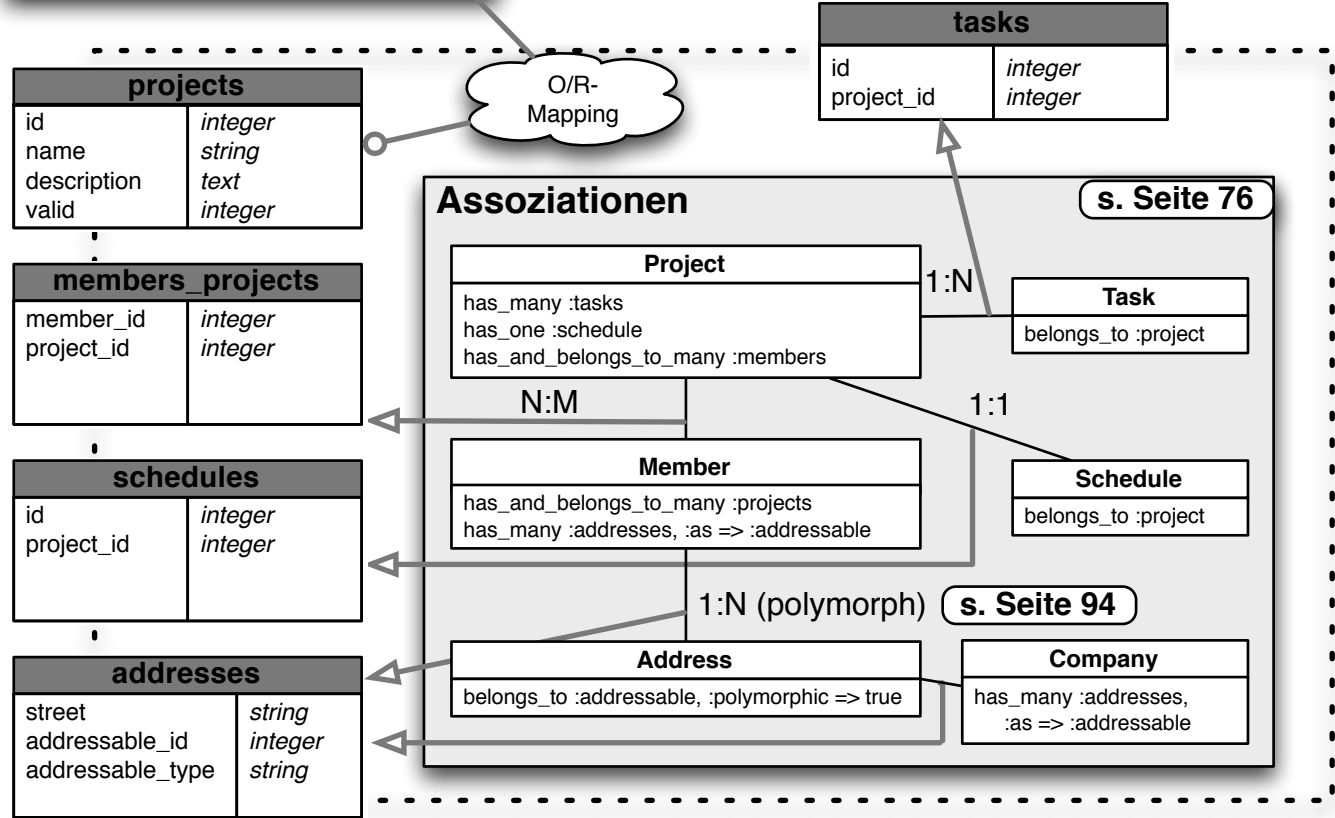
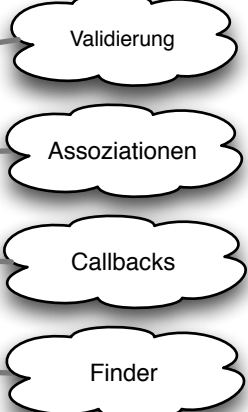


Modelle s. Seite 59

```
class Project < ActiveRecord::Base
  validates_presence_of :name
  has_many :tasks

  def after_save
    send_mail
  end

  def self.all_valid
    find(:all, :conditions => "valid = 1")
  end
end
```



CRUD

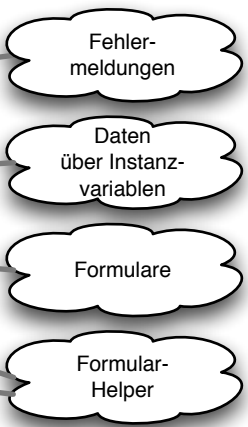
Erzeugung s. Seite 66 new create	Laden / Finden / Ordnen s. Seite 70 Project.find(1) Project.find(:first) Project.find(:all) Project.find(:all, :limit => 5) Project.find(:all, :conditions => "name = 'Rails'") Project.find(:all, :order => "name") Project.find_by_sql("select * from projects") Project.find_by_name("Web 2.0")	Aktualisieren s. Seite 67 save save! update_attribute update_attributes update(id, {}) update_all	Löschen s. Seite 69 delete delete_all destroy destroy_all
--	--	---	---

