Evolution of CICS for Application Programmers

Jani Katta

Westfield Group

August 14th

Share session # 13358

Agenda

- History
- Evolution
- Application developers
    - Languages supported
    - Features
    - Support for new technologies
    - What's new
- Development tools

History

- CICS – Customer Information Control System
- Designed to support BTAM terminals
- First release supported 50 odd terminals
- Online Transaction Processing System (OLTP)
- Released in 1968
- Macro commands

Evolution

- Terminal support
  - Hundreds of thousands of users/terminals
- Language support
  - From limited languages to 7 languages
  - Java, PHP support
- File / Databases – DB2, IMS, VSAM
  - MQ support
- Non-terminal transactions
- Web services support, Web 2.0

Application developers

- Languages supported
  - Assembler
  - COBOL
  - REXX
  - C
  - Java
  - PL/I
  - PHP
- All functions are invoked by "EXEC CICS" commands.

File / Database

- VSAM – read, browse, readnext, readprev, write, rewrite, delete
- DB2 SQL – select, insert, delete
- MQ – open, get, put, put1, close
- IMS DB - CBLTDLI

Ex: 1) EXEC SQL

SELECT EMPLOYEE_NUM, EMPLOYEE_NAME
FROM EMPLOYEE
WHERE DEPT_NUM = '1234'

END-EXEC.
2) EXEC CICS READ
DATASET(datasetname)
INTO(storage-area)
RIDFLD(key-info / rrn)
END-EXEC.

SHARE in Boston

Passing data

- Passing data between programs
  - Temporary Storage queues
  - Transient Data queues
  - Transaction Work Area
  - Commarea
  - Channels/Containers
  - Files/Databases

Temporary Storage queues

- Used to store data temporarily
- Application can build their queues – as they are temporary
- CRUD – Create Read Update Delete
- Queue name – max of 8 bytes
- Application has to delete the queue or deleted during recycling CICS
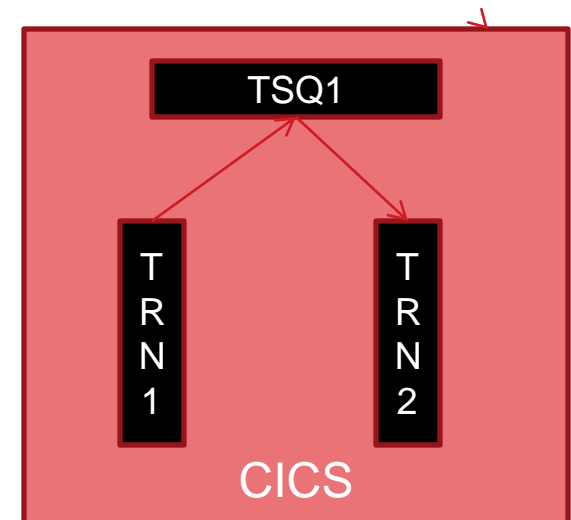
```
EXEC CICS
        WRITEQ TS QUEUE(queuename)
        FROM (working-storage-variable)
        LENGTH (queue-length)
        ITEM(item-no)
        RESP(ws-resp)
END-EXEC.
```

TSQ1

TRN1

TRN2

CICS

Transient Data

- Simple way to pass data sequentially.
- Retrieved in the same sequence it is written. FIFO – first in first out
- Record gets deleted after you read – similar to MQ.
- Can not update the record after you write.
- TD queues need to be defined by System programmers in Destination Control Table (DCT) and have a destination

