PRESQL2

A program to process SQL type syntax to be used with MySQL and Mariadb (or others via ODBC) for input to the GnuCOBOL compiler along with other support programs and files.

Set up and Usage Manual

Applewood Computers
17 Stag Green Avenue
Hatfield
Hertfordshire
AL9 5EB
United Kingdom

Phone: 44-(0)1707-263712

Support email: vbcoen@gmail.com

Author:	Vincent Bryan Coen.
Document update:	3 August 2022
Revision:	38
Software Version Date:	9 May, 2019.
Software Version	2.223. (presql2)
Word Processor:	LibreOffice 7.3 or later

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Contents

Pre Processor for MySQL Tools & Documentation.

The following tools originating from Currey Adkins (Jim Currey) but somewhat modified by myself, will allow preprocessing of MySQL type commands into the source of a Cobol program prior to running the GnuCOBOL compiler, in addition, other tools that act as support to this function are also included.

All references to MySQL also include Mariadb, a drop in replacement (usage wise).

By standard convention the input file name should have extension name of .scb and the output file name should be .cbl. Note that scb stands for s = SQL source, cb = cobol.

This package does NOT use EXEC statements but a variation of the theme as extra commands are injected in to the source pack. See the Syntax on page 17 for full details.
At some point in the future and if others request, I will add in direct support for EXEC SQL syntax that will support both fixed and free source formats.

As supplied, the tools will accept a source file in free or fixed format.

The primary benefit of using this sub system is that the pre processor accepts both free and fixed format program sources and provides additional tools for creating copy books (see bldcopy2) used within a SQL based program along with prtschema2 that will print out the entire table structure of a given database. See page 12 for more information.

By using this software you are, in law accepting the GPL terms and conditions as given starting on page 25 and on page 30 of this manual. You should therefore read all before proceeding further.

Program and Files Listing

cobmysqlapi39.c	Source of Mysql client library that needs to be included for each compile. This originally came from dbpre and includes many extra functions and changes made by myself. ***Use this or v38 versions only***.
*presql2.cbl	Source for the pre SQL processor. May be additionally supplied as presql2o and presql2M – for Mariadb (M) and Mysql (o) versions - Yes there are differences and stored in the Variations directory – see below for more information.
*bldcopy2.cbl	Source for tool to create a copy book taken from a database table.
*prtschema2.scb	Source fed to presql2 of the DB print schema program. May be additionally supplied as prtschema2o & M in the Variations directory. For normal usage, programs in the Variations directory should not be needed.
prei.scb	Source demonstrating many of the sql syntax in a menu, based on SQL so do not run against presql2 as it will not find the test DB it uses.
preo.cbl	As above but o/p (output) from an early version of the presql processor.
Test1.cbl	Source that will show various Mysql system information.
Presql2.pdf	This file.
Copying.pdf	GNU General Public License
Documentation_License.pdf	Documentation Creative Common License.

Note that these following four copy books should be copied and modified to work for your

applications and site requirements and are as used but modified by myself for the open source ACAS application package which are in the included ACAS directory. They are slightly different to as used for the presql2 package, see mysql-procedures.cpy.

+mysql-procedures.cpy	Cobol procedures to be included in to target source containing called routines introduced by the presql2 process. Free format. They also include additional code not currently used by ACAS for locked tables or records as not thought to be needed, may be !
+MYSQL-PROCEDURES.CPY	Likewise but for fixed format.
+mysql-variables.cpy	Cobol variables for WS to be included into target source needed by code introduced by the presql2 process, Free format.
+MYSQL-VARIABLES.CPY	Likewise but for fixed format.

The fixed format versions are not as up to date as the free one's which are used in the ACAS code..
Updated versions of these are also in the ACAS directory modified for the ACAS applications available via www.sourceforge.net under the name ACAS but note this version has not yet been completed but nightly source builds are available via my website at:
http://www.applewood.linkpc.net/files/acas/nightlybuilds/.

The following are bash scripts to compile the primary programs.

cobmysqlapi39.sh	Compiles cobmysql39.c using gcc with o/p named as cobmysqlapi.o
bldcopy2.sh	Compiles bldcopy2 to executable.
presql2.sh	Compiles presql2.cbl to executable.
prtschema2.sh	Compiles prtschema2.scb to executable (must run presql2 first to create prtschema2.cbl before using the GnuCOBOL compiler).

Within the Variations directory are program names ending with 2o or 2M that have a call to read a param file where presql2 uses presql2M.param or presql2o.param, etc.
These files must be changed to reflect your set up, i.e., location of the sql server, user and password names etc. These were used in special cases, although recent changes have made them somewhat redundant.

The scripts have also been updated to reflect this change. These extra programs taken from the original supplied and modified sources, allow me to use both MySQL and MariaDB servers running on my development system (MariaDB as distro standard) and MySQL on a OSX based computer. These have slight differences in the system tables which for normal programming will not be seen but as one of the supplied tools access a specific system table which do have differences, these extra tools have been created from the original source - see later in this manual for more information.

*When running presql2 (for processing with both MySQL and MariaDB) it creates three fields that are very large (i.e., 432Mb or higher) and the compiler objects to them, so I have a UDF REPLACE statement at the beginning of the source file to change it from x(94967295) to x(1048576). It is possible at some point in the future, that the first field size gets changed to yet another value in which case, using a text editor change it in presql2.scb before compiling. You will see the reported error when the compiler runs. This is the only program with the issue. The value used to replace it is 1024 * 1024 which hopefully is big enough but in any event is not used within ACAS. This is a reasonable compiler limitation otherwise you would have very large FD records that could exceed installed Ram :)*

Your Cobol source files will NOT be affected by this problem (unless your programs actually uses the tables within information_schema which for any normal work is not used) only for the specific tool, prtschema2 created by presql2 and once created it should not need to be recompiled .

Clearly the sizing defaults for specific field types within the MySQL and Mariadb systems for the (current) versions I am using are large, so you could change the defaults to use reduced sizes to get totally around the problem.

