

DataMapper == ActiveRecord 3.0?

- Miksi? (vrt. ActiveRecord)
 - Enemmän Rubyä, vähemmän SQL:ää
 - AR: Migraatio/malli -ongelmat
 - DataMapper: malli ohjaa tietokantaa eikä toisinpäin
 - Intuitiivinen olioidentiteetti
 - `entity.children.all? {|c| c.parent == entity} # => true`
 - Fiksumpi SQL-generointi: Lazy loading + Smart Associations
 - tekstikentät luetaan oletuksena vain tarvittaessa, assosioitujen tietojen lukeminen

Esimerkki

```
▪ class Employee
  include DataMapper::Resource

  property :id, Integer,          :serial => true
  property :name, String         # i can has ruby semantics!
  property :role, String,        :accessor => :private
  property :description, Text,    :lazy => false
  property :deleted_at, ParanoidDateTime
  belongs_to :company
  has n, :projects
end
```

Muuta

- Taulujen luominen
 - `DataManager#auto_migrate!` # checkout & deploy?

- Suorituskyky

- ks. <http://work.rowanhick.com/2008/02/19/rails-vs-merb-activerecorddatamapper/>

Rails: 42.16 req/s (baseline)

770 testiriviä

Merb + ActiveRecord: 57.80 req/s (1.37x)

Merb + Datamapper: 72.98 req/s (1.73x)

Rails: 22.82 req/s (baseline)

11050 testiriviä

Merb + ActiveRecord: 29.85 req/s (1.30x)

Merb + Datamapper: 33.91 req/s (1.48x)

Mutta ihan oikeasti...

- => Jarkon vuoro :*)