



z/TPFDF V1.1

TPF Users Group Spring 2008 SDO Access to z/TPFDF Databases

Name: Sasha Krymer
Venue: Main Tent

AIM Enterprise Platform Software
IBM z/Transaction Processing Facility Enterprise Edition 1.1.0

Any reference to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.

© 2008 IBM Corporation

Agenda

- **The Data Access Problem**
- **Service Data Objects (SDO) and Service-Oriented Architecture (SOA)**
- **SDO and z/TPFDF**
- **Metadata**
- **Security**
- **Configuration**
- **z/TPFDF SDO Data Collection**
- **Application Programming Interface**
- **Application Development**
- **APAR Information**
- **Additional Information**

The Data Access Problem

- **Access z/TPFDF data remotely**
 - Within Service-Oriented Architecture (SOA)-based environment
 - Standard data format
 - Simple access technique
 - Simple data manipulation
 - Means for data exchange between applications

The Data Access Problem (cont.)

- **Solution**

- Use Service Data Objects (SDO)

- Standard data format for business data in SOA
- Standard application programming interface (API) for data manipulation
- Allows both data retrieval and update – just 2 new APIs needed!
- Serializable
 - Easy to transmit and/or store

The Data Access Problem (cont.)

- **Benefits of SDO-based solution**
 - z/TPFDF - hierarchical data representation and SDO flexibility enables simple z/TPFDF-SDO mapping
 - Generic, well-defined, object-oriented interface
 - Part of SOA

The Data Access Problem (cont.)

- **SQL-based approach considered, however:**
 - No clear-cut, easy-to-use mapping to hierarchical data
 - Needs additional configuration artifacts and artificial data structures
 - Not tied to the data organization, like SQL to relational databases

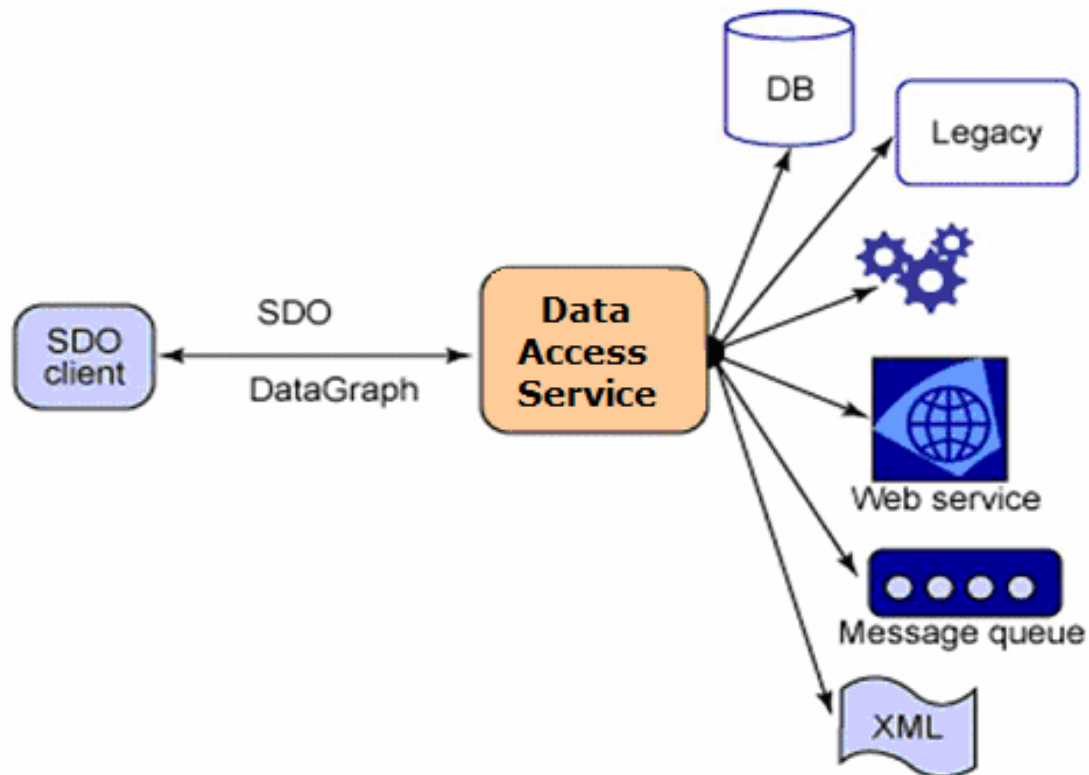
SDO and SOA

- **New model of data access**
- **Complementary technology for SOA**
- **Developed jointly by IBM and BEA**
- **Standardized using Java Specification Request (JSR) 235**
- **Supported by Open SOA**
 - <http://www.osoa.org/>
 - IBM, BEA, Oracle, SAP, Sun, Sybase, etc

SDO and SOA (cont.)

