

APPENDIX I

<u>I.1</u>	<u>CASC2D-SED Sub-routines List</u>	I-2
<u>I.2</u>	<u>CASC2D-SED Code</u>	I-4
<u>I.3</u>	<u>CASC2D-SED Input/Output Files</u>	I-82
<u>I.3.1</u>	<u>Input Files</u>	I-82
<u>I.3.2</u>	<u>Output Files</u>	I-83
<u>I.4</u>	<u>Time-Series Grids Display</u>	I-86
<u>I.4.1</u>	<u>AML programs</u>	I-86
<u>I.4.2</u>	<u>DS.aml parameter file example</u>	I-106

I.1 CASC2D-SED SUB-ROUTINES LIST

PROGRAM	CALLS	LOCATION
main()	ReadProjectFile(argv[])	ReadProjectFile.c
	ReadControlFile	ReadControlFile.c
	ReadGridHeaders()	ReadGrids.c
	GridsMemAlloc()	MemAlloc.c
	ReadInputGrids()	ReadGrids.c
	InitializeVars()	InitializeVars.c
	ReadChannlFile	ReadChannFile.c
	ChannToplg()	ChannToplg.c
	CompInitialVol()	CompVolumes.c
	OvrlDepth()	OvrlDepth.c
	ChannDepth()	ChannDepth.c
	OvrlRout()	OvrlRout.c
	ChannRout()	ChannRout.c
	RoutOutlet()	RoutOutlet.c
	Settling()	Settling.c
	RoutSedOut()	RoutSedOut.c
	SedStats()	SedStats.c
	WriteOutflow()	WriteOutflow.c
	WriteGrids()	WriteGrids.c
	Reset()	Reset.c
	CompFinalVol()	CompVolumes.c
	WriteSummFlow()	WriteSummFlow.c
	SedVolumes()	SedVolumes.c
	WriteSummSed()	WriteSummSed.c
GridsMemFree()	MemFree.c	
RunTime()	RunTime.c	
ReadControlFile()	floatMemAlloc2d()	MemAlloc.c
	intMemAlloc2d()	MemAlloc.c
GridsMemAlloc()	intMemAlloc2d()	MemAlloc.c
	floatMemAlloc2d()	MemAlloc.c
OvrlDepth()	intercept()	intercept.c
	rain()	rain.c
	infiltr()	infiltr.c
rain()	intercept()	intercept.c
ChannDepth()	NewChannDepth()	Fcts_Chann.c

PROGRAM	CALLS	LOCATION
OvrlRout()	ovrl	OvrlRout.c
ovrl()	RoutSedOvrl()	RoutSedOvrl.c
RoutSedOvrl()	TransferSed()	Fcts_sed.c
ChannRout()	chnchn	ChannRout.c
chnchn()	chnDischarge()	Fcts_Chann.c
	RoutSedChn()	RoutSedChn.c
RoutSedChn()	TransferSed()	Fcts_sed.c
RoutOutlet()	chnDischarge()	Fcts_Chann.c
SedStats()	RoutSedChn()	RoutSedChn.c
	RoutSedOvrl()	RoutSedOvrl.c
WriteGrids()	write2dTS()	WriteGrids.c
	write3dTS()	WriteGrids.c
GridsMemFree()	iMemFree2d()	MemFree.c
	fMemFree2d()	MemFree.c

I.2 CASC2D-SED CODE

<u>File</u>	<u>Page</u>
All.h	I-5
CASC2D-SED_sed.c	I-9
ReadProjectFile.c	I-12
ReadControlFile.c	I-16
ReadGrids.c	I-19
MemAlloc.c	I-22
InitializeVars.c	I-24
ChannToplg.c	I-28
CompVolumes.c	I-32
OvrlDepth.c	I-34
infiltr.c	I-37
intercept.c	I-38
rain.c	I-39
ChannDepth.c	I-40
OvrlRout.c	I-42
RoutSedOvrl.c	I-45
ChannRout.c	I-47
RoutSedChn.c	I-50
RoutOutlet.c	I-52
Settling.c	I-54
RoutSedOut.c	I-55
SedStats.c	I-56
WriteOutflows.c	I-58
WriteGrids.c	I-60
Reset.c	I-64
WriteSummFlow.c	I-65
SedVolumes.c	I-67
WriteSummSed.c	I-68
MemFree.c	I-74
RunTime.c	I-75
Fcts_chann.c	I-76
Fcts_misc.c	I-78
Fcts_sed.c	I-79
Fcts_stats.c	I-80

I.3 CASC2D-SED INPUT/OUTPUT FILES

Model input and output files are defined in a Project file. The project file name represents the argument of the CASC2D-SED program. The list of input/output files and their definition is found in Table I-1, Table I-2 and Table I-3 below. An example of a project file is found in Appendix III.

I.3.1 Input Files

Required CASC2D inputs and options are:

(1) Simulation control data: computational time step, simulation duration, and number of rows and columns and cell size of the grid.

(2) Precipitation data: precipitation duration and a) if uniform rainfall, the intensity or b) number of raingages, rainfall data time step, and rainfall gages UTM coordinates if distributed rainfall.

(3) Basin outlet characteristics: a) row, column and overland slope of the basin at the outlet and b) link characteristics if channel routing is selected: (link number, total nodes, type, width, depth, side slope, roughness and sinuosity)

(4) Soil parameters: number of different soil types, and a) if infiltration occurs: hydraulic conductivity, suction head, and moisture deficit and b) if erosion option is selected: percentages of sand, silt and clay in soil and USLE soil erodibility factor.

(5) Land use parameters: number of different soil uses, roughness coefficient and interception depth and, if erosion option is selected: USLE cover and management practice factors.

The input files are: the simulation control file (specifying parameters defined in the last 5 paragraphs), channel data file, and precipitation file (if distributed rainfall is

used). Input as raster ASCII maps are: DEM, soil index, land use index, watershed mask, and channel links and nodes (if channel routing is selected). The storage depth ASCII grid is an optional input grid.

Examples of input files can be found in Appendix III.

I.3.2 Output Files

CASC2D-SED creates the next output files:

(1) Summary file: contains information on the simulation parameters, channel topology inferred from the link and node maps and the final results of the hydrological and erosion components.

(2) Flow and sediment flow files: containing the water flow [m³/s] and sediment flow [m³/s] at the outlet. Results can optionally be written at other internal locations inside the watershed.

(3) Time-series grids: They are optional and represent simulation values at predetermined times. Output raster maps include: water depth, rainfall rates, accumulated infiltrated depth, total sediment volume, suspended sediment volume, and sediment flux and concentration by size fraction. Time series maps can be later used as input in an animated file.

