IBM Software Group

Data Analytics using MapReduce framework for DB2's Large Scale XML Data Processing

George Wang

Lead Software Egnineer, DB2 for z/OS

IBM

DB2 Information Management Software

© 2014 IBM Corporation

Disclaimer and Trademarks

Information contained in this material has not been submitted to any formal IBM review and is distributed on "as is" basis without any warranty either expressed or implied. Measurements data have been obtained in laboratory environment. Information in this presentation about IBM's future plans reflect current thinking and is subject to change at IBM's business discretion. You should not rely on such information to make business plans. The use of this information is a customer responsibility.

IBM MAY HAVE PATENTS OR PENDING PATENT APPLICATIONS COVERING SUBJECT MATTER IN THIS DOCUMENT. THE FURNISHING OF THIS DOCUMENT DOES NOT IMPLY GIVING LICENSE TO THESE PATENTS.

TRADEMARKS: THE FOLLOWING TERMS ARE TRADEMARKS OR ® REGISTERED TRADEMARKS OF THE IBM CORPORATION IN THE UNITED STATES AND/OR OTHER COUNTRIES: AIX, AS/400, DATABASE 2, DB2, e-business logo, Enterprise Storage Server, ESCON, FICON, OS/390, OS/400, ES/9000, MVS/ESA, Netfinity, RISC, RISC SYSTEM/6000, System i, System p, System x, System z, IBM, Lotus, NOTES, WebSphere, z/Architecture, z/OS, zSeries

The FOLLOWING TERMS ARE TRADEMARKS OR REGISTERED TRADEMARKS OF THE MICROSOFT CORPORATION IN THE UNITED STATES AND/OR OTHER COUNTRIES: MICROSOFT, WINDOWS, WINDOWS NT, ODBC, WINDOWS 95, WINDOWS VISTA, WINDOWS 7

For additional information see ibm.com/legal/copytrade.phtml

Agenda

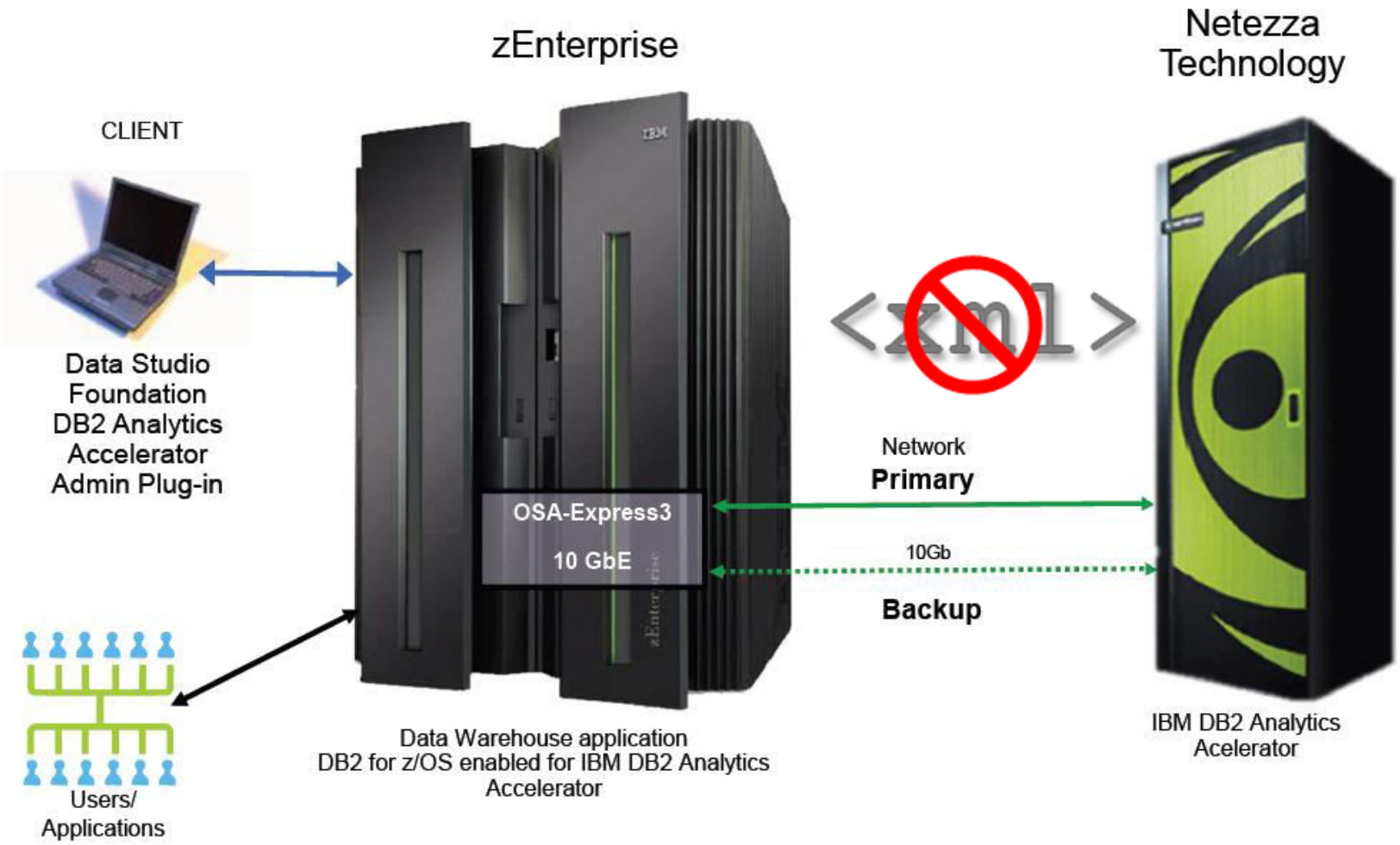
- Motivation
- Project Overview
- Architecture and Requirements
  - Technical design problems
  - Hardware/software constraints, and solutions
  - System Design and Implementation
- Performance and Benchmark showcase
- Conclusion, Recommendations and Future Work

