

Automating Remote DB2 Binds - Using - CA Endeavor® Change Manager

New England DB2 User Group - March 2008



Rose A. Sakach

Endevor Practice Leader - RSH Consulting, Inc.

R.Sakach@RSHConsulting.com - 617.969.9050 - www.rshconsulting.com

Abstract

In a typical DB2 programming environment, a DB2 bind invocation is required whenever the DB2 source code has been modified. This session will demonstrate the benefits of utilizing Endeavor to automate the DB2 bind function. In addition, we will examine how to achieve automation when the DB2 subsystems associated with specific life cycle stages reside on multiple mainframe system images.

Topics include determining your DB2 environment, creating and utilizing Endeavor symbolic parameters, and options for coding DB2 bind statements. Sample Endeavor processors and symbolics will be provided.

Agenda

A Brief Endeavor® Overview

Why automate

Terms and process review

Information gathering

Mapping life cycles

Connecting DB2 subsystems

Processor considerations

Communication

A Brief Endeavor Overview

- **Change Control** - The process of controlling software changes throughout an application life cycle to ensure only approved, tested software is implemented into production
- **Life Cycle** - The phases through which an application passes from conception to the termination of its use, including design, development, test, implementation, operation, maintenance, and modification
- **Change Management Tool** - allows you to automate and control the movement of application software through your applications Life Cycle
- **Sample Application Life Cycle:**
 - Test – Individual programs are developed and unit tested
 - QA - Applications are system tested and approved for production
 - Production - Production applications are stored and executed

A Brief Endeavor Overview

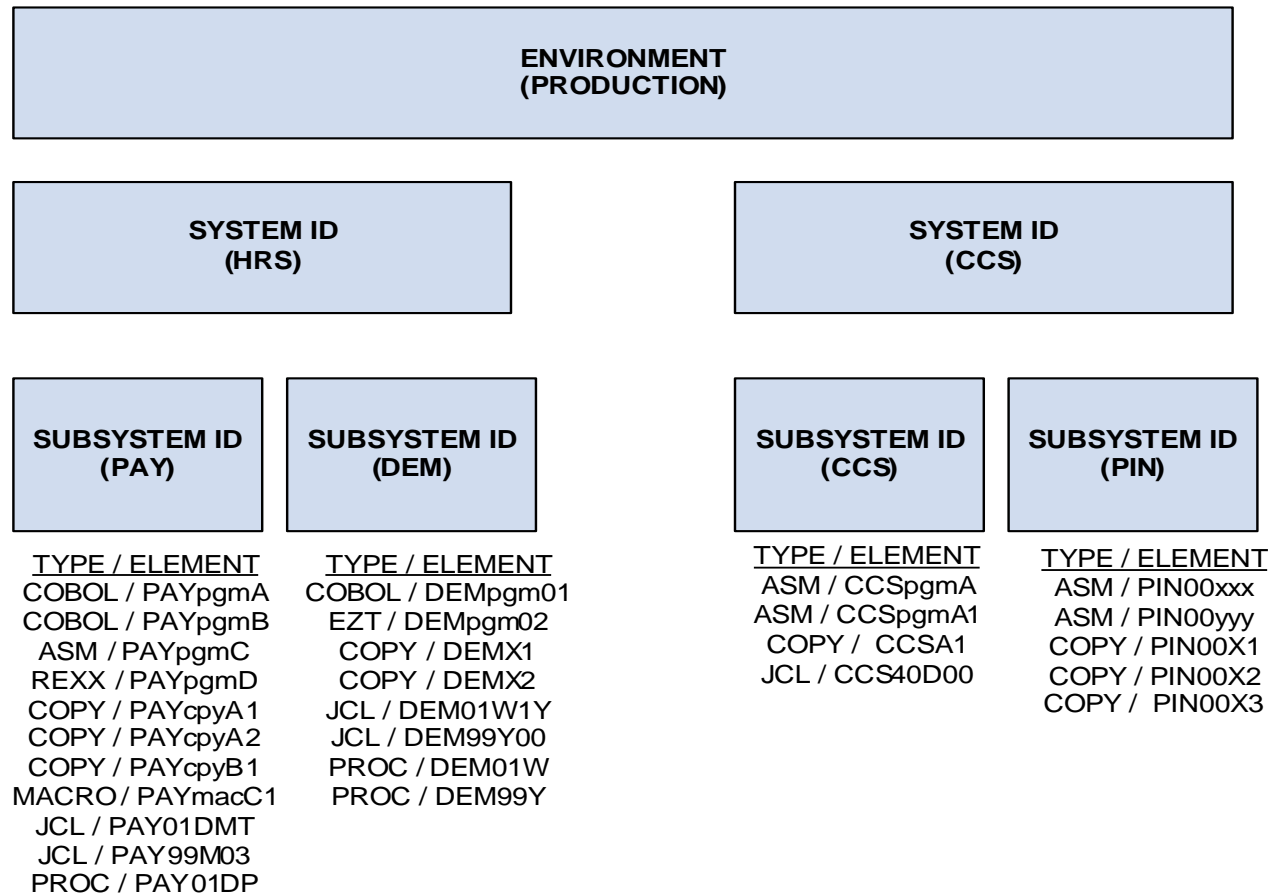
EN-DEV-OR is an acronym!

