

Rails

The Good Parts

with ❤️ from datarockets

Criticism

- ActiveRecord: validations, callbacks, and more
- ActionView: global helpers, views, rails-ujs
- ActionMailer and class methods
- Lack of abstractions for business logic
- Many-many more...

Good Parts

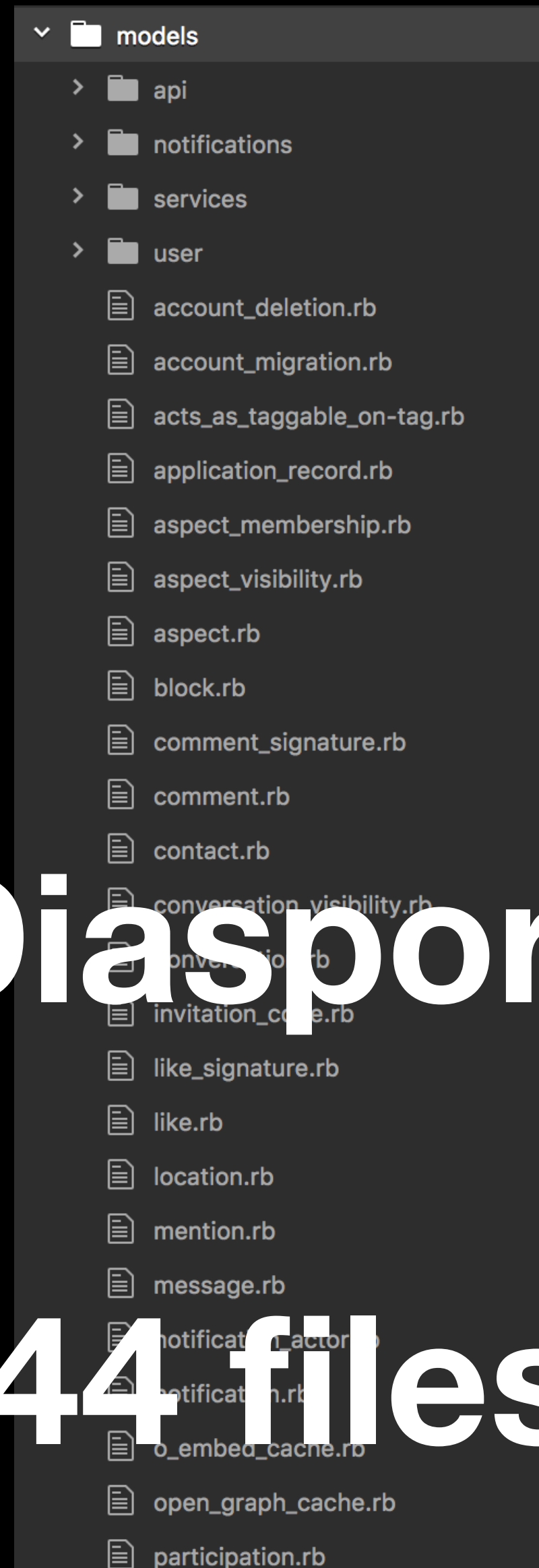
- ActiveRecord associations and queries DSL
- Database migrations
- Form helpers
- Security

