Adventures in Mainframe Data Integration: How MDI is Changing the Value and Economics of the Mainframe Session #25987

Colleen Gordon, *MDI Product Manager*, Luminex

Art Tolsma, *CEO*, Luminex

FTP – Super Brief Overview

- FTP – File Transfer Protocol – Developed in the 1970's
  - Current RFC 959 (https://tools.ietf.org/html/rfc959) – from 1985
- Two TCP connections for most commands
  - DATA channel – system uses to send/receive data files
  - CONTROL channel – PUT and GET commands
- Both can be encrypted (FTP/S) BUT…
  - Requires certificates and ideally client certificates to do well
  - Configuration can be tedious
- Don't confuse with SFTP – a construct of SSH and entirely different
- Then there is FTP on Z …

FTP – on Z

- Base functionality the same as FTP on other systems (same feature set)
- Three modes of operations
  - Basic - SITE FILE=SEQ – transfer files back and forth
  - JES - SITE FILE=JES – allows you submit jobs directly to JES
  - SQL - SITE FILE=SQL – allows you to issue SQL commands; connect to DB2, pull some rows, update rows, delete rows
- This makes securing (or abandoning) FTP on Z even more important than your other platforms
  - If a hacker gets access to your FTP; this is a list of things they can do!
- Certificates can also be used here
  - Need Certificate Authority (CA)-signed certs to make it effective
    - Like Verisign
  - Self-signed certificates are better than nothing but smart hackers can get around
  - DATA+CTRL must be encrypted
- But you say …

But We Only Use It Internally!

This is why Mark and Chad (RSM) will have job security forever!

FTP – The Risks!

- Treat your network as hostile; even though inside your firewalls
  - Can't possibly verify that all of your (hundreds or thousands of) nodes have not been compromised
- Using plain FTP is essentially the same as using plain HTTP
- It provides neither encryption nor tamper resistance
- This means passwords are transmitted in clear and an attacker can sniff the passwords
- The attacker can also modify the traffic; such as injecting malware into downloads
- This means that plain FTP should not be used for anything where these attacks can be a problem, i.e. for most things were FTP is actually still used today ☺

Mark's Own Experience!

- While penetration testing a z/OS system found packet capture file
- Offloaded and saw FTP protocol was in the capture
- Looking at the capture – found credentials!

Info

```
Response: 220-FTPDI1 IBM [redacted], 19:02:55 on 2018-10-
Response: 220 Connection will close if idle for more than 50 minutes.
Request: FEAT
Response: 211 no Extensions supported
Request: USER badguy
Response: 331 Send password please.
Request: PASS P@ssw0rd
Response: 230 BADGUY is logged on.  Working directory is "BADGUY.".
Request: PWD
Response: 257 "'BADGUY.'" is working directory.
```

