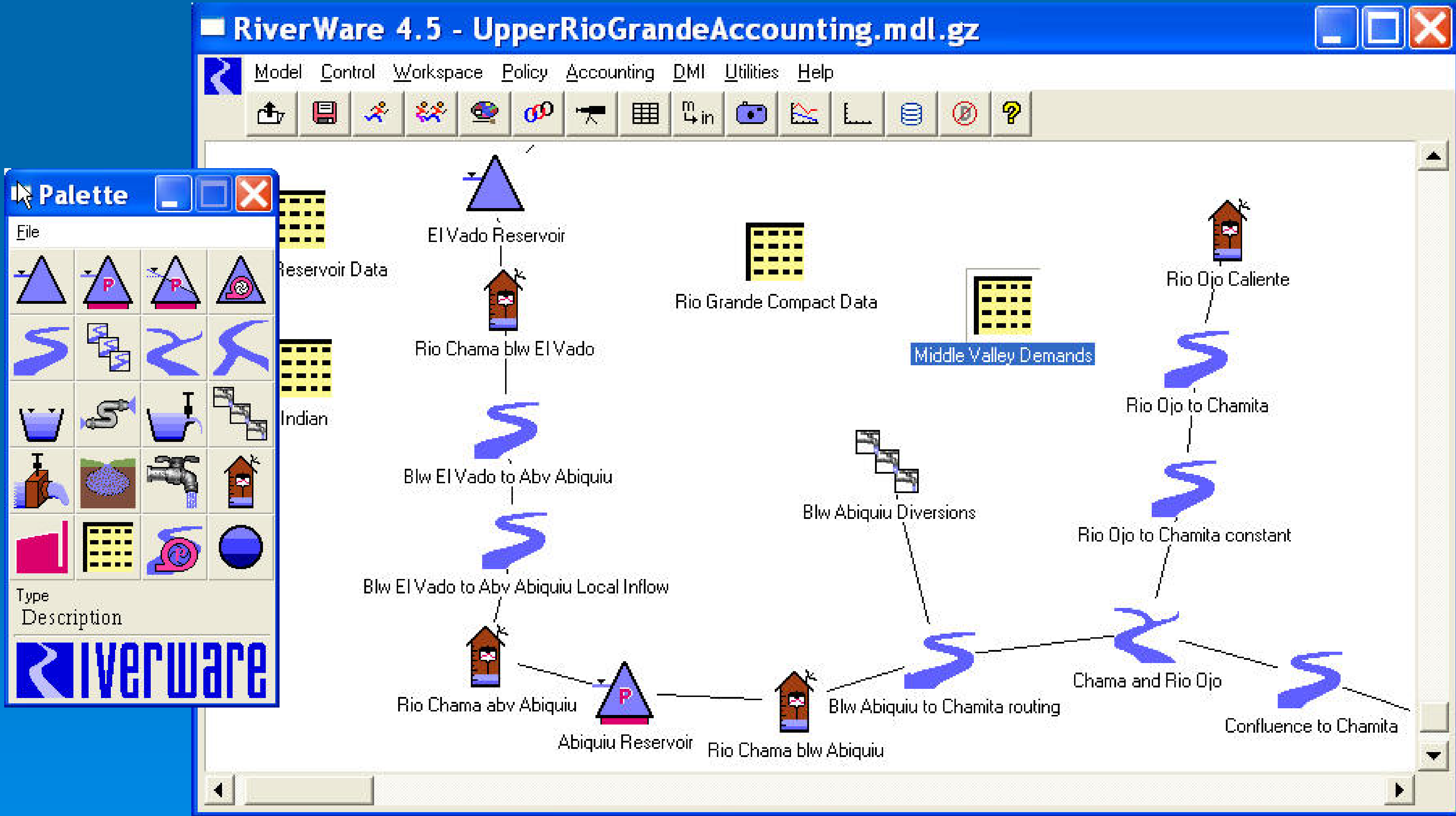
Part 2
Objects, Methods & Simulation

Object-Oriented Modeling Approach

- Objects represent features of the basin
- Objects contain their own physical process algorithms
- Objects contain their own data

