PgPower

Rails Postgres for Power Users

Mike Sandler
TMX Credit

Migrations & the Database

Migrations make database changes easier:

- easy DSL for the most common changes
- convenient conventions
- apply/revert changes; create schema dump

Migrations & the Database

Migrations make database changes easier:
- easy DSL for the most common changes
- convenient conventions
- apply/revert changes; create schema dump

But...

Advanced DB features not available in the DSL

- schemas, foreign key constraints, table & column comments, partial and functional indexes, concurrent index creation, etc

Problem

Can't we just use SQL for the 'fancy' stuff?

- schema dumper won't know how to dump it
- in most cases schema dumper will fail when trying to generate a schema.rb

Can't we just dump to sql?

- we could

Problem

Can't we just use SQL for the 'fancy' stuff?

- schema dumper won't know how to dump it
- in most cases schema dumper will fail when trying to generate a schema.rb

Can't we just dump to sql?

- we could

But...

There's got to be a better way!

Enter PgPower

Extends migrations DSL and schema dumper

- database schemas
- foreign key constraints and foreign key indexes
- partial indexes and indexes on expressions
- database comments on tables and columns
- add indexes concurrently (coming soon)
- more to come

Usage: setup

Install Gem
- In Gemfile: gem 'pg_power'
- bundle install

Works in Rails 3.1, 3.2
Works on Ruby 1.8.7, 1.9
Tested on Postgres 9.0, 9.1, 9.2

Usage: schemas

```
# Migration
 create_schema 'geography'
 create_table 'counties', :schema => 'geography' do |t|
  ...
 end

 move_table_to_schema :zip_codes, :geography
 move_table_to_schema 'geography.counties', :public
 drop_schema :old_schema
```

Usage: foreign keys

- base functionality imported from 'foreigner'
- expanded to be schema aware
- added index auto-generation

```
# use schemas, defaults for fk column and fk constraint name
add_foreign_key('content.comments', 'content.posts')

# skip index generation
add_foreign_key(:comments, :posts, :exclude_index => true)

# specify name of fk column and constraint
add_foreign_key(:comments, :posts, :column => :blog_post_id,
                :name => 'comments_posts_fk')
```

Usage: indexes

```ruby
# add a partial index to a table
add_index(:posts, [:country_id, :user_id], :where => 'active')

# add a partial index to a table in a different schema
add_index('blog.comments', :user_id, :where => 'active')

# add an index to an expression (functional index)
add_index(:comments, 'lower(text)')
```

Usage: comments

```
set_table_comment :phone_numbers,
                     'Stores normalized phone numbers'
set_column_comment :phone_numbers, :exchange,
                     '3 digit exchange of the number'

remove_table_comment :phone_numbers
remove_column_comment :phone_numbers, :exchange

# set comments for multiple columns on the same table
set_column_comments :phone_numbers,
           :area_code => '3 digit area code',
           :exchange  => '3 digit exchange of the number'
remove_column_comments :phone_numbers, :area_code, :exchange
```

Usage: concurrent indexes

Coming soon - currently in code review

```
# add index concurrently
add_index :posts, :user_id, :concurrently => true

# also works when creating foreign keys
add_foreign_key :posts, :users, :concurrent_index => true
```

More Info

https://github.com/TMXCredit/pg_power
https://rubygems.org/gems/pg_power

http://tmxcredit.com/tech-blog

@stealthmk

Mike Sandler
CIO, TMX Credit