There are other differences between the two, within information_schema which is why I have two versions for each program that will be affected.

For users sticking to usage of only one SQL server the problems should be reduced but be aware of this when compiling prtschema2 at any time. If it is a problem the compiler will soon tell you :)

Timing note:
On my systems when running prtschema2 against mariaDB (on a Mageia v8 distro using a AMD FX8350 8 core cpu and 16Gb ram) where MariaDB is the only one offered, it takes 15 seconds to run and when run against MySQL running on a Mac Pro (2008 model with 2 Xeon quad core cpu's & 24Gb ram) it is less than 1 second. Both SQL servers are at the basic configuration as out of the box.
For production this might well be changed :)

Notes for above:
** Programs starting with '*' must be changed to reflect your MySQL installation such as user, password, socket location before compiling them. Look at the files with .param extensions for these values and change accordingly. However if using the programs in the Variations directory then program changes might also be required.*
+ Likewise you also might need to change these, again to reflect your installation and applications
However if using the program names ending with 2 or 2o or 2M they will use a parameter file see above for more information that should remove the need for any code changes to the programs presql2, prtschema2 & bldcopy2.
HOWEVER these param files must be in the current working directory. All comments regarding RDB for short should mean RDBMS.

In addition, I have included the Cobol + SQL source files for one of the ACAS FH (File Handlers) – Cobol file processing with call to the DAL (Data Access Layer that only processes the RDB table) subject to a system parameter and the corresponding DAL module that is called by any Cobol program requiring access to a specific RDB table (or in the case of acas011 a Cobol flat file). along with a data load program for the same table / Cobol flat file and these are in the ACAS directory.

Application programs generally will call acas011 (amongst others) which handles Cobol file processing or it will call stockMT for table handling. There is one FH and DAL for each file and matching table. This application (ACAS) is available via SF if wanted, but the RDB version is still under development so it is not yet been uploaded – hopefully later this year however a nightly build of the sources is available via http://applewood.linkpc.net/files/acas/nightlybuilds/

The data for processing tables such as location of server, user name & password etc is held in the ACAS system parameter file and this can be seen being read by the stockLD program as against using the .param file method. This is so that the coding style can be migrated over to use ODBC, Postgres, Oracle or DB/2 etc, with minimum effort or that's the theory.

More information regarding these ACAS modules:

The input for file / table processing consists of two elements passed via linkage in File-Access that contains :

1. Two fields -

```
     03  File-Function   pic 99.
         88  fn-open            value 1.
         88  fn-close           value 2.
         88  fn-read-next       value 3.
         88  fn-read-indexed    value 4.
         88  fn-write           value 5.
         88  fn-Delete-All      value 6.    *> 10/10/16 - Delete all records.
         88  fn-re-write        value 7.
         88  fn-delete          value 8.
         88  fn-start           value 9.
         88  fn-Read-Next-Raw   value 13.   *> 14/11/16 - Special 4 LD.
         88  fn-Write-Raw       value 15.
         88  fn-Read-By-Name    value 31.   *> 15/01/17 for Salesled (SL160)
         88  fn-Read-By-Batch   value 32.   *> 08/02/17 for OTM3 (sl095)
         88  fn-Read-By-Cust    value 33.   *> 09/02/17 for OTM3 (sl110, 120, 190)
         88  fn-Read-Next-Header value 34.  *> 18/04/17 for Invoice (sl020, 50, 140)
*>
     03  Access-Type     pic 9.                *> For rdbms 2 should cover all !!!
         88  fn-input           value 1.
         88  fn-i-o             value 2.
         88  fn-output          value 3.
         88  fn-extend          value 4.       *> not valid for ISAM
         88  fn-equal-to        value 5.
         88  fn-less-than       value 6.
         88  fn-greater-than    value 7.
         88  fn-not-less-than   value 8.
         88  fn-not-greater-than value 9.      *> Not currently used (06/04/2012
```

and these are set for every access prior to the call to (in this instance) acas011 that in turn could call stockMT.

Also in File-Access are these fields (among others)

```
         07  FA-File-System-Used  pic 9.
             88  FA-FS-Cobol-Files-Used        value zero.
             88  FA-FS-RDBMS-Used              value 1.
             88  FA-FS-Valid-Options           values 0 thru 1.
     03  RDB-Data.
         05  DB-Schema   pic x(12)  value spaces.
         05  DB-UName    pic x(12)  value spaces.
         05  DB-UPass    pic x(12)  value spaces.
         05  DB-Host     pic x(32)  value spaces.
         05  DB-Socket   pic x(64)  value spaces.
         05  DB-Port     pic x(5)   value spaces.
```

FA-File-System-Used specifies what data access is to be used, i.e., Cobol files or tables and the content of RDB-Data has the data needed to connect to a specific MySQL server and comes from the applications system parameter file.
It could be via a call to read a param (eter) file, but this method is not in use as it allows any application user to see the access data and password although there are ways around this under Linux such as execute only file settings for the param file so cannot be read. Also for RDB migration at some point, as discussed earlier.

Original source used.

acas011.cbl	FH for Stock Cobol file processing.
stockMT.scb	DAL for Stock Table processing.
stockLD.cbl	Loading Cobol file to table data – used when migrating with example of calls to FH/DAL.
Fhlogger.cbl	Can be ignored, it is called by some of the above modules and used in testing to record all file & table processing.

Sources used as input after copy books processed with extensions changed to COB or SCB.

acas011.COB	As input to GnuCOBOL.
stockLD.COB	As input to GnuCOBOL.
fhlogger.COB	As input to GnuCOBOL.
stockMT.SCB	As input to presql2 using command presql2 stockMT.SCB stockMT.COB

Then output from presql2

stockMT.COB	As input to GnuCOBOL.

