



Technical Documentation Version 5.1

Multiple Run Management



C A D S W E S

Center for Advanced Decision Support for Water and Environmental Systems

These documents are copyrighted by the Regents of the University of Colorado. No part of this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means electronic, mechanical, recording or otherwise without the prior written consent of The University of Colorado. All rights are reserved by The University of Colorado.

The University of Colorado makes no warranty of any kind with respect to the completeness or accuracy of this document. The University of Colorado may make improvements and/or changes in the product(s) and/or programs described within this document at any time and without notice.

Multiple Run Management Table of Contents

Multiple Run Management	1
Introduction	1
Running a pre-configured MRM model	1
Managing MRM Configurations	2
Creating a new configuration	2
Editing an existing configuration	3
Deleting an existing configuration	3
Copying an existing configuration	3
Applying / cancelling changes to an existing configuration	3
Setting Up A Multiple Run Configuration	5
Create the configuration	5
Name	5
Mode	6
Policy	6
Input	6
Description	6
Output	7
Run Parameters	8
Saving and Restoring Initial Model State	9
Concurrent Runs	9
Consecutive Runs	11
Iterative Runs	12
Policy Based Runs	15
Rulebased Runs	15
Constraint Set Runs	17
Input DMI Runs	17
Index-Sequential Runs	18
Combined Runs	20
Index Sequential / DMI Mode	21
I/O-optimization Runs	22

Multiple Run Management

1. Introduction

Multiple Run Management is a RiverWare utility for setting up and running various types of model runs in series. The runs are set up through a Multiple Run Control Interface and submitted to RiverWare, which then carries out all runs and outputs the results into an ASCII file.

A multiple run is defined as a set of one or more model runs, where for all runs:

- The model configuration (object network) remains constant.
- The simulation/optimization run duration is the same (except for consecutive and/or I/O-optimization runs).
- The timestep size is constant.
- The same set of slots is saved to the output.

Runs are conducted automatically; i.e., RiverWare controls and invokes each run without user interaction.

2. Running a pre-configured MRM model

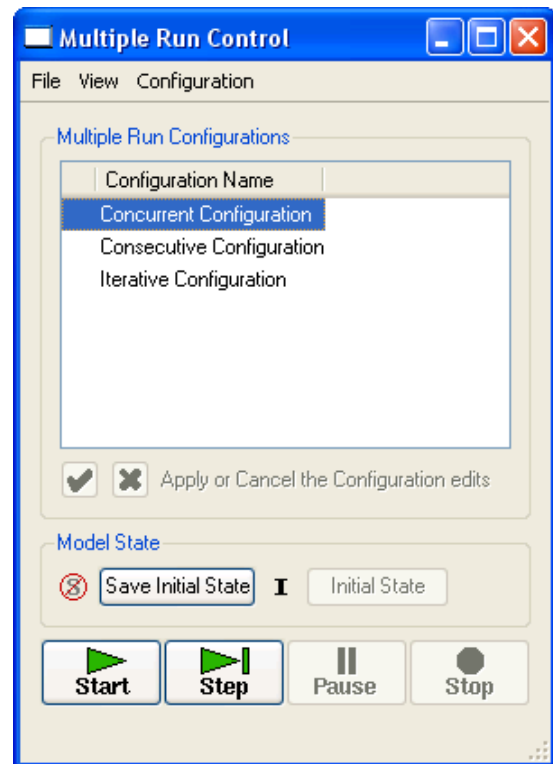
This section describes how to run a pre-configured MRM model. It is assumed that the MRM configuration is already set up and the user wishes only to run the multiple runs and view the multiple run output.

- Open the MRM Dialog by selecting **Control** ➔ **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar.



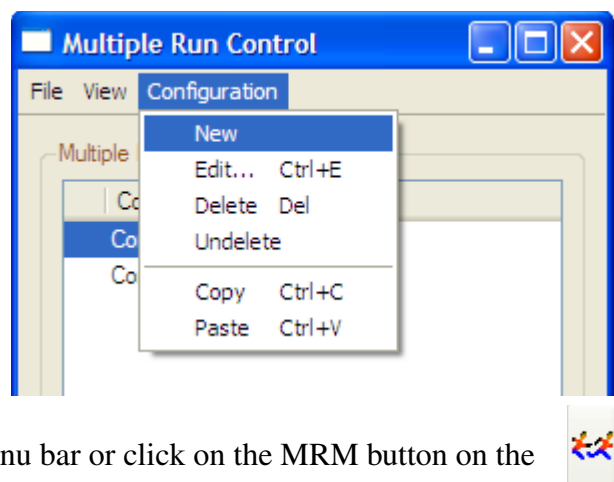
- Highlight the desired configuration
- Press the start button
- After the run has finished, determine where the output was placed by selecting **Configuration** ➤ **Edit...** and clicking on the Output tab in the ensuing Multiple Run Editor dialog.
 - If there is a value in the Control File field this means that data was sent to the RDF file (or files) specified in the control file using the “file=” keyword. If there are no “file=” keywords specified in the control file, the data will go to the file specified in the Data File field.
 - If there is a value in the DMI field, this means that data was sent to multiple files using a DMI. Check the diagnostics output window for DMI diagnostics that show where output was sent. DMI slot diagnostics must be turned on during the run for this information to be printed.

For more information on output, see click [HERE \(Section 4.7\)](#).



3. Managing MRM Configurations

This section describes how to manage MRM configurations using the Multiple Run Control dialog. The basics of creating, deleting, and opening a configuration for editing are described here.



- Open the MRM Dialog by selecting **Control** ➤ **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar.

3.1 Creating a new configuration

- Select **Configuration** ➤ **New** from the Multiple Run Control Dialog

- Highlight the newly created configuration and select **Configuration** ➤ **Edit...** or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...**
- In the Multiple Run Editor, make any necessary edits to the new configuration and click **OK** (see the “Setting Up a Multiple Run Configuration” section for details)
- To apply the new multiple run configuration edits and have them saved with the model file, click the green check mark in the Multiple Run Control Dialog

3.2 Editing an existing configuration

- Highlight the desired configuration in the Multiple Run Control dialog and select **Configuration** ➤ **Edit...** or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...**
- Apply desired edits in the Multiple Run Editor dialog and click **OK**
- To apply the changes to the configuration, click the green check mark in the Multiple Run Control dialog

3.3 Deleting an existing configuration

- Highlight the configuration to be deleted and select **Configuration** ➤ **Delete** or right-mouse click on the highlighted configuration to bring up a context menu and select **Delete**
- Apply the deletion by clicking the green check mark in the Multiple Run Control dialog

3.4 Copying an existing configuration

An easy way to create a new configuration similar to an existing one is by copying the existing configuration and making the slight changes. Configurations can only be copy and pasted within one model (i.e., within one open session of RiverWare).

