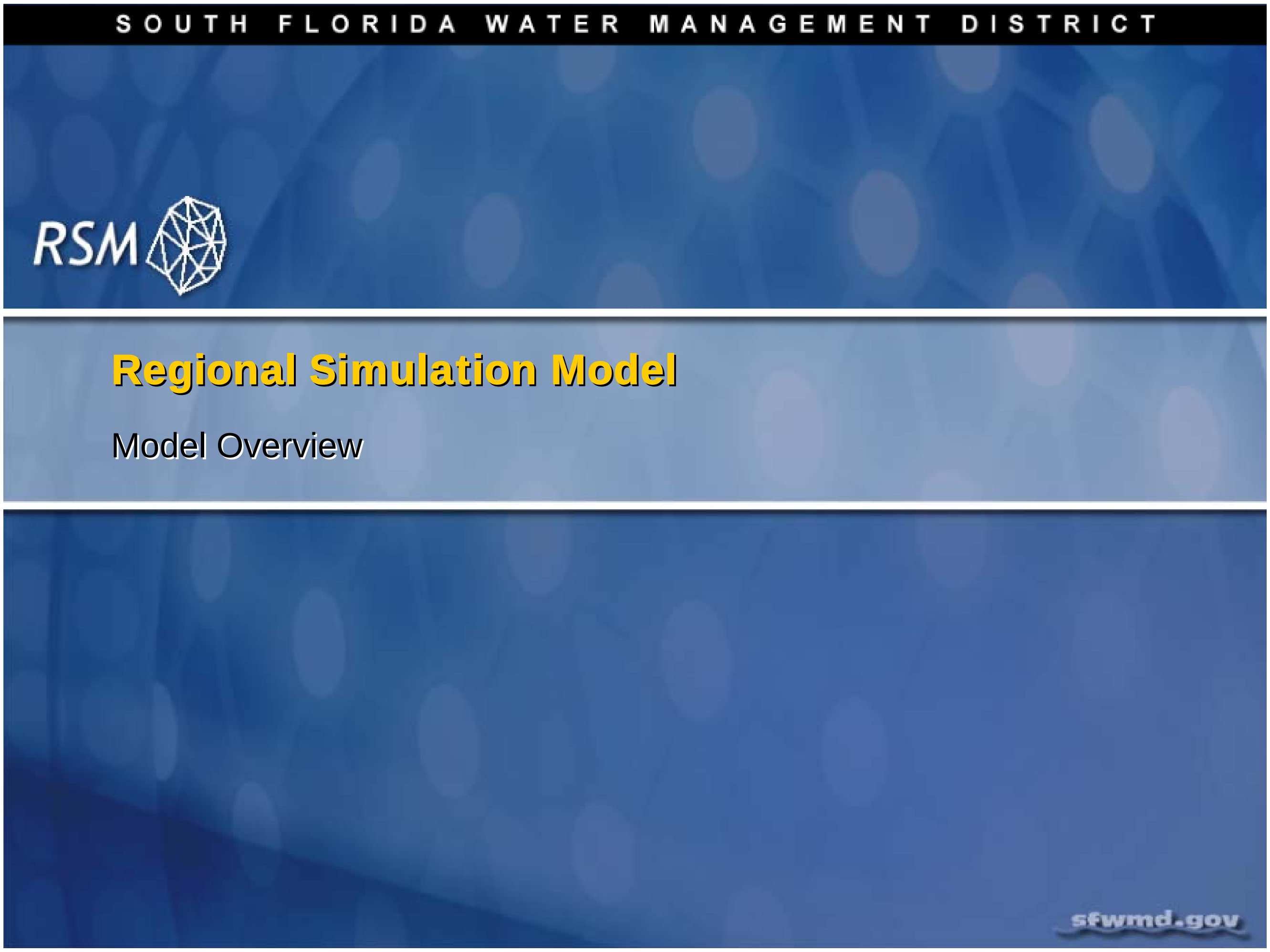


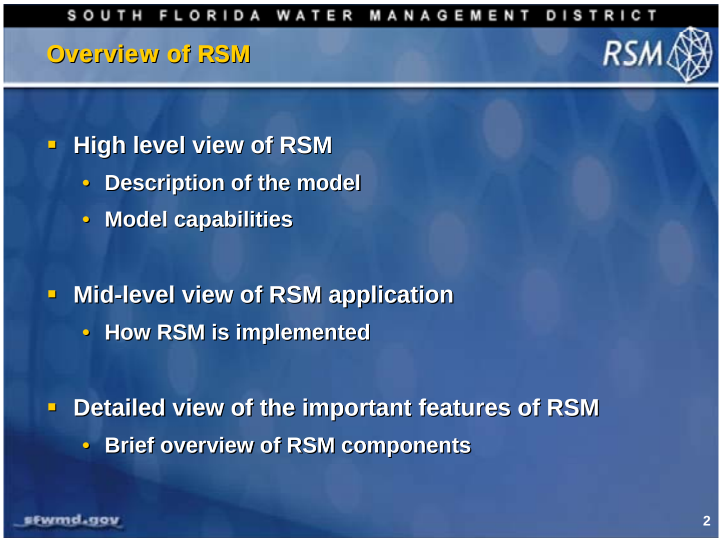
Lecture 1: Regional Simulation Model - Model Overview

In this lecture, the Regional Simulation Model (RSM) is introduced with a discussion of the primary capabilities of the model to help new users understand the RSM compared with other regional hydrologic simulation models.

NOTE:

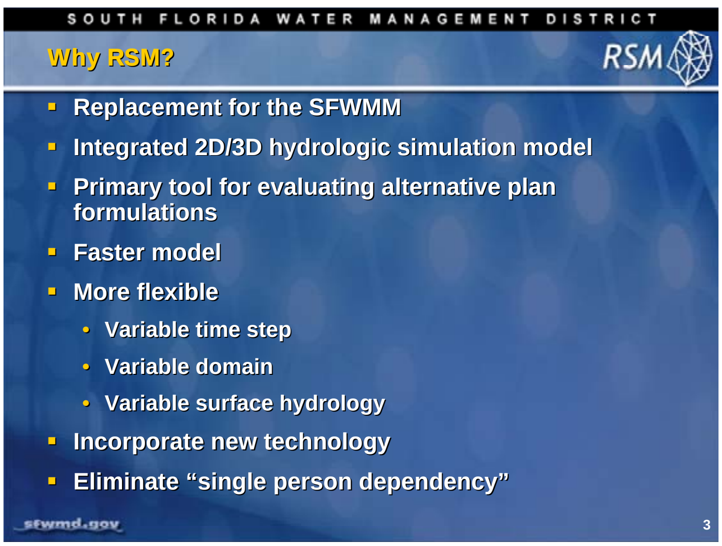
Additional Resources

Additional resource materials can be found in the labs/lab1_BM1 directory.



The RSM is applied through subregional model implementations to provide a tool for the evaluation of alternative water resource management plan formulations.

This session concludes with a discussion of the development of the RSM and the key components of the model.

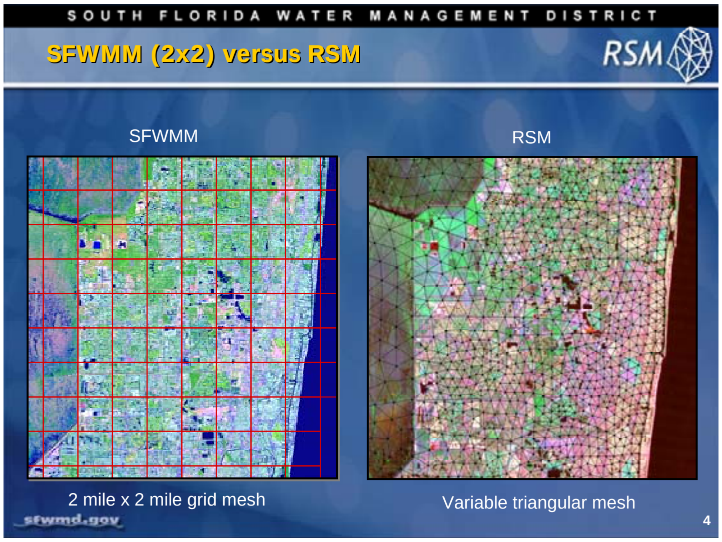


Several factors resulted in the need to develop an improved regional hydrologic simulation model:

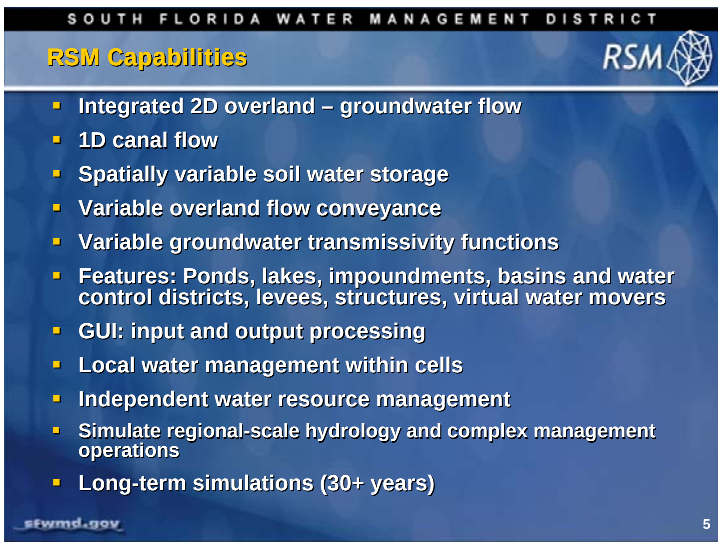
A faster model was required to efficiently test alternative management formulations, because the turnaround for the 36-year model runs and post-processing of the results generated by the South Florida Water Management Model (SFWMM) required almost one week. The new RSM capability includes a run time of less than 4 hours and complete turnaround of results in less than one day of processing time.

The timestep for the SFWMM is fixed at one day, whereas the RSM needs to be flexible with a variable timestep applicable to different domains and different types of surface hydrology. And, to accommodate model implementations that will require short timesteps for model stability, the RSM needs a flexible timestep. Additionally, the capability of a variable domain allows subregional RSM implementations to evaluate local projects more efficiently. Implementation of variable surface hydrology allows the use of simplified algorithms for regional simulation and more detailed simulation of local hydrology and water management systems where appropriate. The SFWMM does not have this flexibility.

It is also important to create a model that is easy to modify and enhance, and able to incorporate new technology (numerical solution methods and hydrologic algorithms) when they become available. The RSM has an architecture that allows the implementation of new technology. The SFWMM is a hard-wired model, which makes it difficult to modify and add new functionality. And, because the hydrology of the model was hard-coded, any changes in the model require modifying and recompiling the code.



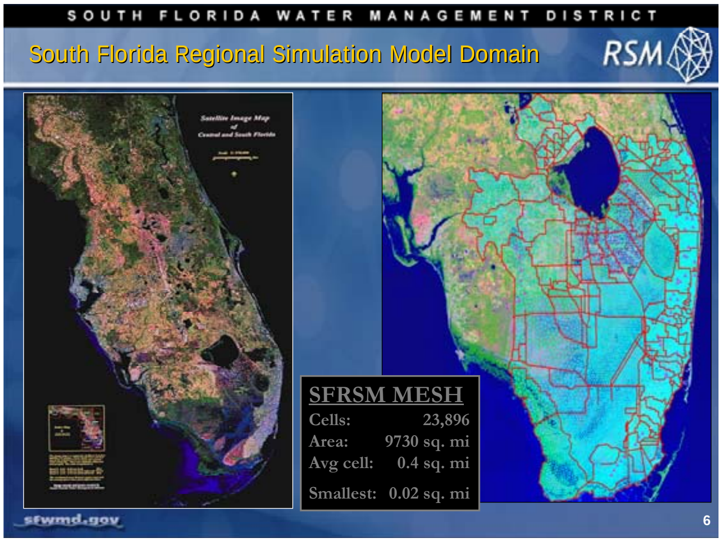
The Regional Simulation Model provides the capability for better resolution of local hydrology and spatially varying properties. The RSM is an improvement over the South Florida Water Management Model (SFWMM) because the SFWMM was a finite difference model that was solved for a two-mile by two-mile grid, whereas the RSM is solved for an irregular triangular mesh with a resolution ranging from 400 acres down to 20 acres, depending on the hydrology of the area. Where the hydrologic gradients are greater the resolution is higher.



General RSM capabilities include:

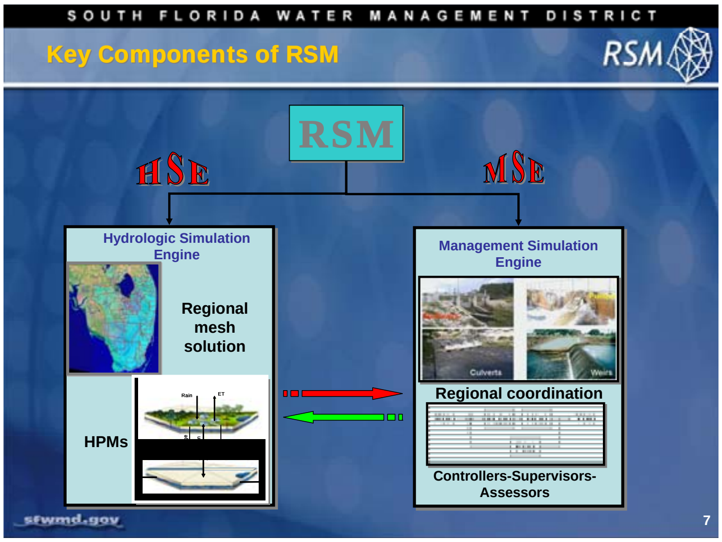
- Two-dimensional (2-D) overland,
- Two and three dimensional (2-D and 3-D) groundwater flow
- One-dimensional (1-D) canal flow
- Separate generation of the 2D irregular-triangular mesh and overlying canal network, which provides increased flexibility.
- Spatially variable soil water storage and aquifer specific yield and distribution of porosity with depth.
- Spatially variable overland flow resistance and groundwater transmissivity on a cell basis.
- A variety of waterbody types that have different properties and methods.
- A wide range structure types, including: pumps, weirs, culverts and generic H-Q relationships.
- Virtual watermovers that can be used to connect any two waterbodies to simulate the appropriate water movement such as recycled gray water.
- Capability to simulate agricultural and urban surface water management systems and irrigated land.

Additionally, the Management Simulation Engine (MSE) provides the means to implement regional water management policies and rules. These features provide a high degree of flexibility in the creation and implementation of the RSM for modeling regional water management systems.



The RSM was developed to replace the SFWMM for simulating the water management in the Central and Southern Florida Flood Control Project (C&SF Project).

The mesh includes nearly 24,000 cells and 5,000 canal segments. The mesh is finer in the developed area near the coast and coarser in the Everglades, where the land is flat and gradients are small.



The RSM consists of two distinct components:

- Hydrologic Simulation Engine (HSE)
- Management Simulation Engine (MSE)

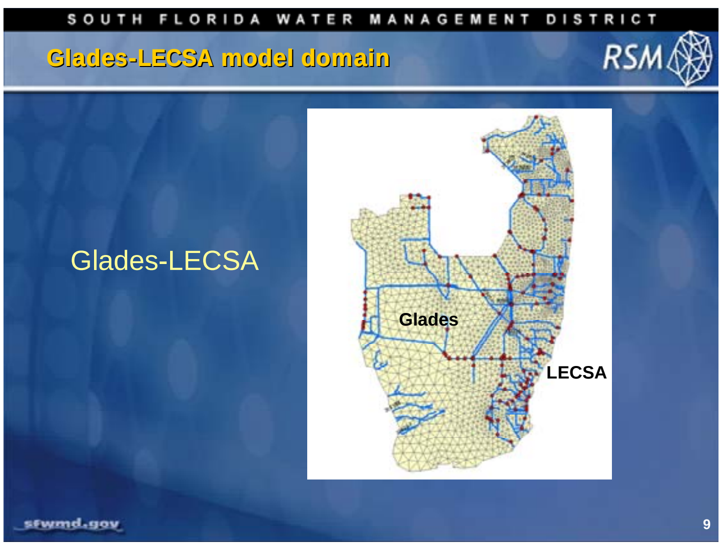
The HSE contains the hydrologic processes. And, the MSE contains the water management rules, policies and constraints which are applied to structures through controllers and supervisors that manage groups of controllers.

The MSE was developed separately from the HSE to maintain a clear distinction between the integrated surface water/groundwater model and the management practices for flood control, water supply and environmental projection that are imposed on the regional system.

The HSE and MSE share “monitors” which are used to output the values of any state variables or flows; HSE calculates the values and MSE uses those values to set constraints on the allowable flows in HSE.

