DHI TECHNOLOGY

THE MIKE CUSTOMISED PLATFORM

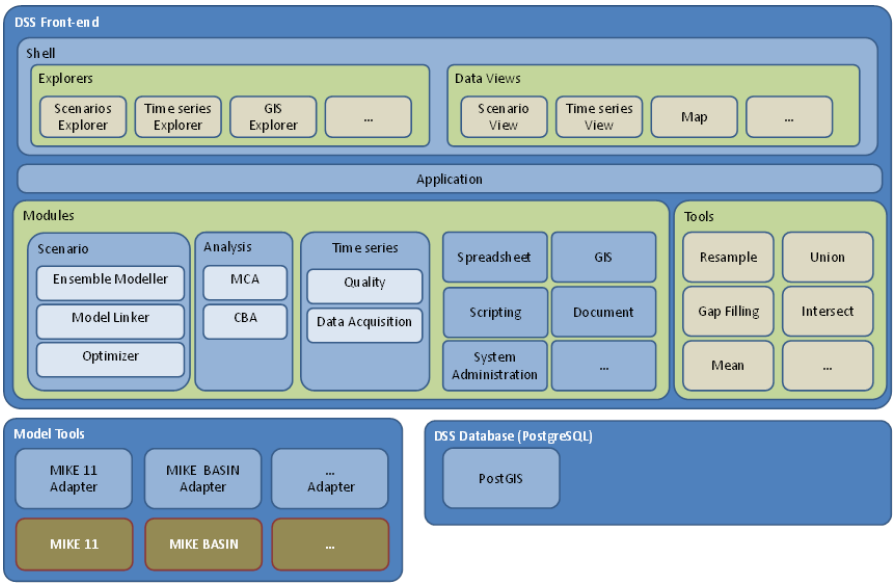
Software Architecture

THE MIKE CUSTOMISED PLATFORM

Today integrated water information solutions are increasingly being applied as backbone systems in various fields such as:

- Managing water related resources
- Flood and drought planning
- Real-time structure operations
- Hazard and emergency planning
- Mitigation management

Such solutions are usually driven by client-specific requirements and by having to integrate with existing applications and infrastructure components (water allocation, asset management, SCADA systems and so on). Software vendors cannot meet such demands through a traditional 'one-size-fits-all' product offering.



Therefore, what is required is a flexible solution development framework which implementation projects can leverage on when establishing solutions. Such a framework shall:

SUMMARY

THE PLATFORM

Water information solutions of today need to:

- Use off-the-shelf components for standard functionality
- Seamlessly integrate existing infrastructure with tailor-made components and off-the-shelf components
- Be able to enhance a delivered solution with new modules over time

The Platform provides solution developers with a flexible and extendible framework for meeting clients' specific water information management needs.

CHALLENGE

- Inadequate or inaccurate design and operational data available
- Cost-ineffective solution and uncertain safety levels

KEY FEATURES

- Extendible with respect to user interface as well as business logic
- Programmatic access to all business functionalities in the standard module
- Host applications are highly customisable with respect to user interface navigation, look and feel
- Interfaces with a growing number of commercial and open source modelling systems
- Supported by service and maintenance

© DHI / Photo: Shutterstock © saicle

- be extendable and highly customisable
- come with standard modules for standard functionality
- be high quality
- include detailed documentation

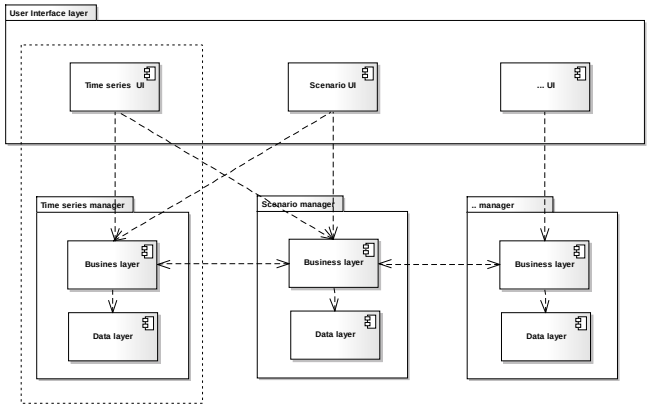
THE OFFERING - MIKE CUSTOMISED PLATFORM

The Platform consists of components, plugins and host applications. Included are also a set of rules regulating how functionality-providing plugins can be added to a hosting application and how these individual plugins can interact. These rules are named protocols.

Where:

- **Components** include a pool of standard user interface controls and infrastructure components. These can help with:
  - Reading and writing data from a database
  - Handling user login and data access rights
  - Storing configuration settings
- **Plugins** are functionality components providing the features requested by the clients. The Platform includes a large number of plugins. Solution-implementing projects can add to this pool by developing components adhering to the Platform protocols.
- **Application hosts** are applications that leverage on the standard user interface and infrastructure components for hosting the plugins and having them seamlessly interacting.

MIKE CUSTOMISED comes with two such hosts – a Windows desktop host and a Web host. Projects that have very unique needs can, however, easily construct their own specific hosts.



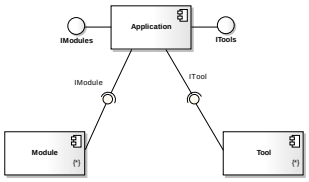
ARCHITECTURE

The framework protocol enforces a horizontal layered architecture. There are layers for data access, business functionality and user interactions. Also, there are vertical layers separating the functionality in distinct domains called managers. This allows for:

- scripting of all the available business functionality
- having the functionality batch executed and also on remote computers
- making it possible for one plugin to access the functionality of other plugins

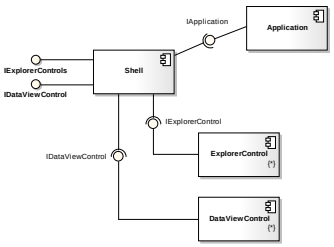
FUNCTIONALITY

The functionality exposed by the Platform is typically provided through module or tools components.



Modules implement functionality related to a well-defined category of data. For example, the 'time series manager' module handles functionality related to time series. On the other hand, 'the scenario manager' module handles functionality related to scenarios and simulations. The business logic of a module is exposed via an API. Activation of modules follows the Platform's plugin loading protocol by implementing the IModule interface. This implies that organisations can construct their own modules. These can be made accessible with the Platform.

Tools are 'cross cutting' components that are not associated with a particular module, but can be used by any module. Furthermore, tools are normally lightweight, with more limited and specific functionality than the modules. They are activated following the Platform's plugin loading protocol. This implies that organisations can add to the provided pool of standard tools.



Any application that wants to make use of modules and tools can do so by simply instantiating the Platform's Application object which will enumerate the available modules and tools and make them available. Such an application is called a host.

The two types of Application hosts with the Platform are:

- an advanced desktop application
- an equally advanced web application

These hosts not only make use of the Application, Modules and Tools plugins but also all the other user interface plugins. Solution implementing projects can add their own developed plugins.

Contact: MIKE CUSTOMISED—mikecustomised@dhigroup.com
For more information visit: www.dhigroup.com

© DHI