- **Environment for Development and Operations**
- **Developed in 1980's**
- **Introduced in 1986 by Condor Technology (NYC)**
- **Subsequent owners BST, Legent, CA**
- **Renamed to CA - Software Change Manager for Mainframe (SCM for Mainframe) in 2007**

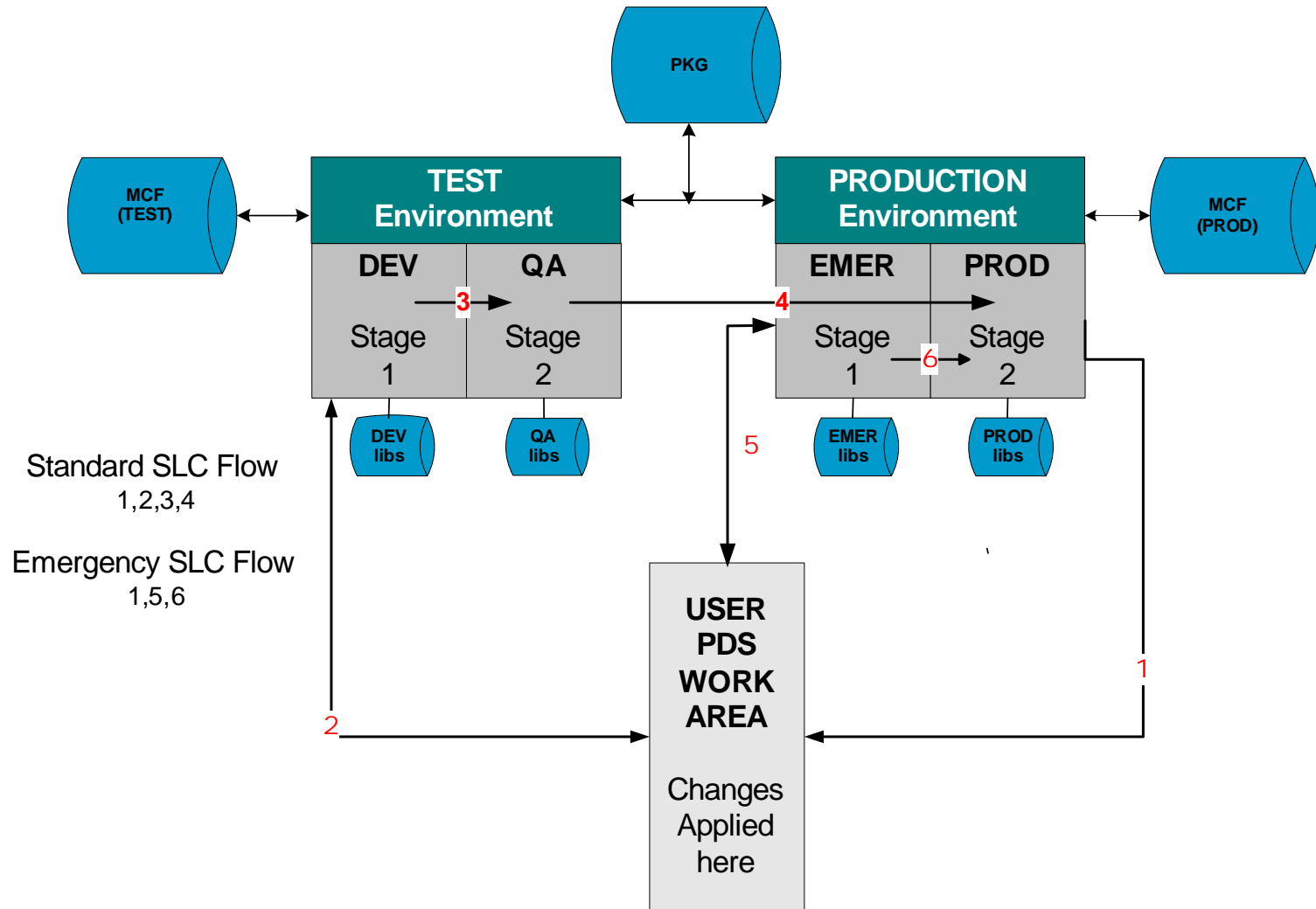
Automated Mainframe Software Configuration Management