Hopefully I have provided enough information for you to make sense of the source code. These sources are still subject to change, as this version of ACAS is still in active development see above for more information about obtaining development code but the current version can always be found at : http://applewood.linkpc.net/files/acas/nightlybuilds/ as ACAS-Nightly.rar.

Set up

The following assumes you have installed Mysql or MariaDB along with the same version or later of the C client libraries (connector-c) and include sources as a current version compatible to the server you are using and have the RDB up and running, maybe using phpMyAdmin to verify operations or MySQL Workbench for the version of MySQL installed. A copy of the connector-c package is supplied with the ACAS system as used during development and testing. See details of nightly builds.

Of course you will also have installed the :

1. GCC development system
2. GnuCOBOL v3 or v4 system, compiled and installed it as needed. [No testing of earlier compilers has been made and therefore may not work out of the box.]
3. Extract the MySQL presql2 files to a directory that you can compile from or is linked.
4. Change the sources of mysql-* cpy (for free format) or MYSQL-*.CPY (for fixed) libraries to reflect your installation. You may well want to update the fixed ones to that of the free versions if being used. If you run to a directory containing Cobol copy books then copy these to it but do not change them for use by the presql2 tools unless really needed. Note that these copy books are for the presql tools and there are differences used with ACAS and for your Cobol applications.
5. Change the sources of *presql2.cbl, *prtschema.scb, bldcopy.cbl to reflect your installation.
   Here the only real changes needed are with the presql2.param, prtschema2.param and bldcopy2.param files and they have each the same content so can be copied over after changing the first one.
6. Using the included Bash scripts to compile all but start with the APIs - cobmysqlapi39.sh as cobmysqlapi.o is needed for the compile scripts for the three Cobol programs.
   Note that I use this version of the APIs as I want access to Rollback, Commit along with using SQL-STATE error numbers which is not in the earlier version and this is useful for getting error info when using INSERT and UPDATE which does not provide errno output.

If you have not been able to compile all without errors you will need to find and fix all issues otherwise you can now move the binaries to your ~/bin directory where they can be found.
You are now ready to create Cobol & SQL source programs using the SQL syntax, see page 17.

Use the various source programs included, to give you an idea of coding styles and formats etc such as prei.scb, stockMT.scb (& stockLD.cbl), bldcopy.cbl, prtschema.scb and for that matter presql2.cbl.

Note that stockMT. & stockLD has been run against MariaDB on a Mageia v6 X64 Linux installation , Mysql on a Mac Pro with OSX 10.15 and the *nix environment on a IBM m/f (mainframe).

If you wish to take a look at the ACASDB tables which is included within the ACAS directory, look at ACASDB.sql and that can be used to create the DB along with some rows used in basic unit testing for stock control and the other systems. This is the latest output from a backup made when I built the tools. Once installed in your mysql system you can compile the ACAS programs using the supplied scripts to help verify the presql2 tools.
Note that there might be some test data in tables STOCK-REC, SYSTEM-REC and others such as SLLEDGER and PULEDGER and you might want to modify the engine and character set used, see lines with

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;.
```

Also present in same directory is the sources in various guises (see ***More information regarding these ACAS modules***) above :

acas011.cbl	Stock F.H. (Cobol File Handler) that has simple call to
stockMT.scb	Stock DAL (Data Access Layer) that accesses the data base table.
stockMT.cbl	Stock DAL as output from presql2 and I/p to cobc.
stockLD.cbl	Load the STOCK-REC table from a Cobol flat file when migrating.
fhlogger.cbl	Can ignore, but called by some of the above modules to create logging data which is used for testing..

Also included are scripts for compiling these which will act as a test to confirm your MySql includes are present as well as the libraries that are included with the binaries for stockMT and some of the others.

I recommend you look closely at the code of stockMT.scb/ SCB as it should help you in working out the flow of SQL for MySQL used with presql2 for all of the access types you will use in your own programs.
See the directory ACAS and STOCKMT.SCB source for examples of usage of the syntax and the reason why most of the generated code does NOT end in a period (full stop).

A bit of background to the ACAS sources here :
ACAS was originally written to use standard Cobol file verbs, OPEN, READ, WRITE etc but in migrating over to use of RDB table processing it was understood that some ACAS sites would not want the extra complexity of installing and supporting a RDB system so both Cobol files or Tables are supported via the ACAS system parameter file which has fields to specify what is to be used along with the RDB system parameters required. ACAS users can be used by 1 or 2 users only for very small businesses or for medium one where three or more users access the ACAS system then RDB tables are used and the reason is that RDB has good table and record locking where Cobol flat files it is not so robust.

There are two modules for each Cobol file the F.H. (File Handler) name starting with acas0 and the linked one for the DAL as the name of the table then 'MT' so in the examples we have the F.H. acas011 and the matching DAL as stockMT. (The name acas011 derives from the file id of the file name in the copybook.)

So processing wise, the individual ACAS programs that require data calls acas011 for open, close, read, write, rewrite etc. The FH checks the system params to see what is used file or table and if file, processes the request otherwise if table calls stockMT for table processing.

So there is one acas0 FH for each Cobol file used and each one calls its own matching table DAL. In addition to the two modules accessing the data there is also a program that loads up separate tables from the Cobol files - see example stockLD.cbl and no, there is not a load file from table loader as it is not felt to be required - OK, no one has asked for them.

I hope the above is helpful in helping to understand the usage of the SQL pre-processor and its usage in real applications. The authors of it have been using it for many years in a range of applications.

The decision to use this MySQL / Mariadb SQL processor say over others, is that it provides more functionality, therefore making time for the programmer easier and shorter to implement the requirements of RDBMS usage, and in the case of most, supports both free and fixed cobol source formatting.

Performance considerations

This is very dependent on the set up of the RDB server but out of the box installations has shown no noticeable issues other as discussed earlier that under Mariadb running prtschema2 which takes 15 seconds to list out (without printing) the test database (ACASDB with 25+ tables and one table having a large number of columns, e.g., 160). Have also run prtschema2 against Mysql running on OSX or the m/f which shows a performance improvement over mariadb when running prtschema2 as Mysql runs in under a second.

Running the ACAS applications during unit testing has not shown any delays but there is not a large amount of data records held at the moment that would help to show up such issues.
One area of possible concern is that there is no provision for prepared statements and this will not help when processing a large amount of data say in running a billing cycle batch job or again a large printing process but batch processes do not normally have a user watching while it runs.

Prepared statements and triggers code can be added by yourself if it is needed and if so let us know of your code changes so we can add it in to the presql2 system.

There there are other issues that might slow down the job step/processes such as :

1. Speed of the Dasd (Direct Access Storage Device – a hard drive) in data throughput that is holding the RDB.
2. Dasd partition type.
3. Dasd spare capacity – free space should be more than enough say 25% or better at any time.
4. Fragmentation of large tables but **not** if using SSD.
5.. Spread of data across more than one Dasd.
6. Frequency of clean up - removing dead rows etc from the server and any other re-organisation. Also see (4).
7. Usage of RDB parallel processing including back ups in real time when using more than one server.
8. Speed of the system used to hold the RDB including that of the connection to the LAN and the LAN itself. 1Gb is better than 100Mb. Large sites run =>10Gb LANs
9. If the applications and the RDB is running in 32 as against 64 bit native mode.
10. Spare resources available on the server and for that matter the system any application is running on such as extra CPU's and Ram.
11. The platform in use, e.g., Windows or *nix – the later being faster and more efficient.

There is a lot more …. but now we are moving in to the arena of MySql tuning and that is another book.

Processing Steps for the Pre-Processor.

1. Having created tables in your MySQL database you can get a listing of them by running :

prtschema2 DataBase-Name PrintSpool-Name

This will create a file schema.t and print it out retaining the file. I use a non existent spool name to avoid printing.

For help on the parameters just use :

prtschema2 help | HELP

The bar '|' stands for OR so, use one of the options.

2. To create copy book entries that can be included within a Cobol source using COPY copybookname run :

bldcopy2 OutputFileName DataBaseName TableName FieldNamePrefix

Where FieldNamePrefix = the letters used as xyz- that will prefix all Cobol field names.
For help on the parameters just use :

bldcopy2 help | HELP

You will need to do this for each table in the DB that you are going to use in this way.

An example using ACASDB (in directory ACAS) for stock-rec and was created using :

