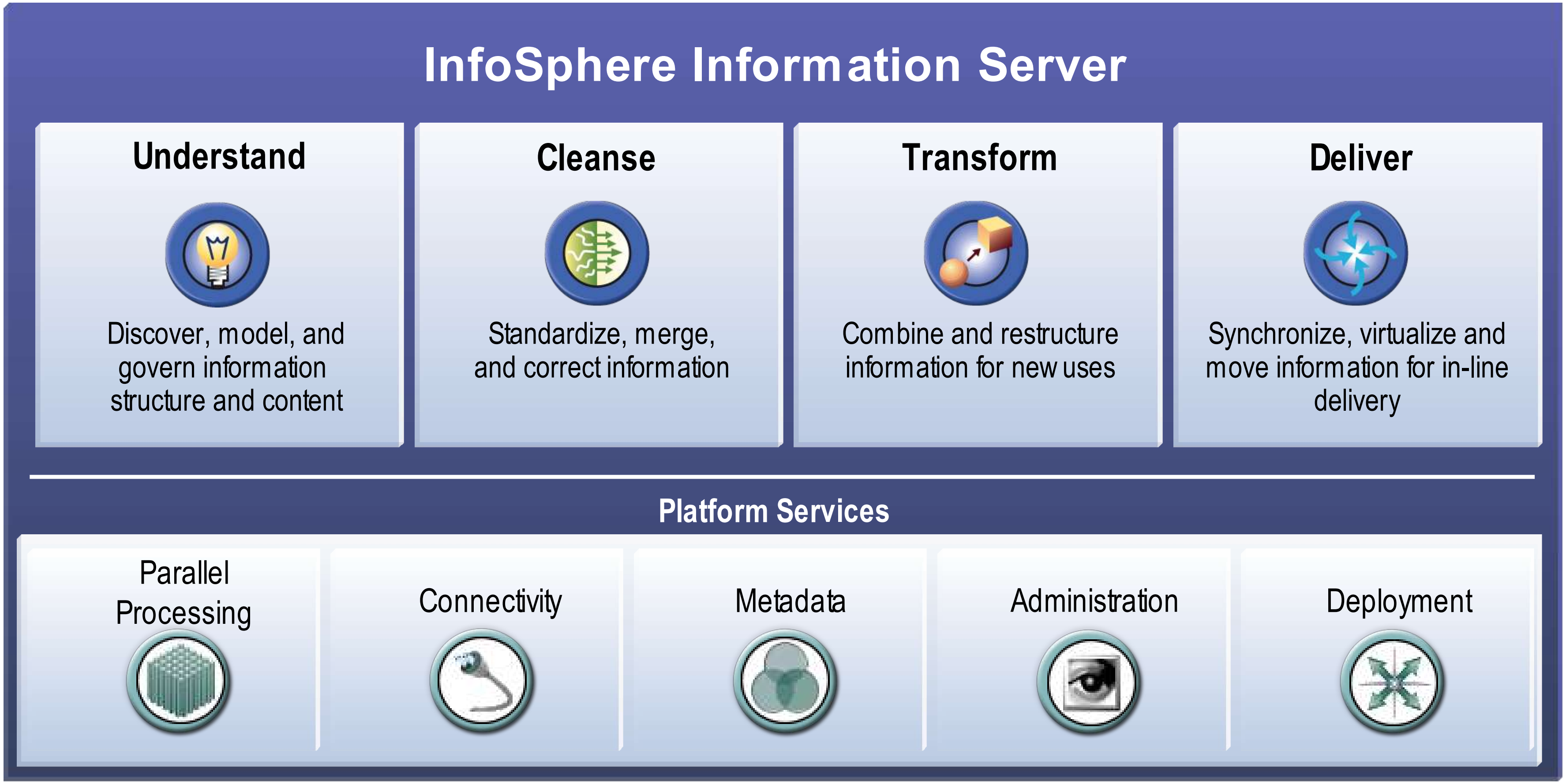
# InfoSphere CDC Technical Features & Industry Applications

Frank Ketelaars
Sr. Technical Field Specialist – Worldwide

E-Mail: fketelaars@nl.ibm.com

# Information Server - Delivering information you can trust



## InfoSphere Information Server

**Understand**

Discover, model, and govern information structure and content

**Cleanse**

Standardize, merge, and correct information

**Transform**

Combine and restructure information for new uses

**Deliver**

Synchronize, virtualize and move information for in-line delivery

**Platform Services**

Parallel Processing

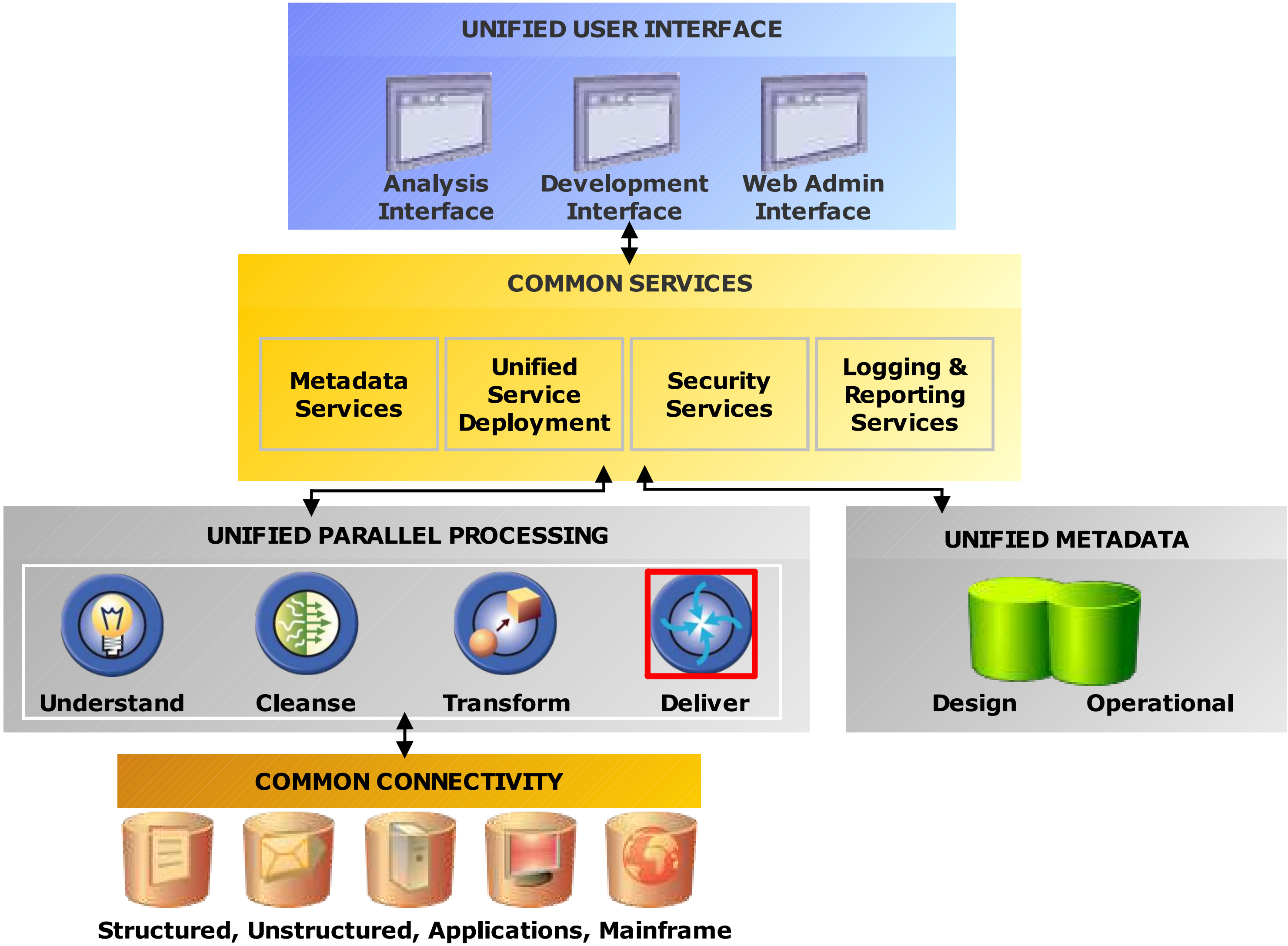
Connectivity

Metadata

Administration

Deployment

# InfoSphere Information Server Architecture



# Summary of problems CDC addresses

## Business intelligence and reporting

- *Yesterday's* data inadequate for inventory and purchasing decisions
- Batch window reduction

## Integration of business applications

- Up to date information flowing between applications (a.o. for eBusiness)

