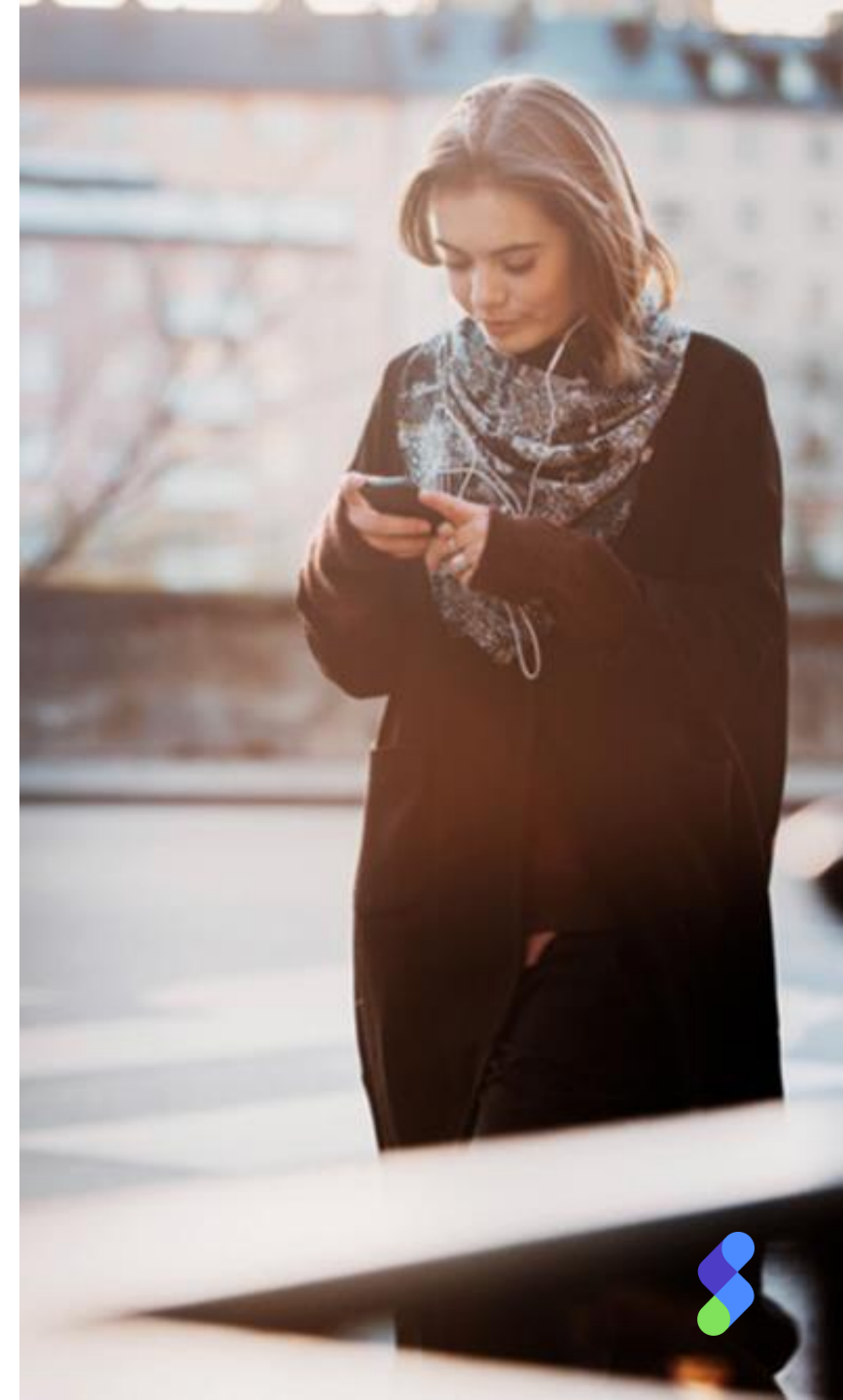# Migrating your IMS databases to DB2 to reduce risk and cost, without making changes to application code!

- Presented by:

- Mike Picou – Migration Solutions Consultant
- Bill Bostridge – VP - Data Infrastructure Optimization

Agenda

- Introduction

- Why migrate from IMS to DB2

- Customer Stories

- Buy-in across your organization

- Value Proposition

- Q&A

- Next Steps

IMS

- IMS is a hierarchical database and has been in organizations for decades

- Applications are mission critical and complicated

- Huge costs in supporting IMS

    - Staffing, IMS DB licensing, 3rd party management tools for IMS (BMC, Compuware, CA, IBM)

    - Enforced upgrades to current versions of IMS DB (On average every 2 years)

- Benefits in moving to Db2

    - Web applications, new fields, new lines of business

    - Drastically reduced infrastructure costs

- But rewriting applications is hard to get right

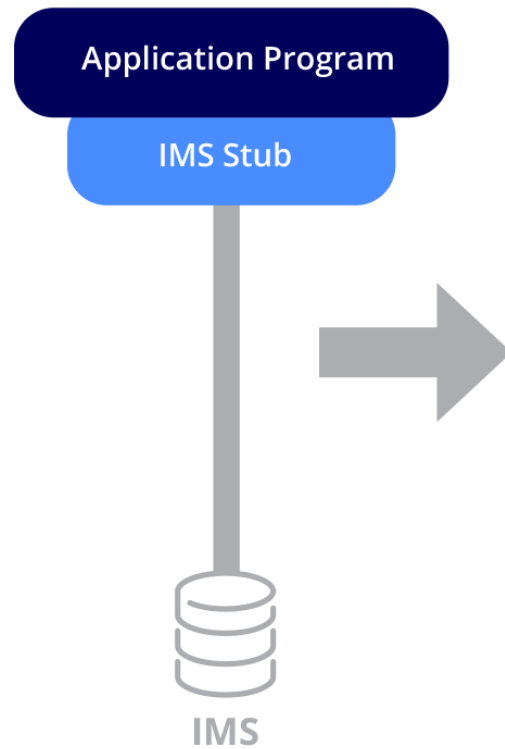   - DL/2 doesn't touch application code

What is DL/2?