- Highlight the configuration to be copied and select **Configuration** ➤ **Copy** or right-mouse click on the highlighted configuration to bring up a context menu and select **Copy**
- Select **Configuration** ➤ **Paste** to paste the copied configuration into the open Multiple Run Control dialog or right-mouse click on the highlighted configuration to bring up a context menu and select **Paste**
- Highlight the “Copy of” configuration and select **Configuration** ➤ **Edit...** (or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...**)
- Change the name, if desired, and apply any other edits. Click **OK** in the Multiple Run Editor and click on the green check mark in Multiple Run Control dialog


3.5 Applying / cancelling changes to an existing configuration


Changes to a MRM configuration must be accepted twice (once in the Multiple Run Editor and once in the Multiple Run Control dialog) to be fully active with the model file.

- Accept changes to a configuration in the Multiple Run Editor first by clicking **OK** or **Apply**
- Apply the edits to a configuration in the Multiple Run Control dialog by clicking on the green check mark
- Cancel changes to a configuration in the Multiple Run Editor by clicking **Cancel** or **Reset**
- Cancel edits to a configuration in the Multiple Run Control dialog by clicking on the red X mark


NOTE: You must save the model for any changes to the MRM configuration to be saved with the model file.



Edit notations: Whenever a configuration is edited the change is noted with an icon in the Multiple Run Control dialog until those changes have been either accepted or cancelled.

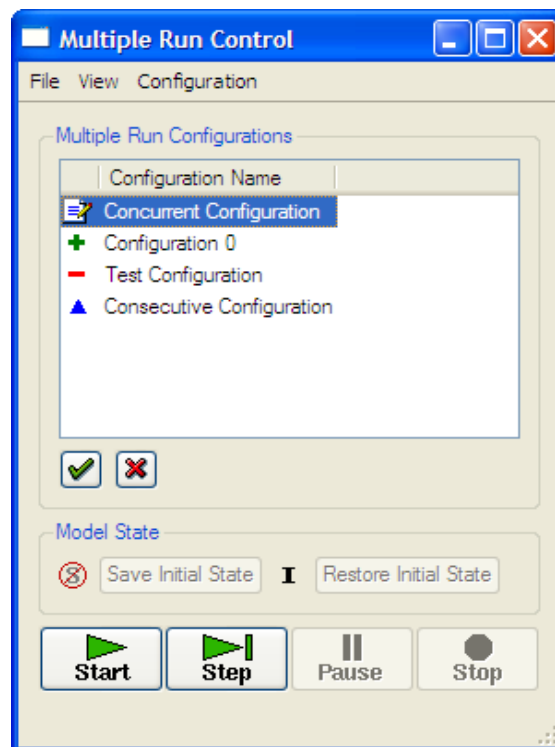
 **Edit** The configuration is currently being edited

 **Delta** The configuration has been edited/changed

 **New** The configuration has been newly added

 **Delete** The configuration is set to be deleted

These icons are visible until the edits have been either accepted  or rejected .



4. Setting Up A Multiple Run Configuration

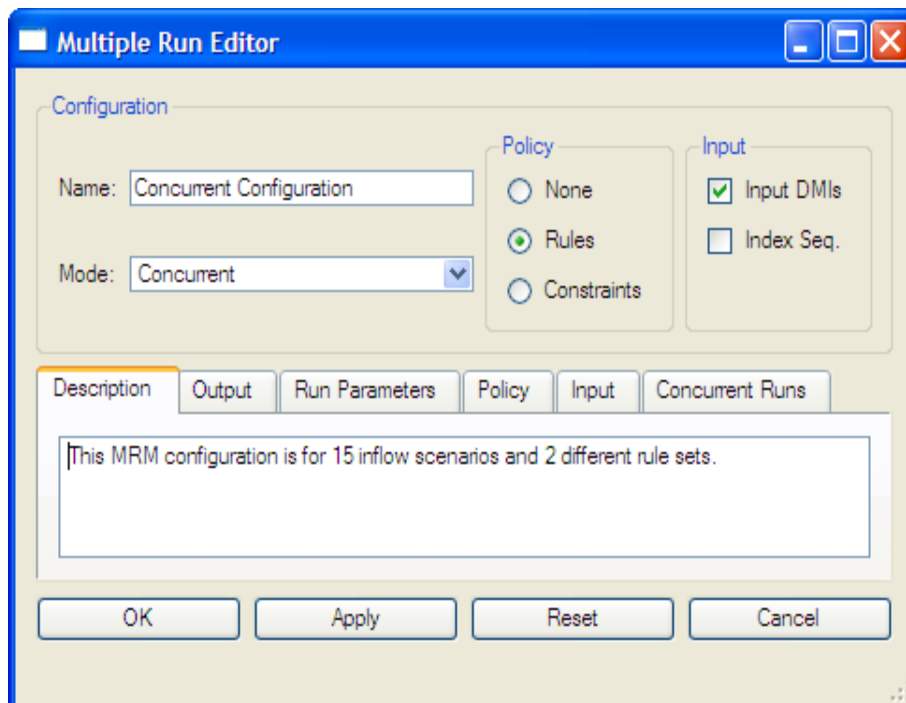
This section provides the steps to setting up a multiple run configuration. Setting up a multiple run configuration requires specifying several parameters in the Multiple Run Editor. The parameters and brief descriptions are provided below. For parameters that require in-depth discussion, additional documentation is provided in a separate section of this document.

- Open the MRM Dialog by selecting **Control** ➤ **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar.



4.1 Create the configuration

- Select **Configuration** ➤ **New** from the Multiple Run Control Dialog
- Highlight the newly created configuration and select **Configuration** ➤ **Edit...** or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...**
- In the Multiple Run Editor, make edits to the new configuration as described below
- After completing the edits, click **OK** in the Multiple Run Editor and click the green check mark in the Multiple Run Control Dialog



4.2 Name

- Provide a unique name for the configuration in the Name field

4.3 Mode

Concurrent Runs: Concurrent runs are multiple runs of which the time horizons are identical. The begin and end times, and timestep length, are the same for all runs.

Consecutive Runs: Consecutive runs are multiple runs of which the time horizons are laid out consecutively in time. The end time of one run is the initial time of the next run. Timesteps do not vary among the different runs in a consecutive run.

Iterative Runs: Iterative runs are multiple runs where MRM rules at the end of each run examine the state of the system and, if appropriate, set inputs for the subsequent simulation run. If no values are set and/or modified or the maximum number of iterations occurs, then the simulation ends. As in concurrent runs, the time horizons, begin and end times and timestep length are all the same for all runs.

- Select the Mode of the configuration (Concurrent, Consecutive or Iterative) from the Mode section
- The details of the Concurrent, Consecutive or Iterative mode must be specified under the Concurrent Runs, Consecutive Runs or Iterative Runs tab (see “Configuring Concurrent/Consecutive Runs/Iterative Runs” for details)

4.4 Policy

- Select the Policy setup of the configuration (None, Rules, or Constraints) from the Policy section
- The Policy name (i.e., ruleset/ constraint set name and location) must be specified under the Policy tab (see “Configuring Policy Based Runs” for details)
- Note that Policy is not applicable to Iterative runs, but an iterative run can be a rule based run where one ruleset is used for all iterations.