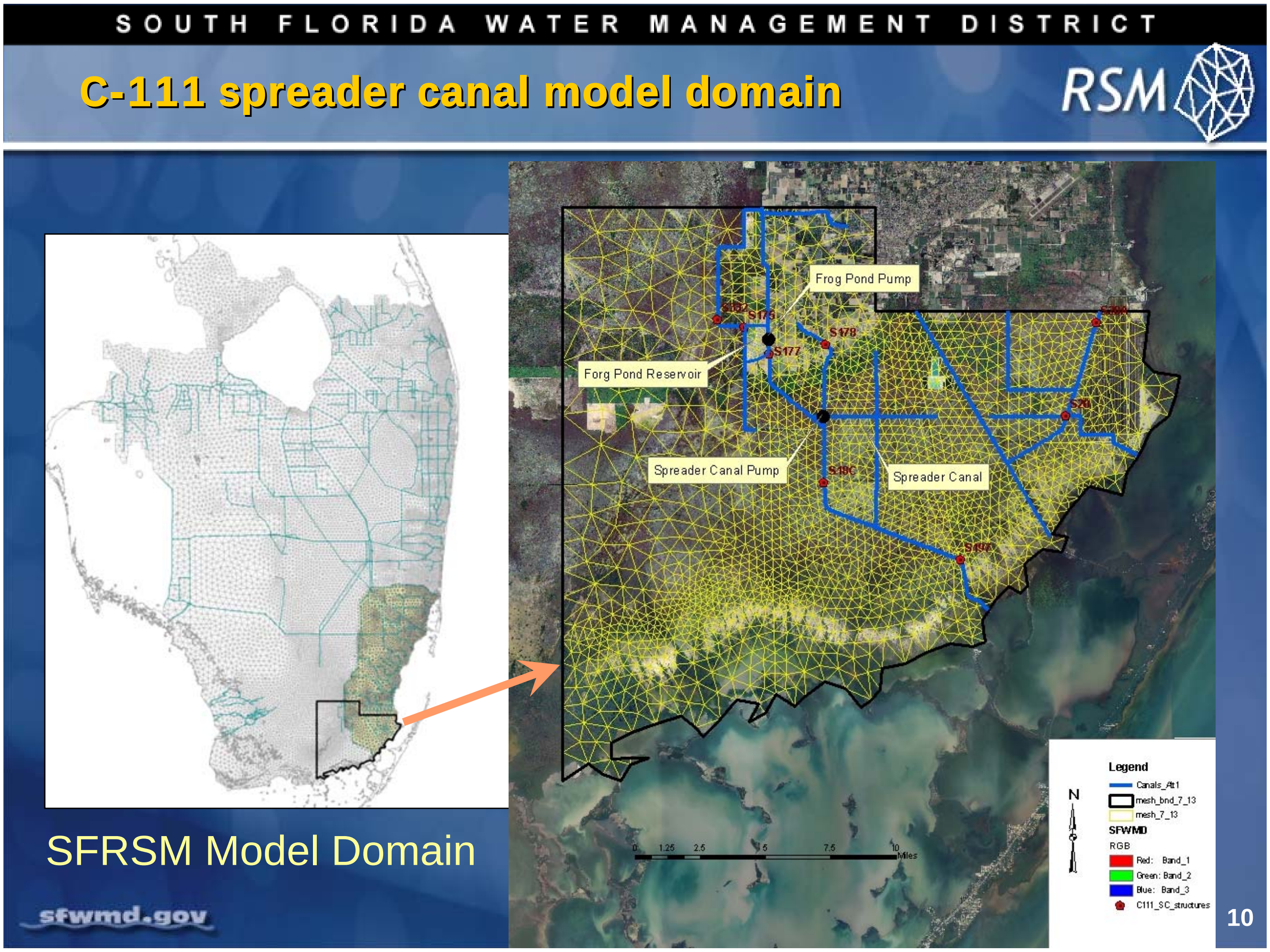
The RSM currently is applied to subregions within the south Florida domain. Each of the subregional models is created to address specific water resource management issues or alternative plan formulations for the Comprehensive Everglades Restoration Plan (CERP).

The subregional models include the C-111, Biscayne Bay Coastal Wetlands, Glades-LECSA and Northern Everglades models. The plug and play concept is the application of the RSM in a linked-node formulation that simulates “basins” and lakes without a mesh or canal network.

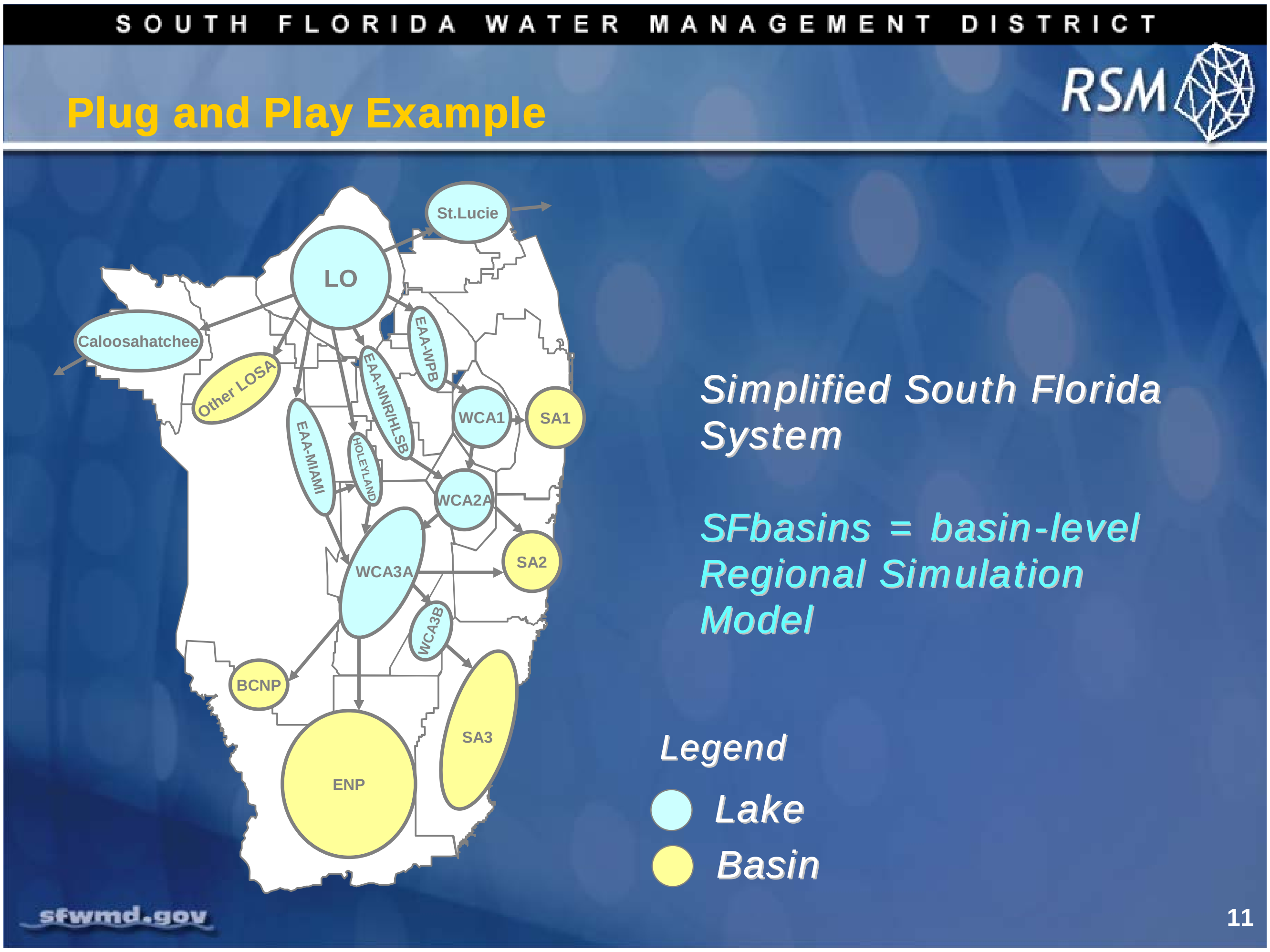


The Glades-LECSA region includes the Miami-Dade, Broward and Palm Beach service areas along the coast of Florida. These service areas are strongly linked to the Water Conservation Areas (WCAs) and the Lake Okeechobee Service Area (LOSA).

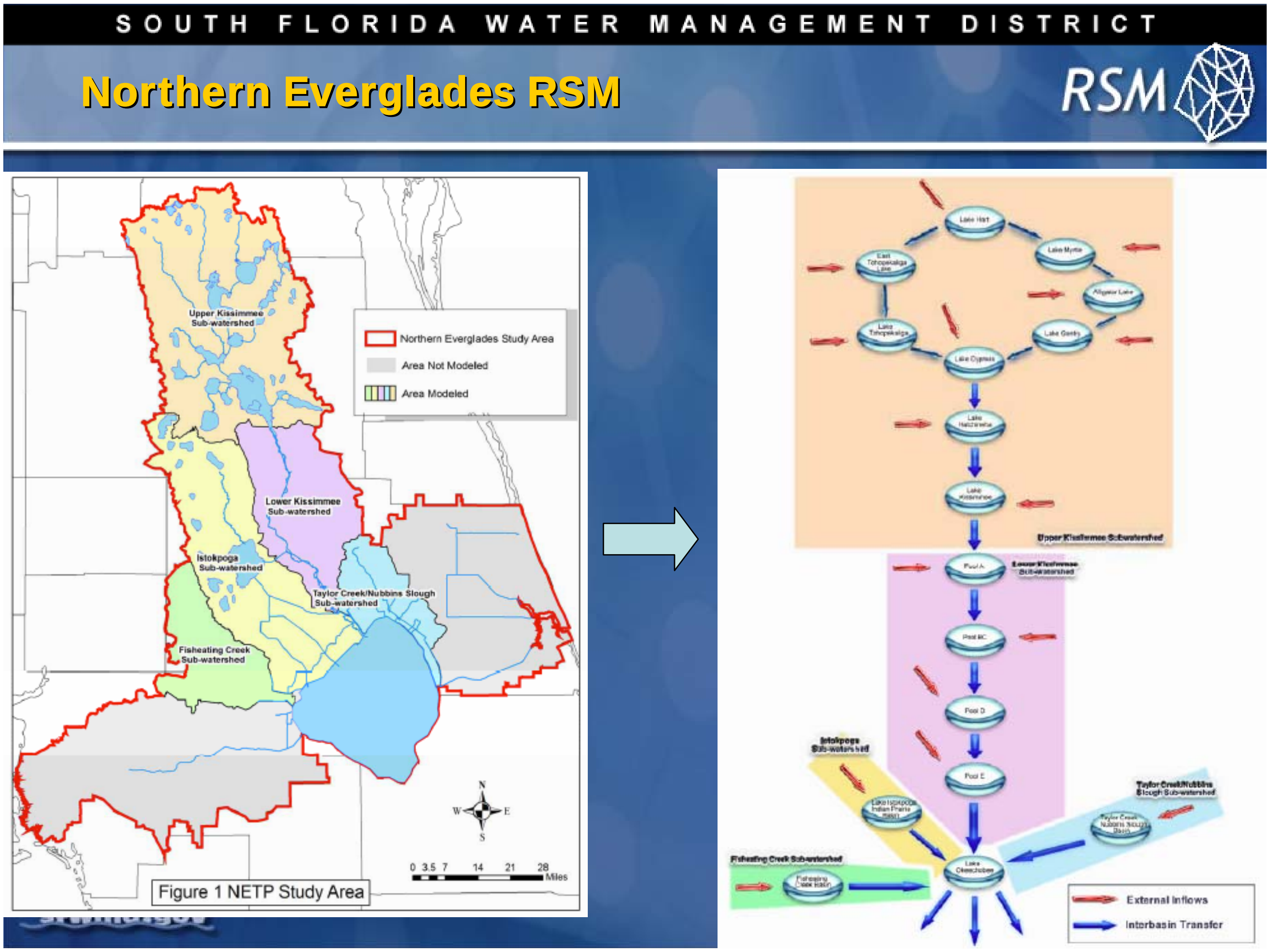
This model links two areas with distinctly different hydrologic processes. The Glades region is primarily low gradient native wetlands and the LECSA is highly developed agricultural and urban land.



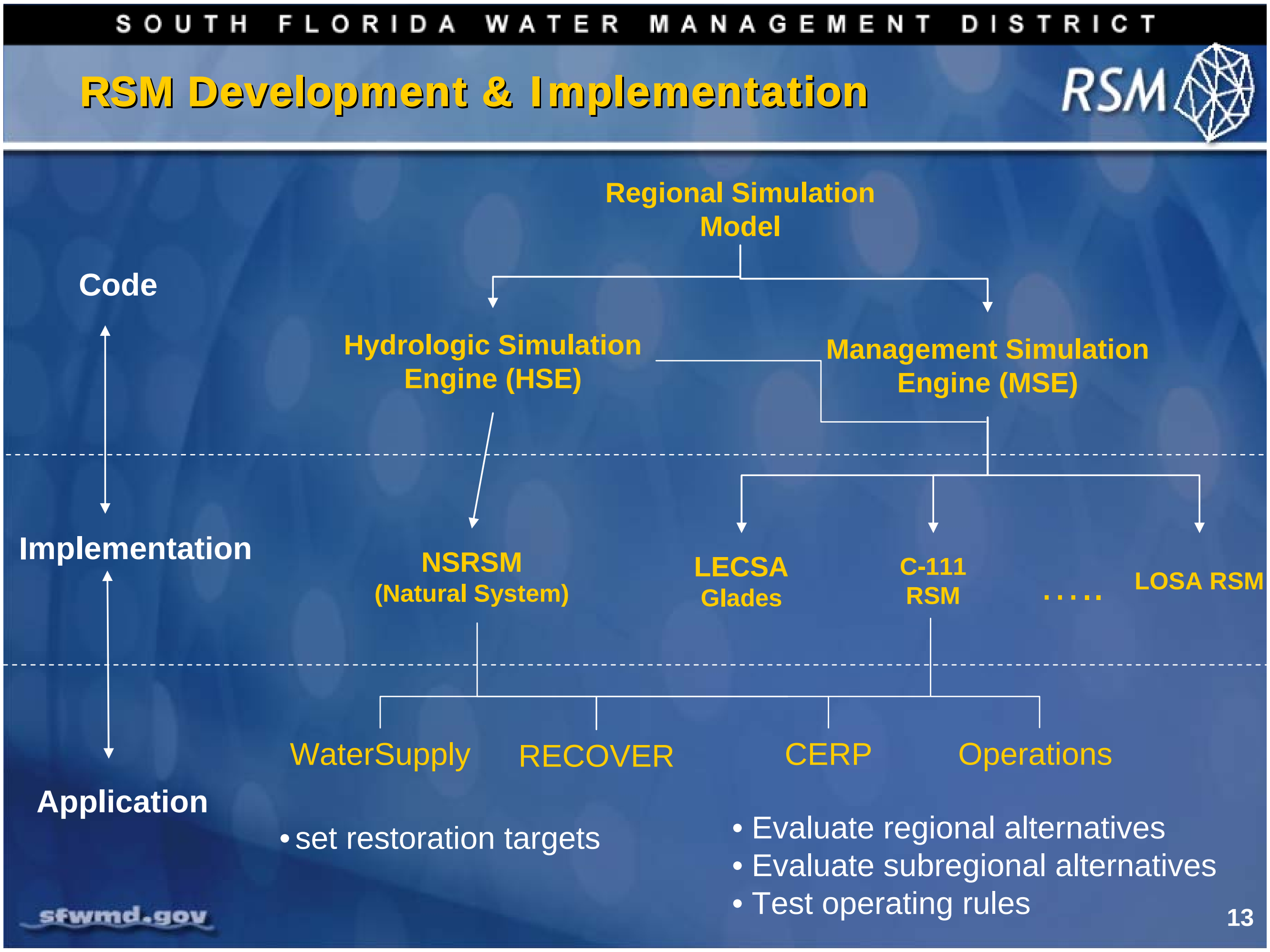
In addition to the larger subregional models, special subregional models are created to address specific questions. The C-111 Spreader Canal model was developed to plan alternatives for improving sheet flow to Florida Bay. The model was developed from the Miami-Dade model, which is part of the LECSA model, and extended into the Everglades to provide a quiescent western boundary. The mesh was designed to provide the appropriate hydrology for selected vegetation zones along the coast. Other specific subregional models will be developed in the future.



The Plug and Play south Florida RSM model was developed to address regional-scale water management rules and policies. The major subregions of south Florida that have controlled inflows and outflows are represented as either lakes or basins. In this implementation, the hydrology within a lake or basin can be represented by simple algorithms or complete subregional mesh and canal components. This provides a useful tool for testing alternative water supply, flood control and environmental water control rules and constraints. It runs fast and has a complete MSE component for specifying rules.

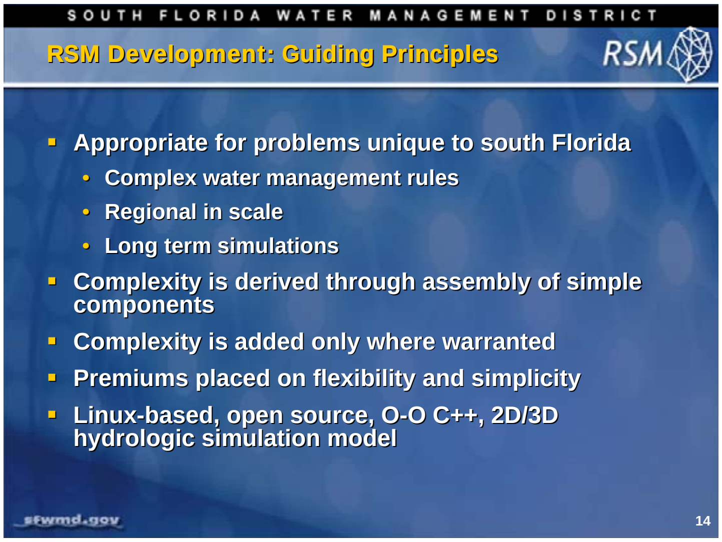


The Northern Everglades model is an example of the plug and plan approach that was developed to evaluate alternative management scenarios for the Lake Okeechobee Technical Plan. The plan addresses water quality treatment projects and water storage requirements to meet the requirements of the *Northern Everglades and Estuaries Protection Program* signed into law, by Florida Governor Crist, in 2007. The model is a linked node model that includes the important lakes and basins in the Northern Everglades and the proposed water storage and stormwater treatment areas (STAs).



The creation and promulgation of the RSM will go through three stages: Code development, Implementation and Application. The code development will include the development of the HSE and the MSE including the benchmarks that demonstrate the functionality of the RSM features.

The RSM is currently implemented through a regional natural system RSM, along with a series of subregional models developed to meet specific requirements. Once the RSM has been implemented and calibrated for the selected subregional domain with the appropriate features, each RSM will be turned over to clients for model application-evaluation of alternative plan formulations.

