Recurring Dates

John Crepezzi

@seejohnrun

outlawlovesong.com

John Crepezzi (@seejohnrun)

Patch

common time problems

- Daylight Savings Time
- Time Zones
- UTC / GMT Offsets
- February

recurrence problem

- *Consider:* "The first and last monday of each month"
- Difficult to implement
- Ugly
- Slow
- **Not a problem you should need to solve.**

ice_cube's solution

github.com / seejohnrun / ice_cube

- The iCalendar RFC (& examples)
- Super-clean, Ruby syntax
- Awesome name???

ice_cube's name

ice_cube example

```
schedule = Schedule.new(Time.local(2010, 1, 1))
schedule.rrule Rule.monthly.day_of_month(13).day(:friday)

schedule.first 10

# Fri Aug 13 00:00:00 -0400 2010
# Fri May 13 00:00:00 -0400 2011
# Fri Jan 13 00:00:00 -0500 2012
# Fri Apr 13 00:00:00 -0400 2012
# Fri Jul 13 00:00:00 -0400 2012
# Fri Sep 13 00:00:00 -0400 2013
# Fri Dec 13 00:00:00 -0500 2013
# Fri Jun 13 00:00:00 -0400 2014
# Fri Feb 13 00:00:00 -0500 2015
# Fri Mar 13 00:00:00 -0400 2015
```

ice_cube example

```ruby
schedule = Schedule.new(Time.local(2010, 1, 1))
schedule.rrule Rule.monthly.day_of_month(13).day(:friday)

schedule.first 10

# Fri Aug 13 00:00:00 -0400 2010
# Fri May 13 00:00:00 -0400 2011
# Fri Jan 13 00:00:00 -0500 2012
# Fri Apr 13 00:00:00 -0400 2012
# Fri Jul 13 00:00:00 -0400 2012
# Fri Sep 13 00:00:00 -0400 2013
# Fri Dec 13 00:00:00 -0500 2013
# Fri Jun 13 00:00:00 -0400 2014
# Fri Feb 13 00:00:00 -0500 2015
# Fri Mar 13 00:00:00 -0400 2015
```

ice_cube rules

```ruby
# Every month
rule = Rule.monthly

# Every other month
rule = Rule.monthly(2)

# Every nth month
rule = Rule.monthly(n)
```

Rule.yearly, Rule.monthly, Rule.weekly, Rule.daily
Rule.hourly, Rule.minutely, Rule.secondly

ice_cube validations

```ruby
# Every day
rule = Rule.daily

# Every day that is a friday
rule = Rule.daily.day(:friday)

# Every day that is a friday the 13th
rule = Rule.daily.day(:friday).day_of_month(13)
```

#day, #day_of_week, #day_of_month, #day_of_year, #month_of_year, #hour_of_day, #minute_of_hour, #second_of_minute

ice_cube schedules

```ruby
schedule = Schedule.new(Time.now)

# Occurs every day
schedule.rrule Rule.daily

# BUT, doesn't occur on days that are saturdays
schedule.exrule Rule.daily.day(:saturday)

schedule.occurs_on?(Date.today) # false
schedule.occurs_on?(Date.today + 1) # true
```

ice_cube blackouts

```
schedule = Schedule.new(Time.now)

schedule.rdate Time.local(2010, 5, 10, 6)
schedule.exdate Time.local(2010, 4, 16, 10)
```

Order of Precedence:

Exclusion Dates

Inclusion Dates

Exclusion Rules

Inclusion Rules

ice_cube's code

❧ *Querying*

```ruby
# Has an occurrence on a given date?
schedule.occurs_on? Date.today

# Has an occurrence that starts at a time?
schedule.occurs_at? Time.now

# Has an occurrence that is occurring at a given time
schedule = Schedule.new(start_time, :duration => 3600)
schedule.occurring_at? Time.now
```

ice_cube's code

❧ *Basic Expansion*

```ruby
# First 10 occurrences
schedule.first(10)

# All occurrences
schedule.all_occurrences

# Occurrences until a time
schedule.occurrences Time.tomorrow

# Occurrences between two times
schedule.occurrences_between Time.now, Time.tomorrow
```

ice_cube speaks

```ruby
rule = Rule.monthly.day_of_week(:friday => [-2])

rule.to_hash
rule.to_yaml
# "--- \n:rule_type: ...

rule.to_ical
# "FREQ=MONTHLY;BYDAY=-2FR"

rule.to_s
# "Monthly on the 2nd to last Friday"
```

ice_cube speaks

```
schedule = Schedule.new(Time.now)
schedule.rrule Rule.monthly.day_of_week(:friday => [-2])
schedule.exrule Rule.monthly.day_of_month(13)

schedule.to_hash
schedule.to_yaml
# "--- \n:rdates: ...

schedule.to_ical
# DTSTART;TZID=EDT:20100828T110326
# RRULE:FREQ=MONTHLY;BYDAY=-2FR
# EXRULE:FREQ=MONTHLY;BYMONTHDAY=13

rule.to_s
# "Monthly on the 2nd to last Friday / not Monthly on the
13rd day of the month”
```

ice_cube's magic?

- Lots of combinations
- Scary performance implications
- Cool pattern

ice_cube usage

- Events Calendars
- Timed Notifications
- Schedule Conflict Resolution
- Really complex task automation
- **Make something awesome and let me know.**

gem install ice_cube

outlawlovesong.com

John Crepezzi (@seejohnrun)