CA Telon® Application Generator

Datacom Database SQL Option Guide

r5.1

CA Product References

This document references the following CA products:

- CA Telon® Application Generator (CA Telon)
- CA Datacom®
- CA Datacom®/DB
- CA IDMS™
- CA IDMS™ SQL

Contact CA

Contact Technical Support

For your convenience, CA provides one site where you can access the information you need for your Home Office, Small Business, and Enterprise CA products. At http://ca.com/support, you can access the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Provide Feedback

If you have comments or questions about CA product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA product documentation, complete our short customer survey, which is also available on the CA Support website, found at http://ca.com/docs.

Contents

Chapter 1: Introduction

This guide describes how to use CA Datacom Database SQL Option with CA Telon. In this guide, CA Telon Application Generator is referred to as CA Telon.

This section contains the following topics:

Overview

This chapter contains installation and operational instructions for the CA Telon CA Datacom Database SQL Option.

Audience

To use this guide, you should be a skilled user of CA Datacom Database SQL Option and use this guide in conjunction with the CA Telon system documentation.

Where to Find Additional CA Datacom SQL Information

You can find additional information about databases and CA Datacom SQL in the following guides:

- *CA Datacom/DB Database and System Administration Guide*.
- *CA Datacom/DB SQL Users Guide*

Product Installation

Install the CA Telon Base Product using the procedures described in the *Installation Guide* manual and include the password for the CA Datacom Database SQL Option. If the base installation has already been done, repeat the installation and select only the CA Datacom Database SQL Option.

CA Datacom Database SQL Option Customization

The procedure described in the section, "Customize INSTALL Dataset Names," of the *Installation Guide*, should be used with the contents of the CA Datacom Database SQL Option INSTALL PDS to customize CA Telon for your site.

Creating Job Streams and PROC JCL

The following table lists INSTALL members used to create job streams and PROC JCL.

Member	Label	Activity
$JOB		Job stream JCL to execute PROC
$010		JCL procedure parameters
$020	SCRATCH1	Scratch step
$030	IEBUP1	Obtain CA Telon source and update maclib
$040	CONTROL	Create screen definition
$050	GEN	Assembly step; to generate COBOL or PL/I source code
$060	IEBUPSRC	Store temporary file
$070	PRINT	Print panel image
$080	RESOLVE	Resolve COPY statement
$090	DB2PC	DB2 precompiler step (DB2 only)
$100	ECP	ECP compiler step (CICS only)
$110	DCOMPRE	CA Datacom Database SQL Option precompiler step (CA Datacom only)
$120	ISQLDML	CA IDMS SQL precompiler step[1]
$130	COB/PLI	COBOL compiler or PL/I editor step
$140	LKED	Linkage editor step
$150	BCF	CA IDMS SQL access module[1]
$160	DBRMSRC	Capture and print related source
$280	RESOLVE2	Resolve COPY statement[2]
$290	DB2PC2	DB2 precompiler step (DB2 CICS client only)

Member	Label	Activity
$300	IDMSDML2	CA IDMS precompiler step (CA IDMS CICS client only)
$310	ECP2	ECP compiler step[2]
$320	DCOMPRE2	CA Datacom Database SQL Option precompiler step (CA Datacom CICS client only)
$330	ISQLDML2	CA IDMS SQL precompiler step[3]
$340	COB2	COBOL compiler step[2]
$350	LKED2	Linkage editor step[2]
$360	BCF2	CA IDMS SQL access module step[3]
$370	DBRMSRC2	Capture and print related source[2]

[1]CA IDMS/SQL only
[2]CICS client only
[3]CA IDMS/SQL CICS only

#CUSTCTL Parameters

The following table contains a brief description of the parameters added to #CUSTCTL for the CA Datacom Database SQL Option. You can find descriptions of other parameters in the *Installation Guide* manual.

@ name	Change to
@DCSLIB	CA Datacom Database SQL Option MUF LOADLIB
@DCBLIB	CA Datacom Database SQL Option BASE LOADLIB
@DCCA90	Unicenter LOADLIB
@DCOBJ	CA Datacom Database SQL Option OBJECT Library
@DCOMINCL	CA Datacom Database SQL Option INCLUDE Library
@DATACOM	'Y' to request the CA Datacom Database SQL Option

Other INSTALL Members

The table below lists and describes the other members in INSTALL. These members are used only by the CA Datacom Database SQL Option, except TRGDATA and TRGHELPM. The chapters in this document containing instructions on how to use these members are also listed.