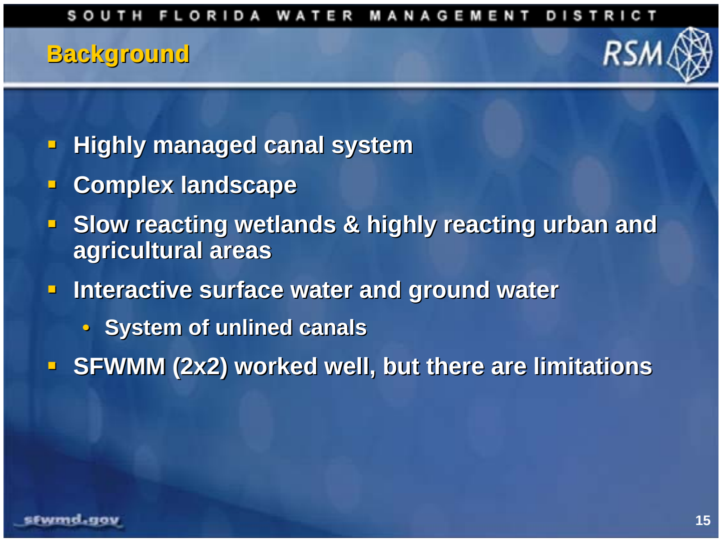


There are several guiding principles for the development of the RSM. These principles drove the early development of the model and guide the creation of new code.

The RSM was developed to resolve regional water distribution and management in our low-gradient physiography where the canal network is intimately connected with the aquifer. Although strongly influenced by local hydrology, the RSM was developed to evaluate water management practices and policies at a regional scale for long simulation periods.

The model is developed based on simple water flow components for groundwater and surface water flow. Although hydrology is a complex non-linear set of processes, the RSM is built from simple components describing the water movement between adjacent mesh cells and other waterbodies. The concept was to create simple components that could be combined to represent the complexity of the system rather than building a highly complex model. This approach is applied to the conceptualization of the hydrology as well as creation of the source code; the methods are small and focused. As a result the model is highly flexible and can be implemented in different ways.

The RSM was implemented in Linux using open source compilers and an open architecture, which enables other modelers to add features in the future.



The development of RSM was predicated on the need for a hydrologic model that could handle south Florida conditions. These include the following:

- A highly managed canal system
- Complex landscape
- Hydrologically responsive systems
- Highly responsive canals

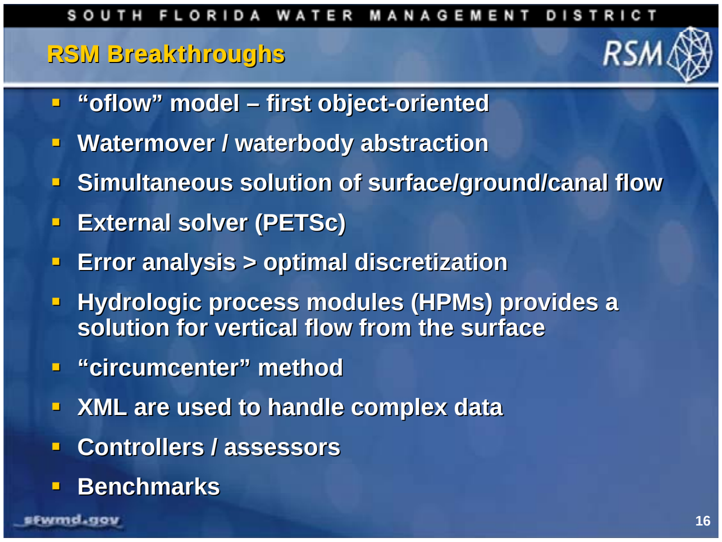
South Florida contains 1,200 miles of primary canals and many more miles of secondary & tertiary canals. The primary canals are part of the C&S Project. The secondary canals are connected to the primary system through pumps or discharge structures. The canal system is managed to provide water for urban and agricultural use, provide drainage for flood control, and provide capacity for maintaining water levels and hydroperiods in the wetlands. This requires a hydrologic model that can simulate surface and groundwater hydrology, interaction with canals and the management practices necessary to move water through the network of canals to meet those needs.

South Florida consists of a mosaic of different landscapes that includes highly managed urban land and agricultural land, and both natural and restored wetlands. These landscape types are juxtaposed in close proximity with little physiographic difference except water management practices.

The hydrologies of the different landscapes have different response times that affect how the hydrologic model must respond. Within the same basins there are urban and agricultural water management systems that respond quickly to rainfall events on water supply demands, while the adjacent wetlands are likely to respond more slowly.

The canals in south Florida are highly connected to the aquifer. Throughout south Florida, the primary aquifer is the Surfical Aquifer System, which has high conductivities. Intersected by deeply cut, unlined canals, there is considerable and rapid water exchange between the canals and the aquifer. During high flows, as canal stages rise, water flows into the aquifer resulting in considerable temporary storage as well as flow around canal structures.

A successful regional Hydrologic Simulation Model must be capable of handling these conditions. When the development of the RSM began 15 years ago there were no models available that could meet these requirements.



There have been a series of breakthroughs that have led to the successful development of the RSM. These are useful to know because they are key to understanding the character and uniqueness of the RSM.

The Oflow program was the first application of an object-oriented model for hydrology and it forms the backbone of RSM.

The waterbody/watermover abstraction led to the development of generic equations for describing water movement between generic waterbodies while satisfying the governing equations.

The generic equations for flow led to the capability of creating a single matrix equation for simultaneous solution for surface water/groundwater/canal flow.

The use of an external sparse matrix solver (PETSc) provided an efficient means for solving the simultaneous equations. The PETSc solver is maintained and upgraded by Argonne National Labs.

The external solver also provided a means for conducting an error analysis of the results that allows for the quantification of errors resulting from poor mesh discretization.

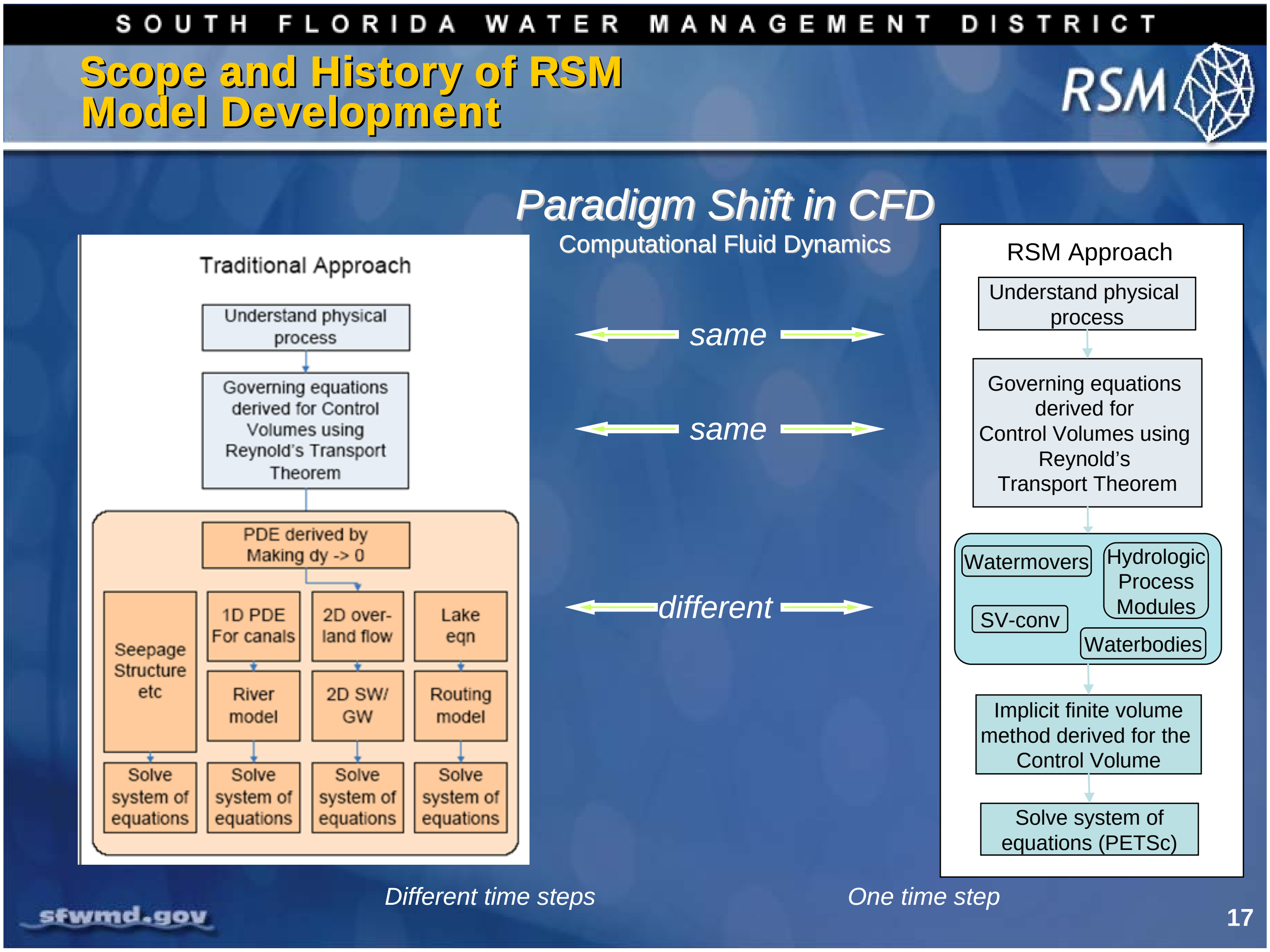
The circumcenter method greatly increases the accuracy of the numerical analysis using irregular triangular meshes.

The HPMs provide a means to include soil processes and local water management systems within cells into the RSM.

The use of the XML allows the RSM to store input data in complex datasets with error checking and referencing of common data.

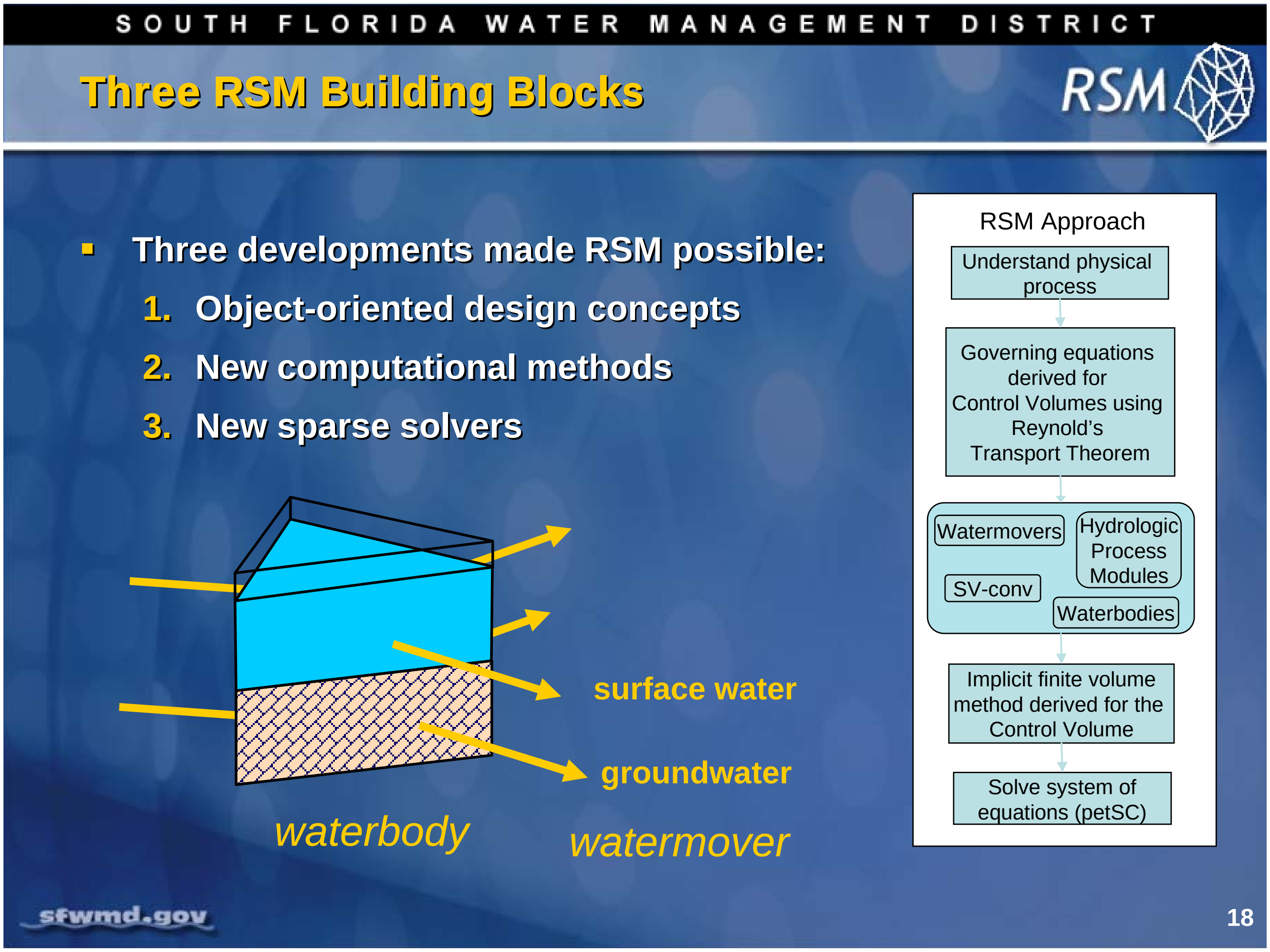
The development of controllers and assessors provides tools for implementing water management rules without changing the core hydrologic model.

The benchmarks demonstrate feature functionality, provide a teaching tool and provide a set of checks for code developers to validate model enhancements to ensure that existing functionality is not broken.

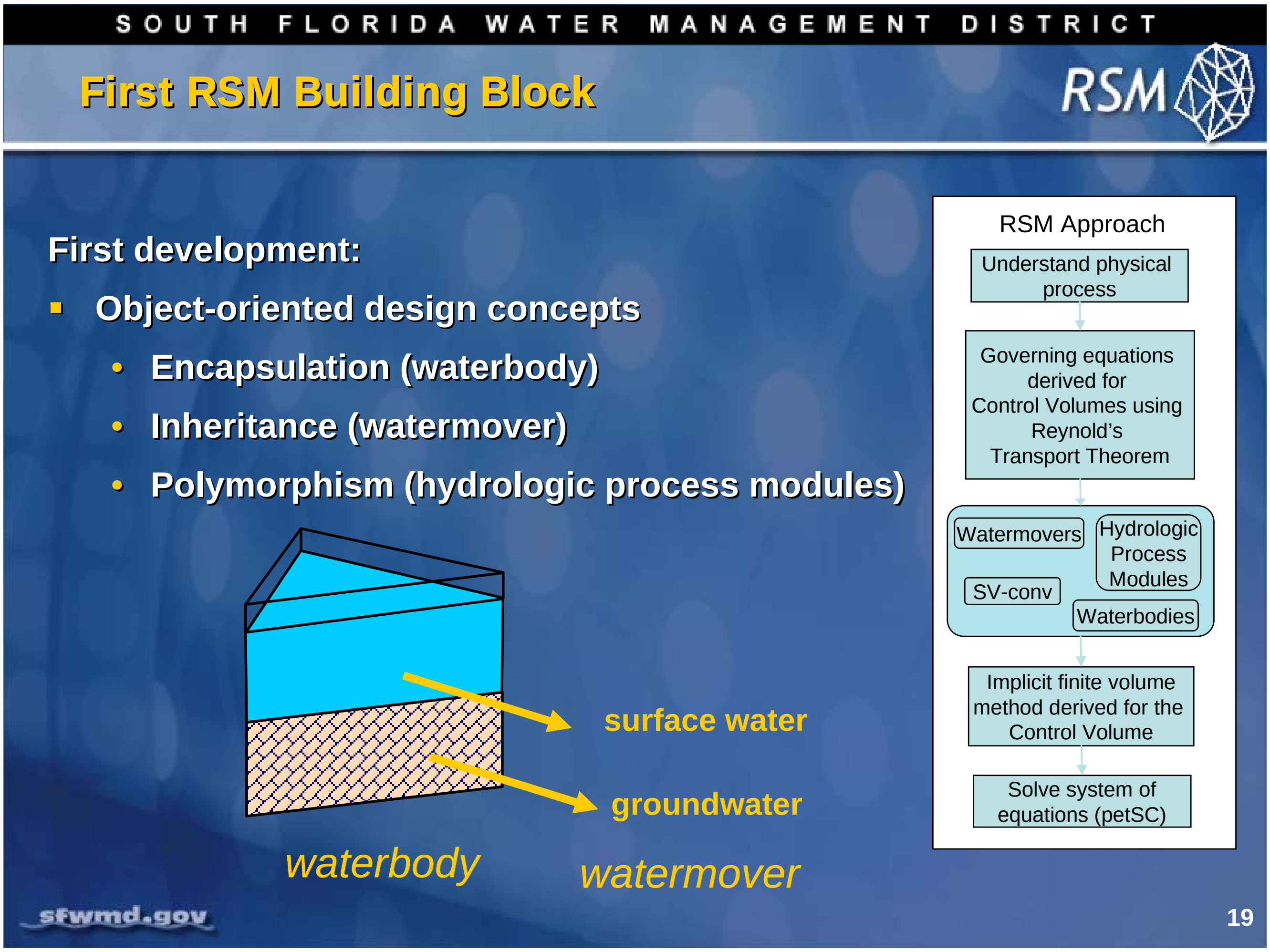


To understand the RSM it is necessary to understand the differences in its development compared with other models. The RSM represents a shift in the development and implementation of hydrologic models. In the past the models were composed of routines describing different hydrologic processes that were solved using different solution techniques.

The RSM is composed of waterbodies and watermovers that are used to construct a single set of simultaneous equations that are solved using a sparse matrix solver. The Hydrologic Process Modules (HPMs) provide a mechanism for solving selected local hydrologic processes at a smaller timestep and applying those results as boundary conditions to the appropriate waterbodies. This greatly reduces the processing time.