- DL/2 is a combination of IMS to Db2 migration and mapping tools to migrate IMS segments to Db2 tables

- DL/2 is the interface between the IMS applications, which do not change and the migrated IMS data that is now residing in Db2

- DL/2 allows the replacement of IMS DB with Db2, eliminating the need for IMS support tools (BMC, Compuware, etc.)

- DL/2 consolidates your database maintenance activities to one platform – Db2

- DL/2 is not replicating data to Db2. DL/2 is replacing IMS DB so that all your production data is now residing in Db2

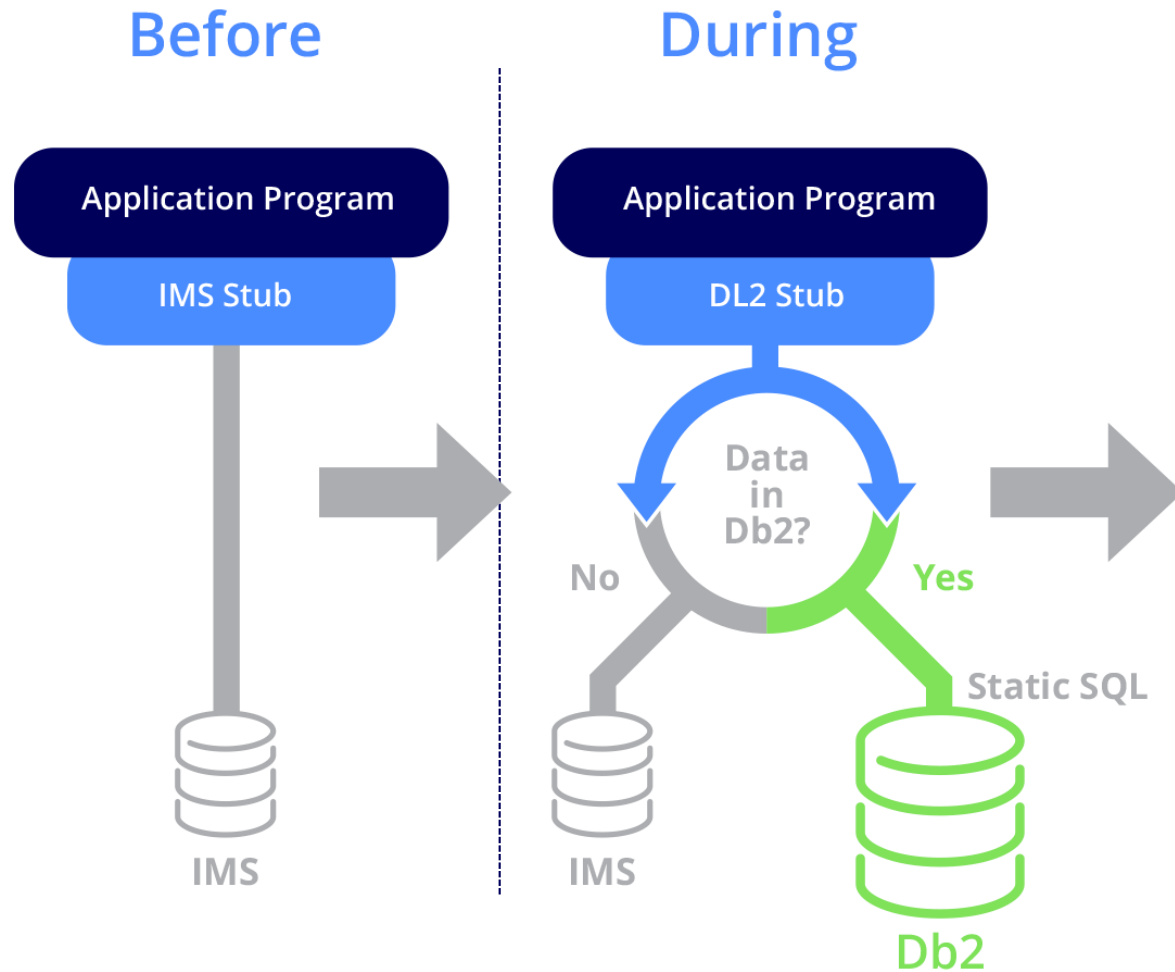- DL/2 is a transparent data migration route from IMS DB on z/OS to DB2 on z/OS
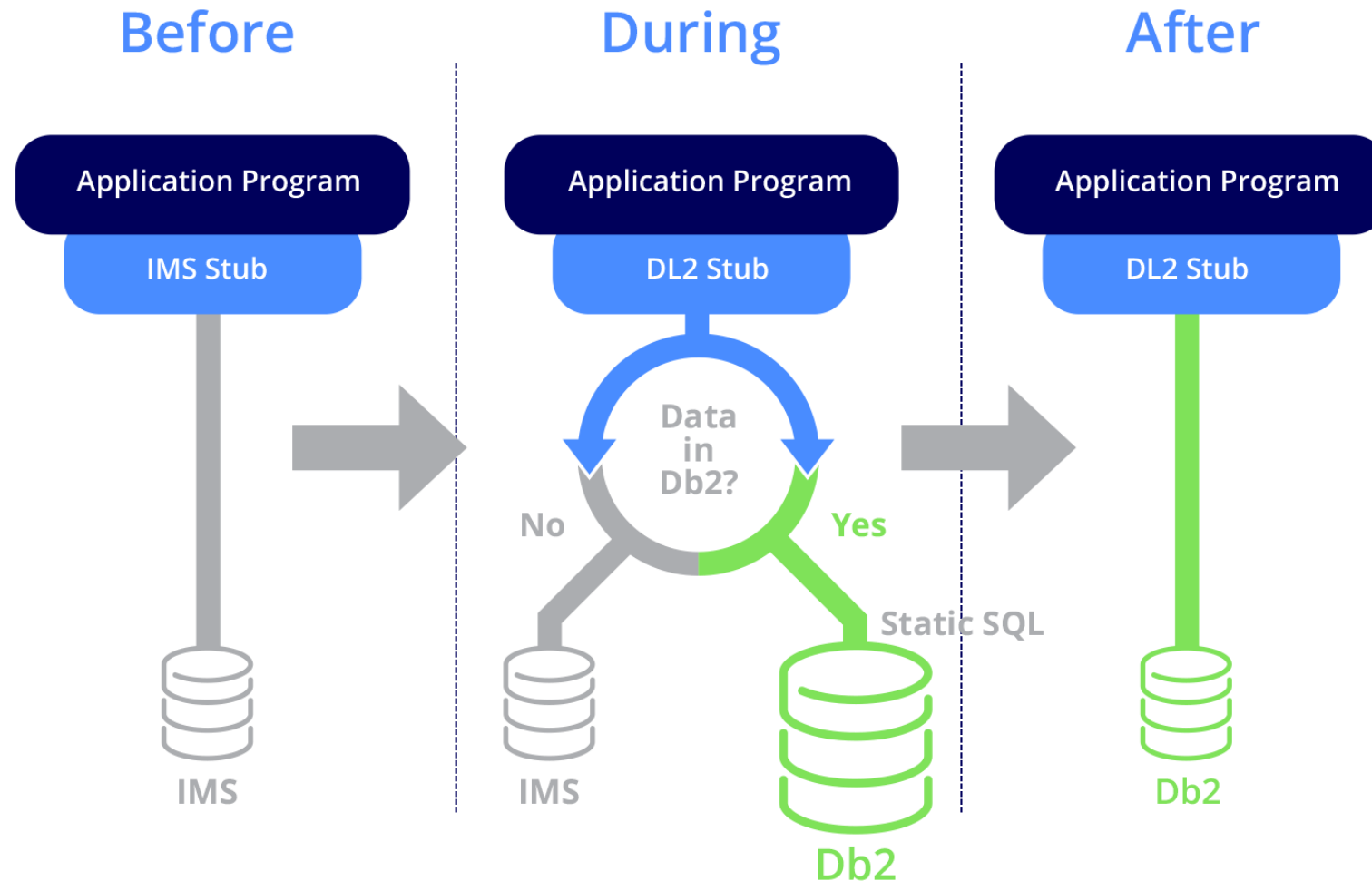
Transparent Migration Process Overview