… continued

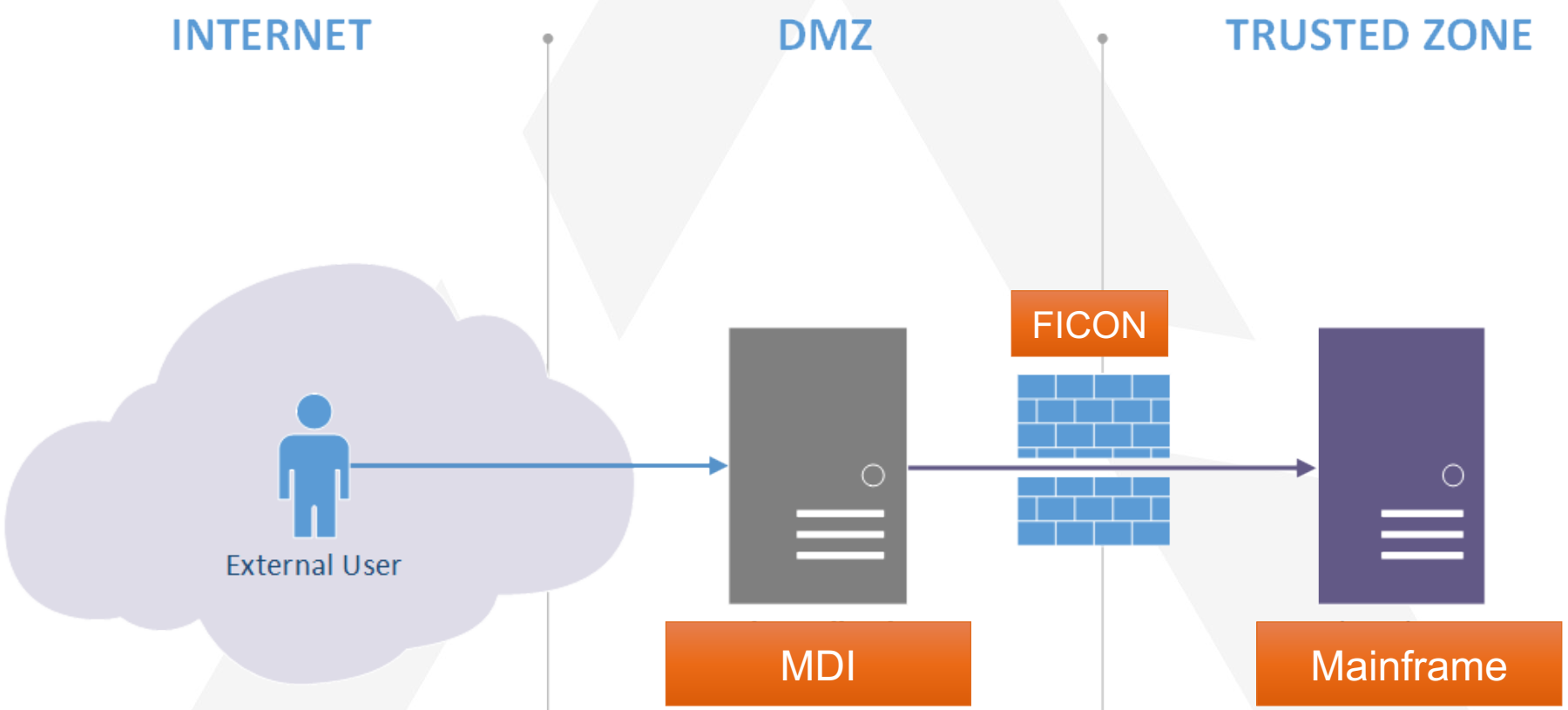
- Those credentials had no access to TSO
- However, they did have access to update 1 APF-authorized library!
- But the Userid had access to FTP!
- The rest is history
- Disabling or encrypting FTP would have prevented this
- Disabling may be better as locking down FTP is very difficult – shrink the MF attack surface

What is FICON?

- FICON (for *Fibre Connectivity*) is a high-speed input/output (I/O) interface for mainframe connections to storage devices

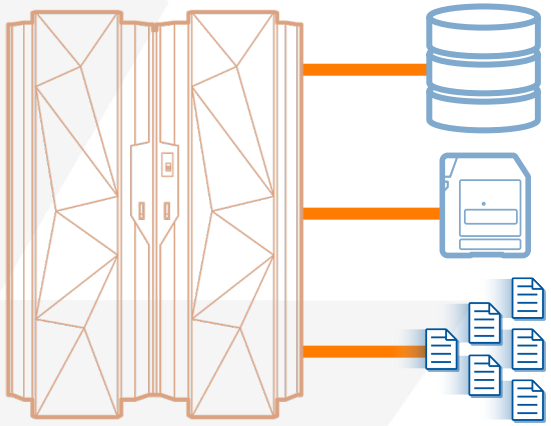
- FICON channel features include:
  - A mapping layer based on the ANSI standard Fibre Channel-Physical and Signaling Interface (FC-PH), which specifies the signal, cabling, and transmission speeds
  - 100 Mbps bi-directional link rates at distances of up to twenty kilometers, compared to the 3Mbps rate of ESCON channels at distances of up to three kilometers.
  - Support for full-duplex data transfers, which enables simultaneous reading and writing of data over a single link
  - Multiplexing, which enables small data transfers to be transmitted with larger ones, rather than having to wait until the larger transaction is finished

