Z Data Tools and APIs Overview

Mainframe Data Access Modernization Solution

HCL SOFTWARE

Content

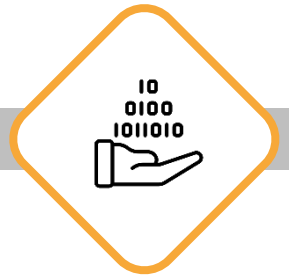
The Problems / Challenges

- *80%* of the world's corporate data still resides on mainframe systems today (H. Tessler, Enterprise Systems Media).
- Significant % of mainframe workloads are written in *COBOL, PL/I or Assembler* language – data layout defined in those languages.
- *Monolithic system* with decades of incremental enhancements.
- *Highly risk averse* due to mainframe workloads typically represent the most important enterprise systems (e.g. core banking systems) and cost sensitive.
- *Skills shortage* with new generation IT professionals gravitated towards new technology (e.g. REST APIs and Java).
- *Difficult to discover and integrate* existing mainframe assets with the rest of enterprise systems.

ZDT Overview

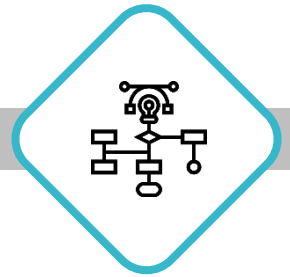
- HCL Z Data Tools (ZDT) is a set of tools that help you to manipulate data stored on z/OS systems interactively and in batch processing.
- Designed to deal with large production data efficiently while protecting integrity and privacy.
- Access to z/OS data made simple by its intuitive and consistent user interface across a wide range of data stores available on the platform.
- Access to data across multiple z/OS systems / LPARs made simple by its remote systems support.
- A must have tool for every z/OS user including system programmers, application developers, ISPF users and much more.
- Comprehensive support for a wide range of data store types:
  - *Data Sets (VSAM, QSAM, PDS, PDSE, AIM, OAM)*
  - *DB2*
  - *IMS*
  - *WebSphere MQ*
  - *CICS*
  - *Hierarchical File System under UNIX System Services*

ZDT Advantages



ZDT can provide a generic data access solution without programming

- A wide range of file types are supported out of box – no programming required.
- ZDT enables manipulation of data interactively or in batch.



ZDT templates describe logical layout of data and allows transformation without programming

- Templates are created from COBOL, PL/I and Assembler COPYBOOKs or describe layouts in XML format.
- Describe most complex layout and data structures using the template's criteria infrastructures. For example, segmented records, nested REDEFINEs and PL/I arrays.
- Change column orders and hide columns based on your requirements.
- Specify criteria to select a subset of data you are interested to work with.

ZDT template repository for retaining mainframe knowledge for the future

- Template repository contains a list of mapping between data source and matching template to use.
- This enables new users to start working with your mainframe data quicker.
- Once the repository is setup by expert, ZDT leverages the information during its operation to simplifies tasks for end users all around.