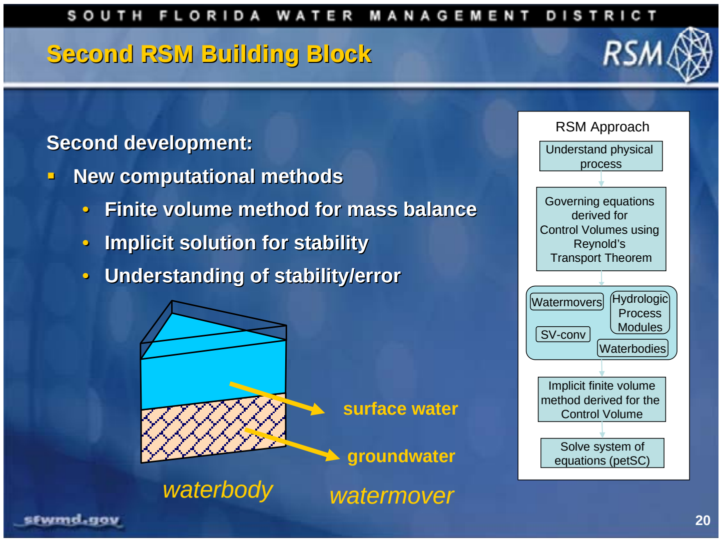


The RSM is constructed differently from many integrated surface and groundwater models. The model is built around the concept of waterbodies and watermovers that move the water between waterbodies. The use of object-oriented concepts allows us to build the RSM in a simple and efficient manner. New computational techniques developed in the last 15 years have enabled implementation of a large-scale hydrologic model. The availability of sparse matrix solvers has provided a means to solve the implicit solution of the RSM efficiently with a tool that is maintained and upgraded by an outside agency.

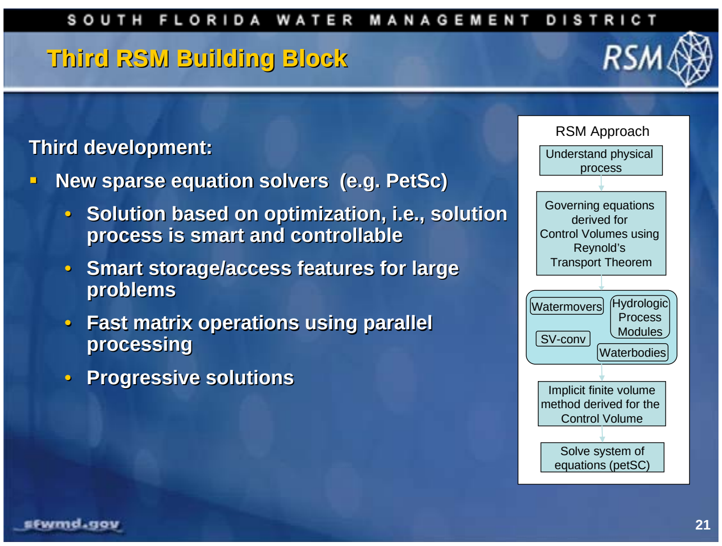


The success of the RSM is based on the implementation of the hydrologic model in object-oriented code. There are several features that make the RSM efficient:

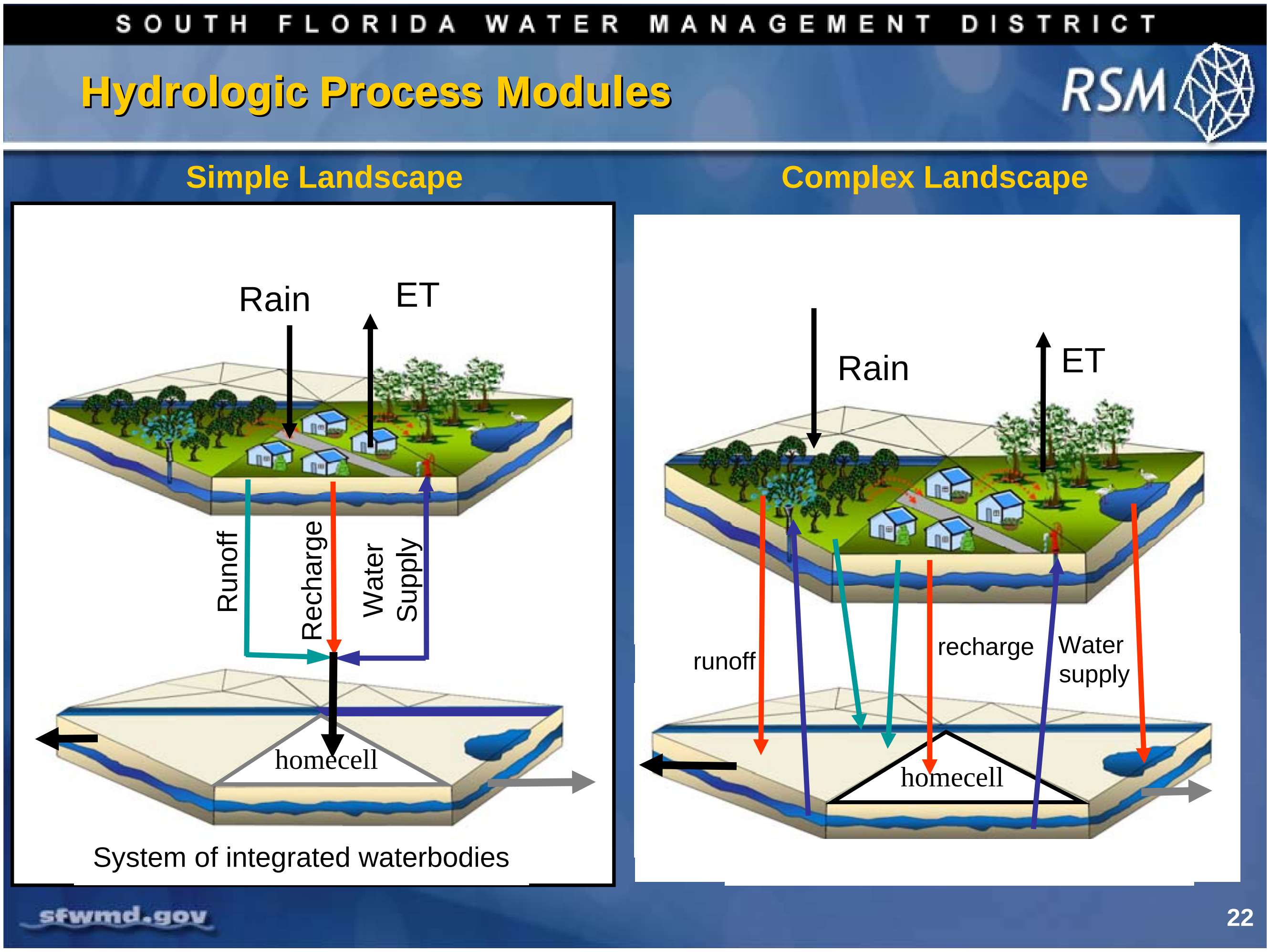
- The concept of encapsulation refers to the capability of attaching the methods for defining the size, shape and connectivity of any waterbody to that waterbody; the user does not need to worry about those functions.
- The concept of a waterbody and a watermover are key components in understanding the RSM. The RSM is a collection of waterbodies and their associated watermovers. The waterbody knows its water volume and depth and can determine the appropriate watermover (e.g., surface water or groundwater). Inheritance allows the watermovers, such as structure flows to inherit the important functions of a "watermover" from the base class and only add the necessary methods for the current watermover (e.g., culvert or pump). This also greatly reduces the amount of redundant code.
- Polymorphism is useful in implementing HPMs where the code for interacting with the mesh is a standard interface that each HPM inherits from the base class.
- The object-oriented code allows the developers to create many alternative process algorithms that are only implemented (instantiated) through the XML input files.



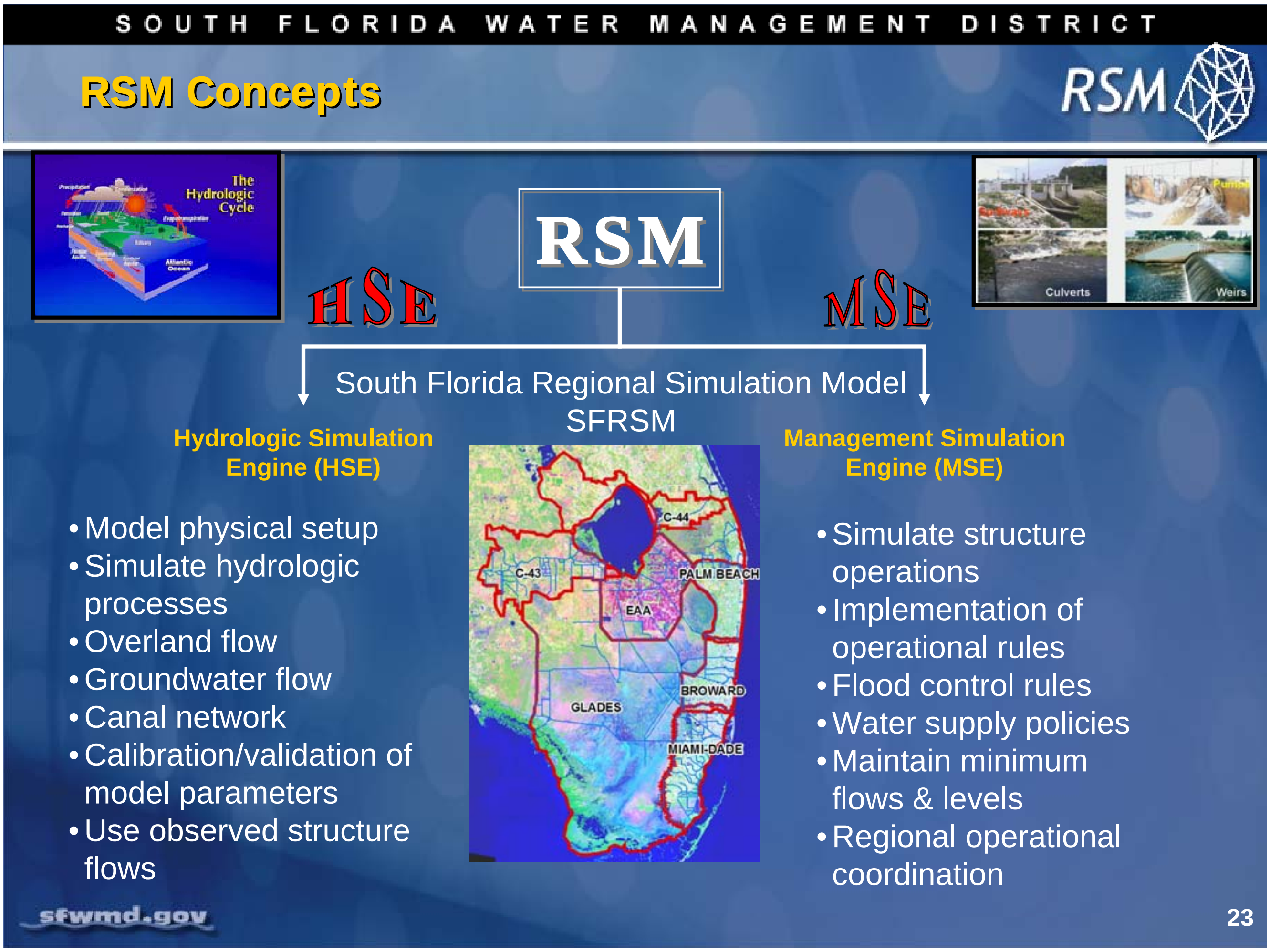
The second important development that led to the creation of the RSM was the availability of new computational methods. The finite volume method allowed for the implementation of the governing equations in the integral form rather than the differential form which in turn allowed for the creation of the control volumes for waterbodies and water balances for those control volumes. This, in turn, led to the creation of a simple set of equations that could be assembled into a single matrix for an implicit solution. With the finite volume method there are suitable ways to evaluate the amount of error that occurs for any formulation of the waterbodies (mesh discretization) and watermovers (transport equation). The implementation method also provides a means of estimating the error in the water balance directly.



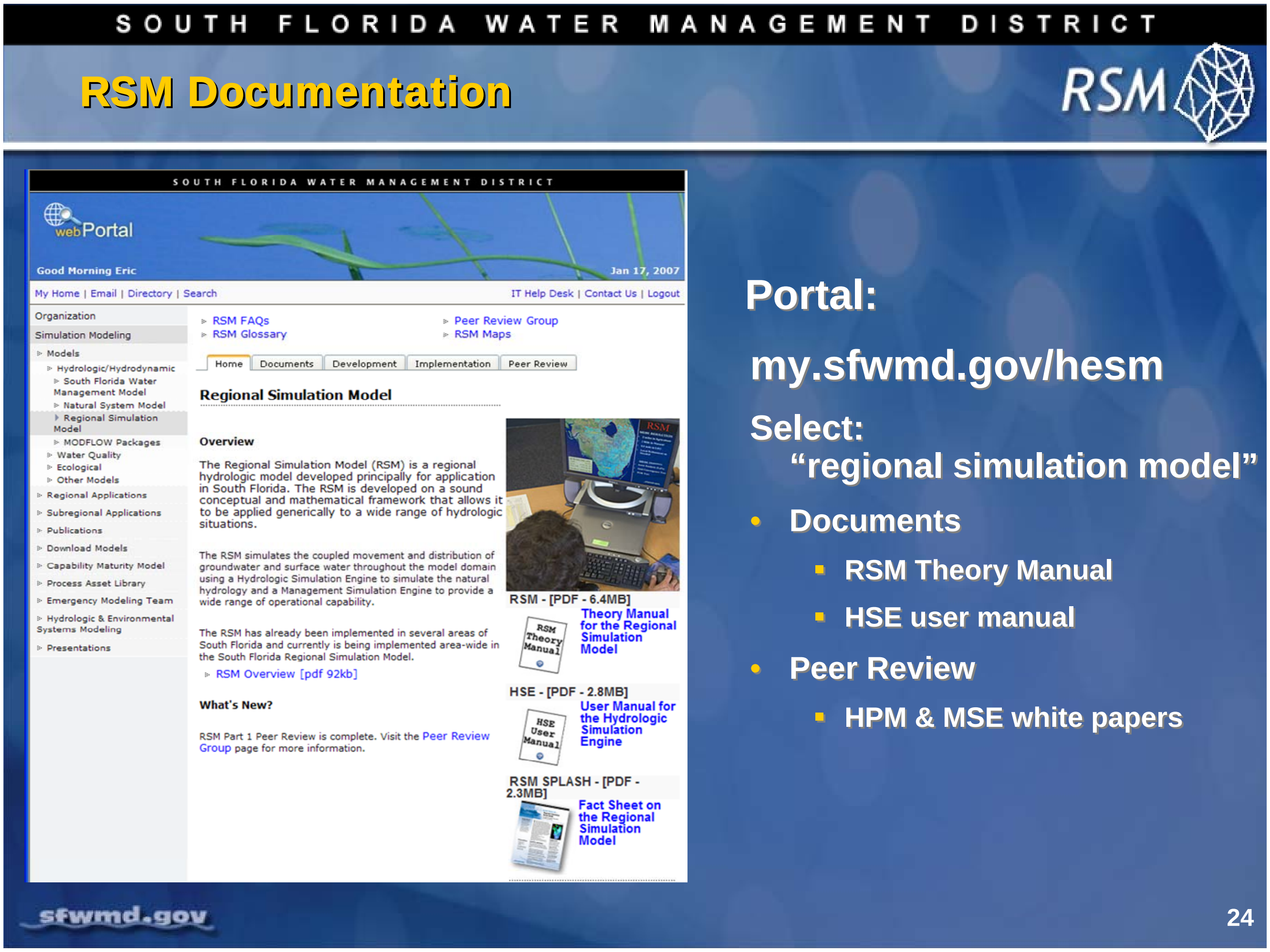
A third breakthrough was the development of new matrix solvers. The matrix for solving a regional-scale RSM contains the flow equations for as many as 24,000 cells and 5,000 canal segments. However, the matrix is sparse with only three to five terms per row/column in the matrix. New sparse solvers such as PETSc, developed by Argonne National Labs, have provided means for efficiently solving the RSM matrix. The PETSc is an open source software package that is maintained and upgraded by Argonne. This tool has allowed the development of the RSM to be focused on additional hydrologic and water management features.



The Hydrologic Process Modules (HPMs) were developed to provide the upper boundary conditions to the 2D-mesh. In the simplest case, the HPMs process rain and potential evapotranspiration and produce recharge to the cell. For complex landscapes that include agricultural and urban water management systems, the HPMs simulate the effects of irrigation, drainage, runoff, urban consumptive use and sewage disposal. The resulting stressors can be directed to any waterbody. The HPMs can also simulate detailed infiltration and percolation in different soil types. These HPMs can be spatially distributed and implemented for multiple cells or within single cells. This provides RSM with a high degree of flexibility in modeling the local hydrology.



