

# Functional improvements to Mediaflux Portals

This document lists the functional goals of the Mediaflux portal focussed portion of the RDS A1.7 project.

## High level descriptions

Features are prioritised according to their estimated value in driving adoption of the sharing of research data already stored in Mediaflux, and the adoption of Mediaflux as a data sharing tool.

### ***Predictable page structure: high priority***

Rewrite the Mediaflux portal plugin to generate a clean, predictable web page structure that is:

- addressable by portal customisation tools constructed with conventional web technologies, CSS and JavaScript;
- compliant with best practices in web page construction;
- Redresses where possible existing accessibility issues with portal pages.

This feature is both an enabling step for features listed below, and a direct improvement for portal users that employ assistive technologies such as screen readers.

### ***Integration of CSS supplied by the Mediaflux node operator: high priority***

Allow portal configuration to specify one or more Cascading Style Sheet documents, and include those documents in web pages produced by the Mediaflux portal plugin so that the CSS rules defined apply to the content of the portal web pages.

This feature is intended to allow Mediaflux operators to use existing web technology to customise portals, allowing Mediaflux installations to leverage existing knowledge, and to customise portals in ways not supported by the existing portal interface.

### ***Integration of JavaScript supplied by the Mediaflux node operator: medium priority***

Allow portal configuration to specify one or more JavaScript documents, and include those documents in web pages produced by the Mediaflux portal plugin, so that the JavaScript

behaviours defined apply to the content of the portal web pages. In addition, emit "event triggers" for page rendering events to allow JavaScript behaviours to occur.

This feature is intended to allow Mediaflux operators to use existing web technology to customise and extend portals, allowing Mediaflux installations to leverage existing knowledge and engage with existing software such as web maps, to extend portals in ways not supported by the existing portal interface.

***Replacement of formatting and layout instructions in default/example portal configuration files with equivalent expression in CSS: low priority***

The sample configuration file for Mediaflux portals includes a large amount of information about colour, size, and other formatting information. A sample CSS file should be supplied along with the sample configuration file, and the formatting information moved into that CSS file.

This feature is intended to give Mediaflux operators a strong correct example of the best way to control the presentation of a portal.

## Technical specifications

QCIF/JCU and Architecta have collaboratively produced a technical specification for the features listed; that specification document is available at <http://danielbaird.com/mf-portal-modern/MFPortalsComponentsSpec.pdf> .