- **Inventory Management**
- **Source Management**
- **Output Management**
- **Configuration Management**

A Brief Endeavor Overview



A Brief Endeavor Overview



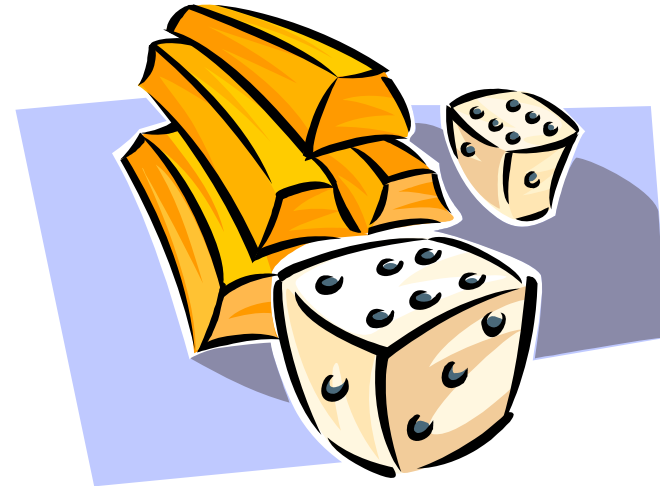
Why Automate

Streamline the process

Reduce risk

Improve quality

Significant cost savings

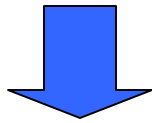


Terms and Process Review

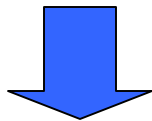
- **Subsystem id (SSID)**
- **Remote location id**
- **DB2 pre-compile or DB2 coprocessor**
 - **SQL syntax validation**
 - **Host language replacement**
 - **Consistency token**
 - **Version parameter**
 - **Data base request module**
- **DB2 bind**
 - **Package**
 - **Collection**
 - **Plan**

