Procedures for Estimating Volatile Organic Compound Emissions from Pesticides

F. Spurlock

STATE OF CALIFORNIA
Environmental Protection Agency
Department of Pesticide Regulation
Environmental Monitoring Branch
Sacramento, California 95814-5624

EH02- 03

Procedures for Estimating Volatile Organic Compound Emissions from Pesticides

F. Spurlock

STATE OF CALIFORNIA
Environmental Protection Agency
Department of Pesticide Regulation
Environmental Monitoring Branch
Sacramento, California 95814-5624

EH02- 03

CONTENTS

I. INTRODUCTION

This report describes technical procedures for calculating the Department of Pesticide Regulation's (DPR) annual pesticide volatile organic compound emissions (VOC) inventory as of July, 2002. The VOC inventory is a database of estimated VOC emissions from agricultural and commercial structural pesticide applications. Additional background information on DPR's VOC emissions project is available on the web at http://www.cdpr.ca.gov/docs/pur/vocproj/vocmenu.htm.

IA. Software and background knowledge required

The VOC inventory is calculated using combined data from the pesticide use report (PUR), Product/Label database, pesticide emission potential (EP) database, and various geographic coordinate and Air Resources Board administrative area data. The software programs and their principal roles in calculating the inventory are:

ORACLE – Source of PUR and Product/Label data. The source data are downloaded from ORACLE using Structured Query Language (SQL).

SAS – Actual VOC calculations, post-processing diagnostics and data analysis.

EXCEL – Input data pre-processing, post-processing diagnostics, data analysis, and maintenance of the EP database.

Calculation of the inventory as outlined in this report assumes knowledge of SQL, SAS programming language, and familiarity with EXCEL – including use of EXCEL's text file import and export options. In addition, knowledge of the structure and variables of the PUR and the Product/Label databases is also required. Table schema for the PUR and Product/Label database (Appendix 1) provide an overview of database structures. The following electronic documentation files/data dictionaries for these two databases are currently available (July, 2002) on the server EMPM, in the directory *i:\spurlock\VOC\documentation\,* or available from the author on CD-ROM upon request.

label_database_data_dictionary.xls – an EXCEL spreadsheet file that contains variable descriptions for the Product/Label tables.

PUR_data_dictionary.pdf – A copy of DPR's 1998 CD-ROM PUR documentation that includes a brief description of all variables in the PUR.

VOC_lookup_tables.xls – A spreadsheet that contains lookup tables of various codes (air basin codes, county codes, site codes, etc.) and other useful information.

PUR_outlier_description.doc – Contains a brief description of the PUR outlier tables.

IB. Filename Conventions

Year. The variable '***YY***' in a filename denotes the inventory year, e.g. ***YY***=99 denotes files associated with PUR and emissions data year 1999.

ORACLE download file extensions. All data files that originate from ORACLE are assigned a file extension of ".dat". These *.dat files originate from the PUR or the Product/Label database and include both fixed column-width text format files and delimited files where the delimiter is the "~" (tilde) character. The particular format is evident from the SQL query used to extract the data (e.g., Appendix 2). All SQL query files have an extension of ".SQL".

SAS file extensions. All files that originate from processing in SAS are assigned a file extension of ".out". These SAS output files are all fixed column-width text files with no headers. The order and column widths of the data fields may be determined from the respective SAS source program (Appendices 3, 4). All SAS program language files have an extension of ".SAS".

EXCEL text files. Certain text data files that originate from EXCEL are assigned an extension of ".csv"; these are comma-delimited text files.

IC. Overview of Calculation Procedure.

The VOC inventory is calculated using the following steps.

Download “raw” PUR data and Product/Label data.
Prepare data input files.
Calculate draft VOC inventory.
Conduct post-processing diagnostic analyses.
Revise input data files or program based on diagnostic results, repeat steps 3 -5
Calculate final VOC inventory.

Figure 1 illustrates the general data flow and input during a typical calculation with no repeated steps.

ID. Emission Potentials.

Emission potentials are derived using a variety of methods. These methods are subject to periodic update, and a detailed explanation of current methods for estimating EPs is available on-line (Spurlock, 2002a).

IE. Emission Data.

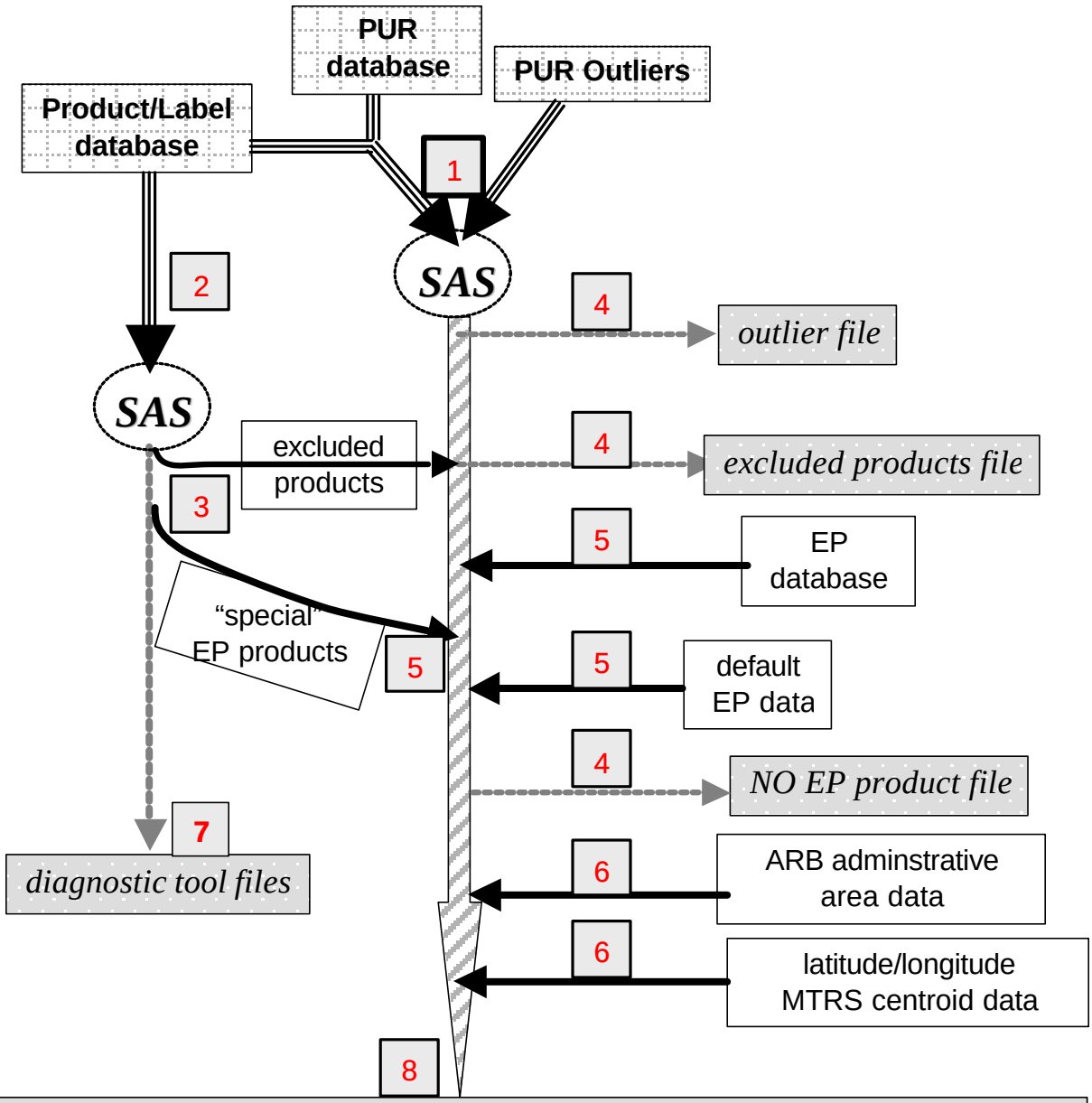
The VOC inventory is a catalog of estimated VOC emissions from individual pesticide applications that have been reported in the PUR. The emission from each pesticide application is calculated as follows:

VOC emission (lbs) = [lbs pesticide product applied] * [product-specific EP]

Not all pesticide applications or products are included in the inventory; the following products/applications are excluded from the inventory, either by excluding them from the initial PUR data download, or by removing them during subsequent processing.

