

# Developing Your First Application Using ColdFusion 9 and ORM

ColdFusion Developer Week  
September 15, 2011





Bob Silverberg

e:[bob.silverberg@gmail.com](mailto:bob.silverberg@gmail.com) t:[@elegant\\_chaos](https://www.instagram.com/elegant_chaos)

w:[www.silverwareconsulting.com](http://www.silverwareconsulting.com)



# Applications Need Data



[http://www.flickr.com/photos/library\\_of\\_virginia/2898506631/](http://www.flickr.com/photos/library_of_virginia/2898506631/)



# The Traditional Approach

```
<cfquery name="qUser">  
  SELECT  IDUser, Name, Email  
  FROM    User  
  WHERE   IDUser = <cfqueryparam value="#url.id#" />  
</cfquery>
```

```
<form method="post" action="user.cfm?id=#url.id#">  
  Name: <input type="text" name="name" value="#qUser.Name#" />  
  Email: <input type="text" name="email" value="#qUser.Email#" />  
  <input type="submit" value="Submit" />  
</form>
```



# A Different Way of Working with Data



<http://www.flickr.com/photos/lafruu/4154809953/>



# Query vs Object





# Using Objects for Output

```
<form method="post" action="user.cfm?id=#url.id#">  
  Name: <input type="text" name="name" value="#qUser.Name#" />  
  Email: <input type="text" name="email" value="#qUser.Email#" />  
  <input type="submit" value="Submit" />  
</form>
```

```
<form method="post" action="user.cfm?id=#url.id#">  
  Name: <input type="text" name="name" value="#User.getName()" />  
  Email: <input type="text" name="email" value="#User.getEmail()" />  
  <input type="submit" value="Submit" />  
</form>
```



# What is an ORM?





# What is Hibernation?





# CF / Hibernate Integration



**INTEGRATION**

Just because you can, doesn't mean you should.



# Application Settings

Application.cfc:

```
<cfcomponent output="false">
  <cfset this.name = "ormBlog" />
  <cfset this.ormenabled = true />
  <cfset this.datasource = "myDSN" />
  <cfset this.ormsettings = {} />
  <cfset this.ormsettings.cfclocation = "model" />
  <cfset this.ormsettings.dbcreate = "update" />
</cfcomponent>
```



# Mapping an Object

User.cfc:

```
<cfcomponent persistent="true">
  <cfproperty name="Id" fieldtype="id" generator="native" />
  <cfproperty name="Name" />
  <cfproperty name="Email" length="200" />
  <cfproperty name="Active" type="boolean" notnull="true" />
</cfcomponent>
```



# Listing Users

```
<cfquery name="qUsers">  
  SELECT  IDUser, Name, Email, Active  
  FROM    User  
</cfquery>
```

```
<cfquery name="qUsers">  
  SELECT  IDUser, Name, Email, Active  
  FROM    User  
  WHERE  Active = 1  
</cfquery>
```

```
<cfquery name="qUsers">  
  SELECT  IDUser, Name, Email, Active  
  FROM    User  
  ORDER  BY Name  
</cfquery>
```

```
<cfset Users = EntityLoad("User") />
```

```
<cfset Users =  
  EntityLoad("User",{Active=true}) />
```

```
<cfset Users =  
  EntityLoad("User",{ }, "Name") />
```



# Creating / Getting a User

```
<cfquery name="qUser">  
  SELECT  IDUser, Name, Email, Active  
  FROM    User  
  WHERE  IDUser =  
         <cfqueryparam value="#url.id#" />  
</cfquery>
```

```
<cfset User = New model.User() />  
<cfset User = EntityNew("User") />  
  
<cfset User =  
  EntityLoadByPK("User",url.id) />
```



# Deleting a User

```
<cfquery name="qUser">  
  DELETE FROM User  
  WHERE IDUser =  
    <cfqueryparam value="#url.id#" />  
</cfquery>
```

```
<cfset User =  
  EntityLoadByPK("User",url.id) />  
<cfset EntityDelete(User) />
```