Examples of CASC2D-SED output files can be found in Appendix III.

Table I-1 CASC2D-SED input and output text files

	Project name	File variable	Description
Input Text Files	CONTROL	control_file	Controls the simulation.
	PRECIP	rain_file	Precipitation data if gages are present [in/hr]
	CHANNEL	chn_file	Channel characteristics data for each link and node.
Output Text Files	SUMMARY	Summ_file	Simulation summary file
	DISCHARGE_OUT	dis_out_file	Hydrograph at outlet and, if selected, at internal locations [m ³ /s]
	SEDIMENT_OUT	sed_out_file	Sedigraph at outlet and, if selected, at internal locations [m ³ /s]

Table I-2 CASC2D-SED input grids

Project name	File variable	Variable Read / Displayed	Description
ELEVATION	elev_file	e[j][k]	Grid elevation at (j,k) [m]
SOIL	soil_file	isoil[j][k]	Soil type index at (j,k)
LANDUSE	iman_file	iman[j][k]	Land use / land cover index at (j,k)
MASK	shap_file	ishp[j][k]	Watershed and channel network definition at (j,k)
LINK	link_file	link[j][k]	Link number at (j,k)
NODE	node_file	node[j][k]	Node number at (j,k)
STORAGE_DEPTH	sdep_file	sdep[j][k]	Storage deposition at (j,k) [m]

Table I-3 CASC2D-SED time-series output grids

Project name	File variable	Variable Read / Displayed	Description
RAINFALL	rname	rint[j][k]	Rainfall rates at (j,k) [in/hr]
INF_DEPTH	vinfname	vinf[j][k]	Volume of infiltrated water at (j,k) [m]
WATER_DEPTH	dname	h[j][k] or hch[j][k]	Depth of water (channels and overland) at (j,k) [m]
SUS_SED	susname	totsus[j][k]	Total volume of suspended sediment at (j,k) [mm]
SED_CONC	cname	$\Sigma q_{ovs}[\text{SizeFr}][j][k]$ / (water volume)	Total suspended sediment concentration at (j,k) [m ³ /m ³]
EROS_DEP	erosdepname	totnetv[j][k]	1. Total net erosion at (j,k) [mm]
SAND_FLUX_OUT	sandfluxname	SedFluxOut[1][j][k]	Total sand flux out of (j,k) [m ³ /s]
SILT_FLUX_OUT	siltfluxname	SedFluxOut[2][j][k]	Total silt flux out of (j,k) [m ³ /s]
CLAY_FLUX_OUT	clayfluxname	SedFluxOut[3][j][k]	Total clay flux out of (j,k) [m ³ /s]
TOTAL_FLUX_OUT	totalfluxname	qss[j][k]/dt	Total sediment flux out of (j,k) [m ³ /s]
SAND_MFAC_OUT	sandMFACname	MFAC[1][j][k]	Max. flux-averaged concentration of sand out of (j,k) [m ³ /s / m ³ /s]
SILT_MFAC_OUT	siltMFACname	MFAC[2][j][k]	Max. flux-averaged concentration of silt out of (j,k) [m ³ /s / m ³ /s]
CLAY_MFAC_OUT	clayMFACname	MFAC[3][j][k]	Max. flux-averaged concentration of clay out of (j,k) [m ³ /s / m ³ /s]
TOTAL_MFAC_OUT	totalMFACname	MaxFluxCout[j][k]	Max. flux-averaged concentration of sediment out of (j,k) [m ³ /s / m ³ /s]

I.4 TIME-SERIES GRIDS DISPLAY

I.4.1 AML programs

```
/* #####
/* #####          DS.AML          #####
/* #####

/* =====
/* PURPOSE:
/* Animates a time series from Arc Info ASCII grids created by
/* the CASC2D-SED hydrological model.  The user has the possibility
/* of creating an animated MPEG file (movie) from the ArcPlot
/* display window by creating a screenshot of each display.
/* In general, this AML can be used to create a movie from any
/* time series of Arc Info ASCII grids if these ASCII grids have
/* the same Prefix (root) name and the Suffix (file extension) in
/* an ordered sequence of arabic numbers.
/*
/* =====
/* INPUT
/* 1. A time series of Arc Info ASCII grids:
/*   Example 1: one map per frame and n number of frames are shown
/*   depth.1; depth.2; ....; depth.n;
/*   Example 2: four maps per frame and n number of frames are shown
/*   depth.1; depth.2; ....; depth.n;
/*   erosion.1; erosion.2; ....erosion.n;
/*   flow.1; flow.2; ....; flow.n;
/*   rain.1; rain.2; ....; rain.n;
/* 2. A parameter file that will control the display of the grids or
/* map composition.  Optionally, this parameter file might not be
/* given but the user needs to enter data manually.
/*
/* =====
/* OUTPUT:
/* 1. A series of Arc Info grids (if user doesn't opt to kill them)
/*   after making the movie)
/* Example: User chooses to show every other grid in his input
/* series, starting in grid 6 and ending in grid 10:
/*   depth6; depth8; depth10
/* 2. An MPEG file (if user decides to create a movie using the
/*   generated grids)
/*
/* =====
/* USAGE:
/* Arc: &run ds <ParameterFile>
/*
/* =====
/* ARGUMENTS:
/* ParamFile: Name of the input data file.  If not specified,
/*   the user must enter the input data interactively.
/*
/* =====
```

```

/* VARIABLES:
/*   Globals:
/*   .CreateMovie: Index: 1: Movie is created; 0: not created
/*   .ElapsedTime: Elapsed time (secs) between CASC2D generated maps
/*   .EndFrame: Frame in which the display or movie will end
/*   .FrameStep: Number of frames to skip when displaying series
/*   .iter: Counter used to name multiple sequential images
/*   .MapTitleXX: Title for each of the displayed maps
/*   .MovieName: Name given to the final MPEG file (movie)
/*   .NameXX: Prefix (root) given to each of the thematic series
/*   .RMTXX: Remap table for each of the thematic series
/*   .SecsPause: Seconds to halt the series display
/*   .ShdSetXX: Shade set for each of the thematic series
/*   .StartFrame: Frame in which the display or movie will end
/*   .wPath: workspace location.  Enter full path if the workspace
/*           is in another location
/*
/* =====
/* CALLS:
/*   APdisplay.aml;
/*
/* =====
/* ROUTINES:
/*   GetUserInputData -- Prompts the user to enter input data
/*   GetFileInputData -- Input data is read from a file
/*   CreateGrids      --
/*       1. Creates the grids from the ArcInfo
/*           ASCII files if they don't exists.
/*           They will be named as: PrefixSufix
/*           Example: depth1; depth2; ....depthn;
/*       2. Writes grid stats. in text files
/*   DisplayGrids    -- Displays the series of grids in an Arc Plot
/*                       display window according to the specified
/*                       remap table and shadesets defined previously
/*                       by the user.
/*   Animate         -- Executes the MPEG encoder
/*   KillAllGrids    -- Created grids can be cleaned once the screen
/*                       shot has been taken if the user decides so.
/*   GetUserSpecs    -- Get the user's ArcPlot configuration
/*   PutUserSpecs    -- Restores the user's AP configuration
/*   bailout         -- bails of of aml
/* =====
/* NOTES:
/* Must have the Berkeley MPEG encoder in:
/*                                     $ARCHOME\bin\mpeg_encode.exe
/*
/* =====
/* HISTORY:
/* 9 Aug 1996 Original coding - Ian DeMerchant
/*   (as mpeg_encode.aml in ArcTools)
/* 15 Nov 2000 Major revision - Stephen Lead (slead@esriau.com.au)
/*   (as movie.aml)
/* 15 Jul 2001 Revision -- Rosalia Rojas (CSU)
/*   (as DisplaySeries.aml)
/* This AML has been rewritten for handling different options.
/* The movie can display 1, 2, 3, 4 or 6 maps per frame
/* User is able to define the remap table and color scheme