Before

Application Program

IMS Stub

IMS

Transparent Migration Process Overview

Transparent Migration Process Overview

The justification for DL/2

- DL/2 makes it easier and less risky to move IMS applications to Db2 to alleviate staffing challenges and licensing costs associated with IMS.

- DL/2 converts the underlying database to Db2 without changing the IMS application programs, and eliminates the need for third-party IMS tools or IMS DBA and Systems expertise.

- Once databases have been converted, SQL and Db2 skills can be used to make changes and/or add new functionality to existing applications

- In many cases DL/2 becomes a stepping stone to full application modernization using native Db2/SQL

Benefits of transparent data migration

- Simplicity
  - One less platform to maintain
  - Eliminate replication of data – "one version of the truth"
  - Eliminate on-going DBA and programming support for IMS DB
  - Consolidation of application maintenance to Db2 /SQL skills

- Cost Reduction
  - Eliminate software licensing costs for IMS DB and IMS tools
  - Eliminate enforced IMS DB upgrades – V13, V 14 V15 and so on…

- Reduce Migration Risk
  - Over time, rewrite components to native SQL
  - Make an orderly move to modernize your applications

Benefit to IMS application user community

Question? – Does you organization just want to remove IMS DB and the associated support overhead and on-going licensing costs?
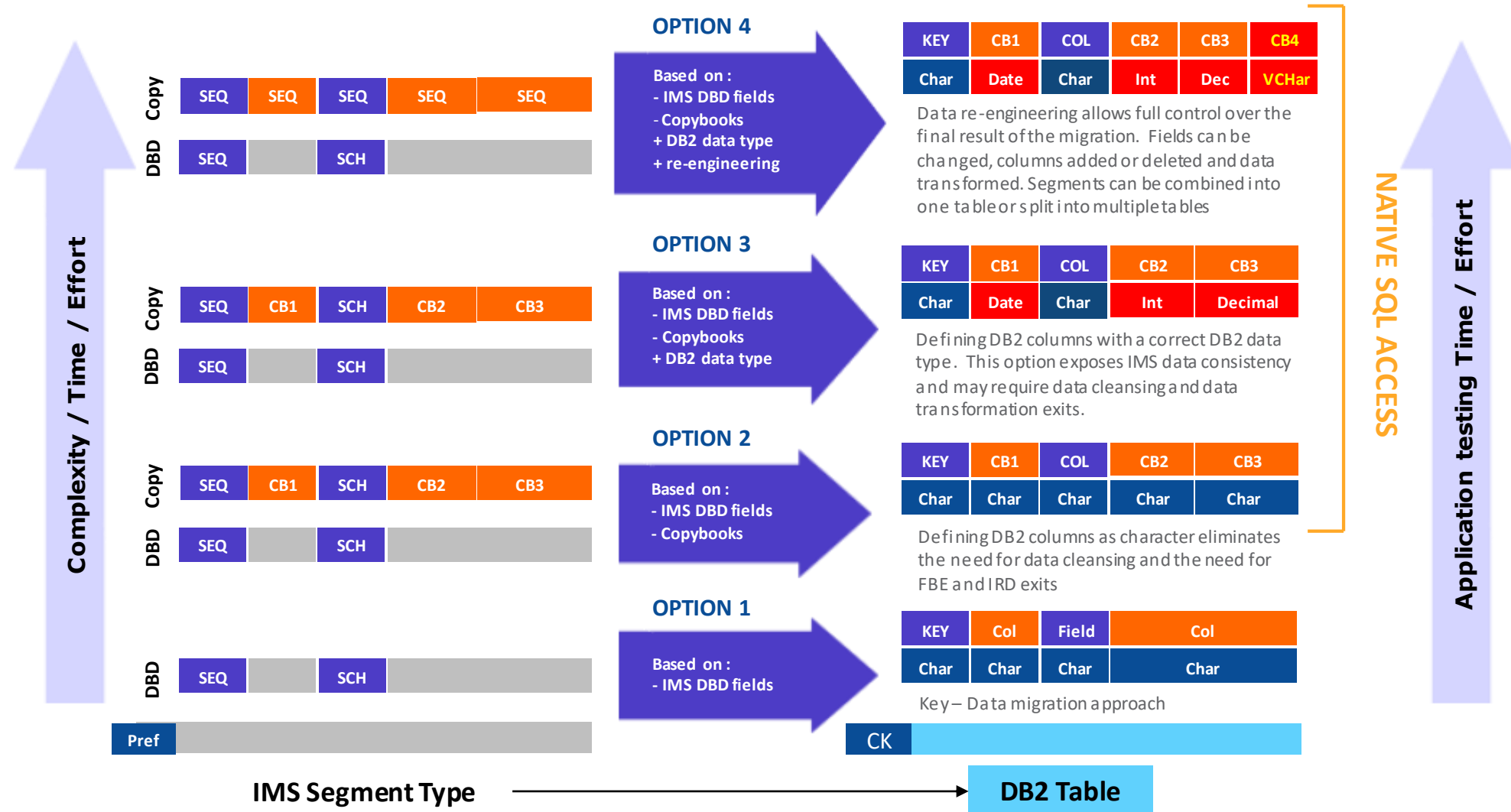
"We just want to get out of IMS"

OR

We would like to get our IMS data into a relational model in DB2 to further the life of the applications and support new business initiatives
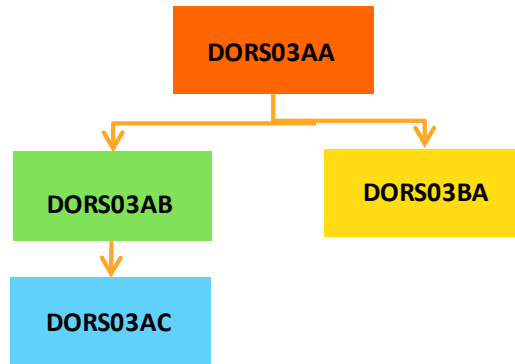
"We want to be in DB2"

Increased Agility

Higher Service Levels

Cost Reduction

Reduced Complexity

Migration options

Complexity / Time / Effort

Application testing Time / Effort

NATIVE SQL ACCESS

OPTION 4

Based on :
- IMS DBD fields
- Copybooks
+ DB2 data type
+ re-engineering

KEY	CB1	COL	CB2	CB3	CB4
Char	Date	Char	Int	Dec	VCHar

Data re-engineering allows full control over the final result of the migration. Fields can be changed, columns added or deleted and data transformed. Segments can be combined into one table or split into multiple tables

| | Copy | SEQ | SEQ | SEQ | SEQ | SEQ |
| DBD | | SEQ | | SCH | | |

OPTION 3

Based on :
- IMS DBD fields
- Copybooks
+ DB2 data type

KEY	CB1	COL	CB2	CB3
Char	Date	Char	Int	Decimal

Defining DB2 columns with a correct DB2 data type. This option exposes IMS data consistency and may require data cleansing and data transformation exits.