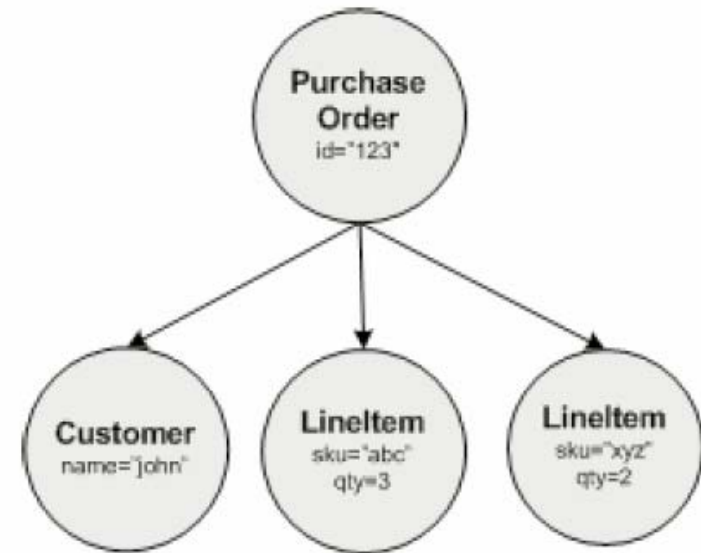
- **Convenient and generic way to access data**
 - Universal model for business data
 - Common unifying format for exchanging data between services
 - Includes dynamic interfaces
 - Not tied to the data organization, like SQL to relational databases
 - Object-oriented, thus maintenance is easier

SDO and SOA (cont.)



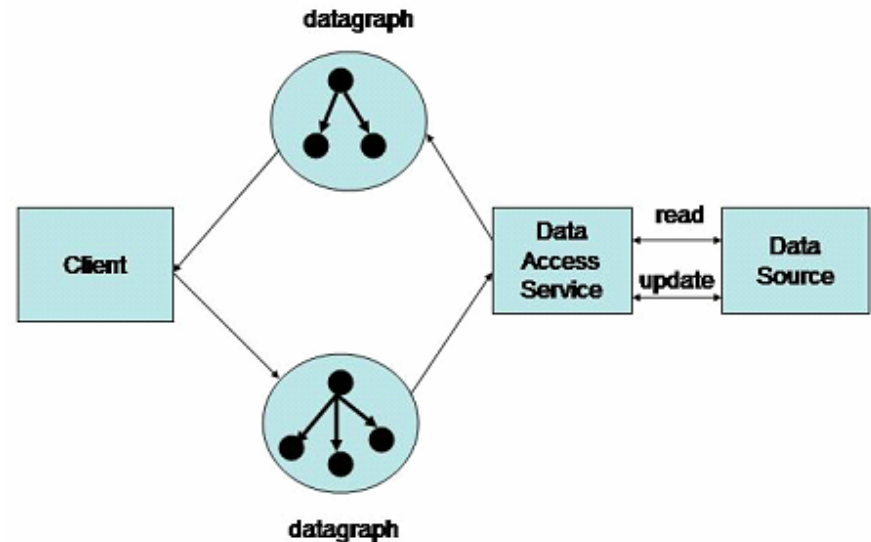
SDO and SOA (cont.)

- **DataObject** – entity representing fragment of data
 - Property (single-valued, many-valued, “simple”, “complex”)
 - Type (String, Integer, Date, Boolean, DataObject)
 - z/TPFDF
 - DataObject – subfile, LREC
 - Property – field in LREC, reference (link) from LREC to another subfile
- **DataGraph** – graph representing data (non-persistent)
 - Tree of DataObjects connected by reference
 - Nodes are accessed via root & references
- **ChangeSummary**
 - change history for DataGraph and DataObjects

