

# Chapter 12. Compiling and Performance

## Contents

- [Introduction](#)
  - [Compiling SMOKE for UNIX](#)
  - [SMOKE Performance](#)
    - [SGI configuration](#)
    - [Sun configuration](#)
- 

## Introduction

The first part of this chapter provides instructions on how to compile SMOKE for UNIX platforms other than those supported by MCNC. The SMOKE compile scripts are configured to use a UNIX c-shell environment, so they will not be useful for creating Windows-based SMOKE files. SMOKE will be supported by MCNC for Windows-NT/2000 in the future, but is not supported at this time. For users with operating systems supported by MCNC, users need only to download the SMOKE executables available from the MCNC-Environmental Programs registration page, accessible at <http://envpro.ncsc.org/products> (then click on “Register”).

The second part of this chapter contains information about performance for the SMOKE test cases provided in the SMOKE stand-alone version in the \$SCRIPTS/run directory. More information on these scripts is available in [Chapter 10: Using SMOKE: Scripts, Models-3, and EDSS](#).

## Compiling SMOKE for UNIX

In order to compile SMOKE, the first step is installing SMOKE on your system. Installation instructions are provided on-line when you download SMOKE, from the web site [http://envpro.ncsc.org/EDSS/edss\\_register/install\\_smoke.html](http://envpro.ncsc.org/EDSS/edss_register/install_smoke.html).

Prior to SMOKE compilation, users must take the following steps:

- Install SMOKE as described on the web site
- Install the I/O API library (libioapi.a) and NetCDF library (libnetcdf.a) in the \$SMK\_BIN directory. This can also be accomplished with a symbolic link to the libraries if they are located elsewhere. The libioapi.a and libnetcdf.a files available from the MCNC web site are inconsistent with the ones provided with the SMOKE binary tar files, because those on the web site were compiled with older compilers (FORTRAN 77) that

are not compatible with the current FORTRAN 90 compilers needed to compile SMOKE executables. Therefore, users may have to compile the I/O API and NetCDF libraries on their systems from scratch. See the I/O API documentation <http://envpro.ncsc.org/products/ioapi/H.AVAIL.html> for more information on availability and compiling of the latest I/O API.

- If needed, update the file `$EDSS_ROOT/scripts/platform` to have the compiler options for your operating system. To decide whether you need to update this file, enter the following command from the UNIX prompt after having setup for EDSS (i.e., installed the EDSS Root and logged back into UNIX):

```
echo $EDSS_OS
```

If the value that comes back from this command is not listed as a “case” in the platform file in lines 41 – 60, then you need to update the platform file to have your case in lines 41 – 60, and you need to update for your case in the section comprised of lines 64 – 101. You will need to determine which compiler options are needed on your system for maximum optimization (like the `-O3` option), if any options are needed for portable executables (like `-static`), or if any options are needed for extra memory allocation (like `-bmaxdata:0X20000000` on AIX systems).

- If needed, update the file `$EDSS_SUBSYS/edss_tools/setup/sysflags` to have the compiler options for your operating system. See the instructions above for “platform” updates to determine if you need this. All of the same cases available in the platform file are also available in the `$EDSS_SUBSYS/edss_tools/setup/sysflags` file.
- If needed, update the file `$SMKROOT/assigns/sysflags` to have the compiler options for your operating system. See the instructions above for “platform” updates to determine if you need this. All of the same cases available in the platform file are also available in the `$SMKROOT/assigns/sysflags` file.

Once the appropriate libraries and compile options have been installed (as just described above), then you are read to use the SMOKE compile scripts, which are quite simple to use. To compile SMOKE, you must then:

- Source an Assigns file to ensure that your system is configured properly for installation
- Change directories to `$SMKROOT/scripts/make`
- Optionally edit the header of the Makeall file to turn off building one or more libraries needed for SMOKE (libedsstools, libsmoke, or libmo5b). You can only turn off these options if these libraries have been previously compiled for your system.
- At the UNIX prompt, type:

```
Makeall
```

The only errors that are acceptable when you use this script are errors after the “/bin/rm”

commands. The “/bin/rm” commands will fail if there are no object files (\*.o files) in your SMOKE source code directories, which will be the case if you have not previously compiled SMOKE.

## SMOKE Performance

The tables below provide SMOKE “user” CPU times for the SMOKE programs run using the scripts provided with the SMOKE installation. The “user” CPU times are provided here because these are least influenced by issues such as disk speed and network configurations (such as disks mounted on a remote computer). These CPU times are comparable to wall-clock times when disk storage speeds are high and local disks are used by the system.

Please note that the “net96, baseA” case is the entire 1996 National Emissions Inventory (NEI) for a 36-km grid over the entire US. The “m3demo” case is an Eastern U.S. inventory with a 36-km grid over only the Eastern U.S. Mobile emissions for the net96 case use emissions only (no VMT used in the SMOKE processing), whereas the m3demo case uses VMT for mobile emissions and computes emission factors with Mobile5b based on gridded, hourly temperatures.