ZDT saves cost by powerful set of tools

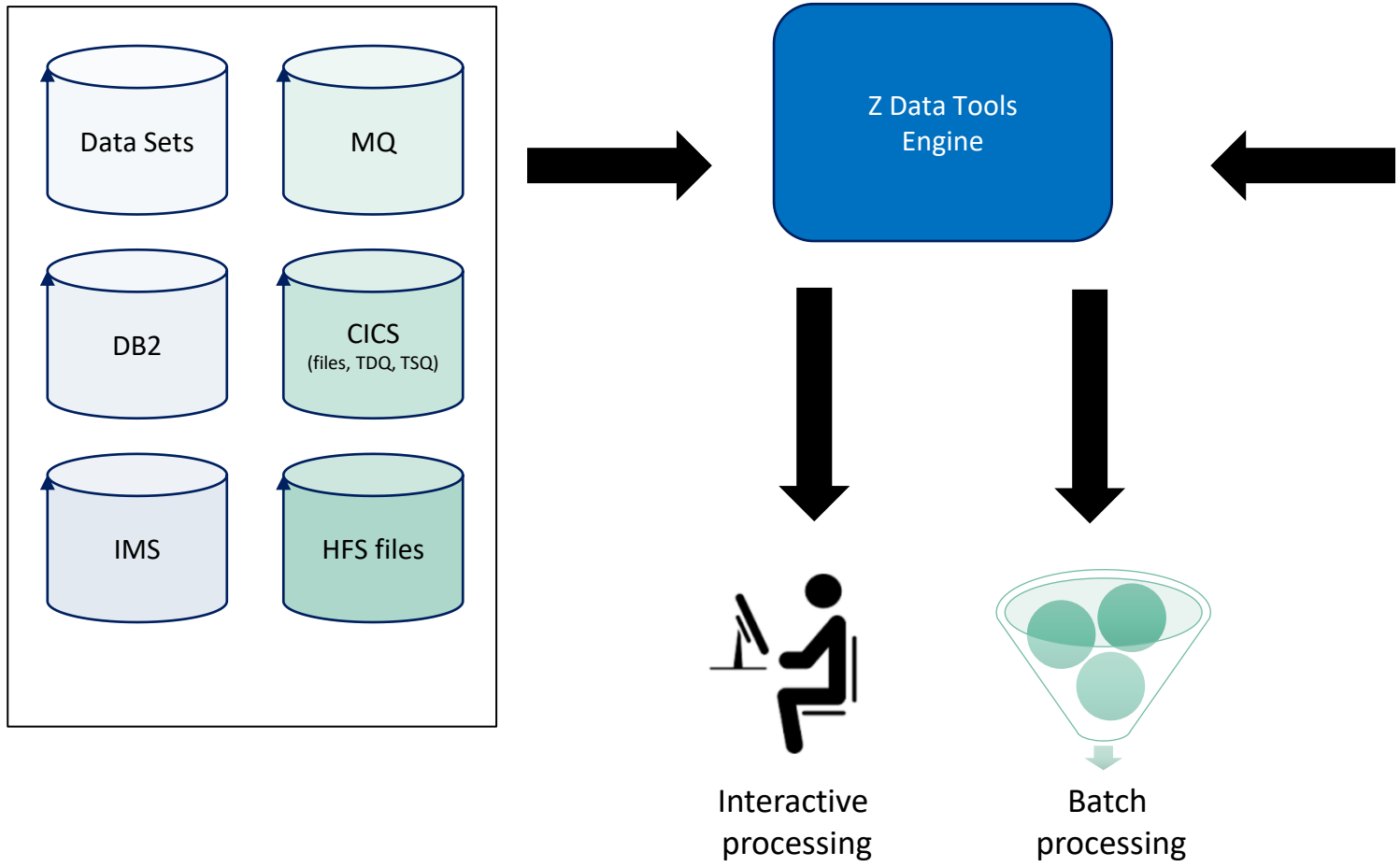
- ZDT provides a powerful set of batch functions such as copy, compare, edit, export, load, print and much more.
- ZDT's remote systems support enables access to data on remote systems as well while hiding much of mainframe specific complexity.
- Replace in-house apps with powerful ZDT functions for improved efficiency & ease of maintenance.

ISPF user friendly productivity improvement features

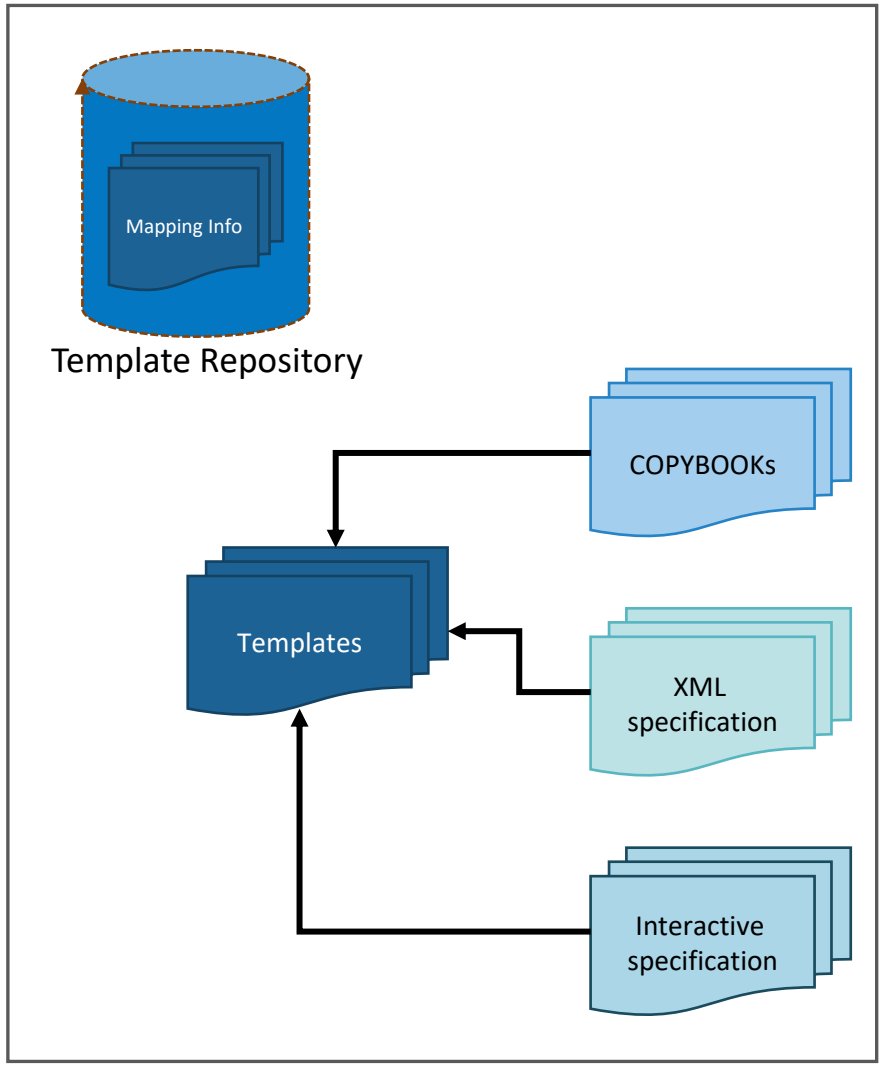
- Specify a group of resources to work with on a local system as well as remote systems.
- Search text and change across the specified group of resources with powerful navigation panel display.
- Clone the specified group of resources to create backup.
- Edit JCL with ease.

