

Rails GUI Development with Ext JS

martin.rehfeld@glnetworks.de
@ RUG-B 10-Jan-2008



What's wrong with ActionView::Helpers ?

- Nothing!
- But...
 - often feels like assembling a spacecraft molecule by molecule
 - which is fine for building the *one* GUI that no one else has, but get's boring for standard apps

Wouldn't it be nice, if...

- ...you had rich GUI components like
 - Data Grids
 - Tree Views
 - Tabs
 - Toolbars / Menus

Maybe Ext JS is for you

 **Ext JS** a foundation you can build on

HOME DOWNLOAD LICENSE LEARN FORUM BLOG STORE

Download Ext 2.0

Combination Samples

Grids and Trees

Windows

Tabs and Layouts

Forms & ComboBox

More Samples

Company	Price
3m Co	\$71.72
Alcoa Inc	\$29.01
Altria Group Inc	\$85.81
American Express Compan	\$52.55
American International Gro	\$84.13
AT&T Inc.	\$31.61

Basic Array Grid

Light	Price	Avi
Shade	\$9.58	Apr
Shade	\$0.06	Dec
Mostly Sunny	\$4.69	May
Sun or Shade	\$7.16	Apr
Mostly Sunny	\$9.00	Jun
Sunny	\$2.44	Mar

Editable Grid

Author	Title
Sidney Sheldon	Master of the G
Sidney Sheldon	Are You Afraid
Sidney Sheldon	If Tomorrow Co
Sidney Sheldon	Tell Me Your Dr
Sidney Sheldon	Bloodline
Sidney Sheldon	The Other Side

XML Grid

Company
Industry: Automotive (1 Item)
General Motors Corporation
Industry: Computer (5 Items)
Hewlett-Packard Co.
Intel Corporation

Grouping

\$100.00	\$600.00
\$100.00	\$600.00
\$100.00	\$400.00
\$100.00	\$200.00
100	\$600.00
\$100.00	\$2,400.00

Live Group Summary

I want to define some a tooltip on a Ext.f the control or on the label. Thanks Marc

Add an option for the Form to sub Feature Requests Forum

We have a JAVA web framework built usecases where the user is not allowed teh Java class the value should ...

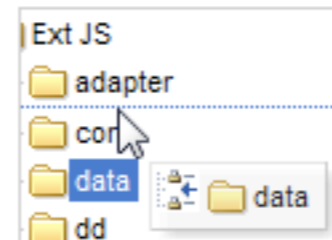
Page 1 of 17

Paging

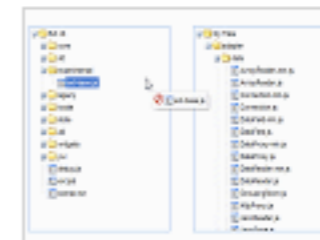
Expander Rows, Collapse and F

Company
3m Co
Alcoa Inc
Altria Group Inc
American Express Company
American International Group, Inc.

Grid Plugins



Drag and Drop Reordering



Multiple trees

Task	Order	Step
Check the system	1	1
Check the system	2	2
Check the system	3	3
Check the system	4	4
Check the system	5	5
Check the system	6	6
Check the system	7	7
Check the system	8	8
Check the system	9	9
Check the system	10	10
Check the system	11	11
Check the system	12	12
Check the system	13	13
Check the system	14	14
Check the system	15	15
Check the system	16	16
Check the system	17	17
Check the system	18	18
Check the system	19	19
Check the system	20	20

Column Tree

<http://www.extjs.com>

Ext JS

Ext 2.0 Final Released

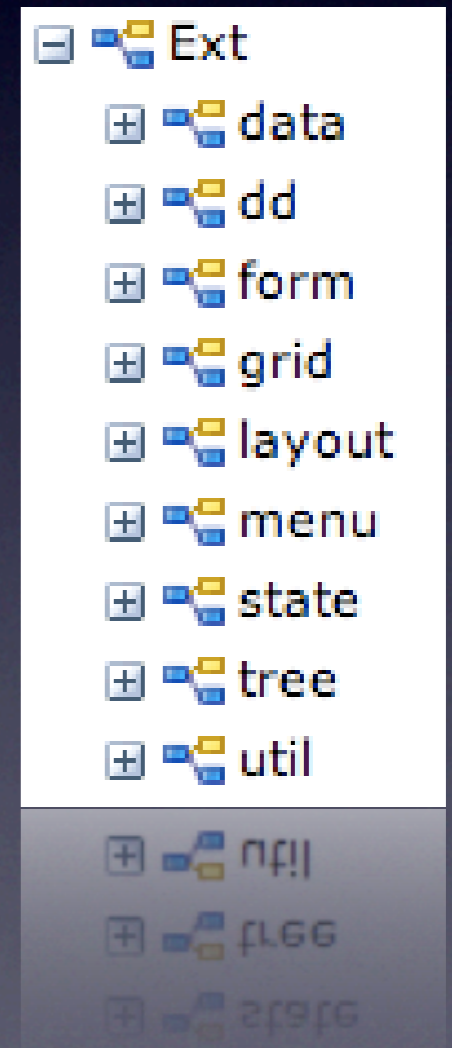
December 4, 2007 by ReyBango



The Ext team is proud to announce that the official release of Ext v2.0 is available for download. This new version of the Ext framework is the culmination of many long hours of work and dedication by the Ext Core team as well as our community of testers and many other people who have worked on the Ext

- JavaScript Framework
- Dual-License: LGPL + Commercial
 - rich GUI components
 - interface to JSON or XML based data sources (~ web services)
 - keeps state in session, i.e. cookies
-  cross-browser

6+ 1.5+ 2+ 9+



Segregation of Duties



- Model: business as usual
 - Controller: REST web service providing special :ext_json format, handle page flow / redirects
 - View: provide layout, DOM structure and include JavaScript
- *program* GUI and interactions in JavaScript
 - use Rails backend to load and persist data as needed

Test Drive: Scaffolding

Listing Posts

+ New... | Edit... | - Delete...