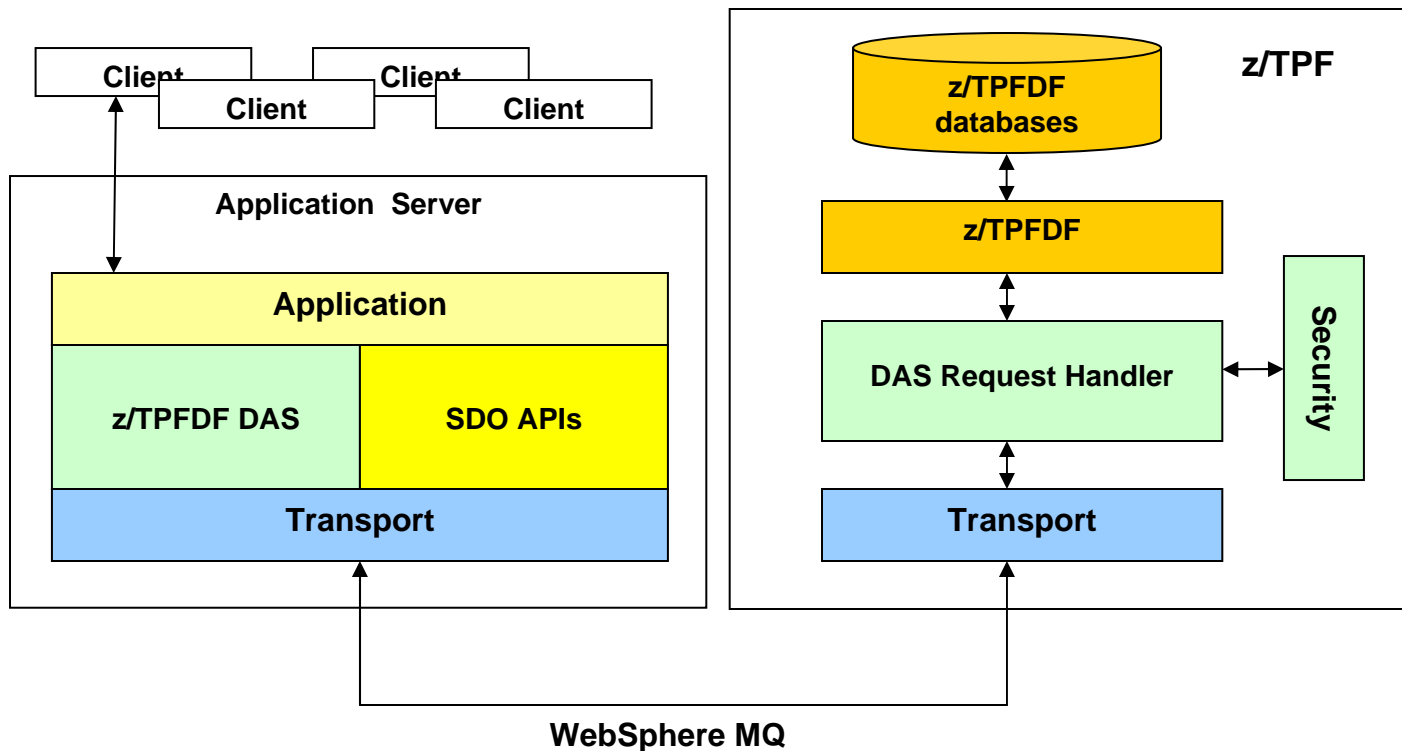


SDO and SOA (cont.)

- **Data Access Service (DAS)**
 - Specific form of SCA (Service Component Architecture) service
 - Load DataGraph from a data source or service, for example
 - XML data sources – XML file DAS
 - Relational databases – JDBC DAS
 - Propagate changes back into the data source
 - Disconnected model