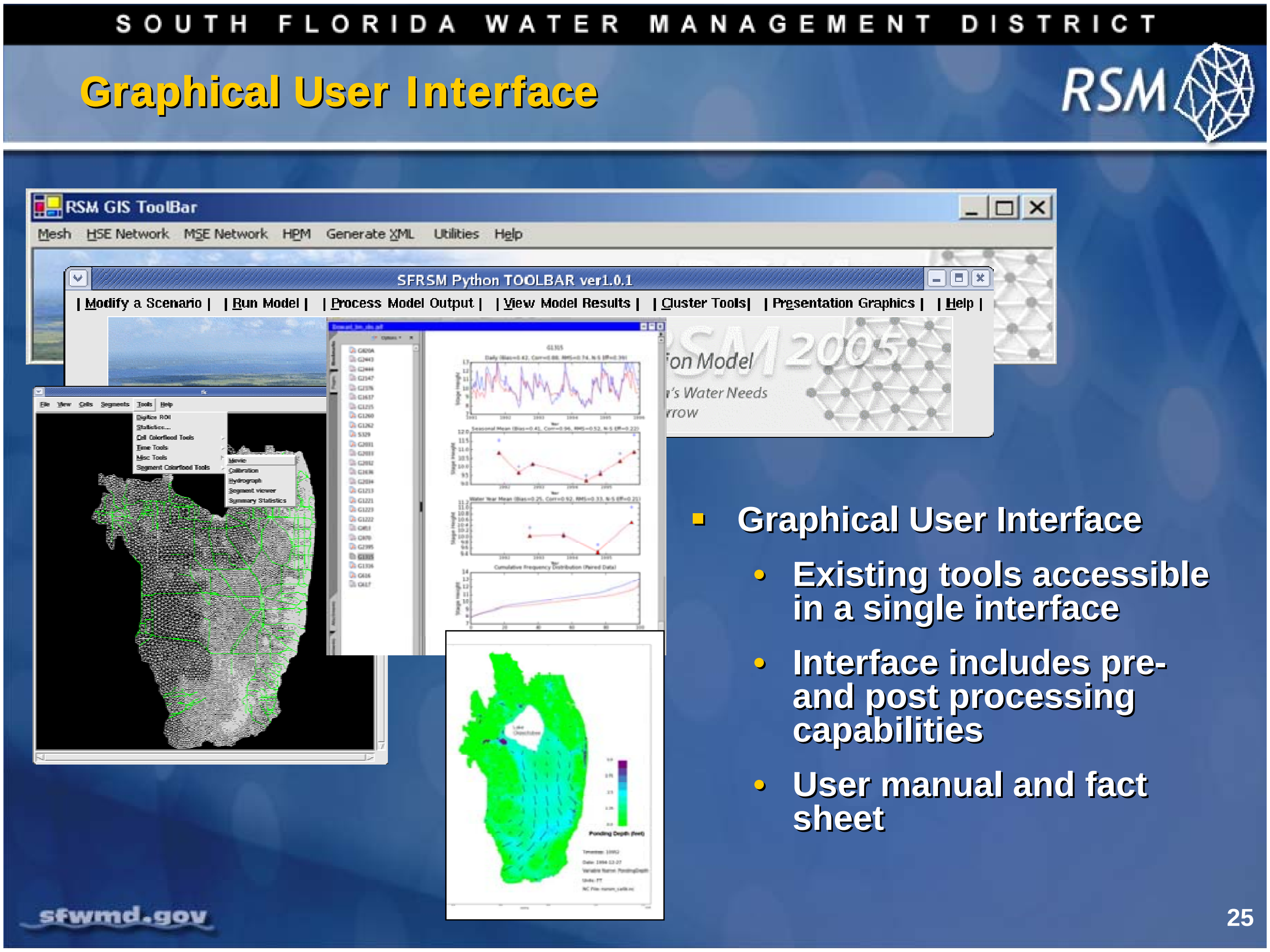
The RSM consists of two engines that coexist in the same model. The Hydrologic Simulation Engine (HSE) simulates the hydrologic processes. And, the Management Simulation Engine (MSE) models the effect of operational rules and management practices on the movement of water in the regional canal system. The MSE has been developed to operate separately from the HSE using the same system monitors to quantify the state of the system. The MSE operates at different levels to provide management control at the regional level, basin level and control of flow at individual structures.



The *Theory Manual for the Regional Simulation Model* and *User Manual for the Hydrologic Simulation Engine* for the RSM in available on the South Florida Water Management District's internet portal.

- From my.sfwmd.gov/hesm, select the Regional Simulation Model.

Documents describing additional RSM components and subregional models, as well as background reference documents detailing the fundamental equations underlying the RSM are located under 'Documentation and Peer Review' on the portal webpage.



The typical input for the RSM is a set of simple ASCII files for model control and parameter values. Typical output includes .DSS files for time series data (upper graph) and netCDF files for water budgets and flow vector files (lower graphic). Additionally, a collection of custom GIS utilities, the RSMGIS toolbar, is under development. It helps create the RSM input data files from a SFRSM geodatabase and user interaction.

A separate GUI (RSM GUI) is being developed to post-process the model output to provide water budgets, time series graphs, and animations. The RSMGIS toolbar and RSM GUI will reduce the time for model implementations, reduce input errors, and record model run statistics and model versions. These tools are frequently updated to provide additional functionality.

KNOWLEDGE ASSESSMENT

1. What was the motivation for developing the RSM?
2. What are the two components of the RSM?
3. List the three components of the RSM-HSE?
4. What does the Management Simulation Engine do?
5. How is the RSM-HSE solved?
6. What are the two versions of the RSM?
7. What are the guiding principles in the RSM development?
8. What is the paradigm shift in the RSM development?
9. What were the key breakthroughs in the RSM development?
10. How do we solve the system of flow equations for head?
11. What are the purposes of HPMs?

Answers

1. RSM was developed to provide a more flexible and faster model to provide a tool for evaluating alternative plans for water resource management projects.

2. The RSM is composed of the Hydrologic Simulation Engine (HSE) and the Management Simulation Engine (MSE).

3. The HSE is composed of the waterbodies, watermovers and hydrologic process modules.

4. The MSE simulates the operation of the structures, implements water management rules and policies, and coordinates the regional canal system.

5. The RSM-HSE is a system of equations for surface, groundwater and canal flow that is solved **implicitly** at each timestep.

6. The two RSM versions are the mesh/canal network and the linked basin-lake implementations.

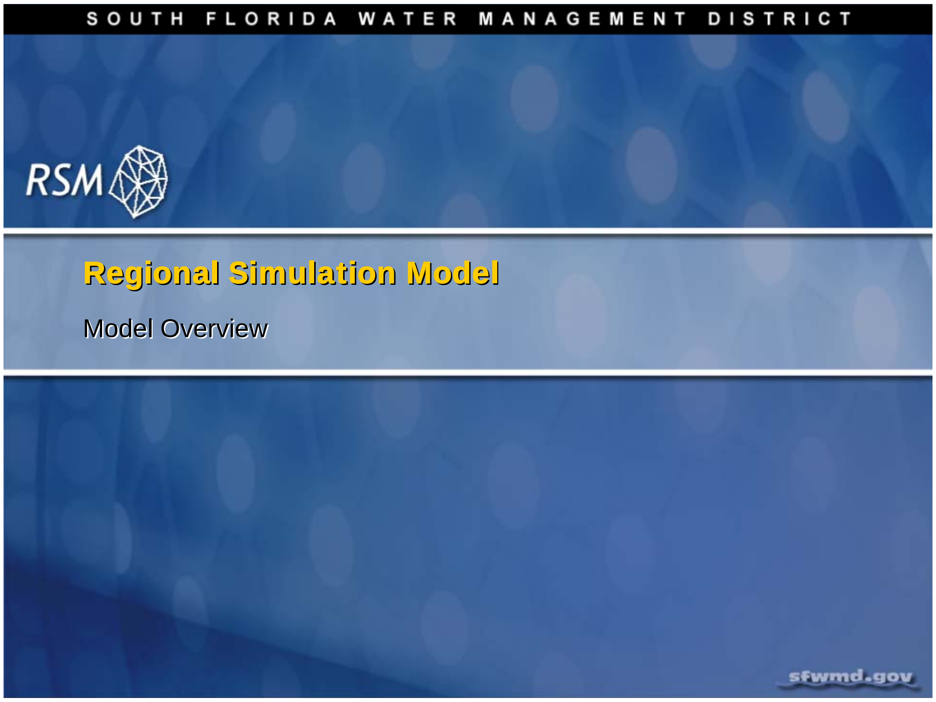
7. The guiding principles for RSM development include the following
    a. keep it simple; add complexity only as necessary
    b. complexity is derived through assembly of simple components
    c. object oriented C++
    d. open source
    e. long-term regional simulations

8. The paradigm shift in RSM is the solution of the flow equations for all hydrologic processes in a single time step rather than different systems of equations for different processes.

9. There were several key breakthroughs for RSM to be successful including:
    a. waterbody/watermover abstraction
    b. adoption of the “circumcenter” method
    c. use of an external matrix solver
    d. adopting XMLs for handling input data
    e. developing benchmarks to document and test model features
    f. utilizing HPMs for surface water hydrology in a complex landscape

10. The system of equations for surface, groundwater and canal flow is solved using an external matrix solver.

11. The HPMs were developed to handle the vertical solution of water infiltration and percolation through the soil. HPMs can handle the surface hydrology of the complex landscape of urban-agricultural-native land uses in South Florida.



Lab 1: RSM Benchmarks

Time Estimate: 1.5 hours

Training Objective: Compile and run RSM and gain familiarity with RSM Benchmarks and model output

The Regional Simulation Model (RSM) consists of several directories that include the source code, benchmarks, documents and supporting libraries and utilities. This lab session explores the components of the RSM and how to run the model, both from the command line and through the RSM Graphical User Interface (RSM GUI).

The benchmarks are an important component of the RSM. They document the various features of the model, provide a quality assurance check for developers to make sure that any future development does not break any current features, and serve as a training tool for new users to explore the functionality of the model features.

NOTES:

For ease of navigation, you may wish to set an environment variable to the directory where you install the RSM code using the syntax

```
setenv RSM <path>
```

Modelers at the District should use the following NAS path:

```
/nw/oomdata_ws/nw/oom/sfrsm/workdirs/<username>/trunk
```

```
setenv RSM
/nw/oomdata_ws/nw/oom/sfrsm/workdirs/<username>/trunk
```

Once you set the RSM environment variable to your trunk path, you can use $RSM in any path statement, such as:

```
cd $RSM/benchmarks
```

Training files are currently located in the following directories:

```
INTERNAL_TRAINING
|
|____data
|         |__geographic
|         |__C111
|         |__rain+et
|         |__glades-lecsa
|         |__losa_eaa
|         |__BBCW
|
|____trunk
|         |___benchmarks
|         |___hpmbud
|
|____labs
```

Files for this lab are located in the **`labs/lab1_BM1`** directory. Additional materials in the directory include:

`RSM_HPM_whitepaper.doc` (*Hydrologic process modules of the regional simulation model: An overview*. [Flaig, E.G., R. Van Zee, and W. Lal, 2005])

`Petsc_manual.pdf`

`hecdss.pdf`

`RSMGUI_Chapter10.doc`

`SimpleWaterBudgets.doc`

Activity 1.1: Implement RSM GUI (SFWMD modelers only)

Overview

Activity 1.1 addresses the first part of the lab objective, compile and run RSM. There are two exercises:

- **Exercise 1.1.1** Compile and run the RSM
- **Exercise 1.1.2** Run Benchmark 1 (BM1)

The start-up procedures will depend on how the RSM is implemented on your system. The RSM may be implemented on a computer running the Linux Redhat5.0 operating system located on your network or on a local Linux laptop. We assume that you are beginning at the Linux prompt in the directory that you have created that contains The RSM (for modelers at the South Florida Water Management District, this would be where you checked out the trunk from the RSM repository).

Exercise 1.1.1 Compile and run the RSM

1. Navigate to the RSM directory: **`cd $RSM`**

Directories in Trunk directory:	Contents
Assessor	User specified assessors.
Benchmarks	70 benchmarks and *.dtd files.
Budtool	Budtool water budget utility.
fcl_lib	Fuzzy controller source files.
Glpk	General linear programming package source files.
Hpmbud	HPM water budget utility.
Libutils	Various data input/output utility programs
Mse_tools	
Python	
Src	
Utils	

Files in Trunk directory:	
Makefile	Compiles and links source files for running RSM.

2. Compile RSM: **`$RSM/make`**
3. Navigate to the Benchmark directory: **`cd $RSM/benchmarks`**
4. Run test.script: **`. $RSM/benchmarks/test.script`**

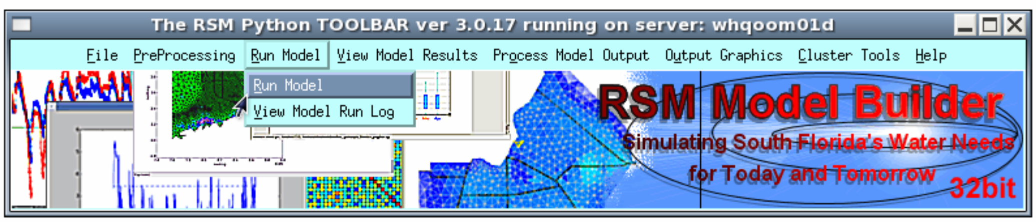
This runs each benchmark and does a binary compare [diff command] of the current results with the results from a previously run. This ensures that the RSM is running correctly. This is necessary to run each time the RSM is installed on a new computer and recompiled. Compile time will range from 5-45 minutes depending on your computer setup.

Benchmarks

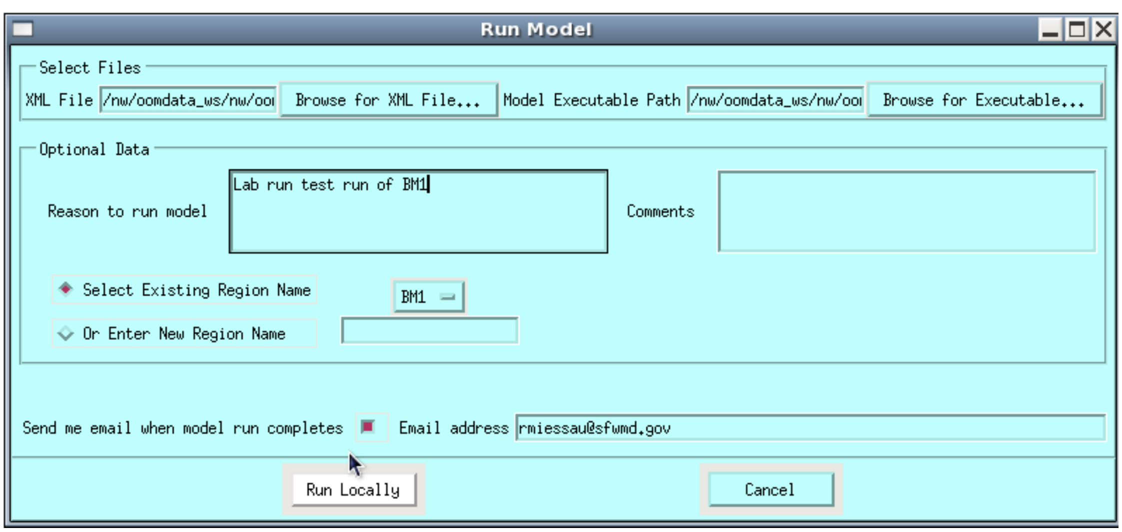
In the benchmark directory you will find benchmark directories from BM1 to BM76. Each directory contains the input files for running the benchmark. The descriptions.pdf file contains a brief description of each benchmark. The features tested in each benchmark are presented in (**Table 1.1**). This is helpful when looking for the benchmark that includes the appropriate syntax for a specific feature. The benchmarks are useful for determining the impact of selected variables of the feature on the RSM output. We will run a simple benchmark to illustrate how the RSM runs.

Exercise 1.1.2 Run Benchmark 1

5. Navigate to Benchmark 1: **`cd $RSM/benchmarks/BM1`**
6. Start the RSM Graphical User Interface (RSMGUI) **`rsmgui`**
   - Run BM1 using the **Run Model** tool found under the Run Model dropdown menu:



7. Select: **`Run Model`**
   - Add a note such as **“`Lab run test run of BM1`”**



8. Click on **"Browse for XML File…"**
   - Select **/benchmark/BM1** directory
   - Select **"run3x3.xml file"** and press **"open"**
9. Click on **"Browse for Executable"**
   - set directory for correct executable file **/trunk/src/hse**
10. Set up optional data as defined below:

Features in the Run Model tool	Description	
XML File	The RUN XML to be used to make this run	**../benchmarks/BM1/ run3x3.xml**
Model Executable Path	The compiled HSE executable	**../trunk/src/hse**
Reason to Run Model	Optional comment to document the reason for making this run	
Comments	Optional comments about this run	
Select Existing Region Name	Select the region (names) for this run from a dropdown list of previously made runs [BM1] or enter a new region name	**BM1**
Enter New Region Name	Input the region (name) for this run and it will be added to the dropdown list for future use in this tool	
Send Email	Email address where an automated email notice will be sent upon completion of the run	**enter your home email address**
Run Locally	Execute this run on the server where the RSMGUI is being run	

When the model runs the typical runtime output will occur in a new terminal window. The typical output includes several sections. A critical section is the model setup. If there is an error in the input it will stop in this section. Error handling in the RSM is primitive but this section will indicate where in the code the model stopped and thus where you should look for resolving the problem.