RiverWare

The RiverWare Workspace

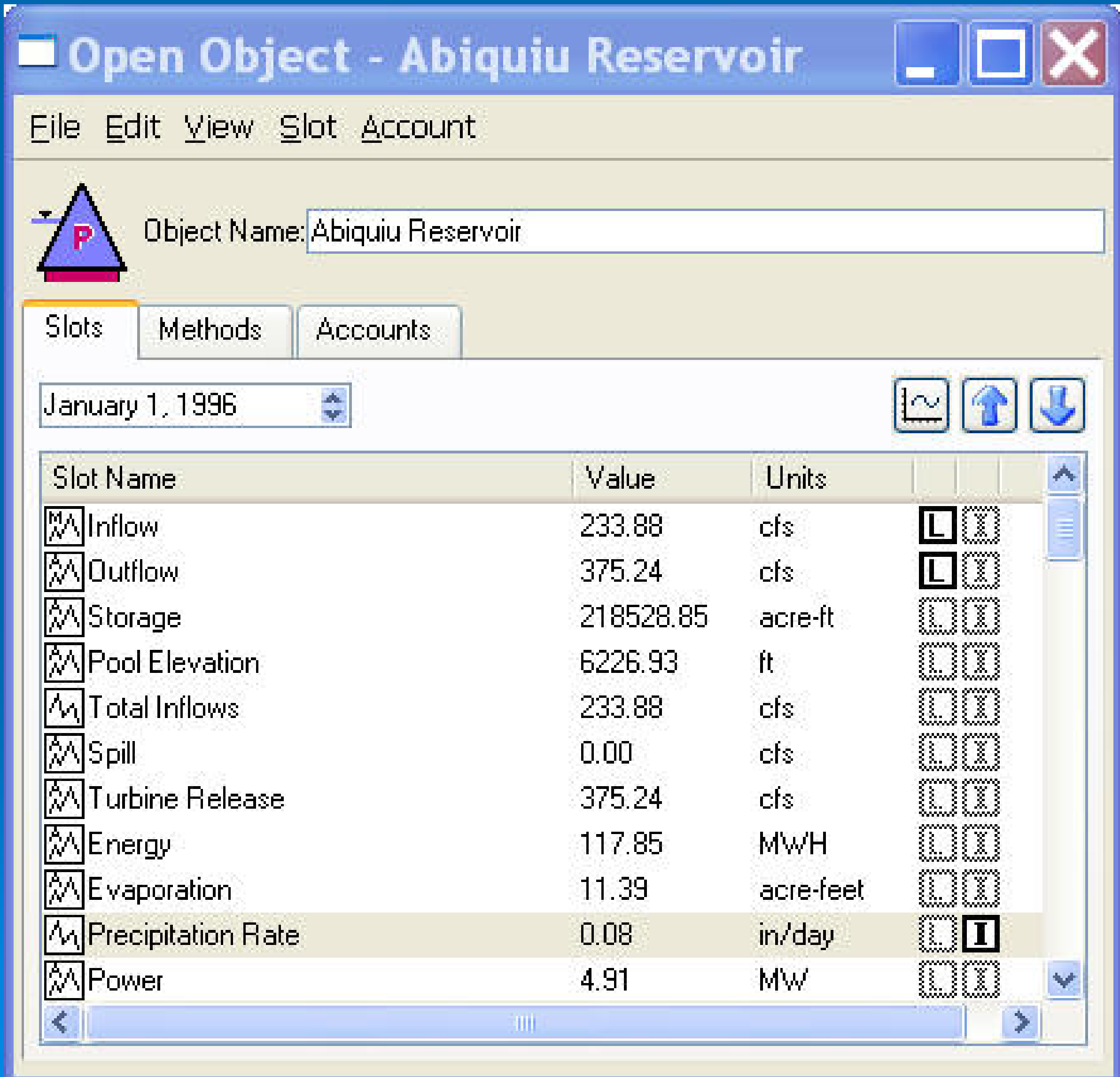


Objects on the Palette and Their Methods

- Reservoirs Storage (*mass balance, release, spill)*
  - Power Reservoirs
    - Level *(+ tailwater, power, energy, eis)*
    - Sloped (+ *wedge storage)*
    - Pumped Storage *(+ pump/generators)*
- Confluence - *mass balance*
- Bifurcation - *mass balance*
- Bi-directional Gravity Flow Canal - *bi-directional gravity flow*
- River Reach / AggReach- *routing, water quality*
- AggDiversion Site - *demands, consumption, return flow, available water*
- Water User - *demands, consumption, return flow*
- Diversion - *pumped or gravity diversion structure*
- Groundwater Storage - *gw interaction for return flows, seepage, conjunctive use*
- AggDistribution Canal - *calculates diversion schedules, routes flows*
- Stream Gage - *input for river gage data; propagates flow value u.s. and d.s.*
- Inline Power Plant – *generation is function of flow*
- *Control Point – constrains upstream flood releases*
- Thermal Object - *economics of thermal power system*
- Data Object - *user-specified data*

RiverWare

Open Object…………



SLOTS
(data structures)

METHODS (physical process algorithms)

ACCOUNTS ownershipt of water in this object

Riverware

Slots - RiverWare's Data Structures

Slots on Objects hold all numerical data

Slot Types:

- Timeseries, Aggseries, Multislots (inputs and outputs)
- Periodic Slots (repeating timeseries data)
- Tables (functional relationships and parameters)
- Scalar Slots – hold a single value

Expression Slots on data objects

User-Selectable Methods

Method Categories on each object type.

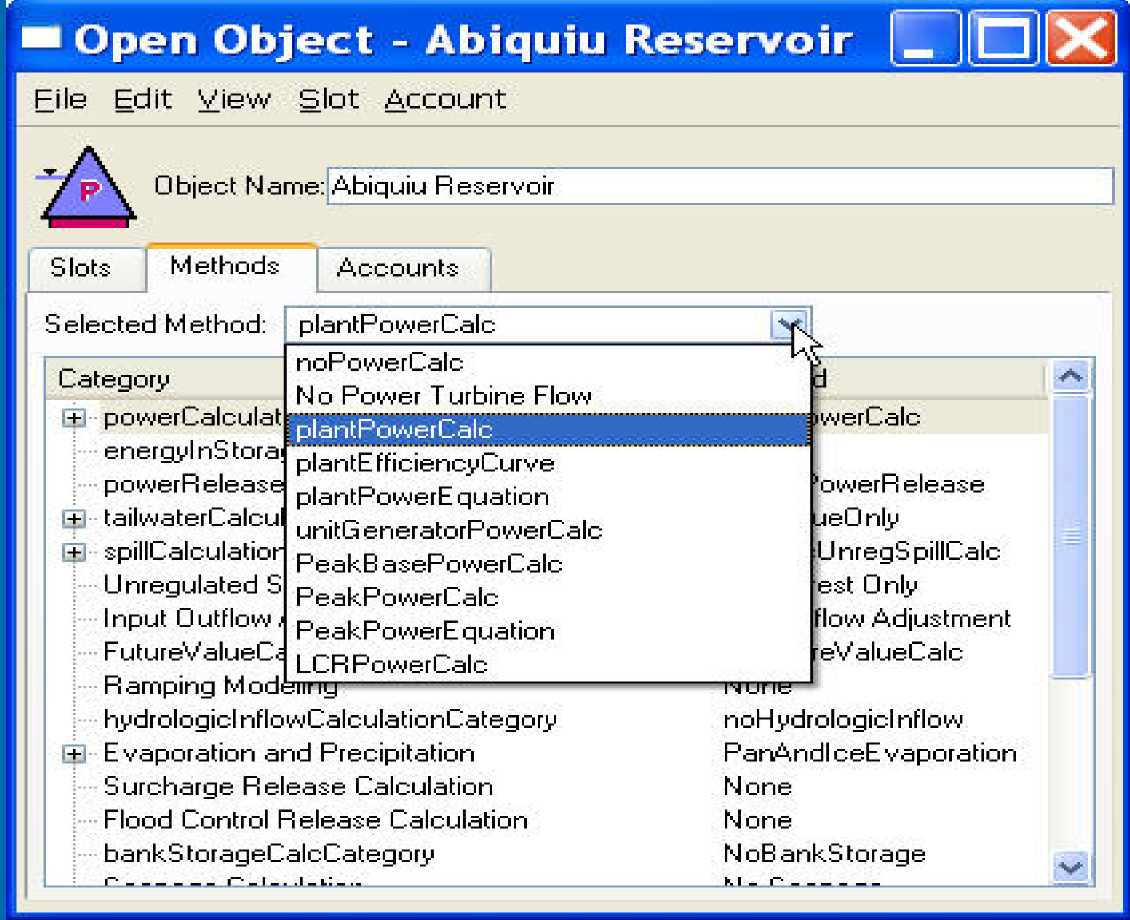
User selects methods based on: timestep size,
available input data,
required outputs
institutional requirements