MDI is DMZ for File Transfer



A DMZ for File Transfer

- If we can remove the use of FTP on the mainframe, even closing the ports and disabling the service, this wards off some of the bad folks as that may be their attack vector
- Move all of that data transfer workload to MDI its faster (FICON) and its secure
- FICON was designed specifically for the mainframe its attributes are Fast, Efficient and Secure
- Uses are:



- DASD
- Tape
- … and now File Transfer

The Good News

- File transfer from mainframe is a huge attack surface
- Moving it to a separate, hardened “DMZ” type host improves security posture
- Flexibility of having a host with multiple secure forms of transport is a huge plus
- Doing so with little or no change to existing code on the mainframe is also a plus

Summary

- Shrinking the attack surface on the mainframe by moving file transfer to a secure, smaller-footprint, easily-hardened box is the right answer
- Not doing security right from the get-go might actually make things worse: high concentration of customer data in one place, with weak controls
- Patching & patch visibility are paramount!
- Turning off legacy protocols (like FTP) is a huge win

Traditional Data Transfer Options

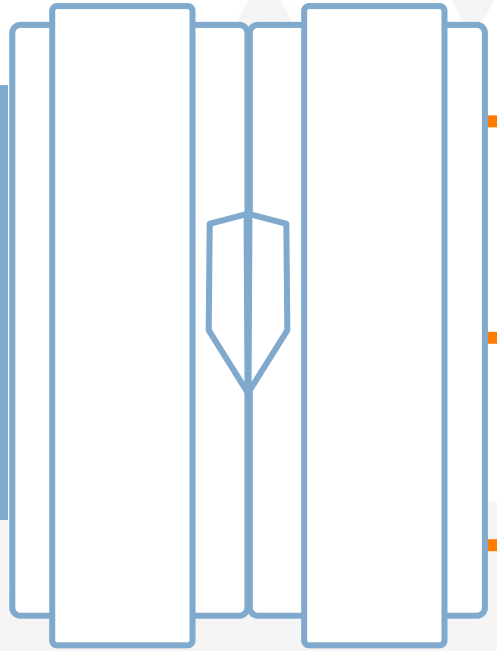
- TCP/IP
  - FTP or File Transfer Protocol – Developed in 1970s
  - Can be encrypted (FTPS)
    - Requires certificates
  - SFTP
    - Encrypted File Transfer on the mainframe
- Vendor Supported Options
  - Several choices
  - All use TCP/IP to secure file transfers on the mainframe