## Real-time event detection

- Pro-actively monitor and respond to business changes

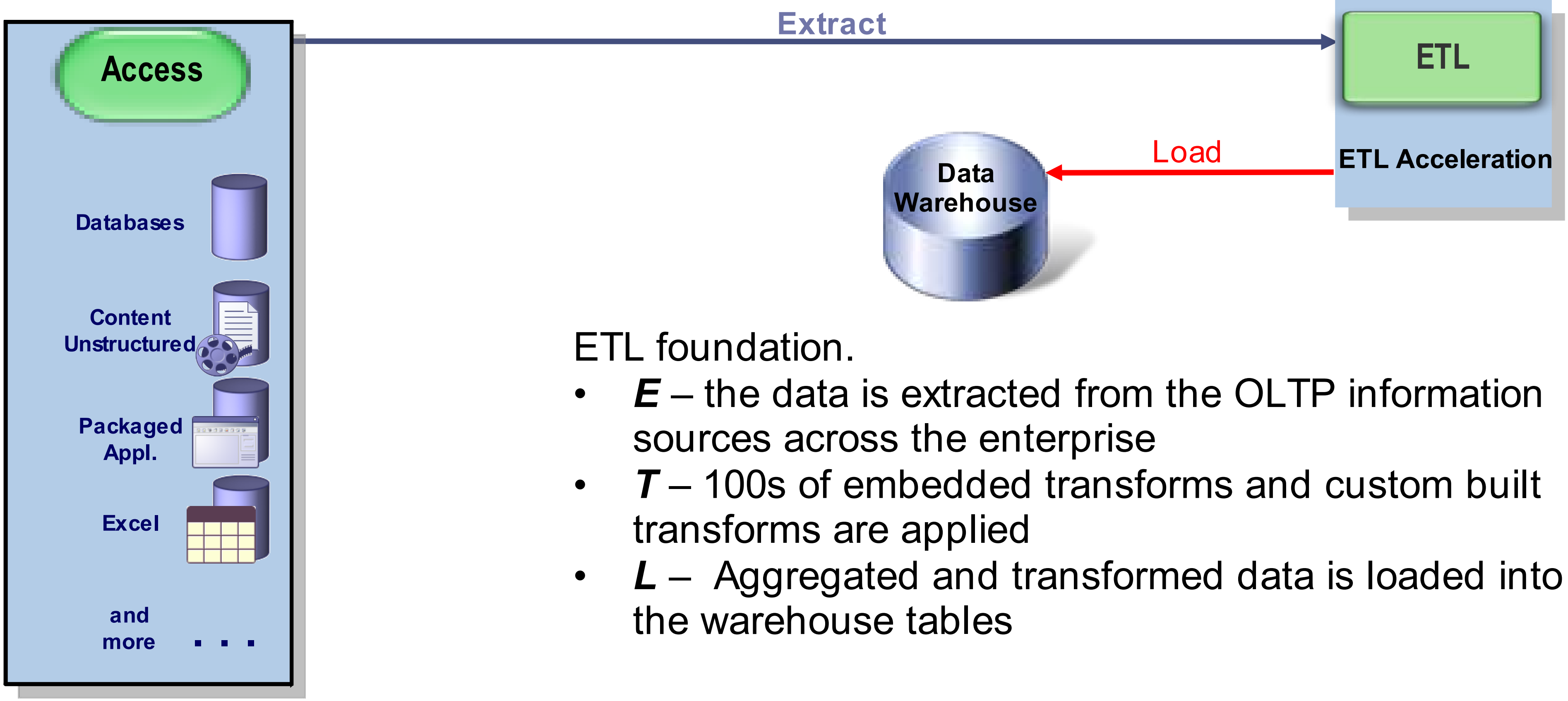
## Migration of applications and databases

- Minimize downtime and mitigate risk migrating versions, database types and hardware

***……Without slowing the performance of production systems***

# Traditional ETL …

## *Leverage access technology and an ETL*

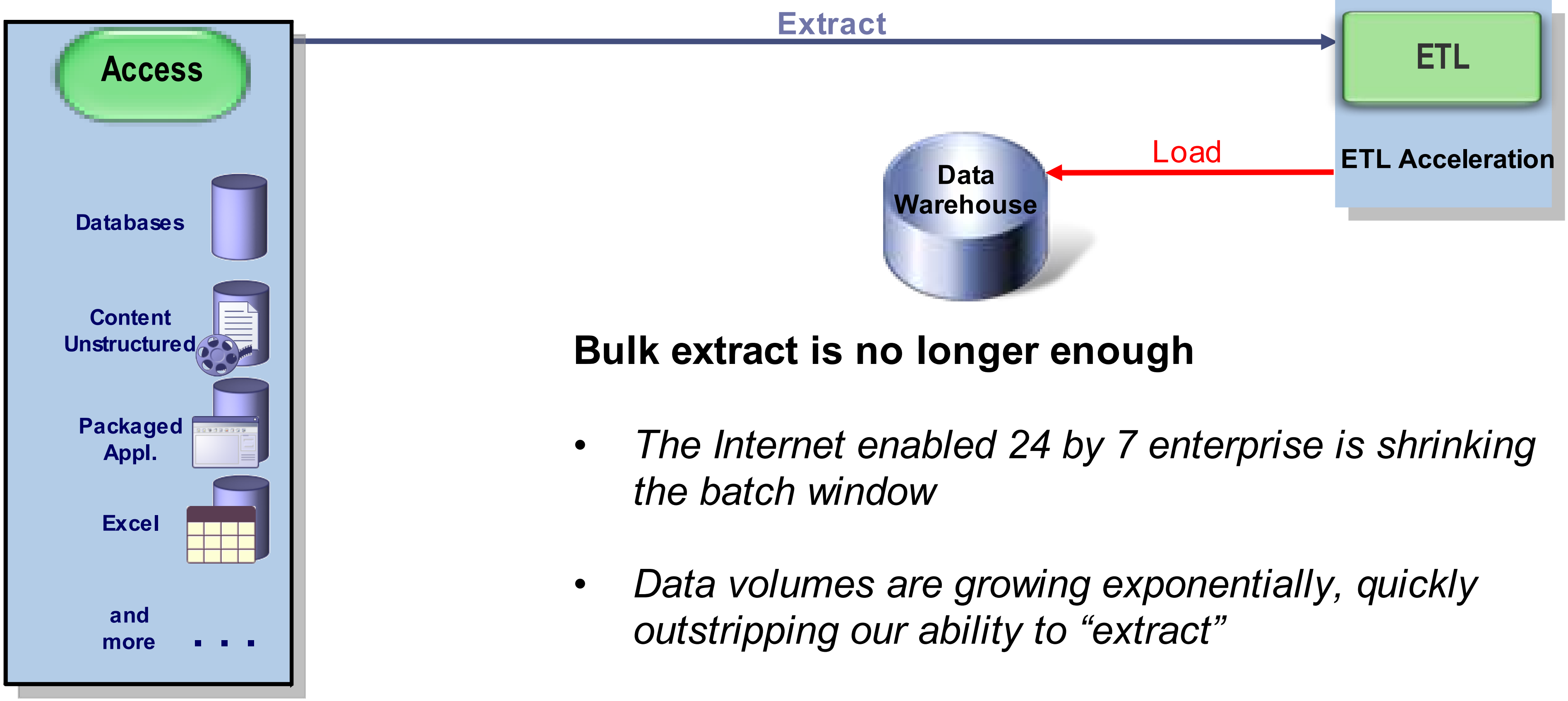


ETL foundation.

- ***E*** – the data is extracted from the OLTP information sources across the enterprise
- ***T*** – 100s of embedded transforms and custom built transforms are applied
- ***L*** – Aggregated and transformed data is loaded into the warehouse tables

# Information demands are changing…

*Requirements for "real time", dynamic environments*

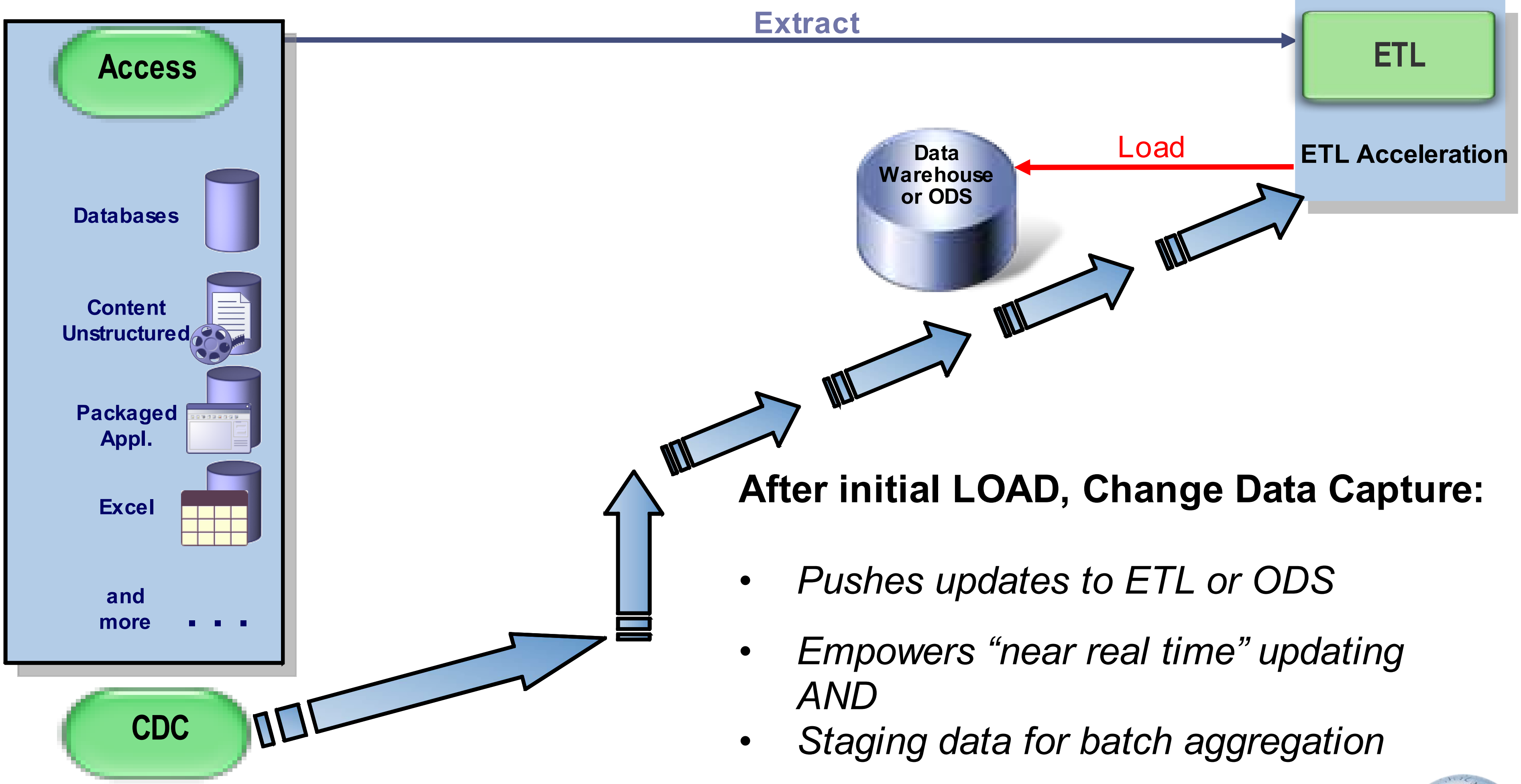


**Bulk extract is no longer enough**

- *The Internet enabled 24 by 7 enterprise is shrinking the batch window*
- *Data volumes are growing exponentially, quickly outstripping our ability to "extract"*
- *Yesterday's data is no longer "enough", operational BI demands varying degrees of data latency*

