

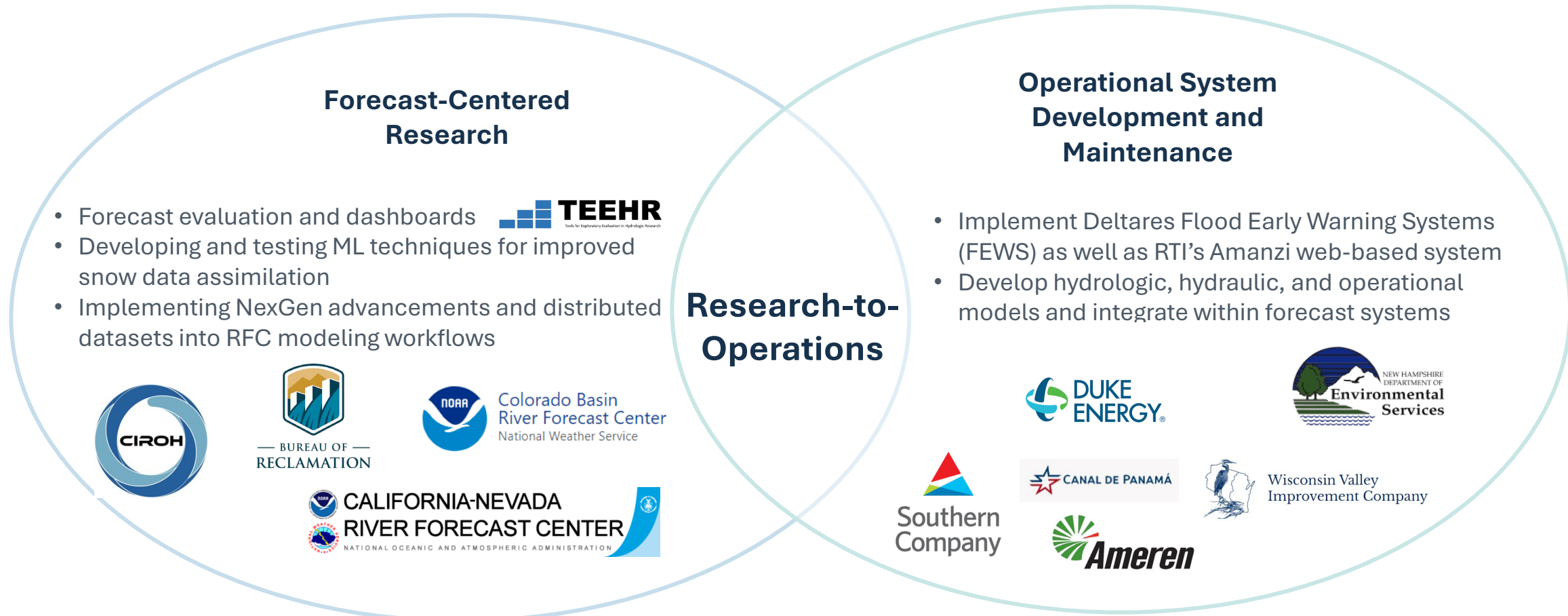
Practical Considerations for Developing RiverWare Models for Hydropower Operations