#	Title	Body	Published	Visible from	Visible to
1	First Post	This is my first post.	true	2008/01/07	2008/01/31
2	#2	My second post	true	2008/01/06	2008/01/31

Page 1 of 1 | Record 1 - 2 of 2

Data Grid with
toolbar and pagination

Forms with validation
(client side + server side)

Create Post

Title: !

Body: ! Title can't be blank

Published:

Visible from:

Visible to:

Save Back

Demo

Ext Scaffold Generator

- Create Rails app

```
rails ext_demo
```

- Download & extract Ext to public/ext

```
http://extjs.com/deployp/ext-2.0.zip
```

- Install ext_scaffold plugin:

```
script/plugin install http://rug-b.rubyforge.org/svn/ext\_scaffold
```

- Generate Demo resource

```
script/generate ext_scaffold Post title:string body:text  
published:boolean visible_from:datetime visible_to:date ;  
rake db:migrate
```

- Enjoy :-)

```
script/server -> http://localhost:3000/posts
```

Code

Model

- Nothing special :-)

```
class Post < ActiveRecord::Base
  validates_presence_of :title, :message => "Title can't be blank"
end
```

- Validation errors will be presented in forms just as Ext's own (client side) validation failures
- Ext Scaffold will return JSON error structure with per-field messages

Controller

- Support `ext_json` format -> `render :json`

```
def index
  respond_to do |format|
    format.html # index.html.erb (no data required)
    format.ext_json { render :json => Post.find(:all).to_ext_json }
  end
end

def create
  @post = Post.new(params[:post])

  respond_to do |format|
    if @post.save
      flash[:notice] = 'Post was successfully created.'
      format.ext_json { render(:update) { |page| page.redirect_to posts_url } }
    else
      format.ext_json { render :json => @post.to_ext_json(:success => false) }
    end
  end
end

...
```

#to_ext_json ?

- Implemented as extension to Array and ActiveRecord::Base respectively

- Array form

```
{"results": n,  
  "posts": [ {"id": 1, "title": "First Post",  
              "body": "This is my first post.", "published": true, ... },  
             ...  
            ]  
}
```

- Single (valid) record

```
{"data": { "post[id]": 1, "post[title]": "First Post",  
          "post[body]": "This is my first post.",  
          "post[published]": true, ...},  
  "success": true}
```

View

- Supported by (some tiny) helpers

```
<% javascript_tag do -%>
  var post_form_items = [
    <%= ext_field(:field_label => 'Title', :name => 'post[title]',
      :xtype => :text_field) %>,
    <%= ext_field(:field_label => 'Body', :name => 'post[body]',
      :xtype => :text_area) %>,
    <%= ext_field(:field_label => 'Published', :name => 'post[published]',
      :xtype => :checkbox) %>,
    <%= ext_field(:field_label => 'Visible from', :name => 'post[visible_from]',
      :xtype => :datetime_select) %>,
    <%= ext_field(:field_label => 'Visible to', :name => 'post[visible_to]',
      :xtype => :date_select) %>
  ];
<% end -%>

<%= ext_form_for :post,
  :element => 'post-form',
  :form_items => :post_form_items,
  :mode => :new %>
<div id="post-form"></div>
```

Resulting „HTML“

```
<script type="text/javascript">
  var post_form_items = [
    { fieldLabel: 'Title',      xtype: 'textfield',      name: 'post[title]'},
    { fieldLabel: 'Body',      xtype: 'textarea',      name: 'post[body]'},
    { fieldLabel: 'Published', xtype: 'checkbox', inputValue: '1', name: 'post[published]'},
    { xtype: 'hidden',        value: '0', name: 'post[published]' },
    ...
  ];

  Ext.onReady(function(){
    var panel = new Ext.FormPanel({
      url:      '/posts',
      title:    'Create Post',
      renderTo: 'post-form',
      baseParams: {authenticity_token: 'my_secret'},
      items:    post_form_items,
      buttons:  [ { text: 'Save', type: 'submit',
                    handler: function(){
                      panel.form.submit({url: '/posts?format=ext_json',
                                           waitMsg: 'Saving...'});
                    } },
                  { text: 'Back', handler: function(){ window.location.href = '/posts'; } }
                ]
    });
  });
</script>
<div id="post-form"></div>
```

Conclusion

- Using Ext gives
 - a very slim Rails backend,
just as we like it :-)
 - lots of JavaScript in the view :-(
● then again:
loads of functionality „for free“ and less
involvement of the backend in user
interactions :-))))

Q & A



Martin Rehfeld
martin.rehfeld@glnetworks.de
<http://inside.glnetworks.de>