

Introducing cf_sebForm 1.0 Beta

Posted At : November 16, 2007 7:00 AM | Posted By : Steve

Related Categories: sebtags

The cf_sebForm tag set creates accessible forms that provide both client and server side validation. They can interact with CFCs without breaking encapsulation. The forms are skinnable and completely customizable. It also facilitates using the same code as both an add and an edit form.

Here is a simple example of using cf_sebForm:

```
<cf_sebForm>
  <cf_sebField fieldname="fName" label="First Name">
  <cf_sebField fieldname="lName" label="Last Name">
  <cf_sebField fieldname="bdate" label="Birth Date">
  <cf_sebField type="submit" label="Save">
</cf_sebForm>
```

This is a simple form that doesn't (as yet) perform any function. It has three optional text fields with no limit on length.

If I want to limit the allowable length on a field, I would add a "Length" attribute. For example, to limit the fname field to 40 characters:

```
<cf_sebField fieldname="fName" label="First Name" Length="40">
```

If I want to make a field required, I add a required attribute with a value of true. This will add both a clients-side check with JavaScript as well as server-side validation. To make the last name required, for example:

```
<cf_sebField fieldname="lName" label="Last Name" required="true">
```

I can also change the field type by adding a type attribute. For example, I can use a "date" type for bdate:

```
<cf_sebField fieldname="bdate" label="Birth Date" type="date">
```

This will indicate that the field is a date field and add date validation for the field. Types of "date2" and "xdate" are also available (A link to a full list of is at the end of this entry).

The form can automatically submit to a CFC method, passing each form field in as an argument to the method. For example, to call the "callMethod" method of the "Application.MyComponent" component upon submit, use the following code:

```
<cf_sebForm
  CFC_Component="#Application.MyComponent#"
  CFC_Method="callMethod"
>
```

This is especially useful when in conjunction with a [PageController](#) or similar concept.

This is a very rough introduction to cf_sebForm. I am hoping to expand on this over the next several entries (which I will relate to this one). Although they aren't written yet, here are some of the topics I am planning for the near future.

- Adding and editing records with cf_sebForm
- Styling and skinning cf_sebForm
- Basic sebForm field types
- File handling with sebForm

- Custom field types with cf_sebForm

If you would like to see me cover one of the above topics next (or if you have a question or topic idea), let me know.

Documentation:

- [cf_sebForm](#)
- [cf_sebField](#) (including list of types)
- [cf_sebSubField](#)

Examples

- [Demonstration Site](#) (uses sebtags in conjunction with DataMgr)
- [Old Examples](#) (very old examples of using cf_sebForm)

[Download from RIAForge](#)

The cf_sebForm custom tags are part of the [sebtags custom tag set](#) which is open source and free for any use.