```

```

/* for each of the maps in the frame.
/* User is able to define a starting and ending frame and
/* show them in a predefined interval.
/* =====

&ARGS DSPParamFile

&IF [NULL %DSPParamFile%] &THEN
  &CALL GetUserInputData
&ELSE
  &CALL GetFileInputData
&CALL CheckInputErrors
&CALL CreateGrids

/* IN CASE THAT A MOVIE IS NOT TO BE CREATED, THE USER IS GIVEN THE
/* POSSIBILITY OF DISPLAYING THE GRIDS AS MANY TIMES AS (S)HE WISHES

&SV ShowSeriesAgain = .FALSE.

&do &until ^ %ShowSeriesAgain%
  &CALL DisplayGrids
  &IF %CreateMovie% = 0 &THEN
    &DO
      &TYPE
      &TYPE !!! Hello there !!!
      &SV ShowSeriesAgain = ~
      [QUERY 'Do you want to display again the time series' .FALSE.]

      /* Include the next line in case that you changed your mind
      /* after displaying the series (ex. you decide to kill grids)
      &IF %ShowSeriesAgain% &THEN &CALL GetFileInputData
      &TYPE
    &END
  &end

&IF %CreateMovie% = 1 &THEN ; &CALL Animate

&CALL KillAllGrids

&RETURN

/* #####
/* ##### ROUTINE GET_USER_INPUT_DATA #####
/* #####

&ROUTINE GetUserInputData

/* GET THE NUMBER OF MAPS PER FRAME */

&TYPE
&SV .MapsPerFrame = ~
      [RESPONSE 'How many maps per frame? 1,2,3,4 or 6?']
&TYPE

```

```

/* DEFINE WHICH FRAMES WILL BE SHOWN

&SV .StartFrame = [RESPONSE 'Start movie in frame number']
&TYPE

&SV .EndFrame = [RESPONSE 'End movie in frame number']
&TYPE

&SV .ElapsedTime = ~
                    [RESPONSE 'Elapsed show time between frames (sec.)']
&TYPE

&SV .FrameStep = [RESPONSE 'Show frame step']
&TYPE

&SV .SecsPause = [RESPONSE 'Seconds to pause frames display in AP']

/* Ask for the location of the folder where the ASCII grids are
&SV .wPath = [RESPONSE 'ASCII grids location (full path)']
&TYPE

/* GET THE NAMES OF THOSE MAPS

&DO MapNumber = 1 &TO %.MapsPerFrame%
  &sv .Name%MapNumber% = ~
    [RESPONSE 'Name (prefix) of map no. '%MapNumber%' :']
&END
&TYPE

/* ASK THE USER TO ENTER THE REMAP TABLE NAME FOR EACH OF
/* THE MAPS (S)HE WANTS TO DISPLAY

&DO MapNum = 1 &TO %.MapsPerFrame%
  &sv .RMT%MapNum% = ~
    [RESPONSE 'Remap table name for the '[value Name%MapNum%]' map']
&END
&TYPE

/* ASK THE USER TO ENTER THE SHADESET FILE NAME FOR EACH OF
/* THE MAPS (S)HE WANTS TO DISPLAY

&DO MapNo = 1 &TO %.MapsPerFrame%
  &sv .ShdSet%MapNo% = ~
    [RESPONSE 'Shadeset file name for the '[value Name%MapNo%]' map']
&END
&TYPE

/* ASK THE USER TO ENTER THE TITLE TO DISPLAY FOR EACH OF
/* THE MAPS (S)HE WANTS TO DISPLAY

&DO MapNo = 1 &TO %.MapsPerFrame%
  &sv .MapTitle%MapNo% = ~
    [RESPONSE 'Title to display for the '[value Name%MapNo%]' map']
&END
&TYPE

/* ASK WHETHER THE USER WANTS TO CLEAN THE WORKSPACE

```

```

/* AFTER THE MOVIE IS DONE.

&SV OrderToKill = ~
  [RESPONSE 'Kill grids after they are displayed (1:yes;0:no) ?']
&TYPE

/* SCARE THE USER IF (S)HE WANTS TO CREATE THE MPEG FILE
  &TYPE
  &TYPE WARNING !!!!
  &TYPE Be aware that creating a movie will take for about
  &TYPE a minute per frame. Make sure that this time series
  &TYPE is what you want to be displayed.
  &TYPE
  &TYPE
  &TYPE

&SV .CreateMovie = ~
  [RESPONSE 'would you like to create a movie (1:yes;0:no) ?']
&TYPE

&IF %CreateMovie% = 1 &THEN
  &SV .MovieName = [RESPONSE 'Movie name']
&TYPE

/*&SV .3Drendering = ~
/*  [RESPONSE 'would you like a 3-D representation (1:yes;0:no) ?']
/*&TYPE

/*&SV .SurfaceModel = ~
/*  [RESPONSE 'Name of the surface model (include path) ?']
/*&TYPE

/*&SV .Zscale = ~
/*  [RESPONSE 'Vertical scale (>1 exaggeration; <1 reduccion):']
/*&TYPE

&RETURN

/* #####
/* #####      ROUTINE GET_FILE_INPUT_DATA      #####
/* #####

&ROUTINE GetFileInputData

&SV AMLunit = [open %DSPParamFile% openstat -read]

/* First four lines are comments
&SV line1 = [read %AMLunit% readstat] /* comment line
&SV line2 = [read %AMLunit% readstat] /* comment line
&SV line3 = [read %AMLunit% readstat] /* comment line
&SV line4 = [read %AMLunit% readstat] /* comment line
&TYPE

&SV line5 = [unquote [read %AMLunit% readstat]]
&SV .MapsPerFrame = [extract 2 %line5%]
&TYPE Maps Per Frame: %MapsPerFrame%