| | Copy | SEQ | CB1 | SCH | CB2 | CB3 |
| DBD | | SEQ | | SCH | | |

OPTION 2

Based on :
- IMS DBD fields
- Copybooks

KEY	CB1	COL	CB2	CB3
Char	Char	Char	Char	Char

Defining DB2 columns as character eliminates the need for data cleansing and the need for FBE and IRD exits

| | Copy | SEQ | CB1 | SCH | CB2 | CB3 |
| DBD | | SEQ | | SCH | | |

OPTION 1

Based on :
- IMS DBD fields

KEY	Col	Field	Col
Char	Char	Char	Char

Key – Data migration approach

| DBD | SEQ | | SCH | |

| Pref | |

| CK | |

IMS Segment Type →

DB2 Table

Db2 database design options - Inputs

- IMS database definition (DBD)
- Optionally, IMS segment copybooks

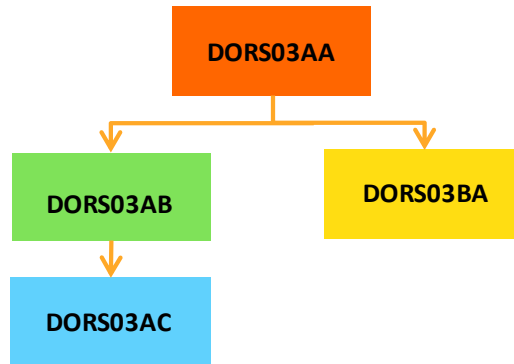- A degree of normalization is achievable within confines of existing IMS database

```
SEGM    NAME=DORS03AA,PARENT=0,BYTES=212
            …
 FIELD NAME=(CMDORNUM,SEQ,U),BYTES=6,START=1
 FIELD NAME=CMOFFSTA,BYTES=002,START=007
 FIELD NAME=CMCDATE,BYTES=008,START=089

SEGM    NAME=DORS03AB,PARENT=DORS03AA,BYTES=290
            …
 FIELD NAME=(CMREFIX,SEQ,U),BYTES=2,START=1
 FIELD NAME=CMOPFXTA,BYTES=12,START=270
            …
LCHILD NAME=(DORX03AB,DORXDB3A),PTR=INDX
XDFLD   NAME=XSTADOCN,NULLVAL=C' ',
         SRCH=(CMOPFXTA,CMCMPREFIX)
```

```
01  DORS03AA.
  05  CMDORNUM                        PIC X(6).
  05  CMOFFSTA                        PIC XX.
  05  ------------------------------------
  05  CMCDATE.
      10  CMCDATE-CT            PIC 99.
      10  CMCDATE-YMD.
          15  CMCDATE-YR        PIC 99.
          15  CMCDATE-MO        PIC 99.
          15  CMCDATE-DY        PIC 99.
  05  HEIGHT                          PIC 99.
  05  WEIGHT                          PIC XXX.
  05  ------------------------------------
…
```

Db2 database design options – Option 1

DORS03AA
DORS03AB
DORS03BA
DORS03AC

- DB2 columns derived from DBD fields
- All DB2 columns are CHAR

```
SEGM    NAME=DORS03AA,PARENT=0,BYTES=212
          …
 FIELD NAME=(CMDORNUM,SEQ,U),BYTES=6,START=1
 FIELD NAME=CMOFFSTA,BYTES=002,START=007
 FIELD NAME=CMCDATE,BYTES=008,START=089

SEGM    NAME=DORS03AB,PARENT=DORS03AA,BYTES=290
          …
 FIELD NAME=(CMREFIX,SEQ,U),BYTES=2,START=1
 FIELD NAME=CMOPFXTA,BYTES=12,START=170
          …
LCHILD NAME=(DORX03AB,DORXDB3A),PTR=INDX
XDFLD  NAME=XSTADOCN,NULLVAL=C' ',
        SRCH=(CMOPFXTA,CMCMPREFIX)
```
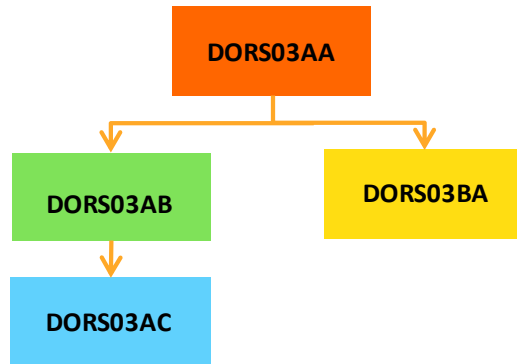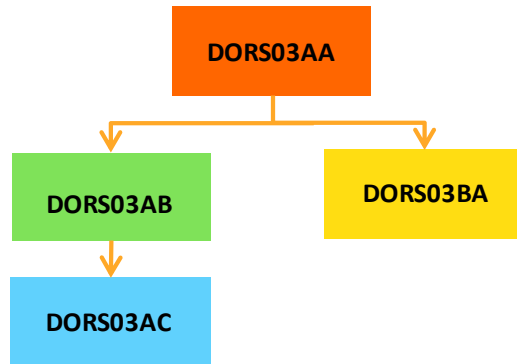
```
CREATE TABLE DB2XEST.DORS03AA
   (CMDORNUM    CHAR(6)    NOT NULL WITH DEFAULT
   ,CMOFFSTA    CHAR(2)    NOT NULL WITH DEFAULT
   ,SEGDAA _01 CHAR(80)   NOT NULL WITH DEFAULT
   ,CMCDATE     CHAR(8)    NOT NULL WITH DEFAULT
   ,SEGDAA _02 CHAR(116)  NOT NULL WITH DEFAULT)
 PRIMARY KEY(CMDORNUM));

CREATE TABLE DB2XEST.DORS03AB
   (CMDORNUM    CHAR(6)    NOT NULL WITH DEFAULT
   ,CMPREFIX    CHAR(2)    NOT NULL WITH DEFAULT
   ,SEGAB_01    CHAR(168)  NOT NULL WITH DEFAULT
   ,CMOPFXTA    CHAR(12)   NOT NULL WITH DEFAULT
   ,SEGAB_02    CHAR(104)  NOT NULL WITH DEFAULT)
  PRIMARY KEY(CMDORNUM, CMPREFIX)
,FOREIGN KEY(CMDORNUM) REFERNCES DB2XEST.DORS03AA
          ON DELETE CASCADE);
```