IBM's Big Data Portfolio

IBM views Big Data at the enterprise level thus we aren't honing in on one aspect such as analysis of social media or federated data

1. Data Warehouse (Information Server, DB2 Analytics Accelerator, Netezza, etc.)
2. InfoSphere BigInsights (Hadoop etc.)
3. Stream data capture and analysis
4. Federated data discovery and analysis

IBM DB2 Analytics Accelerator

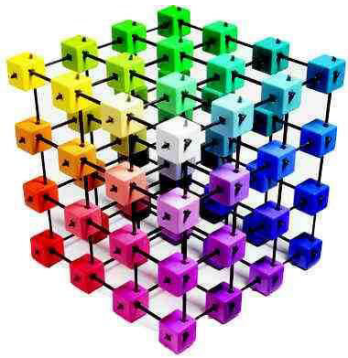


XML Database

- XML is known to be a promising and desirable data format for storing and modeling data
- XML database offers the ability to store data and documents without requiring a database schema
- XQuery scripting language allows an expression or predicate to be used to process XML data. It's built on XPath expression
- XML data can be manipulated using XQuery script language with increasing demand

Use case

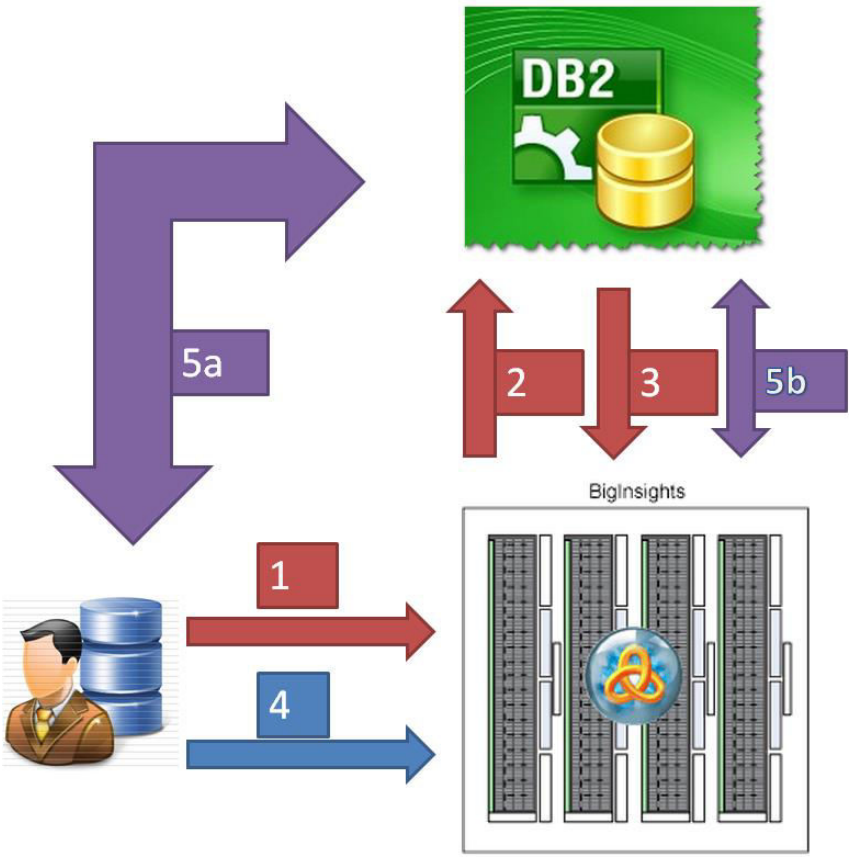
- A DB2 client is to query over 400 TB of tax payers' profile information formatted in XML data representation using XML query technology.
- Requirement: interactively analyze XML data in real time
- Problems:
  - No such technology to analyze XML on HDFS
  - Large scale data offloading process is a performance problem
  - No backend support for importing data from Hadoop by DB2 application in XML format
- Summary:
  - The lack of analytical query processing technology in Big Data restricts DB2 clients from using OLAP application on XML data



Project Overview

- Build an interface for RDBMS and Big Data
  - Allow customers to move operational data in XML from System z for integration with other data
- Enable Online Analytical Processing(OLAP) applications with XML data in DB2 using XQuery technology
  - Invest business value in building a cloud-enabled framework to allow machines to process data analytics in XML representation using XQuery support
- Explore a Big Data appliance on InfoSphere BigInsights with System z
- Meet the demand by DB2 customers with new workloads to System z
  - Use the gravitational pull of its transactional data control

Project Architecture



1. User issues a SQL command to query on DB2 table to populate BigInsights' Hadoop
2. BigInsights queries XML table in DB2 database
3. Loading DB2's XML data on Hadoop
4. User issues a Jaql's XPath query on XML data. Result of the query is stored on HDFS

5a. User runs a DB2 UDF to retrieve result of XPath's query from HDFS back to DB2 database

5b. DB2 requests and stores XPath result from HDFS

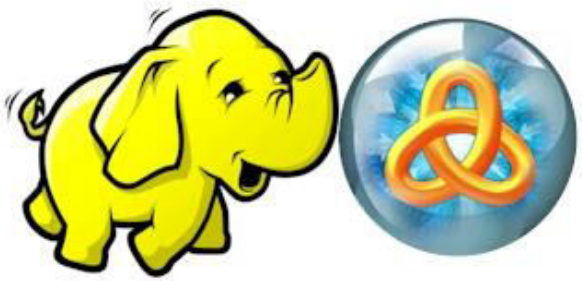
Technical Design Problem

- **BigInsights**
  - provides data analyzing capabilities in large volumes of data

DB2 for z/OS

- Provides both XML and XQuery supports

But…. they don't talk to each other on z/OS!