```
bldcopy2 stock-rec ACASDB STOCK-REC XYZ1
```

3. To create a Cobol source from the Sql source file where this file has extension of .scb (which stands for s=sql, cb = cobol) run :

presql2 InputFileName.scb OutputFileName.cbl FIXED | FREE | fixed | free

For help on the parameters just use :

presql2 help | HELP

Also see example scripting in comp-stockMT.sh within the directory ACAS.

This program will accept the GC >>SOURCE FIXED or >>SOURCE FREE source line but the first one **MUST** be within records 1 or 2 of the source code. It does support extra examples of these further down in the sources so jumping from fixed to free and/or free to fixed within the sources is supported.

If the program finds an error in your SQL syntax or a lack of required parameters including lack of end sequence, e.g.,

/MYSQL INIT\ without /MYSQL-END\ or a missing line between such as BASE=

It will abort having displayed the error.

I did have a look at changing this, but it would involve a lot of changes to the code and the coding structure and I had a more pressing need to just use it.

It does help to keep your coding, somewhat more precise though !

Also, do **NOT** include any floating comments within the /MYSQL blocks. For example :

```
/MYSQL FETCH\              *> any comments
      TABLE=table name     *> or any other comments
/MYSQL-END\                *> or any more comments
```

I have now changed the coded for this, but not necessarily checked that this is fully working although 'should' be.

I also recommend that all the PSQL (presql2 SQL commands) are coded from CC 3 (Column 3) and **always** in upper-case. It will not find any in lower case or a mixture of both.

I did have a quick look to support lower-case and the floating comments but ran out of time and it was easy to deal with – just don't do it..

Updates*:* I have starting looking at this problem and hopefully will deal with it, but avoid it anyway. Likewise using floating comments before the PSQL commands should be avoided but hopefully I will have fixed these issues although the Cobol output file (for processing with cobc) may have these comments removed but there again the primary source file is the one input to presql2.

Special versions of all three programs for use with Mysql as against Mariadb, have also been created and stored in the Variations directory, named as :

presql2o
presql2M

and likewise :

prtschema2o
prtschema2M
  along with the scripts to compile them.

These are specific for MySQL (presql2o) and MariaDB (presql2M). There is a minor difference between these two RDB versions but not for normal user programming, and this causes a problem with prtschema2.. Basically there is an extra column added to the COLUMNS table (in the information_schema database) along with differing data definitions and it was easier to just create two programs of each to handle it. The other two programs are just for my convenience as I use multi RDB servers on different boxes or systems.

They use a parameter file to hold the RDB information to connect to the right server, therefore simplifying the need to change the programs sources.
These files are called presql2o.param and presql2M.param respectively. Note that the .param extension is a fixed name.

Here is a sample of a param file for the Mysql version running on a system other than the one for development :

```
DBHOST=192.168.1.5
DBUSER=mysql
DBPASSWD=mysqlpass
DBNAME=information_schema
DBPORT=3306
DBSOCKET=/tmp/mysql.sock
```

and here is one for running on the same as development but using mariadb :

```
DBHOST=localhost
DBUSER=dev-prog-001
DBPASSWD=mysqlpass
DBNAME=information_schema
DBPORT=3306
DBSOCKET=/home/mysql/mysql.sock
```

If you look at the code for presql2o you will see at paragraph 2131-Get-RDB-Params a call to cobmysqlapi using "read_params" and here is a snippet of the code for WS and procedure div:

WS Division :

```
 copy “COPY MYSQL-VARIABLES”
```

This copybook contains all of the WS-Mysql variable required for your program.

Procedure Division :