Db2 database design options – Option 2


```
01  DORS03AA.
   05   CMDORNUM                        PIC X(6).
   05   CMOFFSTA                        PIC XX.
   05   --------------------------------------
   05   CMCDATE.
      10   CMCDATE-CT                    PIC 99.
      10   CMCDATE-YMD.
         15   CMCDATE-YR                 PIC 99.
         15   CMCDATE-MO                 PIC 99.
         15   CMCDATE-DY                 PIC 99.
   05   HEIGHT                           PIC 99.
   05   WEIGHT                           PIC XXX.
   05   --------------------------------------
...
```

```
SEGM   NAME=DORS03AA,PARENT=0,BYTES=212
         …
 FIELD NAME=(CMDORNUM,SEQ,U),BYTES=6,START=1
 FIELD NAME=CMOFFSTA,BYTES=002,START=007
 FIELD NAME=CMCDATE,BYTES=008,START=089

SEGM   NAME=DORS03AB,PARENT=DORS03AA,BYTES=290
         …
 FIELD NAME=(CMREFIX,SEQ,U),BYTES=2,START=1
 FIELD NAME=CMOPFXTA,BYTES=12,START=170
         …
LCHILD NAME=(DORX03AB,DORXDB3A),PTR=INDX
XDFLD  NAME=XSTADOCN,NULLVAL=C' ',
        SRCH=(CMOPFXTA,CMCMPREFIX)
```
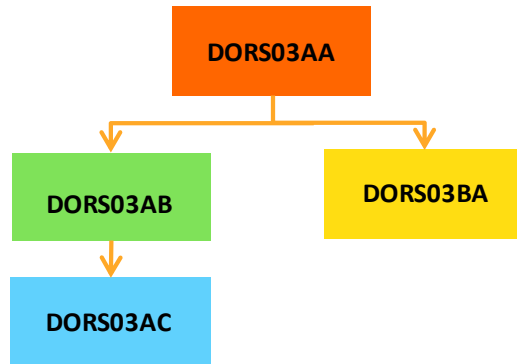
```
CREATE TABLE DB2XEST.DORS03AA
  (CMDORNUM       CHAR(6)    NOT NULL WITH DEFAULT
  ,OFFNSE_STATUS  CHAR(2)    NOT NULL WITH DEFAULT
  ,------------------------------------------------
  ,CMC_DT_CC      CHAR(2)    NOT NULL WITH DEFAULT
  ,CMC_DT_YR      CHAR(2)    NOT NULL WITH DEFAULT
  ,CMC_DT_MO      CHAR(2)    NOT NULL WITH DEFAULT
  ,CMC_DT_DY      CHAR(2)    NOT NULL WITH DEFAULT
  ,OBJECT_HEIGHT  CHAR(2)    NOT NULL WITH DEFAULT
  ,OBJECT_WEIGHT  CHAR(3)    NOT NULL WITH DEFAULT
  ,-----------------------------------------------)
  …
  PRIMARY KEY(CMDORNUM));
```

Db2 database design options – Option 3


```
01  DORS03AA.
   05  CMDORNUM                      PIC X(6).
   05  CMOFFSTA                      PIC XX.
   05  ---------------------------------------
   05  CMCDATE.
      10  CMCDATE-CT                 PIC 99.
      10  CMCDATE-YMD.
         15  CMCDATE-YR              PIC 99.
         15  CMCDATE-MO              PIC 99.
         15  CMCDATE-DY              PIC 99.
   05  HEIGHT                        PIC 99.
   05  WEIGHT                        PIC XXX.
   05  ---------------------------------------
...
```

```
SEGM   NAME=DORS03AA,PARENT=0,BYTES=212
          …
 FIELD NAME=(CMDORNUM,SEQ,U),BYTES=6,START=1
 FIELD NAME=CMOFFSTA,BYTES=002,START=007
 FIELD NAME=CMCDATE,BYTES=008,START=089

SEGM   NAME=DORS03AB,PARENT=DORS03AA,BYTES=290
          …
 FIELD NAME=(CMREFIX,SEQ,U),BYTES=2,START=1
 FIELD NAME=CMOPFXTA,BYTES=12,START=170
          …
LCHILD NAME=(DORX03AB,DORXDB3A),PTR=INDX
XDFLD  NAME=XSTADOCN,NULLVAL=C' ',
          SRCH=(CMOPFXTA,CMCMPREFIX)
```

```
CREATE TABLE DB2XEST.DORS03AA
   (CMDORNUM        CHAR(6)    NOT NULL WITH DEFAULT
   ,OFFNSE_STATUS   CHAR(2)    NOT NULL WITH DEFAULT
   ,-------------------------------------------------
   ,CMC_DATE        DATE       NOT NULL WITH DEFAULT
   ,OBJECT_HEIGHT   DEC(2,0)   NOT NULL WITH DEFAULT
   ,OBJECT_WEIGHT   CHAR(3)    NOT NULL WITH DEFAULT
   ,-------------------------------------------------)
   …
   PRIMARY KEY(CMDORNUM));
```