Power Reservoirs -----> Power Calc Category

plant power, unit generator power
Peak Power, LCR Power

Reaches ------> Routing Category

no routing, Timelag, impulse-esponse,
Muskingum, Muskingum-Cunge,
Kinematic

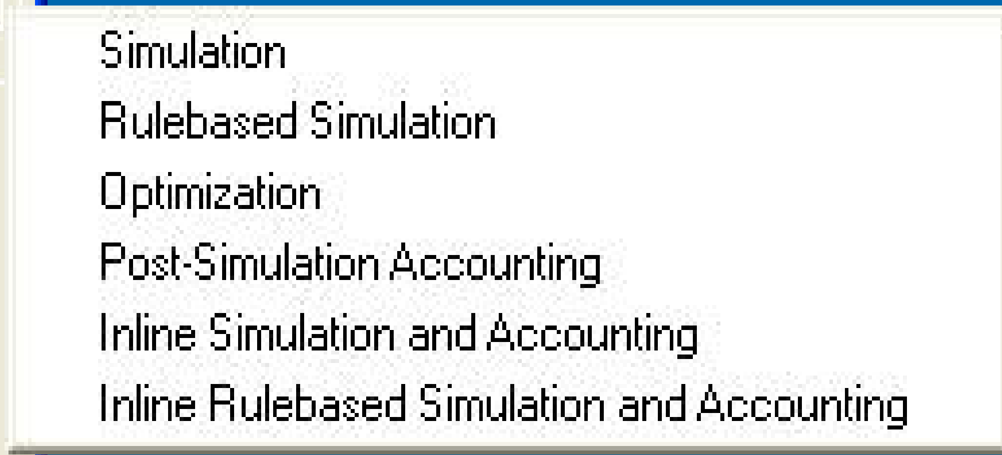


Each simulation object has:

- General slots
  (e.g., reservoirs have inflow, outflow, storage)
- Method-specific slots
  (e.g., Muskingum has routing coefficients)

RiverWare

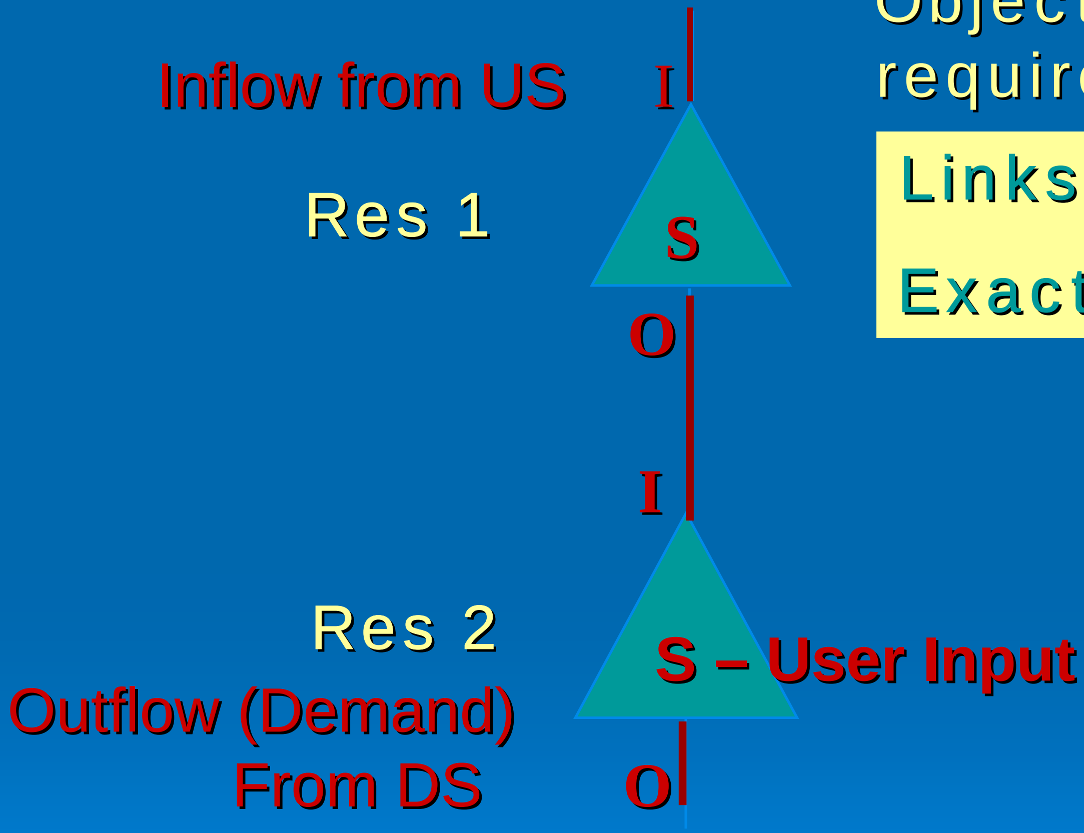
Select Solver…



Simulation Model

- Objects on Workspace Represent Features of the River and Reservoir System
- Objects contain their own data
- Objects contain their own physical process models
- Objects know only about themselves
  - when they get a new value
  - how to use their data to simulate
- Simulation is executed by a “controller,” one object at a time, as requested by the objects

