Optimization Using **OASIS with OCL™ :** California Application

by Jeffrey K. Meyer, Anthony P. Pulokas, Harold W. Meyer, and Daniel P. Sheer

Abstract

An effective way to simulate large scale surface water systems is to use an optimization model. Some modelers in the water resources field have used linear programming to optimize the operation of a system over a period of record with a single optimization. **OASIS with OCL™** works very differently. OASIS simulates a period of record by optimizing the operations for a single time step, then going on to the next time step. Modeling the very large and complex California water system using a monthly time step requires the use of several resources. One of which is OCL, short for *operations control language,* and it gives tremendous power to the modeler. The role of OCL is similar to that of a "scripting language" or "macro language" in other kinds of computer programs. *OCL is not the source code for OASIS.* Rather, it is a form of *input* to OASIS, in which the user enters special operating rules. The user w-rites rules in OCL using various *simulation commands.* The formulation of the linear program is done using constraints and goal programming which describe the California system. The source code of the model *never* has to be modified or recompiled. OASIS also uses existing models or modules run in parallel to calculate delta outflow requirements during run time.

Introduction

The California water system is one of the largest and most complicated in the world. The system includes both the federal Central Valley Project (CVP) and State Water Project (SWP) facilities which are operated separately. The projects also share ownership in some of the facilities and operate them jointly. The operation of the projects is complicated by a ecologically sensitive river delta which is geographically located at the center of the system.

The CVP is the largest water storage and delivery system in California. The federal facilities include 7 major reservoirs including the 4.55 million acre feet (maf) Shasta reservoir, the largest reservoir in the state. In addition to its reservoirs, the CVP uses several canals and export facilities to meet its demands. CVP reservoirs provide a total storage capacity of over 12 maf, nearly 30 percent of the surface storage in the California, and deliver about 7 maf annually for agricultural, urban, and wildlife refuge use. Deliveries to the CVP contractors in the Sacramento Valley receive a supply of about 2.3 maf/yr. Settlement contractors on the San Joaquin River receive delta water from Northern California which is exported at Tracy Pumping Plant, stored in San Luis Reservoir and/or pumped directly via the Delta-Mendota Canal. About 90 percent of the south-of-delta contractual delivery is for agriculture and urban uses. The remaining 10 percent is delivered to wildlife refuges.

The SWP facilities include 10 major reservoirs, including the 3.5 maf Lake Oroville, the second largest reservoir in the state. The SWP delivers its supplies using 20 dams, 662 miles of aqueduct, and 26 power and pumping plants. The SWP contracts were signed for an eventual annual delivery of 4.2 maf. Of this annual entitlement, about 2.5 maf was to serve Southern California and about 1.3 maf was to serve the San Joaquin Valley. The remaining 0.4 maf annual entitlement was to serve the Feather River service area and the San Francisco Bay and Central Coast regions. Currently, the Southern California use of SWP supply has reached about 60 percent of full entitlement. (California Department of Water Resources, 1998)

Because of the complexity and size of the California system, it was necessary to use a model that wasn't too small or simple. WRMT developed and equipped **OASIS with OCL™** with the intent to model a complex system like California. OASIS is a generalized program for modeling the operations of water resources systems. OASIS simulates the routing of water through a system represented by nodes and arcs. The routing may account for both human control and physical constraints **on** the system.

OPTIMIZATION USING OASIS with OCL™

We cannot emphasize enough that OASIS is a **generalized** program. It is not a model of California's State Water Project system, New York City's watershed, or the canals of South Florida. Rather, it **is** a program that allows the user to model all of these and virtually any other system in the world. We often refer to OASIS as being completely *data-driven*. That is to say that the user specifies the features and operating rules of the system through OASIS's input data, not by altering OASIS's source code.

