RunSetup Quick Reference

Version 4.0.5

Revision History
1/8/2010   Edited by AECOM Consult, Inc.
4/15/2010   Edited by RSG, Inc.

Syntax:

RunSetup [-flag] [control_file] [parameter]

Purpose:
1. Create control and batch processing files using master control files and keyword replacement procedures;
2. If a parameter is specified, the program will create multiple instances of the control files based on incrementing the run number and previous run number based on the parameter range. For example, parameter = "2..10" will process the master control files 9 times starting with run number 2 and previous run number 1 and ending with run number 10 and previous run number 9; and
3. Create control and batch processing files using a series of RunSetup control file.

Required Keys

MASTER_CONTROL_FILE (OR _#) (13)	[project_directory]*filename* (4)

Optional Keys

TITLE	Text
REPORT_FILE	*Filename*
REPORT_FLAG	FALSE {true/false/yes/no/1/0}
MAX_WARNING_MESSAGES	100,000
MAX_WARNING_EXIT_FLAG	TRUE {true/false/yes/no/1/0}
PROJECT_DIRECTORY	*Pathname* (@PROJECT@)
PROGRAM_DIRECTORY	*Pathname* (@BIN@) (12)
CONTROL_DIRECTORY	[project_directory]control (1)
NETWORK_DIRECTORY	[project_directory]network (1) (@NETWORK@)
BATCH_DIRECTORY	[control_directory] (1)
BATCH_NAME	[batch_directory]/[file prefix (14)]*Name*[.bat or .sh]
PROGRAM_FLAGS	NULL { -q, -h, -k, -p, -n, -b }
EXIT_CHECK	batch command (10)
FILE_FORMAT	TAB_DELIMITED (2)
MODEL_NAME	NULL (@MODEL@)
ALTERNATIVE_NAME	NULL (@ALT@)
ANALYSIS_YEAR	NULL (@YEAR@)

MASTER_CONTROL_FILE	[project_directory]*filename* (4)
DIRECTORY	*Pathname* (@DIR@)
DESCRIPTION	NULL (batch file comment line) (11)
PROGRAMS	script[group number] (5)
CONTROL_NAME	[file prefix (14)].*Name*.ctl (6)
RUN_NAME	[parameter] or NULL (@RUN@)
PREVIOUS_RUN_NAME	[parameter-1] or NULL (@PREVIOUS@) (3)
PARAMETER	NULL (@PARAM@)
PARAMETER1	NULL (@PARAM1@)
PARAMETER2	NULL (@PARAM2@)
PARAMETER3	NULL (@PARAM3@)
APPEND_KEY	NULL (7)
APPEND1_KEY	NULL (7)
APPEND2_KEY	NULL (7)
APPEND3_KEY	NULL (7)
COMMENT_FLAG	FALSE {true/false/yes/no/1/0} (@COMMENT@) (8)
COMMENT1_FLAG	FALSE {true/false/yes/no/1/0} (@COMMENT1@) (8)
COMMENT2_FLAG	FALSE {true/false/yes/no/1/0} (@COMMENT2@) (8)
COMMENT3_FLAG	FALSE {true/false/yes/no/1/0} (@COMMENT3@) (8)
MASTER_CONTROL_FILE_#	[project_directory]*filename* (4)
DIRECTORY_#	*Pathname* (@DIR@)
DESCRIPTION_#	NULL (batch file comment line) (11)
PROGRAMS_#	script[group number] (5)
CONTROL_NAME_#	[file prefix (14)].*Name*.ctl (6)
RUN_NAME_#	[parameter] or NULL (@RUN@)
PREVIOUS_RUN_NAME_#	[parameter-1] or NULL (@PREVIOUS@) (3)
PARAMETER_#	NULL (@PARAM@)
PARAMETER1_#	NULL (@PARAM1@)
PARAMETER2_#	NULL (@PARAM2@)
PARAMETER3_#	NULL (@PARAM3@)
APPEND_KEY_#	NULL (7)
APPEND1_KEY_#	NULL (7)
APPEND2_KEY_#	NULL (7)
APPEND3_KEY_#	NULL (7)
COMMENT_FLAG_#	FALSE {true/false/yes/no/1/0} (@COMMENT@) (8)
COMMENT1_FLAG_#	FALSE {true/false/yes/no/1/0} (@COMMENT1@) (8)
COMMENT2_FLAG_#	FALSE {true/false/yes/no/1/0} (@COMMENT2@) (8)
COMMENT3_FLAG_#	FALSE {true/false/yes/no/1/0} (@COMMENT3@) (8)
NUM_SPLIT_FILES	NULL (@SPLITS@)
SETUP_CONTROL_FILE_# (13)	*filename.ctl*

TRANSIMS
Open-Source

SETUP_PARAMETER_#	NULL {start_run..end_run}

Notes

1	If the *Pathname* does not include a directory symbol (i.e., / or \), the project directory is added to the beginning of the path name
2	{VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, CSV_DELIMITED, DBASE, LANL, SQLITE3}
3	If a key value includes @PREVIOUS@ and the PREVIOUS_RUN_NAME is NULL, the whole key value is replaced with NULL
4	At least one Master Control File must be specified. The master control file is copied to the output control directory using the control name with all @@ keywords replaced by the corresponding default or group specific value. The keywords include: @PROJECT@, @BIN@, @NETWORK@, @DIR@, @FORMAT@, @MODEL@, @ALT@, @YEAR@, @RUN@, @PREVIOUS@, @GROUP@, @PARAM@, @PARAM1@, @PARAM2@, @PARAM3@, @COMMENT@, @COMMENT1@, @COMMENT2@, @COMMENT3@, @SPLITS@
5	For each program in the program list, a separate call using the current control file is added to the batch file. The programs may be executable program names or batch processing scripts.
6	If the control name is not provided, the program name is used.
7	The append keys are added to the end of each control file. These keys may include @@ keywords that are overridden based on the current value of each keyword.
8	If the comment flag is true, the @COMMENT@ keywords are replaced with "##". Otherwise the keywords are replaced with "".
9	A new control group is created for each master control file key included in the program control file. All keys with the same group number (@GROUP@) override the corresponding default key for that group.
10	In a Windows executable the default exit check is "if %ERRORLEVEL% == 1 exit 1". In a Linux executable the default exit check is "if ["$?" == "1"]; then exit 1; fi"
11	In a Windows executable the description text is preceded with "rem ". In Linux the text is preceded with "# ".

12	The program path is added to the end of the system path variable. In a Windows executable the syntax is path=@PATH@;program directory. In a Linux executable the syntax is:
	    #!/bin/sh
	    export PATH=$PATH:program direcotry

13	For applications with setup control files, the master control file key is not required. A setup control file is the filename of another RunSetup control file. When setup control file keys are provided, the program processes each file in # order and adds the batch commands to a single output batch file.
14	The control and batch file prefix is constructed using the provided keywords in the following way: [@RUN@][.@ALT@][.@YEAR@][.@MODEL@].*filename*

TRANSIMS Open-Source