Simulation – the OO model



Objects solve when they have required knowns and unknowns

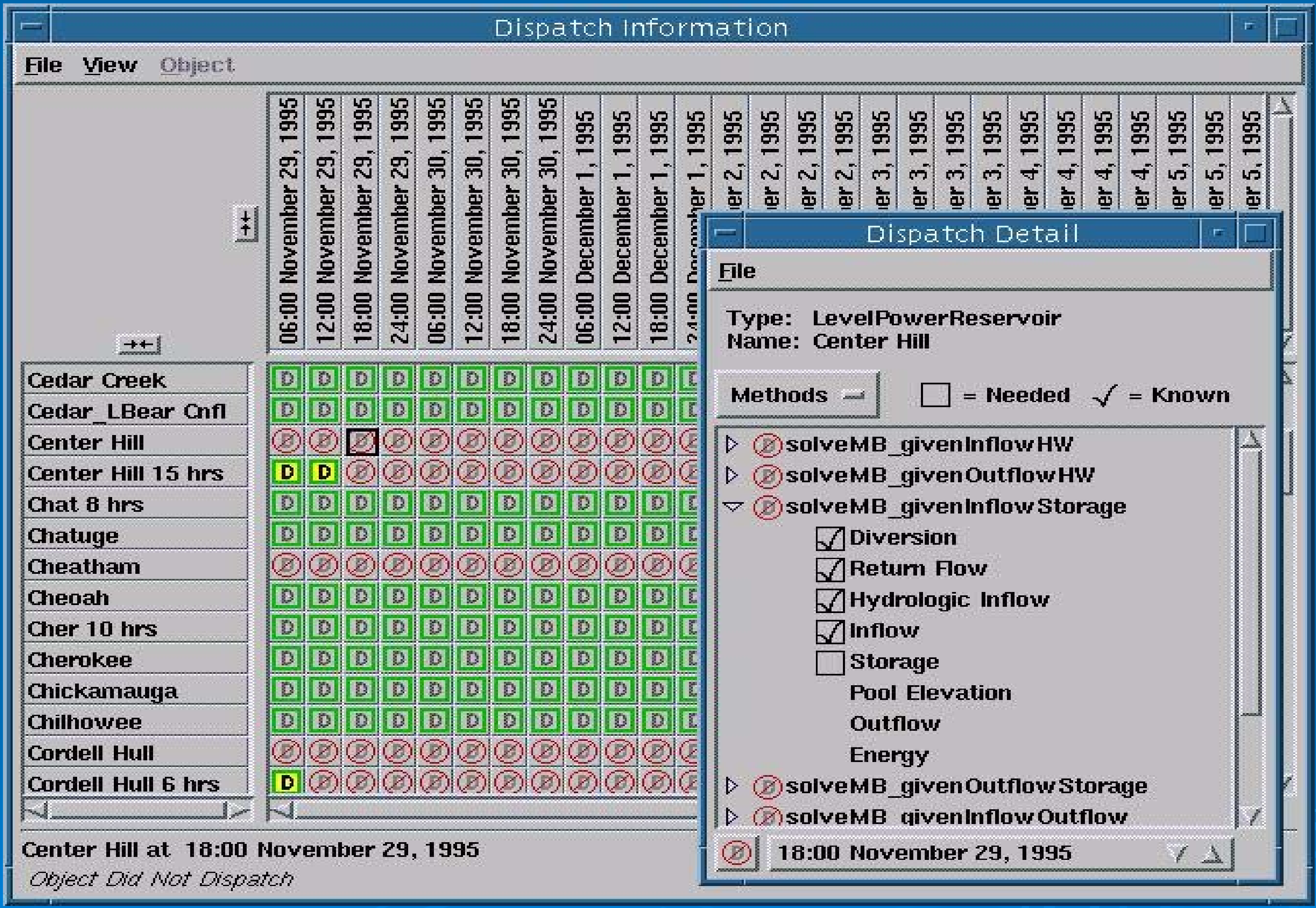
Links propagate values

Exactly determined system

$$S_t = S_{t-1} + I_t - O_t$$

$$I_t = S_t + O_t - S_{t-1}$$

Simulation Analysis Utility



Part 3
Rulebased Simulation

RiverWare

Multi-objective River and Reservoir System Modeling:

Hydrologic processes + operating policies

Flood Control	23.55	1
Water Supply	19.34	2
Navigation	16.2	5
Water Quality	326.9	3
Aquatic/Riparian Habitat	4.1	6
Recreational Flows	9.8	4
Recreational Lake Levels	0.34	8
Hydropower	101.9	7

Weights/Penalties

Priorities

Rulebased Simulation

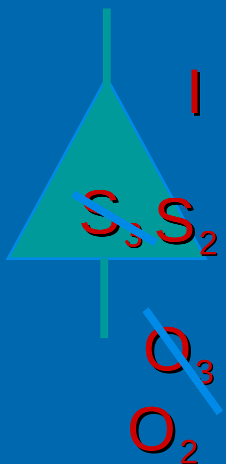
- Simulation is under-determined
- Operating policies are prioritized rules
  - IF (state of system)
  - THEN (set decision variables)
- Rules alternate with simulation to solve system

RiverWare

Rulebased Simulation

- Same set of decision variables as simulation
- solves system at each timestep using a set of prioritized rules
- Rules set values such as releases and withdrawals based on state of the system after previous rule has fired
- After each rule is executed, the simulation propagates the effects of the rule
- Higher priority rules are special conditions and dominate lower priority rules which represent ideal conditions

