

i18n and L10n with Rails 3

Mattt Thompson (@matt)
Hacker, Gowalla

Austin on Rails, July 2010

i18n

L10n

i18n



internationalization

Supporting International Data

- Text Content
- Numbers
- Dates / Times
- Calendars
- Time Zones
- Currencies

L10n



localization

Translating UI and Content

- UI Strings
- Data

Globalization



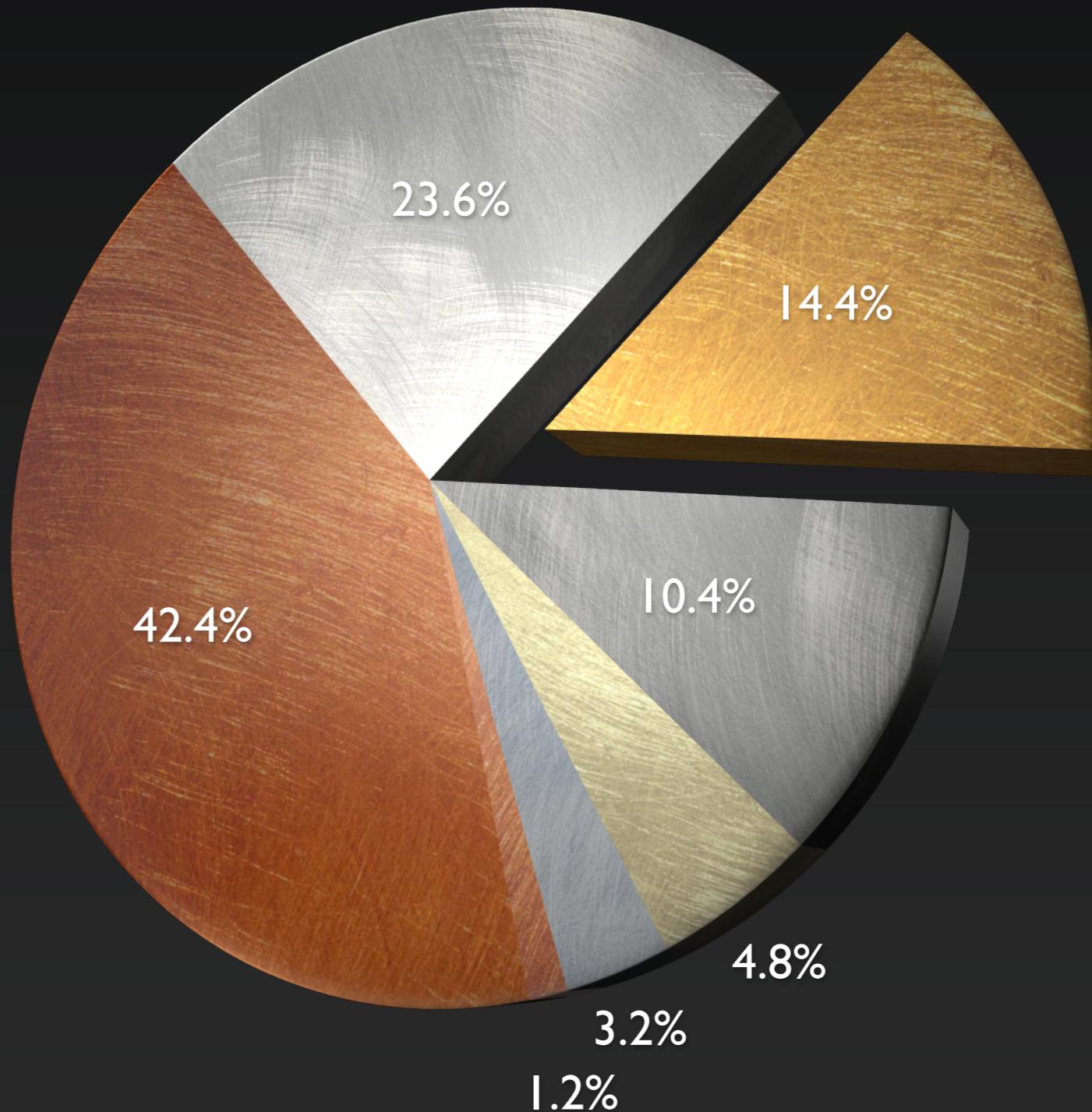
Globalization

i18n

L10n

Why?

