Memorandum

To: Brian Macpherson (Colorado Water Conservation Board),
Carolyn Kemp (CWCB),
Mary Halstead (Colorado Division of Water Resources)
From: Steve Malers (Open Water Foundation, OWF)
Subject: Notes on MODSIM
Date: February 20, 2019

Contents

Introduction

This memorandum provides notes about MODSIM, including programming language, development environment, and development team, in order to provide perspective for whether StateMod should be converted to a different language. These notes are based on a conversation between Steve Malers of the Open Water Foundation and Enrique Triana of RTI on January 31, 2019, and follow-up review of an initial draft of this memorandum.

Background

MODSIM (see http://modsim.engr.colostate.edu/) is a generalized river basin decision support system and network flow model developed at Colorado State University. It has been applied in numerous basins around the world and it is used to simulate river basins throughout Colorado, including the Poudre River, Arkansas River, and is also used by Colorado Springs Utilities and Aurora Water, among others.

Technical Notes

The following are notes on specific technical topics.

Programming Language

MODSIM is coded primarily in Microsoft Visual Studio C#, with some Visual Basic .NET and compiled to the .NET framework. The original Fortran code was translated to C++ many years ago and the C++ code was converted to C# by Andre Dozier several years ago, to better integrate with Microsoft tools such as Microsoft Access for data management and out-of-the-box functionality for enhanced Graphical User Interface features.

Component Libraries Used

Important components include the following.

- TeeChart for graphing and other visualization (https://www.steema.com/), which requires developers to purchase a license
- Microsoft Access Engine (https://www.microsoft.com/en-us/download/details.aspx?id=54920), which is free to download

License and Open Source Considerations

The MODSIM license is displayed during the installation process and can be accessed from the Help / About MODSIM menu after installation. The license language is as shown below. MODSIM is not open source, but it is distributed with libraries (dlls) that expose the variables, objects and functions to allow preprocessing, postprocessing and run time customization.

Terms:
Colorado State University (CSU) is providing this software free of charge, by using the software you agree to accept all of the risk arising from its use. CSU does not warrant or guarantee that it will work or meet your needs and will not be responsible for any liability or damages arising from your use of the software.

Although MODSIM is license free and distributed at no cost, its source code is protected by copyright law and international treaties. Any unauthorized attempts to reverse engineer the source code or hide the identity of Colorado State University as the sole owner of the software in usage of MODSIM is prohibited.

Supported Operating Systems

The following is information about supported operating systems:

- MODSIM is supported on Microsoft Windows and requires the Microsoft .NET framework

- MODSIM has not been run on Linux using a .NET environment but this is theoretically possible with the correct software tools.

Development Environment

The development environment uses:

- Microsoft Visual Studio for .NET (C# and VB.NET)
- Version control
    - More than 20,000 lines of code.
    - A private Bitbucket repository is used for version control, with public-facing issues and Wiki (see https://bitbucket.org/engr_csu/csu_geodss/issues)

Software Testing

Software testing is implemented as follows:

- A battery of test networks, each testing individual features, is run representing model features. Results are compared to a previous run to ensure that results are expected. The comparison is done by custom code that compares the database output table by table and generates a report with differences (locations, times) to facilitate debugging. The expected results are saved in a repository.
- A few more comprehensive networks, with multiple features, are also included in the battery of test networks.
- Tests are run prior to public release of the software.
- There are currently no unit tests at the function/method level.
- There is currently no automated testing of user interface.

Software Performance Optimization

- User experience metrics like run times, loading times, and GUI response are monitored for new releases and improving (reducing times) is a goal in implementation of new features and updates.

Development Team

The development team consists of:

- John Labadie is the original author of MODSIM.
- Other developers have contributed over the years, including students and professionals that learned MODSIM while at CSU or through other training.
- Currently the software is supported by Enrique Triana (RTI International) through a cooperation agreement with CSU for MODSIM maintenance and development. RTI offers a commercial option for support and feature development
- Major enhancements to the software have occurred over time, primarily through funded research projects that have added major components.