```

```

&TYPE

&SV line6 = [unquote [read %AMLunit% readstat]]
&SV .StartFrame = [extract 2 %line6%]
&TYPE Starting Frame = %.StartFrame%
&TYPE

&SV line7 = [unquote [read %AMLunit% readstat]]
&SV .EndFrame = [extract 2 %line7%]
&TYPE Ending Frame = %.EndFrame%
&TYPE

&SV line8 = [unquote [read %AMLunit% readstat]]
&SV .ElapsedTime = [extract 2 %line8%]
&TYPE Elapsed time between frames = %.ElapsedTime%
&TYPE

&SV line9 = [unquote [read %AMLunit% readstat]]
&SV .FrameStep = [extract 2 %line9%]
&TYPE Frame Step = %.FrameStep%
&TYPE

&SV line10 = [unquote [read %AMLunit% readstat]]
&SV .SecsPause = [extract 2 %line10%]
&TYPE Seconds to pause frame display in AP = %.SecsPause%
&TYPE

&SV line11 = [read %AMLunit% readstat] /* blank line

&SV line12 = [UNQUOTE [READ %AMLunit% readstat] ]
&SV .wPath = [extract 2 %line12%]

&TYPE Time series grids location = %.wPath%
&TYPE

&SV line13 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
  &SV .Name%MapNumber% = ~
    [EXTRACT [CALC %MapNumber% + 1] %line13%]
  &TYPE Map No. %MapNumber%: [VALUE .Name%MapNumber%]
&END
&TYPE

&SV line14 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
  &sv .RMT%MapNumber% = ~
    [EXTRACT [CALC %MapNumber% + 1] %line14%]
  &TYPE Map No. %MapNumber% Remap Table: [VALUE .RMT%MapNumber%]
&END
&TYPE

&SV line15 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
  &sv .ShdSet%MapNumber% = ~
    [EXTRACT [CALC %MapNumber% + 1] %line15%]
  &TYPE Map No. %MapNumber% shadeset: [VALUE .ShdSet%MapNumber%]
&END

```

```

&TYPE

&SV line16 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
  &sv .MapTitle%MapNumber% = ~
    [EXTRACT [CALC %MapNumber% + 1] %line16%]
  &TYPE Map No. %MapNumber% title: [VALUE .MapTitle%MapNumber%]
&END
&TYPE

&SV line17 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
  &sv .MapSubTitle%MapNumber% = ~
    [EXTRACT [CALC %MapNumber% + 1] %line17%]
  &TYPE Map No. %MapNumber% subtitle: [VALUE .MapSubTitle%MapNumber%]
&END
&TYPE

&SV line18 = [unquote [read %AMLunit% readstat]]
&SV .OrderToKill = [extract 2 %line18%]
&IF %.OrderToKill% = 1 &THEN; &TYPE Grids deleted after displayed
&ELSE &TYPE Generated grids will not be deleted
&TYPE

&SV line19 = [unquote [read %AMLunit% readstat]]
&SV .CreateMovie = [extract 2 %line19%]
&IF %.CreateMovie% = 1 &THEN; &TYPE An MPEG file will be created
&ELSE &TYPE No MPEG file will be created
&TYPE

&SV line20 = [unquote [read %AMLunit% readstat]]
&SV .MovieName = [extract 2 %line20%]
&TYPE Movie Name = %.MovieName%
&TYPE

&SV closestat = [CLOSE %AMLunit%]

&RETURN

/* #####
/* #####          ROUTINE CREATE_GRIDS          #####
/* #####

&routine CreateGrids

/* OPEN AND WRITE TITLES IN THE FILES THAT WILL HOLD THE GRIDS STATS.

&DO i = 1 &TO %.MapsPerFrame%
  &SV fileexists = [EXIST %wPath%[value .Name%i%]_desc.txt -FILE]
  &IF %fileexists% &THEN
    &SV NULL [DELETE %wPath%[value .Name%i%]_desc.txt]
    &SV fileunit%i% ~
      [open %wPath%[value .Name%i%]_desc.txt openstat -WRITE]
    &SV NULL [WRITE [value fileunit%i%] ~
      'GridName      Min.      Max.      Mean      StdDev']
    &SV NULL [WRITE [value fileunit%i%] ~
      '-----']]

```

```

&END

&FORMAT 3

/* THE INPUT ASCII GRIDS (FROM CASC2D_SED) ARE CALLED
/*      ex. depth.1, depth.2, AND SO ON
/* GRIDS ARE GOING TO BE CALLED ex. depth1, depth2 AND SO ON

/* IF OLD GRIDS ARE PRESENT, THEY WILL BE REUSED, OTHERWISE, THEY
/* WILL BE CREATED. MAKE SURE THAT THOSE OLD GRIDS ARE THE ONES YOU
/* WANT TO USE, OTHERWISE, KILL THEM BEFORE STARTING THE CURRENT AML

&DO FrameNumber = %.StartFrame% &TO %.EndFrame% &BY %.FrameStep%

    &TYPE Frame Number: %FrameNumber%

    &DO MapNumber = 1 &TO %.MapsPerFrame%
        &SV String := %.wPath%[value .Name%MapNumber%]
        &SV gridexist = [EXISTS %String%%FrameNumber% -grid]
        &IF %gridexist% eq .FALSE. &THEN
            &DO
                &TYPE Creating grid
                ASCIIGRID ~
                    %String%.%FrameNumber% %String%%FrameNumber% FLOAT
                &TYPE Calculating statistics
                &DESCRIBE %String%%FrameNumber%
                &SV DataLine := ~
                    [FORMAT '%1,10% %2,-15% %3,-15% %4,-15% %5,-15%' ~
                    %GRD$GRID% %GRD$ZMIN% %GRD$ZMAX% %GRD$MEAN% %GRD$STDV%]
                &SV NULL [WRITE [value fileunit%MapNumber%] %DataLine%]
                /*&CALL ComputeStats
            &END
        &END
    &END

&END

&DO i = 1 &TO %.MapsPerFrame% /* Closing the stats. files
    &SV NULL [CLOSE [value fileunit%i%]]
&END
&RETURN

/* #####
/* #####          ROUTINE DISPLAY_GRIDS          #####
/* #####

&ROUTINE DisplayGrids

/* THE VARIABLE iter WAS ORIGINALLY INSIDE THE animate.aml
/* WE INITIALIZE IT OUTSIDE THE LOOP THAT CREATES THE GRIDS

&SV .iter -1

AP

&CALL GetUserSpecs

&SV timesec 0 /* Initialize total time to zero before displaying