TDQ1

T
R
N
1

CICS
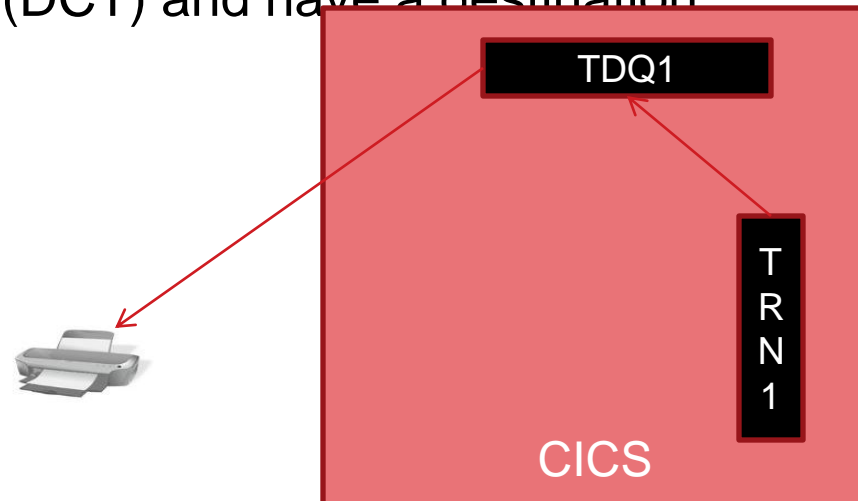
Transaction Work Area

- Defined at the transaction level.
- Each instance of the transaction gets the amount of storage defined in Transaction Control Table
- Application program can address it, modify the area
- The next program gets the updated info
- Should be defined in LINKAGE SECTION.
  LINKAGE SECTION.
  01 LN-TWA.
      05 LN-TWA-EMP-NO   PIC X(10).
      05 LN-TWA-EMP-FIRST-NAME   PIC X(50).
      05 LN-TWA-EMP-LAST-NAME   PIC X(50).

      ..
      PROCEDURE DIVISION.
      …
      EXEC CICS ADDRESS TWA(ADDRESS OF LN-TWA)
      END-EXEC.

      ..
      EXEC CICS READ DATASET('EMPDET') RIDFLD(LN-EMP-NO) INTO(LN-TWA)
      END-EXEC.
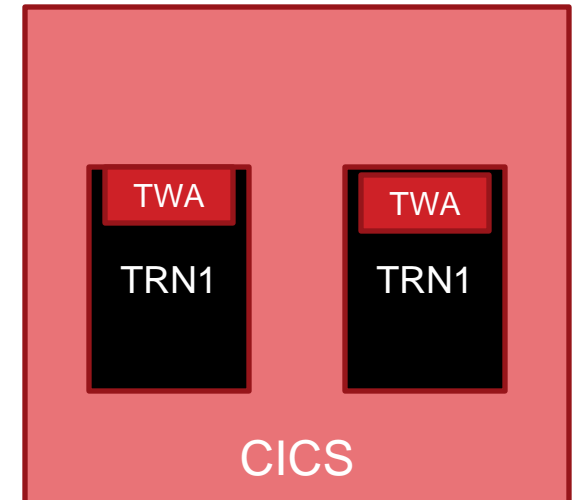
Commarea

- Most commonly used way to pass data in CICS
- Max limit of 32k

        EXEC CICS

                LINK PROGRAM(program name)
                COMMAREA (ls-comm-area)
                LENGTH (ls-comm-area-length)

        END-EXEC.

- Pass data between programs via LINK/XCTL
- Pass data between transaction using START/RETURN TRANID

Channels/Containers

- Channels/Containers
  - Container – blocks of data to pass between programs
  - Channels – grouping of containers
  - Supported for LINK/XCTL/START/RETURN
  - More structured way to pass data
  - No limit on the amount of data ☺

Channels / Containers (contd..)

Channels/containers example

PROG1

PUT CONTAINER('Department') CHANNEL('Employee') FROM(dept-data)
PUT CONTAINER('HR) CHANNEL('Employee') FROM(hr-data)
LINK PROGRAM('PROG2') CHANNEL('Employee)
GET CONTAINER('Pension') CHANNEL('Employee') INTO(pension-data)

Employee

Department

HR

Pension

PROG2

GET CONTAINER('Department') INTO(dept-data)
GET CONTAINER('HR) INTO(hr-data)

...