1. Those products with certain formulations as indicated by the following values of product.formula_cd in the Product/Label database.
   D0 (gel/paste/cream), F0 (impregnated material), I0 (paint/coatings), T0 (other, liquid), U0 (other, dry)

Figure 1. VOC inventory data flow



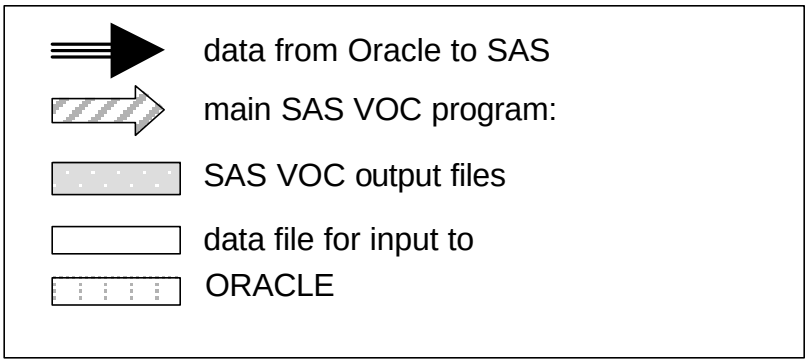
PRIMARY OUTPUT FILES:

1. ***YY***_ARB_VOC.out: detailed use/emission data by PUR.use_no

2. ***YY***_DPR_VOC.out: use/emission summarized by prodno, air district, ozone non-attainment area, month, site

Figure 1 (continued)

key:



annotations:

1 PUR and PUR-outlier data are extracted from ORACLE to text input files for the main SAS VOC program, VOC_***YY***.SAS (see Table 2).

2 Product/Label data are extracted for input to SAS pre-processing programs.

3 The SAS pre-processing programs generate ancillary input files required by the main SAS VOC program VOC_***YY***.SAS, and to create certain files needed later for diagnostics and output data analysis (see Table 3).

4 VOC_***YY***.SAS creates various secondary output data files during processing (see section IID.). These are used for documentation and diagnostic purposes.

5 Emission potential (EP) data are assigned to products to enable calculation of pesticide product VOC emissions. EP data come from 3 sources: the EP database, assignment of default EPs (based on product formulation code), or assignment of default EPs specially defined for particular special products.

6 ARB administrative area designations (air basin, air district, and non-attainment area) are assigned to all VOC emission records. Latitude/longitude MTRS centroid coordinates are assigned only to those records associated with specific MTRS through the corresponding PUR records.

7 Additional secondary output files created during SAS pre-processing are used for post-processing diagnostics and output data analysis purposes (see Table 3).

8 VOC_***YY***.SAS produces 2 main output files of estimated VOC emissions (see section IID.). The ***YY***_ARB_VOC output file contains unaggregated VOC emission and pesticide use data indexed by PUR.use_no, analogous to the PUR and containing about 2.5 million records. The DPR output file ***YY***_DPR_VOC.out contains the same data, aggregated by month, site, and product. This smaller output file contains about 350,000 records.

2. Those products that are registered solely as technical pesticides. These are commonly used in product reformulation. These are identified using the Product/Label database according to following condition.

IF AND ONLY IF 99000 ≤ prod_site.site_code ≤ 99999

3. Those products that are registered solely as (i) adjuvants or (ii) adjuvants and fertilizers. These are identified using the Product/Label database according to following condition.

prod_type_pesticide.typepest_cd = (i) only A0, or (ii) only (A0 and J0)

4. Reported pesticide applications that are identified as "probable outliers" in the PUR using standard criteria (Wilhoit, 1998). Probable outliers are defined as those PUR records that exceed criterion 1a or criterion 2b or criterion 4d in Wilhoit (1998).

5. Reported pesticide applications to sites that are not considered to be "agricultural" or "commercial structural". See *i:\spurlock\VOC\documentation\ VOC_lookup_tables.xls* for a list of PUR.site_code included in the VOC inventory and their classification as to agricultural or commercial structural.

IF. Allocating PUR and Emissions to ARB Administrative Areas

The estimated VOC emission corresponding to each pesticide application is reported in the final ***YY***_ARB_VOC.out data file with the associated air district, air basin, and ozone non-attainment administrative area designation. For PUR records that include spatial attribute data (Meridian/Township/Range/Section, MTRS) the administrative area designations are assigned using lookup tables that associate each MTRS with it's corresponding air district, air basin, and non-attainment area.

Certain types of PUR data include the county in which the pesticide application was made, but no MTRS data. These "nonspatial" PUR data are reported to the counties in

monthly summaries, and are identified by PUR.record_id equal to “C”, “G”, or “2”. During calculation of the inventory, nonspatial PUR data are assigned to one of 3 use categories (Table 1), enabling their corresponding VOC emissions to be allocated to different adminstrative areas that cross county lines.

Table 1. Assigned use categories for “nonspatial” PUR data (PUR.record_id = C, G, or 2).

Use Category	PUR.site_code included[A]
commercial structural/ landscape maintenance	10, 30, 61001,63000,67003, 72000
rights-of-way	40
commodity fumigations	all PUR.site_code > 89 except (61001, 63000, 67003, 72000)

[A] for a list of PUR.site_code see *i:\spurlock\VOC\documentation\VOC_lookup_tables.xls*

The VOC emissions are allocated to geographic subsets of the county by mimicking the known geographic distribution of other county data that are presumed similarly distributed. For example, commercial structural and landscape maintenance (CSLM) pesticide applications are assumed geographically distributed similarly to population. Using county census data, fractions have been derived that are used to similarly allocate a county’s CSLM emissions among the different geographic subsets of the county that have unique administrative area designations. Rights of way applications are similarly allocated according to the proportion of roads, waterways, power lines, and other public property within a county as determined using data from Teale Data Center’s Geographic Information System Library. However, commodity fumigations are allocated based on opinions of county agricultural commissioner staff in the respective counties as no obvious well-known surrogate data set is available.

II. SPECIFIC PROCEDURES FOR CALCULATING THE VOC INVENTORY

Note: All SQL queries and SAS programs are provided in Appendices 2-4, respectively.

IIA. Data Download

Download PUR, outlier, and Product/Label data from ORACLE. All SQL queries and their resultant data files are listed in Table 2. The PUR data is downloaded into two text files, one containing PUR records with spatial attributes (Meridian/Township/Range/ Section, MTRS), and a second file containing those PUR records lacking MTRS. The file names are spatial_***YY***.dat and nonspatial_***YY***.dat, respectively, for use year '***YY***'.

Table 2. SQL queries for downloading data from ORACLE PUR and Product/Label databases.

ORACLE source table	SQL query filename	output data filename
PUR	spatial_***YY***.sql	spatial_***YY***.dat
PUR	nonspatial_***YY***.sql	nonspatial_***YY***.dat
outlier***YY***	outlier***YY***.sql	outlier_***YY***.dat
prodsite	prodsite***YY***.sql	prodsite***YY***.dat
typepest	typepest***YY***.sql	typepest***YY***.dat
product	short_product_tilde.sql	short_product_tilde.dat
product	product_tilde.sql	product_tilde.dat
prod_chem, product, chemical	special_AIs_for_preprocessing.sql	special_AIs_preprocessing.dat
prod_chem	prodchem_tilde.sql	prodchem_tilde.dat
prod_chem	mebr.sql	mebr.dat

IIB. SAS Data Preprocessing

Run the following SAS data pre-processing programs. Note that these programs read various *.dat files created from the SQL queries above. Consequently paths to the data files must be identical to those in the FILENAME specification section listed in the beginning of each SAS program.

Table 3. SAS pre-processing programs

SAS Program	Output file(s)
SPECIAL_PROD_PREPROCESS.SAS	special_prods.out
EXCLUDED_PRODUCTS.SAS	excluded_prods.out
LINK_PRODCHEM_PRODUCT_TABLES_2GET_ MAXCHEMPRODPCT.SAS	max_chem_prod_name_regno_formcd.out arb_toxic_chems.out

IIC. Additional Data Files

Obtain the remaining data files required to calculate inventory. These are files that are only updated/modified infrequently if at all. Consequently they usually do not need to be generated every year, and can be copied from the location of the previous year's files. The one likely exception is the product emission potential data file, which is updated more frequently than the others.