```

```

&FORMAT 1

/*&IF %.MapsPerFrame% = 1 &THEN
/*  &RUN DS1.AML
/*&ELSE
    &RUN APdisplay.aml

&CALL PutUserSpecs

QUIT

&RETURN

/* #####
/* #####          ROUTINE ANIMATE          #####
/* #####

/* THIS PART OF THE CODE IS A MODIFIED COPY OF THE animate
/* routine WRITEN BY STEPHEN LEAD IN THE animate.aml

&ROUTINE animate

/* TO CREATE THE ANIMATED MPEG FILE, WE USE THE BERKELEY MPEG ENCODER
/* WITH THE DEFAULT MPEG PARAMETER FILE DEFINED IN THIS AML.

/* THE USER CAN CHANGE THESE PARAMETER VALUES TO HIS/HER LIKING
/* ACCORDING TO THE USERS-GUIDE.PS CREATED AT THE BERKELEY MULTIMEDIA
/* RESEARCH CENTER, WHICH CAN BE FOUND AT:
/* http://bmrc.berkeley.edu/frame/research/mpeg/mpeg\_encode.html

/* A GOOD EXPLANATION AND EXAMPLE OF THE MPEG PARAMETER FILE IS FOUND
/* AT: http://www.math.arizona.edu/swig/animation/MPEG/\(by Mark Hays\)

/* Calculate the first and last images - the temporary files
/* contain a prefix with 5 trailing digits. We need to substitute
/* zeros for any leading blank values.

&s first 00000
&s last %.iter%
    &do i = 1 &to [calc 5 - [length %.iter%]]
        &s last 0%last%
    &end

/* Write out a parameter file for this animation.
&s param_file [scratchname -file]
&s fileunit [open %param_file% openstat -write]
&if %openstat% <> 0 &then
    &do
        &s str Could not create %param_file%...; &call bailout
    &end

&s null [write %fileunit% 'PATTERN IBBPBBPBBPBB']
&s null [write %fileunit% 'INPUT']
&setchar &function '|*' '|*|'
&s null |* write %fileunit% ~
        |* quote %.MovieName%* [%first%-%last%] *| *|
&setchar &function [ ]

```

```

&s null [write %fileunit% 'END_INPUT']
&s null [write %fileunit% [quote OUTPUT %wPath%.MovieName%]]
&s null [write %fileunit% [quote INPUT_DIR [show work]]]
&s null [write %fileunit% ~
  [quote INPUT_CONVERT [joinfile [joinfile ~
    [pathname $ARCHOME] bin -SUB] rasttopnm -FILE] *]]
&s null [write %fileunit% 'BASE_FILE_FORMAT PNM']
&s null [write %fileunit% 'SLICES_PER_FRAME 1']
&s null [write %fileunit% 'PIXEL HALF']
&s null [write %fileunit% 'RANGE 9']
&s null [write %fileunit% 'PSEARCH_ALG TWOLEVEL']
&s null [write %fileunit% 'BSEARCH_ALG CROSS2']
&s null [write %fileunit% 'GOP_SIZE 12']
&s null [write %fileunit% 'PQSCALE 10']
&s null [write %fileunit% 'IQSCALE 10']
&s null [write %fileunit% 'BQSCALE 10']
&s null [write %fileunit% 'REFERENCE_FRAME ORIGINAL']
&s null [write %fileunit% 'FORCE_ENCODE_LAST_FRAME']
&s null [close %fileunit%]

/* EXECUTE THE ENCODER
&sys [joinfile [joinfile [pathname $ARCHOME] bin -SUB] ~
  mpeg_encode -FILE] %param_file%

/* DELETE THE TEMPORARY FILES
&s deletestat [DELETE %param_file% -file]
  &DO i = 0 &TO %.iter%
  &SV value %i%
    &DO a = 1 &TO [calc 5 - [LENGTH %i%]]
    &S value 0%value%
    &S delstat [DELETE %.MovieName%%value% -file]
  &END
&END

&RETURN

/* #####
/* #####          ROUTINE KILL_ALL_GRIDS          #####
/* #####
/* KILLS THE INPUT GRIDS IF THE USER DECIDED SO

&ROUTINE KillAllGrids
&IF %.OrderToKill% = 1 &THEN

  &DO FrameNumber = %.StartFrame% &TO %.EndFrame% &BY %.FrameStep%

  &type Killing Grids Number: %FrameNumber%

  &DO MapNumber = 1 &TO %.MapsPerFrame%
    &sv String := %wPath%[value .Name%MapNumber%]
    &sv gridexist = [exists %String%%FrameNumber% -grid]
    &if %gridexist% &then
      KILL %String%%FrameNumber% ALL
  &END

&END

&RETURN

```

```

/* #####
/* #####          ROUTINE GET_USER_SPECS          #####
/* #####

/* DETERMINES THE USERS SPECIFICATIONS BEFORE IMPLEMENTING NEW ONES
/* WITH THIS AML

&ROUTINE GetUserSpecs

    /* GET THE NUMBER OF DECIMALS
    &SV UsersDecimals [SHOW &FORMAT]
    &TYPE UsersDecimals %UsersDecimals%

    /* GET THE DISPLAY WINDOW SPECIFICATION
    &SV UsersAPdisplay [SHOW DISPLAY]
    &TYPE UsersAPdisplay %UsersAPdisplay%

    /* GET THE SHADESET
    &SV UsersShadeSet [SHOW SHADESET]
    &TYPE UsersShadeSet %UsersShadeSet%

    /* GET THE TEXT SPECIFICATIONS
    &SV UserTextFont [SHOW TEXTFONT]
    &SV UserTextQuality [SHOW TEXTQUALITY]
    &SV UserTextSpacing [SHOW TEXTSPACING]
    &SV UserTextSize [SHOW TEXTSIZE]

&RETURN

/* #####
/* #####          ROUTINE PUT_USER_SPECS          #####
/* #####

/* RESTORES THE USER'S SPECIFICATIONS

&ROUTINE PutUserSpecs

    /* RESTORE THE USER'S NUMBER OF DECIMALS
    &FORMAT %UsersDecimals%

    /* RESTORE THE USER'S DISPLAY WINDOW SPECIFICATION
    DISPLAY %UsersAPdisplay%

    /* RESTORE THE USER'S SHADESET
    SHADESET %UsersShadeSet%

    /* RESTORE THE USER'S TEXT SPECIFICATIONS
    TEXTFONT %UserTextFont%
    TEXTQUALITY %UserTextQuality%
    TEXTSPACING %UserTextSpacing%
    TEXTSIZE %UserTextSize%

&RETURN