PUT CONTAINTER('Pension') FROM(pension-data)
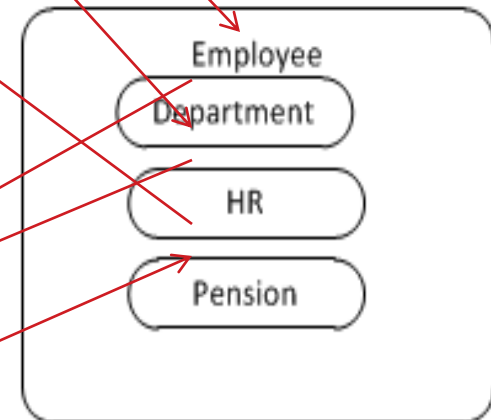
MQ

- Guaranteed delivery
- API's are same across batch and CICS
- Can connect to a Queue Manager
- No need to connect to the queue manager
- Trigger CICS transactions based on queue depth
  - Trigger on every
  - Trigger on first

Other features

- Storage control
  - GETMAIN, FREEMAIN
- Program control
  - LINK, XCTL, CALL
- Interval control
  - WAIT

SHARE
in Boston

Transaction Control blocks

- TCB – transaction control blocks
- QR – quasi-reentrant
- Only one QR TCB per region
- Program need to be defined as quasi-reentrant
- QR makes sure that program is unaffected from previous runs
- Till 1989 - all user applications transactions ran under QR TCB

Transaction Control blocks (contd..)

- Open TCBs
    - Started in 1990's
    - Used to run thread safe applications in its own TCB
    - Application can share QR and open TCB
    - Multiple open TCBs can run at a time
    - J8, J9 for Java programs
    - L8, L9 for application programs (non-Java)
    - There are other TCBs available for different needs
    - Programs and transactions need to be thread-safe
    - MAXTCB parameter – to control the number of open TCBs

Thread safe

- Piece of code is **thread-safe** if it only manipulates shared data in a manner that guarantees safe execution by multiple threads at the same time.
- Few system and program settings
  - STGPROT
  - EXECKEY
  - CONCURRENCY
  - API
- Can achieve CPU savings for some applications
- DB2 applications have greater benefits
- With every release there are more and more application and system commands that are made thread safe.
- Caution: if you have non-thread safe API's you could result in decrease in performance due to TCB switching

Thread safe (contd..)
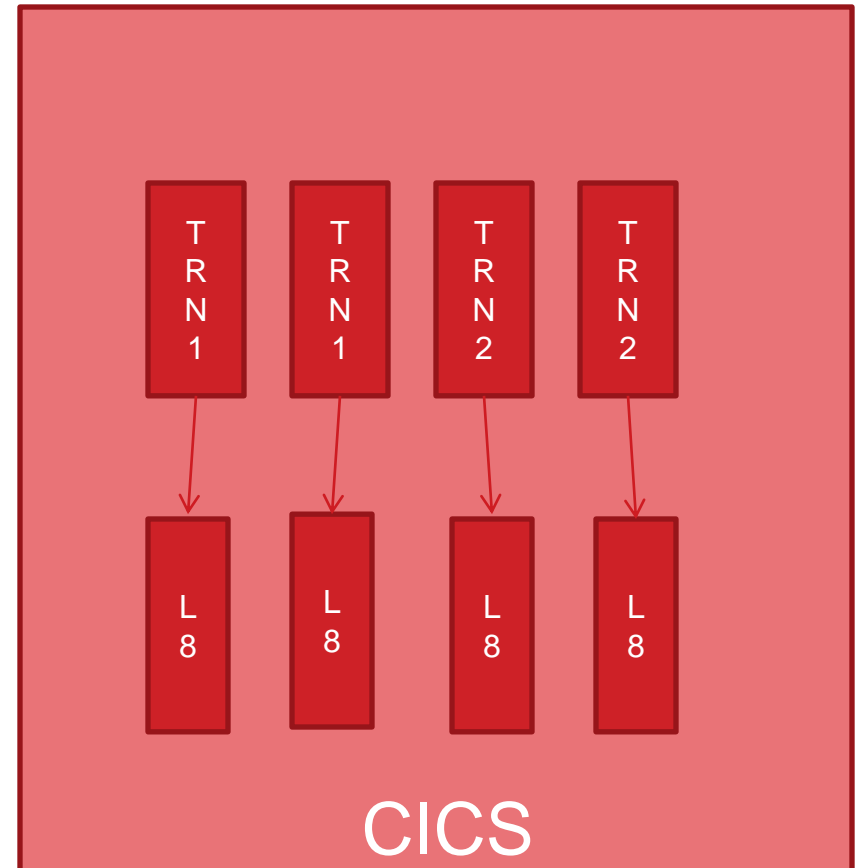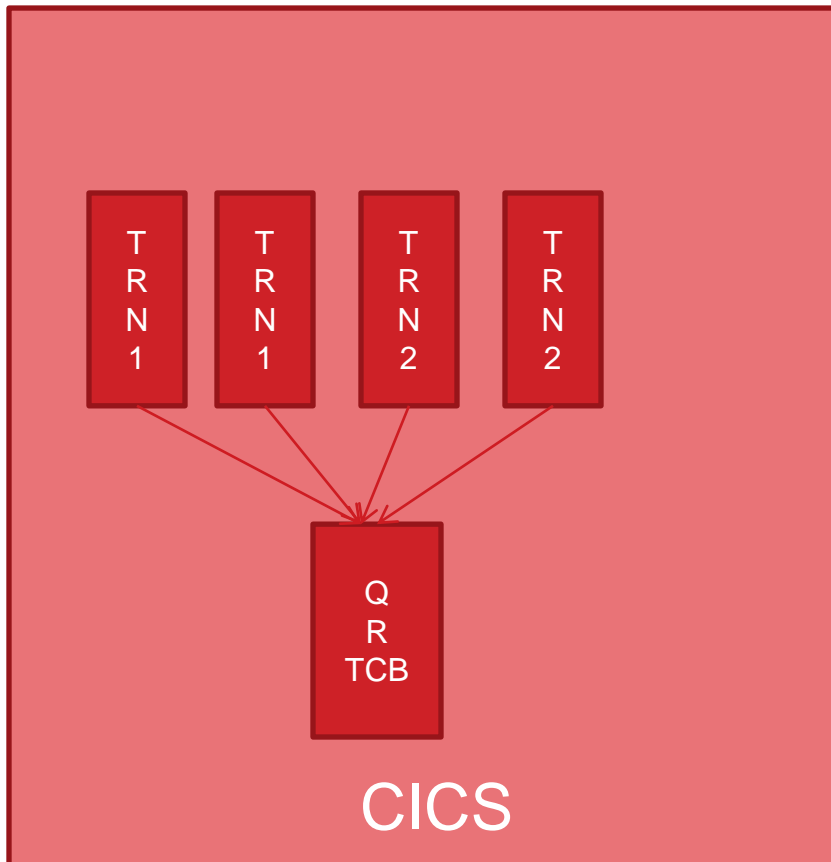
TRN1, TRN2 are thread safe transactions and CICS is configured for it.
A simple example.