1. 'ep_voc_***YY***.csv' – The most recent file of product emission potentials. This file may be changed depending on changes to EP estimation methods, or addition of new EP data from registrant submissions.

2. 'ep_defaults.csv' – a file containing "default" emission potentials based on formulation code. This file is modified only if/when the default EPs are updated.

3. 'lat_long.csv' – a file containing decimal lat/long coordinates for all California MTRS centroids. This file will probably never change.

4. 'statewide.csv' – a file containing air basin, air district, and non-attainment area codes corresponding to all California MTRS. This file should only be modified if ARB's administrative boundaries change.

5. 'nonspatial_admin1.csv', 'nonspatial_admin2.csv', 'nonspatial_admin3.csv', 'nonspatial_admin4.csv', 'nonspatial_admin5.csv', 'nonspatial_admin6.csv' – These files contain data that are used to allocate estimated emissions from nonspatial PUR data to various administrative areas (air basins, air districts, and non-attainment

areas). These files should only be modified if ARB's administrative boundaries change.

IID. Calculating the Inventory

Copy all input data files to the appropriate input file directory as listed in the FILE REFERENCE section of the main SAS VOC calculation program: VOC_***YY***.SAS. Run this program to calculate the inventory, and save the log file for documentation and trouble-shooting purposes.

The VOC_***YY***.SAS output files are

1. ***YY***_VOC_outlier.out

   a list of PUR use_no that are purged before actual calculation of VOC emissions using outlier***YY*** data from ORACLE

2. ***YY***_all_excluded.out

   A list of products that are purged because they are:
   only adjuvants, or
   only (adjuvants and fertilizers), or
   only technicals

3. ***YY***_NoEP.out

   A list of those products that do not have an Emission Potential (EP) in the EP database. These products are often newly registered, and are usually assigned default EPs or EPs of products for which they are sub-registrations or label revisions (Spurlock, 2002a) before actual VOC calculations begin. A product's sub-registration or label revision status may be determined from the California product registration number. A detailed description of California's pesticide product registration numbering system in Appendix 5 *i:\spurlock\VOC\documentation\VOC_lookup_tables.xls*.

4. ***YY***_ARB_VOC.out

The final ARB ***YY*** output file - contains about two million records

5. ***YY***_DPR_VOC.out

A sometimes more useful compact output file, with emissions summarized by prodno and administrative area - contains about 350 - 400 thousand records

IIE. Post-processing Diagnostics

All SAS VOC programs have been tested extensively on a variety of test and actual data sets, however extensive post-processing diagnostic analyses of the emission data is still required to check for possible remaining errors in programming or especially in the input data - both the PUR and the Product/Label database have non-zero error rates. Diagnostics include detailed examination of:

1. outlier records – [***YY***_VOC_outlier.out]
2. excluded product records – [***YY***_all_excluded.out]
3. products without EPs in the EP database – [***YY***_NoEP.out]
4. the primary VOC output data [***YY***_ARB_VOC.out and/or ***YY***_DPR_VOC.out]
5. fumigant use and emissions – including methyl bromide, chloropicrin, metam-sodium, and 1,3,-dichloropropene. Fumigants receive extra attention because of their large relative contribution to overall pesticidal VOC emissions.

The VOC_***YY***.SAS output “outlier”, “excluded product”, and “NoEP product” files may be viewed and manipulated directly in a spreadsheet program as they are comprised of a relatively small number of records (typically about 6,000, 1,000, and 2,000 records, respectively). The primary VOC output data files [***YY***_ARB_VOC.out, ***YY***_DPR_VOC.out] will require pre-analysis summarizing in SAS if spreadsheet software is to be used for diagnostic analysis; most spreadsheet applications have limitations on the number of allowable records.

Specific procedures for diagnostic analysis of the VOC data are not presented here. These will probably vary from year to year as the calculation, EP estimation, and VOC programming procedures continue to evolve. The general goals are to compare pesticide use data and calculated emission data to look for inconsistencies, to evaluate and verify emissions from the highest contributing products and active ingredients (e.g., fumigants), and to manually inspect the outlier, excluded product, and NoEP output files for accuracy. Finally, additional post-processing analyses will usually be required based on management priorities (e.g., Spurlock, 2002b).

IIF. Inventory Revision

The procedures presented here are applicable to calculation of the VOC inventory for a single year – typically for annual updates as new annual PUR data is released. However, on occasion major revisions to the calculation procedures or input data may require a complete recalculation of VOC emissions for all years in the inventory, from 1990 to the present. This may occur, for instance, when EP estimation procedures or EP defaults are changed to improve the accuracy of the emission estimates. In this or similar situations, revision of the entire inventory is warranted so that potential emission trends can be accurately evaluated.

IIG. Documentation

Copies of all SQL queries, input data files, output files, and SAS log files should be transferred to electronic media - such as CD - and archived for documentation purposes.

All major modifications to the calculation procedure or input data should be documented in both a memorandum AND electronic file. A copy of the electronic file should be deposited in the documentation directory, currently located at *i:\spurlock\VOC\documentation\.*

III. REFERENCES

Wilhoit, L. 1998. A Computer Program to Identify Outliers in the Pesticide Use Report Database. Pest Management Analysis and Planning Program, Dept. Pesticide Regulation. pub. PM 98-01.

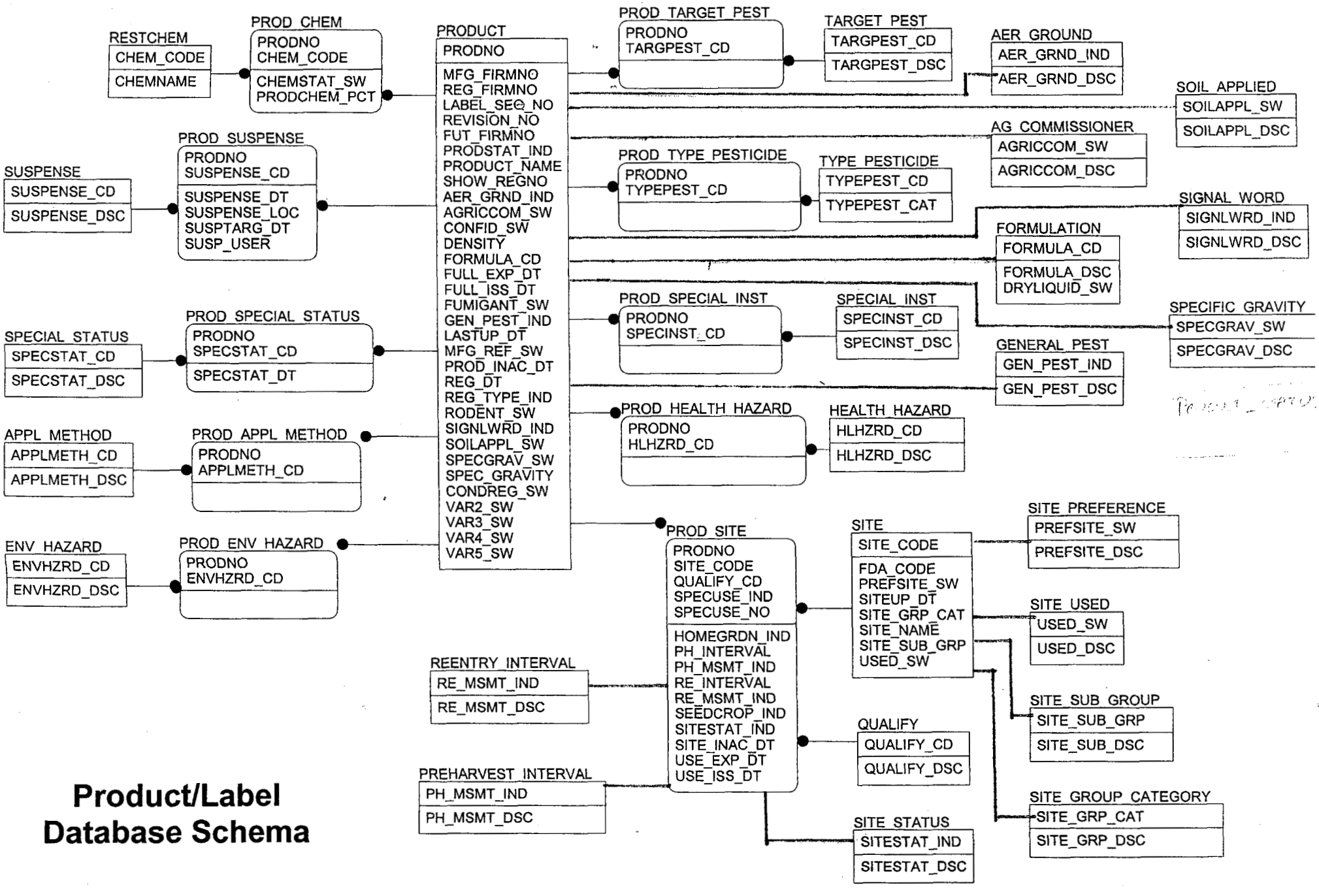
Spurlock, F. 2002a. “Methodology for Determining VOC Emission Potentials of Pesticide Products”, memorandum to J. Sanders, January 7, 2002. available on-line *http://www.cdpr.ca.gov/docs/pur/vocproj/intro.pdf*