Db2 database design options – Option 4


```
01  DORS03AA.
   05  CMDORNUM                      PIC X(6).
   05  CMOFFSTA                      PIC XX.
   05  --------------------------------------
   05  CMCDATE.
       10  CMCDATE-CT                PIC 99.
       10  CMCDATE-YMD.
           15  CMCDATE-YR            PIC 99.
           15  CMCDATE-MO            PIC 99.
           15  CMCDATE-DY            PIC 99.
   05  HEIGHT                        PIC 99.
   05  WEIGHT                        PIC XXX.
   05  --------------------------------------
...
```

```
SEGM    NAME=DORS03AA,PARENT=0,BYTES=212
        ...
 FIELD NAME=(CMDORNUM,SEQ,U),BYTES=6,START=1
 FIELD NAME=CMOFFSTA,BYTES=002,START=007
 FIELD NAME=CMCDATE,BYTES=008,START=089

SEGM    NAME=DORS03AB,PARENT=DORS03AA,BYTES=290
        ...
 FIELD NAME=(CMREFIX,SEQ,U),BYTES=2,START=1
 FIELD NAME=CMOPFXTA,BYTES=12,START=170
        ...
LCHILD NAME=(DORX03AB,DORXDB3A),PTR=INDX
XDFLD  NAME=XSTADOCN,NULLVAL=C' ',
       SRCH=(CMOPFXTA,CMCMPREFIX)
```

```
CREATE TABLE DB2XEST.DORS03AA
   (CMDORNUM       CHAR(6)    NOT NULL WITH DEFAULT
   ,OFFICE_STATUS  CHAR(2)    NOT NULL WITH DEFAULT
   ,---------------------------------------------
   ,CMC_DATE       DATE
   ,CMC_DATE_FLAG  CHAR(1)    NOT NULL WITH DEFAULT
   ,OBJECT_HEIGHT  DEC(2,0)   NOT NULL WITH DEFAULT
   ,OBJECT_WEIGHT  CHAR(3)    NOT NULL WITH DEFAULT
   ,---------------------------------------------)
   ...
   PRIMARY KEY(CMDORNUM));
```

Exit performs date validation and sets value in DB2 to NULL if invalid. It also sets the CMC_DATE_FL value if CMC_DATE is '99999999' – a known 'misuse' of CMCDATE

Future application programming

- Future Programming Changes

  ** Data is now in DB2 and may be accessed/changed by either IMS/DLI or Db2/SQL calls

- IMS/DLI

  Continue to use existing without changing application code

  Create new DLI calls – use existing PSBs

- Db2/SQL

  * Can replace DLI calls with SQL reading/updating DB2 tables directly

  Compile process will create program package that is separate from DL2 PSB driver package

  New SQL may be added and may be helpful for analytical style processing

  May allow elimination of some COBOL filtering code

Post migration with DL/2

- SQL can be used to access data

- DB2 built in functions can be used for analytical/decision support type processing
  - Average, Count, etc – Option 3 conversion

- Faster database changes

- Easier modeling, give various database models a try
  - Less COBOL coding to filter records, the records returned are the ones you want

- Analysis tools help spot potential performance problems before production implementation

Post migration with DL/2 cont.

- Sophisticated tuning tools available

- Can test/develop queries in ad-hoc tools such a SPUFI without writing a COBOL program

- Deliver requests for information using ad-hoc reporting tools much faster than writing a COBOL program

- Over time, rewriting to native DB2/SQL maybe of benefit?

Increased Agility

Higher Service Levels

Cost Reduction

Reduced Complexity

Over 100 projects completed globally

- Finance
- Insurance
- Healthcare
- Local and State Government
- Manufacturing
- Retail

- All customers have gained significant fiscal and operational benefits by migrating to Db2 with DL/2

Hear first hand from Damon Anderson

Director of Data Services at Anixter about their successful migration project

http://www.syncsort.com/en/Resource-Center/Mainframe/Webcasts/IMS-to-DB2-Migration

Customer Stories and Use Cases

Johnson Controls is a global diversified technology and multi industrial leader serving a wide range of customers in more than 150 countries. They used IMS on their mainframe and needed to add an additional security layer on top of IMS & Db2. However, IMS was not going to be able to support the new security requirements so they needed to migrate their IMS/DB based applications to Db2. We were able to use DL/2 to quickly and transparently migrate their data to Db2 with minimal risk. They were also able to enjoy significant savings by getting rid of licensing fees for IMS or other supporting tools.

Security

New security requirements made moving to DB2 a necessity

OBJECTIVE

- Support more stringent security requirements for mainframe data

CHALLENGE

- Needed to add security layer on top of IMS & Db2

- Security solution only supported Db2

SOLUTION

- Migrate IMS data to DB2 using Syncsort DL/2

BENEFITS

- Allowed customer to satisfy new security requirement with minimal risk

- Avoided a an IMS upgrade from IBM

- Eliminated IMS software and tools licensing costs

NASCO is owned by and exclusively serves Blue Cross and Blue Shield Plans as a claims processing provider for over 23 million members across the United States. They were concerned with the costs they were spending for IMS and related tools . They also knew they were at risk of a skills shortage as their IMS staff members were headed to retirement and needed to take action to reduce that risk. After looking at all the alternatives, they quickly decided that using DL/2 to migrate their IMS data to Db2 was their best option.