A Better Alternative to Mainframe TCP/IP
MDI Uses FICON as the Network

FICON is an I/O channel technology designed
specifically for the mainframe

Attributes

- Fast
- Efficient
- Secure



Uses

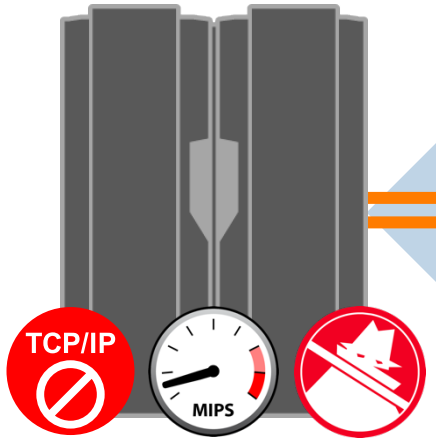
- DASD
- Tape
- File Transfer

Fast, Secure & Efficient Data Movement

MDI is a Data Transfer and Co-Processing Platform

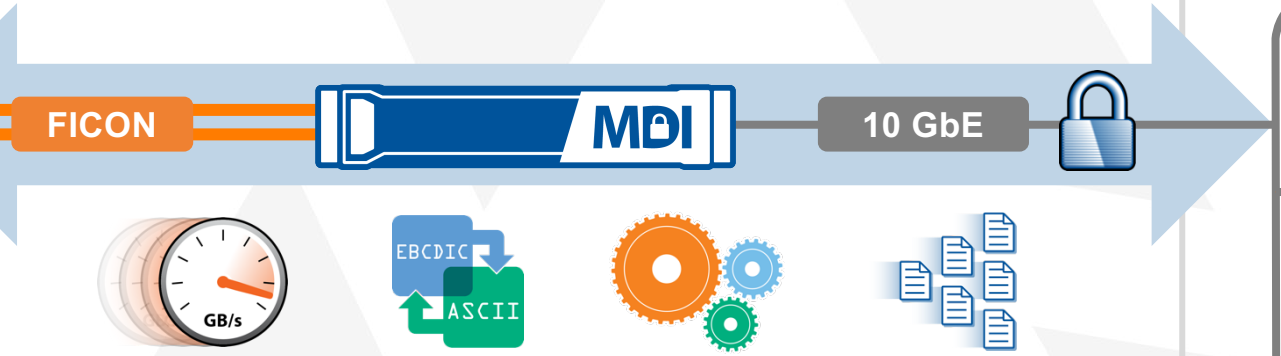
SHARE
EDUCATE • NETWORK • INFLUENCE

Mainframe FICON



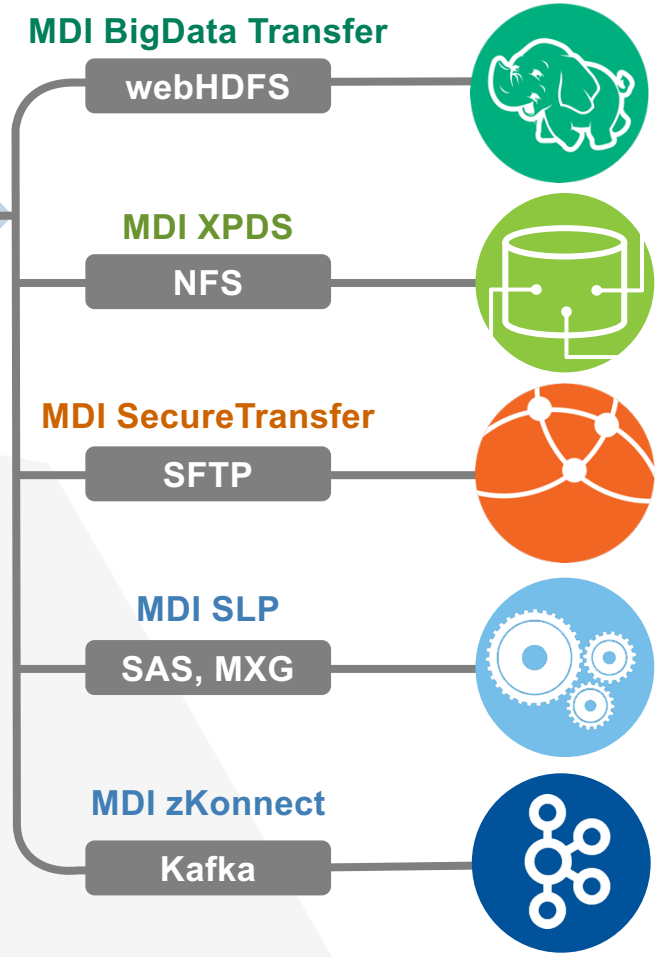
- Secure
- High speed
- Efficient, redundant I/O channels

MVT or Dedicated MDI Platform



- Profile-based architecture for extending processing & interface capabilities
- High speed, scalable transfer rates
- SAF integration & protocol-based encryption
- Bi-directional movement and communication for multi-platform workflows and co-processing
  - Including data translation from EBCDIC to ASCII and between character sets

Data Sharing Targets/Sources



Use Case: Move Large VSAM Files to Azure Data Lake

SHARE
EDUCATE • NETWORK • INFLUENCE

US Insurance Company Challenge

System z data was generated, collected and processed separately than open systems data

Created data silos between mainframe and Open Systems

Made risk analysis even more challenging as the analytics team had to perform analysis on mainframe data differently

Luminex MDI Solution

Use MDI SecureTransfer to Integrate VSAM data from the mainframe into the Azure Data Lake

MDI SecureTransfer moves data faster than SFTP; eliminating hours of transfer time that created data latency