RiverWare User Group Meeting
August 27, 2026

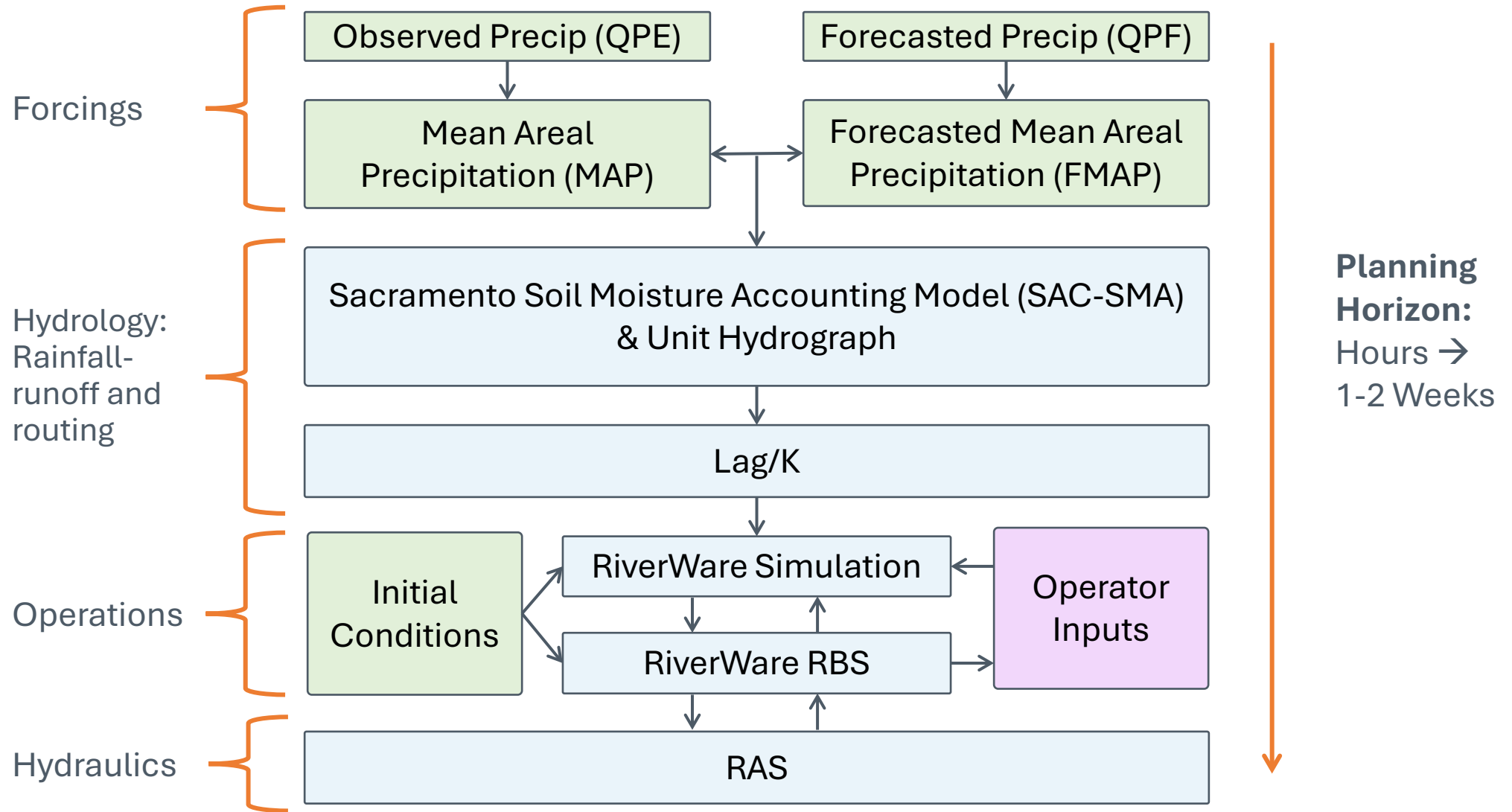
Abby Watson | RTI International

RTI's River Forecasting & Operations Team

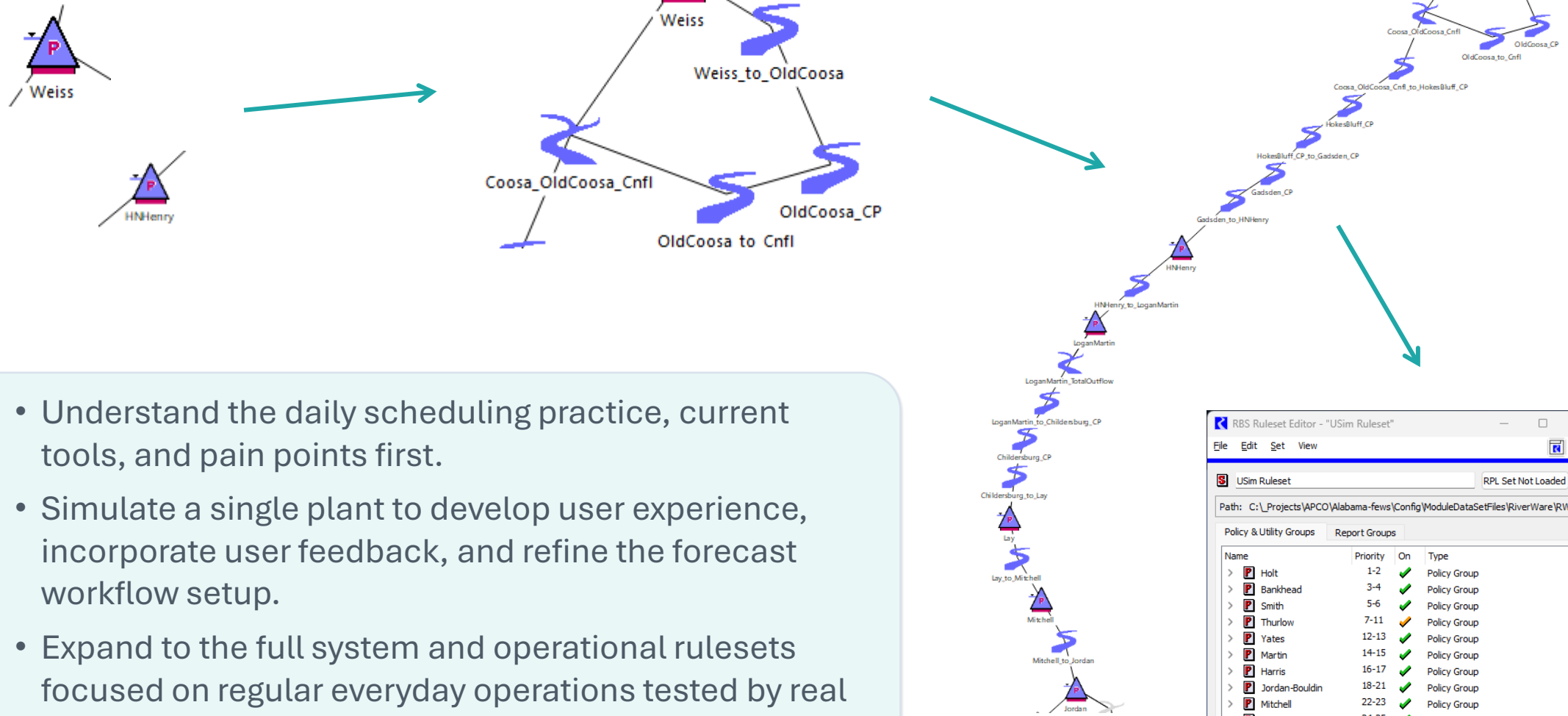
Water capabilities extend beyond to Integrated Water Resources Management, Water Supply Planning, Nature-Based Solutions, Hydrologic Risk Assessment, and Hydrometeorology



River Forecast Systems for Hydropower Operators



Start small.



- Understand the daily scheduling practice, current tools, and pain points first.
- Simulate a single plant to develop user experience, incorporate user feedback, and refine the forecast workflow setup.
- Expand to the full system and operational rulesets focused on regular everyday operations tested by real users under real events.

RBS Ruleset Editor - "USim Ruleset"

File Edit Set View

USim Ruleset RPL Set Not Loaded

Path: C:_Projects\APCO\Alabama-fews\Config\ModuleDataSetFiles\RiverWare\RW_ALF

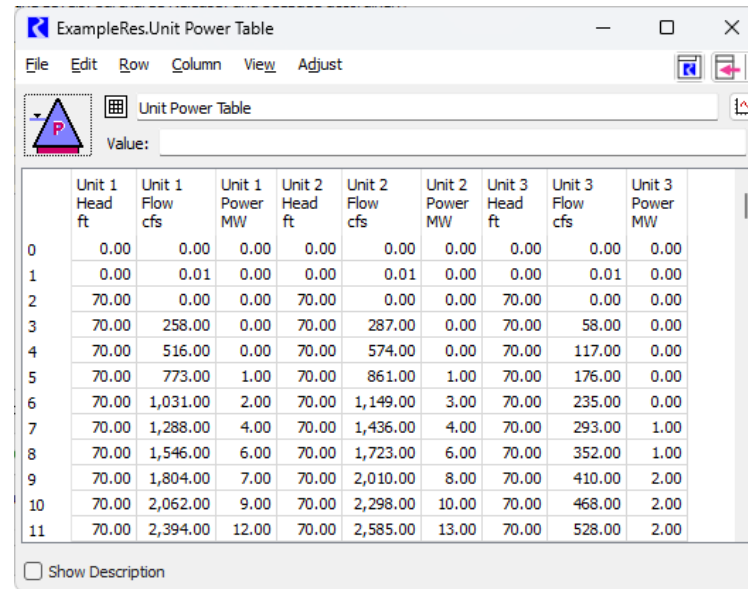
Name	Priority	On	Type
> Holt	1-2	✓	Policy Group
> Bankhead	3-4	✓	Policy Group
> Smith	5-6	✓	Policy Group
> Thurlow	7-11	✓	Policy Group
> Yates	12-13	✓	Policy Group
> Martin	14-15	✓	Policy Group
> Harris	16-17	✓	Policy Group
> Jordan-Bouldin	18-21	✓	Policy Group
> Mitchell	22-23	✓	Policy Group
> Lay	24-25	✓	Policy Group
> Logan	26-28	✓	Policy Group
> Henry	29-30	✓	Policy Group
> Weiss	31-33	✓	Policy Group
> Utility Group		✓	Utility Group

Show: ☐ Set Description ☐ Selected Description ☐ Set Notes ☐ Adv. Properties

Match power representation to operator needs.