Java support

- Support for 64 bit JVMs
- A JVM instance can support multiple transactions
  - Prior versions had a JVM instance for each transaction
- JCICS – Java talking to COBOL programs

- Open Service Gateway initiative (OSGi)
  - OSGi development and packaging now required to deploy CICS applications on a JVM server
  - Existing CICS Java applications using main() method linkage can run unchanged if wrapped in an OSGi bundle
  - All JVM server applications need to be thread-safe

Java support (contd..)

- JCICS example

```
import com.ibm.cics.server.*;
public class Payroll {

...
Task t=Task.getTask();
// create the payroll_2004 channel
Channel payroll_2004 = t.createChannel("payroll-2004");
// create the employee container
Container employee = payroll_2004.createContainer("employee");
// put the employee name into the container
employee.put("John Doe");
// create the wage container
Container wage = payroll_2004.createContainer("wage");
// put the wage into the container
wage.put("2000");
// Link to the PAYROLL program, passing the payroll_2004 channel
Program p = new Program();
p.setName("PAYR");
p.link(payroll_2004);
// Get the status container which has been returned
Container status = payroll_2004.getContainer("status");
// Get the status information
byte[] payrollStatus = status.get();
... }
```

Web Services

- CICS applications can be a service provider and service requester
- Web service assistant
  - Batch utility that helps to transform existing CICS applications into Web Services and also enables to CICS applications to use Web Services provided outside of CICS environment.
  - DFHLS2WS – generates web service binding file from language structure
  - DFHWS2LS - web service binding file from WSDL and also language structure for your application
  - Can do for simpler structures, no nested or max/min boundary conditions

Web Services (contd..)

