



# Getting the Most from RSS

---

## Going Beyond the Documentation

Don McCormack – Global Technical Enablement

# Blocks

## General Properties

- The use case for all blocks except Basic is as a composite, composed of two or more subunits.
- Events and Actions are applied to the entire block. Components within the block can't be treated individually and, except for Standby, all components must have the same distributional properties.
- Subunits for all composites except Series operate in parallel. They differ by the number of components running, when the block fails, and how the block ages.

# Blocks

## General Properties

Block	# Components Running	Fails When	Component Aging
Series	All	One unit fails	Equal
Parallel	All	All units fail	Equal
K out of N	All	Fewer than K operational units	Equal
Standby	K	Fewer than K operational units Switch failure	Variable
Stress Sharing	All	All units Fail Switch failure	Variable

# Basic Blocks

## Arranged as Composite Blocks

- Basic blocks can easily be arranged to act like Series, Parallel, and K out of N blocks. It is much harder to use them for Standby or Stress Sharing blocks with more than two subunits.
- Example: Basic blocks in a K out of N arrangement.
  - Subunits in a K out of N block can't be replaced when they fail, you have to wait until the block fails. Basic blocks are arranged as a 3 out of 5 block where each block is Weibull(2000,1) and takes 40 hours to replace. The comparable K out of N block uses the same distribution and also takes 40 to replace (despite all subunits having failed). The two k out of n arrangements are organized in series so the system fails if either arrangement fails. How much of a difference does this make?

# Events and Actions

## General Properties

- Events
  - Inservice is only available for Basic blocks.
  - Initialization, Block Failure, and System is Down can only be use once for any block.
- Actions
  - Install Used, Change Distribution, and If can only be used with Basic blocks. Minimal Repair can only be used with Basic and Series.
  - There is no limit to how many times an Action can be used for a given block
  - Actions can only be connected to other Actions

# Events and Actions

## Specific Properties

- Events
  - Initialization only occurs once, at the start of the simulation run.
  - System is Down occurs for every block that uses it, regardless of which block brought the system down.
- Actions
  - Turn on System turns on every available block (anything not down due to failure, action tied to Scheduled event or action tied to or Schedule action).
  - A block must be turned on manually after Install New, Install Used, or Change Distribution.
  - Turn on Block does not turn the system on if the block was turned off manually.

# Example

## Events and Actions

- Example: Basic blocks in a Standby arrangement with a cold startup and 30 minute startup time. Eight hour maintenance on the backup is performed after control is passed back to the main unit. 90% switch reliability. Exponential(500) and 8 hour repair/maintenance time for all blocks. Failing blocks are replaced with new. Compare to Standby block with same distribution, repair time, and switch reliability.

# Example

## Events and Actions

- Example: Two pumps operating in parallel with a preventive maintenance performed every two weeks taking 8 hours. The PMs are staggered by one week to keep from having both pumps down at the same time. If a pump fails, a minimal repair taking 4 hours is performed. Simulation is run for one year of usage.



# Stress Sharing

## Custom Stress Sharing

- By default, stress is shared equally among the subunits in the block. This has been implemented by altering the block aging and not the underlying distribution.
  - With no stress sharing, given  $n$  subunits in parallel, the first failure occurs at  $t_1$ , the second at  $t_2$ , etc.
  - With stress sharing, the first failure occurs at  $n*t_1$ , the second at  $n*t_1 + (n-1)*(t_2-t_1)$ , etc.
  - These multipliers are the same values supplied to the custom stress sharing function.
- Example: Custom stress sharing



Thanks!

---



# Additional Material

---

# JSL

## Nonparametric Column

- RSS makes a copy of the entire table containing the column used for Nonparametric and saves it with the RSS script
  - All columns are included, with only the Data Type, Modeling Type, and Format properties saved.
  - Formulas are replaced with the values in the table.
  - All table properties and row states are removed.
- Failure times for nonparametric distributions use random uniform deviates as quantile inputs for the nonparametric CDF. Failure times are bounded by the maximum and minimum values in the data.

# JSL

## RBD to RSS

- Both RBD and RSS are saved as JSL scripts. Both are set to execute on open. To access the code in JMP without execution you must remove the first line of code.
- To convert an RBD to an RSS:
  - Change **Reliability Block Diagram** to **Repairable Systems Simulation**.
  - Each **System Item** corresponds to a different diagram, the RSS is built from the first one. The string appearing as the first argument must match the string in **Open System Item**.
  - Library items cannot be easily incorporated into an RSS. The difficulty is in organizing the diagram elements so as not to be crowded. You can replace the entire **Subsystem** argument with **Basic ( )**.



[jmp.com](https://www.jmp.com)