Spurlock, F. 2002b. “Summary of 1990-2000 VOC Emission Inventory Data, version 01.12”, memorandum to J. Sanders, February 4, 2002. available on-line *http://www.cdpr.ca.gov/docs/pur/vocproj/vocsum.pdf*

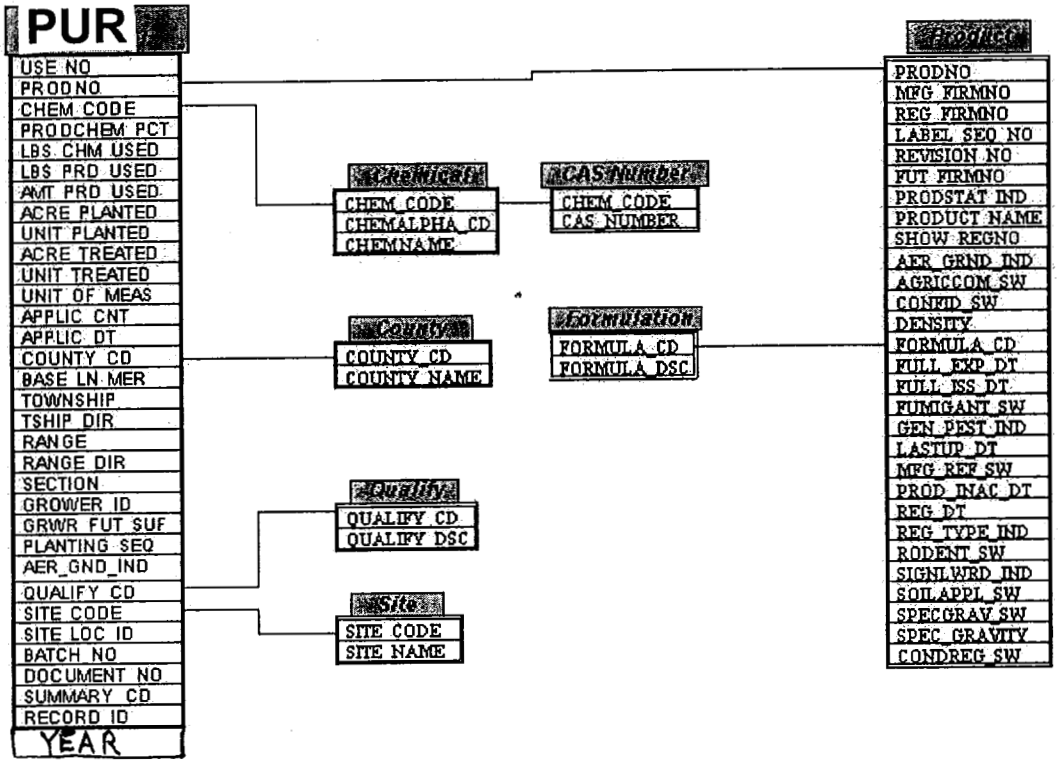
APPENDIX 1

Table Schema for Product/Label and PUR databases

LABEL



Product/Label Database Schema


PUR
USE_NO
PRODNO
CHEM_CODE
PRODCHEM_PCT
LBS_CHM_USED
LBS_PRD_USED
AMT_PRD_USED
ACRE_PLANTED
UNIT_PLANTED
ACRE_TREATED
UNIT_TREATED
UNIT_OF_MEAS
APPLIC_CNT
APPLIC_DT
COUNTY_CD
BASE_LN_MER
TOWNSHIP
TSHIP_DIR
RANGE
RANGE_DIR
SECTION
GROWER_ID
GRWR_FUT_SUF
PLANTING_SEQ
AER_GND_IND
QUALIFY_CD
SITE_CODE
SITE_LOC_ID
BATCH_NO
DOCUMENT_NO
SUMMARY_CD
RECORD_ID
YEAR
CHEM_CODE
CHEMALPHA_CD
CHEMNAME
CHEM_CODE
CAS_NUMBER
COUNTY_CD
COUNTY_NAME
FORMULA_CD
FORMULA_DSC
QUALIFY_CD
QUALIFY_DSC
SITE_CODE
SITE_NAME
PRODNO
MFG_FIRMNO
REG_FIRMNO
LABEL_SEQ_NO
REVISION_NO
FUT_FIRMNO
PRODSTAT_IND
PRODUCT_NAME
SHOW_REGNO
AER_GRND_IND
AGRICCOM_SW
CONFID_SW
DENSITY
FORMULA_CD
FULL_EXP_DT
FULL_ISS_DT
FUMIGANT_SW
GEN_PEST_IND
LASTUP_DT
MFG_REF_SW
PROD_INAC_DT
REG_DT
REG_TYPE_IND
RODENT_SW
SIGNLWRD_IND
SOILAPPL_SW
SPECGRAV_SW
SPEC_GRAVITY
CONDREG_SW

PUR Schema

APPENDIX 2

Structured Query Language (SQL) Queries

APPENDIX 2 – Structured Data Language (SQL) Queries

(NOTE YY=data year, e.g. 98, 99, 00, 01,....)

1. spatial_YY.sql

```
set feedback off
set pagesize 0
set linesize 120
spool spatial_YY.dat
select
  lpad(a.use_no,8)||' '||
  lpad(a.prodno,7)||' '||
  lpad(nvl((to_char(a.applic_dt,'MMDDYYYY')),99999999),8)||' '||
  lpad(nvl(a.lbs_prd_used,0),14)||' '||
  lpad(nvl(a.acre_treated,0),10)||' '||
  lpad(a.unit_treated,1)||' '||
  lpad(a.site_code,6)||' '||
  lpad(a.county_cd,2)||' '||
  lpad(a.county_cd,2)||
  lpad(a.base_ln_mer,1)||
  lpad(a.township,2)||
  lpad(a.tship_dir,1)||
  lpad(a.range,2)||
  lpad(a.range_dir,1)||
  lpad(a.section,2)||' '||
  lpad(a.record_id,1)||' '||
  b.formula_cd
from pur a, product b
where year=20YY
and a.prodno = b.prodno
and (a.site_code between 151 and 61501
or a.site_code in (10, 30, 40, 90, 91,63000,65021,
66000,67000,67001,67002,67003,68009,72000))
and a.record_id not in ('C','G','2')
and b.formula_cd not in ('D0','F0','I0','T0','U0')
;
spool off
```

2. nonspatial_YY.sql

```
set feedback off
set pagesize 0
set linesize 120
spool nonspatial_YY.dat
select
  lpad(a.use_no,8)||' '||
  lpad(a.prodno,7)||' '||
  lpad(nvl((to_char(a.applic_dt,'MMDDYYYY')),99999999),8)||' '||
  lpad(nvl(a.lbs_prd_used,0),14)||' '||
  lpad(nvl(a.acre_treated,0),10)||' '||
  lpad(nvl(a.unit_treated,0),1)||' '||
  lpad(nvl(a.site_code,0),6)||' '||
  lpad(nvl(a.county_cd,0),2)||' '||'99A99A99A99'||' '||
  lpad(nvl(a.record_id,0),1)||' '||
  b.formula_cd
from pur a, product b
where year=20YY
and a.prodno = b.prodno
and (a.site_code in (10, 30, 40, 90, 91,63000,65021,
66000,67000,67001,67002,67003,68009,72000)
or a.site_code between 151 and 61501)
and record_id in ('C','G','2')
and b.formula_cd not in ('D0','F0','I0','T0','U0')
;

spool off
```

