

# Finding Rails

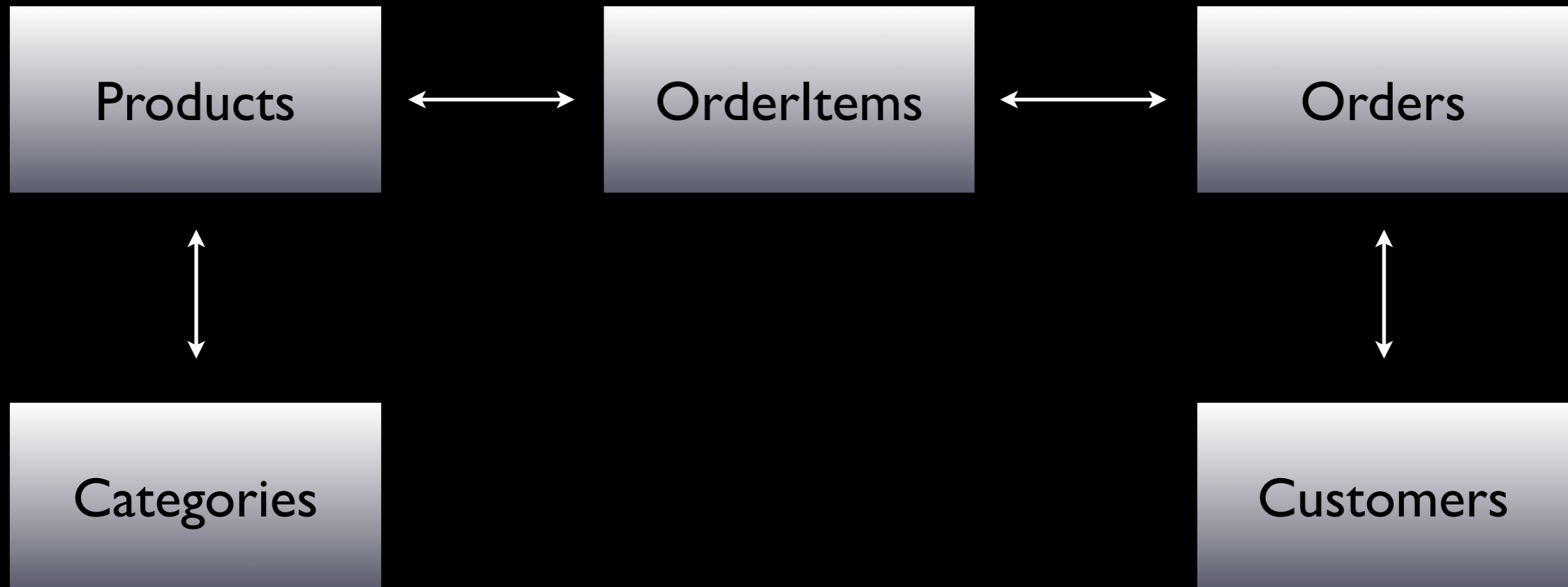
Actually... Find *in* Rails

Sebastian Delmont

[sd@notso.net](mailto:sd@notso.net)

[streeteasy.com](http://streeteasy.com)

# The Cast



# The Cast

```
class Product < ActiveRecord::Base
  has_and_belongs_to :categories
  has_many :order_items
end
```

```
class Category < ActiveRecord::Base
  has_and_belongs_to :products
end
```

# The Cast

```
class Order < ActiveRecord::Base
  belongs_to :customer
  has_many :order_items
end
```

```
class OrderItem < ActiveRecord::Base
  belongs_to :order
  belongs_to :product
end
```

```
class Customer < ActiveRecord::Base
  has_many :orders
end
```

# Finding The One

```
class Customer < ActiveRecord::Base  
end
```

```
@customer = Customer.find(1)
```

```
SELECT * FROM customers  
WHERE (id = 1) LIMIT 1
```

# Finding The One

```
class Customer < ActiveRecord::Base
  set_table_name "our_beloved_customers"
  set_primary_key :customer_ssn
end
```

```
@customer = Customer.find(1)
```

```
SELECT * FROM our_beloved_customers
WHERE (customer_ssn = 1) LIMIT 1
```

# Finding The One

```
class ActiveRecord::Base
  primary_key_prefix = :table_name_with_underscore
end
```

```
@customer = Customer.find(1)
```

```
SELECT * FROM customers
WHERE (customer_id = 1) LIMIT 1
```

# Finding *that* one

```
@customer = Customer.find_by_name("John")
```

```
SELECT * FROM customers  
WHERE (customers.name = "John") LIMIT 1
```

