Agenda

- IMS Transaction and Database Manager
- IMS and DB2
- IMS in a Parallel Sysplex Environment
- IMS integration in SOA
- IMS Tools
- IMS, The World Depends on it!
IMS - What is it?

- Information Management System
- 3 components
  - The Transaction Manager (TM)
  - The Database Manager (DB)
  - Set of system services, providing common services to the other 2
- Tied-up with z/OS operating system
  - Running in multiple address spaces (regions)
  - Runs multiple tasks in each address space
  - Using cross memory services to communicate between the various address spaces
  - Allows Parallel Sysplex support
- Lot of web-enablement capabilities
  - Access to the transactional and data assets

IMS - What is it? …

- A transaction manager
  - Provides high-volume, rapid response transaction management processing for application programs accessing IMS and/or DB2 database, MQ queues
    Managing the application programs — dispatching work, loading application programs, providing locking services
  - Manages input and output messages from network (3270s, APPC, TCP/IP, WebSphere MQ, etc.)
  - Manages “batch oriented” application program – no network access
- A database management system
  - IMS Databases are hierarchical
    Faster than relational databases (DB2)
    Used by companies needing high transaction rates
  - Multiple database types
    Full Function (full IMS feature, not as high transaction rate as Fast Path)
    Fast Path (used by companies needing very high transaction rates)
    Partitioned IMS DB, called High Availability Large Database (HALDB)
    XML Database
IMS Transaction Manager

- Communication Manager
  - Device Dependent Modules for SNA LU0, LU1, LU2, LU6.1
  - APPC/IMS to support SNA LU6.2
  - OTMA to provide direct access to any MVS client
  - WMQ, IMS Connect
- Transaction Manager
  - With IMS Message Queue - IMS Full Function TM
  - Without IMS Q - IMS Fast Path TM
- Application Program Manager
  - Message Scheduling
  - IMS Message processing region (MPP)
  - IMS Batch Message Processing Region (BMP)
  - BMP Message Driven
  - BMP Non-Message Driven
  - IMS Java Dependent Region
  - JBP and JBP
  - Interactive Fast Path (IFP)
    - Using Expedited Message Handler
    - For IMS Fast Path TM only
- Lock Manager
  - Program isolation for internal locking
  - ILRM for global locking
  - Mandatory in a parallel sysplex environment
- Syncpoint Manager
  - Coordination of the Two Phase Commit for MPP, BMP, IFP
  - Resource managers involved: IMS TM, IMS DB, DB2, MQI
- Resource Manager
  - Participant in the Two Phase Commit when z/OS Resource Recovery Services (RRS) is Syncpoint manager
  - APPC distributed syncpoint for transaction access
  - ODBA for database access
  - Two Phase Commit for JBP when DB2 resources are accessed
- Security
  - Using RACF or any equivalent product
  - Using IMS exits
  - Signon exit
  - Transaction Authorization exit and Command Authorization exit
  - Security Reverification exit

IMS Database Manager

- Same architecture than IMS TM/DB
  - DBCTL to access IMS DB (FF and FP)
    - IMS Batch still exist outside of IMS DBCTL scope.
    - No communication manager
    - No transaction manager
    - No message queue
    - No message driven region (MPP, BMP MD, IFP, JBP)
- Lock Manager
- Syncpoint Manager
  - DBCTL as coordinator of the Two Phase Commit for BMP
  - DBCTL as participant of the Two Phase Commit for CICS trans...
  - Access to IMS DB, DB2, MQI in the same unit of work
- Resource Manager
  - With ODBA
  - Syncpoint manager is RRS
- Security
IMS Subsystem Structure

IMS Control Region

IMS Message Queues
IMS Libraries
DLI Separate Address Space
DB Recovery Control Region
MPP Appl. Program
IFP Appl. Program
BMP Appl. Program
JMP / JBP Appl. Program
Fast Path DBs
Logs
Control Region Address Space
Dependent Region Address Space

Network - SNA or TCP/IP

IMS Libraries
IMS Message Queues
IMS Processing Region
MESSAGE QUEUE DATA SET(S)
PROGRAM LIBRARIES