Member	Function	Chapter
CTRDCMLD	Load training data into a VSAM file and then into CA Datacom Database SQL Option tables, load help for training programs into a VSAM file	Model Training System
CTRDCMTB	Define the training tables to CA Datacom Database SQL Option	Model Training System
SQLEXTDC	Execution JCL for the CA Datacom Database SQL Option extract	CA Datacom/DB Datadictionary Extract
TRGDATA	Training data	Model Training System
TRGHELPM	Training help	Model Training System

Chapter 2: CA Datacom/DB Datadictionary Extract

This chapter describes the CA Datacom Datadictionary Extract Utility system used to extract table information from the mainframe and create a Transport file to be loaded into the TDF. This system also provides the ability to create COPY members from the tables selected.

The CA Datacom Datadictionary Extract Utility is a batch process that allows you to extract SQL tables and views from a CA Datacom Datadictionary. The utility outputs the extracted table and column information in transport file format for processing by the Transport_In utility to populate the CA Telon Design Facility Data Administration.

For each extracted table, the utility also supports automatic creation of a copybook member containing COBOL descriptions of all columns in the table. You can name this copybook as a COPY member in CA Telon programs.

CA Datacom Datadictionary Extract Utility processing is based on information specified on one or more control cards in an input file as described in Control Cards. The utility produces a report providing a history of the control card processing.

This section contains the following topics:

System Components

The components for this system are:

- Mainframe JCL member SQLEXTDC in the #PDSQUAL.INSTALL dataset
- From the #PDSQUAL.LOAD dataset, load members:
  - SQBPM100
  - SQBPM200
  - SQBPM300

System Usage

The JCL member SQLEXTDC is used to specify the tables to be extracted. The user is required to supply information concerning COPY members, as well as TDF data administration parameters such as TLNNAME.

Driver (SQBPM100)

This program lets you specify which tables are extracted from the CA Datacom Datadictionary.

The batch driver takes a series of input transactions specifying the selected tables, and optional information concerning copy members.

Once the program processes the input information, the SQBPM200 program is called to build the transport file and SQBPM300 is called to build the copy members. The file SQINTRAN contains a series of 80-character input transactions, that serve as input to the SQBPM100 program.

Build Transport (SQBPM200)

SQBPM100 calls this batch subroutine. The subroutine builds a transport file for all selected tables and produces a report summarizing the results. For more information about selecting tables for extraction, see Table Selection (see page 15).

For each table, if a COPY or DCLCOPY member is to be created, a call is made to the SQBPM300 program.

Build COPY and DCLCOPY Members (SQBPM300)

SQBPM200 calls this batch subroutine to produce COPY and DCLCOPY members. The output sequential file for these members is produced containing IEBUPDT ./ ADD NAME= specifications that can be used to load a PDS on the mainframe with individual members. For more information about requesting COPY and DCLCOPY members, see Copy Member (see page 16).

Control Cards

The control cards connect the batch job to the datadictionary, specify which tables to extract and if COPY or DCLCOPY members are created.

- These control cards connect the batch job to the datadictionary:
  - User Identification (see page 13)
  - User Password (see page 14)
  - Datadictionary Name (see page 14)
- Table Selection (see page 15) specifies the CA Datacom/SQL tables
- Copy Member (see page 16) specifies the COPY and DCLCOPY members

This is a sample of the control cards required to extract a table and create a copybook member:

```
        1         2         3         4         5         6         7    8
12345678901234567890123456789012345678901234567901234567890123456789012345678890...0

USERID   DATACOM-INSTALL
PASSWORD NEWUSER
DBNAME   CASQLDEFAULT
TABLE   EMPDEMO            EMPLOYEE                          DEMOEMP  Y
TABLECPY EMPLCOPY DEMO-EMPL-TABLE              Y
```

User Identification

This control card identifies the user.

Pos	Len	Description
1	8	Control card type—Specify the literal USERID to identify the statement.
10	15	USERID—Specify the User ID of the user creating the extracted transport file. The user must be known by the CA Datacom Datadictionary.

User Verification Example

This statement indicates you want to identify DATACOM-INSTALL as the user creating the extract.

```
        1         2         3         4         5         6         7    8
12345678901234567890123456789012345678901234567901234567890123456789012345678890...0

USERID   DATACOM-INSTALL
```

User Password

This control card identifies the user password.