Plus…

- DB2 performs XML analytics slowly on z platform
- BigInsights does not support XQuery

Environment Requirements

- **Hardware Requirement**
  - One z server machine for storing XML data with DB2 for z/OS
  - Linux machines with RHEL 6.2 as a Hadoop server for BigInsights
    - 40GB of Disk storage
    - 8 GB of memory
    - Minimum of 4 Nodes in Cluster installation
    - x84 64-bit systems
    - zSystems (z/OS, z/VM and zLinux) are incompatible for deployment at this moment
  - DB2_BigXML VLAN for Traffic Flow Survey

- **Software Requirements**
  - IBM InfoSphere BigInsights Enterprise Edition 2.0
  - DB2 for z/OS 10.1
  - Mozilla Firefox 10.0.
  - Eclipse IDE 3.6.2 for Java™ EE
  - Runtimes for Java Technology, Version 6.0.11.0

Hardware constraints and solutions

- Physical cluster allocated for Linux must not drive more than 80GB of day-to-day data traffic flow with z/OS network
  - Prevent network jam in subnet system within intranet network

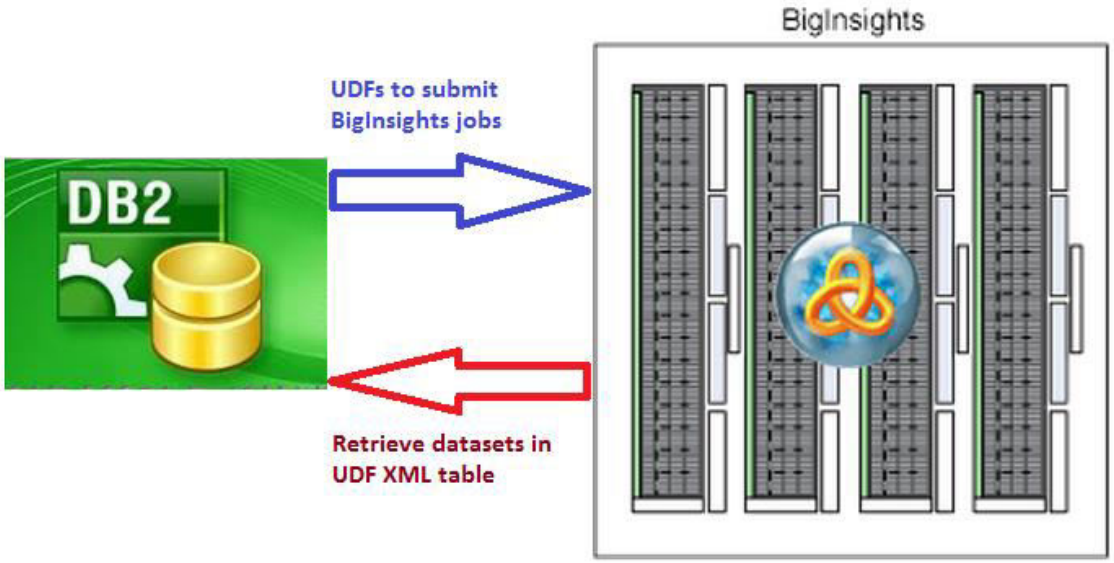
- **Solution**:
  - All 4 Linux machines are clustered within a privileged internal network
  - Keep z/OS system in connection with Linux machines under VLAN DB2_BigXML for persistence
  - Data transfer from z/OS to Linux is remained in small-sized workload

Software constraints and solutions

- **Size of XML files not feasible in Hadoop's storage block**
  - Hadoop allocates storage block with blocksize of 64MB or 128MB
  - XML file's size > 128MB needs to be splitted onto multiple blocks
- **Solution:**
  - Assume each XML file consumes a size of 64MB or less
  - Each node has a 40GB space, so it can take up to at most 321 XML files at a time without file split