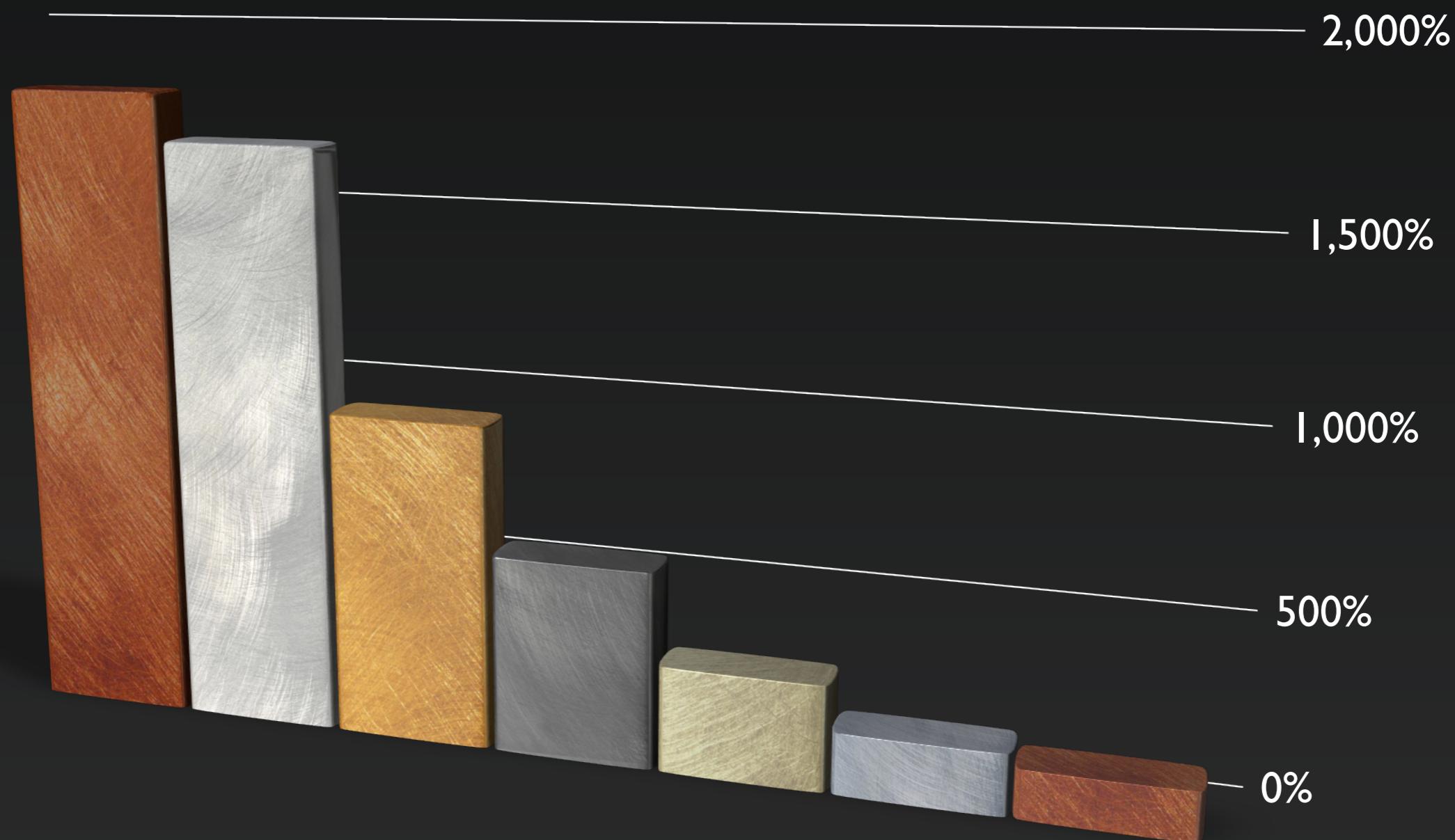
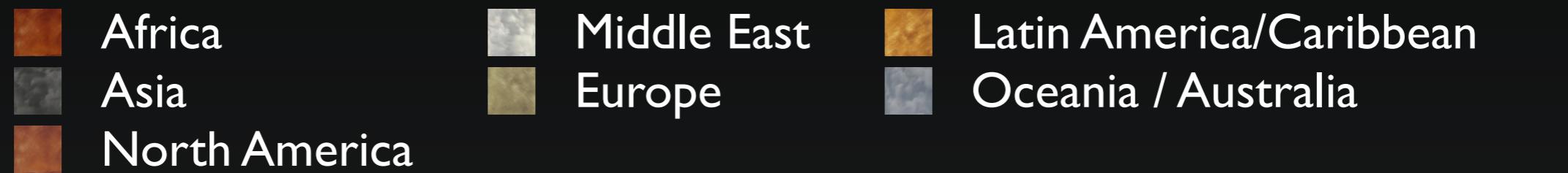
World > USA