Terms and Process Review

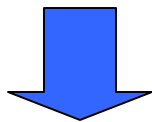
DB2 pre-compile



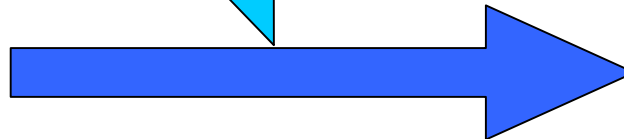
Source Compile



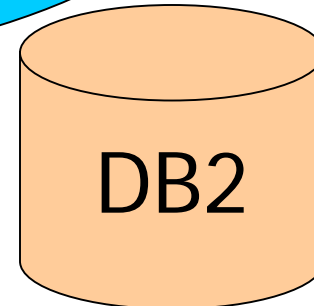
Object Link Edit



DB2 Bind



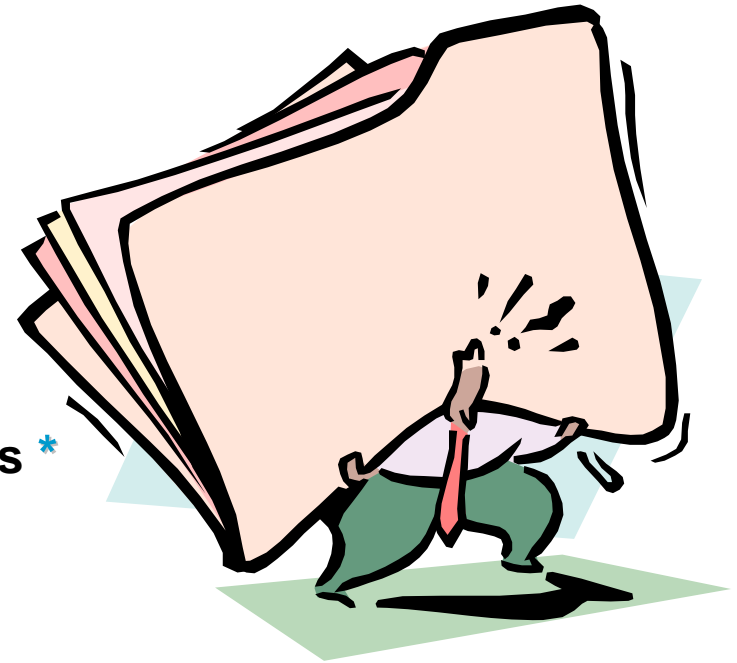
Note:
The DB2
coprocessor
COBOL option
eliminates the
need for the
pre-compile step



Information Gathering

Current DB2 environment

- Total number of DB2 subsystems
- Where they reside (LPAR)
- How/when binds are performed *
- Required authority *
- Use of versioning *
- Bind parameter values *
- DB2 component naming standards *



* **Caution: Be prepared to repeat your data collection process as information may vary from application to application**

Information Gathering

Endevor Stage	DEV	TEST	QA	PROD
LPAR				
SSID / Remote Location Id				
Secondary AUTHID / RACF Group				
Plan				
Owner				
Qualifier				
Package / Collection id				

Information Gathering

Additional DB2 considerations

Default bind card values

- **Dynamic (within application)**
- **Static**

Bind types

- **Packages (collections?)**
- **Plan**
- **Bindonly**

Bind card overrides

Bind Authority

Load module back-out process (versioning)

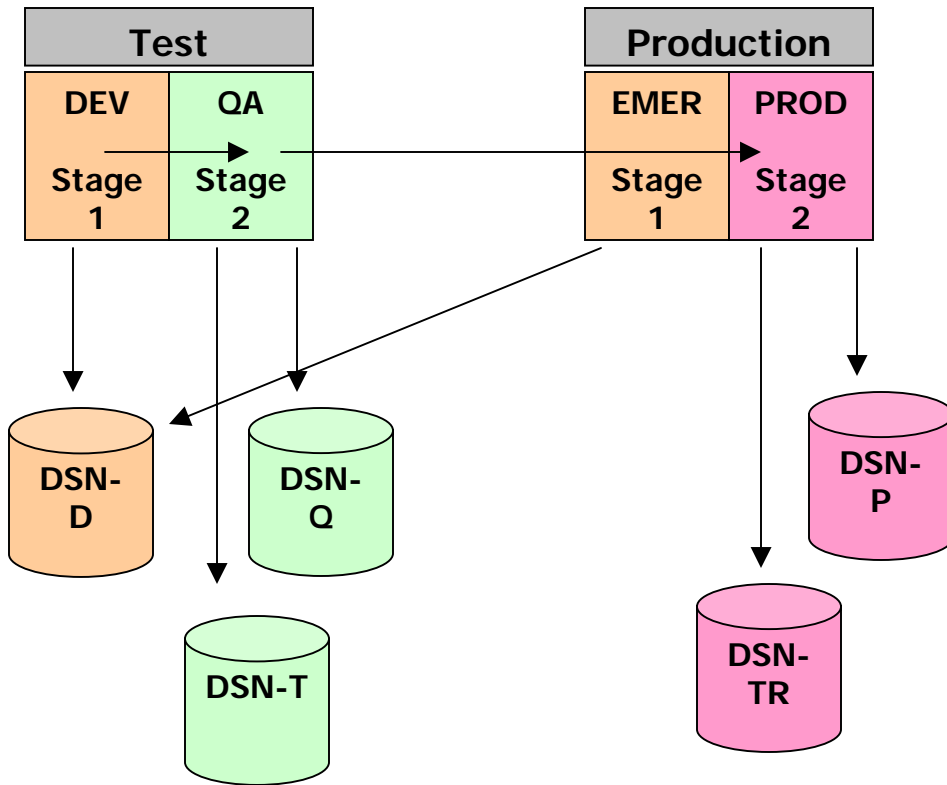
Information Gathering

Endevor considerations

- Ease of use
- Type definitions
- DBRM library allocation
- Alternate user id
- Processors
- Package back-out