Validierung s. Seite 110 validates_presence_of :name validates_acceptance_of :terms validates_associated :tasks validates_confirmation_of :password validates_exclusion_of :name, :in => ["raff"] validates_format_of :name, :with => /.../ validates_inclusion_of :gender, :in => ["w", "m"] validates_length_of :name, :minimum => 4 validates_each :name do rec, attr, vall ... end validates_uniqueness_of :name validates_numericality_of :age	Metadaten s. Seite 75 Project.columns Project.content_columns Project.column_names	Vererbung s. Seite 100 Member Manager members id type int varchar
---	---	---

Callbacks s. Seite 115 Neues Objekt: o.save before_validation before_validation_on_create after_validation after_validation_on_create save before_save before_create after_create after_save Vorhandenes Objekt: o.save before_validation before_validation_on_update after_validation after_validation_on_update save before_save before_update after_update after_save o.destroy before_destroy destroy after_destroy	Transaktionen s. Seite 103 class Project < ActiveRecord::Base def move_task(task, project) transaction(self, project) do self.tasks.delete task project.tasks << task project.save! end end end
--	--

Views s. Seite 153

```
<html>
  <%= error_messages_for("project") %>
  Projekt: <%= @project.name %>
  <%= start_form_tag :action => "update", :id => @project %>
  <%= text_area :project, :description %>
  <%= collection_select(:project, :lead_id, Member.find(:all), :id, :surname %>)
  <%= submit_tag "Speichern" %>
  <%= end_form_tag %>
  <%= link_to "Zurück", :action => 'back' %>
</html>
```



Controller s. Seite 121

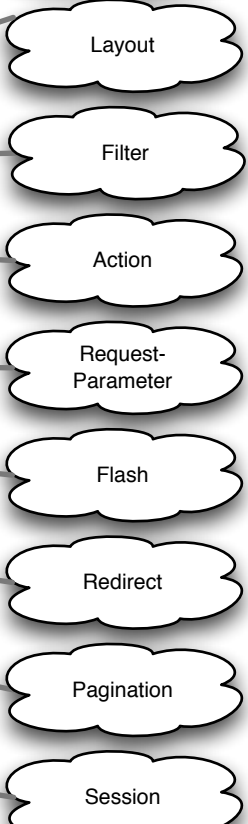
```
class ProjectController < ActionController::Base
  layout "ontrack"
  before_filter :authenticate

  def edit_project
    @project = Project.find(params[:id])
    render :action => "edit_project"
  end

  def update
    @project = Project.update(params[:project])
    flash[:nachricht] = "Projekt erzeugt"
    redirect_to :action => "show", :id => @project
  end

  def list_projects
    @project_pages, @projects = paginate(:projects, :order_by => "start", :per_page => 20)
  end

  def authenticate
    redirect_to :action => "login" unless session[:user]
  end
end
```



Formulare s. Seite 157 <%= start_form_tag :action => "update", :id => @project %> <%= text_field(:project, :name) %> <%= end_form_tag %> <% form_for :project, @project, :url => { :action => :update } do f %> <%= f.text_field :name %> <% end %>	Partials s. Seite 170 View <%= render :partial => "task", :object => @task%> <%= render :partial => "task", :collection => @tasks%> Controller render :partial => "task", :object => @task render :partial => "task", :collection => @tasks	Layout s. Seite 169 <html> <!-- header --> ... <!-- content --> <%= @content_for_layout %> ... <!-- footer --> </html>
---	---	--

Formular-Helfer s. Seite 158 Mit Modellbezug collection_select(:task, :member_id, Member.find(:all), :id, :surname) select(:project, :lead_id) date_select(:project, :start_date) text_field(:project, :name) date_select(:project, :start_date) text_area(:project, :description) checkbox(:project, :reviewed) file_field(:project, :report) hidden_field(:project, :last_edit)	Ohne Modellbezug s. Seite 160 text_field_tag password_field_tag radio_button_tag select_tag options_for_select start_form_tag end_form_tag submit_tag	Navigation s. Seite 165 link_to url_for button_to mail_to Ressourcen einbinden s. Seite 167 stylesheet_link_tag javascript_include_tag image_tag
--	---	--

Filter s. Seite 137 Methoden before_filter after_filter around_filter Klassen class EncryptFilter def self.filter(controller) ... end end	Render render :action => "list" s. Seite 124 render :template => "task/list" s. Seite 125 render :action => :update do page page.insert_html :bottom, "list", :partial => "task" end s. Seite 261	Components s. Seite 172 Controller render_component :controller => "admin", :action => "search" render_component_as_string :controller => "admin", :action => "search" View <%= render_component :controller => "admin", :action => "search" %>
Pagination s. Seite 174 Controller @project_pages, @projects = paginate(:projects, :order_by => "start", :per_page => 2) View <%= pagination_links(@project_pages, :window_size => 3) %>	Redirect s. Seite 129 redirect_to :action => "show", :controller => "project", :id => @project	