```

```

/* #####
/* #####          ROUTINE CHECK_INPUT_ERRORS          #####
/* #####

&ROUTINE CheckInputErrors

/* TODO: this is just one of the errors.  I should check for the rest
&IF %.MapsPerFrame% NE 1 AND ~
    %.MapsPerFrame% NE 2 AND ~
    %.MapsPerFrame% NE 3 AND ~
    %.MapsPerFrame% NE 4 AND ~
    %.MapsPerFrame% NE 6 &THEN
&DO
    &TYPE WARNING !!!!
    &TYPE You have chosen to display %.MapsPerFrame% grids per frame
    &TYPE Please, Choose to display 1, 2, 3, 4 or 6 grids per frame
    &TYPE
    &CALL bailout
&END

&RETURN

/* #####
/* #####          ROUTINE BAIL_OUT          #####
/* #####

&ROUTINE bailout
    &IF NOT [variable str] &THEN &S str Bailing out of %aml$file%
    &RETURN; &RETURN &WARNING %str%
    &TYPE
&RETURN

```

```

/* #####
/* #####          APDISPLAY.AML          #####
/* #####

/* =====
/* PURPOSE:
/* Displays different thematic map series in AP.  Legends are
/* displayed, with the key files being generated automatically if
/* they are not created yet.
/* Titles for each thematic series can be shown.  Time is also
/* displayed.  This time is computed from the elapsed time between
/* CASC2D maps, and the map number in the series.
/*
/* =====
/* ARGUMENTS:
/*  None
/*
/* =====
/* VARIABLES:
/*  Passed from the main program (DS.aml) as global
/*  variables.
/*
/* =====
/* CALLS:
/*  TakeScreenShot.aml
/*
/* =====
/* ROUTINES:
/*  Legend      -- Creates key files and displays map legends
/*               according to the given remap table and shadeset
/*  TitleText  -- Writes map titles (specified in the Parameter file)
/*  TimeText   -- Writes the event simulation time
/*  SetLayoutVariables -- Specifies position of legend, title, sub-
/*               title, simulation time and maps in a given layout
/*
/* =====
/* HISTORY:
/*  August 18, 2001:  This part of the code is taken from the main
/*  program (DisplaySeries.aml) and becomes a stand alone aml.
/*  November 16, 2001: Display2.aml, Display3.aml...Display6.aml
/*  are re-written more generically in only one AML (Display.aml).
/*
/* =====

/* Displays a red line around the AP window
LINECOLOR 2
LINESIZE 0.05

/* I want a white background.  Instead of changing the canvas color

/* I will just put a white patch as background
SHADETYPE COLOR
SHADECOLOR 1
&SV CanvasColor = [SHOW CANVASCOLOR]
&TYPE Canvas Color is %CanvasColor%

```

```

/* SPECIFY ARC PLOT DISPLAY CONFIGURATION
&IF %.MapsPerFrame% EQ 1 OR %.MapsPerFrame% EQ 4 &THEN
  &DO
    DISPLAY 9999 SIZE CANVAS 720 600 /* Landscape
    PAGESIZE 11 8.5 /* Landscape
    /* &IF %CanvasColor% EQ BLACK &THEN PATCH 0 0 11 8.5
    BOX 0 0 11 8.5 /* Landscape
  &END
&ELSE
  &DO
    DISPLAY 9999 SIZE CANVAS 530 720 /* Portrait
    PAGESIZE 8.5 11 /* Portrait
    /*&IF %CanvasColor% EQ BLACK &THEN PATCH 0 0 8.5 11
    BOX 0 0 8.5 11 /* Portrait
  &END

/* LEGEND AND TITLES WILL NOT CHANGE ALONG THE SIMULATION SO THEY ARE
/* DISPLAYED ONLY ONCE AT THE BEGINING (they won't flicker every time
/* the display is refreshed and the display will be faster)

/* Set variables depending on the number of maps per frame
&CALL SetLayoutVariables

/* Creates Legends and Titles in layout
&CALL Legend
&CALL TitleText

/* DISPLAY THEMATIC MAP SERIES

&DO FrameNumber = %.StartFrame% &TO %.EndFrame% &BY %.FrameStep%

  GNDS TRANSPARENT /* NODATA cells will be displayed transparently.
                  /* that I can see the map legend

  &SV timesec [calc %FrameNumber% * %.ElapsedTime%]
  &SV timemin [calc %timesec% / 60]
  &CALL TimeText

  &TYPE Displaying frame Number: %FrameNumber%

  &DO n = 1 &TO %.MapsPerFrame%
    MAPLIMITS [UNQUOTE[VALUE MLimits%n%]]
    MAPEXTENT %.wPath%[VALUE .Name%n%]%FrameNumber%
    MAPPOSITION LR LR
    SHADEDELETE ALL
    SHADESET [VALUE .ShdSet%n%]
    GRIDSHADES ~
              %.wPath%[VALUE .Name%n%]%FrameNumber% VALUE [VALUE .RMT%n%]
    LINECOLOR 2
    LINESIZE 0.04
    /*ARCS h:\gismaps\3hydromodel\30\drainagedata\basins_polys
  &END

  &PAUSE &SECONDS %.SecsPause%

  &IF %.CreateMovie% = 1 &THEN &DO