RBS example



Forecasted Inflow is Input

Not enough data to solve object

Rule 3: Set Storage to guide curve

Object Solves for Outflow given Inflow and Storage at priority 3

Rule 2: If Outflow < Min Flow

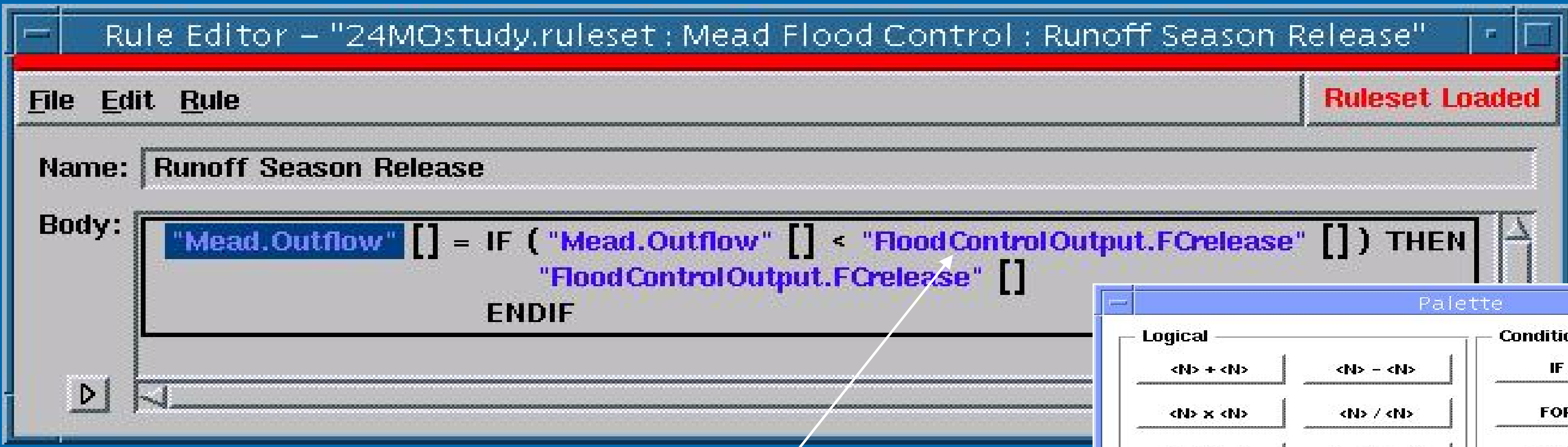
Outflow = Min Flow

Object Re-solves for new storage at priority 2

Rule Language

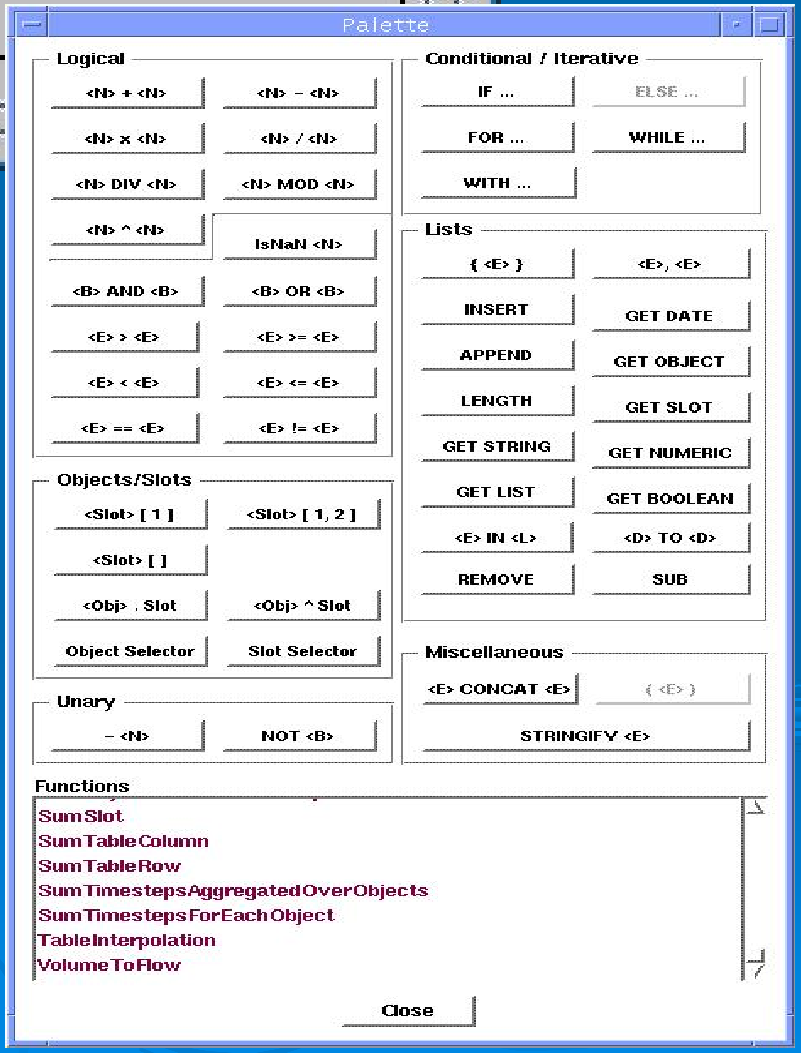
- Rich “programming” language
- Functional language (“referential transparency”)
- Build expressions in
  Graphical Syntax-directed Editor
- Expression Validation
- Automatic Unit Management
- Function Libraries
  - user-defined
  - pre-defined