Mapping Life Cycles

Sample Endeavor SLC / DB2 Database Configuration



DB2 programs “added” to Endeavor Test Stage 1 (DEV) invoke bind to DSN-D

DB2 programs “moved” to Endeavor Test Stage 2 (QA) invoke bind to DSN-T and DSN-Q

DB2 programs “moved” to Endeavor Production stage 2 (PROD) invoke bind to DSN-P and DSN-TR

DB2 programs “added” to Endeavor Production Stage 1 (EMER) invoke bind to DSN-D

Connecting DB2 Subsystems

- Endeavor must reside on an LPAR with one DB2 subsystem
- Distributed data facility (DDF) must be configured for DB2
- DB2 location ids must be defined
- BIND package (Remote location id . Collection id)

Processor Considerations

Endevor “Smart” Symbolics

- **Library names**
- **Bind parameters**

Endevor Site Symbolics (R4.0 +)

- **All symbolic values related to DB2**

Endevor Includes

- **Bind code utilized within more than one processor**

Processor Considerations

Smart Symbolics

```
/**          DB2 subsystem / LPAR XREF
/**          -----
/**          390A = DSNP/DSNTR
/**          390B = DSNT/DSNQ
/**          390C = DSND
/**          390D = DSNZ
/**-----
/**  Connecting DB2 Subsystem = DSNZ on 390D
/**-----
//          DB2SYS='&D2S&MVSID' ,
//          MVSID=' 390D' ,
/******    MVSID=' 390C' ,
//          D2S390C=' DSND' ,
//          D2s390D=' DSNZ' ,
```

Processor Considerations

Smart Symbolics

```
/**
/**          DB2 Qualifiers
/**-----
//          DB2QUAL = '&DB2Q&C1SYS(1,3,0)',
/**-----
/**          Application = AAA
/**          -----
//          DB2QAAA = '&D2QAAA&C1STGID',
//          D2QAAA1 = 'DEVL',
//          D2QAAA2 = 'QUAL',
//          D2QAAAP = 'PRDN',
//          D2QAAAT = 'TRNG',
```

Processor Considerations

Site Symbolics

```
/**
/** DESC: Language and System Libraries
/** For: 390A, B, C Endeavor Map
/**
//      #DB2LIB='DB2.DSNX.SDSNLOAD',
/**
//      #COBLIB='SYS1.SIGYCOMP',
//      #COBSLIB='SYS1.SCEELIB',
/**
//      #MACLIB='SYS1.MACLIB',
//      #MODGEN='SYS1.MODGLIB',
// *
//      #CICSLIB='SYS3.CICS.TS13.LOADLIB',
//      #CICSCOB='SYS3.CICS.TS13.COBLIB',
//      #CICSMAC='SYS3.CICS.TS13.MACLIB',
```

Processor Considerations

Building a Bind Card (Sample include)

```
//GENBINDC EXEC PGM=IEBGENER,  
//      MAXRC=0,  
//      COND=(0,EQ,RETBIND)  
//SYSUT2 DD DSN=&&BNDCARDS,DISP=(OLD,KEEP,DELETE)  
//SYSUT1 DD *  
      DSN SYSTEM(&DB2SYS)  
      BIND PACKAGE(&DB2RMSTD&DB2RMT....&DB2CID)      -  
      MEMBER(&C1ELEMENT)                                -  
      OWNER(&DB2OWN)          QUALIFIER(&DB2QUAL)      -  
      SQLERROR(&DB2SQLE)      VALIDATE(&DB2VAL)        -  
      ISOLATION(&DB2ISO)      RELEASE(&DB2REL)         -  
      ACTION(&DB2ACT)         EXPLAIN(&DB2EXPL)  
  
      END  
/*
```

Processor Considerations

The Bind Step

```
//BIND EXEC PGM=IKJEFT01,MACRC=4,  
// COND((4,LT,RETBIND),(4,LT,GENBIND),(0,LT,SWAPOUT))  
//DBRMLIB DD DSN=&DBRMLIB,DISP=SHR  
//SYSTSPRT DD DSN=&&BINDLIST,DISP=SHR  
//SYSTSIN DD DSN=&&BNDCARDS,DISP=(OLD,DELETE)
```