- Axis 2 support
  - Open source java based engine
  - Can offload some of the processing to specialty engine
  - Needs 4.2 or higher
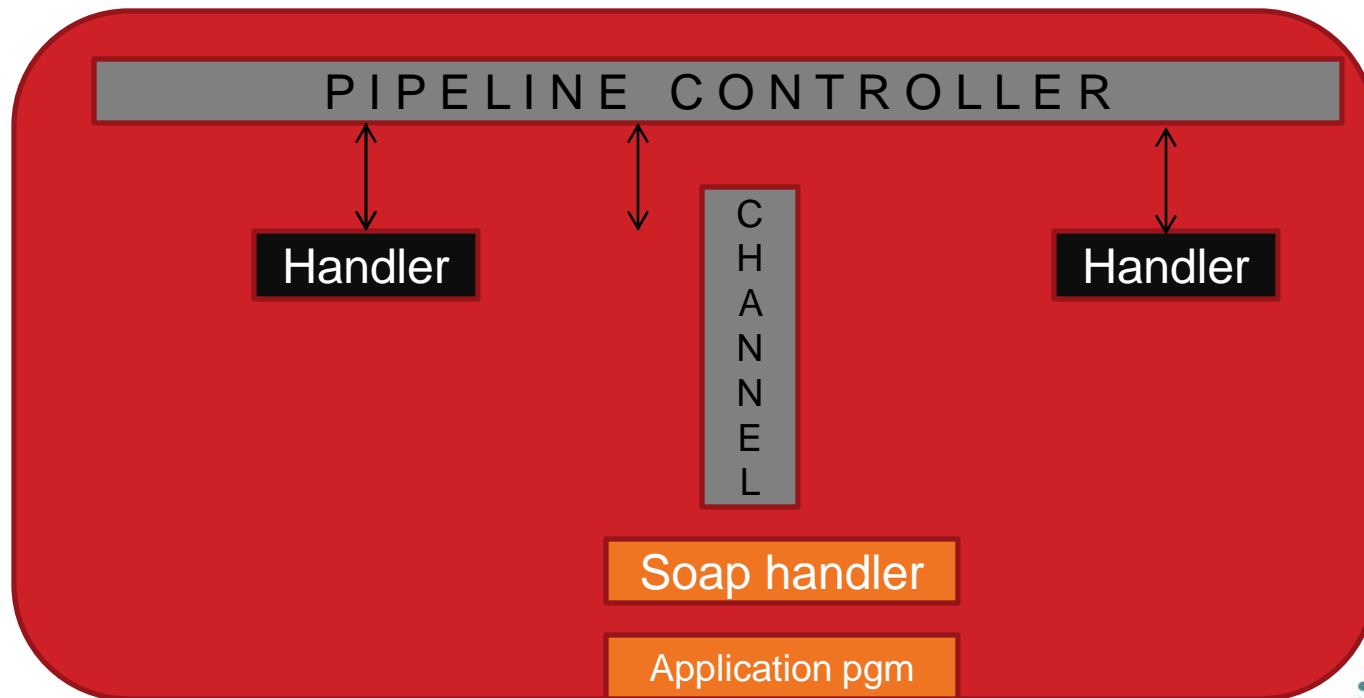
Event Processing

- Can produce events for new and existing applications
- Define and control outside the application code
- Supports events for every user
  - Business user – generate if the order > $20000
  - Application Analyst – Send a message to dashboard when the stock level for an item < 50
  - System programmer – Monitor and optimize performance of system/application

Event Processing (contd..)

- Architecture of EP
- Event sources: CICS applications, RFID sensors
- Event consumers : Dashboards, start new piece of application work

Event Source
- Systems
- Business Processes
- Sensors
- Business
- BAM
- Other?