```
 2131-Get-RDB-Params.                    *> reads file presql2.param
     move     spaces to               WS-MYSQL-Host-Name
                                      WS-MYSQL-Implementation
                                      WS-MYSQL-Password
                                      WS-MYSQL-Base-Name
                                      WS-MYSQL-Port-Number
                                      WS-MYSQL-Socket.
     Call     "read_params"     USING ws-parm-prog-name
                                      WS-MYSQL-Host-Name
                                      WS-MYSQL-Implementation
                                      WS-MYSQL-Password
                                      WS-MYSQL-Base-Name
                                      WS-MYSQL-Port-Number
                                      WS-MYSQL-Socket
     End-call
*>
       display "Using as RDB calls ".
       display "Host=" WS-MYSQL-Host-Name.
       display "BaseName=" WS-MYSQL-BASE-NAME.
       display "User=" WS-MYSQL-Implementation.
       display "Password=" WS-MYSQL-Password.
       display "Port=" WS-MYSQL-Port-Number.
       display "Socket=" WS-MYSQL-Socket.
       Display " ".
```

Of course this code block can be used in your application program menu to obtain the RDB

data.
Note that the param file name is found through a variable, set in the program containing the program name, see start of WS at variable :

```
01  ws-parm-prog-name                pic x(8) value "presql2o".
```

The same changes also apply to bldcopy2 as bldcopy2 and prtschema2 as prtschema2 (o or M).

Problems or questions

Any issues, please use the sourceforge forums for GnuCOBOL and I will respond as soon as practical, marking subject with presql2 or if urgent you can email me direct at vbcoen at gmail dot com but use subject as presql2.. Please report any bugs in programs or omissions in documentation.

Latest versions both tested and nightly sources when available can be found on my website at :

http://www.applewood.linkpc.net/files/MySQL-Pre-Compilers/JC-Versions

At date of writing this manual, it is presql2-2.23.zip but there could be a later version present.

With nightly builds running, there is a directory under that marked nightly but be warned it might have minimal testing as the code comes directly from my development environment and in some cases may well have not even been compiled yet. Yep straight from the text editor. So test before overwriting previous versions and that includes compiling them.

I have started working on an update to presql2 to support EXEC SQL for MySQL and Mariadb and possibly others and this will be for versions 2.3 but at time of writing this is is very early days.

Syntax for MySQL Pre Processor Commands

This version only supports operations on **all** columns in a table (*) and only supports one instance of MySQL at a time.
Note that all values shown as upper case are required, so do NOT use lower or mixed case.
For example on DEFINE you cannot use
/mysql define\ or
table=table name,prefix

The underlined / Uppercase words MUST be typed in upper case as shown.

Parameters in braces {} are optional, defaults will be used if parameter not specified

[For examples of usage of these commands, see the source of stockMT.scb in the ACAS directory.]
I do not use the lock, unlock commands as the RDB using the engine InnoDB which deals with such, at least according to the RDB docs.

Where it is indicated that generated code does not include a terminating period it can be included within one or more cobol IF statements but if before a paragraph or section it will need to be inserted for statement termination.

Processes for Working-Storage

```
*****************************************************************************
*    DEFINE
*****************************************************************************
```

DEFINE is to be used to allow the programmer to have the row definitions of a table placed in a program. The programmer must provide the 01 level information in the actual Cobol program. The prefix must be less than 5 characters.

```
/MYSQL DEFINE\
       TABLE=table name,prefix
/MYSQL-END\
```

```
*****************************************************************************
*    VAR
*****************************************************************************
```

VAR tells the Pre Processor where to place the variables (both static mysql routines) and dynamic (row names)) the programmer should specify with the TABLE= parameter for each table that will be accessed in the program and the prefix to apply to each column name when building the Cobol data name (prefix must be less than 5 characters). Remember you are limited to the maximum size of a Cobol user defined field name which includes the prefix. Base name = DB name.

```
/MYSQL VAR\
       BASE=base name
       TABLE=table name,prefix
{      TABLE=table name,prefix }  ...
       ...
/MYSQL-END\
```

Processes for Procedure Division

*Note: T*here is *No ending period generated for most if not all of these*.

```
*****************************************************************************
*    CLOSE
*****************************************************************************
```

CLOSE will generate "PERFORM MYSQL-1980-CLOSE THRU MYSQL-1999-EXIT"

Note there is *No ending period*.

```
/MYSQL CLOSE\
/MYSQL-END\
```

```
*****************************************************************************
*    DELETE
*****************************************************************************
```

DELETE will delete rows and WS-MYSQL-COUNT-ROWS will contain the number of rows deleted after the deletion occurs.
You should test this number to see if it contains the number of rows that you expect.

```
/MYSQL DELETE\
       TABLE=table name
       WHERE=variable containing conditions
/MYSQL-END\
```

Note here is No ending period.

```
*****************************************************************************
*    FETCH
*****************************************************************************
```

FETCH will generate CALL "MySQL_fetch_row" USING WS-MYSQL-RESULT,
COLUMN-NAME1 COLUMN-NAME2 COLUMN-NAME-3 ….

Note here is No ending period.

WS-MYSQL-COUNT-ROWS will contain the number of rows found.

```
/MYSQL FETCH\
       TABLE=table name
/MYSQL-END\
```

```
*****************************************************************************
```
* FREE
```
*****************************************************************************
```

FREE will generate CALL "MySQL_free_result" USING WS-MYSQL-RESULT

```
/MYSQL FREE\
       TABLE=table name
/MYSQL-END\
```

```
*****************************************************************************
```
* INIT
```
*****************************************************************************
```

INIT will populate the fields needed to open a data base and create :
"PERFORM MYSQL-1000-OPEN THRU MYSQL-1090-EXIT"
Note: No ending period.
Note that if RUNTIME is used the program expects to open a base depending upon run-time variables. Consequently the preprocessor will not move anything into WS-MYSQL-BASE-NAME or WS-MYSQL-PASSWORD.

The programmer must perform the moves before invoking /MYSQL INIT\.

Data variables will be generated based upon the "base name" as a model.
That means that the run-time base must match the "base name" structure.

Also note that both /MYSQL INIT\ and /MYSQL VAR\ require the definition of the base name (BASE=). The base name in the VAR command is used to build the data structures.
The base name in the INIT command is used to open the data base.