Thinking in units of flow

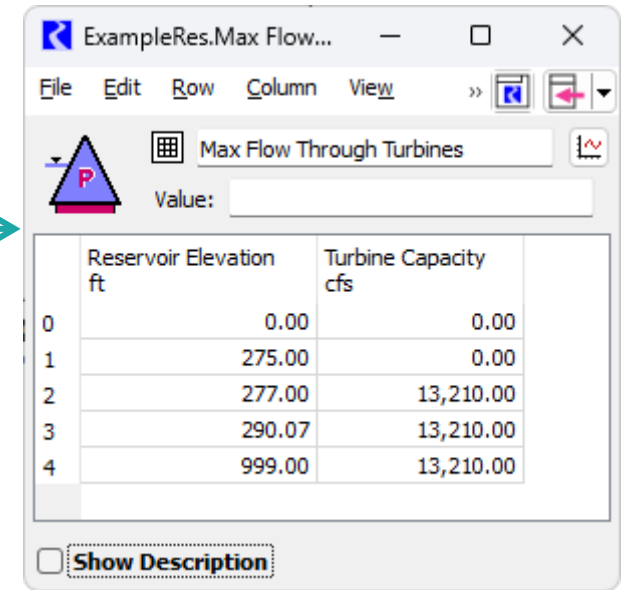
Unit Power Table Method too complex if operators think of efficient and maximum capacity values statically



ExampleRes.Unit Power Table

	Unit 1 Head ft	Unit 1 Flow cfs	Unit 1 Power MW	Unit 2 Head ft	Unit 2 Flow cfs	Unit 2 Power MW	Unit 3 Head ft	Unit 3 Flow cfs	Unit 3 Power MW
0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.01	0.00
2	70.00	0.00	0.00	70.00	0.00	0.00	70.00	0.00	0.00
3	70.00	258.00	0.00	70.00	287.00	0.00	70.00	58.00	0.00
4	70.00	516.00	0.00	70.00	574.00	0.00	70.00	117.00	0.00
5	70.00	773.00	1.00	70.00	861.00	1.00	70.00	176.00	0.00
6	70.00	1,031.00	2.00	70.00	1,149.00	3.00	70.00	235.00	0.00
7	70.00	1,288.00	4.00	70.00	1,436.00	4.00	70.00	293.00	1.00
8	70.00	1,546.00	6.00	70.00	1,723.00	6.00	70.00	352.00	1.00
9	70.00	1,804.00	7.00	70.00	2,010.00	8.00	70.00	410.00	2.00
10	70.00	2,062.00	9.00	70.00	2,298.00	10.00	70.00	468.00	2.00
11	70.00	2,394.00	12.00	70.00	2,585.00	13.00	70.00	528.00	2.00

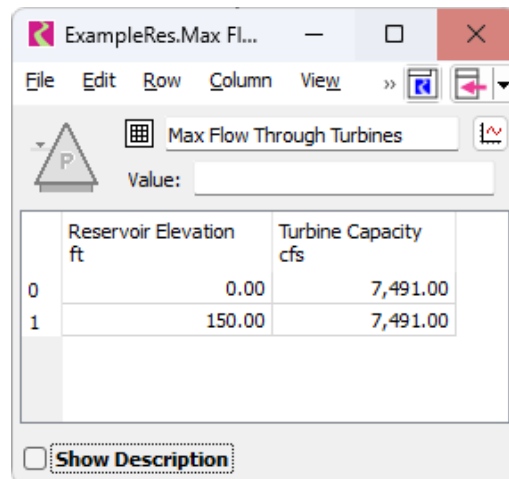
☐ Show Description



ExampleRes.Max Flow...

	Reservoir Elevation ft	Turbine Capacity cfs
0	0.00	0.00
1	275.00	0.00
2	277.00	13,210.00
3	290.07	13,210.00
4	999.00	13,210.00

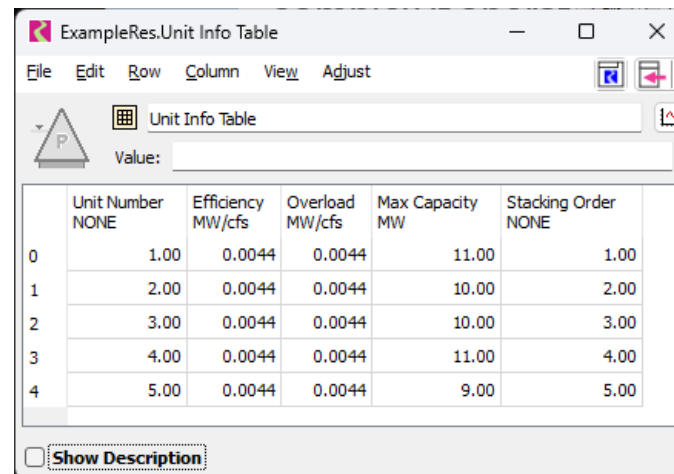
☐ Show Description



ExampleRes.Max Fl...

	Reservoir Elevation ft	Turbine Capacity cfs
0	0.00	7,491.00
1	150.00	7,491.00

☐ Show Description



ExampleRes.Unit Info Table

	Unit Number	Efficiency MW/cfs	Overload MW/cfs	Max Capacity MW	Stacking Order
0	1.00	0.0044	0.0044	11.00	1.00
1	2.00	0.0044	0.0044	10.00	2.00
2	3.00	0.0044	0.0044	10.00	3.00
3	4.00	0.0044	0.0044	11.00	4.00
4	5.00	0.0044	0.0044	9.00	5.00

☐ Show Description

Thinking in units of power

Manually covert power to flow using tabular data entry to mirror existing operator methods and thought processes

Use RPL to improve simulation model robustness.

RPL Viewer - USim Ruleset

File Edit Rule Statement View

Set ExampleRes Turbine Release

11 Set ExampleRes Turbine Release RPL Set Loaded

```
# Set the turbine release
WITH ( OBJECT res = ExampleRes ) DO
  WITH ( NUMERIC turbRel = LimitFlowToTurbineCapacity_USim ( res ,
    res . "User Input Turbine Release FEWS" [],
    @"t" ) ) DO
    res . "Turbine Release" [] = turbRel
    # Assume there is no spill over the Obermeyer
    res . "Regulated Spill" [] = 0.00 "cfs"
    IF ( ( res . "User Input Turbine Release FEWS" [] ) != turbRel ) THEN
      WARNING ( STRINGIFY res ) CONCAT " WARNING: User input turbine release was constrained by the model."
    END IF
  END WITH
END WITH
```

Show: ☒ Execution Constraint ☐ Description ☐ Notes ☒ Comments

Execute Rule Only When

```
# If the reservoir is being simulated
NOT IsNaN ( ExampleRes . "User Input Turbine Release FEWS" [] )
```

Use RiverWare RPL to validate and adjust user inputs

Use RiverWare **WARNINGS** to translate issues to operators

Use execution constraints to increase model flexibility

Use RPL to improve simulation model robustness.