# Shift is to ongoing, incremental updating

*Integrate operational data with Business Intelligence & analytics*



**After initial LOAD, Change Data Capture:**

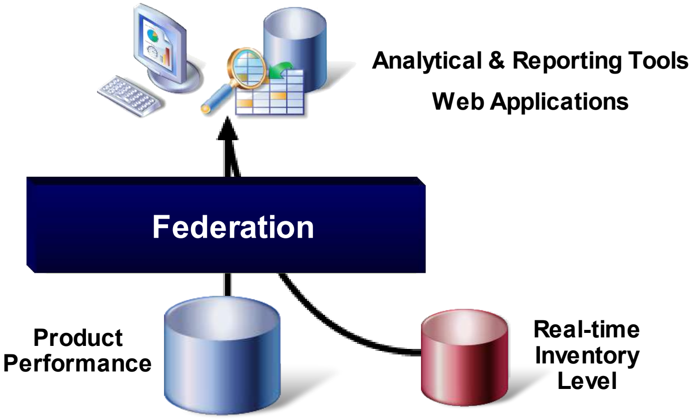
- *Pushes updates to ETL or ODS*
- *Empowers "near real time" updating AND*
- *Staging data for batch aggregation*
- *Optimizes bandwidth utilization*

# Today’s environments require multiple data delivery styles

*Techniques to consider*

## Access

“PULL” Data for Extract of ETL



Analytical & Reporting Tools

Web Applications

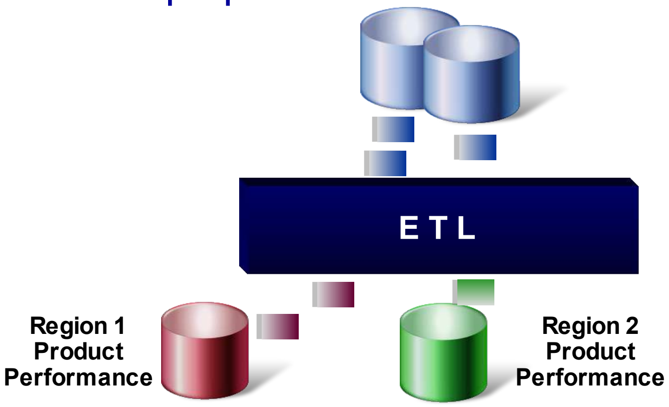
Federation

Product Performance

Real-time Inventory Level

## Extract, Transform, Load

Repurpose Information for ODS/DW



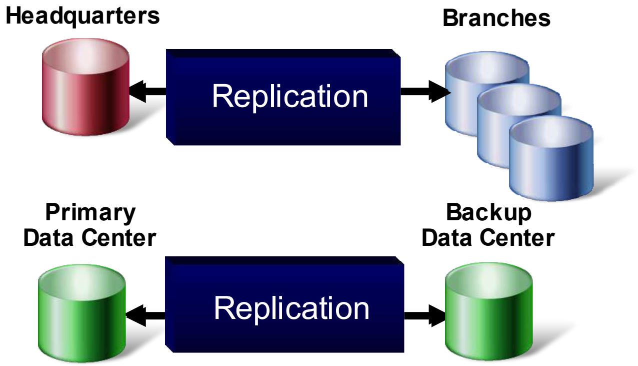
E T L

Region 1 Product Performance

Region 2 Product Performance

## Replication

Synchronize Business and Reporting/DR



Headquarters

Replication

Branches

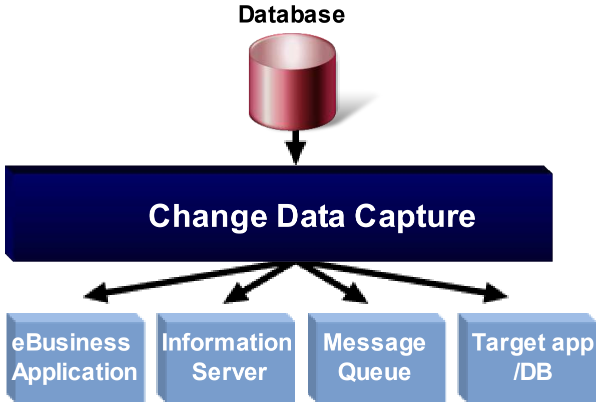
Primary Data Center

Replication

Backup Data Center

## Change Data Capture

“PUSH” Data for Real-Time ODS/DW Updating



Database

Change Data Capture

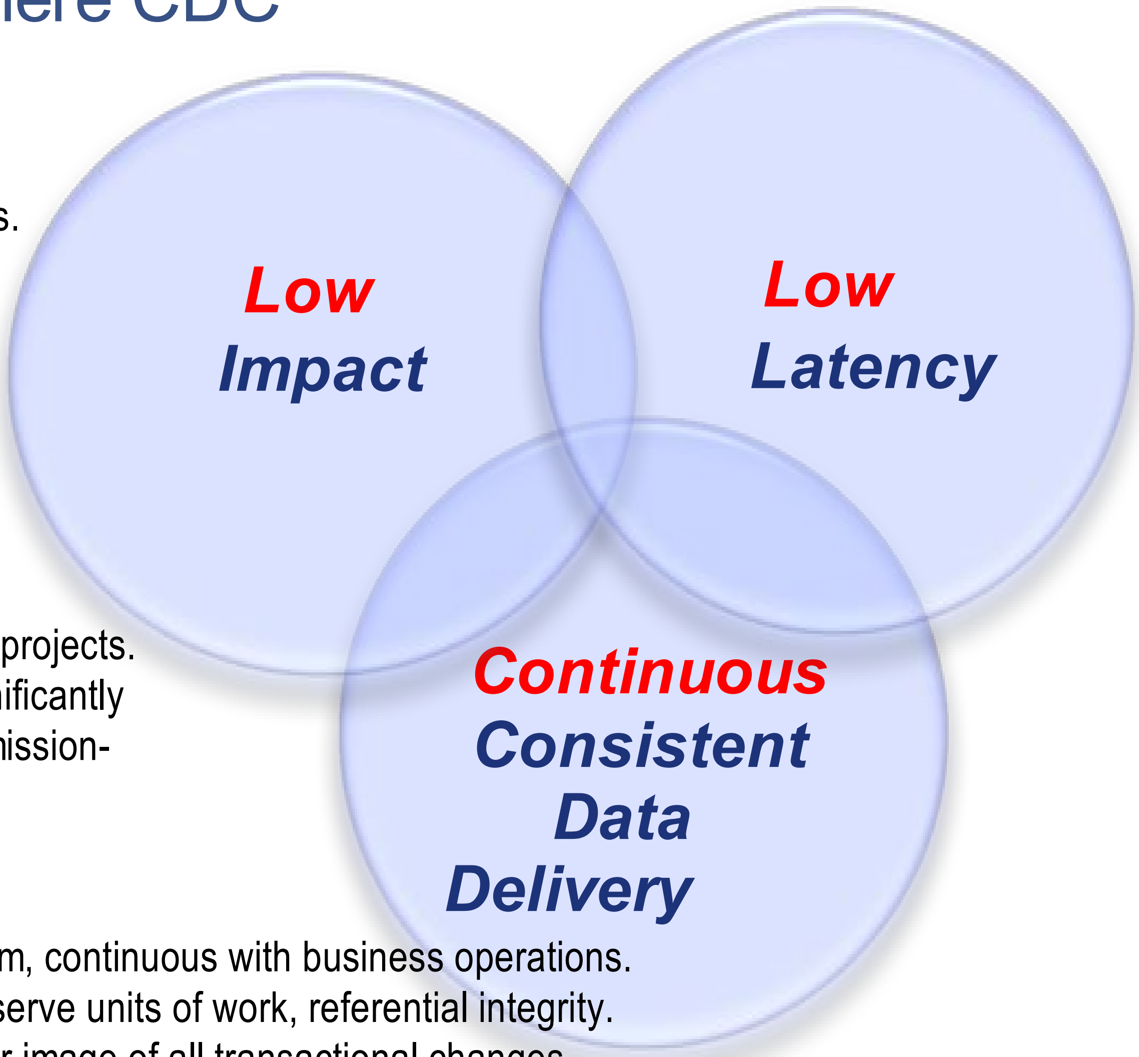
eBusiness Application

Information Server

Message Queue

Target app /DB

# Key elements of InfoSphere CDC



***IMPACT***

1. Reduces risk to operational systems.
2. Non intrusive to applications and databases.
3. Use of native DB logs, very low overhead.
4. No use of database triggers.
5. Management easily integrated into existing IT operations.
6. Help reduce/manage operational windows.