```
/MYSQL INIT\
   BASE=base name
   {HOST}=host name                          default="localhost"
   {IMPLEMENTATION}=implementation name      default="mysql"
   PASSWORD=password
   {PORT}=port number                        default="3306"
   {SOCKET}=socket name                      default="var/run/mysqld/mysqldl.sock"
   {RUNTIME}                                 Take database name only at run time.
                                             WARNING I have not played with this option.
                                             But guess it would have limited usage.
/MYSQL-END\
```

****
* **INSERT**
****

INSERT will generate "insert into TABLE-NAME set COLUMN-NAME1="data1", COLUMN-NAME2="data2" ... for every column in a table.
WS-MYSQL-COUNT-ROWS will contain the number of rows successfully inserted and should be tested for (at least once). Note: N*o ending period*.
You should test for any error arising using such coding sequence as

```
   if       WS-MYSQL-Count-Rows = zero
            call  "MySQL_errno" using WS-MYSQL-Error-Number
            call  "MySQL_sqlstate" using WS-MYSQL-SQLstate
            move  WS-MYSQL-SqlState   to SQL-State
            if    WS-MYSQL-Error-Number  not = "0  "
                  move WS-MYSQL-Error-Number to SQL-Err
                  call "MySQL_error" using WS-MYSQL-Error-Message
                  move WS-MYSQL-Error-Message to SQL-Msg
            end-if
            any additional code . . .
```

```
/MYSQL INSERT\
       TABLE=table name
/MYSQL-END\
```

****
* **LOCK**
****

LOCK will lock a table

```
/MYSQL LOCK/
       TABLE=table name
/MYSQL-END\
```

****
* **PRO**
****

PRO tells the Pre Processor where to place the Mysql procedures [similar to using INCLUDE or COPY].

```
/MYSQL PRO\
/MYSQL-END\
```

****
* **SELECT**
****

SELECT will generate "select COLUMN-NAME1, COLUMN-NAME2, COLUMN-NAME3, ...
from TABLE-NAME
WHERE variable name containing conditionals"
"PERFORM MYSQL-1200-MYSQL-SELECT THRU MYSQL-1209-EXIT"

Note that the SELECT statement will always select all columns (i. e. SELECT *) and that it only handles one table at a time (no joining)
SELECT will also perform a STORE RESULT.

Note: No ending period.

IN MySQL, the proper technique for getting rows (records) is:
1) SELECT (a STORE RESULT will automatically be performed)
2) FETCH the rows that you want one at a time
3) FREE to release the RESULT cursor set created at SELECT time

```
/MYSQL SELECT\
      TABLE=table name
      WHERE=variable name containing conditionals
/MYSQL-END\
```

****
* **SWITCHDB**
****

SWITCHDB will insert a
MYSQL-1240-SWITCH-DB THRU MYSQL-1249-EXIT command.

Note that the programmer will need to insert the new base name into MYSQL-WS-BASE-NAME before executing this command.

```
/MYSQL SWITCHDB\
/MYSQL-END\
```

****
* **UNLOCK**
****

UNLOCK will unlock the data base

```
/MYSQL UNLOCK\
/MYSQL-END\
```

****
* **UPDATE**
****

UPDATE will generate "update TABLE-NAME set COLUMN-NAME1="data1",
COLUMN-NAME2="data2" ...
for every column in a table.

Note: No ending period.

In order to check the status of the UPDATE you should :
1) CALL "MySQL_errno" USING WS-MYSQL-ERROR-NUMBER
2) IF WS-MYSQL-COUNT-ROWS < the number of rows expected and
WS-MYSQL-ERROR-NUMBER not = "0 " then you have an error condition.

Use the ANSI SQL & ODBC 5 character codes in WS-MYSQL-ERROR-NUMBER.

Using v 38 of the api you can also use SQLSTATE which always will create a code, e.g.,

```
call  "MySQL_sqlstate" using WS-MYSQL-SQLstate
move  WS-MYSQL-SqlState   to SQL-State
```