Pos	Len	Description
1	8	Control card type—Specify the literal PASSWORD to identify the statement.
10	15	Password—Specify the password for the user identified by the USERID control card.

User Password Example

```
        1         2         3         4         5         6         7    8
12345678901234567890123456789012345678901234567901234567890123456789012345678890...0

PASSWORD NEWUSER
```

Datadictionary Name

This is an optional control card identifies the name of the datadictionary from which the table definitions are extracted.

Pos	Len	Description
1	8	Control card type—Specify the literal DBNAME to identify the control statement.
10	32	Datadictionary name containing the definitions.

Datadictionary Example

```
        1         2         3         4         5         6         7    8
12345678901234567890123456789012345678901234567901234567890123456789012345678901...0

DBNAME    CASQLDEFAULT
```

Table Selection

This control card identifies the CA Datacom SQL tables or view you want to extract. You can specify up to 20 separate table selections.

Pos	Len	Description
1	8	CONTROL CARD TYPE—Specify the literal TABLE to identify a Table Selection statement.
9	18	SCHEMA—Specify the name of the SQL Schema for the table. This field is supplied **only** for tables created through SQL. This field must be blank for tables created by any other method which are to be extracted (for example, VSAM Transparency).
28	32	TABLE—Specify the name of the SQL table or view to be extracted. This field cannot be blank.
61	8	TLNNAME[1]—Specify the eight-byte CA Telon TLNNAME for the table. If omitted, defaults to the first eight bytes of the table name specified in the SCHEMA parameter. The TLNNAME must be unique for all tables in the TDF.
70	1	COPY[1]—Specify a "Y" in this field to produce a copybook member containing COBOL definitions for the table's columns.
71	1	DCLCOPY[1]—Specify a "Y" in this field to produce a DCL copybook member containing COBOL definitions of the table's columns. **Note:** A DCL copy member has an "01" level label produced as "01 DCL<tablename>." The name of the DCL copy member is the same as the TLNNAME.

[1]Optional

Table Selection Examples

The following statement requests extraction of the CA Datacom table EMPDEMO.EMPLOYEE. The table definition is to be transported into the TDF with a qualifier of DEMOEMP and a TLNNAME of DEMOEMPL.

```
        1         2         3         4         5         6         7    8
12345678901234567890123456789012345678901234567901234567890123456789...0

TABLE    EMPDEMO    EMPLOYEE       DEMOEMP Y
```

Copy Member

If you have specified a "Y" in the COPY parameter of a TABLE statement, you can supply this control card immediately following the TABLE statement to provide additional information about the copy member for the table.

Pos	Len	Description
1	8	CONTROL CARD TYPE—Specify the literal TABLECPY to identify a Copy Member statement.
10	8	MEMBER (Optional)—Specify the name of the copybook member for the generated ./ ADD NAME= statement. If omitted, defaults to the first eight bytes of the table name specified in the TABLE parameter of the Table Selection statement.
19	30	LABEL[1]—Specify the COBOL name of the group definition generated for the table. If omitted, no group level definition is produced in the copybook.
50	1	LEVEL 1[1]—Specify a "Y" in this field, if LABEL parameter has been specified and you want the group definition to contain a level of 01. If omitted, and LABEL has been specified, the group definition contains a level of 03.

[1]Optional

Copy Member Example

The following example produces an ./ADD NAME=EMPLCOPY statement for the copybook and generates a group statement "01 DEMO-EMPL-TABLE." before the elementary COBOL definitions for the columns in the selected table:

```
        1         2         3         4         5         6         7    8
12345678901234567890123456789012345678901234567901234567890123456789012345678890...0

TABLECPY    EMPLCOPY    DEMO-EMPL-TABLE                Y
```

Chapter 3: Loading the TDF with Programs and Tables

CA Datacom SQL tables created by the CA Datacom Datadictionary Extract must be loaded into the TDF using the Transport utility. In addition, there can be existing programs, in CA Telon source code format, that need to be imported into the TDF. It is important to have agreement between the data administration specification in programs to be imported and the data administration definitions contained within the TDF.

Transport

The Transport files created by the SQL Catalog Extract are used to load the TDF data administration files with CA Datacom SQL table information. Note that no provision is provided to modify CA Datacom SQL table information within the TDF, other than the ability to add TLNROW's that make use of column information contained within existing tables.