11. Click on **"Run Locally":**

Look at the results:

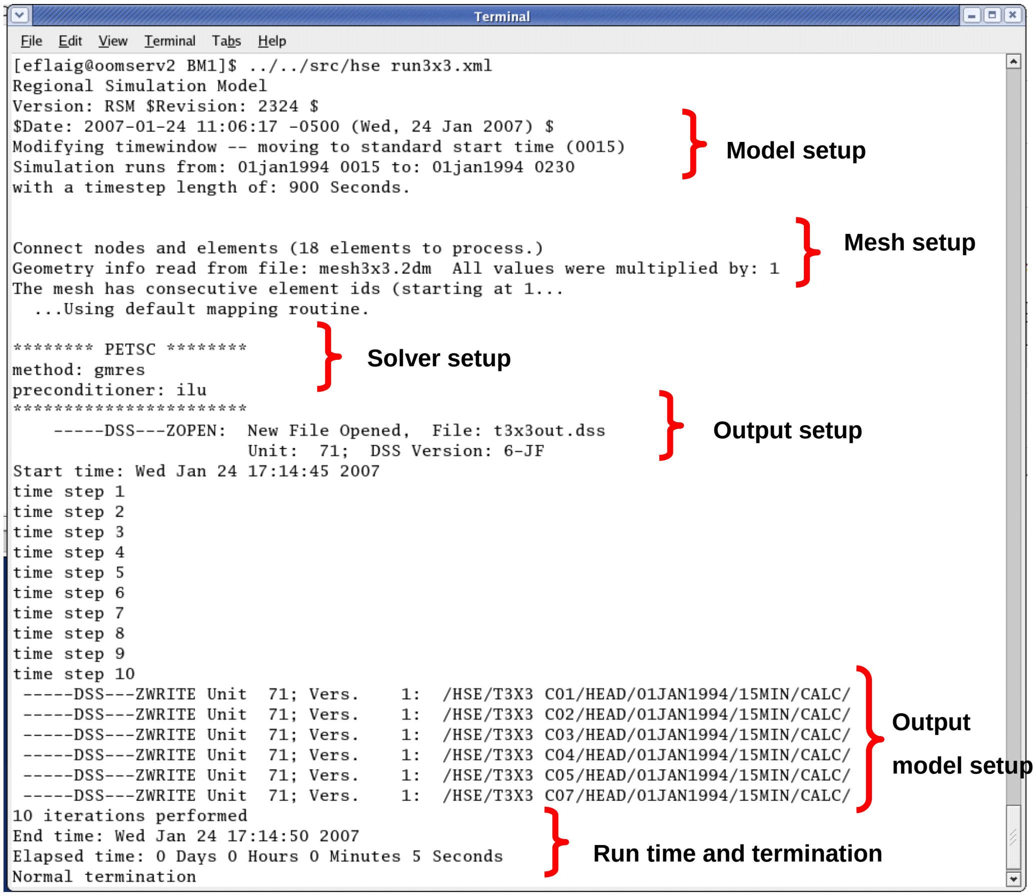
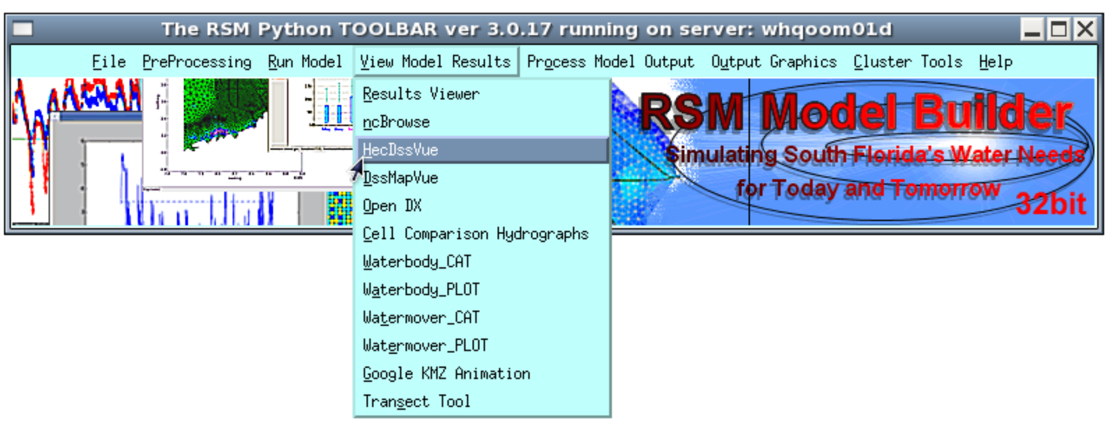


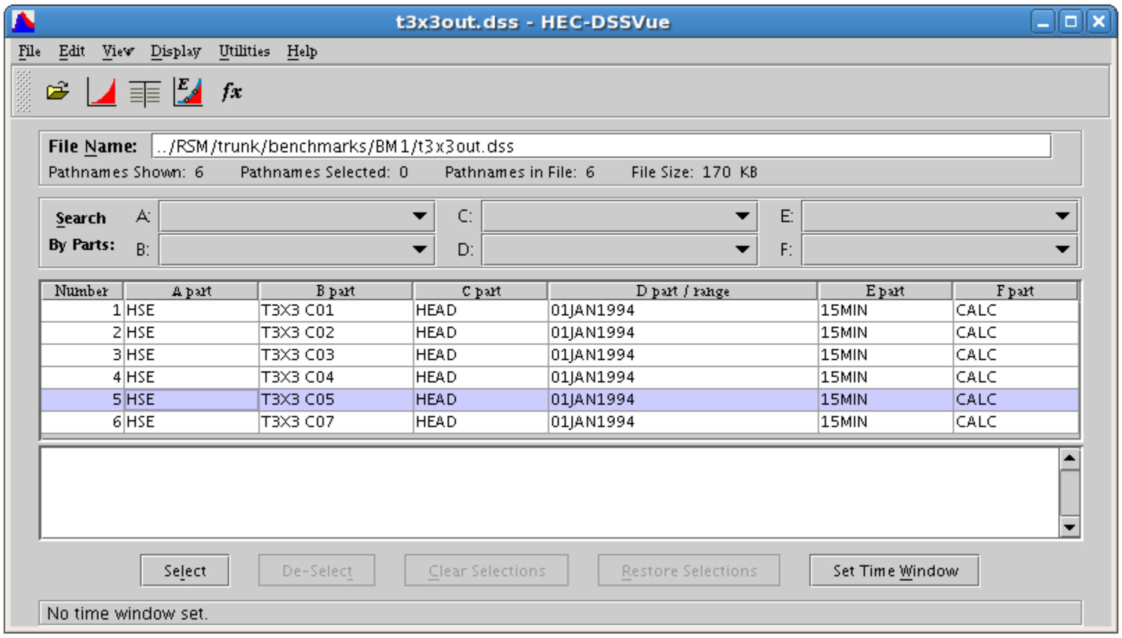
Figure 1.1 Typical runtime output for the RSM

12. Select **HecDssVue** from the RSM GUI found under the **View Model Results** dropdown menu.



- A HEC-DSSVue window will open.

13. Open the **`t3x3out.dss`** file. Select the record for head for **Cell 5**. Graph the head for **Cell 5** by selecting the red graph icon.



- Do not close the Graph window; you will want to compare this graph to the next graph.

14. Click on the "Preprocessing" tab on the RSM GUI and select "Edit an XML file". Click on "Browse" and navigate to **`run3x3.xml`** file in **`BM1`** folder.
15. Edit **run3x3.xml** file:

```
gedit run3x3.xml
```

16. Locate sub-element <mannings> under <mesh>; change the value of **"a"** from **1.0** to ***0.5***.
17. Save **run3x3.xml**.
18. Rerun the RSM and plot the graph of Cell 5 head again.
    - What are the differences?
19. Modify **run3x3.xml**: Change ***transmissivity*** from **0.0** to ***0.2***.
    - What are the changes?
20. Open **hin3x3.dat** using gedit or other editor.
    - (This file contains the initial heads in the grid in consecutive order. Notice that there is a 1-meter groundwater mound in Cell 5 and Cell 14).
21. Change the head in Cell 5 and Cell 14 from **502.0** to ***504.0*** and run the RSM.
    - What are the results?

This example illustrates how to change the parameter values in the main XML file of a benchmark and observe the impact on the results.

> **NOTE: There are two good editors available in Linux:** gedit **and** xemacs**.**

Table 1.1 Quick Reference for Benchmarks

1	Conveyance	43b	Vector
2	Conveyance & transmissivity	44	~~Upwind method for conveyance~~
3	Canal flow	45	User ctrl controller
4*	Stream bank (SP, OB, SB)	45a	User specified controller
5	Single control WM (cells)	45b	User specified supervisor
6	Steady State flow	45c	User-c-ctrl.xml – user controller
7*	Wells BCs	45d	User-c-supervisor.xml-
8	5layer HPM	45e	User-specified fuzzy controller
9	Dual control WM	46	Kala Basin, S.L. application (PRR HPM)
10	Head BC (cells & segments)	47	GLPM supervisor
11	Stream bank water mover	48	~~MSE Network + Graph~~
12	Cell general head BCs	49	MSE Network + User defined supervisor
13	Lake & Pond	50	Hub HPM
14	Culverts water movers	51	Imperv HPM
15	Indexed entry HPMs	52	UrbanDet HPM
16	NSML layer HPM	53	Urban Hub HPM
17	SVconverter lookup table	54	Urban Hub HPM with RO + WS routing
18	Unsat HPM	55	Urban Hub HPM with CU
19	Output: netCDF	56	PRR (Nam) HPM
20	Single controller (cell + segments)	57	Hub HPMs
21	Single controller (c+s)		- one-2-many
22	Pipes		- pumpedditch HPM
23	Weirs		- agimp HPM
24	Indexed entry rain + refET	58	Lake BCs
25	MBU cell HPM	59	Berm seepage
26	Bleeders watermover	60	Trigger
27	Streambanks	61	~~Square-wave propagation~~
28	Delta control Pumps	62	ORM Supervisor
29	Water Distribution	63	MSE_Network & WCU Assessor
30	ENP Testbed	63b	MSE Network + WMM
31	ENP (separate conveyance)		Assessor (mse assessor)
32	Distributed Rain +	63c	Dual WMM Assessor +
	alternative output formats		Reach-WMM Assessor
33	AFSIRS HPM	64	Test canal vertices
33r*	RAMCC HPM	65	Water Control Districts
34	L8 basin application	66	Levee seepage
35	Wall General head BC	67	~~Mixed Kinematrc cump~~
36	Conveyance + transmissivity	68	Genxweir water mover
	(lookup tables)	69	
37	Svconverter lookup table	70	MSE network and Basin Assessor
38	Cadlec mannings	70a	Run 25
39	Multi-Layered aquifer	70b	LOSA
40	PID controller	71	Impoundments
40a	H-q – relationship	72	genManning WaterMover for Impoundments
40b	Gateweir	73	Northern Everglades RSM testbed
41	Set point controller	74	Trigger module for PWS cutback
42	~~General head BC~~	74b	Trigger Module for PWS& irrigation cutback
43	Fuzzy controller	75	Conveyance cap on Manning's n
43a	Scalar	76	Stage-based structure flow management.

NOTE:

Benchmarks marked with strikethroughs are not currently active features.

Activity 1.2: Investigate the RSM and Run Benchmarks BM4, BM9, BM16

Overview

Activity 1.2 addresses the second part of the lab objective "gain familiarity with RSM Benchmarks", and will have three exercises:

- **Exercise 1.2.1** Run Benchmark 4 (stream bank seepage)
- **Exercise 1.2.2** Run Benchmark 9 (dual controllers on a canal)
- **Exercise 1.2.3** Run Benchmark 16 (nsm1layer HPM)

In this activity you will investigate the behavior of several features of the RSM model by running different benchmarks. These benchmarks include BM4 (stream bank seepage), BM9 (dual controllers on a canal) and BM16 (nsm1layer HPM). Each of these benchmarks presents different features and different outputs. The typical benchmark consists of a square mesh with 18 cells and a simple canal with four segments. These benchmarks use the same mesh and canal (**Fig. 1.2**).

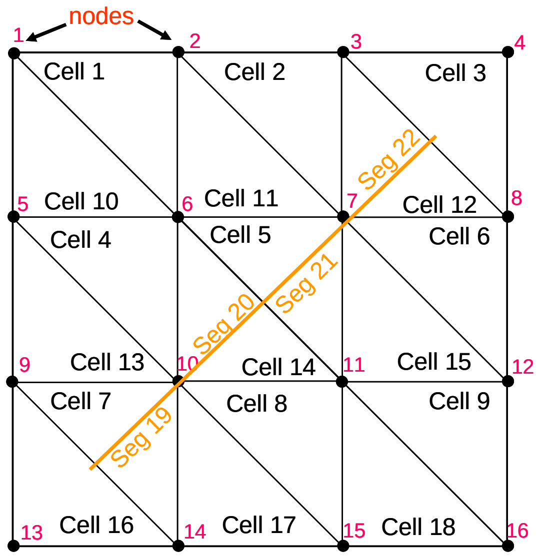


Figure 1.2 Standard 3x3 mesh and four-segment canal for benchmarks

Exercise 1.2.1 Run Benchmark 4 (stream bank seepage)

Benchmark 4 was created to test the interaction between cell waterbodies and segment waterbodies. The interaction consists of groundwater seepage and overbank flow. The initial conditions (**canal3x3.init** and **hin3x3.dat**) place a 2-foot pond in selected cells and segments.

If the heads are provided to the cells or segments in order:

- In which cells and segments does the ponding occur?

Look at the control section of **run3x3.xml** (**Fig. 1.3**)

- What is the model timestep and duration of the simulation?

22. Run the model (**run3x3.xml**) from the RSM GUI:

```
cd $RSM/benchmarks/BM4
rsmgui
```

- Click on **"Run Model",** browse for XML file in **BM4** directory
- Click on **"Run Locally"**

23. Observe results.

- Edit **overbank.dat** from the RSM GUI and observe the volume of overbank for each timestep.
- Run **HecDssVue**
- Open **t3x3out.dss**, select the appropriate records, and graph cell and segment heads.
- Check output block of **run3x3.xml** to make sure cell and segment IDs are correct.

24. Edit **run3x3.xml**.

- Change the streambank seepage, **bank_coeff,** from **0.001** to **1.0.**

25. Rerun model.
26. Regraph cell and segment heads using the **t3x3out.dss** file.

- How did the values change?

NOTE:

Answers can be found in the Lab 1 directory in file answers_lab1.pdf.

```xml
<?xml version="1.0" ?>
<!DOCTYPE hse SYSTEM "../hse.dtd" [
]>
<hse version="0.1">
 <control
      tslen="15"
      tstype="minute"
      startdate="01jan1994"
      starttime="0000"
      enddate="01jan1994"
      endtime="0230"
      alpha="0.500"
      solver="PETSC"
      method="gmres"
      precond="ilu">
 </control>
 <network>
  <geometry file="canal3x3.map"> </geometry>
  <initial file="canal3x3.init"> </initial>
  <arcs>
      <indexed file="arcs.index">
      <xsentry id="1">
            <arcflow n="0.2"></arcflow>
            <arcseepage leakage_coeff="0.000405" />
                  <arcoverbank bank_height="0.001" bank_coeff="0.001"/>
            </xsentry>
      </indexed>
      </arcs>
 </network>
 <mesh>
       <geometry file="mesh3x3.2dm"> </geometry>
       <shead><gms file="hin3x3.dat"></gms></shead>
       <bottom> <const value="0.0">  </const> </bottom>
      <surface> <const value="500.0"> </const> </surface>
      <conveyance>
            <mannings a="1.000" detent="0.00001"></mannings>
      </conveyance>
      <transmissivity>
            <unconfined k = "0.02"> </unconfined>
      </transmissivity>
      <svconverter>
      <constsv sc="0.2"> </constsv>
      </svconverter>
 </mesh>
 <output>
  <segmentmonitor id="19" attr="overbankflow">
   <asciiform file="overbank.dat"></asciiform>
  </segmentmonitor>
```

Bank seepage (→ bank_coeff="0.001")

Mannings n (→ mannings a="1.000")

Figure 1.3 XML input file **`run3x3.xml`** for Benchmark 4

```
  <budgetpackage file="budget.nc"></budgetpackage>
  <cellmonitor id="1" attr="head">
    <dss file="t3x3out.dss" pn="/hse/t3x3 c01/head//15min/calc/"> </dss>
  </cellmonitor>
  <cellmonitor id="2" attr="head">
    <dss file="t3x3out.dss" pn="/hse/t3x3 c02/head//15min/calc/"> </dss>
  </cellmonitor>
  <cellmonitor id="3" attr="head">
   <dss file="t3x3out.dss" pn="/hse/t3x3 c03/head//15min/calc/"> </dss>
  </cellmonitor>
  <cellmonitor id="4" attr="head">
    <dss file="t3x3out.dss" pn="/hse/t3x3 c04/head//15min/calc/"> </dss>
  </cellmonitor>
  <cellmonitor id="5" attr="head">
    <dss file="t3x3out.dss" pn="/hse/t3x3 c05/head//15min/calc/"> </dss>
  </cellmonitor>
  <segmentmonitor id="19" attr="head">
    <dss file="t3x3out.dss" pn="/hse/t3x3 s01/head//15min/calc/"> </dss>
  </segmentmonitor>
  <segmentmonitor id="20" attr="head">
    <dss file="t3x3out.dss" pn="/hse/t3x3 s02/head//15min/calc/"> </dss>
  </segmentmonitor>
  <budget file="budget.out"></budget>
 </output>
</hse>
```

Figure 1.3 (continued) XML input file **run3x3.xml** for Benchmark 4

Exercise 1.2.2 Benchmark 9 (Dual controllers on a canal)

Benchmark 9 was developed to test the use of a dual control pump where the pump rate is determined by the head in the pumped out waterbody and the receiving waterbody. As can be seen from the **`run3x3.xml`** (**Fig. 1.4**) the flow rate for **`watermover 101`** is determined by a matrix. Water flows from Cell 5 into **`Segment 21`**. There is no reverse flow and no gravity flow.

The initial conditions are the same as in BM4. The bank height is set at 0.3 m.

- What was the bank height in BM4?

27. Run **Benchmark9** using **`BM9/run3x3.xml`** using the RSMGUI.
28. Observe the results:
    - Run **`HecDssVue`**.
    - Graph the heads in the cells and Segment 20.
    - How do these heads compare to the heads from BM4?
29. Modify the **`bank_height`**:
    - Edit **`run3x3.xml`**.
        - Change bank height **`0.3`** to **`0.01`**.
30. Rerun **`run3x3.xml`**.
31. Regraph the heads.
        - How did the heads change?
32. Change **`leakage_coeff`** from **`0.000405`** to **`0.01`**.
33. Rerun **`run3x3.xml`**.
34. Regraph the heads.
    - How did the heads change?