Events →

Event Processor

Events

Perform operations on events

→ Actions →

Event Consumer

Consume, react to events
- Alerts
- Trigger workflow
- Automated actions
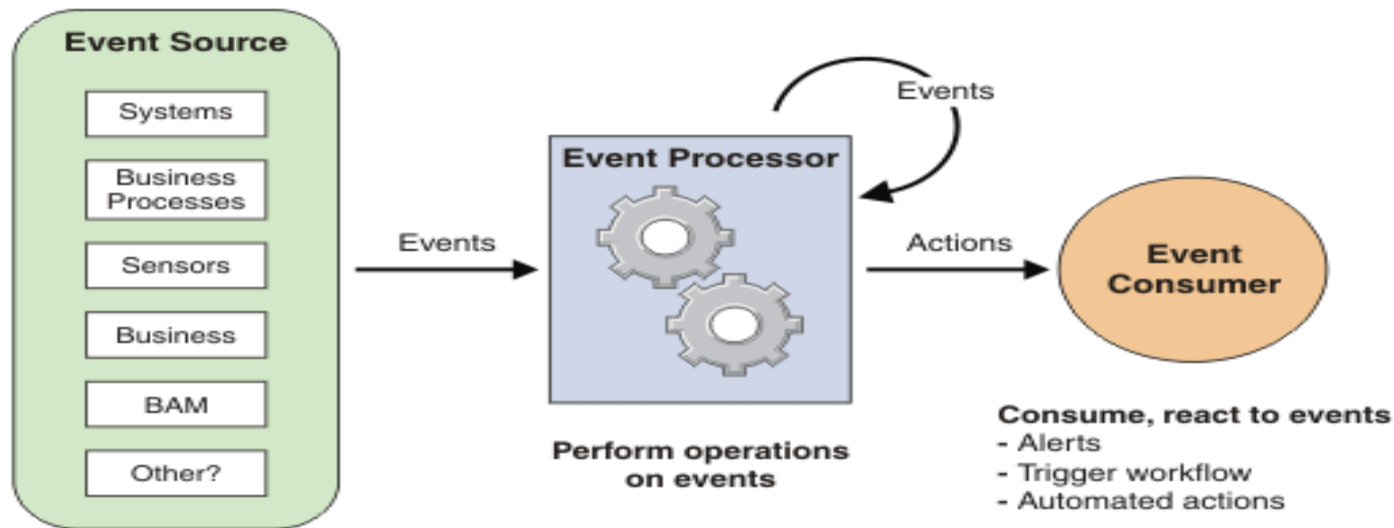
Event Processing (contd..)

- Event adapters support emission of events to the following interfaces:
  - MQ message
  - HTTP message
  - Start another transaction and send information in a channel
  - Start another program
  - Write to a temporary storage
- You need CICS explorer to create EP

Development tools

- CICS Explorer
  - Eclipse plug-in
  - Most of the CEMT, CEDA functions can be accomplished
  - More real-estate
  - Multiple perspectives at a time
  - Few Application development tools that can connect are Debug, File Manager and Fault Analyzer
  - The best thing is FREE!!

Development tools (CICS Explorer)

Development tool(CICS Explorer)

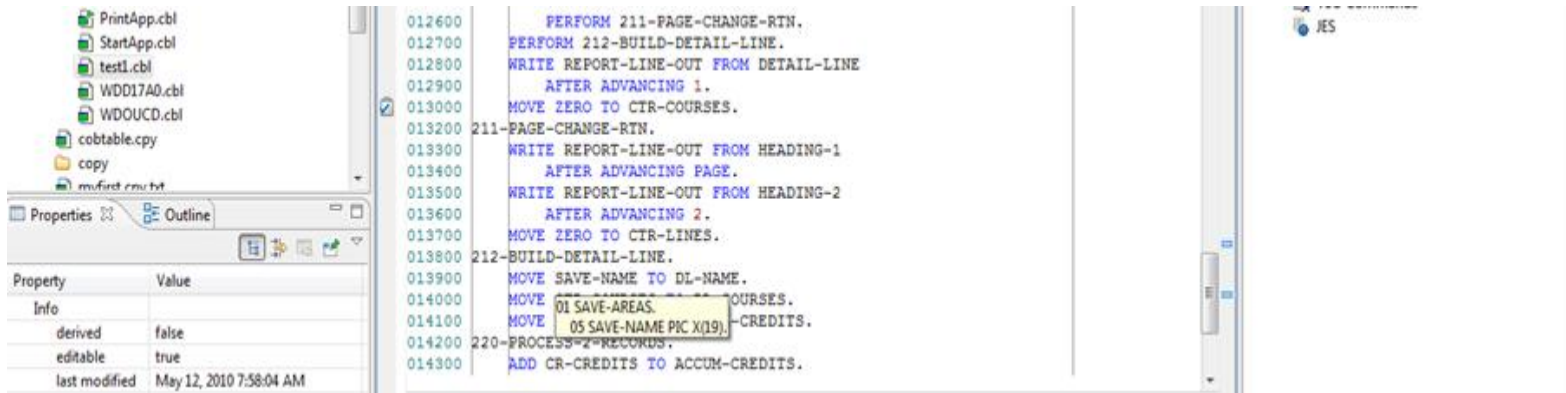- PDS, Unix files, job log at the same time