For the CA Telon Training system, a Transport file is provided to pre-load the TDF prior to the importing of any Training system programs. This file is member CTRDCMTP found within the SOURCE dataset.

Import

It is important to consider what data administration is already contained within the TDF prior to importing programs. Especially in the case of CA Datacom SQL, there can be differences in the area of TLNROW specification between what is contained in the TDF, and what might be in a particular program. Ensure the TDF data administration contains all TLNROW's that might be applied for a particular table before importing programs that specify those TLNROW's.

By use of the CTRDCMTP member described above, the user can import all of the CA Telon Training system programs with no problems.

Chapter 4: Program Generation

There are a number of issues to consider when generating programs for CA Datacom SQL; for example, SQL return codes.

This section contains the following topics:

SQL Return Codes

The logic generated relative to SQL data access for CA Datacom SQL is handled in this manner: The SQLCODE is loaded into the <tablename>-STATUS

The generic DA-STATUS code is loaded as follows:

DA-STATUS	Codes
OK	0
NOTFOUND	+100
DUPLICATE	-263
LOGICERR	-135
SECURITY	-802 or -803
NOTAVAIL	+125
DBMERROR	(any other return code)

CA Datacom Option Features

CA Telon for CA Datacom SQL fully supports:

- 32-byte column names
- Creation and maintenance of CA Telon join tables
- ALIAS specification for tables and joins
- COPY and DLCOPY specifications for table definitions

The DCLCOPY member is used as the include member in the following generated statement for CA Datacom SQL CA Telon programs:

```
EXEC SQL INCLUDE <tlnname>
     END-EXEC
```

If the user does not specify the TLNNAME in the SQL Catalog Extract for CA Datacom, the first eight bytes of the table name are used.

Note: The COPY member is typically used if a second COBOL layout of columns for a table is needed. This member is copied into the SEGMENT-IO-AREA of the generated program.

Chapter 5: Model Training System

The model training system can be used for testing, training and demonstration of system operation accompanies the CA Datacom SQL installation. The CA Datacom SQL feature does not require the implementation of this training system, but the manner in which it is constructed and executed is educational. This section provides a description of the system components and instructions for operation.

System Overview

The model training system provides a series of screen and batch programs to operate upon a data structure related to employees within a company. These are the components of the data structure:

- Employee data—Information concerning the employee (name, address, telephone, etc.).
- Task/project data—Information concerning tasks and projects to which the employee is assigned.
- Time data—Information concerning time allotments by task/project for the employee.

This data is placed in several different files and tables for use by the supplied programs. The contents of each file and table are described below:

Object name	Object type	Contents
TREMPLB	INDEXED	Records for all three data components, with a flag for each record indicating which component is used
TRGEMPLC	CA Datacom SQL	Rows for employee data
TRGTASKC	CA Datacom SQL	Rows for task/project data
TRGTIMEC	CA Datacom SQL	Rows for time data
TRHELP	INDEXED	Help data for application

For file and table layouts of each of these objects, see File Layouts (see page 30).

Programs

The programs supplied perform a variety of functions relative to these objects in both online and batch modes of operation.

Naming Convention

The CA Telon training system uses this program naming convention:

Position	Description	Values
1-2	Header	TR
3-4	Program type	
	■ Screen	■ CP
	■ Batch	■ BP
5-8	Program id	
	■ Screen	■ CCM/[1]
	■ Batch	■ M/[1]00

[1]The / code values are described in the Screen Programs and Batch Programs.

Therefore, CA Telon source with the name TRCCMASD produces a COBOL program named TRCPCCMA, while CA Telon source with the name TRM500BD produces a COBOL program named TRBPM500.

Screen Programs

The screen programs are supplied with ID's of the form CCM*i*. The *i* code within the ID represents these programs:

i Code	Data	Program Description
A	Employee	Add/update screen. Allows the user to add new employees, or update existing employee information.
C	Employee	Combination screen—add, update, delete, inquiry. Allows the user to preform all maintenance functions for the employee data within a single program.
D	Employee	Display screen. Displays employee data.
E	None	Exit screen. Application termination screen.
H	HELP	HELP screen. Provides field and screen level HELP.

i Code	Data	Program Description
L	Employee	List screen. Lists employee data with row incrementation.
M	Employee	Menu screen. Application starting point.
N	Employee	Non-terminal. CICS non-terminal program providing report of employee data.
P	Task/ project	Task/project screen. Maintenance for the task/project data.
T	Time	Time screen. Maintenance for the time data.
X	None.	Screen program which initiates the **N** non-terminal program.
Z	Employee	Zap screen. Deletes employee records.