Functions to constrain user inputs depend on reservoir characteristics

```
RPL Viewer
File Edit Rule Statement View
ExampleRes Energy
79 ExampleRes Energy RPL Set Loaded
WITH ( OBJECT res = GetObject ( "ExampleRes" ) ) DO
  WITH ( LIST turbRelPowerList = GetTurbineReleaseAndPower ( res , res . "Schedule_Energy" [ ] , @ "t" ) ) DO
    # If reservoir is provided an energy schedule, set the Turbine Release to satisfy the energy
    # request, subject to physical constraints
    res . "Turbine Release" [ ] = LimitOutflowToAvailableStorage ( res , turbRelPowerList < 0 > , @ "t" )
    res . "Power" [ ] = turbRelPowerList < 1 >
    # Use of regular Energy slot causes object to be overdetermined. Created a simulated energy slot
    # to hold output energy.
    res . "Simulated_Energy" [ ] = ( turbRelPowerList < 1 > ) * 1.00 "hour"
    # NOTE: Unit flow and power may not perfectly reflect total outputs if the plant is limited by
    # available storage (i.e., very close to full or very close to empty)
    WITH ( LIST unitNumTurbRelPowerList = GetUnitTurbRelAndPowerList ( res , turbRelPowerList < 1 > , @ "t" ) ) DO
      # Output turbine release and power by unit
      FOR ( LIST unit IN unitNumTurbRelPowerList ) DO
        res . "Flow By Unit" [ @ "t" , ( unit < 0 > ) - 1.00 ] = ( unit < 1 > )
        res . "Power By Unit" [ @ "t" , ( unit < 0 > ) - 1.00 ] = ( unit < 2 > )
      END FOR
    END WITH
  END WITH
END WITH

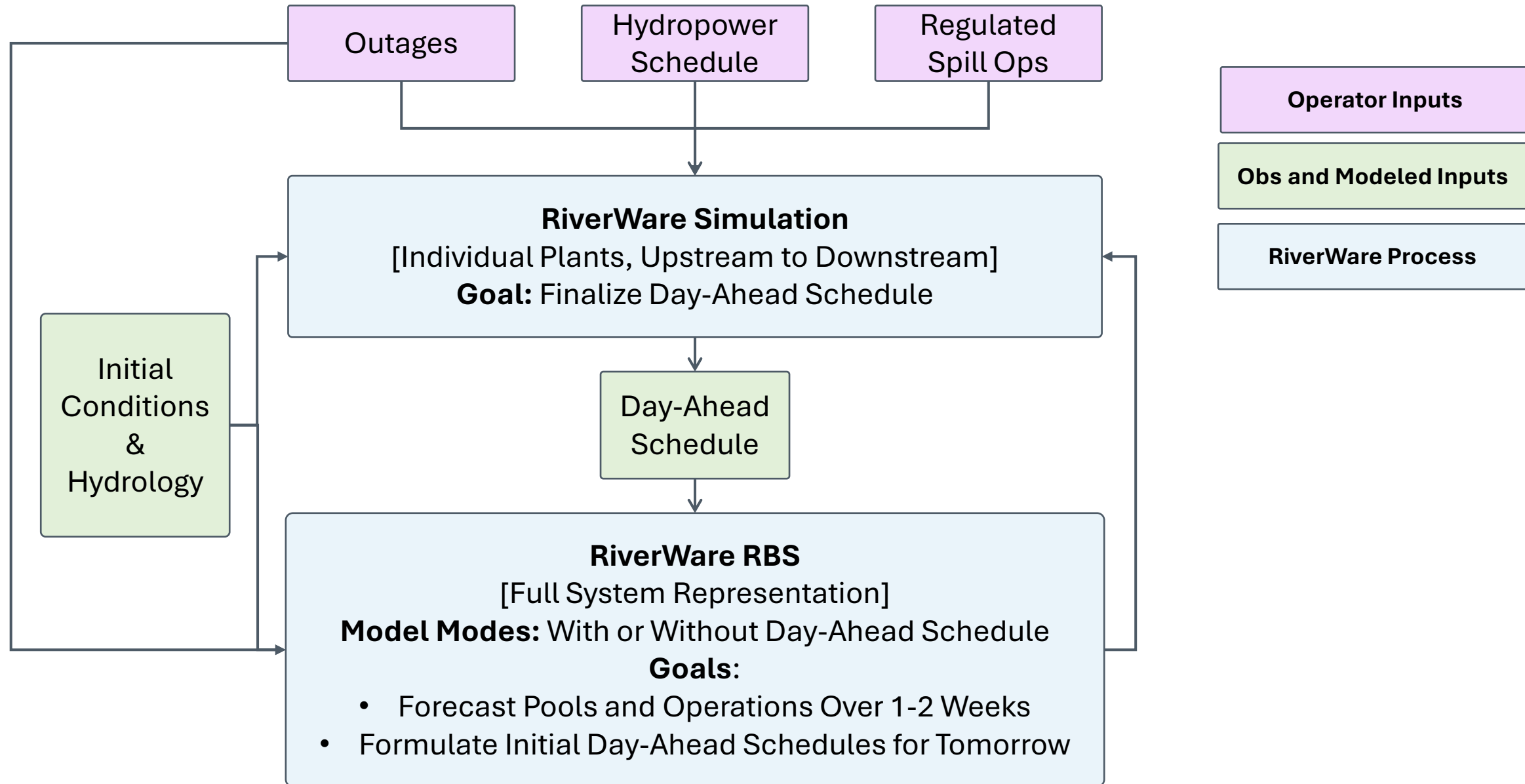
Show: ☒ Execution Constraint ☐ Description ☐ Notes ☒ Comments
Execute Rule Only When

# Force rule to fire once to mitigate iteration errors and check that a schedule is provided
NOT HasRuleFiredSuccessfully ( "Current Rule" )
AND WITH OBJECT res = GetObject ( "ExampleRes" ) DO
  NOT IsNaN res . "Schedule_Energy" [ ]
END WITH
```