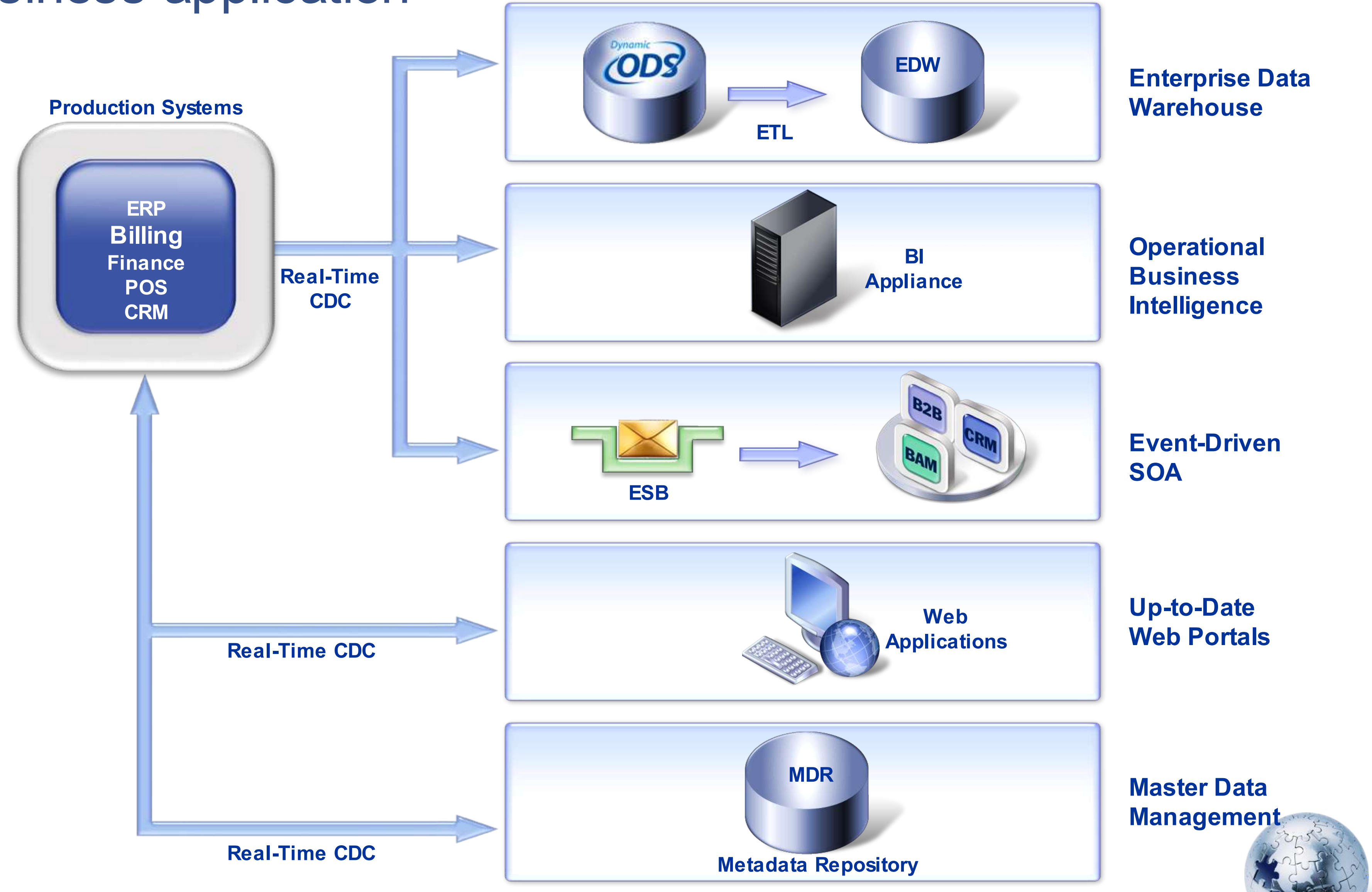
***LATENCY***

1. Near zero latency for pervasive integration projects.
2. ETL can also deliver low latency but at significantly higher impact to production systems and mission-critical applications.

***CONSISTENT DATA DELIVERY***

1. Data pushed, delivered in continuous stream, continuous with business operations.
2. Transaction consistency maintained to preserve units of work, referential integrity.
3. Full transaction granularity, before and after image of all transactional changes.
4. Data event aware, can be used to trigger specific business processes.
5. Fault tolerance, recover to last committed transaction.

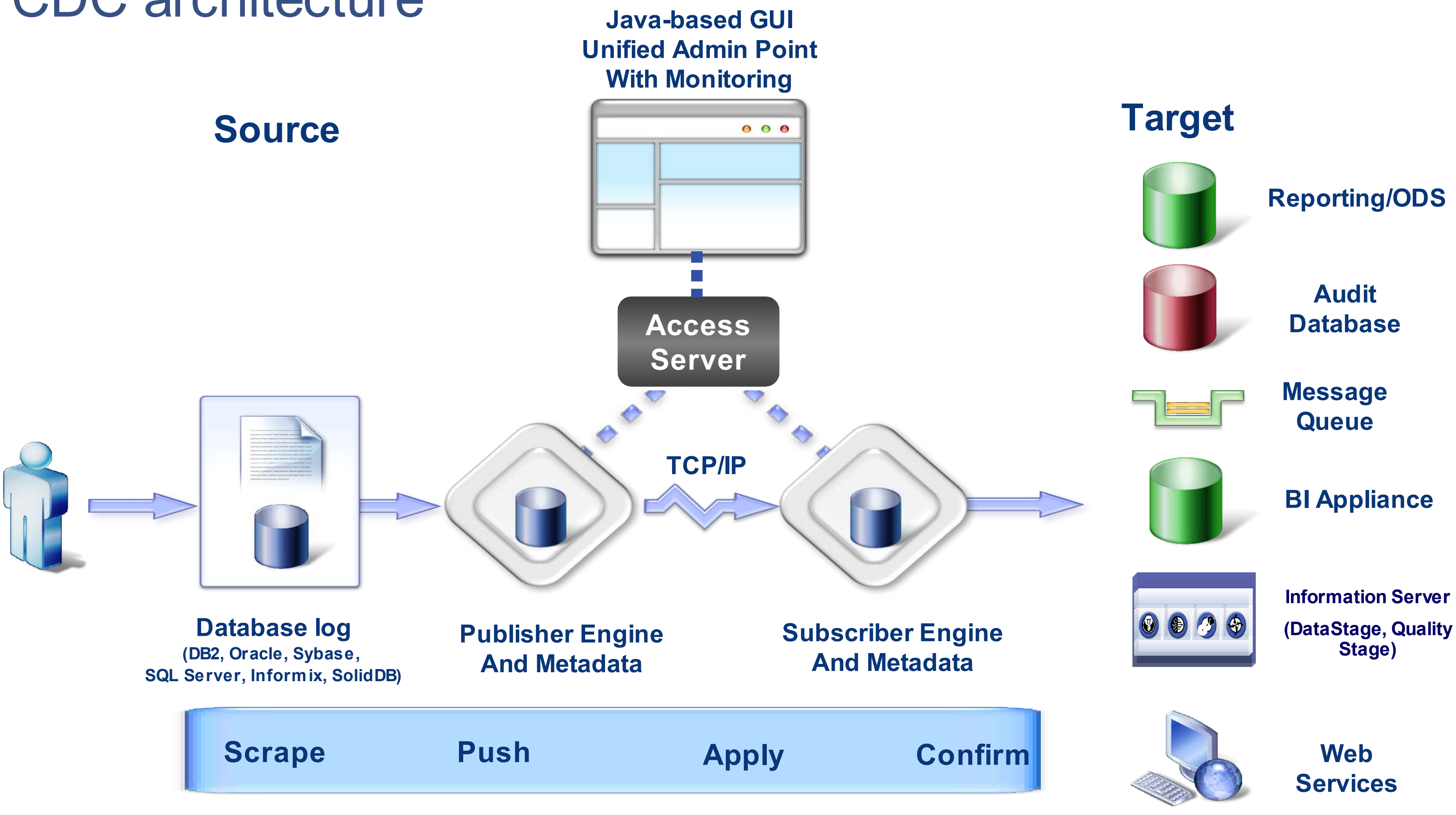
# Business application



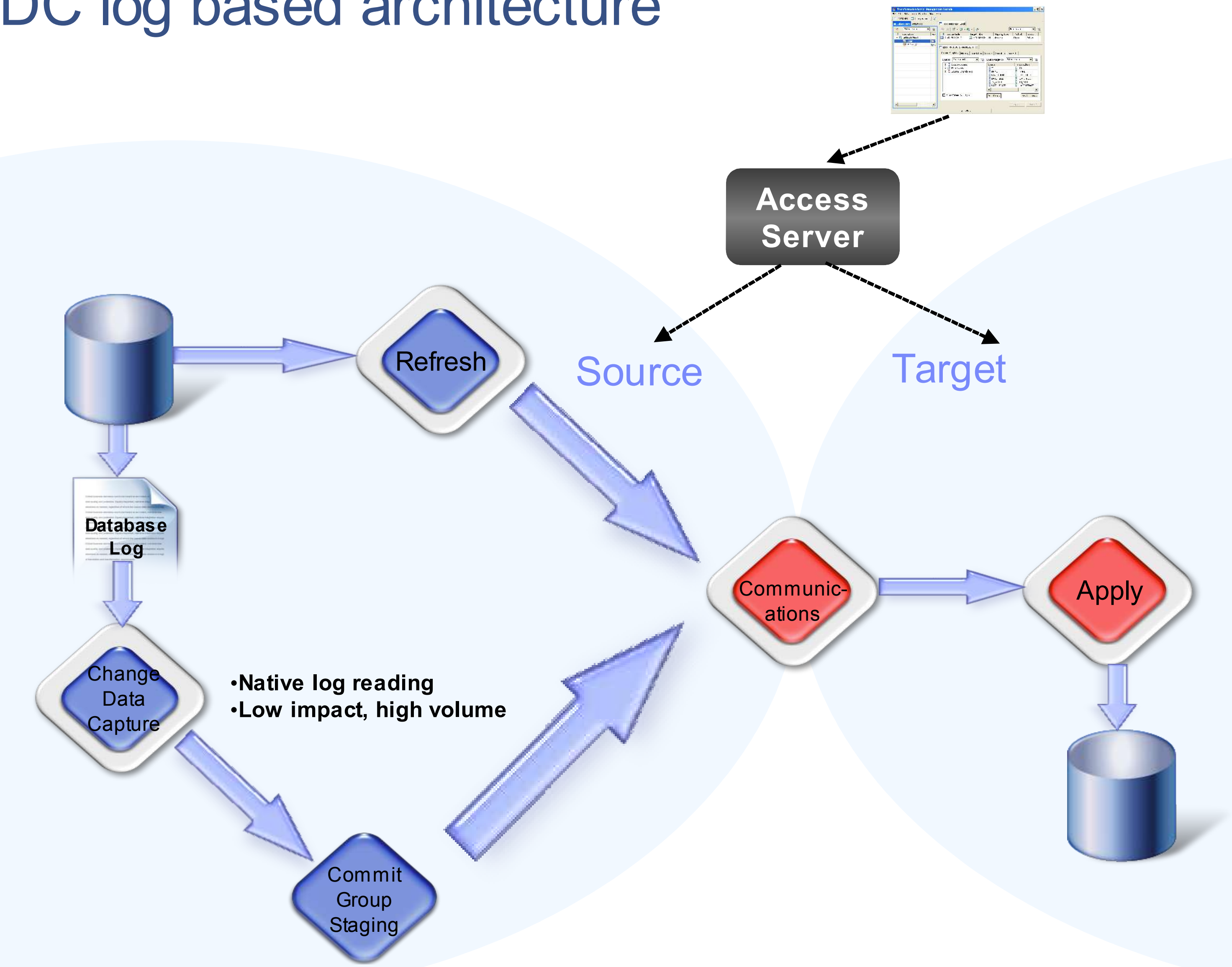
# Key capabilities of InfoSphere CDC

- **Data replication**
- **Near real-time**
- **Multi-platform**
- **Multi-database**
- **Multi-mode**
- **Bi-directional**
- **Conflict detection and resolution**
- **Custom extensions**
- **Column level filtering**
- **Row filtering**
- **Code page conversions**
- **Light transformations**
- **Joins**
- **Journal control columns**
- **Single point of admin**
- **Fault tolerant**
- **And very fast…**

