



**CADSWES**

University of Colorado

Center for Advanced Decision Support for Water and Environmental Systems

# RiverWare Optimization

---

RiverWare User Group Meeting  
August 13-14, 2008

# Outline

---

- What is RiverWare Optimization?
- RPL-based Optimization
- Integer Programming & Unit Power Modeling
- Other Developments

# Similarity Between Optimization and Rule Based Simulation

- Prioritized policy
  - From extreme conditions to normal operations
- Gradually remove the degrees of freedom from the solution

	Priority	On	Type
EndingTargets		✓	Policy Group
EndingTargets	13	✓	Rule
Trib_Targets	14	✗	Rule
Volume		✗	Policy Group
Volume	15	✗	Rule
Canal Slope		✗	Policy Group
CanalSlope	16	✗	Rule
Nomis Bull Run		✓	Policy Group
Nomis Bull Run	17	✓	Rule
Recreation Flows		✓	Policy Group
Recreation Flows	18	✓	Rule
Special Operations		✓	Policy Group
SpecialOperations	19	✓	Rule
Minimum Flows		✓	Policy Group
Trib Minimum Flows	20	✓	Rule
Main River Minimum Flows	21	✓	Rule
Navigation Min Flows	22	✓	Rule
Maximum Flows	23	✗	Rule
TimsFordElevs		✓	Policy Group
TimsFordElevs	24	✓	Rule

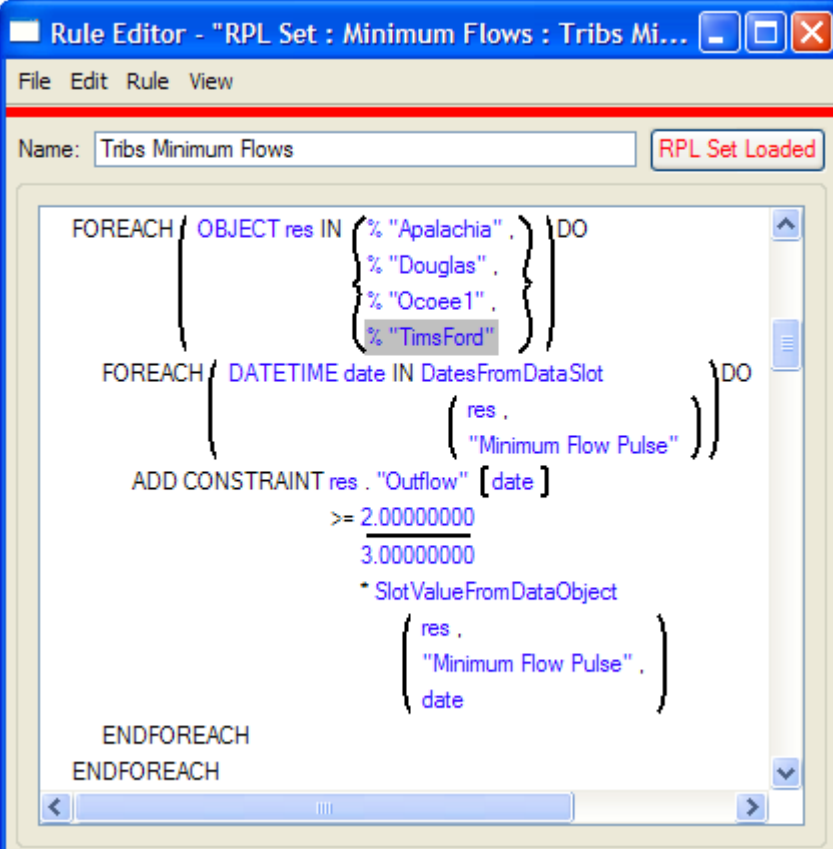
# Main Differences Between Optimization and Simulation

---

- Best solution vs. evaluating inputs or following rules
- Solve all time steps simultaneously vs. stepping through time steps
- Degrees of freedom
  - Equations and Unknowns vs. If-Then
- Approximation vs. Exact calculation
  - Nonlinear functions

# RPL-based Optimization

- Policy is in RPL
  - Some statements added for optimization
- Finished Last Year
  - Replicated results of the Old Optimization
  - Similar run time to Old Optimization