The benchmark provides a “testbed” for exploring the impact of changing parameter values on the heads and water budgets of the waterbodies. It should be realized that the benchmarks are limited because they have fixed, no-flow boundary conditions and thus are only appropriate for short duration simulations.

```
<?xml version="1.0" ?>
<!DOCTYPE hse SYSTEM "../hse.dtd" [
]>
<hse version="0.1">
 <control
      tslen="15"
      tstype="minute"
      startdate="01jan1994"
      starttime="0000"
      enddate="02jan1994"
      endtime="0230"
      alpha="0.500"
      solver="PETSC"
      method="gmres"
      precond="ilu">
 </control>
 <network>
      <geometry file="canal3x3.map"> </geometry>
      <initial file="canal3x3.init"> </initial>
      <arcs>
     <indexed file="arcs.index">
      <xsentry id="1">
       <arcflow n="0.2"></arcflow>
        <arcseepage leakage_coeff="0.000405"></arcseepage>
        <arcoverbank bank_height="0.3" bank_coeff ="0.001"></arcoverbank>
      </xsentry>
     </indexed>
   </arcs>
 </network>
 <mesh>
      <geometry file="mesh3x3.2dm"> </geometry>
      <shead><gms file="hin3x3.dat"></gms></shead>
      <bottom> <const value="0.0">  </const> </bottom>
      <surface> <const value="500.0"> </const> </surface>
      <conveyance>
        <mannings a="1.000" detent="0.00001"></mannings>
      </conveyance>
      <transmissivity>
     <unconfined k = "0.02"> </unconfined>
      </transmissivity>
      <svconverter>
     <constsv sc="0.2"> </constsv>
      </svconverter>
 </mesh>
 <watermovers>
      <dual_control wmID="101" id1="5" id2="21" control="5" cutoff="495"
                   gravflow="no" revflow="no" label="t3x3-1">
             495  500  505  510
       495.0   0 1000 2000 3000
       500.0   0    0 1500 2500
       505.0   0    0    0 2000
       510.0   0    0    0    0
  </dual_control>
```

Figure 1.4 XML input file run3x3.xml for Benchmark 9

```
      <dual_control wmID="102" id1="14" id2="20" control="14" cutoff="495"
                   gravflow="no" revflow="no" label="t3x3-2">
       495 500 505 510
    495.0  0 1000 2000 3000
    500.0  0  0 1500 2500
    505.0  0  0  0 2000
    510.0  0  0  0  0
      </dual_control>
 </watermovers>
 <output>
      <cellmonitor id="1" attr="head">
      <dss file="t3x3out.dss" pn="/hse/t3x3 c01/head//15min/calc/"> </dss>
      </cellmonitor>
      <cellmonitor id="2" attr="head">
      <dss file="t3x3out.dss" pn="/hse/t3x3 c02/head//15min/calc/"> </dss>
      </cellmonitor>
      <cellmonitor id="3" attr="head">
      <dss file="t3x3out.dss" pn="/hse/t3x3 c03/head//15min/calc/"> </dss>
      </cellmonitor>
      <cellmonitor id="4" attr="head">
      <dss file="t3x3out.dss" pn="/hse/t3x3 c04/head//15min/calc/"> </dss>
      </cellmonitor>
      <cellmonitor id="5" attr="head">
      <dss file="t3x3out.dss" pn="/hse/t3x3 c05/head//15min/calc/"> </dss>
      </cellmonitor>
      <segmentmonitor id="19" attr="head">
<asciiform file="s19.dat"> </asciiform>
      </segmentmonitor>
      <segmentmonitor id="20" attr="head">
      <asciiform file="s20.dat"></asciiform>
      </segmentmonitor>
      <segmentmonitor id="21" attr="head">
      <asciiform file="s21.dat"></asciiform>
      </segmentmonitor>
      <segmentmonitor id="22" attr="head">
      <asciiform file="s22.dat"></asciiform>
      </segmentmonitor>
      <segmentmonitor id="20" attr="head">
      <dss file="t3x3out.dss" pn="/hse/t3x3 s01/head//15min/calc/"> </dss>
      </segmentmonitor>

  <budget file="budget.out"></budget>
  <budgetpackage file="budget.nc"></budgetpackage>

 </output>
</hse>
```

Figure 1.4 (Continued) XML input file run3x3.xml for Benchmark 9

Exercise 1.2.3 Run Benchmark 16 (nsm1layer HPM)

Benchmark 16 introduces the <nsm1layer> Hydrologic Process Module (HPM). This HPM is designed to calculate the actual evapotranspiration (ET) based on the crop rooting depth, crop ET adjustment coefficient (Kveg), depth of the water table and the reference crop ET time series.

The benchmark uses three types of HPMs (**Fig. 1.5**); one <layer5>, <layer1nsm> with constant Kveg, and <layer1nsm> with seasonal Kveg. The outputs include the netCDF files for the waterbody and the HPM water budgets, and ASCII files that contain the head data for cells with the different HPM types.

We want to change the output specification from ASCII to .DSS file formats and then change the netCDF files from daily values to monthly values and run the simulation for an entire year.

35. Go to the Benchmark 16 directory:

```
cd $RSM/benchmarks/BM16
```

36. Edit **run3x3.xml**.Change **cell monitor output** from:

```
<asciiform file="c01.head" format="%10.6f">  </asciiform>
<asciiform file="c04.head" format="%10.6f">  </asciiform>
<asciiform file="c07.head" format="%10.6f">   </asciiform>
```

to:

```
<dss file="recharge.dss"
pn="/c1/layer5/recharge//1day/calc/"></dss>
<dss file="recharge.dss"
pn="/c4/1layer/recharge//1day/calc/"></dss>
<dss file="recharge.dss"
pn="/c7/1layer/recharge//1day/calc/"></dss>
```

37. Change the enddate to: **01jan1995**
38. Add wbbudgetpackage to the <output> block in the **run3x3.xml** file:

```
<wbbudgetpackage file="wbbudget.nc" />
```

39. Run the model in the RSM GUI: **rsmgui**
40. Process the water budgets for the three cells in file **run3x3.xml** by using the RSM GUI.

In the RSM GUI select “Process Model Output” and “WBBud”. In WBBud Window:

- Select NetCDF file: **“wbbudget.nc”**
- Select IDs Subset: **1**
- Multiplier: **12** (for inches)
- -t Transform: select **“Depth”**

- Select appropriate output file in the **lab1_BM1 directory** (browse to the appropriate directory)
- Enter **"next"**

41. Graph the heads for the cells using **HecDssVue**.
42. Modify the **Kveg** values from **0.75** to **0.80** and rerun the model look at the water budgets.
    - How does this affect the water budgets?
43. Modify the rooting depth **(xd)** from **2.0** to **1.0**.
    - How does this affect the water budgets?
    - How does this affect the cell heads (water table elevation)?

```
<?xml version="1.0" ?>
<!DOCTYPE hse SYSTEM "../hse.dtd" [
]>
<hse version="0.1">
 <control
       tslen="24"
       tstype="hour"
       startdate="01jan1994"
       starttime="0000"
       enddate="31jan1994"
       endtime="2400"
       alpha="0.500"
       solver="PETSC"
       method="gmres"
       precond="ilu">
 </control>
 <mesh>
       <geometry file="mesh3x3.2dm"> </geometry>
       <mesh_bc>
             <noflow section="ol_gw">
                  <nodelist> 5 6 7 8 </nodelist>
             </noflow>
             <noflow section="ol_gw">
                   <nodelist> 9 10 11 12 </nodelist>
             </noflow>
      </mesh_bc>
      <shead><gms file="hin3x3.dat"></gms></shead>
      <rain> <const value="0.0"> </const> </rain>
      <refet> <const value="0.05"> </const> </refet>
      <bottom> <const value="0.0">  </const> </bottom>
      <surface> <const value="500.5"> </const> </surface>
       <hpModules>
         <indexed file="lu.index">
           <hpmEntry id="1">
            <layer5 ew="0.2" kw="1.0" rd="0.5" xd="2.0" pd="3.0" kveg="0.75">
                  </layer5>
           </hpmEntry>
           <hpmEntry id="2">
                   <layer1nsm kw="1.0" rd="0.5" xd="2.0" pd="3.0" kveg="0.75"
                        imax="0.0">
            </layer1nsm>
```

```
          </hpmEntry>
```

Figure 1.5 XML input file run3x3.xml for Benchmark 16

```
          <hpmEntry id="3">
                 <layer1nsm kw="1.0" rd="0.5" xd="2.0" pd="3.0" kveg="0.00"
                 imax="0.2">
                       <ampmod para="kveg">
                            1     0.75
                            15    0.75
                            16    1.0
                            365   1.0
                       </ampmod>
                 </layer1nsm>
           </hpmEntry>
       </indexed>     </hpModules>
     <conveyance>
           <mannings a="1.000" detent="0.00001"></mannings>
     </conveyance>
     <transmissivity>
           <unconfined k = "0.02"> </unconfined>
     </transmissivity>
     <svconverter>
           <constsv sc="0.2"> </constsv>
     </svconverter>
 </mesh>
 <output>
     <cellmonitor id="1" attr="head">
           <asciiform file="c01.head" format="%10.6f">  </asciiform>
     </cellmonitor>
     <cellmonitor id="4" attr="head">
           <asciiform file="c04.head" format="%10.6f">  </asciiform>
     </cellmonitor>
     <cellmonitor id="7" attr="head">
           <asciiform file="c07.head" format="%10.6f">   </asciiform>
     </cellmonitor>
     <globalmonitor attr="head">
           <gms file="outheads.dat">      </gms>
     </globalmonitor>
     <budget file="budget.dat"></budget>
     <psbudgetpackage file="pseudo.nc"></psbudgetpackage>
     <psbudgetpackage file="pseudo_mo.nc" dbintl="43200"></psbudgetpackage>
     <budgetpackage file="budget.nc"></budgetpackage>
  </output>
</hse>
```

Figure 1.5 (Continued) XML input file run3x3.xml for Benchmark 16

Activity 1.3: Create Simple Water Budgets and Monitors

Overview

Activity 1.3 addresses the third part of the lab objective "gain familiarity with model output" and will have three exercises:

- **Exercise 1.3.1** Review time series data
- **Exercise 1.3.2** Review water budget data for waterbodies
- **Exercise 1.3.3** Review water budget data for HPMs

Run Benchmark 33r and create output:

44. Go to **BM33r** directory:

```
cd $RSM/benchmarks/BM33r
```

45. Edit **run3x3.xml**. Observe the elements in the <output> block.

There are three types of outputs from the RSM:

Monitors	Provide a time series of selected state variables for a waterbody and flows
Budgetpackages	Create water budgets for selected water bodies and sets of waterbodies
Globalmonitors	Output a state variable for each cell

In Benchmark 33r, there are the following outputs:

Cell Monitors	What is monitored? And what is the output file?
Hpmmonitors	What is monitored? And what is the output file?
Budget	Produces the primitive water budget for each waterbody.
Budgetpackage	Produces necessary information to construct a water budget for every waterbody
Hpmbudgetpackage	Produces the necessary information to construct a water budget for every HPM.

46. Run `BM33r/run3x3.xml` using the RSM GUI.

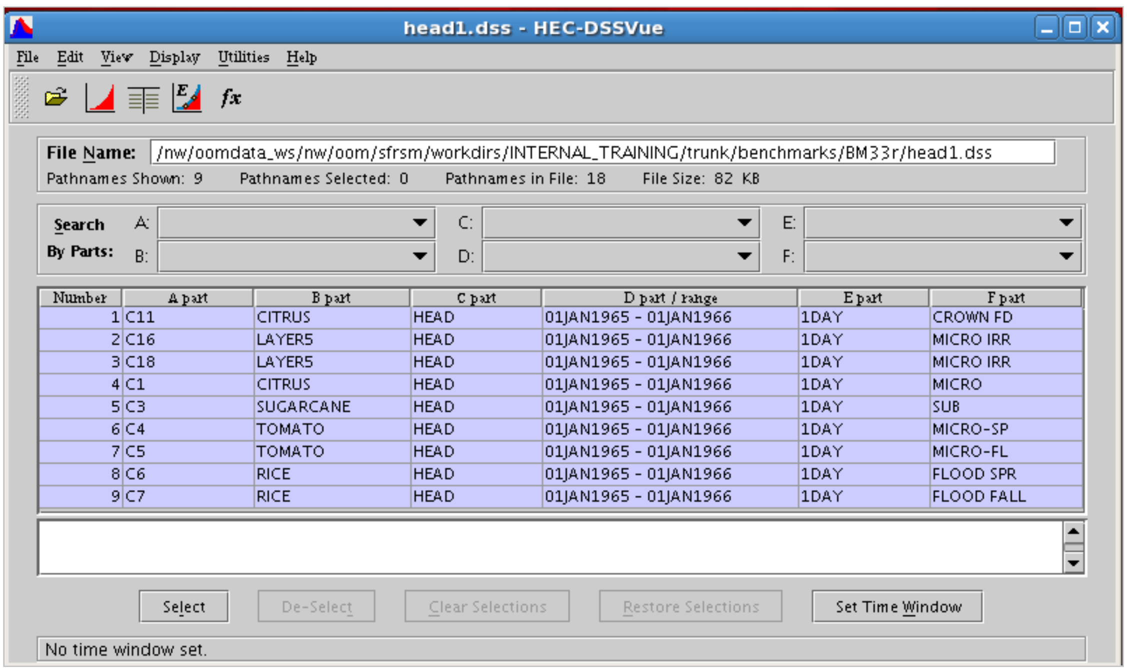
47. Locate output files.

Exercise 1.3.1 Review Time Series Data

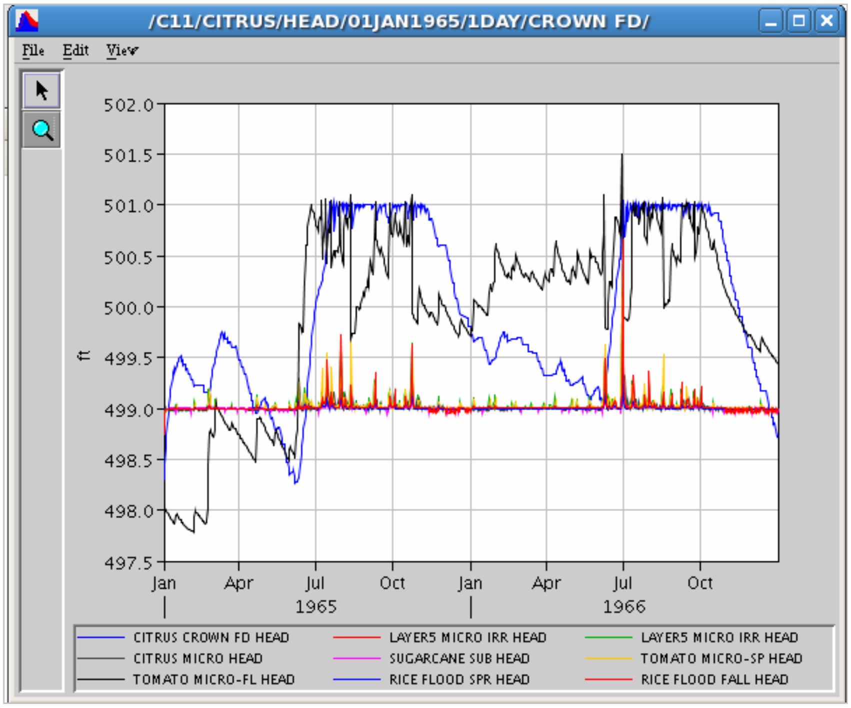
Time series data is stored in a compact binary format, the .DSS format, developed by the United States Army Corps of Engineers (USACE). This data will be viewed using **HEC-DSSVue**. This software was developed by the USACE to view the data.

To implement, select **HEC-DSSVue** from the RSM GUI toolbar.

48. Open **head1.dss** file. There is one cell head record for each cell in the benchmark.
    - Select ***all cells***.



49. Select parameter record and display graph by selecting **graph icon**.



- Why are the water levels different among the different HPM types?

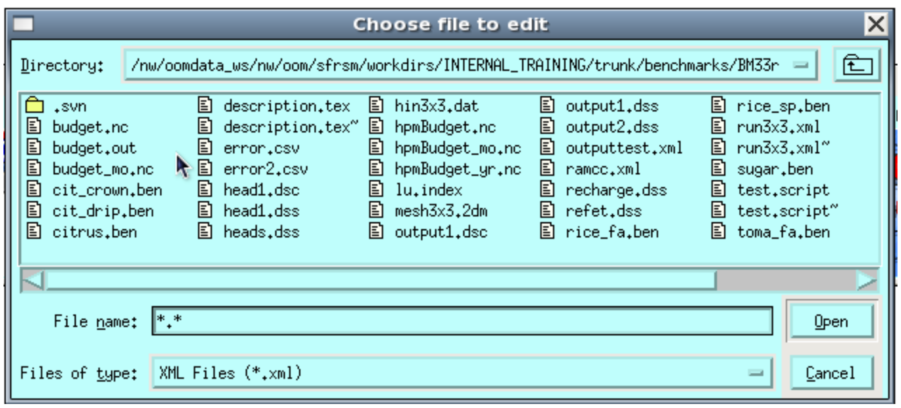
50. Open **output1.dss** file. There are three parameter records (runoff, water supply and water content) for the HPM assigned to each cell.
51. Select the group of parameters for Cell 5 and plot. Then select the group for Cell 1 and plot.
    - What are the differences between the plots?

Exercise 1.3.2 Review water budget data for waterbodies

Primitive Water Budget Data

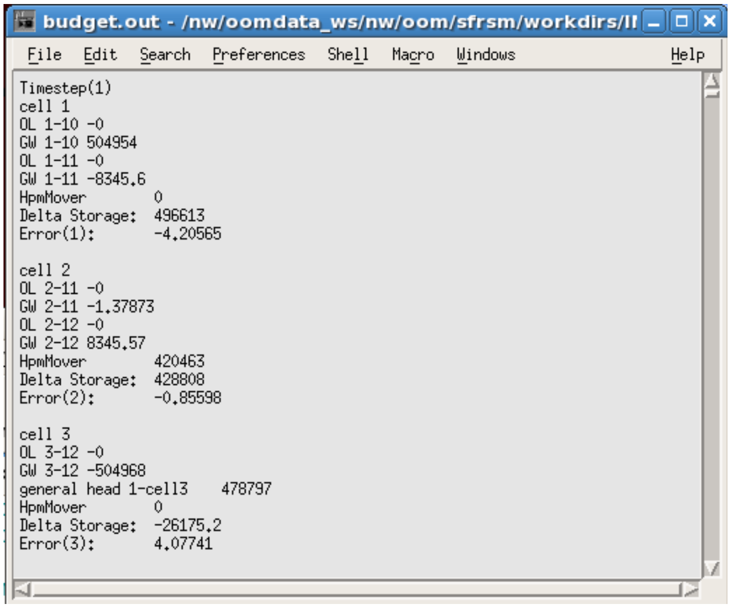
The primitive water budget data is stored in the **`budget.out`** file. This is an ASCII file that contains all of the watermovers and the values for each waterbody for each timestep. The budget package uses the internal RSM class names rather than the standard descriptions: **`darcy_circle`** is groundwater flow and **`mannings_circle`** is surface water (overland) flow.