One reason OASIS is unique **is** that it simulates the routing of water by solving a **linear program.** What this means for the user, is that all operating rules are expressed as *operating goals* or *operating constraints.* What this does *not* mean is that the user must be trained in linear programming in order to use OASIS. To model a system, the user simply needs to approach the problem as a set of goals and constraints. Let us be clear about these terms:

- **A constraint** is a rule that OASIS must obey.

- A **goal** is a rule that OASIS tries to meet. By their nature, goals are in competition with other goals, so typically all goals cannot be satisfied. The user specifies which goals take precedence over others by giving them relative *weights*. Think of this as ranking the goals.

Some modelers in the water resources field have used linear programming to optimize the operation of a system over a period of record with a single optimization. OASIS works very differently. OASIS simulates a period of record by optimizing the operations for a single time step, then going on to the next time step. Thus a 60-year record with a monthly time step would result in 720 separate optimizations. In the other modeling approach, the model has "perfect future knowledge," where the inflows and demands are known for the entire record at the start of the run. This allows the system to respond, for example, to a flood a year before it occurs. OASIS's running from time step to time step is much more realistic since it's more like how the operators, who are not blessed with perfect future knowledge, control the system.

OCL is short for *operations control language*, and it gives tremendous power to the modeler. The role of OCL is similar to that of a "scripting language" or "macro language" in other kinds of computer programs. *OCL is not the source code for OASIS.* Rather, it is a form of *input* to OASIS, in which the user enters special operating rules. The user writes rules in OCL using various *simulation commands*. The source code of the model *never* has to be modified or recompiled.

OASIS has evolved from WRMI's work modeling water resources systems all over the United States and elsewhere. That experience has guided us to develop OASIS with the following criteria:

Flexibilty

WRMI has found that building effective computer models of water resources systems can be a time-consuming, expensive job. After a new model is built, modifications often must be made to the source code in order to study the alternatives. Often, the alternatives may be very complicated. In order to build new models and modify existing ones, OASIS has been designed to be very flexible. For example, the user decides how many nodes and arcs are in the system, and how they connect. Also, the input data can come from different sources, such as time-series databases or time patterns (whose values cycle every year), or the values can be computed with OCL.

Flexibility is the strength of OCL. In the modeling programs of yesterday, a model would follow certain rules of pre-specified form, and the user would only supply certain parameter values for these rules. Although the user can still do this with OASIS, OCL frees the modeler from the limitations pre-specified rule forms. Because it is extremely difficult to foresee every type of rule that a user might want to model, OCL allows the user to write new rules where the user can design the *form* of the rule, as well as the parameter values. OCL also allows the user to add *conditional* ("if-then"-type) logic to the rules.

Standard Features

Standardized features are the complement to flexibility. After all, there is nothing more flexible than starting from scratch and writing a program in FORTRAN or another programming language. OASIS is designed specifically to model the operations of water resources systems, and it relies on many standardized features that are appropriate to that kind of modeling. For example, OASIS knows how to compute evaporation from a reservoir. It knows how to handle flow capacities and minimum flow targets. It automatically ensures that the continuity of flow is not violated. Furthermore, and very importantly, model input and output are handled entirely by OASIS. These standardized features are not limiting to OASIS because the **OCL** allows the user to go beyond these forms whenever needed. However, the job is made easier because the most common tasks are already handled.

Intuitive, Realistic Forms For Operating Rules

Because OASIS simulates routing decisions through linear programming, all simulation rules are represented as either goals or constraints. The fact that rules can be modeled as goals is particularly important (and novel), because goal-seeking behavior is an *efficient* modeling approach which corresponds very well to the way real world operators and planners think of a water resources system. For example, reservoir storage targets, instream flow requirements, and off-stream deliveries are typical goals for a water resources system. Furthermore, these goals are often in competition with each other. The goal-seeking behavior of OASIS handles these rules very elegantly. With other modeling approaches, these competing goals would have to be modeled with a complex set of "if-then" type rules. **OASIS's** approach greatly cuts down on the "if-then" rules.