RiverWare



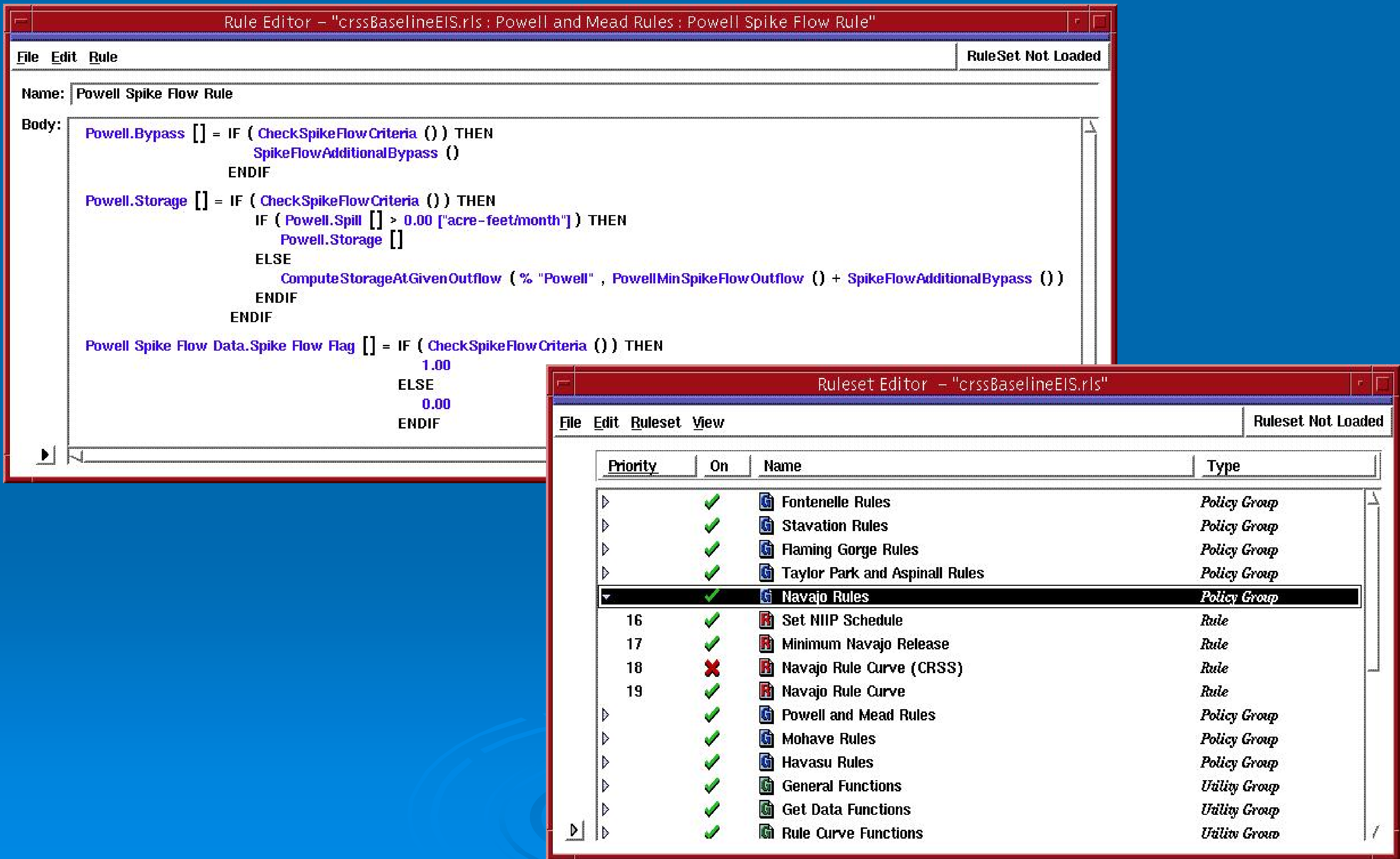
User defined function

Rule Syntax-directed Editor

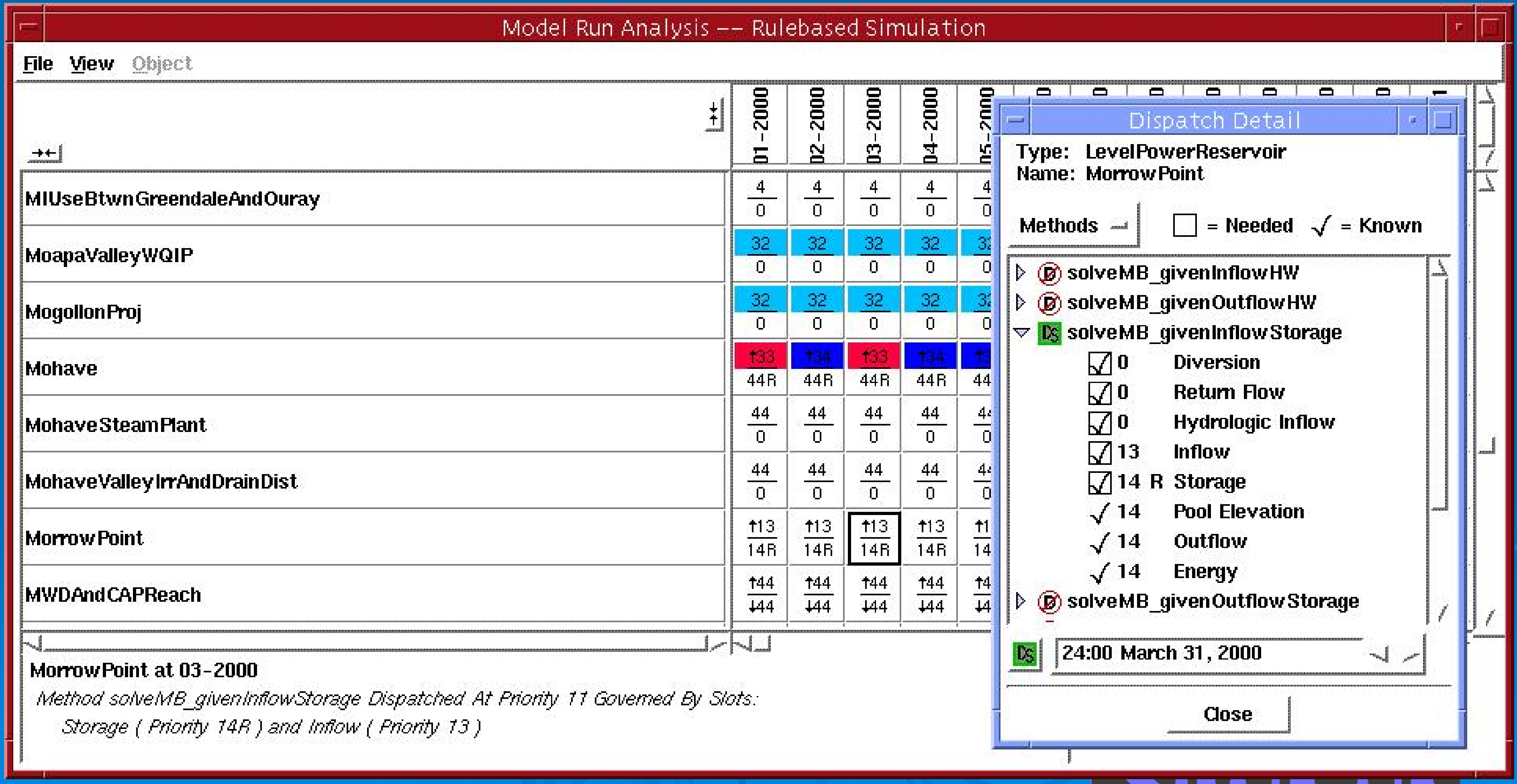


Pre-defined Function Library

A rule example



RBS Run Analysis



Part 4
Water Accounting

Rules +
Accounting
Simulation

Rulebased
Simulation

Optimization

Multi-objective

Simulation

Input-Output Modeling

Calibration

If-then scenarios

Water Ownership, Water Accounting, Water Rights

- “Paper” Accounting
  - optional reconciliation
- Storage, Flow, Diversion Rights
- Exchanges, Loans, Rents, Carryover
- Accrual
- Spreadsheet Solution
- Mix with Rulebased Simulation
- Network Optimization (future development)