# Saving a User

```
<cfquery name="qUser">
<cfif Val(url.id) eq 0>
  INSERT INTO User (Name, Email, Active)
  VALUES (<cfqueryparam value="#form.Name#" />,
  <cfqueryparam value="#form.Email#" />,
  <cfqueryparam value="#Val(form.Active)#" />)
<cfelse>
  UPDATE User
  SET   Name = <cfqueryparam value="#form.Name#" />,
        Email = <cfqueryparam value="#form.Email#" />,
        Active = <cfqueryparam value="#Val(form.Active)#" />
  WHERE IDUser = <cfqueryparam value="#url.id#" />
</cfif>
</cfquery>
```

```
<cfset User.setName(form.name) />
<cfset User.setEmail(form.email) />
<cfset User.setActive(Val(form.active)) />
<cfset EntitySave(User) />
```



# Recap

- EntityLoad(entityname,[filtercriteria],[sortorder])
- EntityNew(entityname)
- EntityLoadByPK(entityname,id)
- EntityDelete(object)
- EntitySave(object)

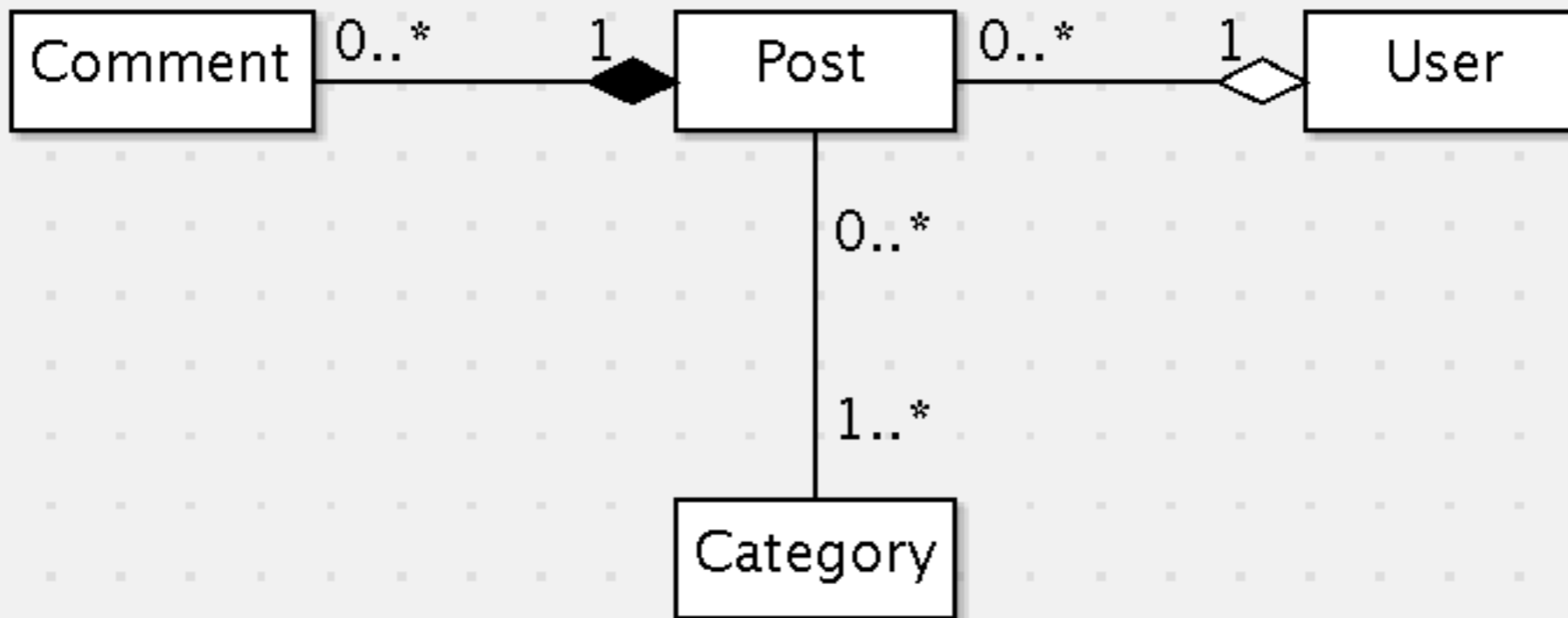


# No Object Is An Island





# Object Relationships





# Post Object Properties

Post.cfc:

```
<cfcomponent persistent="true" output="false">  
  <cfproperty name="IDPost" fieldtype="id" generator="native" />  
  <cfproperty name="Title" />  
  <cfproperty name="Body" />  
  <cfproperty name="DateTime" type="date" />  
</cfcomponent>
```