IMS TM - Message Queue Concept

IMS Control Region

Logical TERMINAL ABC

Program Libraries

IMS Processing Region

TRAN1
GET FROM QUEUE
.......
INSERT TO QUEUE

IMS Processing Region

TRAN2
GET FROM QUEUE
.......
INSERT TO QUEUE

For all INPUT and some OUTPUT messages depending on the protocol.
IMS TM - Logical Terminal (LTERM) Concept

- PHYSICAL TERMINAL 1
  - LOGICAL TERMINAL 1
    - MESSAGES
  - Application 2

- PHYSICAL TERMINAL 3
  - LOGICAL TERMINAL 2
    - MESSAGES
  - LOGICAL TERMINAL 3
    - MESSAGES

- IMSCTL (Control Region)
- MPR (Msg. Processing Region)

APPLICATION PROGRAM 'ANYPGM'
  Read Message from IMS Transaction Q
  Access Resource Managers
  Insert messages on IMS LTERM Q

IMS TM – Big Picture

- IMS Transaction
  - No presentation layer
  - Access to Resource Managers (RM)
  - Very simple design
    - Get Input Message
    - RM calls
    - ISRT Output Message

- IMS Database
  - Hierarchical design
  - JDBC access
  - XML datastore

- IMS MFS
  - Description of input and output messages and device map
    - Not used in client/server implementations

- IMSCTL (Control Region)
- MPR (Msg. Processing Region)

APPLICATION PROGRAM 'ANYPGM'
  Read Message from IMS Transaction Q
  Access Resource Managers
  Insert messages on IMS LTERM Q

- VTAM (LU0, LU1, LU2, LU6.1, LU6.2)
- TCP/IP for zOS
- APPC Comm.
- Open Transaction Manager Access
- IMS Message Queues
- IMS DC
- IMS Dependent Regions
  - IMS Connect
  - DB2
  - MFS
  - DB2
  - IMS Database Manager
  - MQSeries
  - DB2
Access to IMS Transaction – Open Transaction Manager Access (OTMA)

- The “Modern” access model
  - Available for more than 8 years!
- High performance interface between z/OS Client Applications and IMS transaction processing
  - Uses z/OS XCF facilities
- Enables exploitation of existing IMS applications without changing them
  - z/OS clients specify the nature of the flow and synchronisation protocols, on a per-transaction basis
  - Includes user and transaction security

Access to IMS Data - Open Database Access (ODBA)

- A new access model
- Provides a callable interface to IMS databases from any z/OS programs that are not managed by IMS
  - DB2 Stored Procedures
  - or Any applications that use z/OS Resource Recovery Services (RRS) to manage their syncpoint processing
- Connection to IMS TM or DBCTL
  - Uses the Database Resource Adapter (DRA)
  - DL/I calls are issued using the Application Interface Block (AIB) interface
DBRC for IMS Data Integrity

- Control of IMS CTL Logs (mandatory) and IMS Batch logs (optional)
- Management of IMS databases (optional)
  - Control of the data sharing
    - Database Level Sharing
    - Block Level Data Sharing
  - DBRC registration of none, some or all the IMS databases
  - DBRC Database authorization process
- JCL generation for some utilities

RECON data sets
= Central Catalog with Information for IMS Data Integrity

IMS Databases

- Database Definition: a collection of interrelated data items organized in a form for easy retrieval
  - The collection of data is stored in a computer system
  - The retrieval is done by application programs
  - Each item of data only needs to be stored once
    - Shared among the programs and users
- An IMS database is organized as a hierarchy
  - Levels of data
  - Data at lower levels depends on data at higher levels
  - A database is a group of related database records (DBRs)
  - A database record is a single hierarchy of related segments
  - A segment is a group of related fields
  - A field is a single piece of data
    - It can be used as a key for ordering the segments
    - It can be used as a qualifier for searching
    - It may only have meaning to the applications
IMS Database Overview

- **Root**
  - One and only one root for each database record
  - No higher level segments
    - Everything depends on the information in the root
- **Other Segment Types**
  - Up to 254 different segment types
    - 255 including the root
  - Any number of occurrences of each segment type
  - Each segment, except the root, is related to one and only one segment at the next higher level
- **Segments stored with a prefix and a data portion**
  - Prefix Portion
    - *Used only by IMS*
    - *SC = segment code, 1 byte*  
    - *DB = delete byte, 1 byte*  
    - *0 to n pointers, 4 bytes each*
  - Data Portion
    - *What the application program sees*
Segment Relationships

