Legac-E Education

Appendix A

Sample Application

Legac-E Education

Appendix A - Sample Application

Topics **Page**

Appendix A - Sample Application

Introduction

In December 2012 there was a requirement to run a CICS Command Level Workshop in a z/VSE environment.

A review of the established student text led to a desire to have the material re-written, and as a consequence this application was developed with an IBM Supplied sample application providing the inspiration. During 2014 the scope of the application was enhanced to allow its use both with DB2 as an alternative to VSAM files, and to compliment the CICS WEB Enablement course.

In essence the application comprises seven transactions involving eleven programs. (As there are variations to the design the number of programs increases to thirty, not counting two batch programs to populate files). In support of the programs there is a single VSAM KSDS (with an optional AIX and PATH), two Transient Data Queues, and five BMS maps. Finally there are three copy book members although two of them are for administrative purposes given that there potentially are 60 copies of each program.

The program variations arise to provide both for alternative solutions, and use by up to 20 students concurrently. (Student Userids are grouped into batches of twenty.) All elements within the application use two numeric digits so that the resources can be matched to the delegate’s Userid.

Currently the application only exists in COBOL form, but a PLI version is well in hand. (All program names are deliberately seven characters in length to aid the production of the PLI suite.

Note

The application was extended in July 2014 to embrace both DB2/SQL and WEB capable variants (both 3270 Bridge and WEB aware programs). It was enhanced further in 2018 to include examples of RRDS file processing, and the use of SET rather than INTO operands on various commands.

Appendix A - Sample Application

Application Overview

The illustrated opposite is the relationship between transactions and the main programs. (There are two other programs, as all VSAM processing (except Browse operations) is performed in two sub-programs, one for reading and the other for writing.). All transactions and programs are named such that the numeric digits of the Userid avoid delegate duplication.

Transactions / Programs

VMnn / PnnnMNU

This is the menu and the start point of the application all selected functions will return here on completion.

VAnn / PnnnADD

This is the transaction and program combination to add a record to the VSAM KSDS. (The actual write for PnnnADD is performed by program PnnnWTR.)

VBnn / PnnnBRW / PnnnPTH / PnnnAIX

This is the transaction and program combination to browse the VSAM KSDS. (These programs do not use PnnnRDR. The PTH and AIX programs browse via an Alternate Index.)

VDnn / PnnnDEL

This is the transaction and program combination to delete a record from the VSAM KSDS.

VInn / PnnnINQ / CnnnINQ / LnnnINQ

This is the transaction and program combination to display a record from the VSAM KSDS. (The actual read for PnnnINQ is performed by program PnnnRDR.)

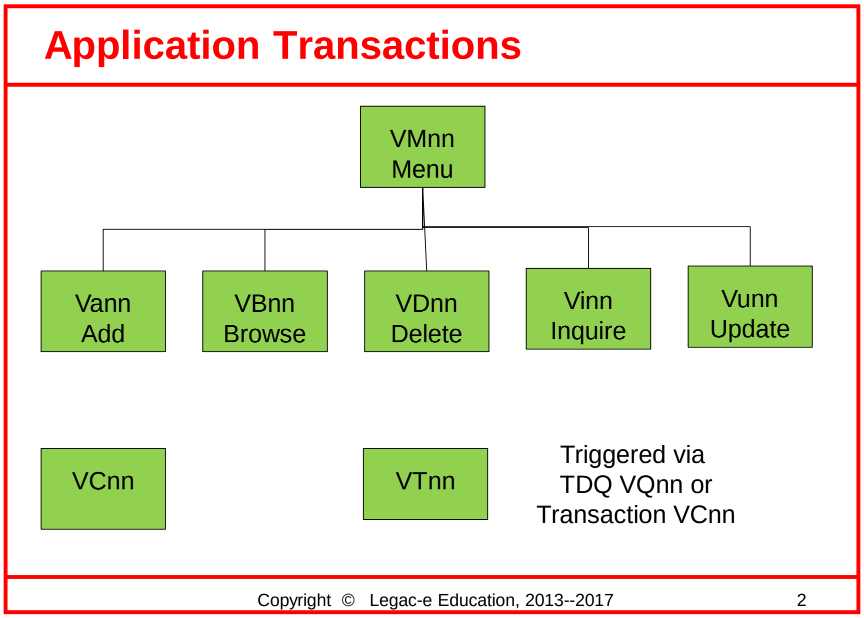
VUnn / PnnnUPD

This is the transaction and program combination to update a record in the VSAM KSDS. (In PnnnUPD the record will be retrieved by PnnnRDR and re-written by PnnnWTR.)

VTnn / PnnnTRG

This transaction and program combination is not driven from the menu and does not use a terminal, but is triggered via a Transient Data Queue.