Ease-of-use allowed for quick deployment of the solution

Outcome

Integrating mainframe data into the Data Lake provided insights to Life & Disability Services; understanding claim volume and severity

Improved customer experience through increased transparency

Fosters the opportunity to drive continuous improvement

Use Case: Rehosting 100 Mainframe Applications

SHARE
EDUCATE • NETWORK • INFLUENCE

US Retail Company Challenge

Client was challenged with moving up to 12TB of new and changed data on a daily basis during the transition from mainframe to x86 application

SFTP used too much system overhead and interfered with production processing

SFTP could not move the data within the required 12 hour window

Luminex MDI Solution

Use MDI Cross Platform Data Sharing (XPDS) to move daily changed files directly to NAS mounted storage

XPDS facilitated moving 2.5TB per hour over the clients dark fiber connection

Outcome

Client was successful in re-hosting 100 applications to x86 environment

XPDS continues to be used to copy historic data to new platform

Client was able to avoid additional expenses by meeting project deadlines

Use Case: Reduce Chargeback and Facilitate Data Sharing for 10 Business Units

SHARE
EDUCATE • NETWORK • INFLUENCE

Large Financial Institution Challenge

Client needed to reduce costs while increasing their ability to share mainframe data with distributed applications

Client used a Vendor Supported solution and was charged for each use

Technical Architecture team challenged with finding a better way to share data

Luminex MDI Solution

Use MDI SecureTransfer and XPDS to facilitate moving data quickly and easily between platforms and reduce costs on the mainframe

XPDS facilitated moving 2.5TB per hour over the client's dark fiber connection

Outcome

Client reduced costs by $4.2M over five years

10 Business Units adopted program, moving 2.5PB of data to commodity storage

Business Units achieved seamless integration between mainframe and distributed applications

Use Case: Move MXG Processing to Open Systems

Large Transportation Company Challenge

Client wanted to reduce MIPS consumption

Reduce wall-clock time to produce daily MXG reports

Free up DASD storage on their mainframe

Luminex MDI Solution

Use MDI SAS Language Processor (SLP) to move SMF data and SAS Language programs to Linux server

Return reporting to the mainframe and distribute from open systems

Utilized existing Open System's SAS Institute license to execute MXG reports

Outcome

Reduced execution time for MXG processing from 6-8 hours to under 28 minutes

Freed up 7TB of DASD space utilized by SMF and PDB storage

Allowed client to keep more SMF data types in PDBs

Removed 16% of processor workload over 6-8 hour period

Other Uses

- Archive mainframe data in the cloud
- Reduce data latency by moving application data to a data warehouse using MDI
- Stream machine data from the mainframe into Kafka
- Replace unsecure FTP to and from the mainframe with secure MDI
- Push data, processed by distributed applications, to the mainframe for batch processing
- Replace expensive Vendor Supported TCP/IP solutions with MDI
- Populate Kafka topics with application data from the mainframe
- Drop data into directories on shared storage for data analytics processing
- Send mainframe data to Pentaho, Tableau, Talend, Cognos and the like
- And the list goes on….

Trends and Perspective

- Multi-cloud IT requires The Integrated Mainframe
  - Applications and processing is done where best fit and can be moved seamlessly between environments
- 19 Sessions at SHARE reference Zowe open source integration software
- Analytics on z Roadmap is focused on making all of the widely used software and tools such as Spark, Python, R, TensorFlow, etc are available on z
- “It’s All about Integration” and “Significant focus to make the mainframe a seamless participant” – Jeff Magdall SHARE Phoenix Session March 2019

Trends and Perspective

- The core premise for Analytics on z
  - “Moving computation is cheaper than moving data”
  - If the data resides on z and real-time (sub-second) processing is required
- IBM has recently promoted “Tailored Fit Pricing” to simplify pricing with a cpu consumption model more similar to cloud IT pricing

- MDI fundamentally changes the Value and Economics of The Integrated Mainframe
  - by making it Secure, Fast, Easy and Cost-Effective to move lots of data to/from the mainframe
  - The z Platform isn’t always the best platform for certain workloads
  - Not all CPU usage is equally valuable
  - Moving data for batch or near-real-time processing is no longer the obstacle it is assumed to be