“World Internet User Distribution by World Region”
Source: Internet World Stats (12/31/2009)



$\partial_{\text{World}} > \partial_{\text{USA}}$



“World Internet Usage Growth by World Region”
Source: Internet World Stats (12/31/2009)

Mind Expanding

*Software localization is
about making the user
feel at home*

Locale

- Primary Language
- Orthography / Script
- Number Format
- Currency
- Date Format
- Calendar
- Measurement Units

Differences Between Locales

	USA	Japan	France
Language	Hello	こんにちは	Bonjour
Number	1,234.56	1,234.56	1 234,56
Currency	\$100.00	一万円	100,00 €
Date	Tuesday, July 27, 2010	火曜日7月2010年27	mardi 27 juillet 2010
Time	7:30 PM	19:30	19:30

I18n API

I18n.locale

```
I18n.locale = :en
```

REST

example.com?locale=ja

example.jp

```
# Get locale from top-level domain or
# return nil if such locale is not available
# You have to put something like:
#   127.0.0.1 example.com
#   127.0.0.1 example.it
# in your /etc/hosts file to try this out locally

def extract_locale_from_tld
  request.host.split('.').last
end
```

ja.example.com

```
# Get locale code from request subdomain
# (like http://it.application.local:3000)
# You have to put something like:
#   127.0.0.1 gr.example.local
# in your /etc/hosts file to try this out locally

def extract_locale_from_subdomain
  request.subdomains.first
end
```

api.example.com

Accept-Language: ja;

```
# Get locale from "Accept-Language" HTTP Header
# or return nil if such locale is not available
#   example: Accept-Language: da, en-gb;q=0.8, en;q=0.7

def extract_locale_from_accept_language_header
  request.env['HTTP_ACCEPT_LANGUAGE'].scan(/^[a-z]{2}/).first
end
```

example.com?locale=ja

ja.example.com



example.jp

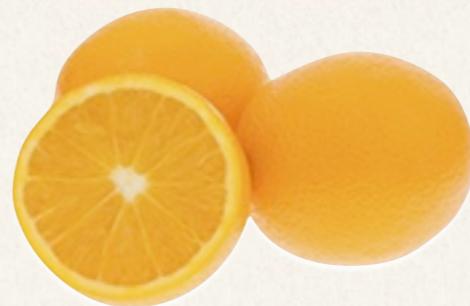
Accept-Language: ja;

```
class ApplicationController < ActionController::Base
  before_filter :set_locale

  def set_locale
    extracted_locale = params[:locale] ||
      extract_locale_from_subdomain ||
      extract_locale_from_tld ||
      extract_locale_from_accept_language_header

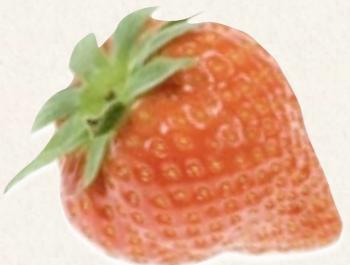
    I18n.locale = (I18n::available_locales.include?
      extracted_locale.to_sym) ? extracted_locale :
      I18n.default_locale
  end
end
```