For layout of TRHELP, see File Layouts (see page 30).

Batch Programs

There are four batch programs of the form M*i*00.

i Code	Program Description	Input Data	Output Data
M000	Creates tables in the CA Datacom SQL database	TRGEMPLC TRGTASKC TRGTIMEC	
M100	Create SQL data. After deleting contents, takes indexed TRGEMPLB file and reloads CA Datacom SQL tables[1]: ■ TRGEMPLC ■ TRGTASKC ■ TRGTIMEC	TRGEMPLB	TRGEMPLC TRGTASKC TRGTIMEC
M500	Create TRGEMPLB data. Takes sequential data contained in file TRGDATA and reloads the TRGEMPLB indexed file[1]	TRGDATA	TRGEMPLB
M700	Create TRHELP data. Takes sequential data contained in file TRGHELP and reloads the TRHELP indexed file[1]	TRGHELP	TRHELP

[1]also provides a report of the processing

To refresh the model training databases, including reloading the HELP and CA Datacom SQL data, submit these job streams in order:

- M500
- M100
- M700

Building the Model Training System

To use the model training system, the application programs must be compiled and linked, the training data must be loaded into the database, and the CICS tables must be updated.

Program Generation and Compilation

The SOURCECC dataset provides the source code for the model training system. These program definitions can be imported into the TDF to browse, modify, and generate the actual COBOL programs.

Online Programs— CA Datacom SQL Access

The INSTALL dataset provides generate, compile, and link JCL procedures and job streams. Job streams JMCPGCL and JMCXGCL, which invoke procedures TLMCPGCL and TLMCXGCL, are used to generate, compile, and link the training system's screen programs. The programs contain CA Datacom SQL access from either CA Telon source (JMCPGCL), or can be accessed directly from the TDF (JMCXGCL) after the programs have been imported.

Batch Programs—CA Datacom SQL Access

Similarly, job streams JMBPGCL and JMBXGCL, which invoke procedures TLMBPGCL and TLMBXGCL, are used to generate, compile, and link the training system's batch programs containing CA Datacom SQL statements from either CA Telon source (JMBPGCL), or can be accessed directly from the TDF (JMBXGCL) after the programs have been imported.

Programs—No CA Datacom SQL Access

Job streams JNBPGCL, JNBXGCL, JNCPGCL, and JNCXGCL and their respective procedures, provide similar functionality to Online Programs—CA Datacom SQL Access and Batch Programs—CA Datacom SQL Access except without the CA Datacom SQL access processing.

Summary of Programs

This is a summary of the jobs streams and procedures used with the training system:

Program ID	Job streams	Procedures	Comments
CCMA	JMCPGCL, JMCXGCL	TLMCPGCL, TLMCXGCL	Add/update screen
CCMC	JMCPGCL, JMCXGCL	TLMCPGCL, TLMCXGCL	Combination screen
CCMD	JMCPGCL, JMCXGCL	TLMCPGCL, TLMCXGCL	Display screen
CCME	JNCPGCL, JNCXGCL	TLNCPGCL, TLNCXGCL	Exit screen
CCMH	JNCPGCL, JNCXGCL	TLNCPGCL, TLNCXGCL	HELP screen
CCML	JMCPGCL, JMCXGCL	TLMCPGCL, TLMCXGCL	List screen
CCMM	JMCPGCL, JMCXGCL	TLMCPGCL, TLMCXGCL	Menu screen
CCMN	JMCPGCL, JMCXGCL	TLMCPGCL, TLMCXGCL	Non-terminal program. Requires specifying parameters: ■ DEFTYPE=ND ■ OPTION=3 ■ ECPPARM='SP,SOURCE'
CCMP	JMCPGCL, JMCXGCL	TLMCPGCL, TLMCXGCL	Task/project screen
CCMT	JMCPGCL, JMCXGCL	TLMCPGCL, TLMCXGCL	Time screen
CCMX	JNCPGCL, JNCXGCL	TLNCPGCL, TLNCXGCL	Screen program which initiates the **CCMN** non-terminal program
CCMZ	JMCPGCL, JMCXGCL	TLMCPGCL, TLMCXGCL	Zap screen
M000	JMBPGCL, JMBXGCL	TLMBPGCL, TLMBXGCL	Create SQL tables batch program
M100	JMBPGCL, JMBXGCL	TLMBPGCL, TLMBXGCL	Create SQL data batch program

Program ID	Job streams	Procedures	Comments
M500	JNBPGCL, JNBXGCL	TLNBPGCL, TLNBXGCL	Create TRGEMPLB data batch program
M700	JNBPGCL, JNBXGCL	TLNBPGCL, TLNBXGCL	Create HELP data batch program

Note: Various COPY members are contained within the SOURCECC dataset, requiring CPYLIB be assigned to SOURCECC. The training data is located in members TRGDATA and TRGHELPM within the INSTALL dataset; for more information, see Loading of CICS Tables.

Loading of BMS Maps

The INSTALL member JCPBMS, which calls PROC TLNCPBMS, is used to assemble the BMS maps for the screen programs listed in Program Generation and Compilation (see page 26). It should be run for these program ID's:

CCMA CCMM
CCMC CCMP
CCMD CCMT
CCME CCMX
CCMH CCMZ
CCML

Loading of CICS Tables

The PCT and PPT entries for the training system are found in the TABLES dataset under the names TRPCTM and TRPPTM. These copy members must be added to existing table source; afterwards, the table source must be re-assembled.

Note: Define the TRHELP dataset to CICS as a VSAM file.

Execution of the Data Build Programs

Once the model training system programs TRM100BD, TRM000BD, TRM500BD, and TRM700BD have been generated, compiled and linked, and the CA Datacom SQL tables defined, follow these steps to populate the tables with data:

1. The source for the training data is found in #PDSQUAL.INSTALL members TRGDATA and TRGHELP. Both members must be copied into sequential datasets with their respective characteristics before the data is loaded.

Member	Name	Characteristics	
TRGDATA	#PDSQUAL.TRGDATA	organization record length record format blocksize	= PS = 60 = FB = n[1] X 60
TRGHELPM	#PDSQUAL.TRGHELP	organization record length record format blocksize	= PS = 44 = FB = n[1] X 44

[1]'n' the appropriate blocking factor for your site.

2. The actual load is done by #PDSQUAL.INSTALL(CTRDCMLD) which populates the CA Datacom tables and the VSAM HELP dataset.

Execution of the CICS Model Training System

After the files, maps, and programs have been defined to CICS, the transaction code TCMM can be entered to invoke the menu program.

Note: Each screen program has its own transaction code of the form TCM*i*, where *i* = A, C, D, E, H, L, M, P, T, X, or Z.

File Layouts

This section provides details about the following file layouts:

- TRGEMPLB
  - TRGEMPL Record Layout
  - TRGTASK Record Layout
  - TRGTIME Record Layout
- TRGEMPLC
- TRGTASKC
- TRGTIMEC
- TRHELP

TRGEMPLB

```
        04  EMPL-KEY-AREA.
            05  EMPL-KEY.
                10  EMPL-ID            PIC X(6).
            05  TASK-KEY.
                10 TASK-PROJ-ID        PIC X(4).
                10 TASK-TASK-ID        PIC X(4).
            05  TIME-KEY.
                10 TIME-YEAR           PIC 9(2).
        04  EMPLOYEE-AREA      PIC X(583).
      04  RECORD-TYPE-INDICATOR   PIC X.

*****************************************************************************
*    TYPE OF RECORD IS DETERMINED BY THE VALUE OF THE                       *
*    RECORD-TYPE-INDICATOR:     1 = TRGEMPL RECORD                          *
*                                   2 = TRGTASK RECORD                      *
*                                   3 = TRGTIME RECORD                      *
*****************************************************************************
********************** LENGTH OF EACH RECORD IS 600 BYTES ************** *
```

TRGEMPL Record Layout

```
*****************************************************************************
*    TRGEMPL RECORD LAYOUT                                                  *
*****************************************************************************
        04 TRGEMPL-AREA REDEFINES EMPLOYEE-AREA.
            05 EMPL-NAME               PIC X(25).
            05 EMPL-DOB                PIC 9(6).
            05 EMPL-SEX                PIC X.
            05 EMPL-PHONE              PIC X(10).
            05 EMPL-STREET             PIC X(25).
            05 EMPL-CITY               PIC X(25).
            05 EMPL-STATE              PIC XX.
            05 EMPL-ZIP                PIC X(5).
            05 EMPL-DOE                PIC 9(6).
            05 EMPL-DEPARTMENT PIC XXX.
            05 EMPL-HOURLY-RATE        PIC 999V99 COMP-3.
            05 EMPL-HOURS              PIC 999V9.
            05 FILLER          PIC X(468).
*****************************************************************************
*    END OF TRGEMPL RECORD LAYOUT                                           *
*****************************************************************************
```

TRGTASK Record Layout

```
*****************************************************************
*    TRGTASK RECORD LAYOUT                                      *
*****************************************************************
        04 TRGTASK-AREA REDEFINES EMPLOYEE-AREA.
            05 TASK-PROJ-STUFF.
                10 TASK-P-DESC         PIC X(22).
                10 TASK-P-PRTY PIC X(2).
                10 TASK-P-CODE PIC X(6).
            05 TASK-TASK-STUFF.
                10 TASK-T-DESC PIC X(22).
                10 TASK-T-PRTY PIC X(2).
                10 TASK-T-CODE PIC X(6).
            05 FILLER          PIC X(523).
*****************************************************************
*    END OF TRGTASK RECORD LAYOUT                               *
*****************************************************************
```

TRGTIME Record Layout

```
*****************************************************************
*    TRGTIME RECORD LAYOUT                                      *
*****************************************************************
        04 TRGTIME-AREA REDEFINES EMPLOYEE-AREA.
            05 TIME-FOR-BATCH.
               10 TIME-BY-QUARTER      OCCURS 4  TIMES.
                  15 TIME-WEEKLY       OCCURS 13 TIMES.
                     20 TIME-REG       PIC S9(3)V9 COMP-3.
                     20 TIME-OT        PIC S9(3)V9 COMP-3.
               10 TIME-TOTALS  OCCURS 4 TIMES.
                  15 TIME-REG-QUARTERLY-TOTAL
      PIC S9(5)V9 COMP-3.
            05 FILLER          PIC X(255).
*****************************************************************
*    END OF TRGTIME RECORD LAYOUT                               *
*****************************************************************
```

TRGEMPLC

```
01  DCLTRGEMPLC.
    05  EMPL-ID               PIC X(6).
    05  EMPL-NAME             PIC X(25).
    05  EMPL-DOB                      PIC S9(6).
    05  EMPL-SEX                      PIC X(1).
    05  EMPL-PHONE            PIC X(10).
    05  EMPL-STREET           PIC X(25).
    05  EMPL-CITY                     PIC X(25).
    05  EMPL-STATE            PIC X(2).
    05  EMPL-ZIP                      PIC X(5).
    05  EMPL-DOE                      PIC S9(6).
    05  EMPL-DEPARTMENT       PIC X(3).
    05  EMPL-HOURLY-RATE              PIC S9(3)V9(2).
    05  EMPL-HOURS            PIC S9(3)V9(1).
```

TRGTASKC

```
01  DCLTRGTASKC.
    05  TASK-EMPLID            PIC X(6).
    05  TASK-PROJID            PIC X(4).
    05  TASK-TASKID            PIC X(4).
    05  TASK-P-DESC            PIC X(22).
    05  TASK-P-PRTY            PIC X(2).
    05  TASK-P-CODE            PIC X(6).
    05  TASK-T-DESC            PIC X(22).
    05  TASK-T-PRTY            PIC X(2).
    05  TASK-T-CODE            PIC X(6).
```

TRGTIMEC

```
01  DCLTRGTIMEC.
    05  TIME-EMPL-ID           PIC X(6).
    05  TIME-PROJ-ID           PIC X(4).
    05  TIME-TASK-ID           PIC X(4).
    05  TIME-YEAR                      PIC S9(2).
    05  TIME-C-1-26            PIC X(156).
    05  TIME-C-27-52           PIC X(156).
    05  TIME-Q-TOT1            PIC S9(5)V9(1).
    05  TIME-Q-TOT2            PIC S9(5)V9(1).
    05  TIME-Q-TOT3            PIC S9(5)V9(1).
    05  TIME-Q-TOT4            PIC S9(5)V9(1).
```

TRHELP

```
03  TRHELP-AREA                PIC X(1058).
 03  FILLER REDEFINES TRHELP-AREA.
       05  HELP-KEY             PIC X(8).
       05  HELP-MSG.
           10  HELP-MSG-LINE    OCCURS 15 TIMES
                 INDEXED BY HDX PIC X(70).
```

Index