# CDC architecture



# CDC log based architecture



- •Native log reading
- •Low impact, high volume

# CDC fundamentals

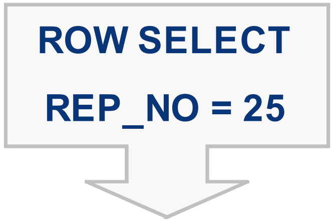
*Across all databases and platforms*

- **Log processing capabilities for major DBMSs**
  - DB2, Oracle, Sybase, SQL Server, Informix, SolidDB
- **Asynchronous push with bookmarking technology**
  - Faster delivery of data because of asynchronous push
  - Non-intrusive to operational data source
  - Transportation of transactions through TCP/IP
  - Keeps Logical Unit of Work (LUoW) when applying transactions to target
  - Bookmark (log position) stored on target along with replicated transactions
    Same Logical Unit of Work
  - Automatic repositioning on restart –
    Mirroring can be interrupted and resumed at will

# Filtering

| CUST_NO | L_NAME | F_NAME | PHONE | REP_NO |
|---|---|---|---|---|
| 58699 | Smith | John | 404-555-3874 | 45 |
| 37283 | Duggan | Ira | 613-555-8367 | 25 |
| 89863 | Quinn | Fran | 905-555-1296 | 11 |
| 89732 | Muntz | Muntz | 704-555-2738 | 25 |

- Integrate entire systems or only a subset of data
- Table/row/column-level filtering options available



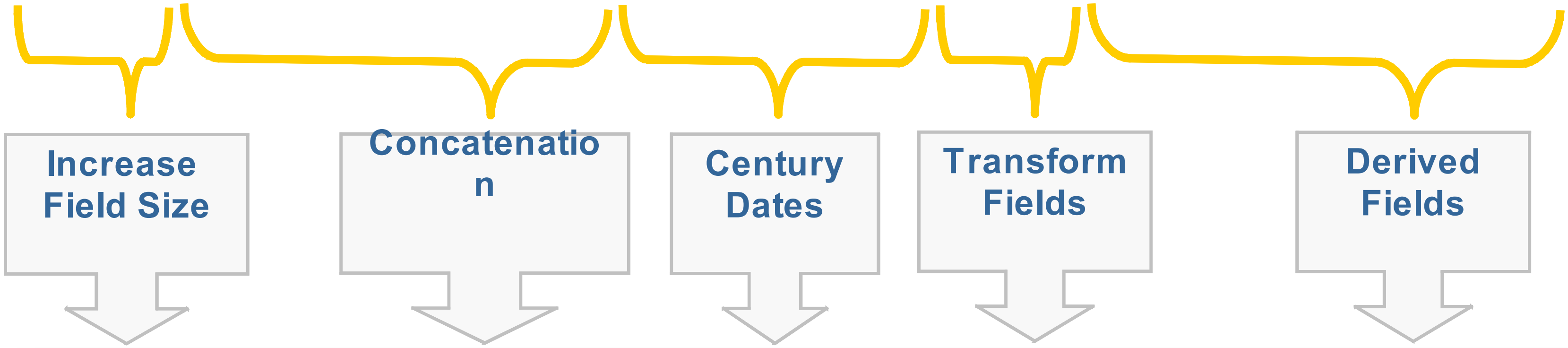
**ROW SELECT**

**REP_NO = 25**

| CUST_NO | L_NAME | F_NAME | REP_NO |
|---|---|---|---|
| 37283 | Duggan | Ira | 25 |
| 89732 | Muntz | Josie | 25 |

# Transformations

| EMP | LAST | FIRST | HIRE_DATE | STAT | SALARY | MAX |
|---|---|---|---|---|---|---|
| 1234 | Moreiro | Nicole | 01/05/97 | A | $55,000 | $60,000 |
| 2345 | Ellison | Val | 04/12/97 | I | $40,000 | $50,000 |



Increase Field Size

Concatenatio n

Century Dates

Transform Fields

Derived Fields

| EMP_ID | FULL_NAME | HIRE_DATE | STATUS | %SALARYMAX |
|---|---|---|---|---|
| 001234 | Nicole Moreiro | 01/05/1997 | Active | 92% |
| 002345 | Val Ellison | 04/12/1997 | Inactive | 80% |

# Journal control information

- Extra audit level information that can be mapped to target table columns.
- Examples
  - &USER to identify user who changed a source transaction in an audit application.
  - &SYSTEM to identify which source system that a transaction originated from when consolidating rows from multiple source systems into a single target table.
  - &TIMSTAMP used by an ETL job to group transactions by date/time.

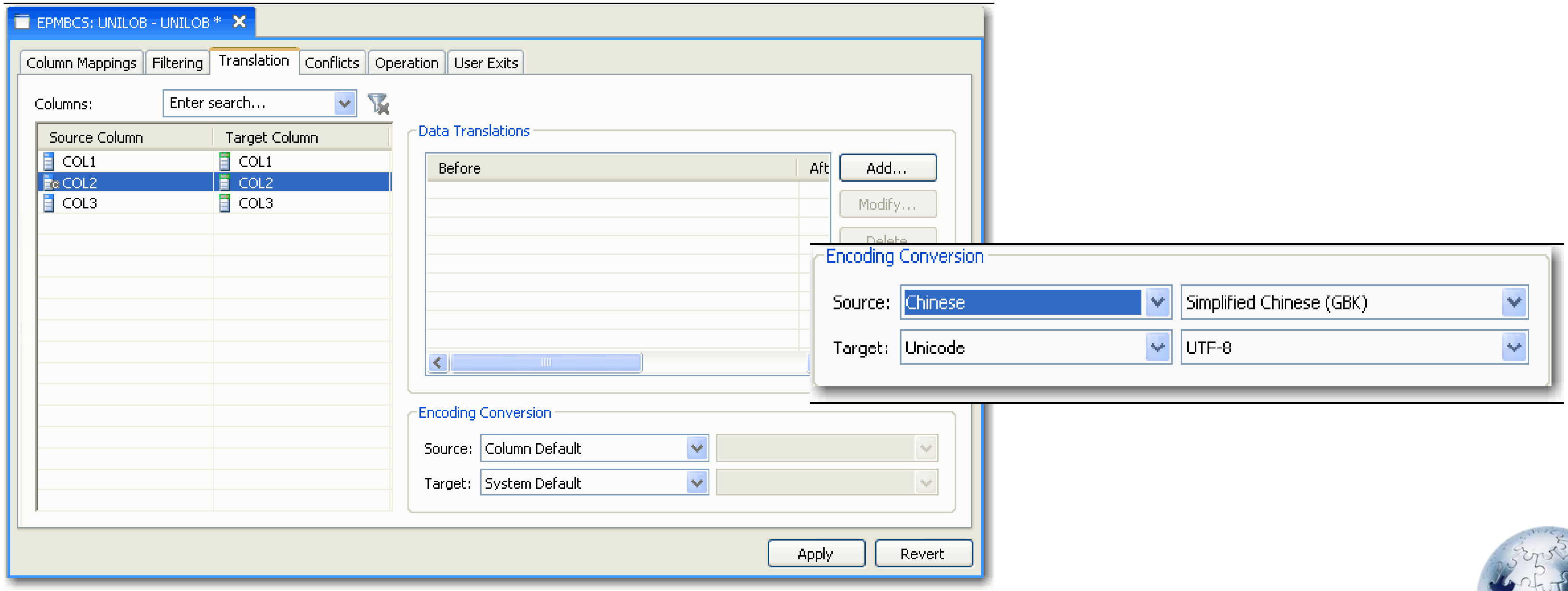
## JOURNAL CONTROL COLUMNS

---------------------------------------------------------------------

| | |
|---|---|
| &CCID | An identifier for the transaction with the update. |
| &CNTRRN | Source table relative record number |
| &CODE | Always "U" for refresh. Always "R" for mirror. |
| &ENTTYP | Indicates the type of update. |
| &JOB | The name of the source job that made the update. |
| &JOBNO | The operating system user Id of the update process. |
| &JOBUSER | The operating system user at the time of the update. |
| &JOURNAL | The name of the journal, as described in Properties. |
| &JRNFLG | Indicates if before image is present |
| &JRNLIB | The name of the journal schema. |
| &LIBRARY | The source table schema or its alias. |
| &MEMBER | The source table name or its alias. |
| &PROGRAM | The name of source program that made the update. |
| &OBJECT | The source table name or its alias. |
| &SEQNO | The sequence number of this update in the journal. |
| &SYSTEM | The hostname of the source system |
| &TIMSTAMP | Time of the update or refresh. |
| &USER | The user ID which made the update. |

# Encoding conversion

- Integrate data from any character encoding
  - Automatic character set conversion
  - User interface driven and managed
  - Conversion is done in-flight (no staging)



Columns: Enter search...

| Source Column | Target Column |
|---|---|
| COL1 | COL1 |
| COL2 | COL2 |
| COL3 | COL3 |

Data Translations

| Before | Aft |
|---|---|

Encoding Conversion

Source: Column Default

Target: System Default

Encoding Conversion

Source: Chinese | Simplified Chinese (GBK)