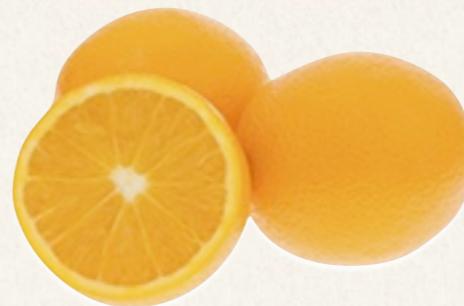
Demo



ZELIG ST. GROCERIES

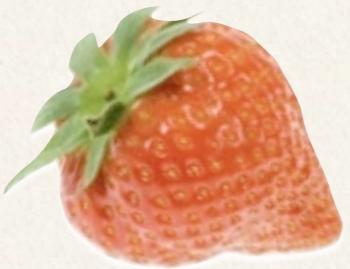
fine organic produce since 1987





ジーリグ通り食料品店

新鮮な果物や野菜を1987年以来



I18n.translate

I18n.translate

<h1>Zelig St. Groceries</h1>
<h2>Fine Organic Produce Since 1987</h2>

```
app/  
  config/  
    locales/  
      en.yml
```

```
en:  
  store:  
    title: Zelig St. Groceries  
    tagline: Fine Organic Produce Since 1987
```

```
<h1><%= t 'store.title' %></h1>
<h2><%= t 'store.tagline' %></h2>
```

Demo

Open 08:00 AM – 06:00 PM Every Day

オープン 08:00 から 18:00まで毎日

I18n. localize

I18n. localize

```
<span id="opening-hours">  
  Open 8:00 AM – 6:00 PM Everyday  
</span>
```

```
<span id="opening-hours">
  <%= t "store.open" %>
  8:00 AM – 6:00 PM
  <%= t "time.everyday" %>
</span>
```

```
<span id="opening-hours">
  <%= t "store.open" %>
  <time id="opens-at">
    <%= l Time.now.beginning_of_day + 8.hours,
      :format => :time %>
  </time>
  <%= t "time.from-to" %>
  <time id="closes-at">
    <%= l Time.now.beginning_of_day + 18.hours,
      :format => :time %>
  </time>
  <%= t "time.everyday" %>
</span>
```

Demo

Gems & Plugins



Apple



Apple

:en



林檎

:ja



Pommes

:fr

Globalize2

```
class Product < ActiveRecord::Base
  translates :title
end
```

easy_globalize2_accessors

```
product.title_fr = "Pommes"
```

```
class Product < ActiveRecord::Base
  translates :title
  globalize_accessors :en, :fr, :ja
end
```



\$0.79 / lb



\$0.79 / lb
:en



0,36 \$ / kg
:fr

Quantity

```
def localized_unit_price(unit_price, unit)
  quantity = Quantity.new(1, Quantity::Unit.for(unit.to_sym))
  factor = case quantity.unit.dimension.name
    when :mass
      case I18n.locale
      when :en
        quantity.convert(unit = :lb)
      else
        quantity.convert(unit = :kg)
      end
    else
      quantity
    end

  price = factor.value * unit_price

rescue
  price = unit_price

ensure
  case I18n.locale
  when :ja
    return "#{'I18n.t ['units', unit].join('.')}  
#{number_to_currency(price)}"
  else
    return "#{'number_to_currency(price)} /  
#{'I18n.t ['units', unit].join('.')}"
  end
end
end
```

Demo

Pro Tips

Mind Your
Charsets

```
<meta charset="UTF-8">
```

```
<html lang="en">
```

Hire a Localizer

Thanks!

ありがとうございます！

Merci Beaucoup!

gowalla.com/jobs

i18n and L10n with Rails 3

Mattt Thompson (@matt)
Hacker, Gowalla

Austin on Rails, July 2010