52. List **`budget.out`** using the **"`edit XML`"** feature of the RSMGUI, found under the **`Pre-processing`** dropdown menu. In the input space for file name, enter **`*.*`** to view all files in the directory.
53. Select the file **`budget.out`**.



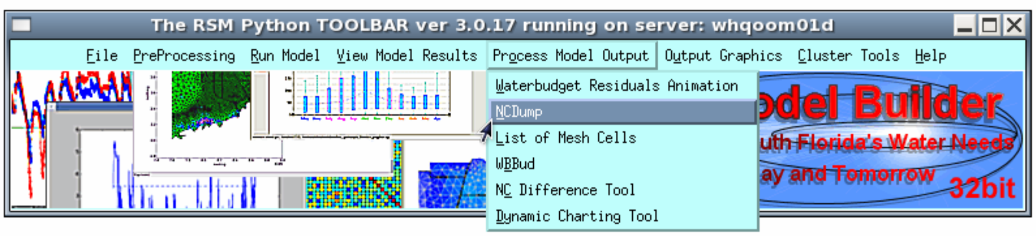
54. The **`budget.out`** file provides the water budget for each waterbody for each timestep. This package is very detailed, so it is used primarily for debugging when an RSM implementation is producing unusual results.

OL	Overland flow
GW	Groundwater flow
HpmMover	Water from the associated HPM
Delta Storage	Change in storage with the last time time-step
Error	Error in the water budget

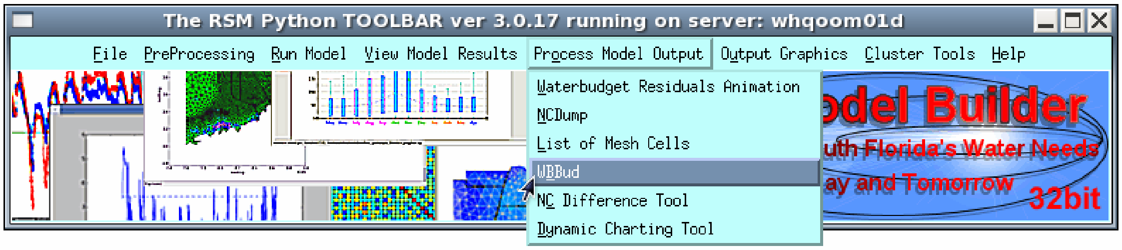


Water Budget Data

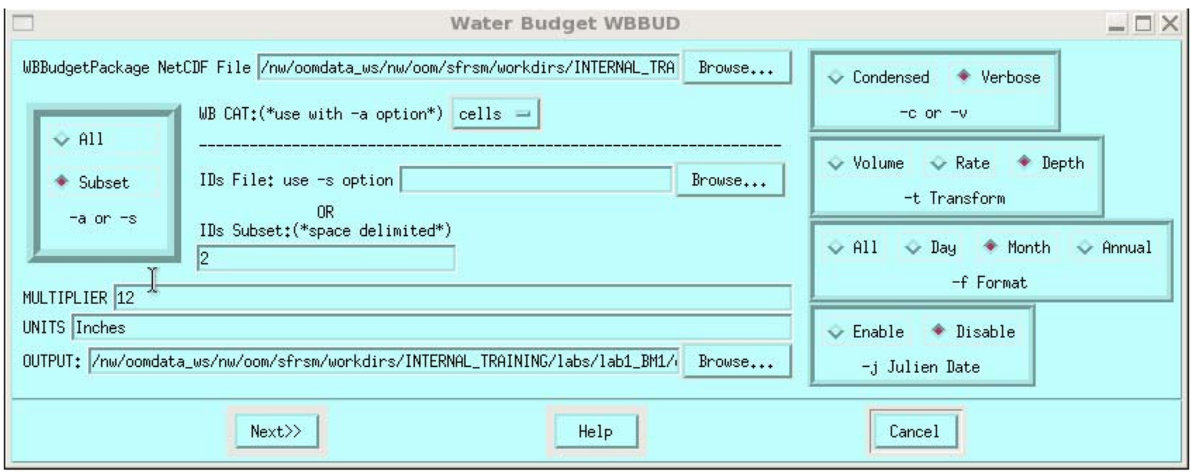
The data for constructing the water budgets is contained in the netCDF files; wbbudget.nc for waterbodies and hpmbudget.nc for HPMs. The NetCDF files are very large; they contain all of the output from the model. They can be viewed using the ncDump utility.



A water budget utility, **wbbud**, was developed to calculate water budgets for any waterbody or group of waterbodies.



55. Run the **wbbud** feature of the RSMGUI found on the **Process Model Output** dropdown menu.



Features in **wbbud** include:

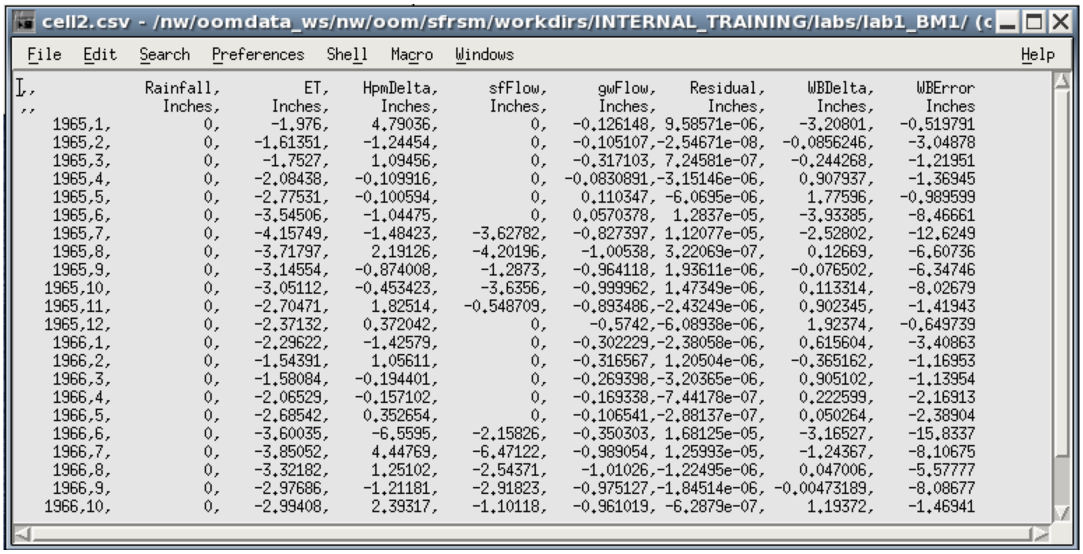
a	Report a summed report for all waterbodies of the specified type
s	Report a summed report for a subset (list) of waterbodies either provided in a file or entered by hand.
m	Multiply the output (i.e. –m 12 to convert inches to feet)
u	Units to be displayed in the header (required field if using the multiplier option)
v	Verbose expanded output
c	Condensed report
t	Transform output to volume, rate or Depth
f	Format the report to summarize all (raw) data, daily, monthly or annual
j	Julian date conversion

56. Enter these options to run **wbbud**:
    - Browse to the file: **[/../benchmarks/BM1/wbbudget.nc]**
    - Select the **Subset** option
    - Enter **2** in the IDs Subset input box
    - Enter a multiplier of **12** to convert feet to inches
    - Enter **Inches** as the label for your output units
    - Enter a path and file name in the output box to create an output file **[../labs/lab1_BM1/cell2.csv]**
    - Select the **Verbose** option
    - Select the **Depth transform** option
    - Select the **Month** format option
    - Disable the **Julian** option
    - Click **Next>>>**

The output will be written to the output file and a window will prompt the user to view the file that has been written.

57. When **wbbud** is complete an alert message will remind you where the output has been written.
58. Click the **View** button to view the output.

The typical results for BM33r for monthly **wbbud** output are as follows:




```
,          Rainfall,        ET,  HpmDelta,    sfFlow,    gwFlow,  Residual,   WBDelta,   WBError
,,           Inches,    Inches,    Inches,    Inches,    Inches,    Inches,    Inches,    Inches
   1965,1,        0,    -1.976,   4.79036,         0, -0.126148, 9.58571e-06,  -3.20801, -0.519791
   1965,2,        0,  -1.61351,  -1.24454,         0, -0.105107,-2.54671e-08, -0.0856246, -3.04878
   1965,3,        0,   -1.7527,   1.09456,         0, -0.317103, 7.24581e-07, -0.244268,  -1.21951
   1965,4,        0,  -2.08438, -0.109916,         0,-0.0830891,-3.15146e-06,  0.907937,  -1.36945
   1965,5,        0,  -2.77531, -0.100594,         0,  0.110347, -6.0695e-06,   1.77596, -0.989599
   1965,6,        0,  -3.54506,  -1.04475,         0, 0.0570378,  1.2837e-05,  -3.93385,  -8.46661
   1965,7,        0,  -4.15749,  -1.48423,  -3.62782, -0.827397, 1.12077e-05,  -2.52802,  -12.6249
   1965,8,        0,  -3.71797,   2.19126,  -4.20196,  -1.00538, 3.22069e-07,   0.12669,  -6.60736
   1965,9,        0,  -3.14554, -0.874008,   -1.2873, -0.964118, 1.93611e-06, -0.076502,  -6.34746
  1965,10,        0,  -3.05112, -0.453423,   -3.6356, -0.999962, 1.47349e-06,  0.113314,  -8.02679
  1965,11,        0,  -2.70471,   1.82514, -0.548709, -0.893486,-2.43249e-06,  0.902345,  -1.41943
  1965,12,        0,  -2.37132,  0.372042,         0,   -0.5742,-6.08938e-06,   1.92374, -0.649739
   1966,1,        0,  -2.29622,  -1.42579,         0, -0.302229,-2.38058e-06,  0.615604,  -3.40863
   1966,2,        0,  -1.54391,   1.05611,         0, -0.316567, 1.20504e-06, -0.365162,  -1.16953
   1966,3,        0,  -1.58084, -0.194401,         0, -0.269398,-3.20365e-06,  0.905102,  -1.13954
   1966,4,        0,  -2.06529, -0.157102,         0, -0.169338,-7.44178e-07,  0.222599,  -2.16913
   1966,5,        0,  -2.68542,  0.352654,         0, -0.106541,-2.88137e-07,  0.050264,  -2.38904
   1966,6,        0,  -3.60035,   -6.5595,  -2.15826, -0.350303, 1.68125e-05,  -3.16527,  -15.8337
   1966,7,        0,  -3.85052,   4.44769,  -6.47122, -0.989054, 1.25993e-05,  -1.24367,  -8.10675
   1966,8,        0,  -3.32182,   1.25102,  -2.54371,  -1.01026,-1.22495e-06,  0.047006,  -5.57777
   1966,9,        0,  -2.97686,  -1.21181,  -2.91823, -0.975127,-1.84514e-06, -0.00473189, -8.08677
  1966,10,        0,  -2.99408,   2.39317,  -1.10118, -0.961019, -6.2879e-07,   1.19372,  -1.46941
```

Exercise 1.3.3 Review water budget data for HPMs

The utility **hpmbud** was developed to calculate water budgets for an HPM or group of HPMs. The **hpmbud** utility is invoked at the command line as follows:

$RSM/trunk/hpmbud/hpmbud –n hpmbudget_mo.nc –s cell2 –d –m 12

Parameters for the hpmbud utility are shown in the table below:

Parameter	Description	Example
-n	(required) List the name of the NetCDF file	**-n hpmbud.nc**
-a	All hpms	**-a**
-s	Uses a subset of hpms provided in a file listing	**-s subset.file**
-d	Converts volume to depth (feet or meters, as defined in the control block in the run file run3x3.xml)	**-d**
-m	Divides each value in the budget by a specific value (a multiplier of 12 provides results in inches if the default unit for the model output is US Standard feet).	**-m 12**

The typical results for BM33r are shown in Figure 1.6 as follows:

```
                    Rainfall  Et CellDelta WSupply CU  Sewer Septic Runoff Seepage Storage ChgResidual
                      /12   FT/12  FT/12 FT/12      FT/12              FT/12  FT/12   FT/12     FT/12
1965  1  31 24  0  0.5197  1.976    0   0.80211    0     0     0    4.1363  0    -4.7904      0
1965  2  28 24  0  3.0488  1.6135   0   0.48126    0     0     0    0.672   0     1.2445    1.92e-06
1965  3  31 24  0  1.2195  1.7527   0   0.96253    0     0     0    1.5239  0    -1.0946    1.92e-06
1965  4  30 24  0  1.3695  2.0844   0   1.2834     0     0     0    0.45852 0     0.10992   1.92e-06
1965  5  31 24  0  0.9896  2.7753   0   2.2459     0     0     0    0.35959 0     0.10059   3.84e-06
1965  6  30 24  0  8.4666  3.5451   0   0.32084    0     0     0    4.1976  0     1.0448       0
1965  7  31 24  0  12.625  4.1575   0      0       0     0     0    6.9832  0     1.4842       0
1965  8  31 24  0  6.6074  3.718    0      0       0     0     0    5.0807  0    -2.1913       0
1965  9  30 24  0  6.3475  3.1455   0   0.48126    0     0     0    2.8092  0     0.87401      0
1965 10  31 24  0  8.0268  3.0511   0      0       0     0     0    4.5222  0     0.45342  -3.84e-06
1965 11  30 24  0  1.4194  2.7047   0   1.2834     0     0     0    1.8232  0    -1.8251    3.84e-06
1965 12  31 24  0  0.6497  2.3713   0   1.6042     0     0     0    0.25467 0    -0.37204      0
```

Figure 1.6. Typical results for Benchmark 33r.

Answers for Lab 1:

Exercise 1.1.1.

9. What are the differences? When a = 1.0, the head at the end of the run (t=2:30) is approximately 501.825. When a = 0.5, the head at the end is approximately 501.675, and the starting value is slightly smaller as well.

10. What are the changes? There was no change when the transmissivity was changed from 0.0 to 0.2

12. What are the results? The head now begins (t=0:15) at 503.83 and drops to 502.82 at t=2:30 compared to heads of 501.96 dropping to 501.68 previously.

Exercise 1.1.2.

1. In which cells and segments does the ponding occur? cells 5 and 14 and segments 1 and 4.

2. What is the model timestep and duration of the simulation? The time step is 15 minutes and the simulation runs for 2:30 hours.

4. Observe results. Overbank.dat file results:

1994 1 1 -20.131637

1994 1 1 -11.480757

1994 1 1 -7.144460

1994 1 1 -6.890670

1994 1 1 -7.242531

1994 1 1 -7.588950

1994 1 1 -7.865267

1994 1 1 -8.059018

1994 1 1 -8.177340

1994 1 1 -8.234370

For Segment 1, both plots are monotonically increasing functions starting at just above 501.0000 M. The final head drops very slightly when changing the bank seepage coefficient (approximately 501.0024 to 501.00235M).

Segment 4 has shows linear increases in head for both scenarios, starting at a stage of 501.006M. A similar magnitude slight drop in the final stage occurs in segment 4 (from approx. 501.0425 to 501.0420M).

7. How did the values change? Answered above

Exercise 1.2.1.

2. How do these heads compare to the heads from BM4? The head in Segment 20 has a slight jump from 501.75 to 501.81m at the very beginning and then decreases until it levels off at approximately 501.10m.

3. How did the heads change? The heads did not change when adjusting the bank height.

6. How did the heads change? The head for Segment 20 appears to have become unstable and oscillates drastically (from 500.5 to 504.5 m) at the start then stabilizes at about 501.6 at time t=4.5 hours. The cell heads behave similarly to the previous case, but cell 5 has some instability at the start.

Exercise 1.2.2.

7. How will you process the Cell 7 water budget?

```
budtool -n budget.nc -s 7 -d -m 12
```

By typing **`budtool –help`**, it lists the arguments for the command, **`"-s 7"`** will yield the water budget for cell 7.

9. How does this affect the water budgets? Higher Kveg values caused higher values for ET and a decrease in storage over the time period.

10. How does this affect the water budgets? Smaller rooting depth (xd) values lead to a smaller ET and increased storage.

How does this affect the cell heads (water table elevation)? The cell heads decrease but level off at a higher elevation. This corresponds to the increase in storage indicated by the wbbud tool.

Exercise 1.3.1.

Cell Monitors

What is monitored? And what is the output file?

The Cell Monitors are:

- Head for citrus (micro and Crown Fd), sugarcane, tomato(micro-sp, micro-fl), rice(flood spr, flood fall). All of these are output to head1.dss
- Recharge for citrus (micro irr) output to recharge.dss
- ET for citrus (micro irr) output to recharge.dss
- Rain for citrus (micro irr) output to output2.dss

hpmmonitors

What is monitored? And what is the output file?

- For citrus micro: water content, wsupply and runoff output to output1.dss and et output to output2.dss
- for citrus crown flood: watercontent, wsupply, and runoff output to output1.dss and et output to output2.dss
- For sugar cane subirrigation: watercontent, wsupply, runoff output to output1.dss and et to output2.dss
- For spring tomatoes: watercontent, wsupply, and runoff to output1.dss; recharge and rain to recharge.dss; et to output2.dss.
- For fall tomatoes: watercontent, wsupply, and runoff output to output1.dss and et output to output2.dss
- For spring rice – seepage irrigation: watercontent, wsupply, and runoff output to output1.dss and et output to output2.dss
- For fall rice – seepage irrigation: watercontent, wsupply, and runoff output to output1.dss and et output to output2.dss

2. Why are the water levels different among the different HPM types? The HPMs simulate the effects of irrigation, drainage, runoff and consumptive use. These parameters vary with respect to which crops occupies the area, thus the different modeled water levels.

4. What are the differences between the plots?

- The water content for citrus is variable but always above 0.8 while the water content for the tomatoes is 0.1 between fall crop seasons and varies from 0.2 to 0.3 during the season with spikes for rainfall events.

- There is no runoff from citrus while there is considerable runoff from tomatoes.

- There is irrigation of tomatoes of 0.06 inches per day and frequent irrigation events of 0.16 inches per day on citrus.

Index