```

```

        &RUN TakeScreenShot
        &TYPE Done: Screen shot of frame Number: %FrameNumber%
&END

&END

&RETURN

/* #####
/* #####          ROUTINE LEGEND          #####
/* #####

/* CREATES FILES TO STORE THE KEY FILES.
/* KEY FILES ARE DERIVED FROM REMAP TABLES

&ROUTINE Legend

&DO k = 1 &TO %.MapsPerFrame%

    &SV fileunit%k% [OPEN %wPath%[value .Name%k%].key openstat -write]

    &IF %openstat% <> 0 &THEN
        &DO
            &SV str Could not create key_file ...; &CALL bailout
        &END

    /* DATA WILL BE TAKEN FROM THE REMAP TABLE (DEFINED BY THE USER)
    /* AND WRITEN TO THE TEMPORAL KEY FILE WITH THE CORRESPONDING
    /* KEY FILE FORMAT (see "Drawing Key Legend" help)

    &SV eof 0

    &SV RemapFile%k% [open [value .RMT%k%] openstat -read]
    &SV line [UNQUOTE [READ [value RemapFile%k%] eof]]

    /* I ASSUME THAT THE USER DOESN'T HAVE MORE THAN 100 CLASSES!!!
    /* I WILL ONLY WRITE THE UPPER LIMIT AS THE TEXT CLOSE TO THE KEY
    /* IF YOU WANT TO WRITE THE WHOLE INTERVAL --> CHANGE NEXT LINES
    /* OF CODE TO READ AND WRITE THE 1st AND 2nd ARGUMENT OF %line%

    &DO i = 1 &TO 100 &WHILE %eof% NE 102
        /* 102 is returned value for EOF
        &SV NULL [WRITE [VALUE fileunit%k%] .%i% ]
            &SV key%i% := [QUOTE < [EXTRACT 2 %line%]]
        &SV NULL [WRITE [VALUE fileunit%k%] [value key%i%]]
        &SV line [UNQUOTE [READ [value RemapFile%k%] eof]]
    &END

    &SV NULL [CLOSE [VALUE fileunit%k%]]

    &SV closestat = [CLOSE [VALUE RemapFile%k%]]
&END

/* TODO:CHANGE THE NEXT LINES WITH THE FONTS AND SIZES OF YOUR LIKING
/* NEXT SPECIFICATIONS WILL BE COMMON FOR ALL LEGENDS

```

```

/* Standard hardware device color index:
/* 1:white; 2:red; 3:green; 4:blue; 5:cyan; 6:magenta; 7:yellow

SHADETYPE COLOR
TEXTCOLOR 1          /* Legend text color: black
TEXTFONT 'Times Bold' /* Legend Font:smooth panel
TEXTSIZE 0.23 0.23   /* Legend text size: .2 inches
TEXTSPACING 0
TEXTQUALITY PROPORTIONAL
LINEWIDTH 0.01       /* Line width around key box
LINECOLOR 0          /* Line color (black) around key box
KEYBOX .25 .18       /* Key box width and height (inches)
KEYSEPARATION 0.05 0.03 /* Key dist. to text and between keys

&DO j = 1 &TO %.MapsPerFrame%
  /*BOX [UNQUOTE[VALUE Area%j%]]
  SHADEDELETE ALL
  KEYAREA [UNQUOTE [value Area%j%]]
  SHADESET [value .ShdSet%j%]
  KEYSHADE %.wPath%[value .Name%j%].key
  &SV NULL [DELETE %.wPath%[value .Name%j%].key] /* Delete key file
&END

&RETURN

/* #####
/* #####          ROUTINE TITLE_TEXT          #####
/* #####

/* CREATES THE TITLES AND SUBTITLES CORRESPONDING TO EACH OF THE
/* THEMATIC SERIES -- TITLES ARE USER DEFINED IN THE PARAMETER FILE

&ROUTINE TitleText

TEXTCOLOR 2          /* Title in red
TEXTSIZE 0.4 0.4     /* Title font height & width: 0.4 inches
TEXTFONT 'Courier bold' /*
TEXTSPACING 0
TEXTQUALITY PROPORTIONAL

/* MOVE: specifies a coordinate point used to position the cursor and
/* to position text drawn with TEXT and TEXTFILE. Specify only the
/* lower left corner of the title and subtitle block.

&DO m = 1 &TO %.MapsPerFrame%

  MOVE [VALUE xTitlePos%m%] [VALUE yTitlePos%m% ]
  TEXT [value .MapTitle%m%]
  MOVE [VALUE xTitlePos%m%] [CALC [VALUE yTitlePos%m%] - 0.4]
  TEXT [value .MapSubTitle%m%]
&END

&RETURN

```



```

&IF %.MapsPerFrame% EQ 2 &THEN          /*
&DO                                     /*
  &SV MLimits1 = '0.1 5.4 8.4 10.9' /*
  &SV MLimits2 = '0.1 0.1 8.4 5.5' /*
  &SV Area1 = '0.1 8.9 2.85 10.65' /*
  &SV Area2 = '0.1 3.6 2.85 5.35' /*
  &SV xTitlePos1 = 2.4                /*
  &SV yTitlePos1 = 10.6
  &SV xTitlePos2 = %xTitlePos1%
  &SV yTitlePos2 = [CALC %yTitlePos1% - 5.3 ]
  &SV xmin = 4.20
  &SV ymin = 0.1
&END

```

1
2

/* this one needs more work. Check values

```

&IF %.MapsPerFrame% EQ 3 &THEN
&DO
  &SV MLimits1 = '0.0 7.4 8.5 11.0'
  &SV MLimits2 = '0.0 3.7 8.5 7.3'
  &SV MLimits3 = '0.0 0.0 8.5 3.6'
  &SV Area1 = '0.1 8.9 2.85 10.65'
  &SV Area2 = '0.1 3.6 2.85 5.35'
  &SV Area3 = '0.1 3.6 2.85 5.35'
  &SV xTitlePos1 = 5.75 /*
  &SV yTitlePos1 = 0.9 /*
  &SV xTitlePos2 = 5.75 /*
  &SV yTitlePos2 = 6.3 /*
  &SV xTitlePos3 = 6.3 /*
  &SV yTitlePos3 = 5 /*
  &SV xmin = 2.20 /*
  &SV ymin = 0.1 /*
&END

```

1
2
3

```

&IF %.MapsPerFrame% EQ 4 &THEN
&DO
  &SV MLimits1 = '0 4.25 5.5 8.5'
  &SV MLimits2 = '5.5 4.25 11 8.5'
  &SV MLimits3 = '0 0 5.5 4.25'
  &SV MLimits4 = '5.5 0 11 4.25'
  &SV Area1 = '0.05 6.95 2.05 8.45'
  &SV Area2 = '5.40 6.95 7.40 8.45'
  &SV Area3 = '0.05 2.75 2.05 4.25'
  &SV Area4 = '5.40 2.75 7.40 4.25'
  &SV xTitlePos1 = 2.70
  &SV yTitlePos1 = 8.05 /*
  &SV xTitlePos2 = 8.10 /*
  &SV yTitlePos2 = 8.05 /*
  &SV xTitlePos3 = 2.70 /*
  &SV yTitlePos3 = 3.85 /*
  &SV xTitlePos4 = 8.10 /*
  &SV yTitlePos4 = 3.85 /*
  &SV xmin = 3.0
  &SV ymin = 0.05
&END