```
/MYSQL UPDATE\
      TABLE=table name
      WHERE=variable name containing conditionals
/MYSQL-END\
```

Syntax for Supported EXEC SQL Commands

ODBC, MySQL and MariabDB commands

This is in development, may be.

Appendix A – Update Information

Taken from README.SVN

2022-08-03 More manual grammar and clean ups to match up with current software version.

2019-05-10 More manual clean ups.

2019-05-09 Grammar, url, and other typo's.

2018-02-27 More grammar, typo's changed program names to include trailing 2. More information in building and more source acas examples i.e., coming from printcbl to include the copy books etc.

2017-08-31 Clean up grammar and minor errors in this document. Added details regarding update sources for presql2 including the nightly builds which are only updated if the new zipped file is different in size to the current nightly archive.

2017-08-31 More grammar and typos cleaned up as well as tidy up some ambiguous details.

2019-09-25 Updated details regarding prtschema2 during set up as excess field size if fixed by running a UDF replace verb as the problem arises to both Mysql and Mariadb systems.

Appendix B – Licenses for Documentation and Software

Creative Commons Attribution-Share Alike v4.0 International Public License

By exercising the Licensed Rights (defined below), You accept and agree to be bound by the terms and conditions of this Creative Commons Attribution-Share Alike 4.0 International Public License ("Public License"). To the extent this Public License may be interpreted as a contract, You are granted the Licensed Rights in consideration of Your acceptance of these terms and conditions, and the Licensor grants You such rights in consideration of benefits the Licensor receives from making the Licensed Material available under these terms and conditions.

Section 1 – Definitions.

a. **Adapted Material** means material subject to Copyright and Similar Rights that is derived from or based upon the Licensed Material and in which the Licensed Material is translated, altered, arranged, transformed, or otherwise modified in a manner requiring permission under the Copyright and Similar Rights held by the Licensor. For purposes of this Public License, where the Licensed Material is a musical work, performance, or sound recording, Adapted Material is always produced where the Licensed Material is synced in timed relation with a moving image.

b. **Adapter's License** means the license You apply to Your Copyright and Similar Rights in Your contributions to Adapted Material in accordance with the terms and conditions of this Public License.

c. **BY-SA Compatible License** means a license listed at creativecommons.org/compatiblelicenses, approved by Creative Commons as essentially the equivalent of this Public License.

d. **Copyright and Similar Rights** means copyright and/or similar rights closely related to copyright including, without limitation, performance, broadcast, sound recording, and Sui Generis Database Rights, without regard to how the rights are labelled or categorized. For purposes of this Public License, the rights specified in Section 2(b)(1)-(2) are not Copyright and Similar Rights.

e. **Effective Technological Measures** means those measures that, in the absence of proper authority, may not be circumvented under laws fulfilling obligations under Article 11 of the WIPO Copyright Treaty adopted on December 20, 1996, and/or similar international agreements.

f. **Exceptions and Limitations** means fair use, fair dealing, and/or any other exception or limitation to Copyright and Similar Rights that applies to Your use of the Licensed Material.

g. **License Elements** means the license attributes listed in the name of a Creative Commons Public License. The License Elements of this Public License are Attribution and ShareAlike.

h. **Licensed Material** means the artistic or literary work, database, or other material to which the Licensor applied this Public License.

i. **Licensed Rights** means the rights granted to You subject to the terms and conditions of this Public License, which are limited to all Copyright and Similar Rights that apply to Your use of the Licensed Material and that the Licensor has authority to license.

j. **Licensor** means the individual(s) or entity(ies) granting rights under this Public License.

k. **Share** means to provide material to the public by any means or process that requires permission under the Licensed Rights, such as reproduction, public display, public performance, distribution, dissemination, communication, or importation, and to make material available to the public including in ways that members of the public may access the material from a place and at a time individually chosen by them.

l. **Sui Generis Database Rights** means rights other than copyright resulting from Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, as amended and/or succeeded, as well as other essentially equivalent rights anywhere in the world.

m. **You** means the individual or entity exercising the Licensed Rights under this Public License. **Your** has a corresponding meaning.

n.

Section 2 – Scope.

a. **License grant**.

1. Subject to the terms and conditions of this Public License, the Licensor hereby grants You a worldwide, royalty-free, non-sublicensable, non-exclusive, irrevocable license to exercise the Licensed Rights in the Licensed Material to:
    A. reproduce and Share the Licensed Material, in whole or in part; and
    B. produce, reproduce, and Share Adapted Material.
2. Exceptions and Limitations. For the avoidance of doubt, where Exceptions and Limitations apply to Your use, this Public License does not apply, and You do not need to comply with its terms and conditions.
3. Term. The term of this Public License is specified in Section 6(a).
4. Media and formats; technical modifications allowed. The Licensor authorizes You to exercise the Licensed Rights in all media and formats whether now known or hereafter created, and to make technical modifications necessary to do so. The Licensor waives and/or agrees not to assert any right or authority to forbid You from making technical modifications necessary to exercise the Licensed Rights, including technical modifications necessary to circumvent Effective Technological Measures. For purposes of this Public License, simply making modifications authorized by this Section 2(a)(4) never produces Adapted Material.
5. Downstream recipients.
    A. Offer from the Licensor – Licensed Material. Every recipient of the Licensed Material automatically receives an offer from the Licensor to exercise the Licensed Rights under the terms and conditions of this Public License.
    B. Additional offer from the Licensor – Adapted Material. Every recipient of Adapted Material from You automatically receives an offer from the Licensor to exercise the Licensed Rights in the Adapted Material under the conditions of the Adapter's License You apply.
    C. No downstream restrictions. You may not offer or impose any additional or different terms or conditions on, or apply any Effective Technological Measures to, the Licensed Material if doing so restricts exercise of the Licensed Rights by any recipient of the Licensed Material.
6. No endorsement. Nothing in this Public License constitutes or may be construed as permission to assert or imply that You are, or that Your use of the Licensed Material is, connected with, or sponsored, endorsed, or granted official status by, the Licensor or others designated to receive attribution as provided in Section 3(a)(1)(A)(i).

b. **Other rights**.

1. Moral rights, such as the right of integrity, are not licensed under this Public License, nor are publicity, privacy, and/or other similar personality rights; however, to the extent possible, the Licensor waives and/or agrees not to assert any such rights held by the Licensor to the limited extent necessary to allow You to exercise the Licensed Rights, but not otherwise.
2. Patent and trademark rights are not licensed under this Public License.
3. To the extent possible, the Licensor waives any right to collect royalties from You for the exercise of the Licensed Rights, whether directly or through a collecting society under any voluntary or waivable statutory or compulsory licensing scheme. In all other cases the Licensor expressly reserves any right to collect such royalties.
4.

Section 3 – License Conditions.

Your exercise of the Licensed Rights is expressly made subject to the following conditions.

a. **Attribution**.

1. If You Share the Licensed Material (including in modified form), You must:
    A. retain the following if it is supplied by the Licensor with the Licensed Material:
        i. identification of the creator(s) of the Licensed Material and any others designated to receive attribution, in any reasonable manner requested by the Licensor (including by pseudonym if designated);
        ii. a copyright notice;
        iii. a notice that refers to this Public License;
        iv. a notice that refers to the disclaimer of warranties;
