  - Never have found a viable analytics workload to run on this appliance
- **Global Clearing Management System**
  - 7 clearing cycles per day for clearing funds between Issuers & Acquirers
  - Interchange Adjustment (IA) process
  - Attended an IDAA PoC which started the discussion of using IDAA for IA
- **IA is not an analytics workload, but does fit a Big Data use case**
  - Mostly manual process today
  - Each case can bring unique requirement
  - Can take weeks
- **The solution**
  - Load the payment transactions from each clearing cycle into IDAA (back loading 2 years)
  - When IAs come up SQL will be used to identify and stage transactions for adjustment
    - COBOL is written today to read from VTS
  - This will then flow through the IA process
GCMS/IDAA PoC Challenge / Response

- The foundation of this process are ISO-8583 records
  - This is a very loose standard with a bit map identifying groups / subgroups of data
- This data was considered too voluminous to maintain online in transactional systems
- The IDAA platform (along with it’s compression capabilities)
  - Offered the storage at an acceptable price point
  - Provided a platform for potentially intense queries that would not interfere with the transactional environment
  - GCMS is a traditional CICS / COBOL DB2 application today, so this is natural
- The IDAA Loader (Data Virtualization) provides for:
  - Loading the appliance without having to load the data into DB2
    - Can provide parallelism for this load (even from tape)
  - Has plan to load multiple accelerators in parallel
  - Has virtualization capabilities to federate many non-DB2 sources
  - Mapping exits / Rocket consulting to develop mapping routines for complex formats
  - Provides LOAD RESUME capabilities (possible future use)
Accelerator Data Consolidation

1. Relational
2. IMS
3. VSAM
4. SMF
5. Sequential

One Consolidated Access Point

Expanded Use Cases
Simplicity
Additional Accelerator Customer Use Cases

- Customizable Virtual Views
  - Loads the result set of the Virtual View into the Accelerator

- Data Validation
  - Data validation tends to be costly
  - Efficiently exploiting Accelerator to validate different versions of the data
QMF for z/OS – an enterprise business analytics suite

- Access all forms of enterprise data on any device
- Add more data sources or users without increasing price
- Addresses all user types and BI needs – No need for separate BI tools for specific data sources or user profiles

- True “data federation” - combine data from multiple structured data sources with one SQL statement
- Combine data from structured and unstructured data
- QMF is zIIP eligible, lowering mainframe CPU consumption

QMF users gain direct access to additional data sources: VSAM, sequential files, ADABAS, and more
QMF users can create their own visualizations and collaborate with other users real-time – truly self service BI!
QMF 11.2.1 with Data Virtualization

- Allows QMF to access non-relational data as if it were relational
- Same ‘mapping’ process as in Accelerator Loader
  - Maps can be shared across tools, but each has its own address space
- VSAM, Flat Files, Analysis of SMF data, IMS, ADABAS, etc.
- Exploits a z/OS server for query processing
  - Can be more performant than using JDBC driver to IMS and joining with DB2
  - With the virtualization technology, the data joining happens in z/OS server address space, not client side
- May not be appropriate where query is executed often and fast result set is needed when accessing large amounts of data
- Complimentary to Accelerator and Loader solution
  - If repeated access and/or high speed results required, load data to IDAA first
What Is Spark?

- An Apache Foundation open source project. Not a product.
- An in-memory compute engine that works with data. Not a data store.
- Enables highly iterative analysis on large volumes of data at scale.
- Unified environment for data scientists, developers, and data engineers.
- Radically simplifies process of developing intelligent apps fueled by data.

Why Spark?

- Spark is open so accelerates community innovation
- Spark is fast 100x faster than Hadoop MapReduce
- Spark is about all data for large scale data processing
- Spark supports agile data science to iterate rapidly
- Spark can be integrated with IBM solutions
Apache Spark is…

Fast
- Leverages aggressively cached in-memory distributed computing and JVM threads
- Faster than MapReduce for most workloads

Ease of use (for programmers)
- Written in Scala, an object-oriented, functional programming language
- Scala, Python and Java APIs
- Scala and Python interactive shells
- Runs standalone, on Hadoop, Mesos, or cloud

General purpose
- Covers a wide range of workloads
- Provides SQL, streaming, machine learning, graphing

from http://spark.apache.org
Apache Spark – How it is being used

- **Spark Streaming & SQL**
  - Streaming ETL - Very good at transforming data.
  - Data Enrichment – combine streaming data with static data
  - Trigger event detection – Fraud detection

- **Spark Machine Learning**
  - Predictions
  - Sentiment analysis
  - Recommendations

- **Uber**
  - Built ETL pipeline of customer data, traffic data, ride requests, etc
  - Used for pricing, marketing, identifying growth areas

- **Pinterest & Netflix**
  - Make relevant recommendations to users
Apache Spark Use Cases

- **Customer Data**
  - Fraud Detection
  - Customer retention
  - Suggestive selling

- **DB2 Tooling Possibilities**
  - Operational Insights
    - SMF, RMF, Audit data, DB2 Performance Tools, etc
  - Anomaly Detection
    - DB2 Log processing
    - DB2 Audit logs
  - Prediction
    - DB2 Performance tuning
    - Utility run time execution
IBM z/OS Platform for Apache Spark, V1.1

z Systems & Apache Spark – A unified analytics platform

Key Business Transaction Systems

Spark Applications: IBM and Other Commercial Vendors

Spark SQL  Spark Stream  MLib  GraphX

Apache Spark Core

RDD  RDD  RDD  RDD

Rocket Data Virtualization Technology

DB2 z/OS  IMS  VSAM  SMF  and more…

Oracle  HDFS

Type 2 / Type 4  Type 4
z Systems & Apache Spark: Strategic Direction

Fast, expressive, cluster computing system leveraging in-memory framework for analytics

Leverage LoZ virtualization benefits

Leverage z/OS data and transactions

Federation across platform Spark implementations will initially need external orchestration
Spark z/OS & IDAA -- Complementary

- IDAA and Spark are complementary solutions. What makes sense to use depends on what needs to be done.
  - Spark SQL that is initiated which qualifies for acceleration will be accelerated on the IDAA transparently to the Spark application
  - Spark does not do query acceleration; IDAA is primarily a query accelerator
  - Spark is not a data store; IDAA can be a data store
  - IDAA interaction is via SQL interfaces to DB2 z/OS; Spark interaction is via Spark jobs written in scala, python, java ---and there are capabilities to interact via machine learning and graph analytics, as well as SQL
  - Spark can access data-in-place, real-time across a variety of environments without moving the data; IDAA data is largely from DB2, other data can be loaded into IDAA.