Physical Water:

Modeled in river basin objects (reservoirs, reaches, etc.)

e.g. storage in a reservoir represents volume of physical water in the reservoir

Paper Water - type and ownership:

Modeled in Accounts on the river basin objects

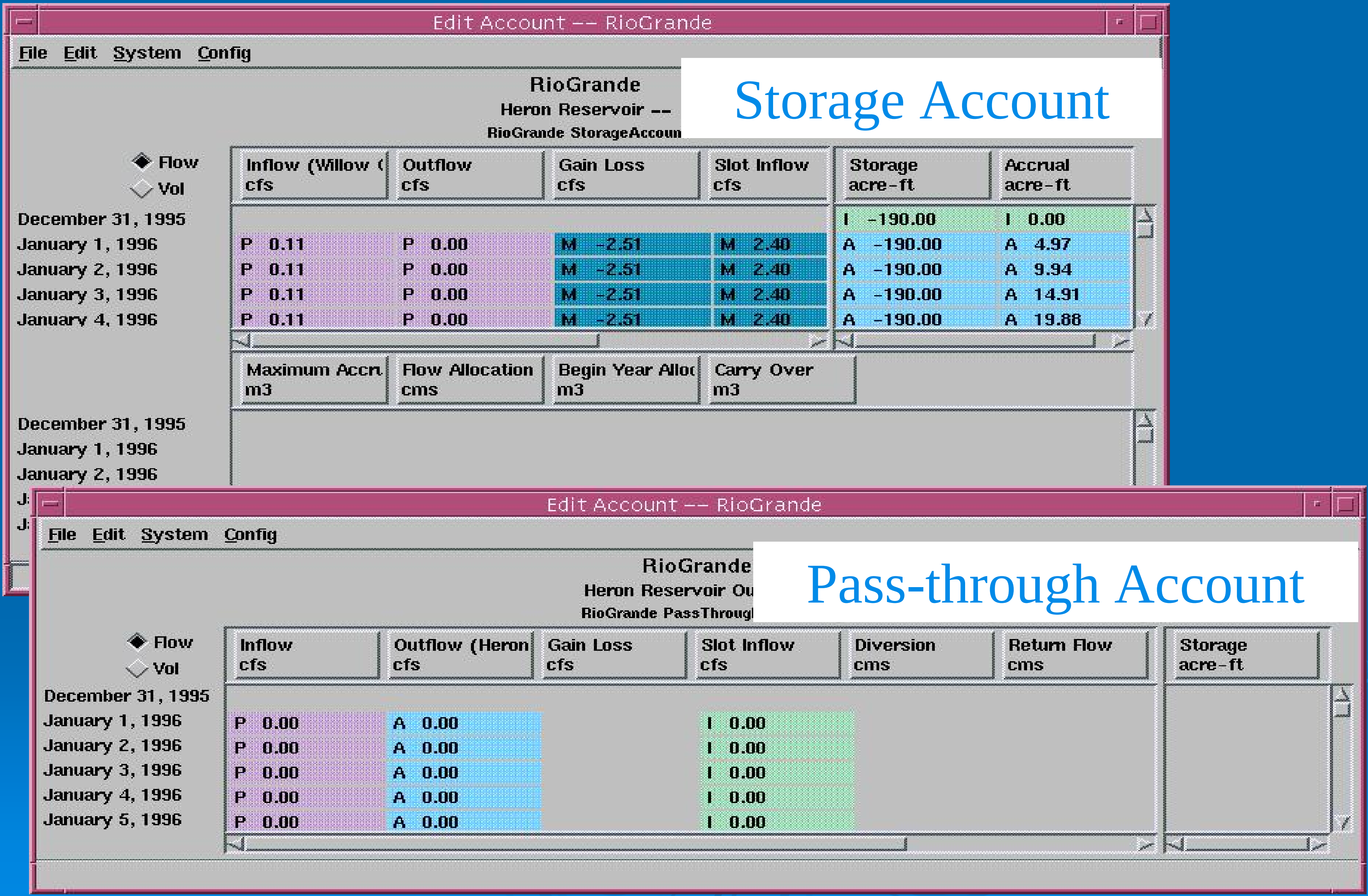
e.g. storage accounts on a reservoir reflect type and ownership of physical water in the reservoir

Reconciliation: controlled by user

Types of Accounts

- Legal Accounts:
  - Storage Account - on Reservoirs
    - Storage, gain/loss, accrual, carry-over, inflow and outflow
  - Diversion Account - on Water Users, AggDiversions
    - diversion, consumption, and return flow
  - Instream Flow (implementation not complete)
- Non-Legal Accounts
  - Pass-through Accounts - Reaches, Confluences, Gages, Reservoirs
    - keep track of water moving between legal accounts
    - outflow = inflow + lags, gain/losses
    - Diversion and return flow

