3.0 Running ESPVS

3.1 Configuration

3.1.1 Binary Code

The ESPVS binary code can be maintained anywhere on the user's system. It is suggested, however, that it be placed in a *../bin* directory. An associated resource file called *ESPVS.res* also is required. This resource file typically resides in */usr/lib/x11/app-defaults* and is an X-Designer byproduct that is created when the files are compiled. If the resource file does not reside in */usr/lib/x11/app-defaults*, then ESPVS has access to the location of this file through the environment variable `XENVIRONMENT`, which can be set explicitly.

3.1.2 Environment Variables

ESPVS identifies directory and file locations by accessing environment variables and *.Apps_defaults* tokens. The following environment variables must point to the appropriate implementation file via a full path name and are required to run ESPVS successfully:

Environment Variable	Implementation File
`APPS_DEFAULTS`	*..../.Apps_defaults.sys*
`APPS_DEFAULTS_USER`	*..../.Apps_defaults.ahps*
`XENVIRONMENT`	*..../ESPVS.res**

*If resource file is not in */usr/lib/x11/app_defaults*

The following additional tokens are required to run ESPVS:

Name	Example Implementation Directory	Purpose
espvs_dir	*/projects/ahps/ESPVS*	Base directory for ESPVS input/output directories
espts_dir	*$(ofs_files)/$(ofs_level)/espfiles*	Directory of ESP output time series - verification files generated are placed in *espts_dir/verification*
ofs_fs5files	*$(ofs_files)/$(ofs_level)/fs5files*	
esp_dir	*/projects/ahps/esp*	New carryover files are written to *$(esp_dir)/files/carryover*

3.1.3 Configuration File

It is suggested that the user maintain a configuration file that can be implemented before running ESPVS. This configuration file would set appropriate environment variables and could override *.Apps_defaults*. Please see a UNIX reference for a discussion of the Korn shell (ksh) and the C shell (csh).

3.1.4 Input/Output Directory Structure

ESPVS dynamically creates temporary operational forecast system (OFS) input files. These files are written to the directory *ofs_input* directory if it exists. If not, ESPVS finds *ofs_dir* and writes to *ofs_dir/input*. A required script run during the generation of traces is found in *$(espvs_dir)/input*. This file is called *espvs_generate.sh*.

Standard output from OFS scripts are generated and written to directories determined from *.Apps_defaults* settings. Typical OFS output logs are written to *$(ofs_output)/$(LOGNAME)*.

3.2 Command Line Options

The following command line arguments are recognized:

`-batch [filename]`
Runs in batch mode using commands in file [**`filename`**].

`-d#,#`
Sets the debug level to # (screen, file).

`-h`
Prints brief help message.

`-nolog`
Disables the logfile.

`-s#,#`
Sets the status message level to # (screen, file).

`-v`
Displays version information.

`-w#,#`
Sets the warning message level to # (screen, file).

3.3 Batch Mode

ESPVS can be operated without the GUI or in batch mode through the use of a batch control file.

To run ESPVS in batch mode enter::

```
ESPVS -batch <filename>
```

where **`filename`** is the name of the batch control file - if the full path name to **`filename`** is not provided then ESPVS will look in the current working directory

3.3.1 Action Commands

The following action commands are available for the control file and are performed in the order that they appear in the batch mode file. Where applicable, they use the current values of the settings commands or defaults if no values have been set.

GenerateTraces

This option will read provided parametric information and generate the verification trace ensembles. All the following information is required:

```
Segment|ForecastGroup|CarryoverGroup=<SEGMENTID|FGNAME|CGName>
TimeZone=<TZ>
CurrentCarryoverDate=<MM/DD/YYYY:h>
SaveCarryoverDate=<MM/DD/YYYY:h>
AnalysisYears=<StartYear,<EndYear>
ForecastWindowStart=<MM/DD/YYYY>
ForecastWindowEnd=<MM/DD/YYYY>
```

LoadData(<SEGMENTName>;<Month>;<DD>;<DataType>)

This option loads all traces identified by segment name, data type, month and day of verification generation. These identifiers are components of the naming convention used in the generation schemes.

CalculateStatistics(<StartMonth>;<StartDay>;<StartYear>;<EndMonth>; <EndDay>;<EndYear>; VOLUME|MEAN|INST_MAX|INST_MIN|DAILY_MAX|DAILY_MIN)

This option calculates verification statistics over a start and end window specifying accumulation for the forecast variable. Data must be loaded prior to calculation of statistics. The following are valid accumulations :

VOLUME - Volume
MEAN - Mean instantaneous
INST_MAX - Instantaneous maximum
INST_MIN - Instantaneous minimum
DAILY_MAX - Mean daily maximum
DAILY_MIN - Mean daily minimum

GenerateTable(<filename>)

This option generates a file (**filename**) comprised of the statistics calculated in the CalculateStatistics batch command option. Data must be loaded and statistics calculated prior to generation of output results.