4.5 Input

- If the Multiple Runs use an Input DMI, toggle the Input DMI box in the Input section.
- If the Multiple Runs use Index Sequential, toggle the Index Sequential box in the Input section.
- The details of the Input DMI and/or Index Sequential setup must be specified under the Input tab (see “Configuring Input DMI Runs” or “Configuring Index Sequential Runs” for details).
- Note that Input is not applicable to Iterative runs.

4.6 Description

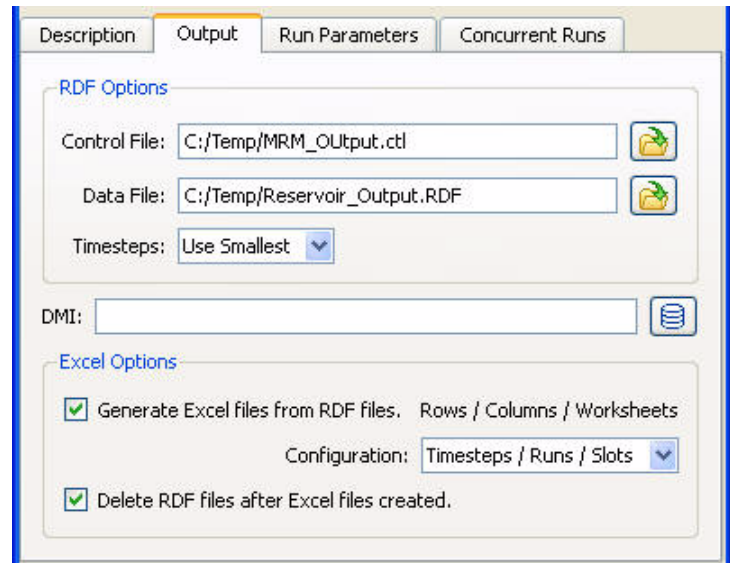
The configuration description may be multiple lines of text. This first non-blank line of the description appears in the MRM output file. This parameter is optional, and will not affect the MRM execution if left blank.

- Click on the Description tab of the Multiple Run Editor and in the text field below enter a description of the configuration.

4.7 Output

During a multiple run, output can go to one or more RiverWare Data Format (RDF) text files and/or an output DMI.

- Click on the Output tab of the Multiple Run Editor.
- Type or use the file chooser button to select a complete file path into the required control file field. The control file is used to specify which slots are output after each MRM run. The control file may contain “file =” specifiers for any line entries in the file. This causes data associated with those lines to go to the specified RDF output file. In the example control file below, if “fileName” is the same for each slot, then the output will go to a single RDF file; varying fileNames will send the output to multiple RDF files.



```
MountainStorage.Inflow: file=fileName
MountainStorage.Outflow: file=fileName
MountainStorage.Storage: file=fileName file=fileName2
```

The user can optionally specify multiple files for a single slot as in the third line above. Also, slots with a timestep different than the model’s timestep can be output using the same syntax. The Timesteps control specifies whether RDF output files may contain slots whose timesteps differ. There are three choices:

1. **Must Match** - All slots written to an RDF file must have the same timestep. Slots whose timesteps differ from the file’s timestep are skipped. (The file’s timestep is determined by the first slot MRM associates with the file.) The user is warned about slots being skipped, and asked whether to continue the MRM run.
 2. **Use Smallest** - Slots written to an RDF file may have different timesteps, with the output written using the smallest timestep. For example, if monthly and yearly slots are written to an RDF file, monthly values will be written and the yearly slots will write 11 NaN followed by a value.
 3. **Use Largest** - Slots written to an RDF file may have different timesteps, with the output written using the largest timestep. For example, if monthly and yearly slots are written to an RDF file, yearly values will be written and the monthly slots will write December values.
- In the control file, there are four ways to specify where the data files are created:
 4. Hard code the path: file=C:/DMI/Data/ResA.Inflow

5. Use an environment variable: `file=$(DMI_DATA)/ResA.Inflow`
6. Use '~': `file=~/ResA.Inflow` where '~' is replaced with `$RIVERWARE_DMI_DIR/<DMI name>`
7. If the filename is not specified, the output file will be created in the directory in which RiverWare resides.

The second option is recommended as it is more transparent than '~' but still allows the model to be moved from machine to machine by setting the environment variable to the appropriate value on each machine.

- Optionally type or use the file chooser button to select a complete file path into the Data File field. This file is used as the RDF output file for any lines in the control file that do not have an explicit file specifier. If the data file field is used and there are no "file=" specifiers in the control file, then all output will go to this single data file.
- The optional DMI field allows selecting or entering an output DMI or a group of output DMIs that will be run after each single run of the multiple run. The DMIs must be previously configured in the DMI Manager in RiverWare. The executable that is configured for the DMI can reference the trace number of the run that is outputting by referencing the last command line parameter passed from RiverWare: `-STrace=traceNumber`
- Excel options are available on the Windows platform. Checking the "Generate Excel files from RDF files" creates Excel files from all of the RDF output files after all runs have completed. Selecting a configuration from the drop-down box controls how timestep, slot, and run data from RiverWare are mapped onto the Excel dimensions of rows, columns, and worksheets. Checking the delete RDF item causes the RDF files to be deleted after the Excel files are created. (The separate ExcelWriter program contains the same functionality for creating Excel files from RDF files and can be used outside of RiverWare to process RDF files from an MRM run.)


4.8 Run Parameters

The Run Parameters describe the initialization and end date of each concurrent run. The run parameters also show the timestep size. Timestep size is dependent on the model and can only be changed in the single run control dialog.

- For concurrent runs, set the run initialization date and end date by toggling the up or down arrow next to the dates or by typing in the date. Alternatively, set the number of timesteps and the end date will automatically update.

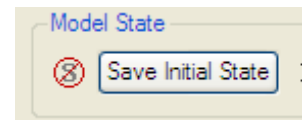
5. Saving and Restoring Initial Model State

This section describes how to save and restore the initial state of a model. The functionality allows users to save model data to a temporary file before a multiple run is invoked and then restore this information after the run has completed.

- Open the MRM Dialog by selecting **Control** ➔ **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar. 
- Save the initial state of a model before invoking a MRM run by pressing the **Save Initial State** button in the **Model State** area of the MRM Control Panel.
- Start the MRM run by pressing the **Start** button.
- After the run has finished, restore the initial model state by pressing the **Restore Initial State** button.

Pressing the **Save Initial State** button saves the entire contents of the model, including both input and output slots, TableSlots, User Methods, run information, and configuration to a file in the user's temporary directory (either /tmp or the directory specified by the TMP environment variable). Once the initial state has been saved, the button is disabled and reads "**Initial State Saved.**"

Save Initial State button prior to saving state -

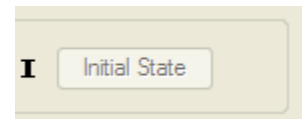


Save Initial State button after saving state -

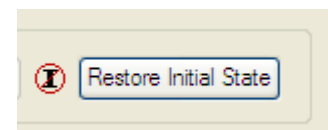


Once an initial state has been saved, it can be restored by clicking on the **Restore Initial State** button. This re-loads the entire contents of the model at the time its state was saved, clearing all data and configuration in the current state of the model. Until a multiple run has occurred, this button is disabled and reads "**Initial State**".