Target: Unicode | UTF-8

# Complementing ETL solution (1)

- **Adding Change Data Capture to your ETL solution**
  - Minimizes impact on the production server.
    - Access to production data no longer required by ETL
    - CDC runs all the time populating deltas/ODS/flat files/queues
    - Less CPU utilization as only Changed data is sent
  - Minimizes total time to completion of DW/Mart loads.
    - No need to wait for production extracts. Changed Data is delivered already
    - Greatly reduced amount of data to process (changed data only)
    - Enables move to a more real time DW/Mart reporting
  - Certain types of Transformation can be performed in-flight by CDC.
    - Date conversions, Code page conversion, column or row level transformations
    - Reduces the total time to completion as the ETL process will not need to handle the simpler, more mundane types of transformations
  - Enhances the granularity of the information loaded into the DW/Mart
    - Detail information of every individual update can be recorded
    - Improves ability to monitor for business events

# Complementing ETL Solution (2)

- **Methods of data collection by ETL using CDC**
  - Delta database tables
    - Populate delta tables with changed data only
    - Greatly reduced amount of data to process (changed data only)
    - Enables move to a more real time DW/Mart reporting
    - Audit Style (Before Image of an Update is optional) or Last Image only (Adaptive Apply) available
  - Delta Flat Files
    - Populate delta flat files with changed data only
    - Greatly reduced amount of data to process (changed data only)
    - Enables move to a more real time DW/Mart reporting
    - Audit Style (Before Image of an Update is optional) available
    - Very efficient as minimal database handling is involved
  - Message Queues.
    - Default format is XML
    - Greatly reduced amount of data to process (changed data only)
    - Enables move to a more real time DW/Mart reporting
    - Audit Style (Before Image of an Update is optional) available

# Complementing ETL Solution (3)

- **Methods of data collection by ETL using CDC.**
  - ODS with CDC created timestamps
    - Collect data from an ODS using changed timestamp column(s)
    - Can update one column with timestamp when row was updated on source and a second with timestamp when row was updated on target ODS
    - ETL process can use target timestamp to eliminate chance of missing data if replication is running behind
  - Native Integration with Information Server (DataStage/QualityStage)
    - Populate message queues, flat files or direct connect with changed data only
    - Greatly reduced amount of data to process.
    - Enables move to a more real time DW/Mart reporting
    - Audit Style (Before Image of an Update is optional) available

# Audit apply atyle

**Source Table**

| Order # | Order Line # | Item | Qty |
|---|---|---|---|
| 1 | 1 | 20 | 100 |
| 1 | 2 | 30 | 200 |
| 2 | 1 | 10 | 300 |
| Deleted | Deleted | Deleted | Deleted |
| 2 | 3 | 20 | 200 |
| 10000 | 1 | 50 | **1000** |
| ~~10001~~ | ~~1~~ | ~~15~~ | ~~100~~ |
| 10000 | 1 | 50 | 500 |
| | | | |

**Target Audit Table**

| Order # | Order Line # | Item | Qty | Change Date | Sourc e Action | Trans . # | Row # |
|---|---|---|---|---|---|---|---|
| 10001 | 1 | 15 | 100 | 21/09/06 10:56:22 | PT | 1 | 82638 |
| 10000 | 1 | 50 | **500** | 21/09/06 10:57:02 | UB | 3 | 82637 |
| 10000 | 1 | 50 | **1000** | 21/09/06 10:57:02 | UP | 3 | 82637 |
| ~~2~~ | ~~3~~ | ~~20~~ | ~~200~~ | ~~21/09/06 11:02:43~~ | ~~DL~~ | ~~3~~ | ~~81732~~ |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |

- **Audit table starts off empty**
- **Insert on the source will result in an insert into the target Audit table**
- **Update on the source will result in an insert of both the Before Image and the After Image of the update into the target Audit table**
- **Delete on the source will result in an insert into the target Audit table**

# Advantages of using Audit apply style

- **Reduces ETL processing time**
  - Loading the Data Ware House / Data Marts will complete faster reducing the load on the Source batch window
- **Ability to capture the BEFORE Image if the source application allows for key value changes**
  - Required, to ensure that the appropriate rows in the DW are processed
- **Timestamps, Trans #, or Sequence number can be used by the ETL process for ordering and grouping of the data elements within their scripts**
- **Source Action Taken can be used by the ETL process to determine what the appropriate action should be within the DW or Marts**
  - An Insert may be used to add to an already summarized total
  - An Update may be used to subtract (before image) and add (after image) to an already summarized total
  - A Delete may be used to subtract from an already summarized total
- **Ability to create Real Time or Active Data Warehousing**
  - Increase the frequency that ETL collects the data

# Adaptive apply style

**Source Table**

| Order # | Order Line # | Item | Qty |
|---|---|---|---|
| 1 | 1 | 20 | 100 |
| 1 | 2 | 30 | 200 |
| 2 | 1 | 10 | 300 |
| 2 | 2 | 40 | 400 |
| 2 | 3 | 20 | 200 |
| 10000 | 1 | 50 | **1000** |
| 10001 | 1 | 15 | **500** |
| | | | |

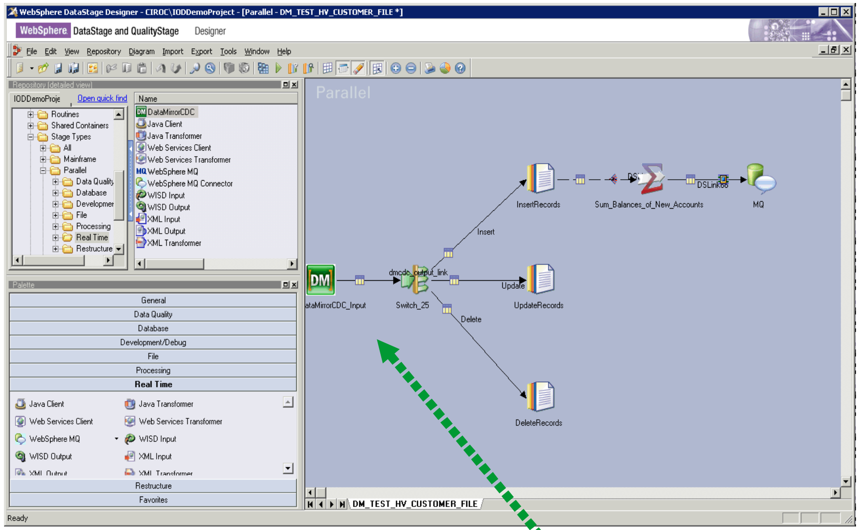
**Target Delta Table**

| Order # | Order Line # | Item | Qty | Change Date | Source Action | Trans. # | Row # |
|---|---|---|---|---|---|---|---|
| **10001** | **1** | **15** | **500** | **21/09/06 11:02:37** | **UP** | **4** | **82638** |
| 10000 | 1 | 50 | **1000** | 21/09/06 10:57:02 | UP | 3 | 82637 |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |

- **Delta table starts off empty**
- **Insert on the source will result in an insert into the target Delta table**
- **Update on the source will result in an update of appropriate row on the target Delta table (if row exists otherwise the row will be inserted)**
- **A second Update to the same row on the source will result in an update of the appropriate row on the target Delta table**

# InfoSphere CDC integration with DataStage/QualityStage

*InfoSphere* *Change Data Capture*



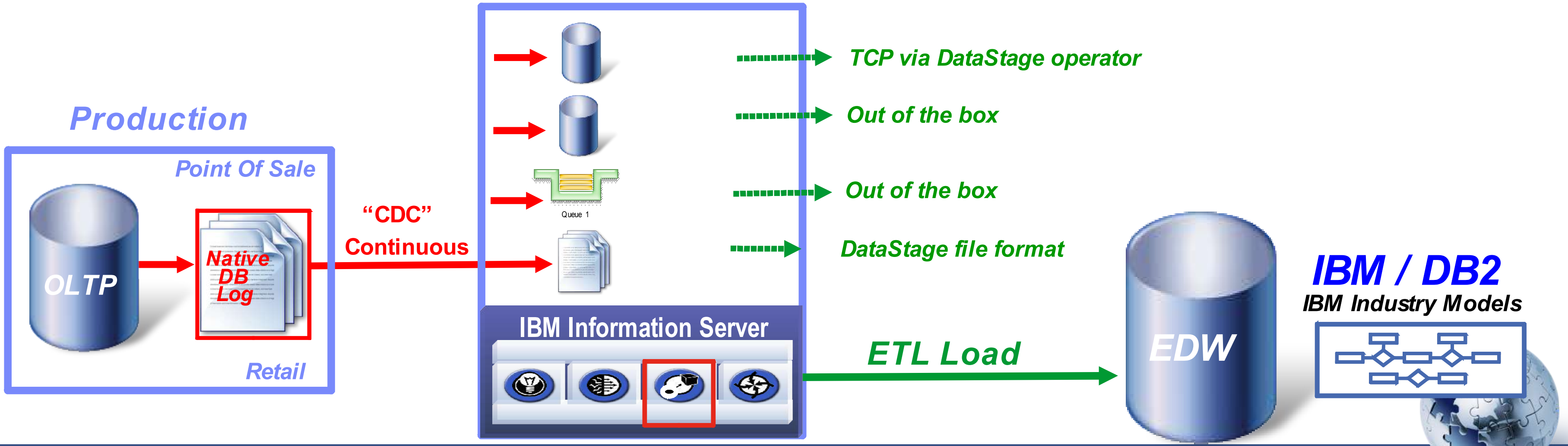
## Technical Benefits

- Extended breadth of native CDC source coverage
- DataStage job generation from CDC definition
- Planned metadata integration for data lineage & impact analysis
- Flexibility – 4 option to stream changes to DS/QS

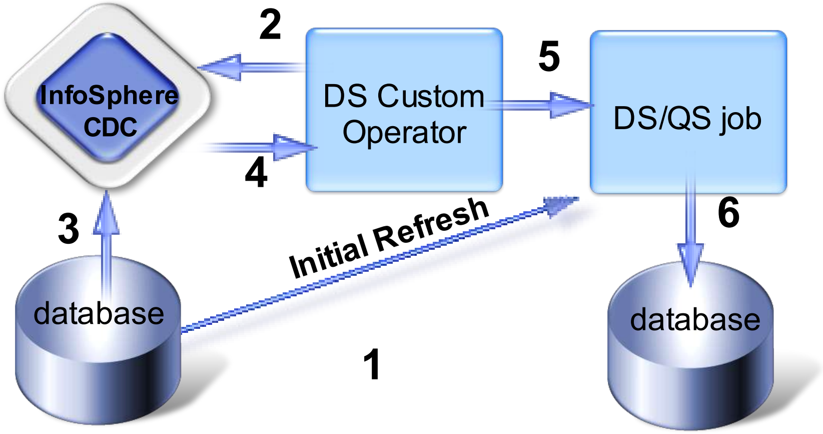
## Busines Benefits

- Reduces risk on operational systems (low impact)
- Allows IT to better manage batch windows
- Can explore benefits of continuous ETL
- Can explore benefits of low latency data delivery to lines of business