Code Organization

Models

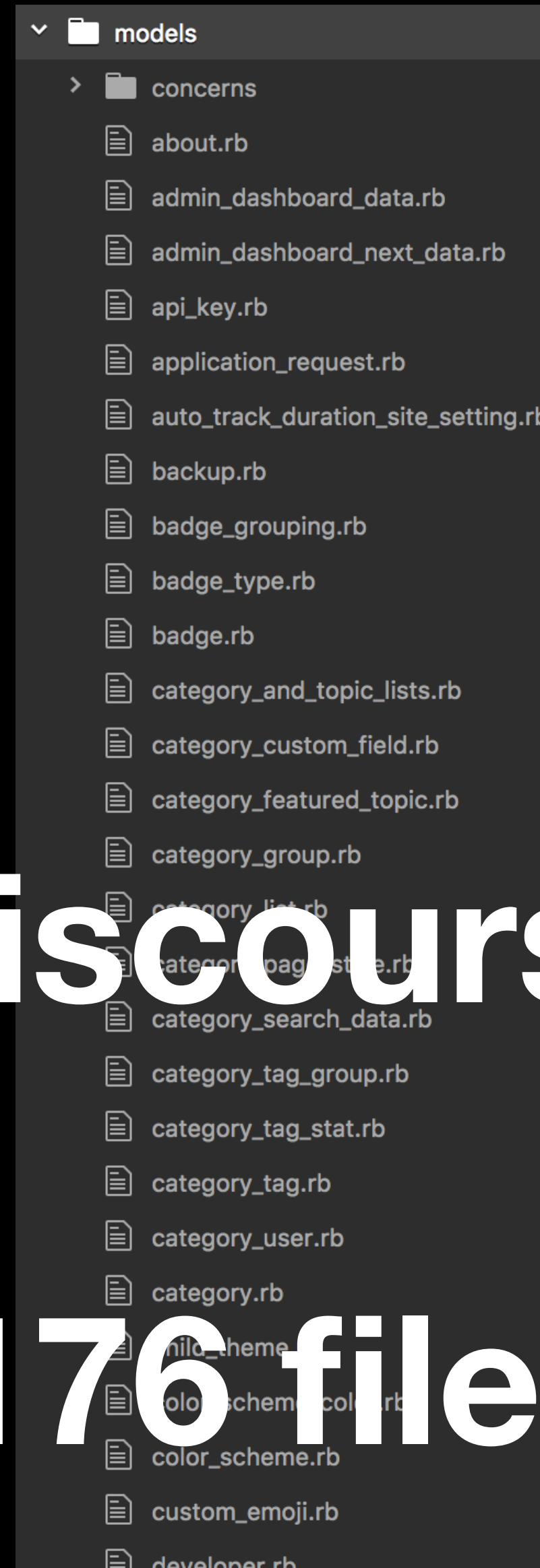


Flat Models Structure



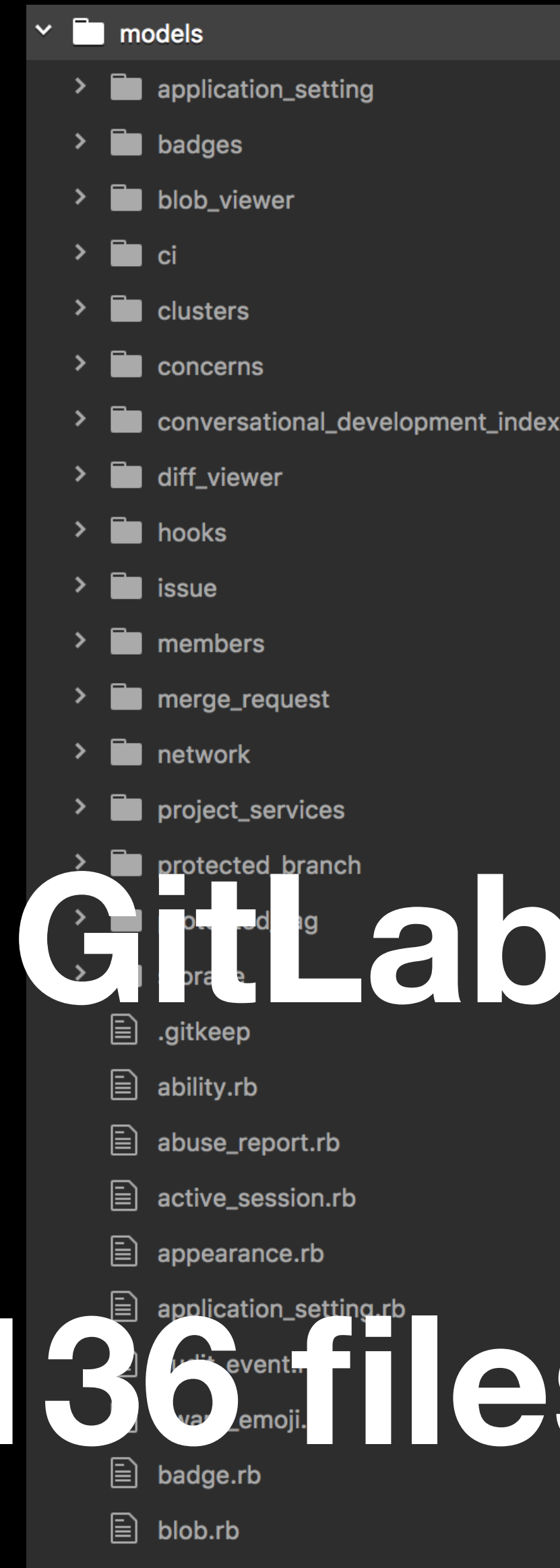
Diaspora

44 files



Discourse

176 files



GitLab

136 files



176?