Restore Initial State button before a multiple run -




Restore Initial State button after a multiple run -



6. Concurrent Runs

Concurrent runs are multiple runs of which the time horizons are identical. The begin and end times, and timestep length, are the same for all runs.

- Open the MRM Dialog by selecting **Control** ➔ **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar. 

- In the Multiple Run Control dialog highlight the concurrent configuration and select **Configuration** → **Edit...** or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...**
- In the ensuing Multiple Run Editor, click on the Run Parameters tab and change the initialization and end dates to reflect the length you wish each multiple run to be

The screenshot shows the 'Run Parameters' tab of the Multiple Run Editor. It contains the following fields:

- Initial: 2004 C.E.
- Start: 2005
- Finish: 2025 C.E.
- Timesteps: 21
- Frequency: Yearly

- Click on the Concurrent Runs tab and check that the number of multiple runs is correct
 - In most cases, the total number of runs should be the product of the number of policy sets, the number of input DMIs, and the number of index sequential runs:

$$\text{\#of MRM runs} = \text{\#of Policy Sets} * \text{\#of Input DMIs} * \text{\#of Index Seq Runs}$$

- If the Pairs option for DMI/Index Sequential Mode has been selected on the Input tab, the total number of runs should equal the number of pairs of Input DMIs and Index Sequential runs. If the number of input DMIs does not match the number of Index Sequential runs, then the number of index sequential runs equals the total number of possible pairs, (i.e., the minimum of the number of input DMIs and the number of Index Sequential runs). (See the Index Sequential / DMI Mode section for more details.)

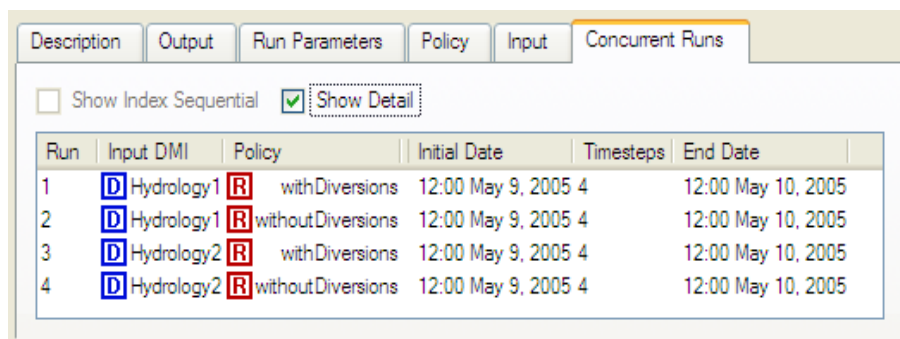
$$\text{\#of MRM runs} = \min(\text{\#of Input DMIs}, \text{\#of Index Seq Runs})$$

Example of Concurrent Runs with 2 Input DMIs and 2 policy sets, no Index Sequential:

The screenshot shows the 'Concurrent Runs' tab with a table of runs. The table has columns for Run, Input DMI, Policy, Initial Date, Timesteps (6 Hour), and End Date. There are four runs listed, each representing a unique combination of the two input DMIs and two policy sets.


Run	Input DMI	Policy	Initial Date	Timesteps (6 Hour)	End Date
1	D 1	R 1	12:00 May 9, 2005	4	12:00 May 10, 2005
2	D 1	R 2	12:00 May 9, 2005	4	12:00 May 10, 2005
3	D 2	R 1	12:00 May 9, 2005	4	12:00 May 10, 2005
4	D 2	R 2	12:00 May 9, 2005	4	12:00 May 10, 2005









- Toggling the Show Details box will show the names of the DMIs and policy sets



7. Consecutive Runs

Consecutive runs are multiple runs of which the time horizons are laid out consecutively in time. The end time of one run is the initial time of the next run. Timesteps do not vary among the different runs in a consecutive run.

- Open the MRM Dialog by selecting **Control** → **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar. 
- In the Multiple Run Control dialog highlight the consecutive configuration and select **Configuration** → **Edit...** or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...**
- Note that there is no Run Parameters tab. The consecutive runs will begin at the start timestep of the model run. If users wish to change the model start timestep, this should be done in the single Run Control dialog.
- Click on the Consecutive Runs tab and, if necessary, change the Initial Date for the consecutive runs. Do this by clicking twice (not double clicking) on the existing initial date and toggling the up/down arrows in the ensuing date-time spinner.
- Change the number of timesteps for the first consecutive run by clicking twice on the existing number of timesteps and toggling the up/down arrows of the ensuing integer spinner. Notice that the Finish Date automatically updates.
- Append a new row for each additional consecutive run by right-mouse clicking below the existing consecutive run and selecting Append Row from the ensuing context menu. Notice that the Initial Date is automatically set to match the Finish Date of the previous consecutive run. Only the Initial Date of the first consecutive run can be changed.
- Change the number of timesteps for each individual run, if necessary. The default is for newly appended runs to have the same number of timesteps as the previous run.

Run	Input DMI	Policy	Initial Date	Timesteps (6 Hour)	Finish Date
1	 1	 1	12:00 May 9, 2005	28	12:00 May 16, 2005
2	 2	 2	12:00 May 16, 2005	28	12:00 May 23, 2005
3	 1	 1	12:00 May 23, 2005	28	12:00 May 30, 2005
4	 2	 2	12:00 May 30, 2005	28	12:00 June 6, 2005

8. Iterative Runs

Iterative runs are multiple runs where MRM rules at the end of each run examine the state of the system and, if appropriate, set inputs for the subsequent simulation run. If no inputs are set or modified or the maximum number of iterations has already occurred, then the simulations end. As in concurrent runs, the time horizons, begin and end times and timestep length are the same for all runs. The iterative runs can use any of the controllers as specified in the single Run Control dialog: simulation (with or without accounting), rule based (with or without accounting) or optimization. If the run is rule based or optimization, the same RPL set or constraint set, respectively, is used in each iteration.

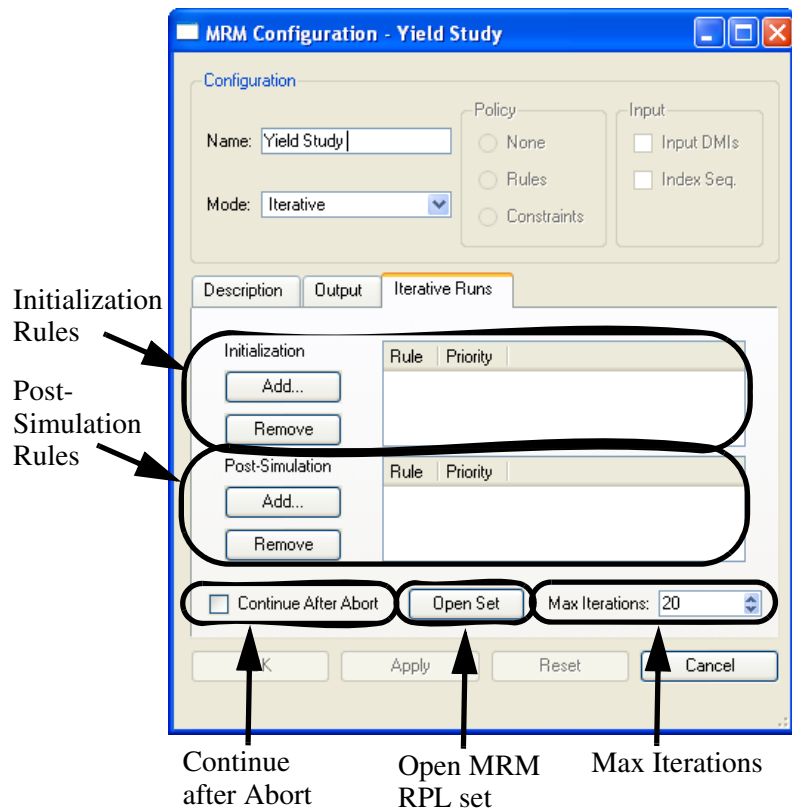
An iterative run executes as follows:

1. Initialize the iteration count.
2. Execute the initialization rule(s), if specified.
3. Perform the simulation run.
4. Execute the post-simulation rule(s), if specified.
5. If the post-simulation rule(s) return “no change”, that is they do not assign one or more new (different) values, the iteration is complete.
6. Otherwise, the iteration count is checked. If it equals the maximum number of iterations specified, then the iteration is complete also.
7. If the iteration is not complete, then increment the iteration count and return to step 3 above.

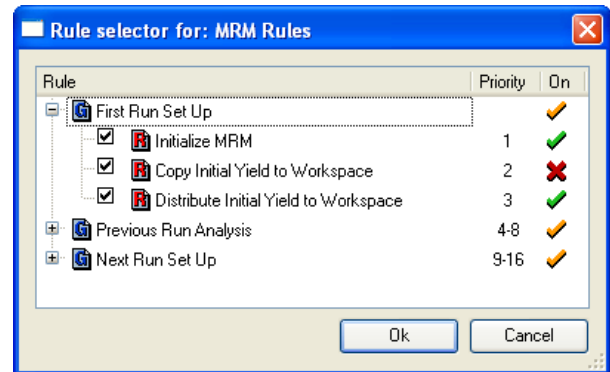
When an MRM rule, either during initialization or post-simulation, sets a value on a slot, the value is given the **i** flag indicating that it is a “computed input.” Values with the computed input flag are cleared at the beginning of an iterative MRM run but are not cleared between iterations of the MRM. This allows values set by the iterative MRM rules to persist between iterations but values are cleared at the beginning of the MRM run. In non-iterative MRM and single runs, values with an **i** flag are also cleared. The user should be aware of this behavior if switching from iterative MRM to another mode.

To configure an iterative run:

- Open the MRM Dialog by selecting **Control** ➔ **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar.
- In the Multiple Run Control dialog highlight the iterative configuration and select **Configuration** ➔ **Edit...** or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...** this will open the MRM Configuration dialog. If no iterative configuration exists, you will need to create a new one and edit it to define it as operating in iterative mode. For more details in doing this, please click [HERE \(Section 4\)](#) above.
- Note that within the MRM Configuration dialog with the iterative mode assigned, there is no Run Parameters tab. The iterative runs will begin at the start timestep of the model run. If users wish to change the model start timestep, this should be done in the single Run Control dialog.
- Click on the Iterative Runs tab. Here you will find 5 significant areas: the Initialization Rules, the Post-Simulation Rules, the Continue After Abort toggle, the Open MRM RPL set button, and the Max Iterations spinner.
- Click **Open Set** button to open the MRM RPL set. This dialog operates very similarly to the standard RBS Ruleset Editor dialog. Create one (or more) Policy Group(s) and associated rule(s). The MRM Rules created are stored within the model file. The set of MRM Rules stored in the model may be available for use with any number of iterative MRM configurations. This set of rules can also be accessed from the workspace **Policy** ➔ **Open MRM RPL Set**.
- Once the MRM Rules have been built, return to the **Iterative Runs** tab of the MRM Configuration dialog. In this tab, the user defines which of the MRM rules should execute as part of the initialization and which should execute after each iterative run, or both. These are defined in the appropriate areas using one of the following methods



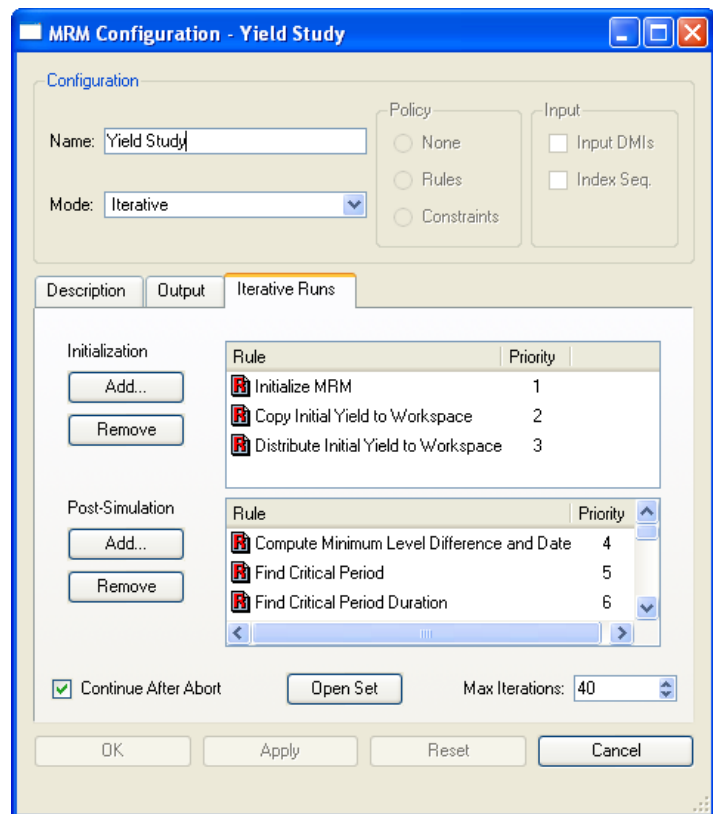
- Use the **Add** and **Remove** buttons. Click the **Add** button to bring up the **Rule Selector** as shown in the following figure. This dialog shows the rule groups in a tree view. Expand the tree-view to show individual rule names, their **Priority** and their **On** status. Click the box to check the desired rules. Click **Ok** to accept and return to the configuration dialog.
- Drag and drop rules from the MRM Ruleset to the Initialization area or Post-Simulation area.



Note: Note that the order of addition into the display does not affect the rule ordering; the order is strictly consistent with priorities defined in the MRM Rules.

- If you would like to view a rule directly from the **Iterative Runs** tab, double click on the rule's name to open the Rule Editor.
- Consider activating the Continue After Abort option. If you want the Post-Simulation Rules to execute and possibly the next iteration to begin after an iteration is pre-maturely aborted (for any reason), check the box.
- Select the Max Iterations spinner and adjust it using the up / down arrows or by typing a value in the field to define the maximum number of iterations to run. A fully configured Iterative Runs tab is shown to the right.
- Still in the MRM Configuration dialog, click OK to accept the configuration settings and close the dialog or Apply to accept them but keep the dialog open. Alternatively, click Cancel to discard the configuration settings and close the dialog or Reset to discard them but keep the dialog open returning the settings to the condition present at the last accepted state.
- Returning to the Multiple Run Control dialog, click the check mark button to Apply the Configuration edits.