Furthermore, the rules that are written in OCL usually look just like the rules that planners, operators, and policy-makers use. For example, an agreement between water users might say that the diversion at point A plus the diversion at point *B* must be less than 70% of the flow at point C. In OCL, the user would write a constraint which is instantly recognizable as the mathematical form of that statement.

Integration With Other Models

OCL allows the user to send and receive data between OASIS and other programs, *while the programs* are running. We can thus say that OASIS and the other programs **are running in parallel.** To **OASIS,** the other programs are "external modules." These external modules can be created from scratch, or existing programs can be fitted to communicate with OASIS. This creates virtually unlimited possibilities for modeling water resources systems. Some tasks that are perfect for external modules are groundwater flow and contaminant transport, biology or ecology, rainfall runoff, snowpack, variable demand, agricultural return flow, river temperature, lake stratification, and tidal mixing. The list goes on.

Because all modules are running in parallel, they can react to each other. For example, OASIS could pass the flows that it has just computed to a water quality model, which would then use those flow to compute water quality data. The water quality data passed back to OASIS could then be used to conditionally set flow targets.

External modules might share input and output databases, or they might use their own input and output. To OASIS, it does not matter what computer language is used for the source code of the external modules. Also, the modular nature of this approach allows different specialists to develop and maintain each module.

Database Storage

OASIS input and output is stored in databases static data is kept in Microsoft Access and time-series data is kept in HEC-DSS databases. (DSS was developed specifically for hydrologic time series data.) Unlike ASCII files, which often have quirky formatting rules, database tiles are always labeled and properly formatted. Databases also provide an effective storage system through which modules can share input data, and are perfect for interaction with an OASIS graphical user interface (GUI). The GUI is still under development, but when it is completed, the user will not even have to interact with database programs. All input will be entered through a series of customized drawings or dialog boxes.

OASIS stores all model output to a database, so there are no secrets in the operation. While the user **may** interact with the databases directly, it is not necessary. Instead, the user can use post-processor programs to report output in exactly the form needed. There is one post-processor program for generating text tables and one for graphical plots. The post-processing is fully configurable -the user specifies what output to display (and it may be the result of complex formulas), and how to format it.

THE CALIFORNIA APPLICATION

The idea for OASIS came from the work Water Resources Management Inc. (WRMI) was doing in Florida five years ago. The work included the need for a model that could be used for planning studies, operation studies, permitting, and regulation. The development of OASIS followed two years later with the first application used to fill the needs of Metropolitan Water District of Southern California (MWD), the largest of the California State Water contractors.

WRMI used OASIS to develop a DWRSIM-screening tool for the MWD. DWRSIM is the official planning model of California's State Water Project (SWP) and Central Valley Project (CVP). It is maintained by the California Department of Water Resources (D WR).

MWD was interested in producing a screening tool for DWRSIM for two main reasons: speed and flexibility. The most important of these is flexibility, for programming time is usually more consuming than execution time. It is very difficult for users to make useful changes to DWRSIM. Practically all of the operating rules of the system is specified in the FORTRAN source code. There are only a limited number of people on DWR staff who are qualified to change the code. Therefore, it is difficult for others such as MWD to do the studies they need. Although DWRSIM is still seen as the official model of the system, a screening tool would allow modelers to try many alternatives, and choose only the most interesting ones to be modeled with DWRSIM.

The screening tool also has a faster run time than DWRSIM, for much of the system's detail has been greatly simplified and much less output is generated. The 1.8 MB output file of the screening tool stands in sharp contrast to the 28.4 MB output file of DWRSIM. In particular, the representation of the California Aqueduct and the Delta Mendota Canal has been simplified by grouping delivery sites together. Also, the San Joaquin basin and the basins of the east-side streams are not being modeled in the screening tool. Instead, the screening tool must use output from DWRSIM (or some other model) as the flow in these streams.

