



# ANF 2012 - Mathrice

S'interfacer avec un LDAP

« Ruby ActiveLDAP »

David Delavennat



# Agenda de la présentation

- Connexion
- Modèle
  - Dataset
  - Associations
  - Manipulations



# Connexion

ActiveLdap::Base.setup\_connection(

:host => 'directory.example.com',

:port => 10389,

:base => 'dc=example,dc=com',

:bind\_dn => "uid=admin,ou=system",

:password\_block => Proc.new { 'secret' },

:allow\_anonymous => false,

:try\_sasl => false,

:method => :plain

)



# Object Model : dataset

```
class User < ActiveLdap::Base
```

```
  ldap_mapping(  
    :dn_attribute => 'cn',  
    :prefix       => "ou=users,...",  
    :classes      => [  
      'top',  
      'inetOrgPerson'  
    ]  
  )  
end
```

```
class Group < ActiveLdap::Base
```

```
  ldap_mapping(  
    :dn_attribute => 'cn',  
    :prefix       => "ou=groups,...",  
    :classes      => [  
      'top',  
      'posixGroup'  
    ]  
  )  
end
```



# Object Model : associations

```
class User < ActiveLdap::Base
```

```
...
```

```
  belongs_to( :groups,
```

```
    :many      => 'memberUid',
```

```
    :class_name => 'Group',
```

```
    :primary_key => 'cn'
```

```
  )
```

```
End
```

```
class Group < ActiveLdap::Base
```

```
...
```

```
  has_many( :members,
```

```
    :wrap      => 'memberUid',
```

```
    :class_name => 'Users',
```

```
    :primary_key => 'cn'
```

```
  )
```

```
End
```



# Object Model : manipulation

```
user = User.new(cn)
```

```
user.sn = cn
```

```
user.email = 'leonardo.davinci@closlucce.fr'
```

```
user.password = ActiveSupport::UserPassword::sha("secret")
```

```
user.save
```

```
user = User.find(cn)
```

```
pp user.groups.map do |group| group.id end
```

```
User.destroy(cn)
```

```
users = User.all
```

```
pp groups.members.map |member| member.id end
```

```
users.each do |user| pp user end
```