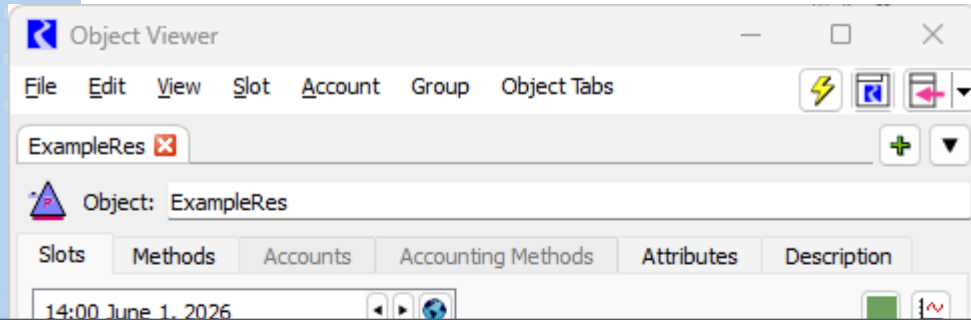
```
RPL Viewer - Example Rules
File Edit Function View
LimitOutflowToAvailableStorage
Arguments: OBJECT res, NUMERIC outflow, DATETIME ts Return Type: NUMERIC
RPL Set Loaded

# Limit outflow based on storage availability
WITH NUMERIC prevStorage = res . "Storage" [ ts - GetTimestep ( ts ) ] DO
  WITH NUMERIC maxStorage = Maxitem ( GetTableColumnVals ( res . "Elevation Volume Table" ,
    1.00 ,
    0.00 ,
    numRows ( res . "Elevation Volume Table" ) - 1.00 ) ) * 0.99 DO
    WITH NUMERIC requiredOut = VolumeToFlow ( maxStorage - prevStorage + FlowToVolume ( TotalInflow ( res , ts ) , ts ) , ts ) DO
      # Ensure we do not release a negative value
      Max ( # Release the minimum between:
        # - Desired outflow
        # - Outflow to hit max storage
        # - Outflow to perfectly empty the reservoir
        Minitem ( {
          outflow ,
          requiredOut ,
          VolumeToFlow ( prevStorage + FlowToVolume ( TotalInflow ( res , ts ) , ts ) , ts ) * 0.98
        } ,
        0.00 "cfs"
      )
    END WITH
  END WITH
END WITH
```

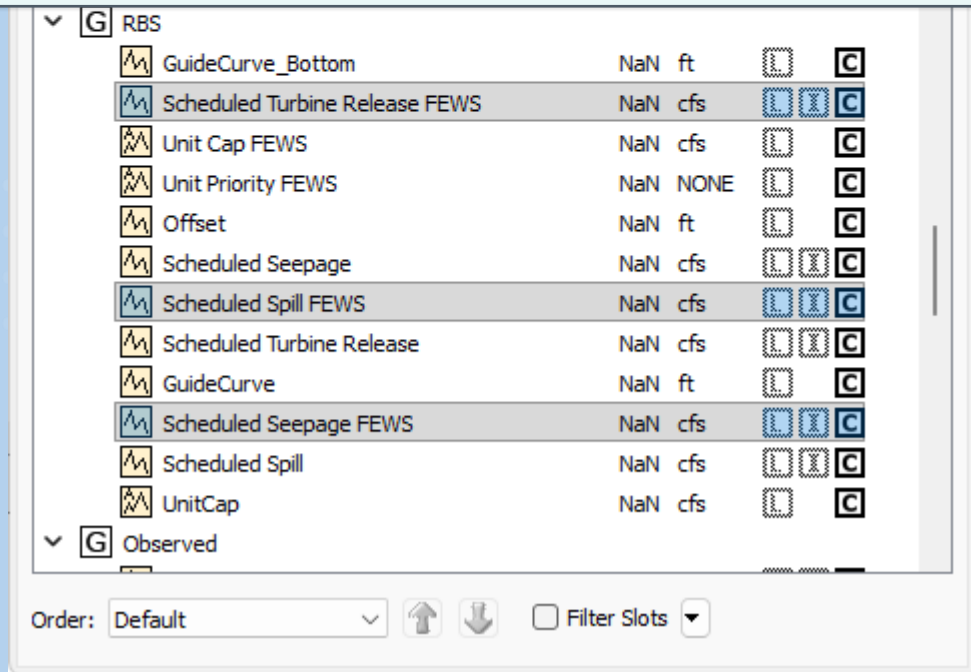
Leverage RiverWare's different run capabilities.



Allow operators control over outputs.