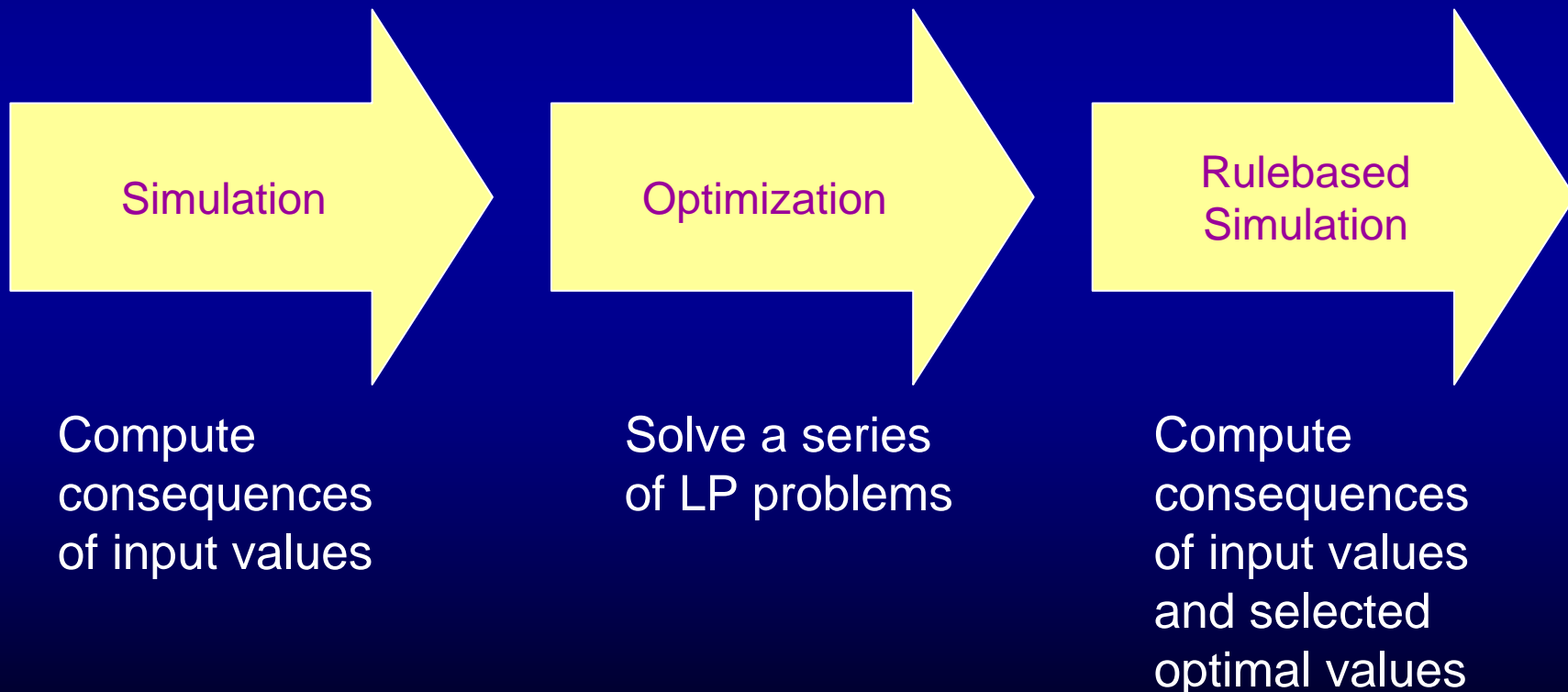


The screenshot shows a window titled "Rule Editor - 'RPL Set : Minimum Flows : Tribs Mi...". The window contains a menu bar with "File", "Edit", "Rule", and "View". Below the menu bar, there is a text field for "Name:" containing "Trib Minimum Flows" and a button labeled "RPL Set Loaded". The main area of the window displays RPL code:

```
FOREACH ( OBJECT res IN { "%Apalachia" ,  
                          "%Douglas" ,  
                          "%Ocoee1" ,  
                          "%TimsFord" } ) DO  
  
  FOREACH ( DATETIME date IN DatesFromDataSlot  
            ( res ,  
              "Minimum Flow Pulse" ) ) DO  
  
    ADD CONSTRAINT res . "Outflow" [ date ]  
                  >= 2.00000000  
                    3.00000000  
                    * SlotValueFromDataObject  
                      ( res ,  
                        "Minimum Flow Pulse" ,  
                        date )  
  
  ENDFOREACH  
ENDFOREACH
```

# A Typical Use of RPL-based Optimization

---



# Integer Programming for Unit Power Modeling

---

- Integer Programming allows additional modeling
  - Discrete variables, a unit is on or off
  - Non-convex nonlinear functions, e.g. power
  - Power as a function of flow and head
  - Cavitation and vibration avoidance zones
  - Policies to prevent “holes” in generation
  - Startup Cost
  - Ancillary Services (Only regulation for now)

# Solving Integer Programs

---

- Integer Programs are difficult
  - NP-Complete
- Branch and Bound
  - Compatible with the existing goal programming
  - Can prove an optimal solution with enough time
  - Expecting to use as a heuristic instead to reduce time
    - Near optimality with a measure of potential improvement



# Integer Programming Status

---

- Mathematical design is done
- Software design is mostly done
- Coding has begun
- Finish this year

# Other Developments

---

- Alternative Solver - postponed
- Developing a new training course for RPL-based optimization
- Additional ideas for improvement from TVA