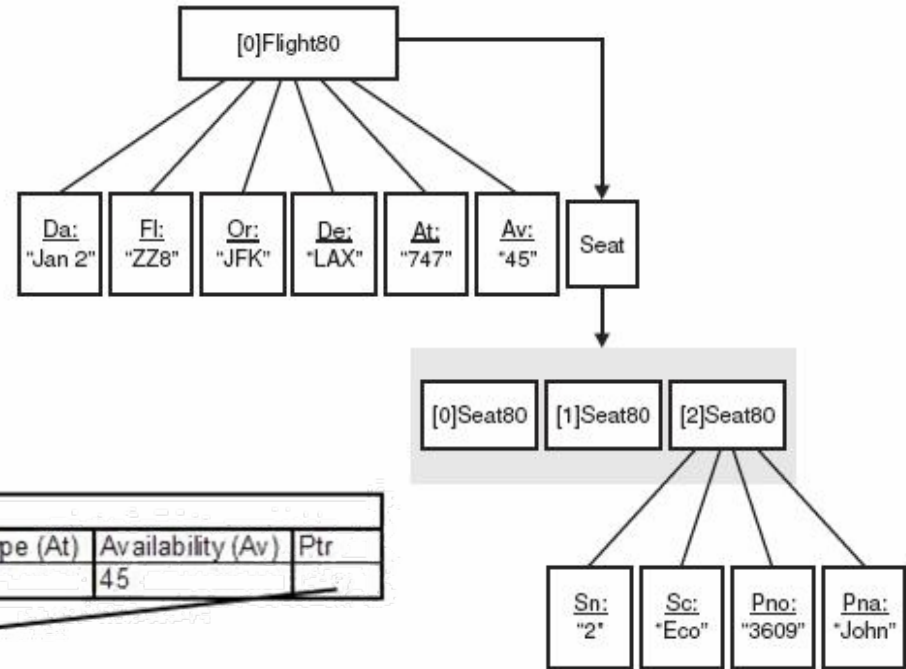


SDO and z/TPFDF



Note: Items shipped are in blue and green

SDO and z/TPFDF (cont.)



IR22DF							
Flight File							
Flight80	Date (Da)	Flight Num (Fl)	Origin (Or)	Destination (De)	Aircraft Type (At)	Availability (Av)	Ptr
80	Jan2	ZZ8	JFK	LAX	747	45	

IR23DF					
Seat File					
Seat80	Seat Number (Sn)	Seat Class (Sc)	Passenger Number (Pn)	Passenger Name (Pna)	Ptr
80	0	Eco	1414	Jerry	
80	1	Bus	7112	Jennifer	
80	2	Eco	3609	John	

Metadata

- **Data about data**
- **SDO data representation is built upon the metadata**
- **Database**
 - Collection of related files (i.e., Flight, Seat and Passenger)
 - Metadata describes structure of the data on database level
- **Describes z/TPFDF databases by mapping DBDEFs and DSECTs using XML**
- **ZUDFM METADATA CREATE**
 - Creates an XML metadata
 - Some updates might be needed (i.e., aliases for fields)
- **ZUDFM METADATA VALIDATE**
 - Validates updated metadata

Metadata (cont.)

- **Versioning**
 - Metadata and DBDEF version mismatches are detected
- **Advantages**
 - All information in one place
 - XML representation – easy to read, update and validate
 - Additional controls
 - Hide fields
 - Make fields read-only
 - Control data types in Java representation

Security

- **Security**
 - Access controlled by user id
 - Different levels of access to different databases – read/write/none
 - Also on subsystem and subsystem user level
 - User exit can control authentication

Configuration

- **Configuration on z/TPF**
 - Restrict amount of I/Os and CPU time per ECB
 - Restrict number of SDO-based ECBs
- **Configuration on the client**
 - Limit maximum amount of data to read

z/TPFDF SDO Data Collection

- **ZUDFC SDO**
 - Collects data pertinent to SDO-based applications
 - Similar format to standard ZUDFC
 - Data can be collected based on users and databases
- **Output results to XML**
 - XSL stylesheet provided
 - Convert output XML into HTML
 - Output can be opened in popular spreadsheet or HTML browser

Application Programming Interface

- **Retrieve data from z/TPFDF database into the Data Graph**
 - readData(file, path, searchkeys, properties)
- **Access and modify data in the Data Graph**
 - Data manipulation is done using standard SDO API
 - The same API as for any other data source
- **Save changes in the database**
 - applyChanges(Data Graph)

Application Development

- **Platform**
 - Any platform that supports Java and WebSphere MQ
- **Standard Java applications**
 - SDO and WebSphere MQ libraries in project's classpath
 - Java tooling can be used (debuggers, IDEs, etc)
 - For example, Eclipse, TPF Toolkit or IBM WebSphere Studio
- **TPF Toolkit**
 - Java IDE
 - Java Debugger
 - Java Profiler
 - Native support for Eclipse plug-ins

APAR Information

- **PK60030 - z/TPFDF APAR**
- **PJ32720 – co-requisite z/TPF APAR**
- **Java DAS code will be available for download on the TPF web site as JAR-file**
- **SDO libraries are freely available**
 - <http://www.eclipse.org/modeling/emf/> - reference implementation
 - <http://incubator.apache.org/tuscany/> - open-source runtime implementation with tools

Additional Information

- **Tonight's demo session**
- **Database Subcommittee**
 - “SDO Data Access Service: Setup and Administration” by Sasha Krymer
- **Application Development Subcommittee**
 - “Application Development using SDODF” by Glenn Katzen