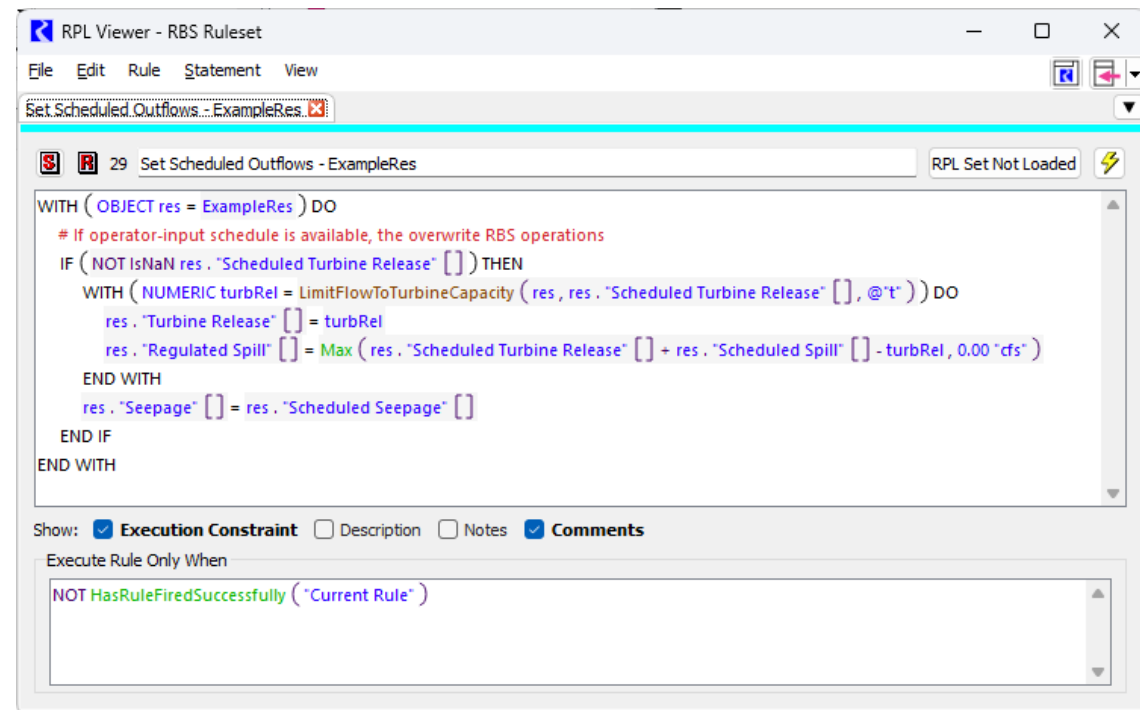


Read in scheduled operations as time series slots

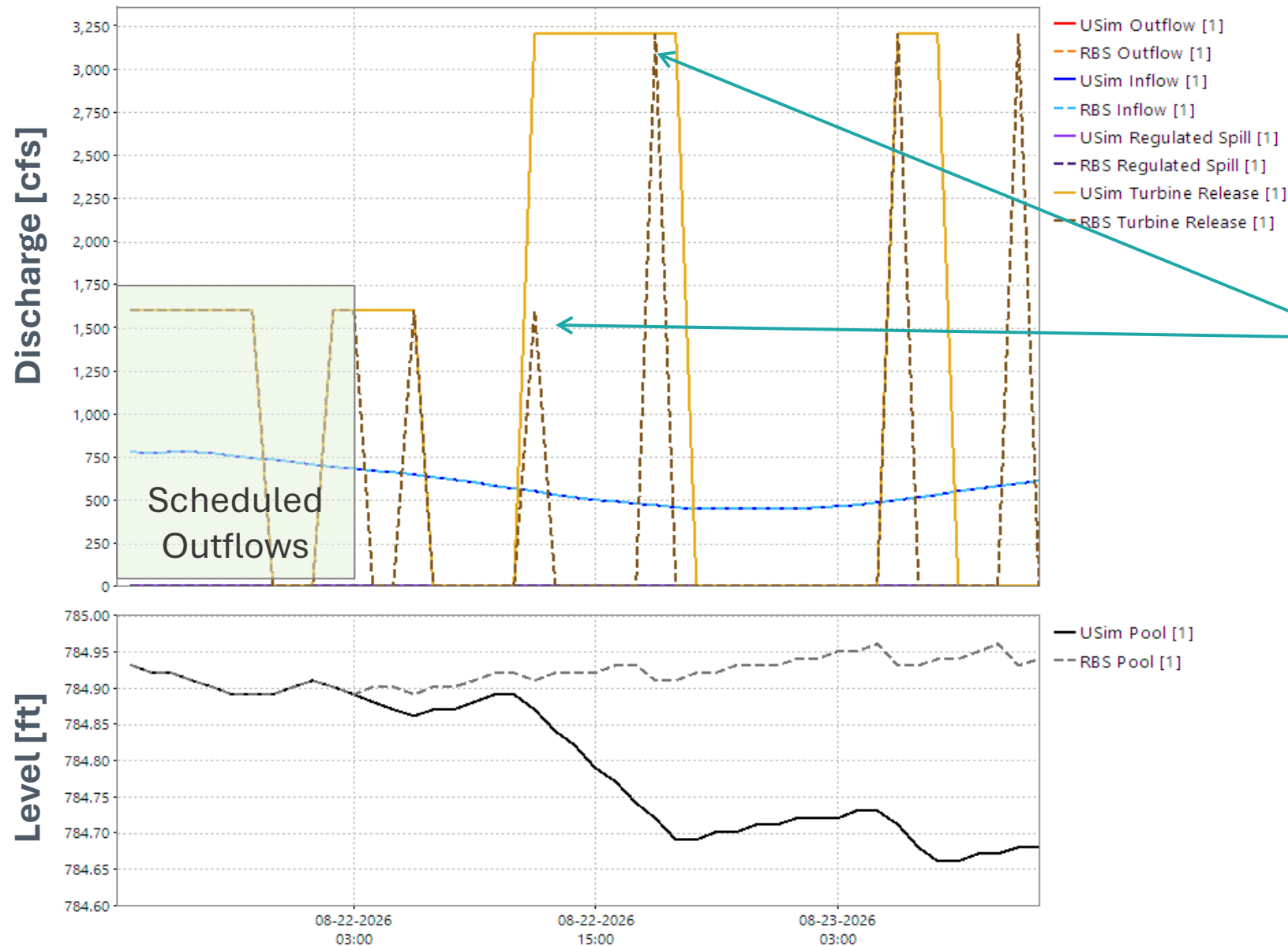


ExampleRes		
Set Scheduled Outflows - ExampleRes	29	✓
Set Spillway Flow - ExampleRes	30	✓
Spillway Flow - ExampleRes Step 3	31	✓
Spillway Flow - ExampleRes Step 2	32	✓
Spillway Flow - ExampleRes Step 1	33	✓
Normal Power - ExampleRes	34	✓
Initial Flow - ExampleRes	35	✓

Highest priority rule to overwrite RBS operations



Allow operators control over outputs.