Remember, the Post-Simulation Rules must make a significant change in values (for example, through a rule assignment or import of data through an input DMI) at the end of each run for the controller to prescribe the next iteration. If the Post-Simulation Rules do not set any values, or if they set values sufficiently



similar to the previous simulation run, the controller will assume that the goal of the iterations has been achieved and stop.

Also be aware that the MRM rules execute differently from the basic ruleset. When the MRM rules execute, each fires once and only once according to the execution order specified. There is no dependency or resolution functionality. In fact, they will all fire even if one of them aborts.

Integer indexed Series Slots work very well with Iterative MRM. Using these slots, the user can store inputs, outputs, or intermediate results based on the index of the run. For example, after each iterative run, the user could store the total volume of water released in an integer index slot in the row corresponding to the run index. At the end of the entire run, all of these volumes are stored and can be reviewed. The GetRunIndex predefined function can be used to get the index of the next iterative run. For more information on Integer Indexed Series, click [HERE \(Slots.pdf, Section 3.4\)](#). For more information on the GetRunIndex function, click [HERE \(RPLPredefinedFunctions.pdf, Section 66\)](#).

9. Policy Based Runs

The policy setup of the configuration (None, Rules, or Constraints) must be selected from the Policy section of the Multiple Run Editor. Selecting Rules will enable the Rulebased Simulation controller when the multiple run is started. Selecting Constraints will enable the Optimization controller when the multiple run is started. Note that only Rules *or* Constraints can be selected, not both.

- Open the MRM Dialog by selecting **Control** ➤ **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar.
- In the Multiple Run Control dialog highlight the desired configuration and select **Configuration** ➤ **Edit...** or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...**

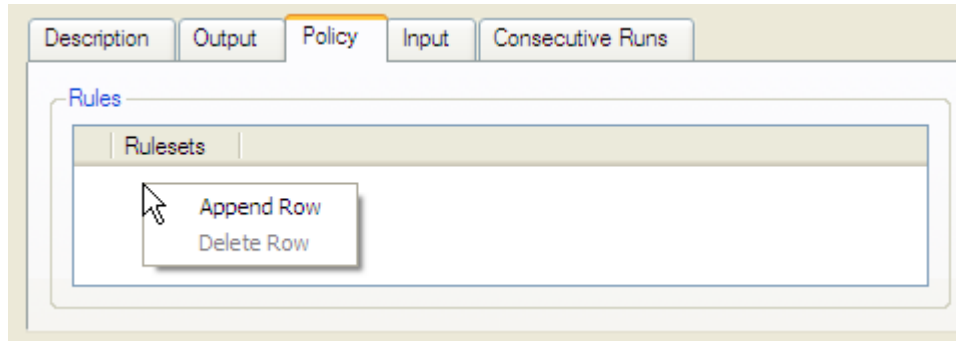


9.1 Rulebased Runs

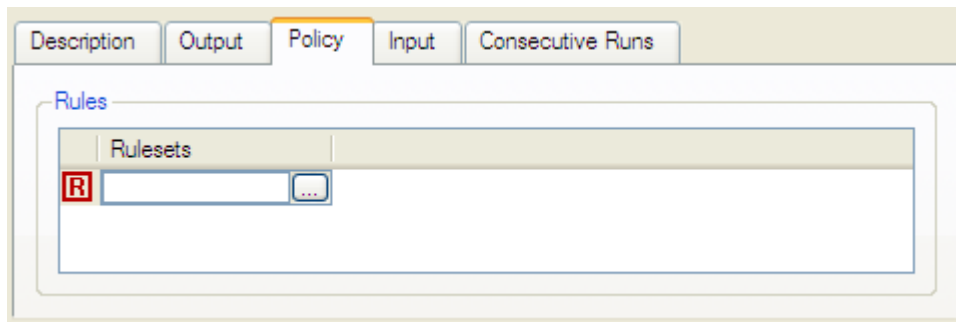
Rulebased multiple runs are runs in which there are one or more rulesets. The user specifies the ruleset to use for each run. Variations in rules can be a function of differences in content, priorities, or both.

- Select Rules under in the Policy section of the Multiple Run Editor and click to the Policy tab

- Append a new row by right clicking in the blank area under the Rulesets label and selecting Append Row from the ensuing pop-up context menu

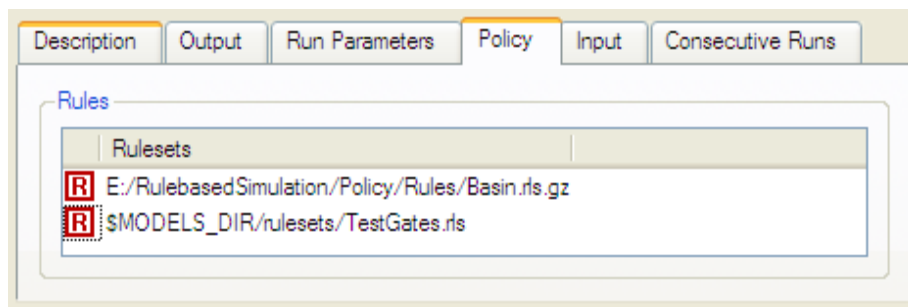


- Select the ruleset by clicking twice in the field next to the red R. This brings up a field in which you can either select the ruleset by browsing with the file chooser or you can type in the path and name of the ruleset directly.
 - To browse with a file chooser to the ruleset, click on the ellipse button next to the blank ruleset name field to bring up the file chooser



- Typing in the path and name of the ruleset directly allows for the user of environment variables in the path. Take care to spell the entire path correctly.
- Repeat the last two steps for each ruleset to be included in the MRM configuration

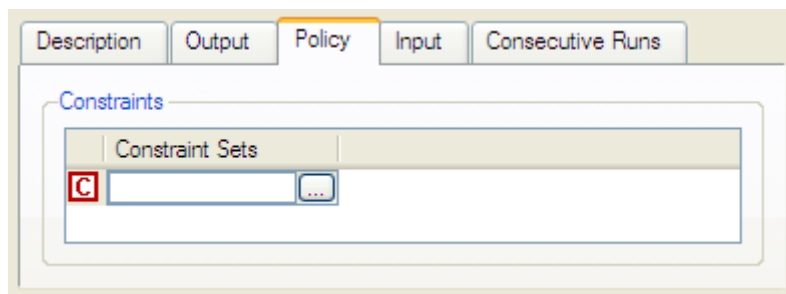
In the figure below, the first ruleset was specified by browsing to the E: network drive, the second ruleset was specified by typing the path in directly and using the environment variable MODELS_DIR