How does it work?

Data to be processed



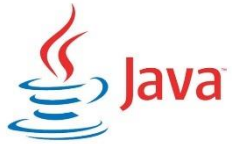
Logical Layout of Data



HCL SOFTWARE

Mainframe Data Access Modernization Challenges

Java™ on Mainframe



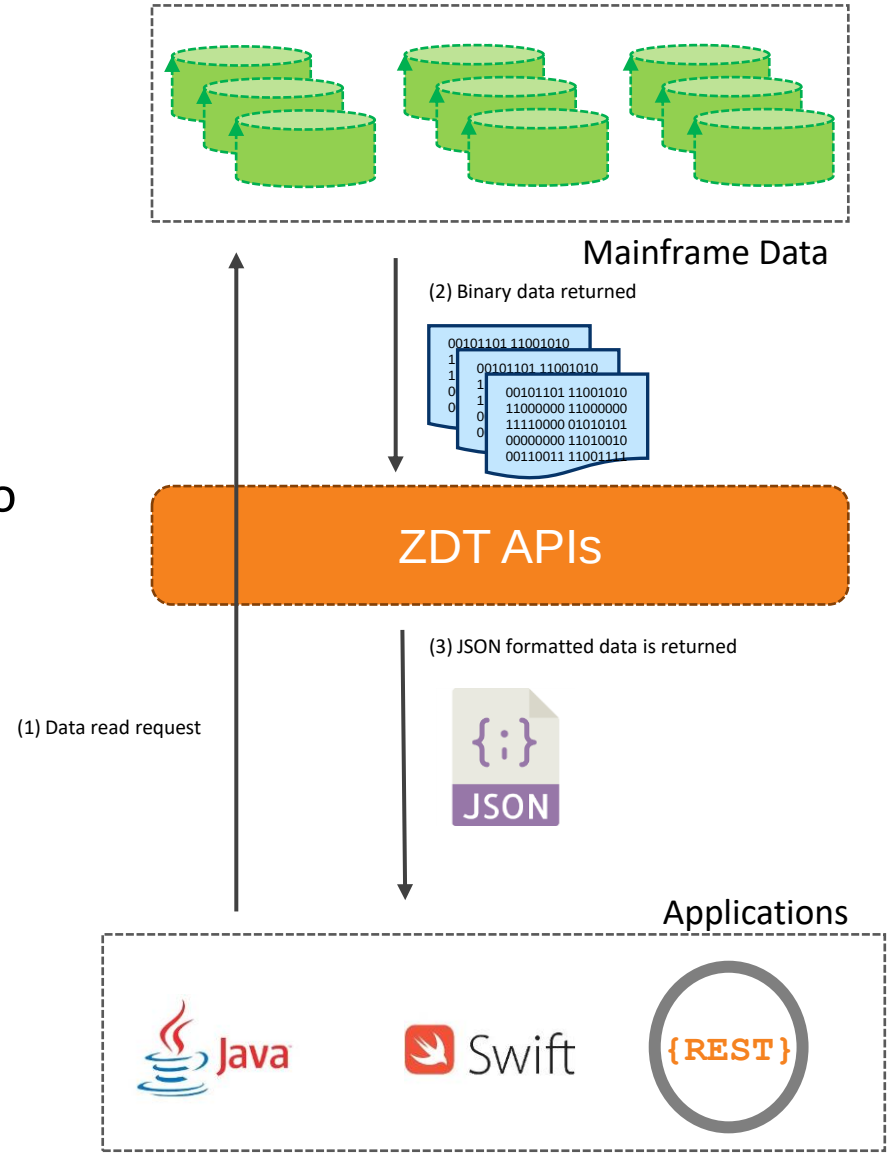
- On z/OS® since 2000.
- IBM® z14 announcement: performance improvement by exploiting new hardware.
- CICS® TS V5.4 announcement: *Support for applications written to the Java™ EE 7 full platform specification.*
- IMS™ 15 announcement: *More expansive and dynamic Java development experience.*
- Offload MIPS to IBM z Integrated information Processor (zIIP) engine.