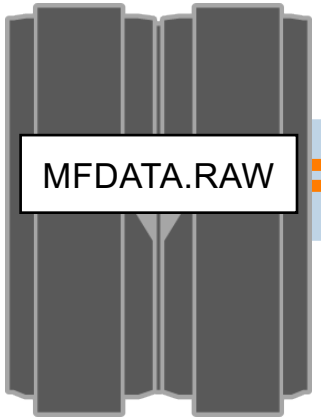
What's Next

- Add Near-real-time processing in addition to batch with zKonnect Mainframe Kafka Producer
  - LUMXPROC - Batch output written to Kafka (tape volser vmethod)
  - XWRITER - JES2 spool files (output data) written to Kafka in near-real time
  - Log Receiver - System and Application log steams to Kafka in near-real time
  - SMF Receiver - SMF log streams sent to Kafka in near-real time
- Customer-driven integrations for off-host processing such as MDI with Hitachi Pentaho Analytics Platform
- Expand and Secure Integrated Processing to complex mainframe Actions that are not REST API enabled

MDI and Pentaho
Offload Analytics and Return Result to Mainframe

SHARE
EDUCATE ▸ NETWORK ▸ INFLUENCE

Mainframe

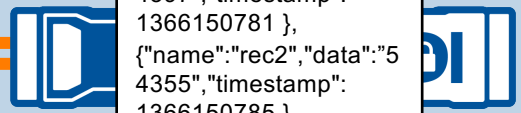


- Execute JCL to begin job
- Set input/output parameters
- Send raw file down tape channel (FICON)

MFDATA.RESULTS

MDI Platform

11010101010
1010101001



{"name":"rec1","data":"3
4397","timestamp":
1366150781 },
{"name":"rec2","data":"5
4355","timestamp":
1366150785 }
…

- Convert EBCDIC to ASCII
- Execute Pentaho Transformation (Pan)
- Return Result to Mainframe

10 GbE
(NFS)

Pentaho

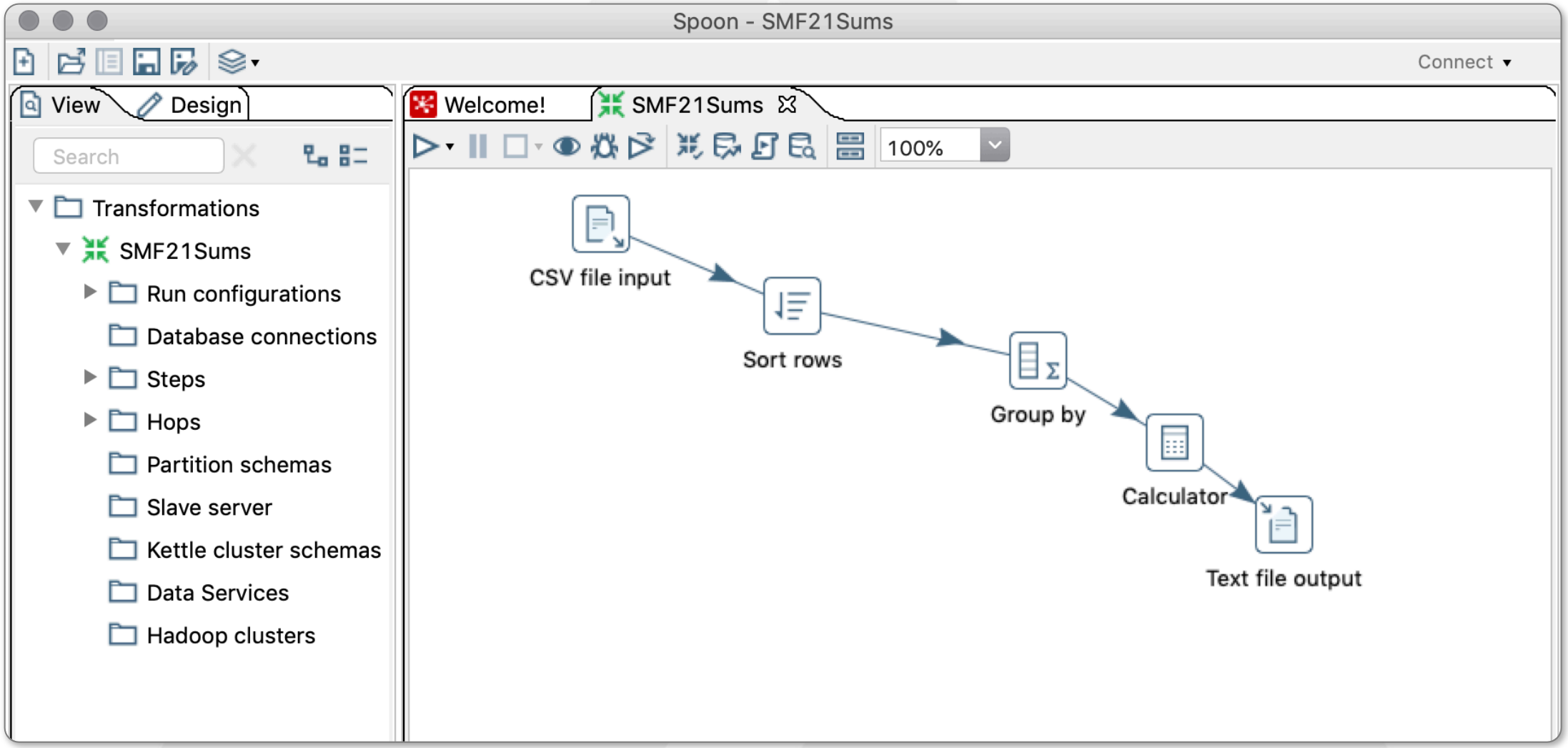
- Perform SORT
- Group on keyword
- Perform aggregation
- Write results output