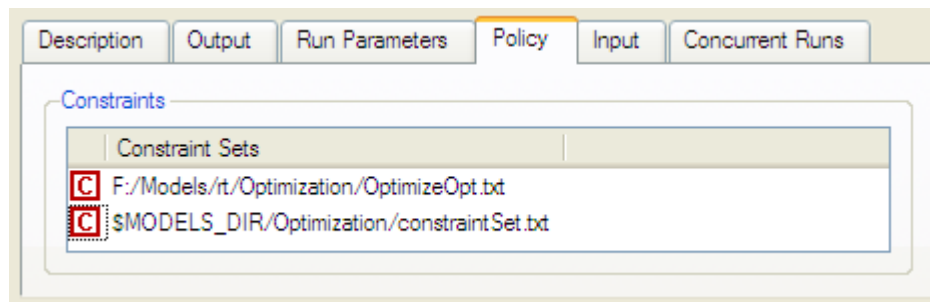
9.2 Constraint Set Runs

These runs are optimization runs in which there are one or more optimization constraint sets. The user specifies the constraint set to use for each run. Variations in constraint sets can be the result of changes in content, priorities, or both. The results of a optimization run might later serve as input for a simulation run (release schedules from the optimization as inputs for simulation).

- Select Constraints under in the Policy section of the Multiple Run Editor and click to the Policy tab
- Append a new row by right clicking in the blank area under the Constraint Sets label and selecting Append Row from the ensuing pop-up context menu
- Select the constraint set by clicking twice in the field next to the red C. This brings up a field in which you can either select the constraint set by browsing with the file chooser or you can type in the path and name of the constraint set directly.
 - To browse with a file chooser to the constraint set, click on the ellipse button next to the blank constraint set name field to bring up the file chooser



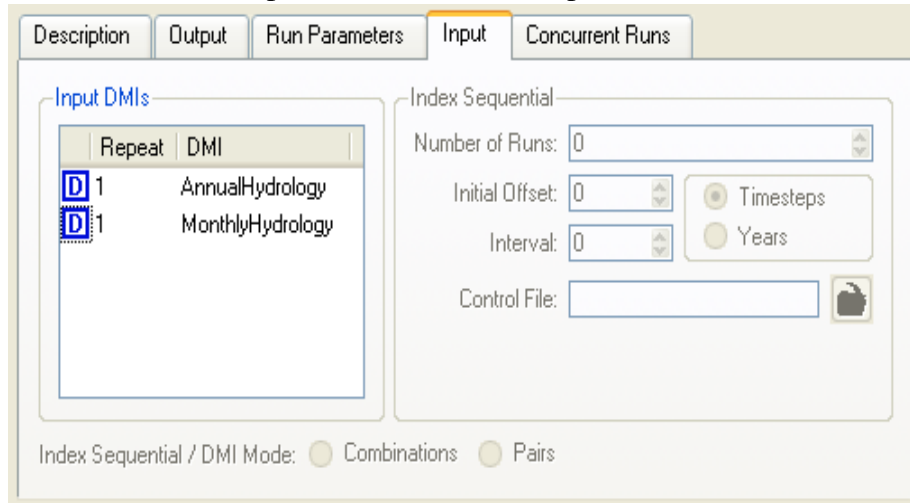
- Typing in the path and name of the constraint set directly allows for the user of environment variables in the path. Take care to spell the entire path correctly.
- Repeat the last two steps for each constraint set to be included in the MRM configuration



10. Input DMI Runs

Input DMI runs vary the input data for a predefined set of slots. For instance, ten runs might represent ten alternative release schedules for a reservoir. The user specifies the number of runs and a DMI for each run. The DMI loads the data into the model for each run. The DMI executables must be previously defined in the RiverWare DMI interface.

- Open the MRM Dialog by selecting **Control** ➔ **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar.
- In the Multiple Run Control dialog highlight the desired configuration and select **Configuration** ➔ **Edit...** or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...**
- Select Input DMIs under in the Input section of the Multiple Run Editor and click to the Input tab



- Append a new DMI by right clicking in the blank area in the Input DMIs section and selecting Append Row from the ensuing pop-up context menu
- Select the input DMI executable by clicking twice in the blank DMI field next to the blue D. Click on the ellipse button and chose the DMI from the ensuing list of previously defined input DMIs
- If the DMI executable will be called more than one time, change the number in the Repeat field by clicking twice and toggling the up/down arrows.
 - DMIs might need to be called more than once in a MRM run, for example, with runs that execute multiple traces using the same DMI. In this situation, the DMI executable should reference the last command line parameter passed from RiverWare: `-STrace=traceNumber`.
- Repeat the last three steps for each input DMI to be included in the MRM configuration.

11. Index-Sequential Runs

Index Sequential runs use an input time series which is systematically shifted between runs. The user specifies the number of runs, the interval (number of timesteps) by which the input data is shifted from one run to the next, and an initial offset (number of timesteps) by which data is shifted for the first run. The slots with input data to be shifted are identified by the user in a DMI control file. This type of run is typically used to perturb (shift) historical hydrologic data in order to be able to conduct statistical analysis of the results.

Index Sequential implies that, given run start time, run timestep, run duration, number of runs, and time offset, input time series can be systematically perturbed between runs. For example, if an Index Sequential run is defined as:

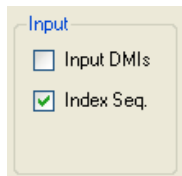
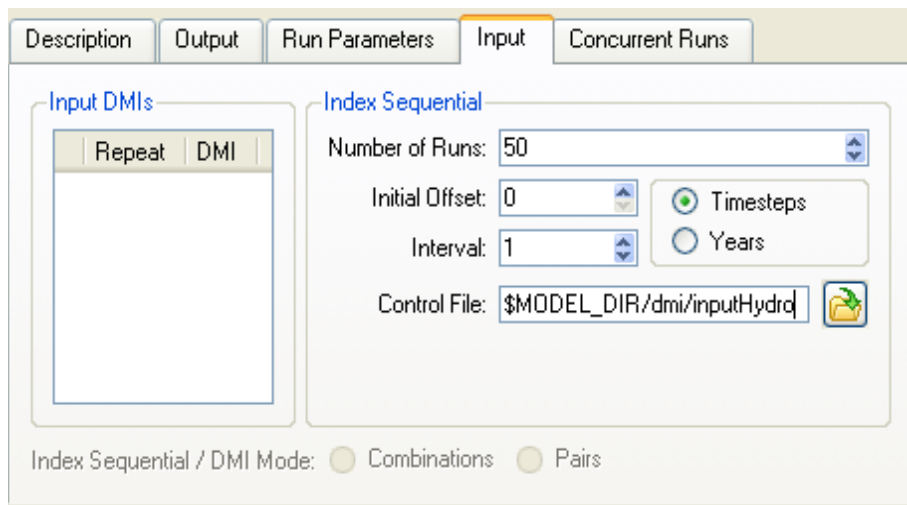
- *input time-series:*
original value vector: 1, 2, 3, 4, 5, 6, 7, 8, 9
original time vector: Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep
- *offset:* 4 timesteps
- *interval:* 1 timestep
- *n_of_runs* = 3

then, the following sequence of runs occurs:

Run 1: Jan = 5, Feb = 6, Mar = 7, Apr = 8, May = 9, Jun = 1, Jul = 2, Aug = 3, Sep = 4
 Run 2: Jan = 6, Feb = 7, Mar = 8, Apr = 9, May = 1, Jun = 2, Jul = 3, Aug = 4, Sep = 5
 Run 3: Jan = 7, Feb = 8, Mar = 9, Apr = 1, May = 2, Jun = 3, Jul = 4, Aug = 5, Sep = 6

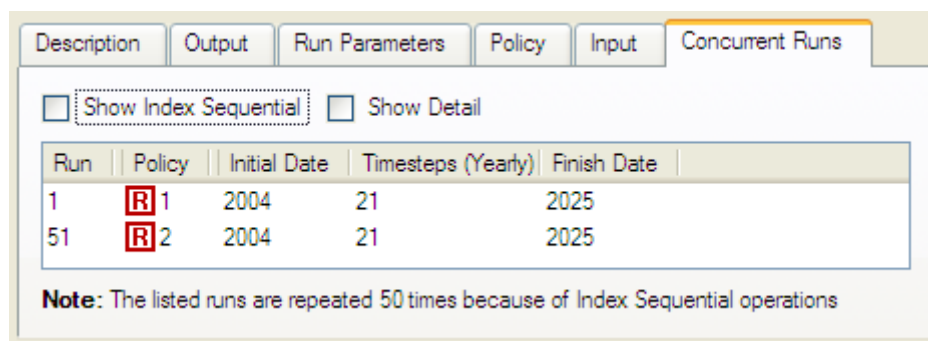
To setup an Index-sequential run:

- Open the MRM Dialog by selecting **Control** ➔ **MRM Control Panel...** from the main RiverWare menu bar or click on the MRM button on the toolbar.
- In the Multiple Run Control dialog highlight the desired configuration and select **Configuration** ➔ **Edit...** or right-mouse click on the highlighted configuration to bring up a context menu and select **Edit...**
- Select Index Sequential in the Input section of the Multiple Run Editor and click on the Input tab

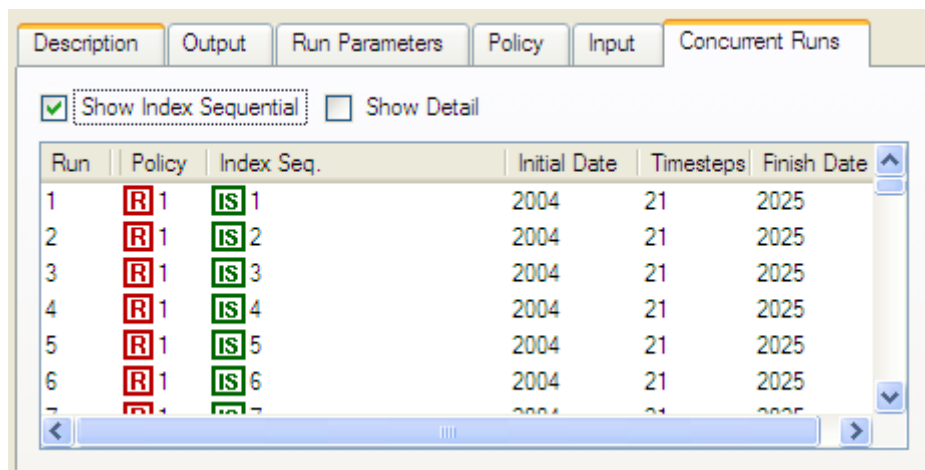


- Specify the Number of Runs by typing in the value or by toggling the up/down arrows
- Specify the Initial Offset by typing in the value or by toggling the up/down arrows

- Specify the Interval by which the data is shifted from one run to the next by typing in the value or by toggling the up/down arrows in the Interval field
- Specify the units for the Initial Offset and Interval as Timesteps or Years
- Specify the Input DMI Control File by typing in the path directly or by using the file chooser to select it
 - Typing in the path and name directly allows for the use of environment variables. Take care to spell the entire path correctly.
- Click on the Concurrent Runs tab and check that the correct combination of policy sets and Input DMIs are set up for the index sequential run



- Toggle the “Show Index Sequential” box at the top of the Concurrent Runs tab to view each index sequential run listed individually



12. Combined Runs

Combined runs systematically combine multiple runs of specific types.

12.1 Index Sequential / DMI Mode

In very specific circumstances, it is possible to alter the mode of how many MRM runs result from the combination of input DMIs, policy sets, and Index Sequential runs. If Index Sequential has been selected *and* there are as many (or more) Input DMIs as Index Sequential runs *and* there are zero or one rulesets selected, then it is possible to choose the Index Sequential / DMI Mode: either Combinations or Pairs.

Combined runs are defined as the Cartesian product of all the involved base-type runs. For example, a combined run consisting of two rule based and four Input DMI runs, results in $2 * 4 = 8$ model runs. All combinations of Run Types are possible except for the following:

- Rule-based (Rulebased Simulation) and Constraint-based (Optimization) runs.
- Index-Sequential and Consecutive runs.

The order in which individual runs within a combined multiple run are executed is determined by the precedence level of each mode. Order of precedence (from highest (1) to lowest (3)) is as follows:

1. Input DMI runs,
2. Rule-based and Constraint-based runs, and
3. Index-Sequential and Consecutive runs

Elements of lower precedence are iterated before elements of higher precedence. For example, in case of a combined run containing two Input DMI runs and two Rule-based runs, the following sequence of four runs is made:

1. Input DMI run 1 && Rule-based run 1.
2. Input DMI run 1 && Rule-based run 2.
3. Input DMI run 2 && Rule-based run 1.
4. Input DMI run 2 && Rule-based run 2.

The Combinations mode results in the default behavior, where the total number of runs is the product of the number of policy sets, the number of input DMIs, and the number of index sequential runs:

$$\text{\#of MRM runs} = \text{\#of Policy Sets} * \text{\#of Input DMIs} * \text{\#of Index Seq Runs}$$

The Pairs mode results in the total number of runs equalling the number of pairs of Input DMIs and Index Sequential runs. If the number of input DMIs does not match the number of Index Sequential runs, then the number of index sequential runs equals the total number of possible pairs, (i.e., the minimum of the number of input DMIs and the number of Index Sequential runs). The Pairs mode is necessary to run the CRSS Lite model.

$$\text{\#of MRM runs} = \min(\text{\#of Input DMIs}, \text{\#of Index Seq Runs})$$

12.2 I/O-optimization Runs

Input/Output (I/O)-optimization runs are a special type of combined multiple run. The input/output character of I/O-optimization runs is in the fact that the values of the slots of the last timestep of run n serve automatically as the initial values for run $n+1$. I/O-optimization is somewhat complicated in that after optimization of run n , a simulation of that same run needs to be carried out in order to provide enough initial data for run $n+1$. The results from the optimization serve as input values for the simulation.

Based on this information, the following sequence for I/O optimization runs can be formulated:

1. Optimize based on current time horizon.
2. Save values for the current time horizon.
3. Simulate for the current time horizon.
4. Instantiate new time horizon.

Go back to step 1 and repeat sequence.