Flat Models Structure

```
post_action_type.rb  
post_action.rb  
post_analyzer.rb  
post_custom_fields.rb  
post_detail.rb  
post_mover.rb  
post_reply.rb  
post_revision.rb  
post_search_data.rb  
post_stat.rb  
post_timing.rb  
post_upload.rb  
post.rb
```

Flat Models Structure

```
post_action_type.rb
post_action.rb
post_analyzer.rb
post_custom_fields.rb
post_detail.rb
post_mover.rb
post_reply.rb
post_revision.rb
post_search_data.rb
post_stat.rb
post_timing.rb
post_upload.rb
post.rb
```

```
class Post < ActiveRecord::Base
  has_many :post_uploads
  has_many :uploads, through: :post_
  has_many :post_details
  has_many :post_revisions
  # ...
end
```

Flat Models Structure

```
post_action_type.rb
post_action.rb
post_analyzer.rb
post_custom_fields.rb
post_detail.rb
post_mover.rb
post_reply.rb
post_revision.rb
post_search_data.rb
post_stat.rb
post_timing.rb
post_upload.rb
post.rb
```

```
class Post < ActiveRecord::Base
  has_many :post_uploads
  has_many :uploads, through: :post_
  has_many :post_details
  has_many :post_revisions
  # ...
end
```

```
post.post_details
```

```
post.post_revisions
```

Namespace Your Models

```
class Post::Upload    # post/upload.rb
  belongs_to :post
  belongs_to :upload, class_name: "::Upload"
end
```

```
class Post::Detail    # post/detail.rb
  belongs_to :post
end
```

```
class Post
  has_many :uploads, class_name: "Post::Upload"
  has_many :details
end
```

Namespace Your Models

```
post_action_type.rb
post_action.rb
post_analyzer.rb
post_custom_fields.rb
post_detail.rb
post_mover.rb
post_reply.rb
post_revision.rb
post_search_data.rb
post_stat.rb
post_timing.rb
post_upload.rb
post.rb
```



```
post/
  action/
    type.rb
  action.rb
  analyzer.rb
  custom_fields.rb
  detail.rb
  mover.rb
  reply.rb
  revision.rb
  search_data.rb
  stat.rb
  timing.rb
  upload.rb
post.rb
```

Namespace Your Models

```
post/  
post.rb
```

Controllers



Custom Actions

```
$ cat app/controllers/users_controller.rb | wc -l
```

```
1097
```

1097?



Custom Actions

```
resources :users do
  put "suspend"
  put "unsuspend"

  put "grant_moderation"
  put "revoke_moderation"

  # ...
end
```

Resourceful Routes

```
resources :users do  
  put "suspend"  
  put "unsuspend"  
end
```



```
resources :users do  
  resource :suspension,  
    only: [:create, :destroy]  
end
```

Resourceful Routes

```
resources :users do  
  put "suspend"  
  put "unsuspend"  
end
```



```
resources :users do  
  resource :suspension,  
    only: [:create, :destroy]  
end
```

```
DELETE /users/:user_id/suspension suspensions#destroy  
POST   /users/:user_id/suspension suspensions#create
```

Resourceful Routes

```
resources :users do  
  put "suspend"  
  put "unsuspend"  
end
```



```
resources :users do  
  resource :suspension,  
    only: [:create, :destroy]  
end
```

```
DELETE /users/:user_id/suspension suspensions#destroy  
POST   /users/:user_id/suspension suspensions#create
```

```
app/controllers/  
  users_controller.rb  
  suspensions_controller.rb
```

Resourceful Routes

```
resources :users do  
  put "suspend"  
  put "unsuspend"  
end
```



```
resources :users do  
  resource :suspension,  
    only: [:create, :destroy]  
end
```

```
DELETE /users/:user_id/suspension suspensions#destroy  
POST   /users/:user_id/suspension suspensions#create
```

```
app/controllers/  
  users_controller.rb  
  suspensions_controller.rb
```



Resourceful Routes

```
resources :users do
  scope module: :users do
    resource :suspension, only: [:create, :destroy]
    resource :moderation_permission, only: [:create, :destroy]
  end
end
```

```
app/controllers/
  users/
    suspensions_controller.rb
    moderator_permissions_controller.rb
  users_controller.rb
```

Service Objects



**TO BE
CONTINUED...** ➔

More

- ActiveRecord Querying
- HTTP2 support
- Logging

Thank you pivohug!

twitter.com/dzhlobo

fb.com/dima.zhlobo

vk.com/datarockets

with ❤️ from datarockets