*DataStage Consumption*

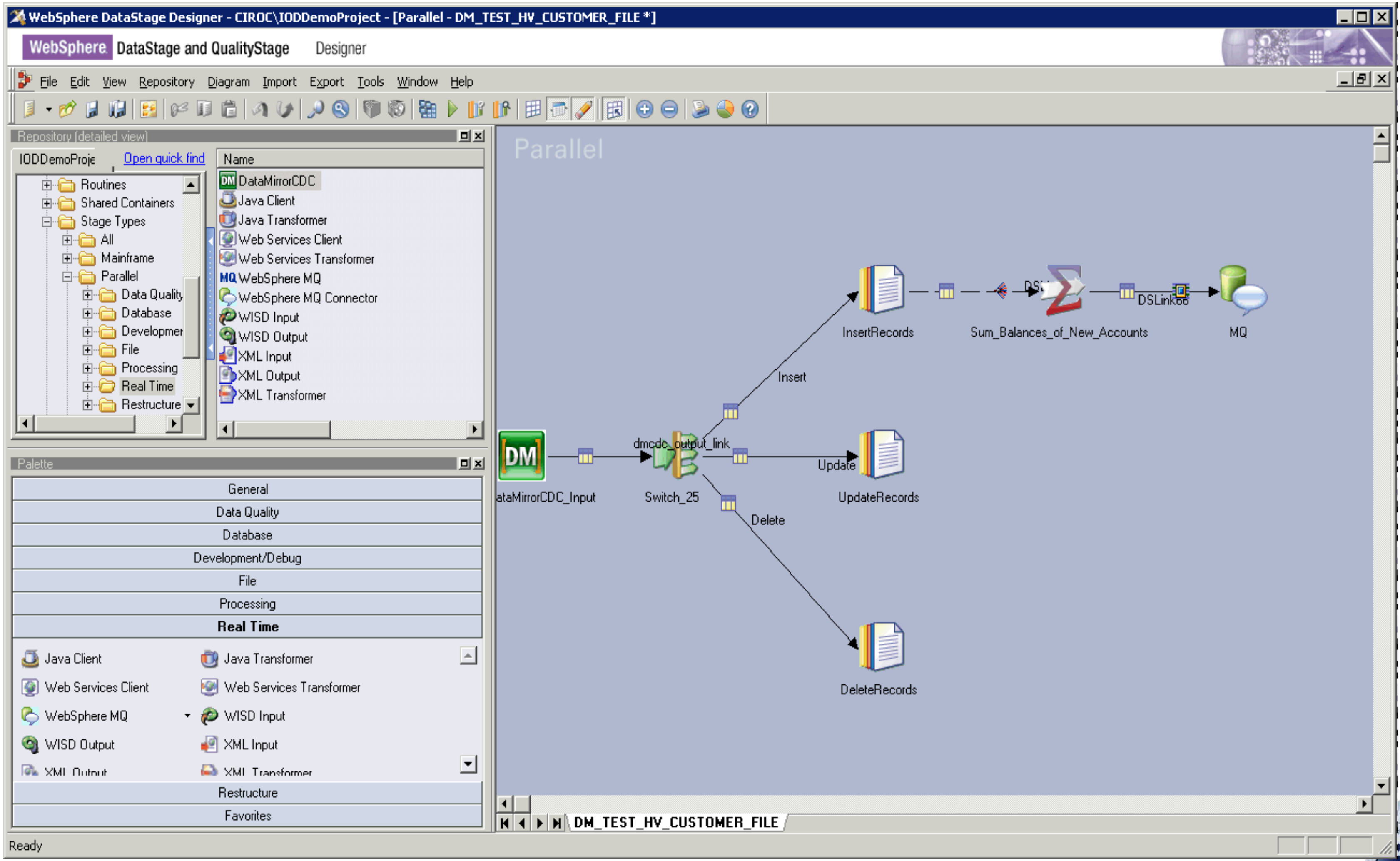


# Direct Connect

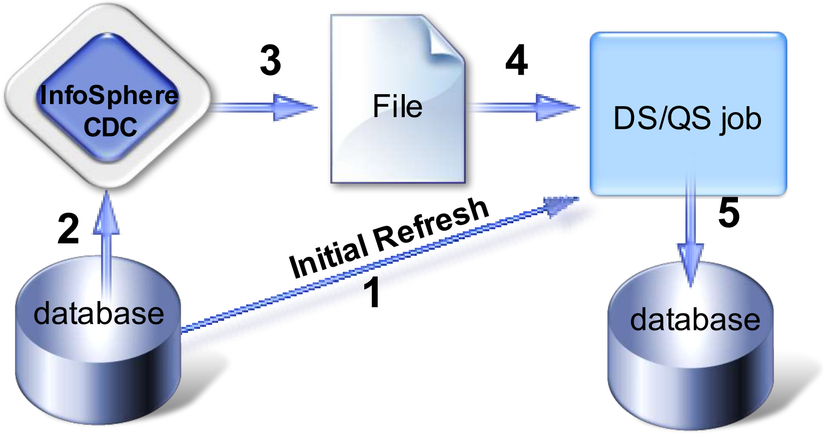


1. DataStage extracts data from source database using standard ETL functions
2. Custom operator, which runs on regular intervals, requests the changed data from CDC
3. CDC captures/collects changes made to remote database
4. Captured changes passed to custom DS operator (listens to TCP/IP port)
5. Custom operator passes data off to downstream stages
6. Update target database with changed data

# Direct Connect example

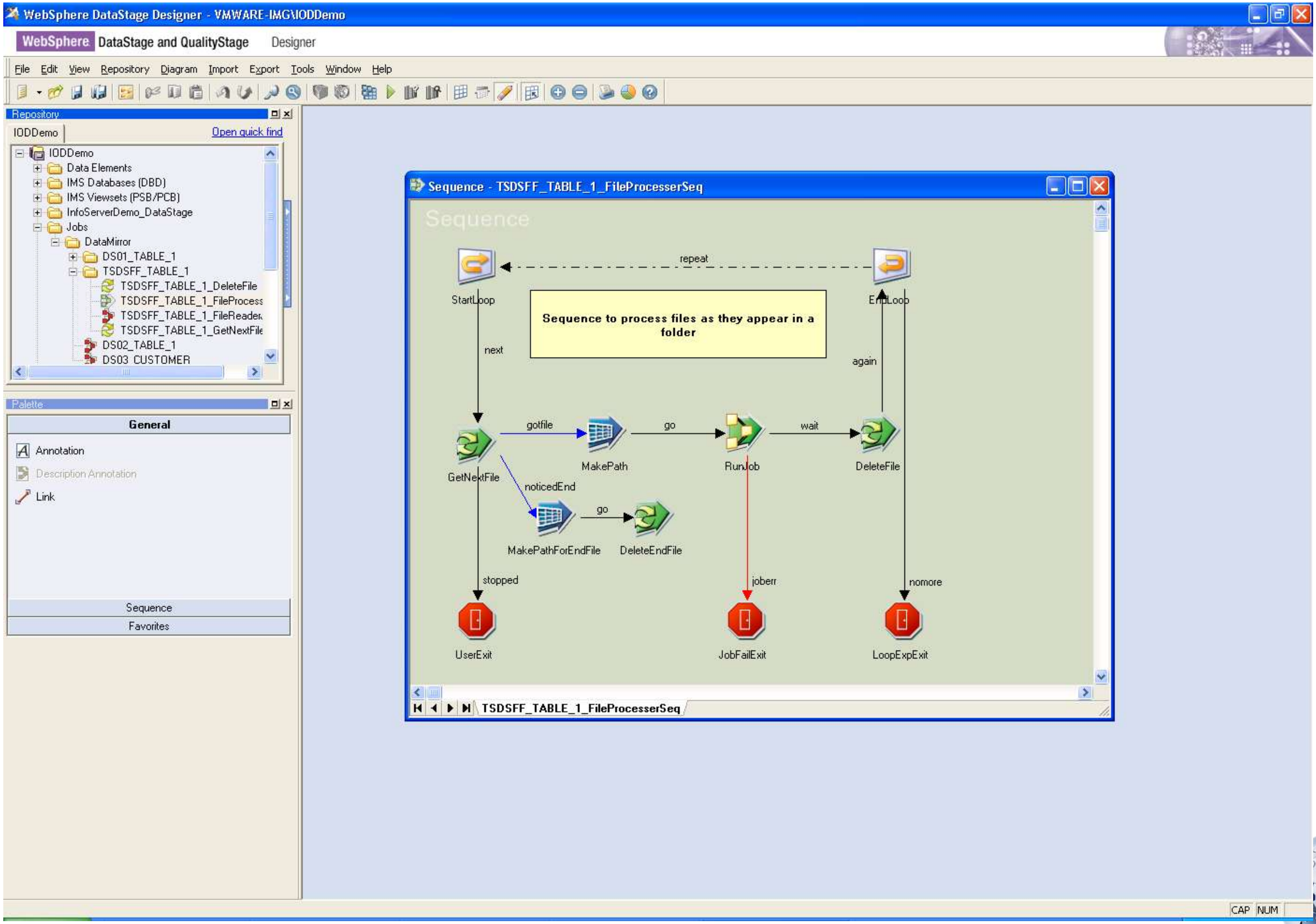


# Flat file based

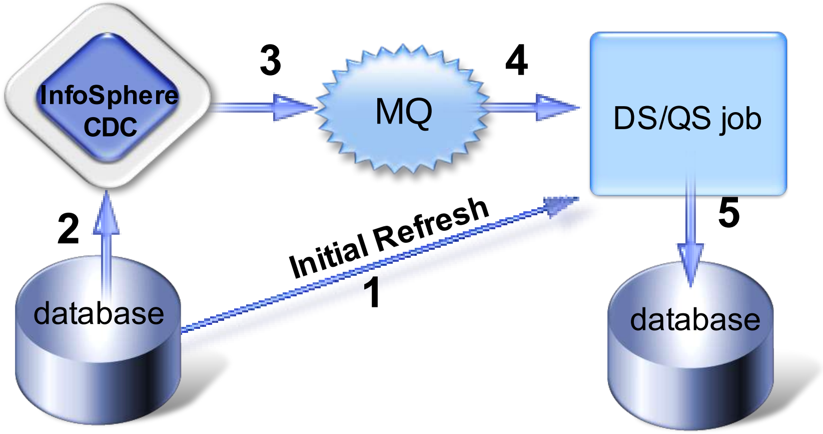


1. DataStage extracts data from source database using standard ETL functions
2. CDC captures changes made to source database via database log
3. CDC writes each transaction to a file
4. DataStage reads the changes from the file
5. Update target database with changes