Group1 123213
Group2 31312
…

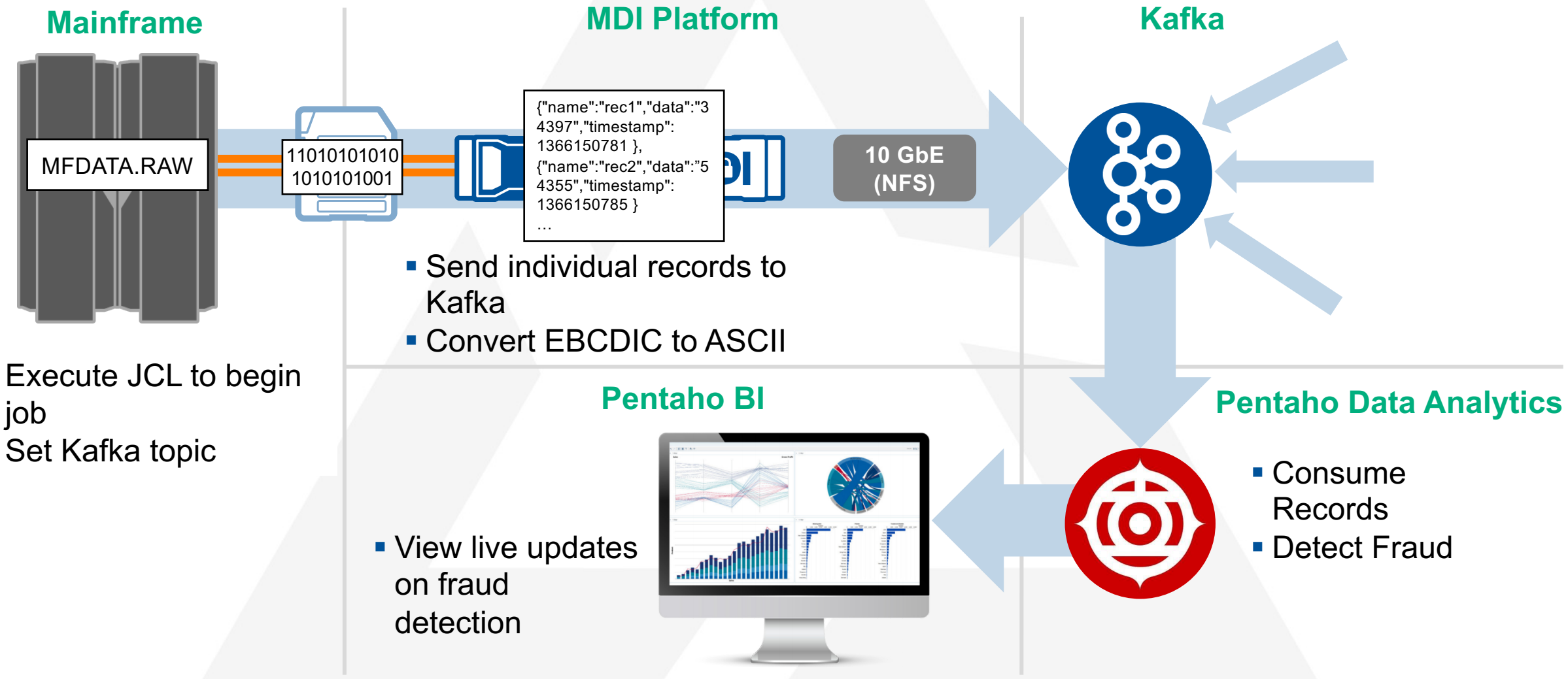
- Results returned to mainframe and/or sent out to open systems

