

Preferences Framework

DRAFT

Overview

The Infusion Preferences Framework offers a reusable set of schemas, programming APIs, and UI building blocks specific to the creation, persistence, and integration of preference editors into a variety of web-based applications, content management systems, and delivery environments. It does not prescribe a single means by which content should be delivered to users. Instead, the Preferences Framework architecture provides an event-driven API that enables different, pluggable personalization strategies to listen for changes in a user's preferences and respond accordingly.

The Preferences Framework allows for the creation of a customized [Preferences Editor](#) by supplying a [Primary Schema](#) and [Auxiliary Schema](#). The [Primary Schema](#) provides a description of the preferences that the editor will manage. It includes both the default values, as well as, the set of values that the preference can take. The [Auxiliary Schema](#) provides the instructions for combining the panels (for setting preferences) and enactors (for applying preferences) that will operate on these preferences.

More Information

- [Preferences Editor](#)
- [Builder](#)
 - [Primary Schema for Preferences Framework](#)
 - [Auxiliary Schema for Preferences Framework](#)
- [Using the Preferences Framework with User-Defined Grades](#)
- [Enactors](#)
- [Panels](#)
 - [Composite Panels](#)
 - [Conditional Subpanels](#)
- [Connecting the Parts of a Preferences Editor](#)
- [Localization in the Preferences Framework](#)

Tutorials

- [Tutorial - Creating a Preferences Editor Using the Preferences Framework](#)
 - [Creating a Primary Schema](#)
 - [Creating Enactors](#)
 - [Creating Panels](#)
 - [Creating and Auxiliary Schema](#)
 - [Instantiating the Preferences Editor](#)
 - [Instantiating the Enhancer and Settings Store Only](#)
 - [Styling the Preferences Editor](#)
- [Tutorial - Full Page Preferences Editor](#)
- [Tutorial - Full Page Preferences Editor \(with Preview\)](#)
- [Tutorial - Creating a New Adjuster UI](#)