US Army Corps
of Engineers®

Hydrologic Engineering Center

Hydrologic Modeling System HEC-HMS

Frequently Asked Questions

Version 3.0
April 2005

Introduction

The Hydrologic Modeling System software development team has been working for almost three years to convert the old program to the Java language. There are many new features in the interface and also many new simulation capabilities. Some users have contacted the development team with questions about the new features and capabilities. Some have expressed concern about the compatibility of existing project data with the new version, and the wisdom of starting new projects with a major new version coming soon. This document has been developed to answer the most common questions.

When will the new version of HEC-HMS be available?

Quality software requires careful development work and even more careful testing. We have been working very hard to make sure that Version 3.0 is truly ready for prime-time use before releasing it. The iterative process of testing and refinement is taking longer than originally planned but will yield a much more stable and robust program. We still do not have a firm release date, but hope to begin beta testing this spring. A call for beta testers will be issued through our email notification list. Beta testers will be selected on the basis of the type of engineering work they plan to do so that we can make sure all areas of the program are covered. Also, beta testers must be willing to provide detailed notes on the performance of the program during their use.

How does it differ from the current version? What is added and what is deleted?

The most notable feature of the new version is the user interface. All versions of the program from 1.0 up to the current 2.2.2 were developed with cross-platform tools that were state-of-the-art when the project began. The company that developed those tools is now out of business so we were forced to find a replacement and chose Java. Instead of just replacing the current program interface with a duplicate written in Java, we took this opportunity to critically examine how the program functions and improve it. The new interface provides much more control, better workflows, and superior graphics, while still being able to do everything the current program can do. In addition to default graphs similar to those included in the current program, graphs and time-series tables can be created that include any result computed by any element in any simulation run. It allows elements to be copied through the clipboard between basin models. Basin models now allow a variety of background map formats. Meteorologic models are automatically synchronized with basin models as subbasins are created, deleted, or renamed. Simulation runs are automatically aware of any changes to the basin model, meteorologic model, and control specifications. Individual basin and meteorologic models are automatically aware of any changes in gage data and other data external to the models. There is also a new grid data manager similar to the gage manager in the current version. The existing paired data manager for user-specified hydrographs and s-graphs has been expanded to all types of paired data. This means, for example, that you will be able to retrieve storage-outflow curves from an external DSS file where they might have been written by another program such as HEC-RAS.

Does the new version include the depth-area option?

The new version will include what we call the Depth-Area Analysis tool. It uses a simulation run configured with a meteorologic model set to use the frequency storm. Using the tool, the user will select several elements in the basin model where results are desired. The tool automatically computes adjusted storm areas for each of the selected elements and produces a table of peak flows and total volumes. Results contained in the table are equivalent to creating a separate simulation run for each desired element where the frequency storm area is equal to the contributing area at the element. This is the first tool in a suite planned to address frequency curve development, uncertainty estimation, and statistical summary generation.

Are there other new technical features?

The new version will also have a snowmelt capability. It uses a very sophisticated temperature index algorithm that works with gridded subbasins (using ModClark) and also the traditional elevation band approach. It differentiates between melt regimes under dry conditions and wet conditions when it is raining or snowing. It includes the concept of cold content to account for rain that falls on a cold pack and freezes rather than going immediately to surface runoff. It also accounts for liquid saturation and groundmelt. There is a new Priestley-Taylor evapotranspiration method for both gridded subbasins and standard subbasins. There is a new exponential loss method and a new gridded version of the deficit constant loss method. Also, the deficit constant loss method (gridded or regular form) no longer contains moisture recovery parameters but instead uses potential evapotranspiration computed by the meteorologic model for more flexibility. Finally, when the reservoir element uses the elevation-area or elevation-storage options, it is now possible to have up to 10 outlets, 10 overflows, and 10 spillways.

Should I invest in training now, knowing that "current training" will focus on the current version, with the new version just around the corner?

There is no reason to delay training if you are going to enroll in a quality training class. Quality classes don't just teach you how to use the program, they also teach you how to do hydrology and water resources engineering. The underlying mathematical algorithms the program uses have not changed so all the engineering skills you develop now are completely compatible with the new version. The technical details you learn about how Muskingum-Cunge works and when to use it as compared to other reach routing methods applies to the new version the same as the current version. Everything you learn now about how to analyze soil data, choose a loss method, and estimate parameters will transfer 100% to the new version. Techniques you learn now for calibrating and verifying a watershed model will be equally valuable with the new version. The small number of users who have tested the new version have had no problem transferring their existing knowledge and skills. None of them have reported any difficulties and most seem to be very comfortable with the new version after only a few hours usage.

I'm about to start a new watershed analysis. Should I start with the current version or wait on the new version?

The new version is fully compatible with existing projects. The first time you open a project with the new version, the data will be automatically converted where necessary. Even though we have not lost a single project in the last 16 months of testing, it is always a good idea to keep backup copies. Once you start using a project with the new version it will not be possible to take it back to the current version. The only caveat is that gridded soil moisture accounting and gridded SCS curve number loss methods have changed how they store gridded parameter data. If you are using either of those two methods there will be some manual work external to the program that will have to be done before you can start using the new version. There really is no reason to delay starting a new project.

Is the new version now an ArcView/ArcMap add in? More tightly integrated with GIS tools?

GIS software is designed to store and analyze maps; it is not designed to perform engineering calculations of the type necessary to simulate the hydrologic cycle. Don't forget that computing a simulation in the program is actually a process of solving mathematical equations, many of them differential equations. It is very difficult to fit differential equation solvers into the framework used to build add-in modules for GIS software. Solving mathematical equations for hydrology requires a wealth of parameters, initial conditions, and boundary conditions that often are not part of the GIS system. Boundary conditions are usually drawn from meteorologic stations recording time-series of precipitation, temperature, and solar radiation. Initial conditions typically come from knowing the state of the watershed at the beginning of the simulation, drawn from field visits or engineering judgment. Much of our parameter data comes directly from past engineering reports. For example, the parameters used to describe a reservoir can be obtained by examining the design drawings for the dam. However, GIS software is extremely useful for delineating subbasin boundaries from elevation data and in processing spatial soil data to estimate parameters for loss methods. We are continuing to improve the HEC-GeoHMS tool; it can delineate subbasins and configure basin models. In the future it will provide more capability for estimating parameters from land use and soil maps. HEC-HMS will continue to be a stand-alone engineering application. Users with access to GIS software can use the HEC-GeoHMS tool (which does tightly integrate with ArcView) to utilize GIS data when it is available.

Are all the bugs fixed?

It is nearly impossible to remove all the bugs from a complex piece of software such as HEC-HMS. However, our efforts in several areas will result in higher quality computer code with fewer bugs. First, the conversion from the old cross-platform system to Java required a programmer to check every line of source code. Many errors and omissions were found and corrected during the conversion process. Second, object-oriented design techniques are much more advanced now than when the HEC-HMS project was started in the early 1990s. We have used the new techniques and emerging best-practices to produce the best source code possible. Third, we are using a much more comprehensive testing program. Testing includes a set of benchmarks where program simulation results are compared to hand-calculations that have been

independently verified. Knowledge of the underlying mathematical algorithms was used to create benchmarks designed to rigorously test all computation features of the program. The benchmarks will eventually be published as part of the program documentation. Fourth, we are spending much more time using the program on our own project work. This way we know how it performs in a typical work environment like our users experience. While it may not be possible to remove every single bug, we expect Version 3.0 to be very robust and stable.