It is important to recognize that the screening tool has simplified the physical representation of the system without sacrificing any of the complexities of the operations in the Sacramento basin and in the San Francisco Bay / Sacramento-San Joaquin River Delta. This means that the screening tool is still a complicated model, for it represents a complicated system. However, the use of OASIS with OCL means that there are fewer complexities as *artifacts of the modeling approach.*

The complexities of the system revolve around the integrated operation of two separate projects sharing river channels, the delta, and export and conveyance facilities to meet both shared and independent goals. Adding to the physical complexity is the political arena in which the CVP/SWP projects exist. In California there is a constant battle of water supply versus the water for the environment. Proposals for compromise are numerous and speed and flexibility in a simulation model is a necessity to quickly and accurately model various scenarios.

Physical Complexities

One of the problems we encountered in modeling the California system was the representation of San Luis Reservoir. San Luis Reservoir is a joint use facility in which both the CVP and SWP own a specific portion of the reservoir which they can operate independently. We found that the reservoir could be modeled by either modeling the one reservoir as two separate reservoirs, or use OCL to simulate the reservoir correctly as a single reservoir. We chose to model the reservoir as a single reservoir with two distinct accounts.

Only a few aspects of the reservoir operation are handled by the standard input. Most of the operation follows complicated rules that are described below. These rules are modeled with OCL commands. Figure 1 shows a schematic of the reservoir. In the OASIS standard input, node 400 is set to represent San Luis Reservoir and has a capacity equal to the physical capacity of the reservoir. Likewise, the evaporation rate is set. OCL commands are used to divide the reservoir into the appropriate proportions.

The following constraint command sets the storage at node 400 equal to the user-defined variables SL_stor_SWP + SL_stor_CVP.

```
CONSTRAINT San_Luis_Split_storage :
{ dstorage400 = dSL_stor_SWP + dSL_stor_CVP }
```

Then the following segment commands divide the accounts into their respective zones. As stated below the A zone equals dead storage, B Zone equals the rule curve minus dead storage, and the C zone equals capacity minus the rule curve.

```
SEGMENT :  SL_stor_SWP
           { { 0 } SL_storA_SWP { [SL_SWP_Dead_stor] }
                   SL_storB_SWP { pattern(SL_SWP_rule) }
                   SL_storC_SWP { [SL_SWP_Max_Stor] } }
```

```
SEGMENT  : SL_stor _CVP
           { { 0 } SL_storA_CVP { [SL_CVP_Dead_Stor] }
                  SL_storB_CVP { pattern(SL_CVP_rule) }
                  SL_storC_CVP { [SL_CVP_Max_Stor] } }
```

The user-defined variables SL_SWP_Dead_Stor, pattern(SL_SWP_rule), and SL_SWP_Max_Stor and the respective CVP components are constants set in OCL.

The advantage of modeling San Luis Reservoir as a single reservoir is that we can use the storage-area-elevation curve which allows us to correctly calculate evaporation and assign it to each account according to storage.

Regulatory Complexities

The CVP and SWP are subject to several regulatory requirements contained in the May 1995 Water Quality Control Plan for the San Fransisco Bay / Sacramento-San Joaquin Delta Estuary. Figure 2 shows a simplified schematic of the delta. Two of the regulatory requirements are the delta outflow requirements and the combined export limit. A result of the regulatory requirements is a loss in water supply to the water users. To mitigate for the loss in supply, the water users have explored the conjunctive use of ground water and surface water. Examples of the solutions to these modeling problems are presented below.

Delta Outflow Requirements

There are two elements that dictate the delta outflow requirement (State Water Resources Control Board, 1995). Those are an outflow requirement dictated by time of year and hydrologic conditions, and water quality. Temporal outflow requirements can easily be handled by a combination of the OASIS standard features in combination with OCL. Water quality is a complex matter because OASIS was created to rout water, not necessarily to calculate water quality.