# Post Object Properties

| component <u>GettingStarted.Relationships.Post</u> |             |                                  |
|--|-------------|----------------------------------|
| PROPERTIES   | IDPost      | 1                                |
|  | Title       | My First Blog Post               |
|  | Body        | Here is some interesting content |
|  | DateTime    | {ts '2009-10-17 00:00:00'}       |
| METHODS  | GETTITLE    |                                  |
|  | SETBODY     |                                  |
|  | GETBODY     |                                  |
|  | SETDATETIME |                                  |
|  | GETIDPOST   |                                  |
|  | GETDATETIME |                                  |
|  | SETIDPOST   |                                  |
|  | SETTITLE    |                                  |



# Displaying the Post

DisplayPost.cfm:

```
<cfset Post = EntityLoadByPK("Post",1) />
<cfoutput>
<h2>#Post.getTitle()#</h2>
Posted on #DateFormat(Post.getDateTime(),"mmm dd, yyyy")#
by ???
<p>#Post.getBody()#</p>
</cfoutput>
```



# Many Posts can be written by one User

Post.cfc:

```
<cfcomponent persistent="true" output="false">
  <cfproperty name="IDPost" fieldtype="id" generator="native" />
  <cfproperty name="Title" />
  <cfproperty name="Body" />
  <cfproperty name="DateTime" type="date" />

  <cfproperty name="Author" cfc="User" fieldtype="many-to-one"
    fkcolumn="IDUser" />

</cfcomponent>
```



# Many Posts can be written by one User

| component <u>GettingStarted.Relationships.Post</u> |          |   |  |  |            |        |   |      |     |       |                          |         |  |
|--|----------|---|--|--|------------|--------|---|------|-----|-------|--------------------------|---------|--|
| PROPERTIES   | IDPost   | 1   |  |  |            |        |   |      |     |       |                          |         |  |
|  | Title    | My First Blog Post  |  |  |            |        |   |      |     |       |                          |         |  |
|  | Body     | Here is some interesting content  |  |  |            |        |   |      |     |       |                          |         |  |
|  | DateTime | {ts '2009-10-17 00:00:00'}  |  |  |            |        |   |      |     |       |                          |         |  |
|  | Author   | <table border="1"><thead><tr><th colspan="2">component <u>GettingStarted.Relationships.User</u></th></tr></thead><tbody><tr><td rowspan="3">PROPERTIES</td><td>IDUser</td><td>1</td></tr><tr><td>Name</td><td>Bob</td></tr><tr><td>Email</td><td>bob.silverberg@gmail.com</td></tr><tr><td colspan="2">METHODS</td><td></td></tr></tbody></table> | component <u>GettingStarted.Relationships.User</u> |  | PROPERTIES | IDUser | 1 | Name | Bob | Email | bob.silverberg@gmail.com | METHODS |  |
| component <u>GettingStarted.Relationships.User</u> |          |   |  |  |            |        |   |      |     |       |                          |         |  |
| PROPERTIES   | IDUser   | 1   |  |  |            |        |   |      |     |       |                          |         |  |
|  | Name     | Bob   |  |  |            |        |   |      |     |       |                          |         |  |
|  | Email    | bob.silverberg@gmail.com  |  |  |            |        |   |      |     |       |                          |         |  |
| METHODS  |          |   |  |  |            |        |   |      |     |       |                          |         |  |
| METHODS  |          |   |  |  |            |        |   |      |     |       |                          |         |  |



# Many Posts can be written by one User

DisplayPost.cfm:

```
<cfset Post = EntityLoadByPK("Post",1) />
<cfoutput>
<h2>#Post.getTitle()#</h2>
Posted on #DateFormat(Post.getDateTime(),"mmm dd, yyyy")#
by #Post.getAuthor().getName()#
<p>#Post.getBody()#</p>
</cfoutput>
```



