Mediaflux® Applications for Web (MAW)

Assemble completely new interfaces in days, rather than months or years

Use existing components or simply build your own in Javascript

Easily create generic or role specific applications, ensuring users are only seeing the information they need

Applications are defined in a single document allowing for easy modification

Expedite tasks by reducing complexity

MAW Features

- Declarative UX Framework
- Loosely Coupled Component Architecture
- Designed for Customer Extensibility

Rapidly develop interfaces for general or role specific tasks.

There's an optimal interface for every need. Through years of innovation people have designed clean, intuitive interfaces that make interacting with computational programs simpler and easier. What these interfaces create is user efficiency while lowering the barrier to entry simply by prioritising the user experience and reducing brute complexity.

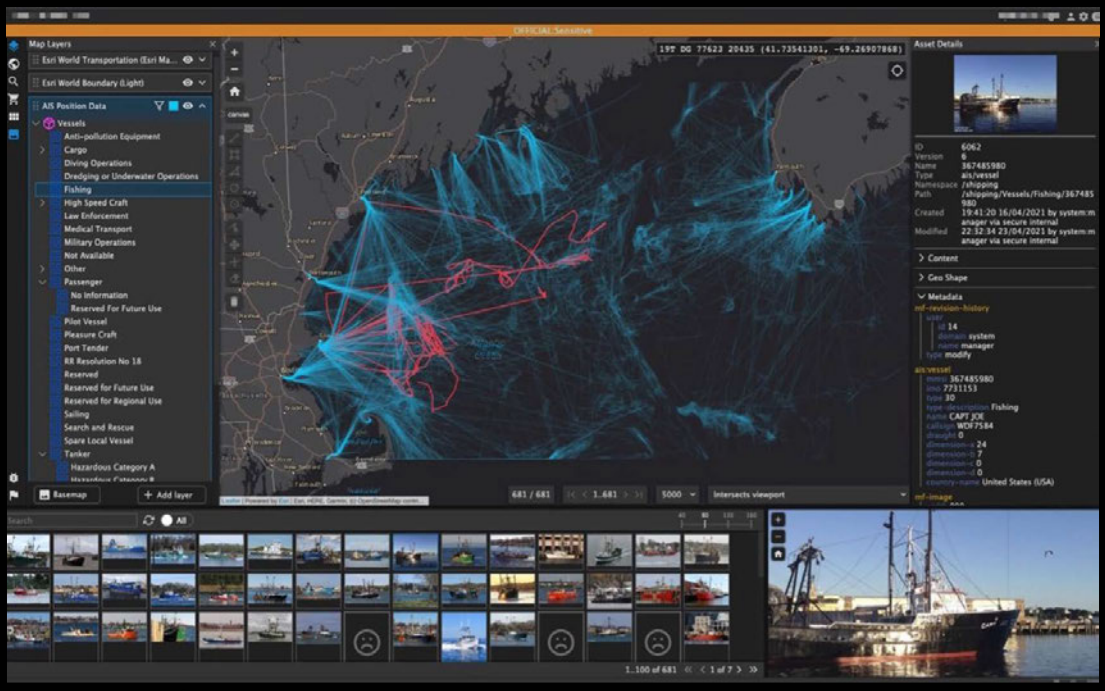
Data management is no different. By working with customers around the globe across a slew of diverse problems we arrived at the conclusion that a new interface technology was required, one where we could create industry, organisational, and role specific applications for our users.

The Mediaflux Applications for Web (MAW) framework makes the rapid development of custom interfaces easy and intuitive. Enterprises are free to plug-and-play components that address common pain points, or build bespoke components that expedite specific tasks. All this is achieved by simply adding a new JavaScript to a single document which contains your current list of built and deployed components.

What is MAW?

MAW is a component-based framework for building web applications to control and monitor a Mediaflux server. MAW applications are built from loosely-coupled, interactive components which allow users to retrieve and display information, change the behaviour of application, and/or execute services on the underlying Mediaflux server. The MAW framework can be used to generate applications ranging from simple dashboards that show the status of data holdings to complete analysis and management systems capable of implementing complex workflows. All of which can be presented simultaneously to provide different views of the same data depending on the needs and permissions of the user.

www.arcitecta.com | talk@arcitecta.com



Maritime Time-Series Application

Simple, Flexible, Extensible.

A MAW application is constructed from a single XML definition that describes the components, their layout, and how they will interact. These definitions can be downloaded by anyone with appropriate permissions, modified, and reuploaded to quickly create custom interfaces simply by tweaking the XML and without any additional development.

Applications, components, and modules are version controlled for simple installation and management. If the existing functionality is insufficient, custom components and modules can be created, adding new and unique capabilities to the system that can be subsequently used in future applications. All of this can be performed by the customer independently or in consultation with Arcitecta.

Components

Components are the fundamental building block of MAW and are implemented in JavaScript with each component generating its own interface. Components operate within the interface offering configuration properties to customise behaviour. They expose data objects and receive or emit state events that can be connected to form interactivity.

Common components include:

- Layout and structural options
- Collection hierarchy navigation
- Tables for service & query results
- Tiles and lists for assets
- Asset and object details
- File and AWS S3 browsers
- Extensive geospatial support
- Charting and visualisation
- Dialogs, forms and other interactions

Modules

Modules are non-visual and provide application wide functionality. This includes support for advanced drag & drop, object specific context menus, advanced state & object selection management, data sources and application tutorials. These capabilities provide the ability to composite disparate components, both generic and custom, into a seamless interface.