Appendix A - Sample Application



Application Transactions

VMnn
Menu

Vann
Add

VBnn
Browse

VDnn
Delete

Vinn
Inquire

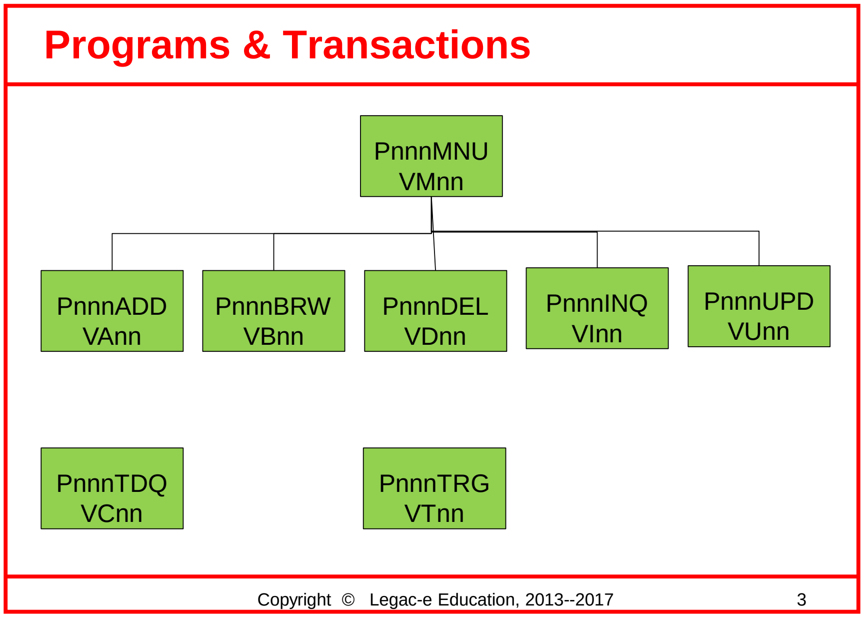
Vunn
Update

VCnn

VTnn

Triggered via
TDQ VQnn or
Transaction VCnn

Copyright © Legac-e Education, 2013--2017 2



Programs & Transactions

PnnnMNU
VMnn

PnnnADD
VAnn

PnnnBRW
VBnn

PnnnDEL
VDnn

PnnnINQ
VInn

PnnnUPD
VUnn

PnnnTDQ
VCnn

PnnnTRG
VTnn

Copyright © Legac-e Education, 2013--2017 3

VCnn / PnnnTDQ

This is a stand-alone Transaction / Program to trigger VTnn so that the intra-partition VQnn can be cleared if necessary.

Appendix A - Sample Application

BMS Maps

The application uses six maps.

MnnnMNU

This is the initial menu delivered by **PnnnMNU**.

MnnnALT

This is the map for for obtaining the alternate key when programs **PnnnPTH** or **PnnnAIX** are used.

MnnnDET

This is an Input/Output used by the ADD, DEL and UPD programs.

MnnnDIS

This is an output only map used by **PnnnINQ** to display the contents of a record.

MnnnBRW

This data display is for the browse functions.

NnnnMNU

This is an alternative menu used in conjunction with programs **XnnnMNU** and **ZnnnMNU**. Unlike **PnnnMNU**, both, **XnnnMNU** and **ZnnnMNU** are designed for re-entry , with **XnnnMNU** being used to demonstrate XCTL, and **ZnnnMNU** being used to demonstrate START.

Appendix A - Sample Application

Files

FnnnKSDS

This is a VSAM KSDS file (**FnnnKSDS**) with a record size of 74 bytes and a key contained within the first 6 bytes. The primary key is a single letter, either D or S, followed by five numeric digits. The file records are described by a single copy book **F74REC**.

The file is loaded by program **KSDSLOAD**.

The alternate index (AIX) uses the Designer name within the base record as a key.

FnnnRRDD

This is a VSAM RRDS file (**FnnnRRDD**) which only contains the records prefixed by the letter D, from the KSDS file. These records are 68 bytes in size, and there are 65,000 slots in the file.

The file is loaded by program **RRDSLOAD**.

FnnnRRDS

This is a VSAM RRDS file (**FnnnRRDD**) which only contains the records prefixed by the letter S, from the KSDS file. These records are 68 bytes in size, and there are 65,000 slots in the file, which means that KSDS records with key values greater than S65000 are not loaded.

The file is loaded by program **RRDSLOAD.**

Appendix A - Sample Application

Transient Data Queues

The application uses two Transient Data Queues, one intra-partition and one extra-partition.

VQnn

This is an Intra-partition queue used to collect audit records from Transaction VDnn.

VXnn

This is an extra-partition data set used as the audit repository by VTnn.

Appendix A - Sample Application

Copy Books / Includes

There are five copy books but two, PDCOMMON and WSCOMMON, are only present to aid maintenance of the multiple copies of the programs and are not recommended as part of a normal application design.

F100REC

This describes the record layout of member ENGINES in &SYSUID..CICS.COB and is used in the KSDSLOAD program, which builds the KSDS Base Cluster.

F74REC

This describes the record layout of the actual KSDS Base Cluster.

PDCOMMON

This is the Procedure Division copy book used in most programs to ensure that Working Storage fields are populated with the correct identities to ensure that each User’s program has consistent naming and does not conflict with any other User. (This approach would not be recommended in a production environment, it is used here merely to aid administration of the course material.)