In order for OASIS to meet the water quality goal it must first know the outflow which would meet the water quality goal. In this case we used an existing model, known as the compute-RMDO module to calculate the outflow requirements. Because OASIS has the ability to run modules in parallel, we were able to pass information to compute_RMDO, run compute_RMDO to determine the necessary outflow requirement, pass the outflow requirement back to OASIS, and allow OASIS to rout the water to meet the outflow requirement.

The commands to run the module are presented below.

```
RUN_MODULE : compute RMDO

    input :   { abs_period , period , year ,
                timesers(year_type/40-30-30) ,
                prev  delta  outflow }

    output :  { min_flow509.999 , x2 min flow ,
                x2_position }
```

The variables following the input command are passed from OASIS to the module compute_RMDO. Variables following the output command are passed back to OASIS . Min_flow509.999 represents the minimum required delta outflow for the current time period.

Combined Export Limit

The combined export limit (State Water Resources Control Board, 1995) places limits on south delta exports at Tracy and Harvey 0. Banks pumping plants. The example below uses three OCL commands to enforce a pumping limit. The *Set* command below is used to set a variable called *the Export-inflow-Ratio* equal to one of five values depending on the first condition that is true. The constraint *define-delta-inflow* is used to set the decision variable, *Delta-inflow,* equal to the sum of the flow in the arc from node 137 to node 503 plus the inflow from the San Joaquin River, the Yolo Bypass and the East Side Streams. Once these two variables are defined, the pumping limit can be defined. The constraint *Export-Inflow-limit* limits the pumping by forcing the flow from node 550 to nodes 59 and 86 to be less than the *Export-inflow-Ratio* multiplied by the decision variable, *Delta-inflow*.

```
set : InExpRatio

 condition febl     : month = 2 and
                      timesers(8_river  index/)(-l)  <=  1000000
            value   :   .45

 condition feb2     : month =  2 and
                      timesers(8_river _index/) (-1) >   1000000
                                        and
                      timesers(8_river_index/)(-1) <=  1500000
            value :     .40

 condition feb-jun : month >= 2    and    morth <= 6
            value   :   .35

 condition jul-dec : month >= 7    and    month <= 12
            value   :   .65

 condition jan      :  month  = 1
            value :     .65
}
```

Now, it will be more convenient if delta inflow is defined as a distinct decision variable. It is a decision variable because Freeport flow, one of the components of delta inflow, is a decision variable. This is simply a substitution of a single variable for a long sum of variables.

```
CONSTRAINT define _delta inflow :
{  dDelta_inflow   ¯=  dfiow137.503       Sacr amento River
                    +  inflow682          San  Joaquin  River
                    t  inflow055          Yolc  Bypass
                    +  inflow514          East  Side  Streams
```

This is the CONSTRAINT that actually enforces the pumping limit:

Tracy + Banks < export inflow ratio * delta inflow

```
CONSTRAINT ExpLimit :
{ dflow550.086  +  dflow550.059  <  Inf_Exp_Limit  *  Delta-Inflow }
```

The arc from node 550 to node 86 represents Banks pumping plant and the arc from node 550 to node 59 represents Tracy pumping plant.

Conjunctive Use Mitigation Tool

The conjunctive use of a ground water basin is mitigation tool explored by many exporters south of the delta. Typically, the operating rule of the ground water basins are to fill the basins in wet periods and withdrawal water from the basins in dry periods when surface water supplies are low.

The *Target* command below is used to set an operating rule for a ground water basin which is represented by node 480. Node 480 is connected to the system via node 44. The arc from node 44 to node 480 is a two-way arc which allows flow in either direction. The convention used is that flow from node 44 to node 480, ground water banking, is positive and flow from node 480 to node 44, ground water withdrawal, is negative. The GW_release target below will deter ground water withdrawals in years that are not dry or critical.

```
Target GW release NOD :    dflow044.480
{
 condition :  timesers(year type/40-30-30) < 4
  priority  :  1
  penalty+  :  30
  penalty-  : 500
    value :    0
```

The target above is placed on flow in the arc from node 44 to node 480. If the year type is normal, abovc normal, or wet, and there is flow in the arc from 44 to 480, the LP loses 30 points. The rule is written this way to deter using surface water to till the basin until all other surface water uses have been met. The ground water basin is the last item to get surface water in wet periods.