# One Post can have Many Comments

Post.cfc:

```
<cfcomponent persistent="true" output="false">
  <cfproperty name="IDPost" fieldtype="id" generator="native" />
  <cfproperty name="Title" />
  <cfproperty name="Body" />
  <cfproperty name="DateTime" type="date" />
  <cfproperty name="Author" cfc="User" fieldtype="many-to-one"
    fkcolumn="IDUser" />

  <cfproperty name="Comments" cfc="Comment"
    fieldtype="one-to-many" fkcolumn="IDPost" type="array"
    singularname="Comment" />

</cfcomponent>
```



**component GettingStarted.Relationships.Post**

**PROPERTIES**

|          |                                  |
|----------|----------------------------------|
| IDPost   | 1                                |
| Title    | My First Blog Post               |
| Body     | Here is some interesting content |
| DateTime | {ts '2009-10-17 00:00:00'}       |

**Author**

**component GettingStarted.Relationships.User**

**PROPERTIES**

|        |                          |
|--------|--------------------------|
| IDUser | 1                        |
| Name   | Bob                      |
| Email  | bob.silverberg@gmail.com |

**METHODS**

**Comments**

**array**

1

**component GettingStarted.Relationships.Comment**

**PROPERTIES**

|           |                            |
|-----------|----------------------------|
| IDComment | 1                          |
| Name      | Joe Commenter              |
| Value     | What a great post!         |
| DateTime  | {ts '2009-10-17 00:00:00'} |

**METHODS**

**METHODS**



# One Post can have Many Comments

DisplayPost.cfm:

```
<cfset Post = EntityLoadByPK("Post",1) />
<cfoutput>
<h2>#Post.getTitle()#</h2>
Posted on #DateFormat(Post.getDateTime(),"mmm dd, yyyy")#
by #Post.getAuthor().getName()#
<p>#Post.getBody()#</p>
<cfloop array="#Post.getComments()#" index="comment">
  <p>
    #comment.getName()# wrote on
    #DateFormat(comment.getDateTime(),"mmm dd, yyyy")#
    <br />
    #comment.getValue()#
  </p>
</cfloop>
</cfoutput>
```



# One Post can belong to many Categories & One Category can have many Posts

Post.cfc:

```
<cfcomponent persistent="true" output="false">
  <cfproperty name="IDPost" fieldtype="id" generator="native" />
  <cfproperty name="Title" />
  <cfproperty name="Body" />
  <cfproperty name="DateTime" type="date" />
  <cfproperty name="Author" cfc="User" fieldtype="many-to-one"
    fkcolumn="IDUser" />
  <cfproperty name="Comments" cfc="Comment"
    fieldtype="one-to-many" fkcolumn="IDPost" type="array"
    singularname="Comment" />

  <cfproperty name="Categories" cfc="Category"
    fieldtype="many-to-many" type="array"
    singularname="Category" linktable="PostCategories" />

</cfcomponent>
```



component GettingStarted.Relationships.Post

PROPERTIES

|          |                                  |
|----------|----------------------------------|
| IDPost   | 1                                |
| Title    | My First Blog Post               |
| Body     | Here is some interesting content |
| DateTime | {ts '2009-10-17 00:00:00'}       |

Author

Comments

Categories

**array**

|            |   |   |            |   |  |      |            |  |            |   |  |       |   |
|------------|---|---|------------|---|--|------|------------|--|------------|---|--|-------|---|
| 1          | <p><b>component <u>GettingStarted.Relationships.Category</u></b></p> <table border="1"><tr><td>PROPERTIES</td><td>IDCategory</td><td>1</td></tr><tr><td></td><td>Name</td><td>ColdFusion</td></tr><tr><td></td><td>OrderIndex</td><td>1</td></tr><tr><td></td><td>Posts</td><td><b>array</b><br/>1 [see cfc1 for Post details]</td></tr></table> <p>METHODS</p> | PROPERTIES                                    | IDCategory | 1 |  | Name | ColdFusion |  | OrderIndex | 1 |  | Posts | <b>array</b><br>1 [see cfc1 for Post details] |
| PROPERTIES | IDCategory  | 1   |            |   |  |      |            |  |            |   |  |       |   |
|            | Name  | ColdFusion                                    |            |   |  |      |            |  |            |   |  |       |   |
|            | OrderIndex  | 1   |            |   |  |      |            |  |            |   |  |       |   |
|            | Posts   | <b>array</b><br>1 [see cfc1 for Post details] |            |   |  |      |            |  |            |   |  |       |   |
| 2          | <p><b>component <u>GettingStarted.Relationships.Category</u></b></p> <table border="1"><tr><td>PROPERTIES</td><td>IDCategory</td><td>2</td></tr><tr><td></td><td>Name</td><td>Hibernate</td></tr><tr><td></td><td>OrderIndex</td><td>2</td></tr><tr><td></td><td>Posts</td><td><b>array</b><br/>1 [see cfc1 for Post details]</td></tr></table> <p>METHODS</p>  | PROPERTIES                                    | IDCategory | 2 |  | Name | Hibernate  |  | OrderIndex | 2 |  | Posts | <b>array</b><br>1 [see cfc1 for Post details] |
| PROPERTIES | IDCategory  | 2   |            |   |  |      |            |  |            |   |  |       |   |
|            | Name  | Hibernate                                     |            |   |  |      |            |  |            |   |  |       |   |
|            | OrderIndex  | 2   |            |   |  |      |            |  |            |   |  |       |   |
|            | Posts   | <b>array</b><br>1 [see cfc1 for Post details] |            |   |  |      |            |  |            |   |  |       |   |