SETFILE

Contains the logic to select one or other of two RRDS files based on the LOCO-TYPE.

WSCOMMON

This is the Working Storage copy book defining the various fields used by PDCOMMON. (This approach would not be recommended in a production environment, it is used here merely to aid administration of the course material.)

Appendix A - Sample Application

How is the application used?

Standard CICS API

A number of exercises are dispersed throughout the course material which provide the opportunity to code and test various common CICS API commands. This all provides practice in map generation and program preparation.

PnnnMNU

This will require the inclusion of a SEND MAP for MnnnMNU before it can be run. All programs except this one, include a RETURN which sets the transaction code to display the initial menu.

PnnnBRW

This will require all the commands associated with bi-directional browsing of the Base Cluster to be included. It also facilitates the use of various AID keys to control the browsing process.

PnnnDEL

This will require a DELETE command to be inserted to produce a functioning program. It also includes the use data extraction and formatting commands plus TDQ write operations. The latter aid understanding of TDQ trigger events.

PnnnINQ

This program is used in several exercises.

Its first use is to demonstrate EXEC CICS LINK using the traditional COMMAREA. An alternative version, CnnnINQ, together with CnnnRDR, is used to demonstrate CHANNELS and CONTAINERS. Another version, LnnnINQ with LnnnRDR, is used to demonstrate COBOL CALL rather than CICS LINK., Yet a third variant SnnnINQ and SnnnRDR use the SET parameter rather than INTO to conserve storage on READ operations.

Finally it is used to show how to send an unformatted message to the screen.

Appendix A - Sample Application

PnnnRDR

This program is used in several exercises.

Its basic role is to demonstrate the use of READ and READ for UPDATE commands.

Later a version **CnnnRDR** is used with **CnnnINQ** to demonstrate CHANNELS and CONTAINERS.

Its final use is as **LnnnRDR** with **LnnnINQ** to demonstrate the use of COBOL CALL rather than EXEC CICS LINK.

The third version of this program (**SnnnRDR**) is used with **SnnnINQ** to demonstrate the use of SET rather than INTO on READ commands.

PnnnTRG

This program is used to demonstrate Transient Data Queue READ operations.

PnnnUPD

This program is used to demonstrate temporary storage queue processing as a means of maintaining the state of processing within a transaction.

PnnnWTR

This program has the role of demonstrating both WRITE and REWRITE commands.

Appendix A - Sample Application

Non-CICS demonstrations

There are skeleton versions of programs **LnnnINQ** and **LnnnRDR** specifically included to demonstrate the use of a standard COBOL / PLI CALL rather than an EXEC CICS LINK.

COBOL / PLI CALL is cited as being more efficient than EXEC CICS LINK. Whilst a static Link-Edit of main program (**LnnnINQ**) and sub-routine (**LnnnRDR**) was used originally, the set up now uses Dynamic Linkage to be closer the the EXEC CICS LINK environment in terms of reduced storage occupancy.

The L prefix is simply used to differentiate source versions the program should be Link-Edited or renamed as their P prefix counterparts.

Appendix A - Sample Application

Key to Program Prefixes

Lead Character	Significance
C	Program uses Channels and Containers
D	Reserved for DB2 version of the application
H	HTML DOCUMENT equivalents of maps.
L	Program uses COBOL CALL
M	All BMS Maps
P	The main application programs, using INTO rather than SET, all others are alternatives to demonstrate alternative concepts.
R	Members containing programs which use the RRDS files, (the actual programs remain as P series for simplicity of installation.).
S	Programs using SET rather than INTO
W	WEB Aware version of the application
X	Members containing programs which exploit START (the actual programs remain as P series for simplicity of installation.).

Appendix A - Sample Application

CICS / DB2 Environment

Jobs are provided to create and populate a DB2 Table using the same data that is loaded into the VSAM KSDS file. It is assumed that a database exists and conforms to the naming convention of; &SYSUID.D, where nnn are the last three digits of the Userid.

DB2CONN Resource

There is a single DB2CONN resource defined within CICS, which is configured to connect to DB2 automatically if the DB2 sub-system is operational.

The AUTHID and SIGNID on the DB2CONN are set to that of the tutor Userid.

DB2ENTRY Resource

There is a separate DB2ENTRY resource for each student. This is a deliberate policy so that if a student locks the Entry Threads they only impact themselves and no other students. The DB2ENTRY naming convention is &SYSUID.E, where nnn are the last three digits of the Userid.

The AUTHID on each DB2ENTRY is the Userid associated with its use.

DB2TRAN Resource

There are five DB2TRAN resources for each student. These are linked to the appropriate DB2ENTRY resources. The DB2TRAN resources are:

```
xxxxVAnn
xxxxVBnn
xxxxVDnn
xxxxVInn
xxxxVUnn
```

In each xxxx relates to the first four characters of the student’s Userid and nn equates to the last two digits of the transaction identity.

DB2 Table

The table used in the DB2/SQL Exercises can be created and populated using member BUILDEMP in &SYSUID..CICS.COB.

This is the end of “Appendix A - Sample Application” description.