```

1	2
3	4

```

&IF %.MapsPerFrame% EQ 6 &THEN
  &DO
    &SV MLimits1 = '0.0 7.2 4.2 10.8' /*Landscape: 0 4.2 3.6 8.4
    &SV MLimits2 = '4.2 7.2 8.4 10.8' /*Landscape: 3.6 4.2 7.2 8.4
    &SV MLimits3 = '0.0 3.6 4.2 7.2' /*Landscape: 7.2 4.2 10.8 8.4
    &SV MLimits4 = '4.2 3.6 8.4 7.2' /*Landscape: 0 0 3.6 4.2
    &SV MLimits5 = '0.0 0.0 4.2 3.6' /*Landscape: 3.6 0 7.2 4.2
    &SV MLimits6 = '4.2 0.0 8.4 3.6' /*Landscape: 7.2 0 10.8 4.2
    &SV Area1 = '0.05 9.15 1.65 10.85'
    &SV Area2 = '4.2 9.15 5.8 10.85'
    &SV Area3 = '0.05 5.55 1.65 7.25'
    &SV Area4 = '4.2 5.55 5.8 7.25'
    &SV Area5 = '0.05 1.95 1.65 3.65'
    &SV Area6 = '4.2 1.95 5.8 3.65'
    &SV xTitlePos1 = 2 /*
    &SV yTitlePos1 = 10.2 /*
    &SV xTitlePos2 = 6.15 /*
    &SV yTitlePos2 = 10.2 /*
    &SV xTitlePos3 = 2 /*
    &SV yTitlePos3 = 6.6 /*
    &SV xTitlePos4 = 6.15 /*
    &SV yTitlePos4 = 6.6 /*
    &SV xTitlePos5 = 2 /*
    &SV yTitlePos5 = 3.0 /*
    &SV xTitlePos6 = 6.15
    &SV yTitlePos6 = 3.0
    &SV xmin = 2.20
    &SV ymin = 0.05
  &END

  &RETURN

```

1	2
3	4
5	6

```

/* #####
/* #####          ROUTINE TAKE_SCREEN_SHOT          #####
/* #####

/* THIS PART OF THE CODE IS A PARTIAL COPY OF THE createimages
/* routine WRITEN BY STEPHEN LEAD IN THE animate.aml

/* IMPORTANT: I HAD TO CHANGE MY SCREEN DISPLAY TO 'TRUE COLORS'
/* (NOT 256 OR 65536 COLORS)

/* CREATE A TEMPORARY SCREENSHOT IMAGE, CONVERT TO SUNRASTER.
/* OMIT THE CONVERSION STAGE IF YOU ARE NOT USING WINDOWS NT.

&SV tempimage [scratchname -file]

/* screensave command:IN WINDOWS NT THE IMAGE IS SAVED AS A BMP FILE
/* UNLESS OTHERWISE SPECIFIED. SINCE JPEG IS ALREADY A COMPRESSED
/* IMAGE, AND THIS IMAGE WILL BE FURTHER COMPRESSED IN THE MPEG
/* ENCODER (THUS, LOOSING RESOLUTION) WE SAVE THE FILES AS SUNRASTER

SCREENSAVE %tempimage%
&SV rimage [scratchname -file]
ARC CONVERTIMAGE %tempimage% %rimage% sunraster none
&SV delstat [delete %tempimage% -file]
&SV delstat [delete %tempimage%w -file]

/* NOTE: ANOTHER WAY TO CREATE AN ANIMATED FILE IS TO CREATE GIF
/* OR JPEG FILES AND TO USE A PROGRAM TO CREATE AN ANIMATED GIF.
/* ADVANTAGE: EASY TO USE, VERY VERSATIL. DISADVANTAGE: SOFTWARE
/* NOT FOR FREE. A GOOD STARTING POINT TO LEARN ABOUT ANIMATED GIFs
/* AND COMPARE SOFTWARE: http://www.webreference.com/dev/gifanim/
/* IN CASE THAT YOU DECIDE TO CREATE AN ANIMATED GIF, YOU CAN END
/* THE AML HERE. NO NEED TO EXECUTE THE animate SUBROUTINE.

/* COPY THIS IMAGE A NUMBER OF TIMES TO PROVIDE MULTIPLE FRAMES,
/* ELSE THE IMAGE ONLY APPEARS FOR A SPLIT SECOND IN THE MPEG FILE.
/* SETTING THE APPROXIMATE NUMBER OF FRAMES PER SECOND TO 40 FRAMES

/* SEEMS TO RESULT IN EACH IMAGE LASTING ABOUT 1 SECOND.
/* THE LARGER NUMBER OF repeats --> THE MOVIE DOES NOT GO SO FAST
/* BUT THE TIME IT TAKES TO CREATE THE MOVIE IS BIGGER AND THE
/* MPEG FILE WILL ALSO BE LARGER

&SV repeats 4

&TYPE Copying the image 4 times

&do a = 1 &to %repeats%
  &SV .iter [calc %.iter% + 1]
  &SV image%.iter% [scratchname -prefix %.MovieName% -file]
  &SV copystat [copy %rimage% [value image%.iter%] -file]
  &if %copystat% > 0 &then
    &do
      &s str Error copying image file...; &call bailout
    &end
&end

```

```

/* DELETE TEMPORARY FILES
&s delstat [delete %rsimage% -file]
&s delstat [delete %rsimage%w -file]

&RETURN

/* #####
/* #####          ROUTINE BAIL_OUT          #####
/* #####

&ROUTINE bailout

    &IF NOT [variable str] &THEN &S str Bailing out of %aml$file%
    &RETURN; &RETURN &WARNING %str%

&RETURN

```

I.4.2 DS.aml parameter file example

```

# DISPLAYSERIES.AML INPUT FILE
# Please, use this file as a template.
# Leave at least one blank space between arguments

No.Maps_Per_Frame_(1;4;6):          4
Start_In_Frame_No.:                306
End_In_Frame_No.:                  307
Elapsed_Time_Between_Frames(sec):   60
Show_By_Frame_Step:                 1
Seconds_To_Pause_Frames_Display_InAP: 5

Map(s)_Location:  h:\simulations\distribprecip\17oct81\30m\hydromaps\
Map(s)_Name(s):   rain infvol depth  conc
Rema(s)_Table(s): rain2.rmp infvol.rmp depth.rmp conc14.rmp
Shade(s)_Set(s):  rain.shd infvol.shd depth.shd conc14.shd
Map_Title(s):     'Rainfall' 'Infiltr.' 'Depth' '      Cv'
Map_Subtitle(s):  ' [mm/h]' ' [mm]' ' [m]' ' [m3/m3]'
Kill_Grids_after_Displayed(1:yes;0:no):  0
Create_Movie(1:yes;0:no):                 0
Movie_Name:                               HydroSumm30.mpeg

```