- **Parent**
  - All segments which have dependent segments at the next lower level are parents of those segments
  - A parent may have any number of dependent segments

- **Child**
  - A segment which depends on a segment at a higher level is a child of that segment
  - Every child segment has one and only one present

- **Twins**
  - All occurrences of a segment type under the same parent are twins
  - There may be any number of twins and they are still called twins

- **Siblings**
  - Segments of different types with the same parent are siblings

Logical Relationships Types

- **Unidirectional**
  - A one-way relationship from one database record to another
  - Applications always start from one place

- **Bidirectional**
  - A two-way relationship between database records
  - Applications may need to start on either side
  - IMS maintains both sides of bidirectional relationships
Secondary Indices

- Processing sequence other than root key
  - Avoid scan for non-key field
- Direct access to lower level segments
  - Faster processing

![Diagram]

IMS Full Function DB - Direct Organization

- Physical storage is independent of hierarchic sequence
  - Pointers are used to maintain segment relationships
    - Pointers are in the segment prefix
    - Segments can be stored 'anywhere'
    - Segments are not physically moved
    - Space from deleted segments can be reused
- Direct Database Types
  - Hierarchic Direct Access Method (HDAM)
    - Randomizing module for direct access to root
    - No sequential access in the root order
  - Hierarchic Indexed Direct Access Method (HIDAM)
    - Access to the root using an index
    - Sequential access in the root order possible
  - And the 2 HALDB Types
    - Partitioned HDAM
    - Partitioned HIDAM
IMS Full Function DB - Sequential Organization

• The data is physically stored in hierarchical sequence
  - Database records are stored in a root key sequence
    If no root key, they are stored as presented
  - Segments in a record are stored in hierarchical sequence

• Sequential Database Types
  - Hierarchical Sequential Access Method (HSAM)
  - Simple Hierarchical Sequential Access Method (SHSAM)
    Root-only HSAM
  - Hierarchical Indexed Sequential Access Method (HISAM)
  - Simple Hierarchical Indexed Sequential Access Method (SHISAM)
    Root-only HISAM using VSAM
  - Generalized Sequential Access Method (GSAM)
    No hierarchy, no database records, no segments
  - And the HALDB PSINDEX
    Partitioned Secondary Index

IMS High Availability Large Database (HALDB)

• IMS High Availability Large Database (HALDB) with IMS V7
  - Extends IMS Full Function database size
  - Up to 1001 Partitions x 10 data set groups x 4G = 40 Terabytes
  - Provides data availability through partition independence
  - Provides easier manageability with smaller partitions of the database

• IMS HALDB Integrated Online Reorganization (OLR) with IMS V9
  - Provides reorganization by partition of HALDBs with concurrent online update and availability
  - Provides recovery from system, IMS, media failure
  - Provides no outage - HALDB partition remains online and available during Reorganization
  - Users can adjust pace of OLR
IMS Fast Path Databases

- **Goal**
  - High Performance
  - High Volume
  - High Availability

- **DEDBs**
  - Similar to HDAM organization
  - Areas and partitioning
  - Multiple Area Data Sets (MADS)
  - Virtual Storage Option to replace MSDB
  - Limitations
    - No access with IMS Batch
    - No logical relationship and secondary indices

- **Main Storage Database (MSDB)**
  - Table-like
  - Resident in main storage
  - To be replaced by DEDB VSO

---

Access to IMS Databases

- **With IMS Transaction Manager**
  - MPP, IFP
  - BMP Message Driven or Non Message Driven
  - JMP, JBP since IMS Version 7

- **With IMS Database Manager (DBCTL)**
  - BMP Non Message Driven
  - JBP since IMS Version 7

- **With CICS Transaction Server**
  - CICS Transactions
  - CICS Java

- **With IMS Batch**
  - z/OS Batch
  - Only for Full Function IMS databases
  - Can be converted to BMP in an IMS TM or DBCTL environment

- **With any z/OS address space under RRS control**
  - Thanks to Open Database Access (ODBA)
  - On the same z/OS image
  - Example: DB2 Stored Procedure
  - Websphere Application Server component
Agenda

- IMS Transaction and Database Manager
- IMS and DB2
- IMS in a Parallel Sysplex Environment
- IMS integration in SOA
- IMS Tools
- IMS, The World Depends on it!