METHODS



# One Post can belong to many Categories & One Category can have many Posts

## Category.cfc:

```
<cfcomponent persistent="true" output="false">  
  <cfproperty name="IDCategory" fieldtype="id" generator="native" />  
  <cfproperty name="Name" />  
  <cfproperty name="OrderIndex" type="numeric" />  
  <cfproperty name="Posts" cfc="Post"  
    fieldtype="many-to-many" type="array"  
    singularname="Post" linktable="PostCategories"  
    inverse="true" />  
</cfcomponent>
```



[cfc1]

component GettingStarted.Relationships.Category

PROPERTIES

|            |            |
|------------|------------|
| IDCategory | 1          |
| Name       | ColdFusion |
| OrderIndex | 1          |

Posts

array

1

[cfc2]

component GettingStarted.Relationships.Post

PROPERTIES

|          |                                  |
|----------|----------------------------------|
| IDPost   | 1                                |
| Title    | My First Blog Post               |
| Body     | Here is some interesting content |
| DateTime | {ts '2009-10-17 00:00:00'}       |
| Author   |                                  |
| Comments |                                  |

Categories

array

1

[\[see cfc1 for Category details\]](#)

2

component GettingStarted.Relationships.Category

PROPERTIES

|            |           |
|------------|-----------|
| IDCategory | 2         |
| Name       | Hibernate |
| OrderIndex | 2         |
| Posts      | array     |

1

[\[see cfc2 for Post details\]](#)

METHODS

METHODS

METHODS



# Putting It All Together

## MaintainPost.cfm:

```
<cfset Post = EntityLoadByPK("Post", Val(form.IDPost)) />
<cfset Post.setTi
<cfset Post.setBo
<cfset User = Ent
<cfset Post.setUs
<cfset Categories
<cfloop condition
    <cfset Post.re
</cfloop>
<cfif StructKeyExists(form, "IDCategory")>
    <cfloop list="#form.IDCategory#" index="i">
        <cfset Category = EntityLoadByPK("Category", i) />
        <cfset Post.addCategory(Category) />
    </cfloop>
</cfif>
<cfset EntitySave(Post) />
```

| struct     |  |
|------------|--|
| BODY       | Here is some really interesting content! |
| FIELDNAMES | IDPOST,TITLE,BODY,IDCATEGORY,IDUSER      |
| IDCATEGORY | 1,2                                      |
| IDPOST     | 1  |
| IDUSER     | 1  |
| TITLE      | My First Blog Post                       |

/>



# What Else Is There?

- Inheritance
- Ad Hoc Querying (HQL)
- Event Handling
- SQL/Performance Tuning
- Caching



# Want to Learn More?

- ColdFusion Documentation
- Blog Posts
  - Mark M, Brian K, Barney B, and more...
- [groups.google.ca/group/cf-orm-dev](https://groups.google.ca/group/cf-orm-dev)
- Java Persistence with Hibernate (Book)



# Q & A

Bob Silverberg

e:[bob.silverberg@gmail.com](mailto:bob.silverberg@gmail.com) t:@elegant\_chaos

w:[www.silverwareconsulting.com](http://www.silverwareconsulting.com)

CodeBassRadio: <http://cbrtune.in>