Before first 4am

Alignment of operations during first day of operations

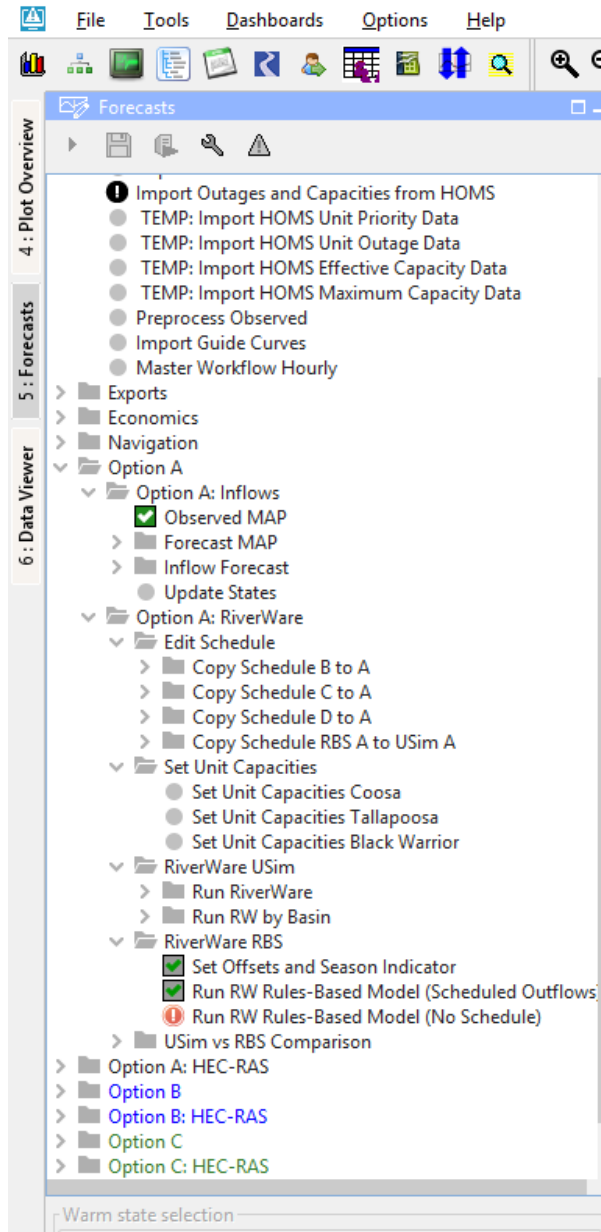
After first 4am

Operations in RBS model vary from the User Input Simulation model

Solid Line = Simulation Model

Dashed Line = RBS Model

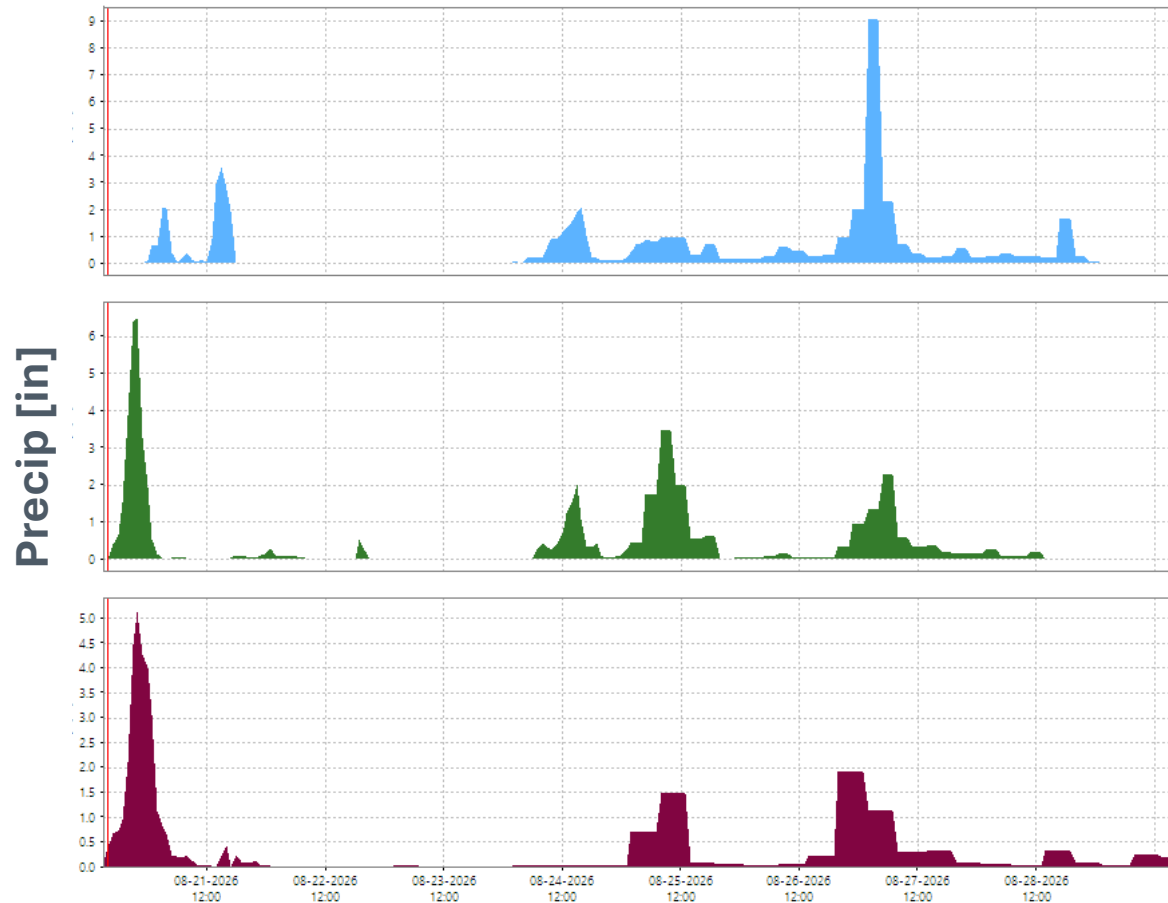
Create a flexible framework for scenario planning.



Four scenario planning workspaces: Options A-D

- 1 Ability to simulate different precipitation forecasts
- 2 Ability to revise outage conditions and unit capacities
- 3 Ability to revise targeted guide curve elevations
- 4 Ability to copy schedules across options
- 5 Ability to compare alternative scenarios directly

Create a flexible framework for scenario planning.



Selected Rainfall

A: No Rain (not shown)

B: HRRR/GFS Merge

C: GFS

D: NBM

- Provide a broader range of plausible outcomes and help quantify forecast uncertainty.
- Different precipitation models perform better for different storm types, locations, and forecast horizons.
- Redundancy improves operational resilience when a forecast source becomes unavailable.

Example Products

HRRR (High-Resolution Rapid Refresh)

ECMWF (European Centre for Medium-Range Weather Forecasts): IFS and AIFS

GFS (Global Forecast System)

Custom vendor forecasts (e.g., Upstream Tech)

NBM (National Blend of Models)

NAM (North American Mesoscale Model)