Development tools

- There are multiple development tools available in the market
- Eclipse plug-in
- Support for most of the languages
- Context menus
- Multiple perspectives at a time
- Integrate with your application development tools

Development tools (contd..)

- Can use same development platform as distributed
- Easy to learn
- Context menus
- Connects to development tools
- Syntax validation
- Remote compilation
- Program flow

Development tools (contd..)
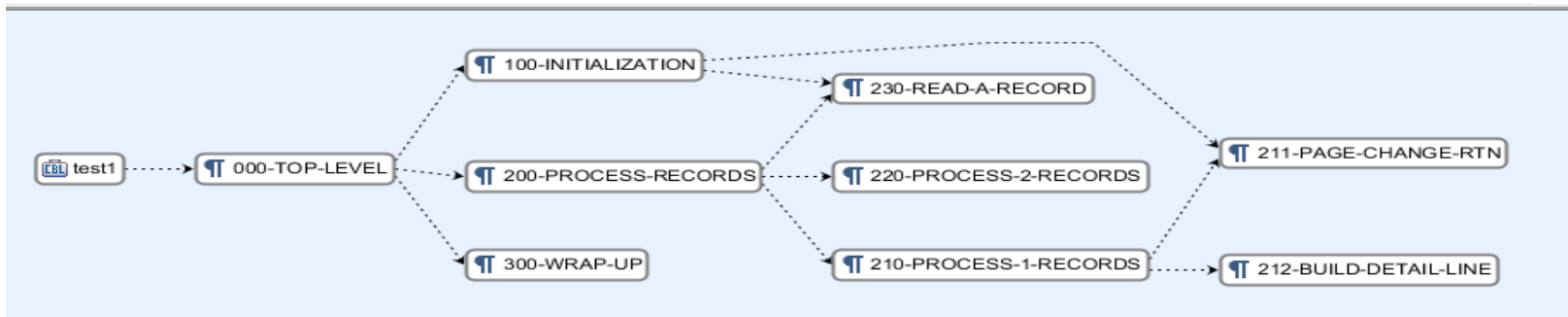
- Mouse over to see definition of field

- Program flow

Complete your sessions evaluation online at SHARE.org/BostonEval

Development tools (contd..)

- Contextual help

```
☑  013000    MOVE ZERO TO CTR-COURSES.
   013200 211-PAGE-CHANGE-RTN.
⚠            WRITE R
   013300    WRITE R    ● REPORT-LINE-OUT              CREDITS-REPORT
   013400        AFT    ● REPORT-LINE-OUT IN CREDITS-REPORT   01 REPORT-LINE-OUT PIC X(60).
   013500    WRITE R
   013600        AFT
```

What's new?

- CICS 5.1
  - Application bundling
    - Bundle all resources of an application
      - *Programs, files, transaction definitions*
    - Can apply policies on the application
    - Helps developers and system admin to check the application status

Additional resources

- Share sessions
  - 13341 – CICS what's in for Application Programmers
  - 13377, 13376 – Modernizing CICS Hands on Lab
  - 13361 – CICS and Java
- CICS TS4.2 Application programming reference
  - http://pic.dhe.ibm.com/infocenter/cicsts/v4r2/topic/com.ibm.cics.ts.applicationprogramming.doc/dfhp4_pdf.pdf
- Event processing
  - http://www.redbooks.ibm.com/redbooks/pdfs/sg247792.pdf
  - http://pic.dhe.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=%2Fcom.ibm.cics.ts.eventprocessing.doc%2Fconcepts%2Fdfhep_support.html
- Webcasts for CICS products
  - http://www.ibm.com/support/docview.wss?uid=swg27007244

Feedback

- Please provide your feedback.

Complete your sessions evaluation online at SHARE.org/BostonEval