## SGI Configuration

The SGI configuration that we used for testing was a 250 MHZ MIPS R10000 Processor (Chip Revision 3.4), with 2 GB of system memory.

	CPU "user" times	
	net96, baseA	m3demo
<b>Area</b>		
Smkinven, 7 emis	663.4	247.3 (sec)
Grdmat	54.2	15.7 (sec)
Spcmat	67.5	29.7 (sec)
Temporal, 25 hours	110.8	60.5 (sec)
Smkmerge, 25 hrs	176	48.4 (sec)
Smkreport "state"	562.7	208.3 (sec)
Cntlmat, projections	n/a	58 (sec)
Grwinven	n/a	156.2 (sec)
<b>Total:</b>	27.2	13.7 (min)
<b>Biogenics</b>		
Rawbio, summer	12.3	7.8 (sec)
Rawbio, winter	12.1	n/a (sec)
Tmpbio, 1 day	2	0.5 (sec)
Smkmerge, 1 day	22	4 (sec)
<b>Total:</b>	0.8	0.2 (min)

## Mobile

Smkinven, 7 emis	353.3	n/a (sec)
Smkinven, 2 activ	n/a	41.5 (sec)
Grdmat	61	15 (sec)
Spcmat	105.1	51.1 (sec)
Premobl, 1 day	n/a	26.6 (sec)
Emisfac	n/a	544.8 (sec)
Temporal, 1 day	176.9	115.7 (sec)
Smkmerge, 1 day	84.3	66.7 (sec)
Smkreport "state.std"	416.4	171.4 (sec)
Cntlmat, projections	n/a	25.4 (sec)
Grwinven	n/a	18.5 (sec)
<b>Total:</b>	20.0	17.9 (min)

#### Point

Smkinven, 7 emis	947	300.3 (sec)
Grdmat	3.6	1.3 (sec)
Spcmat	690.7	284.3 (sec)
Elevpoint	n/a	14 (sec)
Temporal	148.7	81.4 (sec)
Laypoint	209.3	68.6 (sec)
Smkmerge, 3-d	327.9	79.1 (sec)
Smkmerge, UAM	n/a	n/a (sec)
Smkreport "state"	230.8	90.4 (sec)
Cntlmat, projections	n/a	161 (sec)
Grwinven	n/a	183.6 (sec)
<b>Total:</b>	42.6	21.1 (min)

#### All merge

Smkmerge, mole	13.0	3.6 (min)
Smkmerge, mass		3.2 (min)

## Sun Configuration

The Sun configuration that we used for testing was a Sun Ultra 5/10 UPA/PCI (UltraSPARC-IIi 440MHz) with 256 MB memory. The substantially higher CPU times than for the SGI case for some of the steps is likely due to needing more than 256 MB of memory (the SGI case had 2 GB of memory available). Performance information for the net96 case is not yet available.

CPU "user" times

**m3demo**

**Falcon**

**Area**

Smkinven, 7 emis	204 (sec)
Grdmat	14 (sec)
Spcmat	21 (sec)
Temporal, 25 hours	105 (sec)
Smkmerge, 25 hrs	166 (sec)
Smkreport "state"	691 (sec)
Cntlmat, projections	46 (sec)
Grwinven	179 (sec)
<b>Total:</b>	23.8 (min)

**Biogenics**

Rawbio, summer	5 (sec)
Rawbio, winter	n/a (sec)
Tmpbio, 1 day	2 (sec)
Smkmerge, 1 day	2 (sec)
<b>Total:</b>	0.2 (min)

**Mobile**

Smkinven, 7 emis	n/a (sec)
Smkinven, 2 actv	32 (sec)
Grdmat	15 (sec)
Spcmat	42 (sec)
Premobl, 1 day	26 (sec)
Emisfac	581 (sec)
Temporal, 1 day	171 (sec)
Smkmerge, 1 day	234 (sec)
Smkreport "state.std"	932 (sec)
Cntlmat, projections	18 (sec)
Grwinven	15 (sec)
<b>Total:</b>	34.4 (min)

**Point**

Smkinven, 7 emis	242 (sec)
Grdmat	1 (sec)
Spcmat	27 (sec)
Elevpoint	13 (sec)
Temporal	122 (sec)
Laypoint	102 (sec)
Smkmerge, 3-d	314 (sec)
Smkmerge, UAM	n/a (sec)
Smkreport "state"	178 (sec)

Cntlmat, projections	125(sec)
Grwinven	185(sec)
<b>Total:</b>	21.8(min)

**All merge**

Smkmerge, mole	11.6(min)
Smkmerge, mass	11.7(min)