Group1 123213
Group2 31312
…

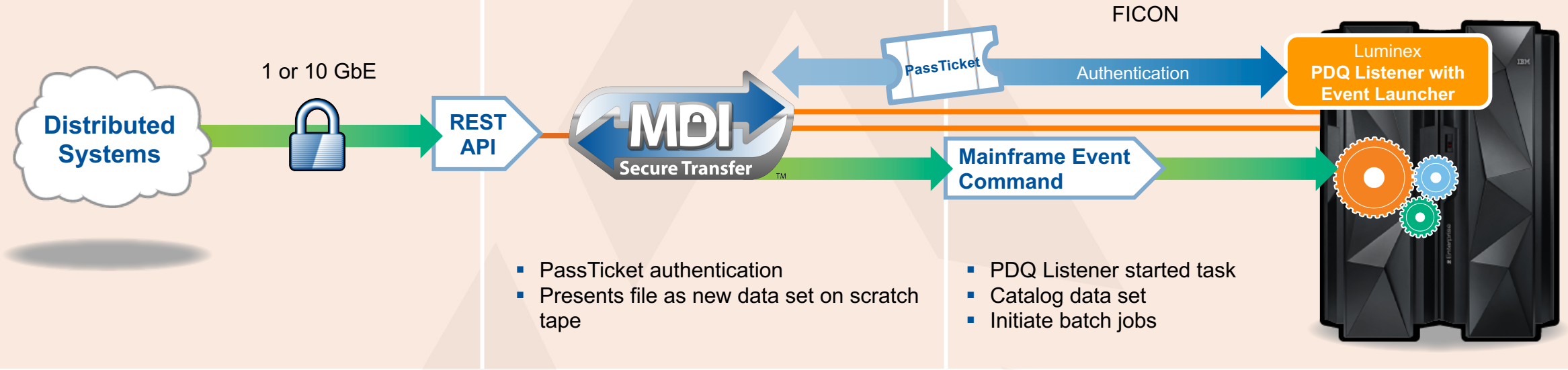
Pentaho Sample Integration (Customer Provided)



MDI and Pentaho – Streaming with Kafka



MDI Futures:
REST API to Initiate Mainframe Processing



- Communication from the MDI Platform to the mainframe, via PDQ, has been expanded from “Catalog Tape” action to any predefined mainframe action with the new Event Launcher feature
- Issue Command on MDI Platform to RUN JOB1 where JOB1 has been previously created on the Mainframe
- Command is sent to the mainframe and executed
- Command Dictionary is published and available via REST API framework for distributed applications and authorized users

Adventures in Mainframe Data Integration: How MDI is Changing the Value and Economics of the Mainframe Session #25987

Colleen Gordon, *MDI Product Manager*, Luminex

Art Tolsma, *CEO*, Luminex