Processor Considerations

Invoking the bind:

- Entry stage **GENERATE**
- Entry stage **MOVE**
- Subsequent stage **MOVE**
- Unused **GENERATE**



Processor Considerations

Entry stage **GENERATE**

- Allocate additional &TEMP files (I.e. Bind listing file, bind card parm file)
- Ensure DB2 pre-compile (or compile if using DB2 coprocessor) step outputs to Endeavor DBRM library
- Retain existing compile and link edit steps; append BIND steps
- Examine CC checks to ensure bypass bind steps on bad RC

Processor Considerations

Entry stage **GENERATE**

- **BIND01 – retrieve user specified card**
 - Use CONWRITE utility (C1BM3000 retrieve will work)
 - Specify “no signout” and “search” (if applicable)
 - Ensure type is defined
 - Write element to temp file (&&BNDCARDS)

- **BIND02 – generate application default bind**
 - Use CC check to bypass if step01 CC=0
 - IEBGENER into temp file (&&BNDCARDS)
 - Use in-stream data to utilize symbolic values

Processor Considerations

Entry stage **GENERATE**

- **BIND03 – swap out user id for alternate id**
 - For use in DB2 address space (ASXBUSER)
 - Follow instructions in PTF #OA00504
 - ✓ Note: PTF Not required >= Release 4.0
- **BIND04 – execute the bind (IKJEFT01)**
 - **MAXRC=4**
 - **SYSTSIN=&&BNDCARDS**
 - **SYSTSPRT=bind listing library**
- **BIND05 – swap alternate id back to user id**
- **Add bind listing file to existing store/print listing step**

Processor Considerations

Entry / subsequent stage **MOVE**

- Allocate additional &TEMP files (I.E. Bind listing file, bind card parm file)
- Retain existing library copy steps; Append BIND steps
- Examine CC checks to ensure bypass bind steps on bad RC
- Insert BIND01 – BIND05 steps as coded in *entry stage* **GENERATE**
- Repeat step with &C1STGID qualification (IF-THEN-ELSE) if BIND card values vary and cannot be symbolically coded
- BIND06 – copy DBRM file to next stage
- Add bind listing file to existing store/print listing step
- BIND07 – copy bind listing file to next stage

Processor Considerations

Unused *GENERATE*

- Create new processor to execute bind at particular stage in the SLC.
 - Insert BIND01 – BIND05 steps as coded in *entry stage GENERATE*.
 - Add step to store/print listing.
- * **Caution: MCF entries are modified (LAST GEN DATE) and may cause element validation errors, particularly if the element is a static subroutine. NOT RECOMMENDED!**

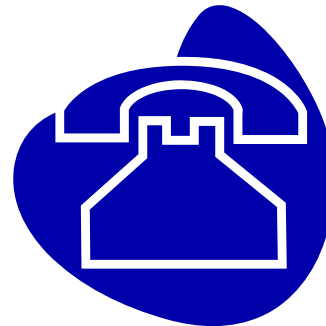
Processor Considerations

- **Plan to test, test, test**
- **Utilize new processors (if feasible) – makes implementation and back-out a snap**
- **Utilize new processor groups**
- **Prepare to back-out**

Communication

- **Prepare your users**

- **Project details should be accessible**
- **Provide sample output**
- **Describe common errors, document message codes, provide resolutions**
- **Contact information**



- **Establish formal Endeavor / DB2 relationship**

Communication

Sample FAQ entry:

Qxx. I submitted a job to ADD/UPDATE/GENERATE my program in Endeavor and the DB2 BIND step fails with RC=12. Do I need to contact Endeavor Administration to correct this error? The messages issued by DB2 look like these:

READY

DSN SYSTEM(DSNZ)

DSNZ NOT VALID SUBSYSTEM ID, COMMAND TERMINATED

bind card parameter information is here.....

READY

COMMAND BIND NOT FOUND

READY

END

READY

Axx. Not necessarily. First CHECK TO BE CERTAIN your Endeavor job ran on 390D. The auto binds are currently setup to connect to the DB2 subsystem on 390D. If you try to run your job on 390A, B or 390C, the bind step will fail with the above error message. If you ran your job on 390D – contact *your-service-desk @ xxx-xxx-xxxx* for assistance.

Questions