BCBS - Claims Processing Provider

Customer was very focused on reducing cost, their team very fully engaged right from the start. CTO buy-in, DBA team and Application teams all supporting the solution in the initial engagement and through the successful Pilot.

NASCO®

OBJECTIVE

- Eliminate IMS DB and IMS tools to reduce significant MLC charges from IBM

CHALLENGE

- Hybrid 3rd party solution from EDS that was complex

- Complex applications with custom interfaces to IBM resources

SOLUTION

- Migrate IMS data to DB2 using Syncsort DL/2

BENEFITS

- Most efficient migration approach to reduce long term costs

- Easier management with consolidation to Db2

- Reduced risk as IMS skills are heading toward retirement

The **Florida Department of Corrections** consistently added functionality to their IMS-based inmate management system. However, application changes were becoming more and more cumbersome using IMS/DB. After failing to get Legislative budget approval for an application re-write, they moved the data out of IMS into Db2 using DL/2. They not only satisfied the inmate management system upgrade challenges but were able to save money on software licensing costs and deliver future changes via SQL/Db2 using the native functionality an option 3 migration allowed.

Florida Department of Corrections

Needed to move inmate management system for the entire state to DB2

OBJECTIVE

- Clean up outdated information and provide modernization platform in Db2

CHALLENGE

- Legislature unable to approve funding for multi-million $ rewrite

- Cost reduction and on-going application modernization

SOLUTION

- Migrate IMS data to DB2 using Syncsort DL/2

BENEFITS

- Significant cost reduction with elimination of IMS DB and IMS tools licensing

- Application change requests now delivered via SQL/Db2 skills

- IMS databases migrated to Db2 implementing Db2 native data types

Value proposition to your organization

Management

- Reduced the overall cost of application maintenance
- Increased IT Staff productivity by consolidating critical data into Db2 platform"
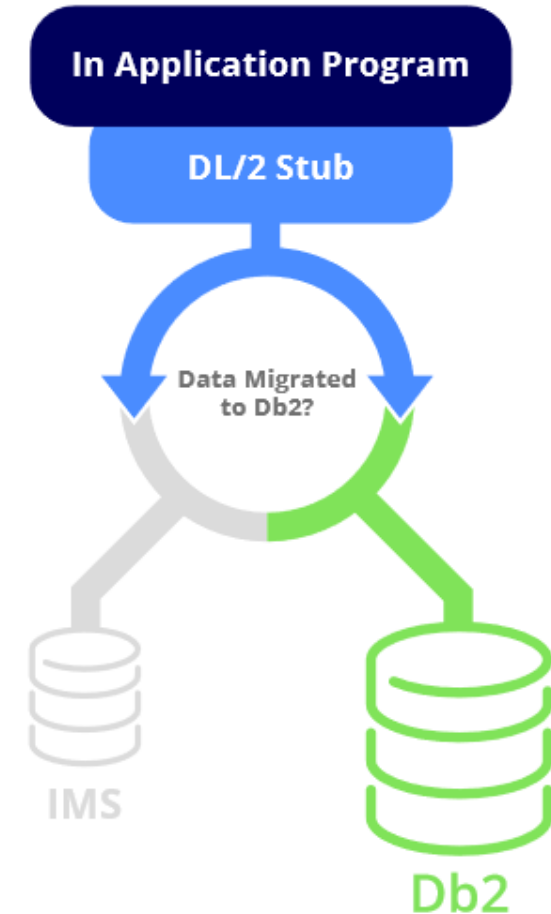- Reduction in software spend by eliminating !MS/DB

Technical

- Db2 provides flexibility to leverage existing data
- Minimized risk by not requiring application changes
- Single operational database environment to support

What are your options?

- Status quo – Do nothing! – Risk as support staff head off to retirement

- DL/2 is the only proven "Commercially Available" transparent data migration tool for IMD DB to Db2 on z/OS

- Code conversion tools – BEWARE
  We have yet to hear of a successful conversion for IMS to DB2

- Over 100 plus migrations completed worldwide

Transparent Data Migration with DL/2

In Application Program

DL/2 Stub

Data Migrated to Db2?

IMS

Db2

Develop the internal business case

Getting buy-in from senior management, the DBA team, and application stakeholders is the key to success!

- Review the current status of your IMS support team
  - What is the timeline for IMS skills to depart the organization
  - What is your backup plan to ensure on-going support of your IMS environment

- Review current IMS DB MLC and IMS Tools licensing costs
  - Costs continue as long as you are still maintaining IMS DB – DL/2 is a one time perpetual license fee

- Discuss with application stakeholder
  - Benefits of shifting data to a relational model in Db2
  - Ease of on-going maintenance
  - Easier access to data
  - Opportunity to modernize applications.

The case for transparent data migration

There are 3 reasons companies choose DL/2 to migrate from IMS to Db2

Extend Life of Applications

Moving to a relational DB model without need to rewrite applications

Staffing Challenges

Fewer and fewer developers and DBA's with IMS Skills

Cost Reduction

License fees for IMS and related tools can be very expensive

Q & A