Access to DB2 from an IMS Application

- The need
  - Take benefit of IMS TM strengths to access DB2 data
  - High system availability of IMS architecture
  - Use all the network connection capabilities of IMS to access DB2 data
  - No impact on IMS users
  - Allow access to both IMS DB and DB2
- IMS External Subsystem Attachment Facility
  - Allows access to DB2 from IMS MPP, BMP, IFP and IMS batch.
  - Receives and interprets requests for access to DB2 databases
  - Uses exits provided by IMS subsystems
  - Allows coordinated recovery of both DB2 and IMS data
- DB2 Recoverable Resource Manager Services attachment facility (RRSAF)
  - Allows access to DB2 from IMS JMP and JBP (new with IMS Java support)
  - JDBC access to DB2
  - Coordinated recovery managed by zOS RRS
Access to IMS Databases using call DLI (ODBA)

- DB2 Stored Procedures Address Space access to an IMS DB Subsystem
- DL/I data can be presented through an SQL interface either
  - locally to this DB2 or
  - to DRDA connected DB2s
- RRS coordinates the commit between DB2 and IMS

DB2 establishes the ODBA environment by issuing the INIT call for the Stored Procedure Address Space. Connection to a specific IMS occurs when the APSB is issued. DB2 issues the commit call on behalf of the Stored Procedure when control is returned to DB2.

Access to IMS Databases using call DLI (ODBA) ...

- JDBC to DB2, DB2 to IMS
Access to IMS Transactions using APPC

- DB2 Stored Procedures Address Space uses MVS/APPC calls to access IMS TM Subsystem
  - On same OS/390 or on different OS/390
- DB2 notifies RRS about Commit/abort decisions
- RRS passes decision to other resources manager
  - Allows 2-phase commit with MVS/APPC applications (e.g. an IMS Transaction)

Access to IMS Transactions using OTMA C/I

- DSNAIMS is a stored procedure that allows DB2 applications to invoke IMS transactions and commands easily, without having to maintain their own connections to IMS.
- This stored procedure uses the IMS Open Transaction Manager Access Callable Interface (OTMA CI) API to connect with IMS and execute the transactions.
- The following is required before installing and executing the DSNAIMS stored procedure:
  - DB2 V7 or later with APARs PQ44819 and PQ89544, and RRSAF enabled.
  - A WLM-managed stored procedure address space in which to run DSNAIMS
  - IMS V7 or later with OTMA CI enabled.
Agenda

- IMS Transaction and Database Manager
- IMS and DB2
- IMS in a Parallel Sysplex Environment
- IMS in a On Demand Environment
  - Link with WebSphere brand
- IMS Tools
- IMS, The World Depends on it!

Parallel Sysplex - Goals and Solutions

Single Image System
(VTAM Generic Resources, Operations)

Workload Balancing
(IMS Shared Queues)

IMS Data Sharing
(VSAM, OSAM, Fastpath)
Parallel Sysplex - Functions

MultiSystem Management and Data Sharing

- Single Image
- Dynamic Workload Balancing
- Unchanged Applications
- Data Sharing
- Base Services

IMS Parallel Sysplex support
- IMS Data sharing
- Automatic Workload balancing with Shared Queue
- Single system image for the user
- Single system image for automation
IMS and Parallel Sysplex – Phase A

IMS N-Way Data Sharing

IMS and Parallel Sysplex – Phase B

IMS Shared Message Queues

- Automatic load balancing
- Enhanced queue manager techniques
Agenda

- IMS Transaction and Database Manager
- IMS and DB2
- IMS in a Parallel Sysplex Environment
- IMS integration in SOA
- IMS Tools
- IMS, The World Depends on it!