The LP also loses 500 points if ground water is withdrawn if the year type is normal, above normal, or wet. This penalty is large because in these wetter periods there should be enough surface water supply to meet all needs without using ground water supplies.

During dry and critical periods there is no need to penalize the flow to or from the ground water basin. Because there isn’t enough supply to meet surface water needs, we don’t have to worry about the router putting water into the ground water basin. Also the basin has been designed to use the water during dry periods, so there is no penalty for withdrawing water from the basin

Conclusion

Creating a working model of the California system demonstrated that OASIS is a generic, yet very powerful and capable model. OCL provides great flexibility in describing the operating rules used by CVP and SWP operators in real time. In addition, by describing the operating rules using the OCL, the rules are documented within the model input. Complex operations like operating a single reservoir using two separate accounts becomes simple using OCL. Existing models or modules like the compute_RMDO, used in our California application, can be used during run time instead of time consuming tasks like rewriting code or iterating between models. Changes in operating policies can be modeled simply by changing input and goal programming allows the model to determine the optimal solution.

References

California Department of Water Resource (1998) . “The California Water Plan Update Bulletin 160-98", Volume I , 3-25 - 3-33.

State **Water** Resources Control Board / California Environmental Protection Agency (1995), “Water Quality Control Plan for the San Francisco Bay / Sacramento-San Joaquin Delta Estuary”