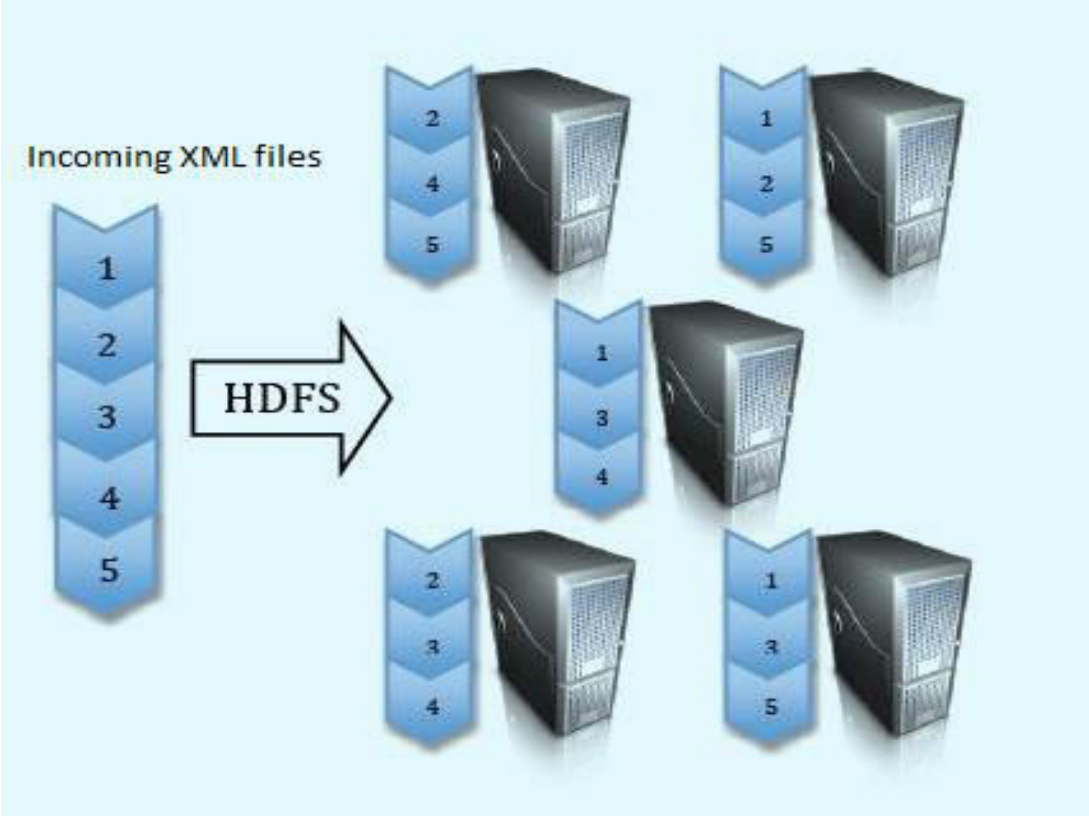
Design and Implementation
- *Connectivity*

- Data transfer
  - Enable DB2's JDBC driver to connect to BigInsights's ad-hoc server via Database Import appliance
  - Connection persists until application commits for both data submission and retrieval



Design and Implementation
- *Systematic tuning in Big Data*

- Distribute incoming XML files to all nodes of Hadoop stored file system.



Design and Implementation
- *Systematic tuning in Big Data*

- **MapReduce Kickoff**
  - spawn mapper function per block to filter relevant information from each XML file
  - each node extracts the file, aggregates and collect filtered data from different nodes into a central repository.
  - XQuery API for Java in Hadoop with Package javax.xml.xquery is imported
  - Use JAQL's XPath API for Java on Hadoop stored file system
  - Send query results back to DB2z

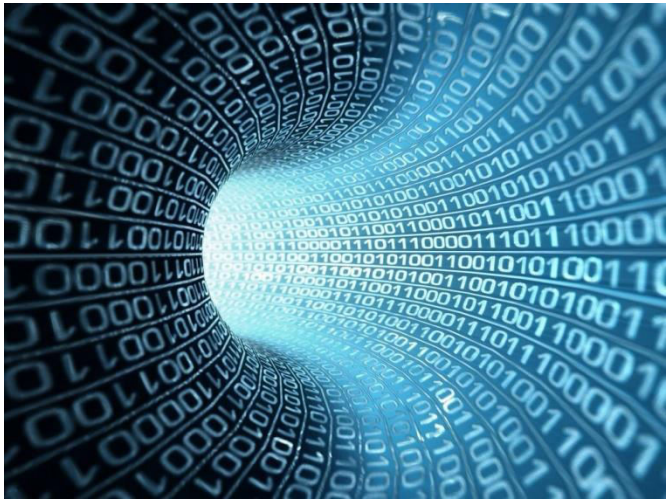
End-User Tier

- From InfoSphere's BigInsights Web Console, deploy Database Import application for loading XML documents using JDBC driver class com.ibm.db2.jcc.DB2Driver.
- User is able to run a dynamic SQL statement to query on the table by using a SELECT statement.
- The result of the query is converted in CSV format which is to be stored at /BigXMLdirectory.