3. outlierYY.sql

```
set feedback off
set pagesize 0
set linesize 102
spool outlierYY.dat
select
use_no||' ,'||
ai_a_1000_200||' ,'||
prd_u_50m||' ,'||nn4
from outlierYY
where ai_a_1000_200 = 'Y' or prd_u_50m = 'Y' or nn4 = 'Y'
;

spool off
```

4. prodsiteYY.sql

```
set feedback off
set pagesize 0
set linesize 13
spool prodsite00.dat
select
 lpad(PRODNO,6)||
 lpad(SITE_CODE,6)
from prod_site
;

spool off
```

5. typepest.sql

```
set feedback off
set pagesize 0
set linesize 9
spool typepest00.dat
select prodno||','||typepest_cd
from prod_type_pesticide
;
spool off
```

6. short_product_tilde.sql

```
set feedback off
set pagesize 0
set linesize 227
spool short_product_tilde.dat
select
PRODNO||' ~'||
substr(PRODUCT_NAME,1,50)||' ~'||
SHOW_REGNO||' ~'||FORMULA_CD
from product
;
spool off
```

7. product_tilde.sql

```
set feedback off
set pagesize 0
set linesize 290
spool product_tilde.dat
select
lpad(product.PRODNO,6)||'~'||
lpad(product.MFG_FIRMNO,7)||'~'||
 lpad(product.REG_FIRMNO,7)||'~'||
 lpad(product.LABEL_SEQ_NO,5)||'~'||
 lpad(product.REVISION_NO,2)||'~'||
 lpad(product.FUT_FIRMNO,7)||'~'||
 lpad(product.PRODSTAT_IND,1)||'~'||'"'||
 rpad(product.PRODUCT_NAME,100)||'"'||'~'||
 lpad(product.SHOW_REGNO,24)||'~'||
 nvl(AER_GRND_IND,' ')||'~'||
 nvl(AGRICCOM_SW,' ')||'~'||
 nvl(CONFID_SW,' ')||'~'||
 lpad(nvl(product.DENSITY,9999999),7)||'~'||
 rpad(nvl(product.FORMULA_CD,' '),2)||'~'||
 rpad(nvl(to_char(FULL_EXP_DT,'MMDDYYYY'),' '),8)||'~'||
 rpad(nvl(to_char(FULL_ISS_DT,'MMDDYYYY'),' '),8)||'~'||
 nvl(FUMIGANT_SW,' ')||'~'||
 nvl(GEN_PEST_IND,' ')||'~'||
 rpad(nvl(to_char(LASTUP_DT,'MMDDYYYY'),' '),8)||'~'||
 nvl(MFG_REF_SW,' ')||'~'||
 rpad(nvl(to_char(PROD_INAC_DT,'MMDDYYYY'),' '),8)||'~'||
 rpad(nvl(to_char(REG_DT,'MMDDYYYY'),' '),8)||'~'||
 nvl(REG_TYPE_IND,' ')||'~'||
 nvl(RODENT_SW,' ')||'~'||
 nvl(substr(to_char(SIGNLWRD_IND,'0'),2),' ')||'~'||
 nvl(SOILAPPL_SW,' ')||'~'||
 nvl(SPECGRAV_SW,' ')||'~'||
 lpad(nvl(substr(to_char(product.SPEC_GRAVITY,'99.9999'),2),' '),7)||'~'||
 nvl(CONDREG_SW,' ')
from product
;
spool off
```

8. special_AIs_for_preprocessing.sql

```
set feedback off
set pagesize 0
set linesize 225
spool special_AIs_preprocessing.dat
  select
      (b.prodno)||'~'||
      a.prodchem_pct||'~'||
      a.chem_code||'~'||
      substr(b.product_name,1,25)||'~'||
      lpad(b.show_regno,24)||'~'||
      substr(c.chemname,100)
      from prod_chem a,product b,chemical c
      where a.chem_code in (385,573,616,136,560,765,
      2045,536,2207,2267,131,618,561,539)
      and a.prodno=b.prodno
      and a.chem_code=c.chem_code
;
spool off
```

9. prodchem_tilde.sql

```
set feedback off
set pagesize 0
set linesize 30
spool prodchem_tilde.dat
select
 PRODNO||' ~'||
 CHEM_CODE||' ~'||
 CHEMSTAT_SW||' ~'||
 PRODCHEM_PCT
from prod_chem
where chem_code !=0
order by prodno
;
spool off
```

10. mebr.sql

```
set feedback off
set pagesize 0
set linesize 20
spool mebr.dat
select prodno||','||
       prodchem_pct
from prod_chem
where chem_code = 385
;
spool off
```

APPENDIX 3

SAS Pre-Processing Programs

EXCLUDED_PRODUCTS.SAS

```
/*April 3, 2002. This program identifies products that are (1) ONLY registered
as technicals, OR (2) ONLY registered as (adjuvants and/or fertilizers). These
are excluded from the VOC inventory calculations and this program's outfile is
used to remove these excluded compounds from the main VOC program VOC.SAS.

I. Technicals
These are products that ONLY registered for sitecodes 99000-99999. The input
file here comes from the Label database table prod_site */


FILENAME prodsite 'E:\00voc\prodsite00.dat';
DATA prodsite;
      INFILE prodsite;
      INPUT prodno 1-6 sitecode 7-12;

DATA nontechsites;
      SET prodsite;
      IF (sitecode<99000 or sitecode>99999);

proc sort; by prodno;

DATA techsites;
      SET prodsite;
      IF (sitecode>98999 and sitecode<100000);

proc sort;by prodno;

DATA techprods;
      MERGE nontechsites (in=aguses) techsites (in=techuses);
      BY prodno;
      IF techuses and NOT aguses;
      type= 'techncl';

PROC MEANS NOPRINT mean;
      VAR prodno;
      BY prodno;ID type;
      output out=techs mean=Nprod;

Data techdone;
      set techs;
      KEEP prodno type;


/*II. Adjuvants/Fertilizers
These are products ONLY registered with prod_site_pesticide.typepest_cd
ONLY = to A0 or J0.*/


FILENAME typepest 'e:\00voc\typepest00.dat';
DATA adjuvant;
      INFILE typepest  DLM="," MISSOVER;
      INPUT prodno typestcd $;

DATA nonadjvnt; set adjuvant;
if (typestcd NE 'A0' and typestcd NE 'J0');

proc sort; by prodno;

DATA adjvnt; set adjuvant;
if (typestcd= 'A0' or typestcd= 'J0');
proc sort; by prodno;
```

```
data adjprods; Merge nonadjvnt (in=pestuses) adjvnt (in=adjuses);
by prodno;
if NOT pestuses and adjuses;
type= 'AdjFert';

PROC MEANS NOPRINT mean;
      VAR prodno;
      BY prodno;ID type;
      output out=ads mean=Nprod;

Data adjuvants;
      set ads;
      KEEP prodno type;

proc append BASE=adjuvants DATA=techdone;

DATA NULL; SET adjuvants;
FILENAME f 'e:\00voc\data\excluded_prods.out';
FILE f;
PUT prodno 1-6 type 10-20;
proc print;
run;
```

```
/*SPECIAL_PROD_PREPROCESS.SAS
April 2, 2002 This program assigns EPs to products that contain
"special" AIs. These are products are not subject to the assignment of default
EPs based on formulation code [ref: Jan. 7, 2002 memorandum; Spurlock to
Sanders: Methodology for Determining VOC Emission Potentials of
Pesticide Products"] The special AIs and their product default EPs are:

1. MeBr, 1,3,-D (telone), and chloropicrin fumigants - EP=100
2. metam sodium - EP = prodchem_pct *0.566 [MITC basis]
3. sulfur prods > 50%AI and no other AIs - EP=0.6
4. petroleum oil (unclassified and mineral oil
      >80% AI and no other AI) - EP=1.53
5. Sodium Chlorate (>17.99% AI) – EP = 0
6. Inorganics - CO2, N2, CL2, SO2, NaOHCl, Vikane
      EP=0
*/

FILENAME specprod 'h:\special_AIs_preprocessing.dat';
FILENAME prodchem 'h:\prodchem_tilde.dat';
FILENAME spec_out 'h:\special_prods.out';


DATA special;
LENGTH prdname $ 25 regno $ 24 ch_name $ 100;
INFILE specprod DLM='~';
INPUT prodno ch_pct chem_cd prdname $ regno $ ch_name $;
drop ch_name;
proc sort;by prodno;

data prodchem; INFILE prodchem DLM = '~';
INPUT prodno chem_cd stat $ ch_pct;
proc sort; by prodno;

data fumis;set special;
if (chem_cd=385 or chem_cd=573 or chem_cd=136);
EP=100;

data metam;set special;if chem_cd=616;
EP=0.566*ch_pct;


data chlorate;set special;if chem_cd=536;if ch_pct>17.99;
EP=0;

data inorg; set special;
if (chem_cd =2207 or chem_cd=2267 or chem_cd=131
      or chem_cd=618 or chem_cd=561 or chem_cd=539);
      EP=0;

data check_1;merge special (in=specprod) prodchem;
by prodno;if specprod;

proc means N noprint;
var chem_cd; by prodno;
ID prdname regno ch_pct chem_cd;
output out=number N=num_AIs;

data sulfur; set number; if num_AIs<2;if (chem_cd=560);
if ch_pct>49.999; drop _TYPE_ _FREQ_ num_AIs;
EP=0.6;

data oils; set number; if num_AIs<2; if (chem_cd=765
or chem_cd=2045);if ch_pct>79.999;
drop _TYPE_ _FREQ_ num_AIs;
```

```
EP=1.53;

proc append base=inorg data =fumis;
proc append base=inorg data =metam;
proc append base=inorg data =sulfur;
proc append base=inorg data =oils;
proc append base=inorg data=chlorate;

proc sort;by chem_cd ch_pct;

DATA _NULL_;
SET inorg;
FILE spec_out;
PUT prodno 1-8 prdname $ 10-34 regno $ 40-64 chem_cd 68-75 ch_pct 80-90
EP 95-102 .2;

proc print;


run;
```