# Flat file example



# MQ based integration

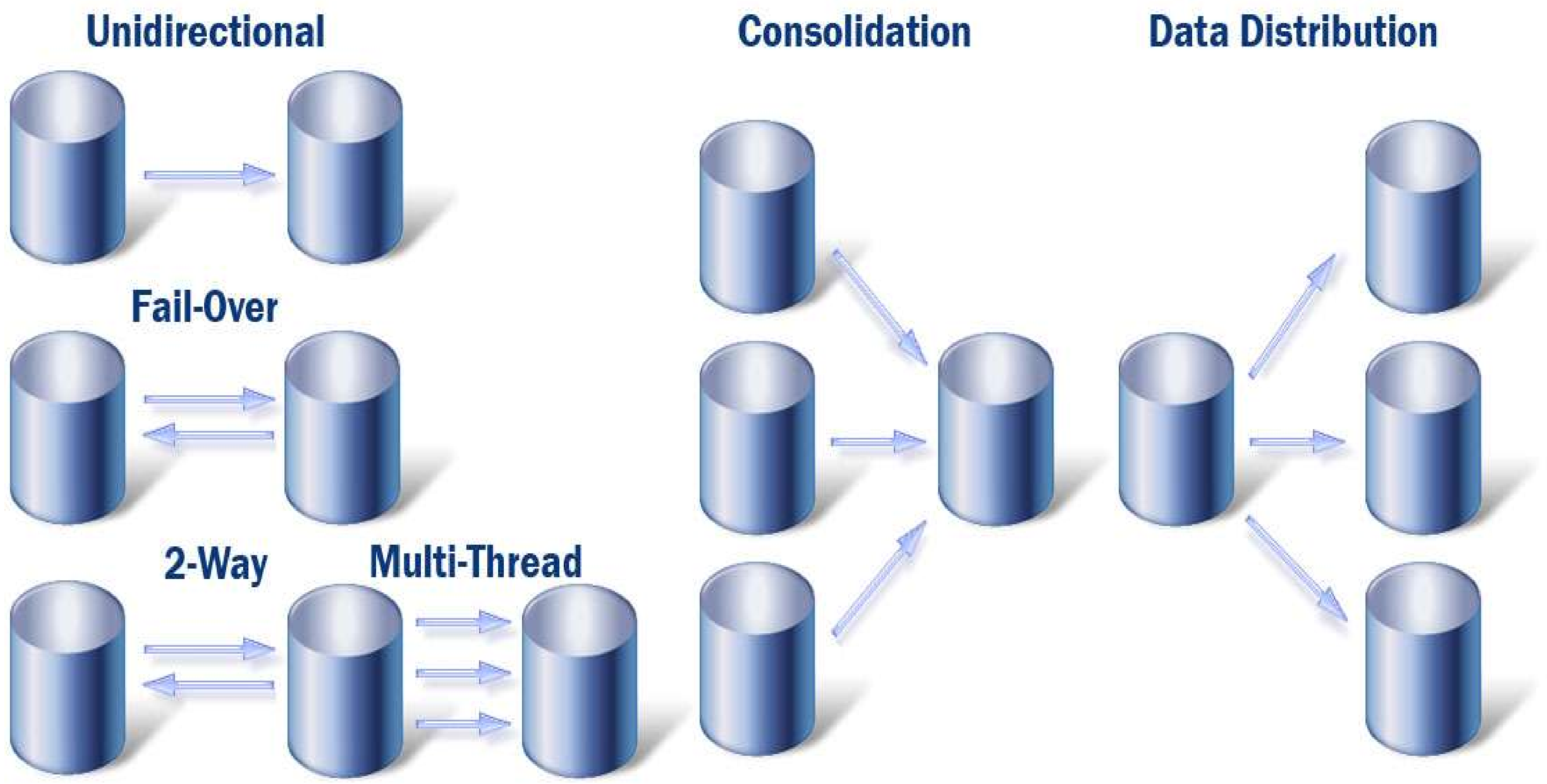


1. DataStage extracts data from source database using standard ETL functions
2. CDC captures changes made to source database via database log
3. Captured changes written to MQ
4. DataStage (via MQ connector) processes message and passes data off to downstream stages
5. Updates written to target database
6. New DataStage Distributed Transaction Stages

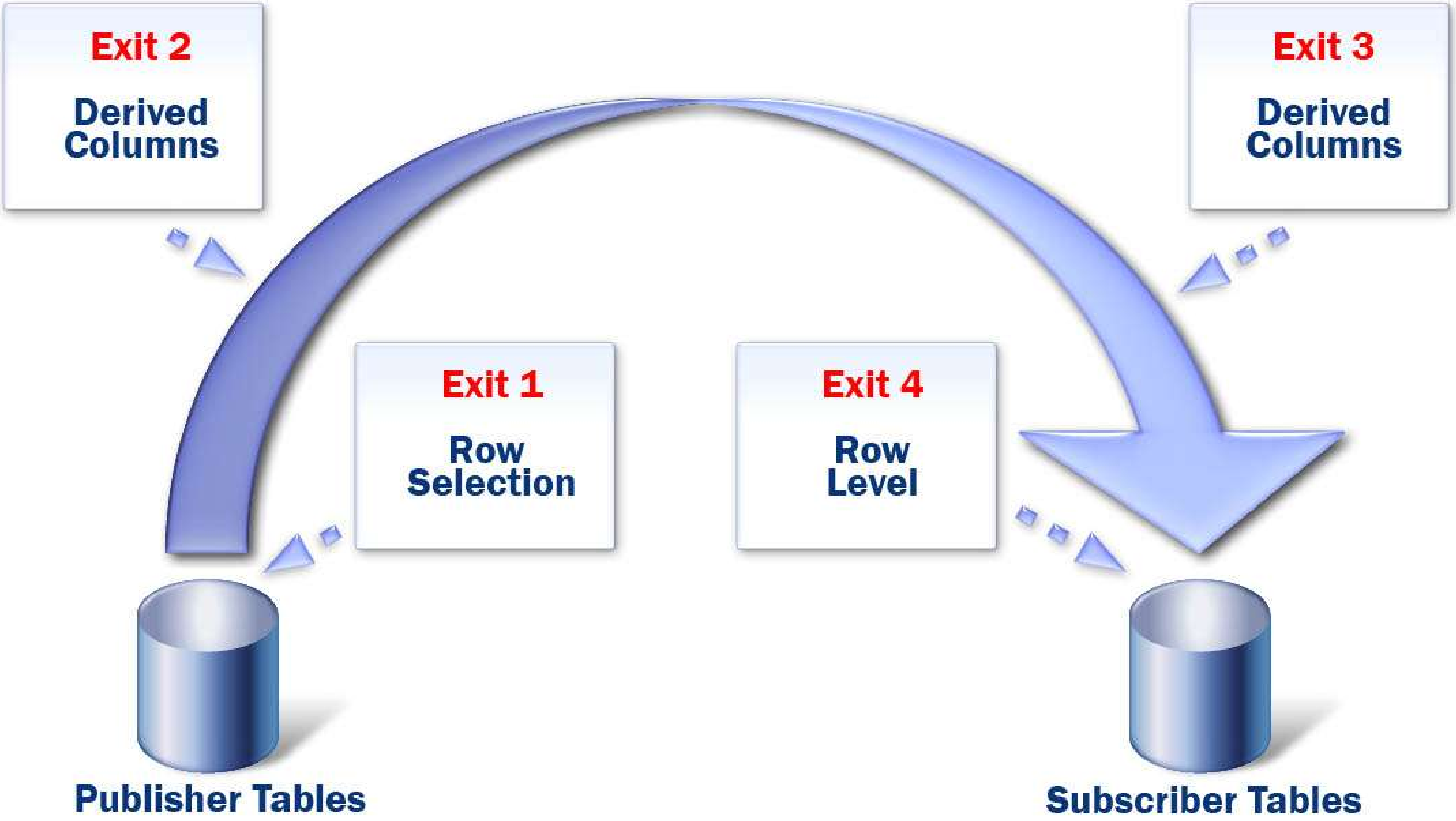
# Modes of replication

- **Continuous mirroring**
  - Changes read from database log.
  - Apply change at the target as soon as it is generated at the source.
  - Replication job remains active waiting for next available log entry.
- **Periodic mirroring**
  - Changes read from database log.
  - Apply net changes on a scheduled basis.
  - Replication job ends when available log entries are processed.
- **Refresh**
  - File/table level operation.
  - Apply a snapshot version of source table.
  - Typically used to achieve initial synchronization of source and target table.

# Flexible implementation



# User exits


Exit 2
Derived Columns
Exit 3
Derived Columns
Exit 1
Row Selection
Exit 4
Row Level
Publisher Tables
Subscriber Tables

InfoSphere™

# Thank You

Questions and discussion

***Bu sunum 25 Haziran 2009 tarihinde Kuruçeşme Divan'da yapılan Gerçek Zamanlı Güvenilir Veri Entegrasyonu toplantısı için hazırlanmıştır.***

***http://www.ibm.com/software/tr***

***http://www.ibm.com/software/tr/data***