Database Import

Middle-Tier

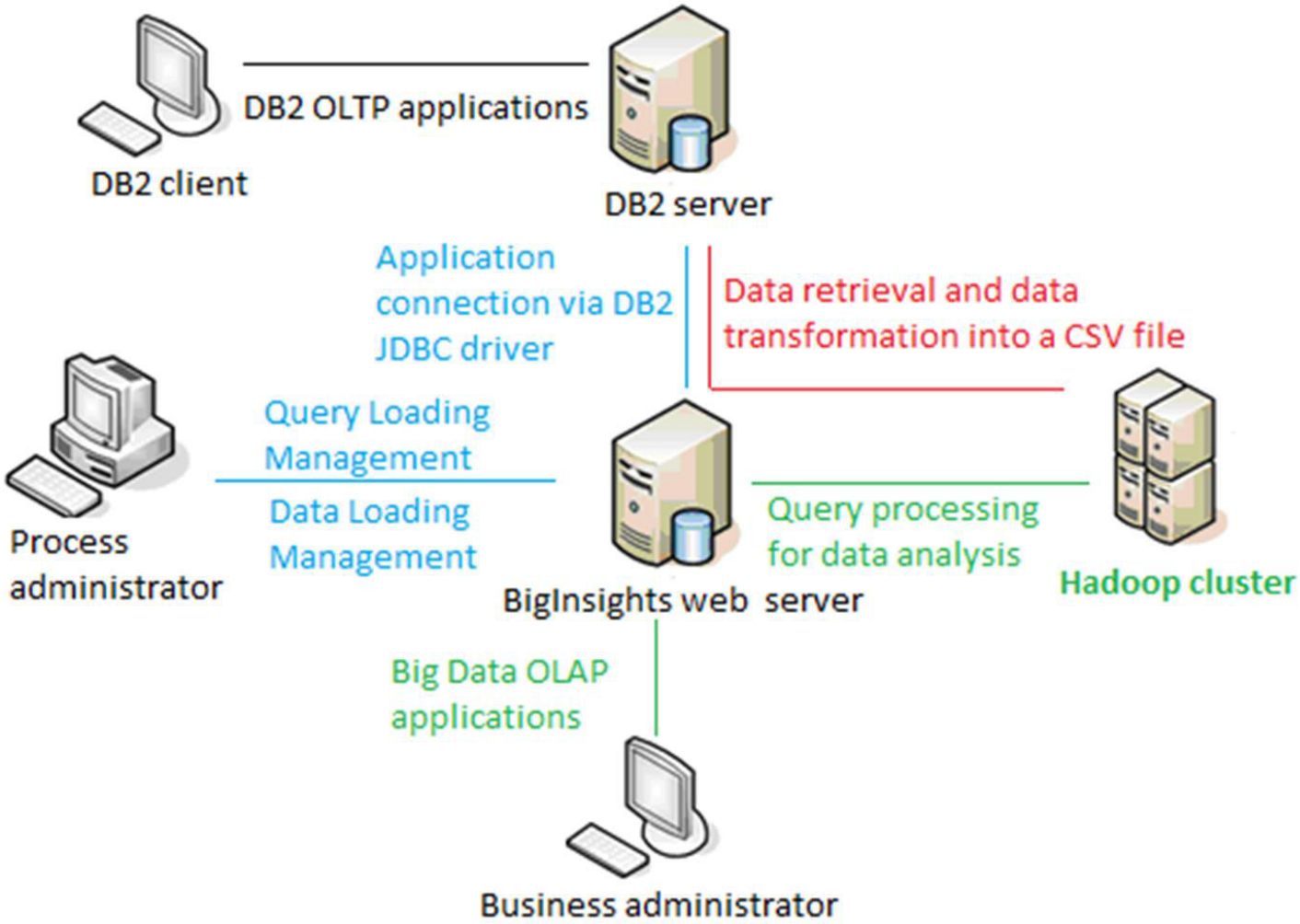
- To allow database driver access, create 2 DB2 database drivers which contain crucial database connection parameters to find the target database destination address with access authentication credentials.