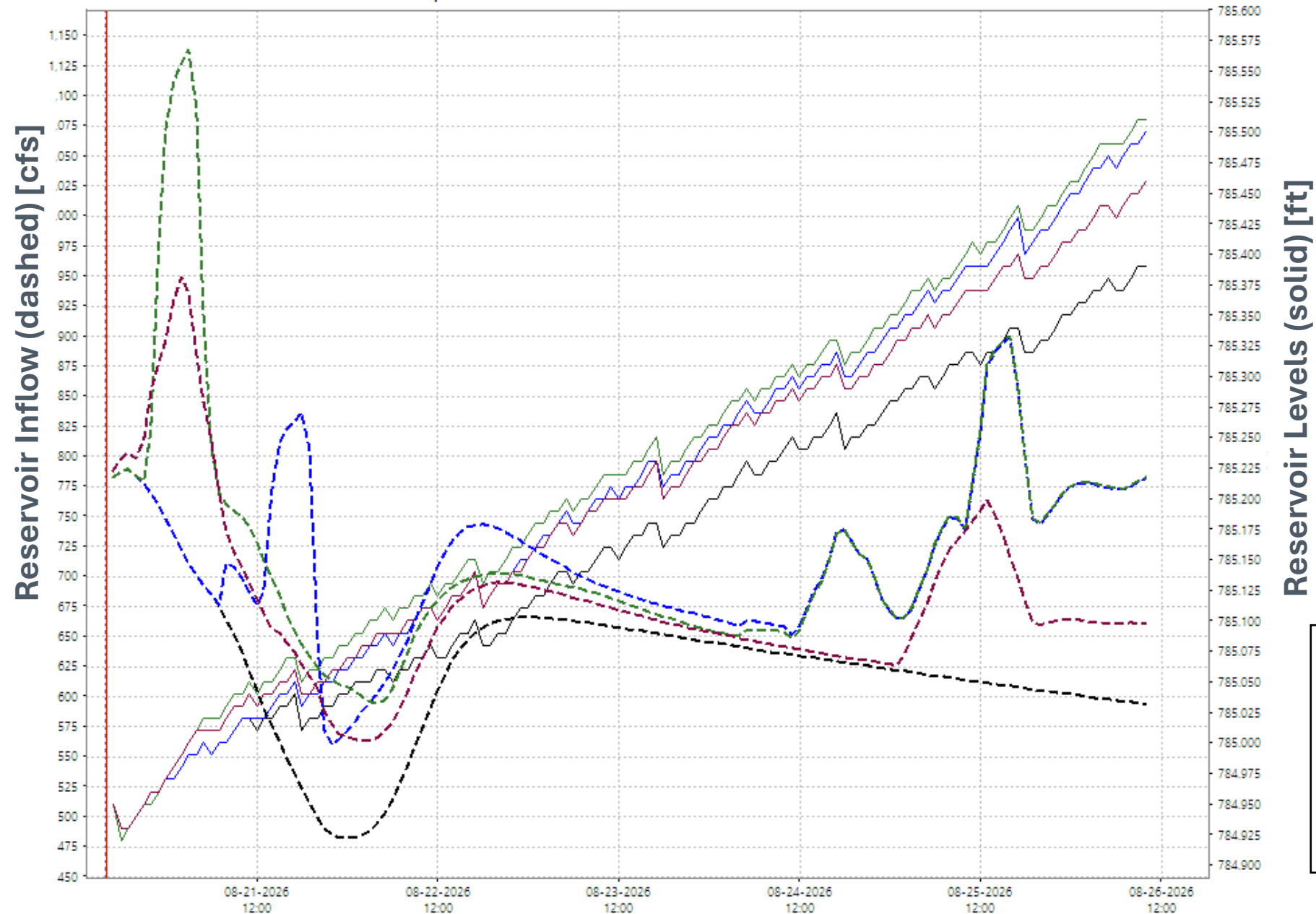
Create a flexible framework for scenario planning.



Different precipitation forcings yield different inflows.

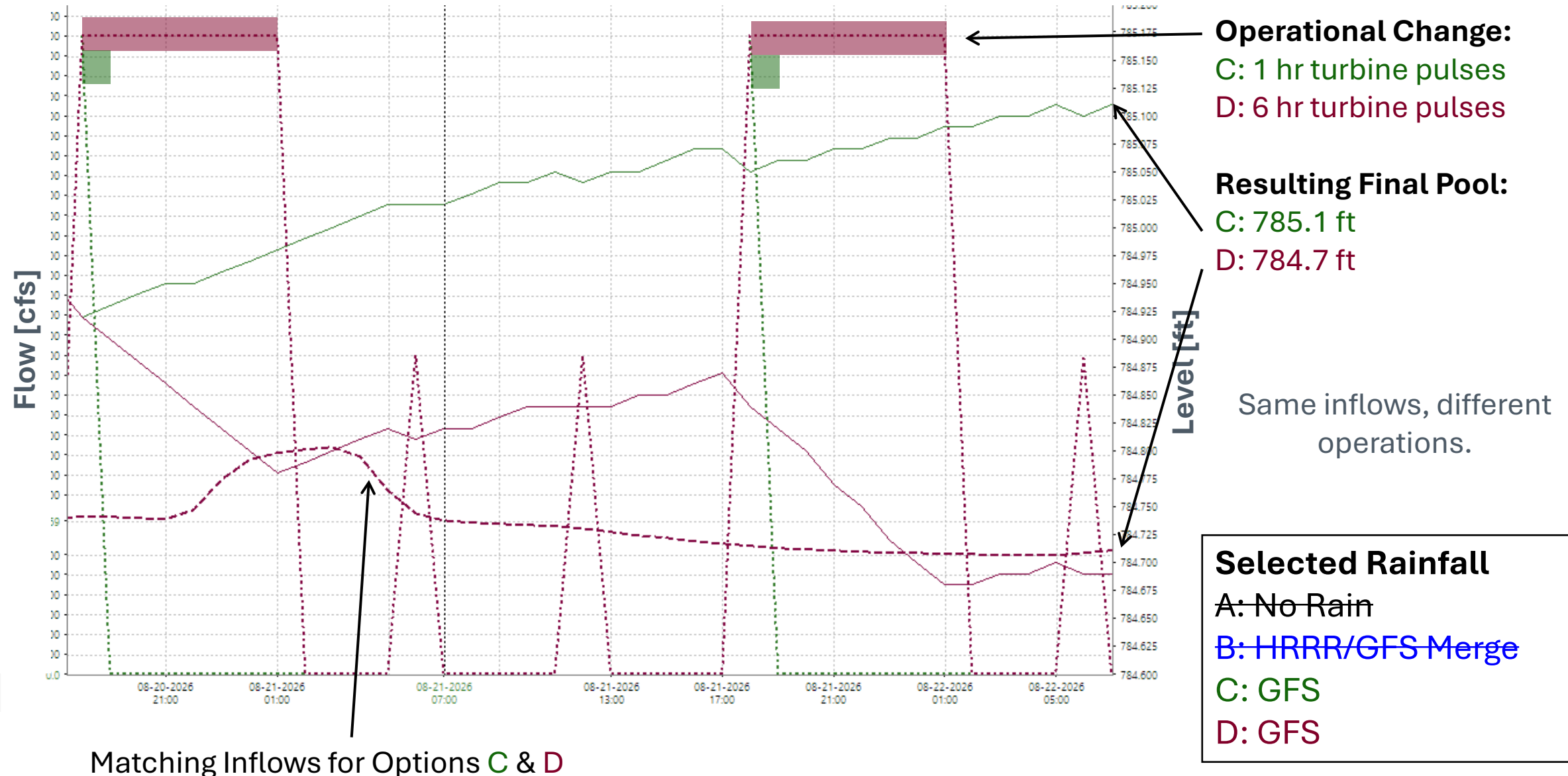
Selected Rainfall
A: No Rain
B: HRRR/GFS Merge
C: GFS
D: NBM

Create a flexible framework for scenario planning.



Selected Rainfall
A: No Rain
B: HRRR/GFS Merge
C: GFS
D: NBM

Create a flexible framework for scenario planning.



Summary.

1. Start small and build incrementally.
2. Match the model's power representation to how operators actually think.
3. Build models to be robust.
4. Provide operator control over modeled results.
5. Create scenario planning workspaces.

Abby Watson | RTI International
aawatson@rti.org