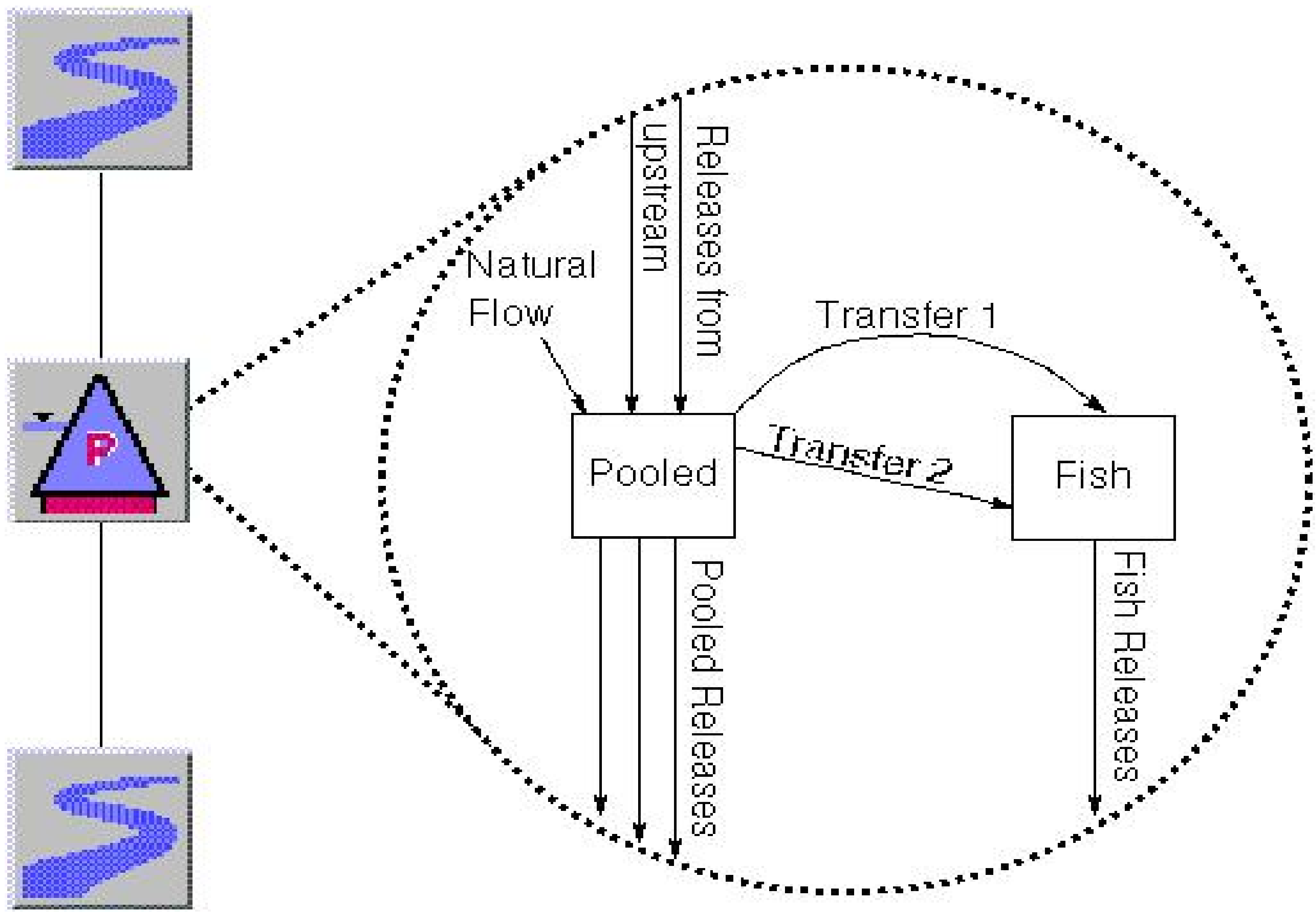
Riverware


Storage Account
Edit Account -- RioGrande
File Edit System Config
RioGrande
Heron Reservoir --
RioGrande StorageAccoun
Flow
Vol
Inflow (Willow (cfs
Outflow cfs
Gain Loss cfs
Slot Inflow cfs
Storage acre-ft
Accrual acre-ft
December 31, 1995 I -190.00 I 0.00
January 1, 1996 P 0.11 P 0.00 M -2.51 M 2.40 A -190.00 A 4.97
January 2, 1996 P 0.11 P 0.00 M -2.51 M 2.40 A -190.00 A 9.94
January 3, 1996 P 0.11 P 0.00 M -2.51 M 2.40 A -190.00 A 14.91
January 4, 1996 P 0.11 P 0.00 M -2.51 M 2.40 A -190.00 A 19.88
Maximum Accru m3
Flow Allocation cms
Begin Year Alloc m3
Carry Over m3
December 31, 1995
January 1, 1996
January 2, 1996
Pass-through Account
Edit Account -- RioGrande
File Edit System Config
RioGrande
Heron Reservoir Ou
RioGrande PassThroug
Flow
Vol
Inflow cfs
Outflow (Heron cfs
Gain Loss cfs
Slot Inflow cfs
Diversion cms
Return Flow cms
Storage acre-ft
December 31, 1995
January 1, 1996 P 0.00 A 0.00 I 0.00
January 2, 1996 P 0.00 A 0.00 I 0.00
January 3, 1996 P 0.00 A 0.00 I 0.00
January 4, 1996 P 0.00 A 0.00 I 0.00
January 5, 1996 P 0.00 A 0.00 I 0.00

Linking Accounts

- Accounting network is formed by links between inflows and outflows of accounts called "supplies"
- Supplies are software objects, can eventually have preferences; amenable to a network solution
- Each supply has the following attributes:
  - Type – Inflow/Outflow or Diversion/Return Flow
  - Destination
  - Release Type

Diagram of Accounts



Accounting Methods

- Introduce water into accounting system through slotInflow, Gain/loss method categories on objects e.g. distribute losses or inflows to the accounts
- Carryover methods on accounts
- Methods are site-specific - defined by user in interpreted language
- Methods have dependencies and re-execute when new values are introduced

Accounting Solution

- "Spreadsheet" type solution
  - account solves whenever it gets a new value (not controlled by dispatcher)
- "After the fact" accounting
  - assumes physical model has solved
- Current solutions:
  - storage accounts solve only for storage balance (user specifies storage account outflows)
  - pass-through accts solve only downstream
- Account slots are set by user input or rules
- On supplies, a NaN is interpreted as zero

Three Accounting Controllers

1. Inline with simulation –
   after each simulation timestep

2. Post simulation -
   after entire simulation run

3. Inline with rulebased simulation

(execution of accounting methods and rules controls when the accounting solution can take place)

Use Rules to Control Accounting

- Rules can be used to set supplies

For example:

FishSupply[Current Timestep] =

FlowTarget() – SumOtherReservoirReleasesOfType(“Fish”)

- If account Inflow is known, this causes the account to solve for Storage

Water Exchanges

- Borrow - a supply to an account or a user input
- Source Payback - an outflow (demand) from an account
- Destination Payback (optional) - a supply to an account

Account balance is updated whenever a new value is set in the borrow or payback

Account balances can be accessed by rules

Future Accounting Enhancements

- Visualization of accounting network
- Automatic generation of pass-through accounts
- Implementation of legal “Instream Flow” accounting
- Automatic reconciliation of paper and physical water
- Network optimization for complete water rights solution

RiverWare