IMS Transactions - Connectivity Solutions

WebSphere Host Access Transformation Services (HATS)

JCA Connector: IMS Connect / IMS Connector for Java

JMS Connector: MQ to IMS Bridge

SOAP Access: IMS SOAP Gateway

WAS can be on z/OS, on Linux for z or on any distributed platform. Qualities of Services will vary.
IMS Databases – JDBC Connectivity Solutions

From WAS on z/OS using IMS JDBC

From Distributed WAS using IMS Distributed JDBC

Thru DB2 Stored Procedure

Using WebSphere II CF

Agenda

- IMS Transaction and Database Manager
- IMS and DB2
- IMS in a Parallel Sysplex Environment
- IMS integration in SOA
- IMS Tools
- IMS, The World Depends on it!
IMS Tools Product Portfolio

- IMS Database Control Suite
- IMS Command Control Facility
- IMS ETO Support
- IMS HP System Tools
- IMS Queue Control Facility
- IMS Workload Router
- IMS Batch Terminal Simulator
- IMS Batch Backout Manager
- IMS Connect Extensions
- IMS MFS Reversal Utilities
- IMS Program Restart Facility

Fast Path
- IBM HP Fast Path Utilities
- IBM Application Recovery Tool for IMS and DB2 Databases
- IMS Database Recovery Facility
- IMS DataPropagator
- IMS DEDB Fast Recovery
- IMS High Perf Image Copy
- IMS High Perf Change Accumulation

Full Function
- IBM HP Pointer Checker
- IBM HP Prefix Resolution
- IBM HP High Performance Unload
- IBM Index Builder
- IMS Parallel Reorganization
- IMS Online Reorganization Facility
- Administration
- IBM Database Control Suite

Performance Management
- IMS Buffer Pool Analyzer
- IMS Network Compression Facility
- IMS Performance Analyzer
- IMS Problem Investigator
- IMS Systplex Manager
- IBM Tivoli OMEGAMON XE for IMS

Utility Management
- IMS Command Control Facility
- IMS ETO Support
- IMS HP System Tools
- IMS Queue Control Facility
- IMS Workload Router

Recovery Management
- IBM Application Recovery Tool for IMS and DB2 Databases
- IMS Database Recovery Facility
- IMS DataPropagator
- IMS DEDB Fast Recovery
- IMS High Perf Image Copy
- IMS High Perf Change Accumulation

Agenda

- IMS Transaction and Database Manager
- IMS and DB2
- IMS in a Parallel Sysplex Environment
- IMS integration in SOA
- IMS Tools
- IMS, The World Depends on it!
IMS - The World Depends on it

- **IMS Growth**
  - IMS MIPS Growth (> 25%/year)
  - V to V upgrades V7 V8 V9
  - New license growth

- **Over 3 million MIPS running IMS !!!**

38 Years of Proven Quality, Performance and Commitment

IMS is CORE to Customer Business

- **Usage numbers continue to increase**
  - US company has exceeded 100 million trans/day
  - US customer business transaction almost $3 trillion/day
  - 4 customers w/o unplanned outage in 8 years
  - Asian bank has exceeded 200,000,000 savings accounts on-line

- **Customer Investment in IMS**
  - Applications continue to expand
    - *Databases growing in size and number*
    - *Transaction volumes rise*
  - New applications in the form of integration
  - Increase in present volumes

Don’t remove traditional TPMs from the picture: Their quality of services is still unmatched.

Gartner Group, June 2004
IMS Evolution

- **IMS V7** – went out of support in September 2005
  - High Availability Large Database – The IMS partitioning solution
  - Enhancement of the Parallel Sysplex support
  - IMS Java – 1st Step
  - JDBC access to IMS database from a z/OS processing environment (CICS, DB2 SP, WAS)

- **IMS V8** – GA in October 2002
  - IMS Java – 2nd Step
  - Enhanced J2EE connector capabilities in conjunction with IMS Connect Version 2
  - New architecture for better Parallel Sysplex operation management

- **IMS V9** – GA in October 2004
  - Online Reorganisation without restrictions for HALDB
  - Storing XML in IMS Databases
  - IMS Java – 3rd Step
  - Distributed JDBC access to IMS Databases
  - Plus a host of enhancements in the areas of security, performance, installation, usability and operations

- **IMS V10** – Announced in October 2006
  - GA for early customers in 01/2007

The Business of IMS

**IMS** is the fastest, most reliable database computing system in the world, plain and simple. When immediate access to mission-critical information is imperative, 90% of the world’s major corporations rely on IMS to provide a continuous link to data that is accurate, up-to-date, and quickly accessed by many end users.

Customers rely on IMS systems to process billions of vital transactions a day. Any time you make an airline reservation, rent a car, get cash from an ATM, or pick up a prescription from the pharmacy, chances are you’ve used IMS.