San Luis Reservoir, Node 400

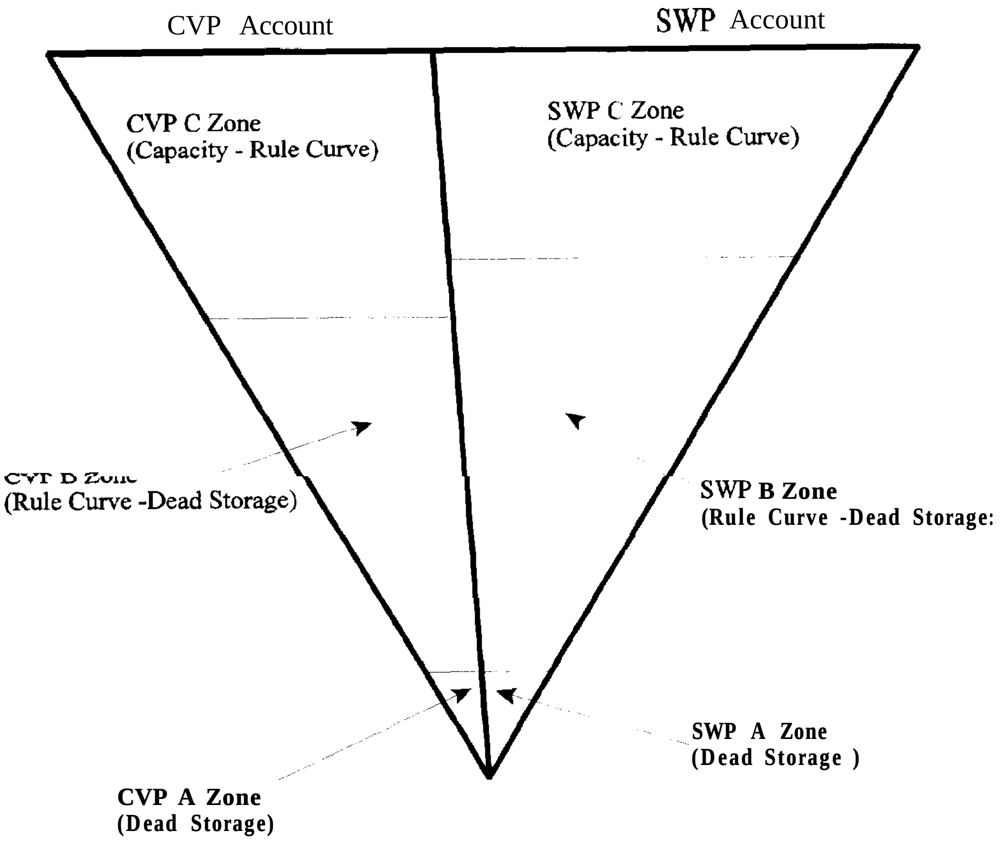

Figure 1

Simplified Delta Schematic

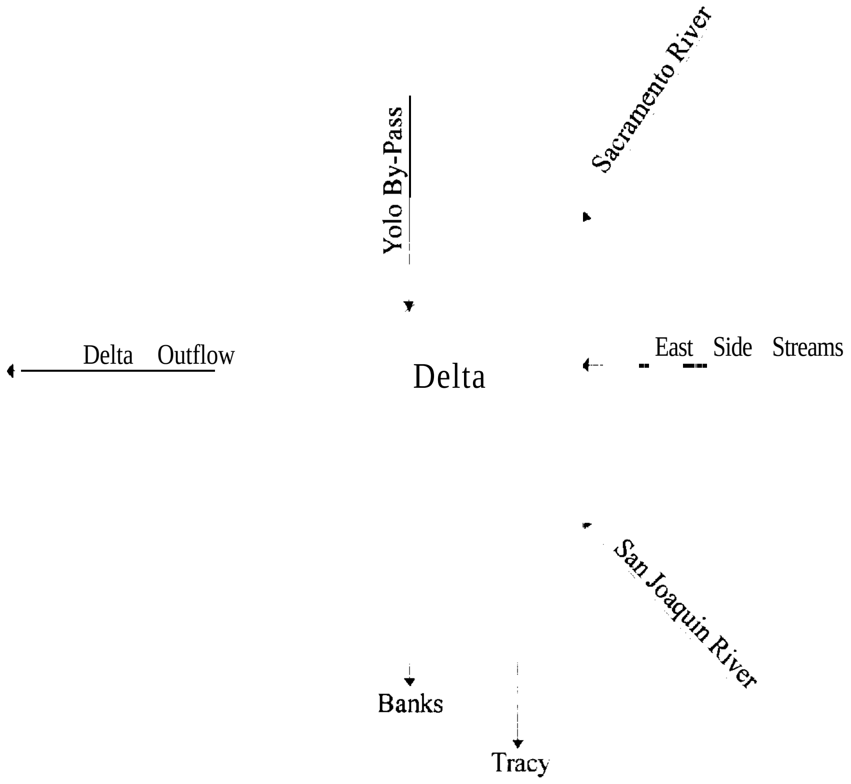

Figure 2

Authors

For further information, please contact:

Jeffrey K. Meyer, Senior Supervisory Engineer, Water Resources Management, Inc. 1851 Heritage Lane, Suite 130, Sacramento, CA 95815. Phone: 916-920-1811. Fax: 916-920-1812. E-mail: wrmi@wrmi.com

Anthony P. Pulokas, Water Resources Engineer, Water Resources Management, Inc. 1851 Heritage Lane, Suite 130, Sacramento, CA 95815. Phone: 916-920-1811. Fax: 916-920-1812.

Harold W. Meyer, Vice President, Water Resources Management, Inc. 1851 Heritage Lane, Suite 130, Sacramento, CA 95815. Phone: 916-920-1811. Fax: 916-920-1812.

Daniel P. Sheer, President, Water Resources Management, Inc. 10440 Shaker Dr., Suite 104, Columbia, MD., 21046 Phone: 410-715-0555. Fax: 410-715-0557.