Data-Tier

- XML data stored in DB2 database is converted to a plain text in a file on Hadoop cluster.
- A plain text with multiple XML tags and unstructured format is transformed using customized application called xmlProcessing which can be deployed from BigInsights web console.
  - This applications reads out each bytes between the first start tag and the stop/end tag. It is assumed that the tags are treated as UTF-8 bytes.
  - In order to rebuild the XML text file back to the structured format that XPath function can parse to query against to, tags are to be removed because the tags are not returned as part of the querying result.

Architecture layout



Performance measurement and Benchmark

- **Data Loading onto Hadoop**
  - Transferring the same data in approximately <1GB of XML data from DB2 to Hadoop cluster had done 5 times. The peak elapsed time was 20 second, the quickest elapsed time was 14 second.
  - The average CPU elapsed time was 16 second.

Application History

Status	Execution Name	Progress	Start Time	Elapsed Time (sec)
No filter applied				
✓	loadXML10	100%	Jul 22, 2013 1:50:39 PM	16
✓	loadXML10	100%	Jul 22, 2013 1:49:52 PM	14
✓	loadXML10	100%	Jul 22, 2013 1:48:38 PM	20
✓	loadXML10	100%	Jul 19, 2013 4:05:27 PM	15
✓	loadXML10	100%	Jul 19, 2013 3:39:53 PM	15

Performance measurement and Benchmark (cont.)

- **Analyzing data using XPath query schema**
  - Testing begins with a query which drives down to 2nd level node tag where the predicate is for a matching condition. This is a simple query. The first run took 28 seconds and second time took 23 seconds.
  - For a complicated query where the a lower node tag evaluation is added in addition to the previous matching condition, the performance looks even better. First time run took 24 seconds and second time run with the same query took only 23 seconds of elapsed time.
  - The average time is about 24.5 seconds of CPU elapsed time.

Application History

No filter applied

Status	Execution Name	Progress	Start Time	Elapsed Time (sec)	Details
✓	analyzeXML2	100%	Aug 4, 2013 1:41:26 AM	23	
✓	analyzeXML1	100%	Aug 4, 2013 1:40:28 AM	23	
✓	analyzeXML2	100%	Aug 4, 2013 1:35:56 AM	24	
✓	analyzeXML1	100%	Aug 4, 2013 1:29:57 AM	28	

Performance measurement and Benchmark (cont.)

- **Retrieving query result from HDFS to DB2**
  - Retrieving query result back to DB2 requires the use of HDFS_READ function to reads file contents from the HDFS and returns them back to DB2 table. CPU time using SELECT from UDF table by HDFS_READ function takes about 0.06 second to retrieve the data.
  - In general, the data retrieval takes less than 0.1 second of CPU elapsed time using HDFS_READ for file size less than 10MB. It is also determined that the elapsed time of each applications varies for every individual application execution due to network traffic and the variation from data node response time to process the query.

```
DSN SYSTEM(SSTR)
RUN PROGRAM(DSNTEP3)
//SYSIN    DD *

TEP3FLAG XMLSIZE 10000;

SELECT XMLVal   FROM TABLE(
   HDFS_READ(
'http://sdsvm904027.svl.ibm.com:8080/data/controller/dfs/BigXML/objectda
taResult.txt',''))
AS T1(XMLVal VARCHAR(10000));
```

Conclusion and Future Work

- ✓Implemented OLAP for query processing on XML data in the MapReduce framework
- ✓Enabled DB2 XML data offloading
- ✓Enabled XQuery queries offloading from DB2z
- ✓MapReduce framework is reworked on BigInsights's ad hoc server to enable XQuery support
- ✓Aggregated Data in BigInsights is allowed to transfer from HDFS back to DB2 forming a XML table in z/OS mainframe

Future work

- Allow XML file separation for multi-block processing
- Allow CDC (Changed Data Capture) schema for continuous online transactional processing (OLTP)
- Customize the query output type instead of plain text format

IBM DB2

Thank you!

George Wang
IBM
fgwang@us.ibm.com