LINK_PRODCHEM_PRODUCT_TABLE_2GET_MAXCHEMPRODPCT.SAS

```
/*April 4, 2002 This program links the prodchem and product
tables and manipulates the data to create maxchem_prodname
_regno_formcd.out which can then be imported into EXCEL. The
MAXCHEM is that AI present at the maximum chem_pct. When more
than one are present at the max chem_pct both are put out by
this program so that further processing must be done in
EXCEL if desired. This program also creates an output file of
products that contain AB2588 toxic air pollutants.

input files:*/
FILENAME product 'h:\short_product_tilde.dat';
FILENAME prodchem 'h:\prodchem_tilde.dat';

FILENAME out 'h:\max_chem_prod_name_regno_formcd.out';
FILENAME toxchems 'h:\arb_toxic_chems.out';

data prodchem; INFILE prodchem DLM = '~';
INPUT prodno chem_cd stat $ pct;
proc sort; by prodno;

DATA product;
LENGTH prdname $ 50 regno $ 24;
INFILE product DLM='~';
INPUT prodno prdname $ regno $ form_cd $;
proc sort;by prodno;

DATA join;
MERGE product prodchem (in=inchem);
BY prodno;IF inchem;


PROC MEANS MAX NOPRINT;
VAR pct;by prodno;
output out=maxchem max=maxpct;
proc sort;by prodno;


DATA join2;merge maxchem join; by prodno;

DATA join3; set join2;
if maxpct=pct then flag=1;else flag=0;


Data maxchem2;set join3;
if flag=1;
drop pct _FREQ_ stat _TYPE_;
proc sort;by prodno maxpct;

DATA _NULL_;
SET maxchem2;

FILE out;
PUT prodno 1-8 prdname $ 10-59 regno $ 62-86 form_cd 90-92 chem_cd 94-100
maxpct 102-110 .4;

DATA tox; length toxchem $ 22; set maxchem2;
IF (chem_cd=3 or chem_cd=597 or chem_cd=573 or chem_cd=385 or
chem_cd=136);
IF chem_cd=3 then toxchem='acrolein';
if chem_cd=597 then toxchem='trifluralin';
if chem_cd=573 then toxchem='1,3-dichloropropene';
if chem_cd=385 then toxchem='methyl bromide';
```

```
if chem_cd=136 then toxchem='chloropicrin';

DATA _NULL_; set tox;
Keep prodno toxchem;
file toxchems;
PUT prodno 1-8 toxchem 13-36;

proc print; run;
```

APPENDIX 4

Main VOC Program Listing:

VOC_*YY*.SAS

```
    /*************************************************************************
*************************************************************************
MAY 2002 VERSION OF PUR VOC PROCESSING PROGRAM "VOC_YY.SAS"

==>> NOTE: "YY" in each filename corresponds to the year (e.g. 90, 91,....)

This program calculates estimated VOC emissions from annual [year = 'YY'] PUR
data. File output consists of:

YY_VOC_outlier.out
      [a list of PUR use_no that are purged before VOC calculation
      because they are probable outliers.]

YY_all_excluded.out
      [a list of products that are excluded because they are:
            only adjuvants,
            only (adjuvants + fertilizers),
            only technicals, or
            are not ozone formers (e g. are inorganic: sodium hypochlorite,
            CO2, N2, sulfuryl fluoride, etc.]

YY_NoEP.out
      [a list of those products that do not have an Emission Potential (EP)
      in the EP database file. These products - including so-called "special AI"
      products - are assigned default EP values.]

YY_ARB_VOC.out
      The final ARB YY output file - contains about 2 Million records

YY_DPR_VOC.out
      A *sometimes* more useful compact output file, with emissions
      summarized by prodno and administrative area - contains about
      350 - 400 thousand files

*************************************************************************
*************************************************************************
*************************************************************************
*************************************************************************
SECTION I
FILE REFERENCES
*****************************************************************
A. Input files
*****************************************************************/
FILENAME excluded 'e:\00voc\data\excluded_prods.out';
FILENAME epdata 'e:\00voc\data\ep_voc00d.csv';
FILENAME defEP 'e:\00voc\data\EP_defaults.csv';
FILENAME abadnaa1 'e:\00VOC\nospatial_admin1.csv';
FILENAME abadnaa2 'e:\00VOC\nonspatial_admin2.csv';
FILENAME abadnaa3 'e:\00VOC\nonspatial_admin3.csv';
FILENAME abadnaa4 'e:\00VOC\nonspatial_admin4.csv';
FILENAME abadnaa5 'e:\00VOC\nonspatial_admin5.csv';
FILENAME abadnaa6 'e:\00VOC\nonspatial_admin6.csv';
FILENAME abadnaa 'e:\00VOC\VOC_TABLES\STATEWIDE.csv';
FILENAME toxics 'e:\00voc\data\arb_toxic_chems.out';
FILENAME mebrdat 'e:\00voc\data\mebr.dat';
FILENAME latlong 'e:\00voc\lat_long.csv';
FILENAME spec_EP 'e:\00voc\special_prods.out';
FILENAME outliers 'e:\00VOC\data\outlierYY.dat';
FILENAME nspatial 'h:\nonspatial_YYa.dat';
FILENAME spatial 'h:\spatial_YYa.dat';

/*************************************************************
B. Output files
```

```
*************************************************************/
FILENAME outly 'e:\00voc\YY_VOC_outlier.out';
FILENAME xclude 'e:\00voc\YY_all_excluded.out';
FILENAME noep 'e:\00voc\YY_NoEP.out';
FILENAME finalARB 'e:\00voc\YY_ARB_VOC.out';
FILENAME finalDPR 'e:\00voc\YY_DPR_VOC.out';
/*********************************************************
SECTION II -
      A. load spatial and nonspatial PUR data
      B. remove "excluded materials"
                  technicals, adjuvants, non-ozone formers
      C. remove outliers
      D. assign EPs
*************************************************************/

DATA nonspatial;LENGTH comtrs $ 11 form_cd $ 2;
    INFILE nspatial;
    INPUT use_no 1-8 prodno 10-16  date $ 18-25 lbs 27-40 acres 42-51
      units $ 53 site 55-60 co 62-63 comtrs $ 65-75 recordid $ 77 form_cd $ 78-
80;


DATA spatial;LENGTH comtrs $ 11 form_cd $ 2;
      INFILE spatial;
      INPUT use_no 1-8 prodno 10-16  date $ 18-25 lbs 27-40 acres 42-51
      units $ 53 site 55-60 co 62-63 comtrs $ 65-75 recordid $ 77 form_cd $ 78-
80;

            PROC APPEND BASE=spatial DATA=nonspatial;
            PROC SORT; BY prodno;

/*****************************************************************
      II.B.       Excluded material screen
                  flagX =1 means product is excluded
                  DATASET passed =VOC1
*****************************************************************/
DATA xcluded;
      INFILE excluded missover;
    INPUT prodno class $;
            PROC SORT; BY prodno;

DATA Xscreen;
      MERGE xcluded (in=notVOC) spatial (in=used);
      BY prodno;
      IF used;
      IF notVOC THEN flagX=1;ELSE flagX=0;

DATA Xout;
    SET Xscreen;
    IF flagX=1;
            PROC MEANS sum N noprint;
            VAR lbs;
            BY prodno;ID class;
            OUTPUT OUT=Xsum SUM=sumlbs N=Numapps;

DATA _NULL_;
      SET Xsum;
    FILE Xclude;
    PUT prodno 1-6 sumlbs 20-30 Numapps 32-40 class 42-60;


DATA VOC1;
     SET Xscreen;
```

```
     IF flagX=0;
     DROP flagX class;
           PROC sort; BY use_no;
/*********************************************************
        II.C. Outlier removal
                  OUTflag = 1 means record is outlier
                  DATASET passed =VOC2
*********************************************************/
DATA outliers;
     INFILE outliers DLM=',';
     INPUT use_no ai_200 $ prd_50 $ nn4 $;
      PROC SORT; BY use_no;

DATA outly;
     MERGE outliers (in=outlier) VOC1 (in=used);
     BY use_no;
       IF used;
     IF (outlier AND used) THEN OUTflag=1;
     ELSE OUTflag=0;

DATA _NULL_;SET outly;
     IF OUTflag=1 AND lbs NE .;
     FILE outly;
     PUT use_no 1-8 prodno 10-16  date $ 18-25 lbs 27-40 acres 42-51
           units $ 53 site 55-60 co 62-63 comtrs $ 65-75 recordid $ 77
          form_cd $ 78-80 ai_200 $ 82-84 prd_50 $ 86-88
                  nn4 $ 90-92;

DATA VOC2;
      SET outly;
      IF OUTflag=0;
      DROP OUTflag ai_200 prd_50 nn4;
           PROC SORT; BY PRODNO;

/*****************************************************************
II.D.       Emission Potentials (EP):
                  assign EPs. 3 types of EPs:
                  from  database
                              defaults
                              special products EP files
                              (sodium chlorate, metam, vikane CO2 etc.)
                  then calculate emission for each use_no
                  DATASET passed=VOC3
*****************************************************************/
DATA EPs;
    INFILE epdata firstobs=2 DLM=',' missover;
    INPUT prodno ep_rog ep_tog junko $ ep_methd;
      KEEP prodno ep_rog ep_tog ep_methd;
      PROC SORT; BY prodno;

DATA spec_EP;/*"Special" EP prods*/;
      INFILE spec_EP missover;
      INPUT prodno 1-8 specEP 95-102;
      proc sort; by prodno;

DATA def_EPs; LENGTH form_cd $ 2;/*default EPs*/
     INFILE defEP DLM=',';
     INPUT form_cd $ def_ep;
           PROC SORT;BY form_cd;

DATA addEPs;
    MERGE VOC2 (in=used) EPs (in=haveEP);
    BY prodno;
    IF used;
```

```
    IF haveEP THEN flagEP=1;
    ELSE flagEP=0;
      IF haveEP THEN flag2EP=1;
      ELSE flag2EP=0;
      Proc Sort; BY prodno;

/*The dataset have_EPs below has EPs assigned from EP database
      and is later merged with rest of records after they are
      assigned specials or defaults*/
DATA have_EPs;
      SET addEPs;
    IF flagEP=1;
      KEEP use_no prodno date lbs acres units site
                  co comtrs recordid form_cd ep_tog ep_rog flagEP ep_methd;

DATA special;
      MERGE special (in=specprod) addEPs (in=used);
      BY prodno;
      IF (used and flagEP=0);
      IF specprod THEN flag2EP=1;

/*The dataset spec_EP below has EPs assigned from special
      products [via Special_Prod_Preprocess.SAS]
      and is later merged with rest of records after needed
      defaults are assigned*/
DATA spec_EP;
      SET special;
      IF flag2EP=0;
                  ep_tog=specEP;
                  ep_rog=specEP;
                  ep_methd= 3;
       KEEP use_no prodno date lbs acres units site
                   co comtrs recordid form_cd ep_tog ep_rog flagEP ep_methd;

DATA no_EP;
    SET special;
    IF (flag2EP=0);
            PROC SORT; BY form_cd;

DATA defaults_2_noEP;
       MERGE no_EP (in=without) def_EPs;
     BY form_cd;
     IF without;
            ep_tog=def_ep;
                  ep_rog=def_ep;
                  ep_methd= 3;
       KEEP use_no prodno date lbs acres units site
                   co comtrs recordid form_cd ep_tog ep_rog flagEP ep_methd;
                  PROC SORT; BY prodno;

PROC MEANS MEAN SUM NOPRINT;
      VAR lbs ep_tog;
      BY prodno;ID form_cd ep_methd;
      OUTPUT out=NO_EPs MEAN=TRASH1 default SUM=sumlbs TRASH2;

DATA _NULL_; SET NO_EPs;
      DROP TRASH1 TRASH2;
      FILE noep;
      PUT prodno 1-8 form_cd 10-13 default 17-22 sumlbs 25-38;


PROC APPEND BASE=have_EPs DATA=defaults_2_noEP;
PROC APPEND BASE=have_EPs DATA=spec_EP;
```

```
DATA VOC3;
      SET have_EPs;
            tog_emit=ep_tog*lbs*.01;
                  rog_emit=ep_rog*lbs*.01;
                  PROC SORT;BY use_no;

/*********************************************************
III. Process nonspatial PUR data:
            allocate VOC emissions in each nonspatial record
            to proper administrative areas (NAA, AB, AD)

            NOTE:
                        cslm = commercial structural & landscape maintenance
                        row  = rights-of-way
                        cfum = commodity fumigation

            DATASET passed = nonspatial_merge
*********************************************************/
DATA mtrs_chk;
      SET VOC3;
            MTRSflag=0;
      IF substr(comtrs,1,2)>0 and substr(comtrs,1,2)<59 and
            substr(comtrs,3,1) in ('H','M','S') and
            substr(comtrs,4,2)> 0 and substr(comtrs,4,2)<49 and
            substr(comtrs,6,1) in ('N','S') and
            substr(comtrs,7,2)> 0 and substr(comtrs,7,2)<48 and
            substr(comtrs,9,1) in ('E','W') and
            substr(comtrs,10,2)>0 and substr(comtrs,10,2)<37
      THEN MTRSflag=1;

DATA VOC4;
      SET mtrs_chk;
      IF MTRSflag =0 THEN comtrs = '99A99A99A99';

DATA nonspatial;
    SET VOC4;
    IF comtrs= '99A99A99A99';
    DROP MTRSflag;
    PROC SORT; BY co;

DATA abadnaa1;
    INFILE abadnaa1 firstobs=2 DLM=',';
    INPUT abadnaa $ AB ad naa co cslmfrac rowfrac cffrac dummy;
      DROP dummy;
      PROC SORT; BY co;
DATA nonspatial1;
      MERGE nonspatial (IN=indat) abadnaa1 (IN=indis);
      BY co;
        IF (indat and indis);


DATA abadnaa2;
      INFILE abadnaa2 firstobs=2 DLM=',';
      INPUT abadnaa $ AB ad naa co cslmfrac rowfrac cffrac dummy;
      DROP dummy;
        PROC SORT;BY co;
DATA nonspatial2;
      MERGE nonspatial (IN=indat) abadnaa2 (IN=indis);
      BY co;
        IF (indat and indis);

DATA abadnaa3;
      INFILE abadnaa3 firstobs=2 DLM=',';
      INPUT abadnaa $ AB ad naa co cslmfrac rowfrac cffrac dummy;
```

```
        DROP dummy;
        PROC SORT; BY co;
DATA nonspatial3;
      MERGE nonspatial (IN=indat) abadnaa3 (IN=indis);
      BY co;
        IF (indat and indis);


DATA abadnaa4;
      INFILE abadnaa4 firstobs=2 DLM=',';
      INPUT abadnaa $ AB ad naa co cslmfrac rowfrac cffrac dummy;
        DROP dummy;
      PROC SORT; BY co;
DATA nonspatial4;
      MERGE nonspatial (IN=indat) abadnaa4 (IN=indis);
      BY co;
        IF (indat and indis);


DATA abadnaa5;
      INFILE abadnaa5 firstobs=2 DLM=',';
      INPUT abadnaa $ AB ad naa co cslmfrac rowfrac cffrac dummy;
        DROP dummy;
      PROC SORT; BY co;
DATA nonspatial5;
      MERGE nonspatial (IN=indat) abadnaa5 (IN=indis);
      BY co;
        IF (indat and indis);


DATA abadnaa6;
      INFILE abadnaa6 firstobs=2 dlm=',';
      INPUT abadnaa $ AB ad naa co cslmfrac rowfrac cffrac dummy;
        DROP dummy;
      PROC SORT;BY co;
DATA nonspatial6;
      MERGE nonspatial (IN=indat) abadnaa6 (IN=indis);
      BY co;
        IF (indat and indis);



           PROC APPEND BASE=nonspatial1 DATA=nonspatial2;
           PROC APPEND BASE=nonspatial1 DATA=nonspatial3;
           PROC APPEND BASE=nonspatial1 DATA=nonspatial4;
           PROC APPEND BASE=nonspatial1 DATA=nonspatial5;
           PROC APPEND BASE=nonspatial1 DATA=nonspatial6;

DATA nonspatial7;
      SET nonspatial1;
            year=substr(date,5,4);
      mo=substr(date,1,2);
      IF (site=10 or site=61001 or site=63000 or site=72000
            or site=30 or site=67003)
      THEN cslmflag=1;ELSE cslmflag=0;
      IF site=40 THEN rowflag=1;ELSE rowflag=0;
      IF (site NE 61001 and site NE 63000 and site NE 72000
            and site NE 67003)
            AND site>89 THEN cfumflag=1;ELSE cfumflag=0;


DATA nonspatial8;
      SET nonspatial7;
      cslmrog=cslmflag*cslmfrac*rog_emit;
```

```
      cslmtog=cslmflag*cslmfrac*tog_emit;
      cslm_lbs=cslmflag*cslmfrac*lbs;
      rowrog=rowflag*rowfrac*rog_emit;
      rowtog=rowflag*rowfrac*tog_emit;
      row_lbs=rowflag*rowfrac*lbs;
      cfumrog=cfumflag*cffrac*rog_emit;
      cfumtog=cfumflag*cffrac*tog_emit;
      cfum_lbs=cfumflag*cffrac*lbs;
      rog_emit=cslmrog+rowrog+cfumrog;
      tog_emit=cslmtog+rowtog+cfumtog;
      sumlbs=cslm_lbs+row_lbs+cfum_lbs;
            PROC SORT; BY prodno;

DATA nonspatial_dat;
      SET nonspatial8;
            lbs=sumlbs;
      KEEP AB AD NAA Co prodno site mo year date comtrs lbs rog_emit tog_emit
            flagEP ep_methd;
/*********************************************************
IV. Process spatial PUR data:
            assign NAA, AD, AB designations to PUR records
            DATASET passed = spatial_merge
*********************************************************/

DATA spatial; SET VOC4;
      IF comtrs NE '99A99A99A99';
            PROC SORT; BY comtrs;

DATA districts;LENGTH comtrs $ 11;
      INFILE abadnaa DLM=',';
      INPUT comtrs $ AB ad naa co;
      DROP CO;
            PROC SORT;BY comtrs;

DATA spatial1;
      MERGE spatial (IN=used) districts;
      BY comtrs;
      IF used;
            mebr=0;
            year=substr(date,5,4);
            mo=substr(date,1,2);
/*********************************************************
V. PREPARE OUTPUT
            merge spatial and nonspatial datasets
            sum emissions over administrative areas, prodnos,sites
            write output files
*********************************************************/

DATA join_dat;
      SET spatial1;
      KEEP AB AD NAA Co prodno site mo year date comtrs lbs rog_emit tog_emit
                  flagEP ep_methd;
            PROC APPEND BASE=join_dat DATA=nonspatial_dat;

PROC SORT;
      BY prodno site mo year AB AD NAA CO;

DATA ARBtoxic;LENGTH toxchem $ 225;
      INFILE toxics missover;
      INPUT prodno toxchem $;
      PROC SORT; BY prodno;

DATA add_tox_flag;
      MERGE ARBtoxic (in=tox) join_dat (in=used);
```

```
      BY prodno;
      IF used;

DATA mebr_prods;
      INFILE mebrdat;
      INPUT prodno;
      PROC SORT; BY prodno;

DATA add_mebr_flag;
      MERGE add_tox_flag (in=used) mebr_prods (in=mebr);
      IF used;
      IF mebr THEN mebrflag=1; ELSE mebrflag=0;
            PROC SORT; BY comtrs;

DATA latlong;length comtrs $ 11;
      INFILE latlong DLM=',' missover;
      INPUT comtrs$ ddlong ddlat;
      PROC sort; BY comtrs;

DATA done;
      MERGE latlong add_mebr_flag (in=indata);
      BY comtrs;
      IF indata;
      IF (site=10 or site=61001 or site=63000
            or site = 67003 or site=72000)
    THEN aguse=0;
    ELSE aguse=1;
            PROC SORT; BY prodno site mo year AB AD NAA CO;

DATA _NULL_;SET done;
      FILE finalARB;
      PUT AB 1-3 AD 5-7 NAA 9-11 Co 13-15 prodno 17-24 site 26-32
            mo 34-36 year 38-42 comtrs 44-56 aguse 58-60 lbs 62-72 .2
            rog_emit 74-86 .2 tog_emit 88-100 .2 toxchem 102-127 mebrflag 130-
132
            ddlat 135-150 .4 ddlong 155-170 .4 date 175-190;

PROC MEANS sum NOPRINT;
VAR rog_emit tog_emit lbs;
      BY prodno site mo year AB AD NAA CO;
      ID mebrflag toxchem aguse;
      OUTPUT OUT=finalDPR SUM=rog tog     sumlbs;

DATA _NULL_;SET finalDPR;
      FILE finalDPR;
      PUT AB 1-3 AD 5-7 NAA 9-11 Co 13-15 prodno 17-24 site 26-32
            mo 34-36 year 38-42 aguse 58-60 sumlbs 62-72 .2
            rog 74-86 .2 tog 88-100 .2 toxchem 102-127 mebrflag 130-132;

run;


/* BE SURE TO RUN EXTENSIVE DIAGNOSTICS ON OUTPUT FILES*****************
**************************************************************************/
```