# Finding *that* one

```
@customer = Customer.find_by_name_and_last_name(  
    "John", "Doe"  
)
```

```
SELECT * FROM customers  
WHERE (name = "John" AND last_name = "Doe") LIMIT 1
```

# Finding :all of them

```
@customer = Customer.find(:all)
```

```
SELECT * FROM customers
```

# Finding :all of some

```
@customer = Customer.find_all_by_name_and_last_name(  
    "John", "Doe"  
)
```

```
SELECT * FROM customers  
WHERE (name = "John" AND last_name = "Doe")
```

# Finding :all of some

```
@customer = Customer.find(:all,  
  :conditions => [  
    "name LIKE :name_pattern",  
    {:name_pattern => "John%"}  
  ]  
)
```

```
SELECT * FROM customers  
WHERE (name LIKE 'John%')
```

# Finding :all of some

```
@customer = Customer.find(:all,  
  :conditions => [  
    "name LIKE ?",  
    ["John%"]  
  ]  
)
```

```
SELECT * FROM customers  
WHERE (name LIKE 'John%')
```

# Finding carefully

```
@customer = Customer.find(:all,  
  :select => "name, email",  
  :conditions => [  
    "name LIKE :name_pattern",  
    {:name_pattern => "John%"}  
  ]  
)
```

```
SELECT name, email FROM customers  
WHERE (name LIKE 'John%')
```

# Paged Finding

```
@customer = Customer.find(:all,  
  :conditions => [  
    "name LIKE :name_pattern",  
    {:name_pattern => "John%"}  
  ]  
  :limit => 10, :offset => 30,  
)
```

```
SELECT * FROM customers  
WHERE (name LIKE 'John%')  
LIMIT 10 OFFSET 30
```

# Paged Finding

```
params[:page] => 30
@pages, @customers = paginate(Customer,
  :conditions => [
    "name LIKE :name_pattern",
    {:name_pattern => "John%"}
  ],
  :per_page => 10
)
```

```
SELECT * FROM customers
WHERE (name LIKE 'John%')
LIMIT 10 OFFSET 30
```

# Finding a Join

```
params[:minimum] = 100
@customers = Customer.find(:all,
  :joins => "INNER JOIN orders ON customers.id =
orders.customer_id"
  :conditions => [
    "orders.total >= :minimum",
    {:minimum => params[:minimum]}
  ],
)
```

```
SELECT * FROM customers
      INNER JOIN orders
      ON customers.id = orders.customer_id
      WHERE (orders.total >= 100)
```

# Eagerly Finding a Join

```
params[:minimum] = 100
@customers = Customer.find(:all,
  :include => :orders
  :conditions => [
    "orders.total >= :minimum",
    {:minimum => params[:minimum]}
  ],
)
```

```
SELECT * FROM customers
      INNER JOIN orders
ON customers.id = orders.customer_id
WHERE (orders.total >= 100)
```

# Eagerly Finding a Join

```
params[:minimum] = 100
@customers = Customer.find(:all,
  :include => :orders
  :conditions => [
    "orders.total >= :minimum",
    {:minimum => params[:minimum]}
  ],
)
```

```
SELECT customers."id" AS t0_r0, customers."name" AS t0_r1,
customers."last_name" AS t0_r2, customers."email" AS t0_r3, orders."id" AS
t1_r0, orders."customer_id" AS t1_r1, orders."created_at" AS t1_r2,
orders."status" AS t1_r3, orders."total" AS t1_r4
FROM customers LEFT OUTER JOIN orders ON orders.customer_id = customers.id
WHERE (orders.total >= 100)
```

# Finding without 'find'

```
@customer = Customer.find(1)
```

```
@customer.orders
```

```
SELECT * FROM customers  
WHERE (customers.id = 1) LIMIT 1
```

```
SELECT * FROM orders WHERE (orders.customer_id = 1)
```

# Finding :through

```
class Order
  has_many :products, :through => :order_items
end
```

```
@order = Order.find(1)
@order.products
```

```
SELECT * FROM orders WHERE id = 1
SELECT products.* FROM products
INNER JOIN order_items ON products.id = order_items.product_id
WHERE (order_items.order_id = 1)
```

# All there is to Find

:select  
:joins  
:include  
:conditions  
:group  
:order  
:limit  
:offset  
:readonly

```
SELECT :select FROM :table :joins :include  
      WHERE :conditions GROUP BY :group  
ORDER BY :order LIMIT :limit OFFSET :offset
```

# All there is to Find

```
find(id)
find(:first, ...)
find(:all, ...)
find_by_xyz
find_all_by_xyz
find_or_create
find_or_create_by_xyz
find_by_sql
connection.execute
{collection}
```

that's all, folks