Challenges

- Most data stored on the platform are defined by COBOL, PL/I and Assembler programs.
- Mainframe data structures typically do NOT map directly to Java and other modern languages.
- Steep learning curve for non-mainframe developers to understand the structure and manipulate mainframe data.
- New languages on the platform.

ZDT APIs Overview

- HCL Z Data Tools APIs (ZDT APIs) is an add-on product which provides Java™ and Swift application programming interfaces (APIs) to manipulate mainframe data.
- APIs leverages ZDT's template infrastructure to understand logical layout of mainframe binary data .
- Mainframe binary data is converted into JSON format and returned to API consumers in Java or Swift for easier interpretation and manipulation for developers with no mainframe background.
- Usage scenario includes but not limited to:
  - Replacement of legacy applications for batch processing of data in Java.
  - Integration of traditional mainframe apps with new apps.
  - Interpretation of mainframe operational data for analytics (eg. SMF data).
  - Manipulation of mainframe data in virtual storage.



HCL SOFTWARE

Use Cases for ZDT APIs

Application Modernization

- Develop new applications in Java™ and Swift Programming language
- Build new business logic with JSON & RESTful API's.

Replacement of legacy applications for batch processing of data in Java

- MIPS reduction (offload to speciality engines zIIP/ IFL's)
- ZDT backend allows for processing of 'always online' CICS resources

Integration of traditional mainframe apps with new apps and data stores

- Traditional mainframe data sources enabled via ZDT API for new or extended use cases, including reflecting data to other data platforms.

IT Operation Analytics

- Interpretation of mainframe operational data for analytics (e.g SMF data).

Manipulation of mainframe data in virtual storage

- For unusual resource types or manipulations

JSON Example

COBOL COPYBOOK

```
01 REC-TYPE01.
    03 REC-TYPE                PIC XX.
    03 REC-ID REDEFINES REC-TYPE  PIC XX.
    03 NAME                    PIC X(20).
    03 EMPLOYEE-NO             PIC 9(4) BINARY.
    03 AGE                     PIC 9(4) BINARY.
    03 SALARY                  PIC 9(7) PACKED-DECIMAL.
    03 MONTH    PIC 9(8) BINARY OCCURS 12 TIMES.
    03 FILLER                  PIC XX.
```

JSON produced:


```
{
 "layouts": [
  {
   "REC-TYPE01": {
    "REC-TYPE": "01",
    "NAME": "Andrew Apple",
    "EMPLOYEE-NO": 6645,
    "AGE": 53,
    "SALARY": 78500,
    "MONTH": [
     30,
     22,
     46
            …
    ],
    "FILLER": "\u0000\u0000"
   }
  }
 ]
}
```

Summary

1. Powerful set of tools to manipulate mainframe data interactively and in batch
2. Powerful framework to represent logical layout of mainframe data (templates)
3. Knowledge retention of your mainframe data through usage of ZDT template repository – all ZDT function leverages the repository to simplify usage even for new users
4. Productivity improvement features for ISPF users
5. Mainframe data access modernization using ZDT APIs in Java™ and Swift languages as well as REST APIs – simplifies application development especially for developers with limited mainframe experiences
6. Continued enhancement to meet mainframe data manipulation requirements today and future

HCL

Relationship™

BEYOND THE CONTRACT

$8.4 BILLION ENTERPRISE | 132,000 IDEAPRENEURS | 44 COUNTRIES

WATCH THE FILM