EPiSERVER

EPiServer and XForms - The Next Generation of Web Forms

How EPiServer's forms technology allows Web site editors to easily create forms, and developers to customize form behavior and appearance.

BACKGROUND

Forms are an everyday occurrence on the Web and you probably use more forms than you realize; a great deal of online interaction takes place with the help of forms technology. It was partly due to this that the World Wide Web Consortium (W3C) sponsored the development of XForms. Instead of further enhancing the existing HTML Forms technology, W3C thought it necessary to handle forms with a different approach, an approach that resulted in XForms 1.0.

EPiServer's forms management is based on XForms, making it easy for editors to create forms and EPiServer developers to customize the behavior and appearance of forms. But, what exactly are XForms?

WHAT ARE XFORMS?

One of the major goals when designing XForms was to separate *data* from *presentation*; something that differs from traditional HTML Web forms. XForms are comprised of separate sections that describe what the form does and how it looks.

The three key elements of XForms are:

- Presentation
  The look and feel of the form, i.e. how it is displayed on the screen.
- Structure
  The layout of the form, i.e. how its controls are organized.
- Data
  The data that is captured by the form.

> "XForms is the W3C's name for a specification of Web forms that can be used with a wide variety of platforms including desktop computers, handhelds, information appliances, and even paper." *(Source: W3C)*

XFORMS ARCHITECTURE

The XForms Specification is only interested in the form's *data*, known as the XForms Model, and the *structure*, known within XForms as the User Interface. The presentation is handled separately by the container or hosting document. XForms does not decide a document format; this is done by the host document. This separation is what allows XForms to be deployed onto multiple devices.
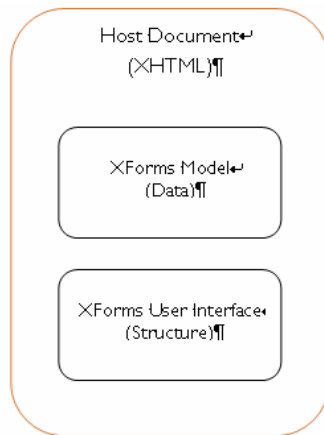
XForms Model

The XForms Model is used to define the data model used by the form and how that data model is constrained and behaves under certain conditions as the XForms User Interface captures data. An important concept in XForms is that forms collect data, which is expressed as XML instance data.

The diagram below shows how the XForms Model has the capability to work with a variety of standard or proprietary user interfaces.

XForms User Interface

The XForms User Interface is used to define the visual structure of the form. (The visual structure is not to be confused with the presentation in terms of look and feel.) For example, an

XForms User Interface could describe an address as a group of fields with several address lines, a field for postal code and a select field for country. The background colors, fonts for labels and the types of controls used, e.g. drop-down list for the country, are part of the host document's presentation.

BENEFITS OF XFORMS

The XForms recommendation may make it look complicated, but the recommendation is addressed to developers who need to know exact details on how to implement XForms. The same form written in XForms and HTML Forms looks pretty much the same. The benefits of XForms become more apparent when you start to design forms that HTML wasn't designed to handle.

XForms technology improves authoring, reuse, internationalization, accessibility, usability, and device independence. Here is a summary of the primary benefits of using XForms, according to W3C:

Enhanced accessibility

XForms separates content and presentation. User interface controls encapsulate all relevant metadata such as labels, thereby enhancing accessibility of the application when using different modalities. XForms user interface controls are generic and suited for device-independence.

Multiple device support

The high-level nature of the user interface controls and the consequent intent-based authoring of the user interface make it possible to re-target the user interaction to different devices.

Existing schema re-use

This obviates duplication, and ensures that updating the validation rules as a result of a change in the underlying business logic does not require re-authoring validation constraints within the XForms application.

Less use of scripting

By defining XML-based declarative event handlers that cover common use cases, the majority of XForms documents can be statically analyzed, reducing the need for imperative scripts for event handlers.

Strong typing

Submitted data is strongly typed and can be checked using off-the-shelf tools. This speeds up form filling since it reduces the need for round trips to the server for validation.

Internationalization

Using XML 1.0 for instance data ensures that the submitted data is internationalization ready.

XML submission

This obviates the need for custom server-side logic to marshal the submitted data to the application back-end. The received XML instance document can be directly validated and processed by the application back-end.
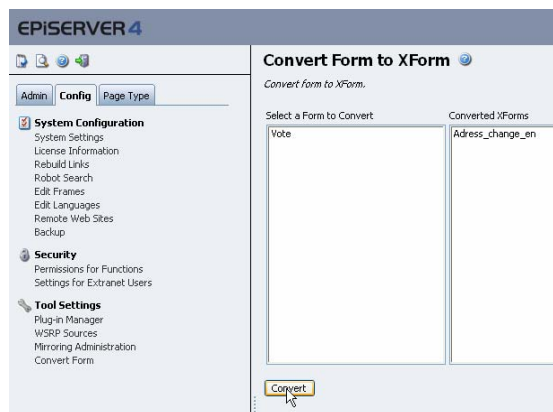
External schema augmentation

This enables the XForms author to go beyond the basic set of constraints available from the back-end. Providing such additional constraints as part of the XForms Model enhances the overall usability of the resulting Web application.

XFORMS IN EPISERVER

XForms technology is available in EPiServer from version 4.50. This does not, however, mean that any forms created in EPiServer versions previous to 4.50 miss out on anything. Any forms that have been created in earlier versions of EPiServer can be easily converted to XForms with the built-in form converter.

> XForms can do everything that the old HTML Forms can and EPiServer developers and editors benefit from the changes.

BENEFITS FOR EPISERVER EDITORS

XForms can do everything that the old HTML Forms can—and more—and both EPiServer developers and editors benefit from the changes. Some of the benefits for editors compared with HTML Forms are:

- Labels are linked to the input for greater accessibility.
- The possibility to add your own Submit button(s).
- Input validators.
- ToolTips for input fields.
- Required validators for all field types.
- Load and re-edit old posts.
- Save and restore values to and from a file.
- Get the initial data for a form from an external document.
- Calculate submitted values from other values.
- Constrain values in certain ways, such as requiring them to be in a certain range.

BENEFITS FOR EPISERVER DEVELOPERS

EPiServer developers do not need to learn how XForms work as the form will be parsed to server controls. The developer can modify the Web server controls in the form before it renders to the client. When the form is posted, you can validate the posted data before continuing with the post. It is also possible to alter and create forms and form data through the API.

As the form and form data is saved as XForms, it opens up the possibility to present the form both as an HTML form, as well as an XForms form, without having to change anything in the form. The posted data for a single form might be the result of both an XForms on an intranet and a standard form on a public Web site.

EPiServer developers benefit from XForms on the following points:

- Easier to handle forms on different pages.
- Possible to change the appearance and properties for the form.
- Possible to validate and change posted data.
- Easier to handle statistics and present posted data.
- Possible to show forms as either HTML Forms or XForms.
- Add your own input validators.

FURTHER REFERENCE

Read further information about XForms:

- http://www.w3.org/MarkUp/Forms/
- EPiServer technical documentation

EPiServer AB

EPiServer AB is a privately owned Swedish product company, founded in 1994, and is the leading company in Content Management and portal solutions through the platform EPiServer. The company is a Microsoft Gold Certified Partner and has held AAA ranking by Dun & Bradstreet since 2000.