Appendix 5

California's Pesticide Registration Numbering System

What's in a Registration Number?

DPR's product/label database contains information on only those products currently or formerly registered in California. The index contains information on approximately 30,000 products. The registration number for all products is in the format listed below where item number 4 is described. To get the best results from this query, enter the firm number and the product number or a portion thereof.

All pesticide products, if properly registered, are identified by a unique number which is required to be located on the front panel of the label. This "registration number" is composed of several components, each of which has a specific meaning. These individual components are separated by a "-". Those products registered in California, have an extra component which identifies individual brand name registrations.

```
   1         2         3
xxxxxxx - xxxxx - xxxxxxx
```

1. Firm or Establishment Number - This component identifies the company that is the primary registrant with the USEPA. The number may be as long as seven digits, or as short as 1. California assigns their own unique firm numbers to companies that register products which are not required to be registered by USEPA.

2. Product or Label Number - This five or less digit number is generally assigned sequentially to each company's individual product as it is registered with the USEPA.

3. Distributor or Sub-Registrant's Number - This number identifies any company that is marketing a product owned by another company, generally the primary registrant. This type of registration is called a distributor registration or a sub-registration and is allowed under existing agreement certified by the USEPA. For a product registered in California, this number represents the company that holds the license for pesticide registration within the state.

```
                      4
xxxxxxx - xxxxx - xx - xxxxxxx
```

4. California Revision Code - California requires companies to register and license individual brand names. The revision code, a sequence of (2) alphabetic letters, creates a unique identifier for each product. A single product may have many brand names registered within the state. Unique revision codes assigned to each one allows for identification of the specific brand name in question. These alphabetic letters may or may not appear on actual product labeling.

| DPR HOME | DATABASES | Last modified: Thursday, November 20, 1997