        v. a URI or Hyperlink to the Licensed Material to the extent reasonably practicable;
    B. indicate if You modified the Licensed Material and retain an indication of any previous modifications; and
    C. indicate the Licensed Material is licensed under this Public License, and include the text of, or the URI or Hyperlink to, this Public License.
2. You may satisfy the conditions in Section 3(a)(1) in any reasonable manner based on the medium, means, and context in which You Share the Licensed Material. For example, it may be reasonable to satisfy the conditions by providing a URI or Hyperlink to a resource that includes the required information.
3. If requested by the Licensor, You must remove any of the information required by Section 3(a)(1)(A) to the extent reasonably practicable.

b. **ShareAlike**.
In addition to the conditions in Section 3(a), if You Share Adapted Material You produce, the following conditions also apply.
1. The Adapter’s License You apply must be a Creative Commons license with the same License Elements, this version or later, or a BY-SA Compatible License.
2. You must include the text of, or the URI or Hyperlink to, the Adapter's License You apply. You may satisfy this condition in any reasonable manner based on the medium, means, and context in which You Share Adapted Material.
3. You may not offer or impose any additional or different terms or conditions on, or apply any Effective Technological Measures to, Adapted Material that restrict exercise of the rights granted under the Adapter's License You apply.
4.

Section 4 – Sui Generis Database Rights.
Where the Licensed Rights include Sui Generis Database Rights that apply to Your use of the Licensed Material:
a. for the avoidance of doubt, Section 2(a)(1) grants You the right to extract, reuse, reproduce, and Share all or a substantial portion of the contents of the database;
b. if You include all or a substantial portion of the database contents in a database in which You have Sui Generis Database Rights, then the database in which You have Sui Generis Database Rights (but not its individual contents) is Adapted Material, including for purposes of Section 3(b); and
c. You must comply with the conditions in Section 3(a) if You Share all or a substantial portion of the contents of the database.
For the avoidance of doubt, this Section 4 supplements and does not replace Your obligations under this Public License where the Licensed Rights include other Copyright and Similar Rights.

Section 5 – Disclaimer of Warranties and Limitation of Liability.
a. Unless otherwise separately undertaken by the Licensor, to the extent possible, the Licensor offers the Licensed Material as-is and as-available, and makes no representations or warranties of any kind concerning the Licensed Material, whether express, implied, statutory, or other. This includes, without limitation, warranties of title, merchantability, fitness for a particular purpose, non-infringement, absence of latent or other defects, accuracy, or the presence or absence of errors, whether or not known or discoverable. Where disclaimers of warranties are not allowed in full or in part, this disclaimer may not apply to You.

b. **To the extent possible, in no event will the Licensor be liable to You on any legal theory (including, without limitation, negligence) or otherwise for any direct, special, indirect, incidental, consequential, punitive, exemplary, or other losses, costs, expenses, or damages arising out of this Public License or use of the Licensed Material, even if the Licensor has been advised of the possibility of such losses, costs, expenses, or damages. Where a limitation of liability is not allowed in full or in part, this limitation may not apply to You.**
c. The disclaimer of warranties and limitation of liability provided above shall be interpreted in a manner that, to the extent possible, most closely approximates an absolute disclaimer and waiver of all liability.

Section 6 – Term and Termination.

a. This Public License applies for the term of the Copyright and Similar Rights licensed here. However, if You fail to comply with this Public License, then Your rights under this Public License terminate automatically.
b. Where Your right to use the Licensed Material has terminated under Section 6(a), it reinstates:
    1. automatically as of the date the violation is cured, provided it is cured within 30 days of Your discovery of the violation; or
    2. upon express reinstatement by the Licensor.

    For the avoidance of doubt, this Section 6(b) does not affect any right the Licensor may have to seek remedies for Your violations of this Public License.
c. For the avoidance of doubt, the Licensor may also offer the Licensed Material under separate terms or conditions or stop distributing the Licensed Material at any time; however, doing so will not terminate this Public License.
d. Sections 1, 5, 6, 7, and 8 survive termination of this Public License.

Section 7 – Other Terms and Conditions.

a. The Licensor shall not be bound by any additional or different terms or conditions communicated by You unless expressly agreed.
b. Any arrangements, understandings, or agreements regarding the Licensed Material not stated herein are separate from and independent of the terms and conditions of this Public License.

Section 8 – Interpretation.

a. For the avoidance of doubt, this Public License does not, and shall not be interpreted to, reduce, limit, restrict, or impose conditions on any use of the Licensed Material that could lawfully be made without permission under this Public License.
b. To the extent possible, if any provision of this Public License is deemed unenforceable, it shall be automatically reformed to the minimum extent necessary to make it enforceable. If the provision cannot be reformed, it shall be severed from this Public License without affecting the enforceability of the remaining terms and conditions.
c. No term or condition of this Public License will be waived and no failure to comply consented to unless expressly agreed to by the Licensor.
d. Nothing in this Public License constitutes or may be interpreted as a limitation upon, or waiver of, any privileges and immunities that apply to the Licensor or You, including from the legal processes of any jurisdiction or authority.

Creative Commons is not a party to its public licenses. Notwithstanding, Creative Commons may elect to apply one of its public licenses to material it publishes and in those instances will be considered the “Licensor.” Except for the limited purpose of indicating that material is shared under a Creative Commons public license or as otherwise permitted by the Creative Commons policies published at creativecommons.org/policies, Creative Commons does not authorize the use of the trademark “Creative Commons” or any other trademark or logo of Creative Commons

GNU GENERAL PUBLIC LICENSE (*Software*) v3 (29 June 2007)

Preamble
The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.
To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.
For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.
For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.
Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.
Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.
The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.
"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.
"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.
To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.
A "covered work" means either the unmodified Program or a work based on the Program.
To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.
To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.
An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked sub-programs that the work is specifically designed to require, such as by intimate data communication or control flow between those sub-programs and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sub-licensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and non-commercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favour of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a

User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensor's or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensor's and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensor's and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits re-licensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such re-licensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant

source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensor's, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross claim or counter claim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the

contributor version. For purposes of this definition, "control" includes the right to grant patent sub-licenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defences to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special

requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS