



Building Mobile (Smartphone) Apps with Ruby & HTML

An introduction to Rhodes

Dasharatham Bitla (Dash)

dash@bitlasoft.com | <http://mobilog.bitlasoft.com>

www.BitlaSoft.com

Smartphones Market

- Smartphones sales are exploding
- Five major smartphone operating systems: iPhone, BlackBerry, Windows Mobile, Symbian, Android
- “native apps” have are winning the day with more than a billion iPhone app downloads on a platform with great Safari browser installed
- Other App Stores are picking up as well ...



The Apple App Store Economy

(based on Dec 09 data)

- 1,35,000+ Apps
- 28,000+ developers
- Submit your app in less than a weeks time
- 58+ million App Store users
- ~300 million downloads a month
- \$250+ million in revenues
 - \$75 million (30%) to Apple
 - \$175 million (70%) to developers

<http://gigaom.com/2010/01/12/the-apple-app-store-economy/>



Why Develop Mobile Apps?

- No/Low upfront Investment
- Can do in your spare time
- Your Imagination is the limit – build anything
- Entrepreneurial Spirit
- Its Fun and good Money
- Submit directly to App Store – no middle man and no need of Telecom Operator Approval
- Promote via Twitter, Facebook, Blog, App review sites, You Tube etc



Complexities ...



Symbian
C++

BlackBerry
JAVA

Apple iPhone
OBJECTIVE C

Android
JAVA

Windows Mobile
C#



**But ...how do you write an app
for all popular devices?**

A Better Way

Using a "smartphone app framework"

Rhodes from Rhomobile



Rhodes

Develop Using



RUBY

HTML

CSS

JavaScript

Style it with



iUI



Develop On



RhoHub™

Write Once
Run Everywhere



What is Rhodes?

- “microframework” for building locally executing natively optimized mobile apps
- the only framework that supports all smartphones
- the only framework that is Model View Controller
- contains first mobile Ruby
- only framework that supports sync
- only framework with a hosted development environment
- **Rhodes** borrows from Rails but not **Rails**



Why Rhodes?

- “mobilize enterprise apps... cost-effectively with a great user experience”
- Provide the high level productivity and portability of web programming. Build the app quickly in HTML and Ruby with rhodes gem and generators
- Device optimization and offline capability of “native mobile apps”
- Best for Data Driven Applications
- Open source for rapid adoption by developers
- Work with synced local data, Includes Offline Storage and connected Sync
- Exploit device capabilities (GPS, PIM data, camera, SMS)
- Available open source: <http://www.github.com/rhobile>

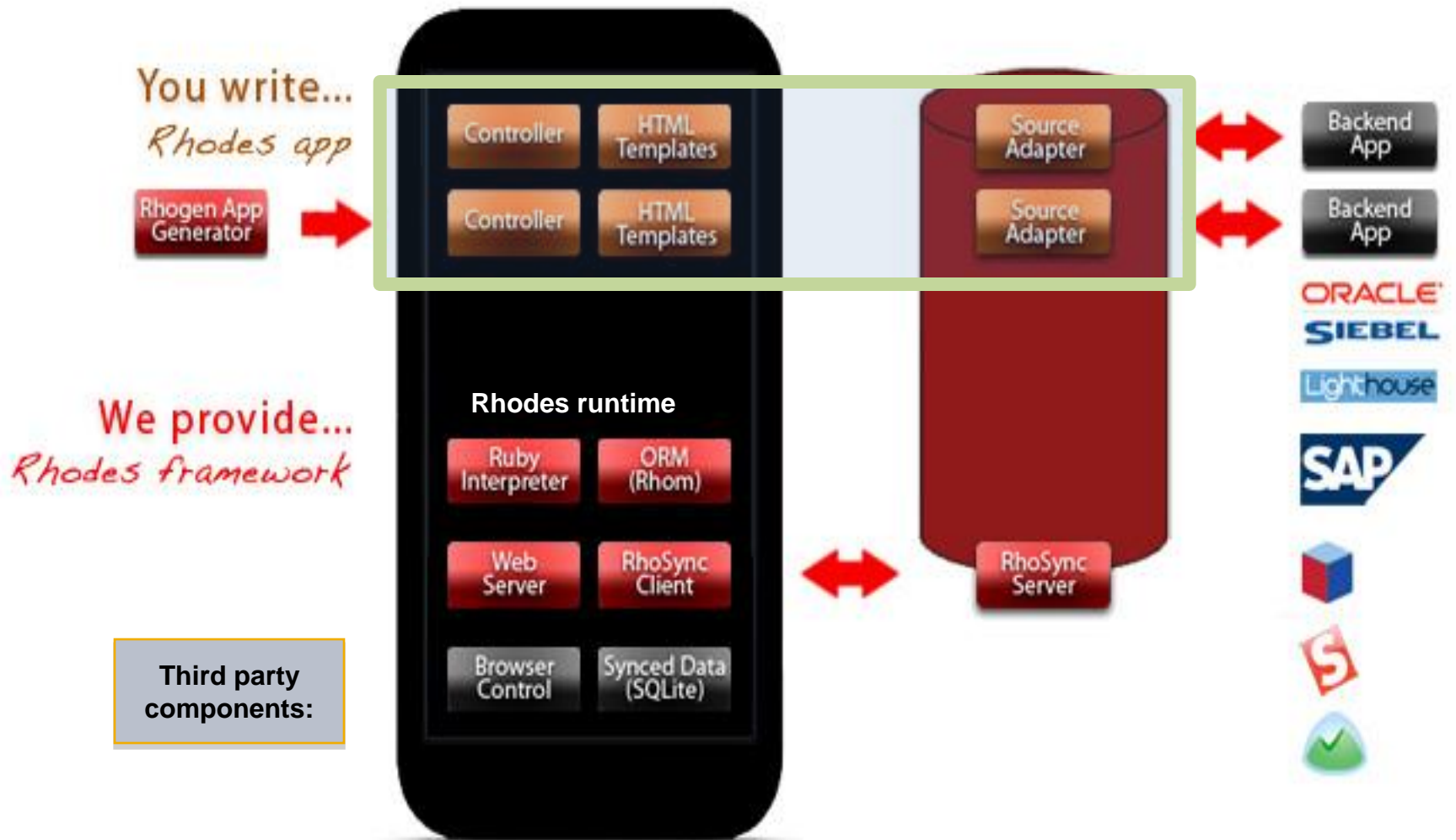


Device Capabilities / Native UI Elements

Capability	iPhone	Windows Mobile	BlackBerry	Symbian	Android
GeoLocation	0.3	0.3	0.3	1.1	1.0
PIM Contacts	0.3	0.3	0.3	1.0	1.0
Camera	1.0	1.0	1.0	1.1	1.0
Date/Time picker	1.2.2	2.0	1.2	2.1	1.2
Native Menu/Tab Bar	1.2.2	2.0	1.2	2.1	1.5
Audio / Video capture	2.0	2.0	2.0	2.1	2.0
Bluetooth	2.0	2.0	2.0	2.1	2.0
Push / SMS	1.2	2.0	1.2	2.1	2.0
Landscape	2.0	2.0	2.0	2.1	2.0
Native Maps	1.4	2.0	1.4	2.0	1.5
Alerts / Audio File Playback	1.2	1.5	1.2	2.0	1.2



Rhodes Architecture



Installing Rhodes

Release Status – 1.4.2 Stable, 1.5 unstable

<http://github.com/rhobile/rhodes>

```
D:\BitlaSoft>ruby -v
```

```
ruby 1.8.6 / ruby 1.8.7
```

```
D:\BitlaSoft>gem install rhodes
```

```
D:\BitlaSoft>gem list rhodes
```

```
rhodes (1.4.2)
```



Installing Device SDKs

iPhone (Need Mac for testing on Simulator)

<http://developer.apple.com/iphone/>

Android

<http://developer.android.com/sdk/index.html>

BlackBerry

<https://www.blackberry.com/Downloads/entry.do?code=060AD92489947D410D897474079C1477>



Setting up Rhodes

D:\BitlaSoft>rhodes-setup

```
D:\BitlaSoft>rhodes-setup
```

```
We will ask you a few questions below about your dev environment.
```

```
JDK path (required) (D:/s/Java/jdk1.6.0_14):
```

```
Android 1.5 SDK path (blank to skip) (D:/work/mobile-apps/android-sdk-windows):
```

```
Android NDK path (blank to skip) (): D:\work\mobile-apps\android-ndk-r3
```

```
Windows Mobile 6 SDK CabWiz (blank to skip) ():
```

```
BlackBerry JDE 4.6 (blank to skip) (C:/Program Files/Research In Motion/BlackBerry JDE 4.6.0):
```

```
BlackBerry JDE 4.6 MDS (blank to skip) (C:/Program Files/Research In Motion/BlackBerry JDE 4.6.0/MDS):
```

```
BlackBerry JDE 4.2 (blank to skip) ():
```

```
BlackBerry JDE 4.2 MDS (blank to skip) ():
```

```
If you want to build with other BlackBerry SDK versions edit: D:/s/InstantRails/ruby/lib/ruby/gems/1.8/rhobuild.yml
```

```
D:\BitlaSoft>
```



Now
Lets Build an **Address Book** App
for
Android and BlackBerry



Generate an AddressBook App

```
D:\BitlaSoft>rhogen app address_book
```

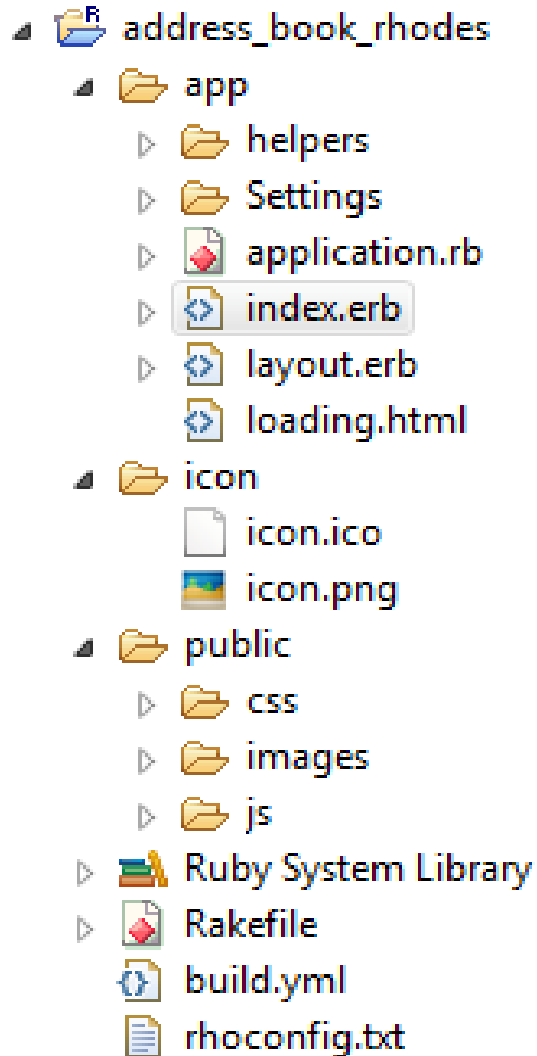
```
Generating with app generator:
```

```
←[32m [ADDED]←[0m address_book/rhoconfig.txt  
←[32m [ADDED]←[0m address_book/build.yml  
←[32m [ADDED]←[0m address_book/app/application.rb  
←[32m [ADDED]←[0m address_book/app/index.erb  
←[32m [ADDED]←[0m address_book/app/layout.erb  
←[32m [ADDED]←[0m address_book/app/loading.html  
←[32m [ADDED]←[0m address_book/Rakefile  
←[32m [ADDED]←[0m address_book/app/helpers  
←[32m [ADDED]←[0m address_book/icon  
←[32m [ADDED]←[0m address_book/app/Settings  
←[32m [ADDED]←[0m address_book/public
```

```
D:\BitlaSoft>
```



Rhodes App Code Structure



- **Build configuration**
- **App Configuration**
- **Device specific CSS**
- **App Icon**
- **Rails like App structure**
- **Easy build/run rake scripts**
- **Layouts**



Generate Contact Model

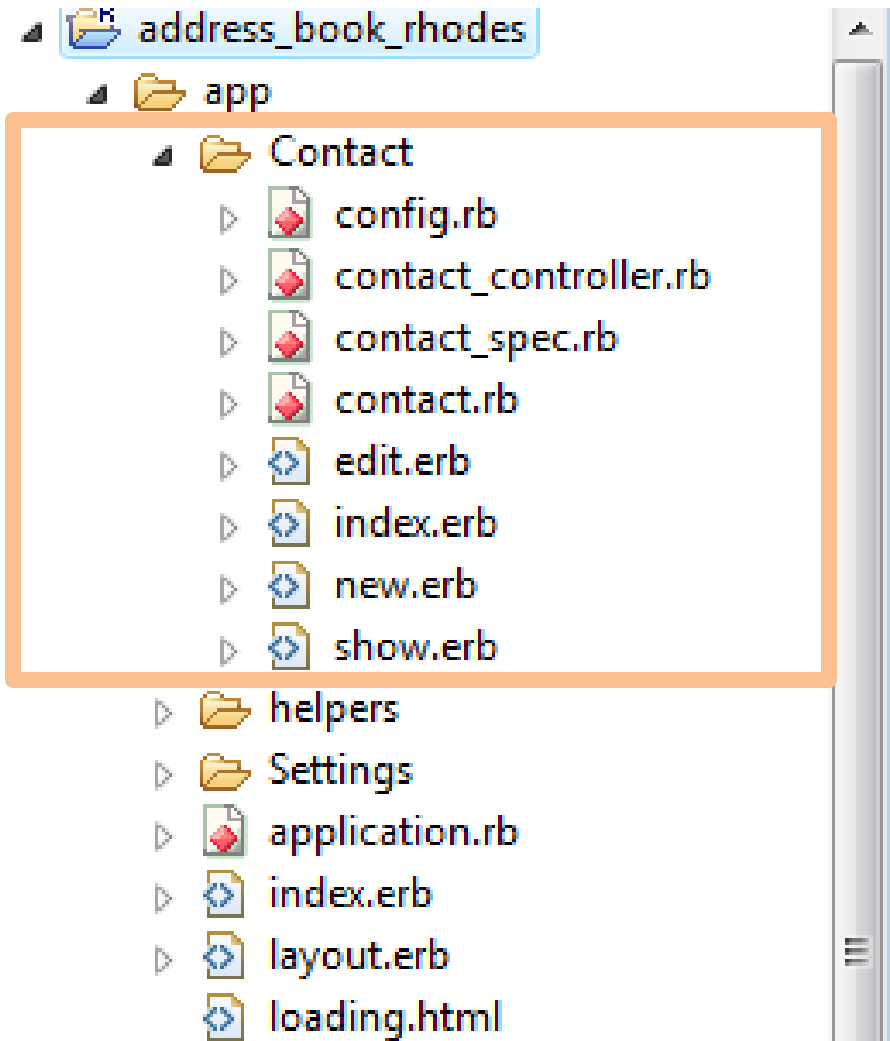
```
D:\BitlaSoft\address_book>rhogen model Contact  
name,phone,email,company,address
```

Generating with model generator:

```
←[32m [ADDED]←[0m app/Contact/config.rb  
←[32m [ADDED]←[0m app/Contact/index.erb  
←[32m [ADDED]←[0m app/Contact/edit.erb  
←[32m [ADDED]←[0m app/Contact/new.erb  
←[32m [ADDED]←[0m app/Contact/show.erb  
←[32m [ADDED]←[0m app/Contact/contact_controller.rb  
←[32m [ADDED]←[0m app/Contact/contact.rb  
←[32m [ADDED]←[0m app/Contact/contact_spec.rb
```



Rhodes App Model Code Structure



- **MVC scaffold generation for Model**
- **Basic cross platform CRUD scaffolding**
- **RhoSync Hooks**



Controller Code

app/Contact/contact_controller.rb

```
1 require 'rails/controllers'
2
3 class ContactController < Rails::Controller
4
5   # GET /Contact
6   def index
7     @contacts = Contact.find(:all)
8     render
9   end
10
11  # GET /Contact/{id}
12  def show
13    @contact = Contact.find(@params['id'])
14    if @contact
15      render :action => :show
16    else
17      redirect :action => :index
18    end
19  end
20
21  # GET /Contact/new
22  def new
23    @contact = Contact.new
24    render :action => :new
25  end
26
27  # GET /Contact/{id}/edit
28  def edit
29    @contact = Contact.find(@params['id'])
30    if @contact
31      render :action => :edit
32    else
33      redirect :action => :index
34    end
35  end
36
37  # POST /Contact/create
38  def create
39    @contact = Contact.new(@params['contact'])
40    @contact.save
41    redirect :action => :index
42  end
43
44  # POST /Contact/{id}/update
45  def update
46    @contact = Contact.find(@params['id'])
47    @contact.update_attributes(@params['contact']) if @contact
48    redirect :action => :index
49  end
50
51  # POST /Contact/{id}/delete
52  def delete
53    @contact = Contact.find(@params['id'])
54    @contact.destroy if @contact
55    redirect :action => :index
56  end
57 end
```

Scaffold Methods

- Index
- Show
- New
- Edit
- Create
- Update
- Delete

View/ UI ERB code

app/Contact/index.erb

```
1 <div class="toolbar">
2   <h1 id="pageTitle">
3     Contacts
4   </h1>
5 </div>
6 <h2><%= @contacts.size %> Contacts</h2>
7 <ul id="contacts" title="Contacts" selected="true">
8   <a class="button right_button" href="Contact/new" target="_self">New</a>
9   <%= @contacts.each do |x| %>
10
11     <li><%= link_to "#{x.name} ", :action => :show, :id => x.object %> <%= x.phone %>
12       <br/>
13       <%= x.company %>
14     </li>
15
16   <%= end %>
17 </ul>
```



Set Start Page – rhoconfig.txt

Startup page for your application

```
start_path = '/app/Contact'
```

Path to the options page (in this case handled by javascript)

```
options_path = '/app/Settings'
```

Location of bundle url (i.e. from rhohub.com)

```
rhobundle_zip_url = ''
```

Optional password to access bundle (usually not required)

```
rhobundle_zip_pwd = nil
```

Rhodes runtime properties

```
MinSeverity = 1
```

```
LogToOutput = 1
```

```
LogCategories = *
```

```
ExcludeLogCategories =
```

```
KeepTrackOfLastVisitedPage = 0
```

```
LastVisitedPage = ''
```

Sync server url. This must be the full path to your sources.

For example: <http://rhomobile.rhohub.com/apps/SugarCRM/sources/>

```
syncserver = ''
```



Build and Launch the App

For iPhone

```
$ rake run:iphone
```

For Android

```
D:\BitlaSoft> rake run:android
```

For BlackBerry

```
D:\BitlaSoft> rake run:bb
```



Rhodes Rake Tasks

```
D:\BitlaSoft\address_book>rake -T
(in D:/BitlaSoft/address_book)
cd D:/s/InstantRails/ruby/lib/ruby/gems/1.8/gems/rhodes-1.5.0
rake clean:android          # Clean Android
rake clean:bb               # Clean bb
rake clean:iphone           # Clean iphone
rake clean:win32            # Clean win32
rake clean:wm               # Clean wm
rake clobber_rdoc           # Remove rdoc products
rake config:checkbb         # Check local blackberry configuration
rake device:android:debug   # Build debug self signed for device
rake device:android:production # Build production signed for device
rake device:bb:debug        # Build debug for device
rake device:bb:production   # Build production for device
rake device:iphone:production # Builds and signs iphone for production
rake device:wm:production   # Build production for device or emulator
rake run:android            # build and launch emulator
rake run:bb                 # Builds everything, loads and starts bb si...
rake run:bbdev              # Same as run:bb, but only supports one app...
rake run:iphone             # Builds everything, launches iphone simulator
rake run:win32              # Run win32
```



Style it with - iWebKit

Native Looking iPhone UI in HTML and CSS



Style it with - jQTouch

**Native Feeling iPhone UI & other WebKit browser Uis
With Native Animations Support**



Other Rhomobile Components

RhoSync
&
RhoHub

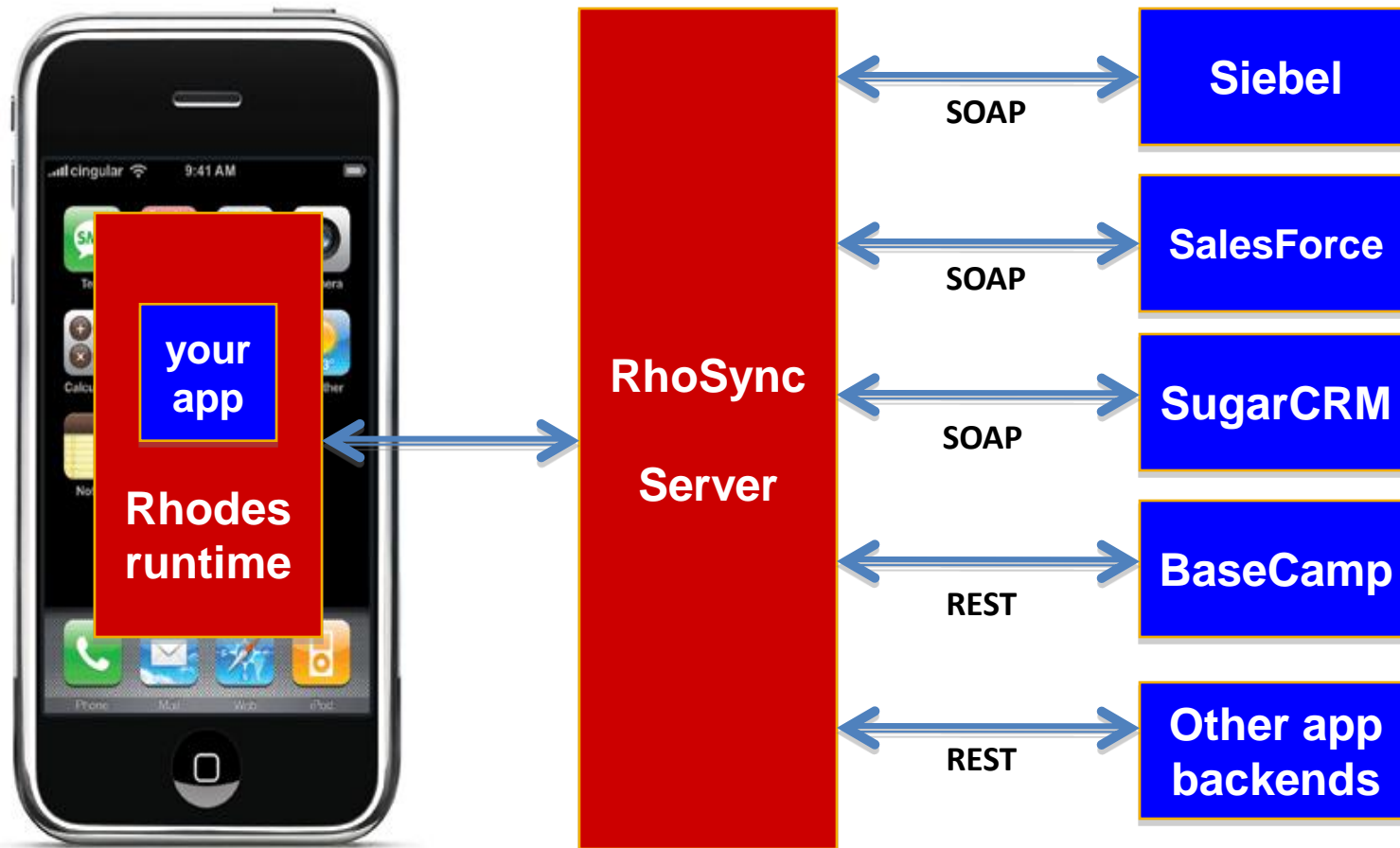


RhoSync

- **Connect mobile apps with the enterprise**
- **Offline Use**
- **Easy Sync Adapter Development**
- **First Web Services Sync Server**
- **First Sync Server To Use Smartphone Push**
- **Simplifies Smartphone Data Access**
- **RhoSync is a Rails App**



RhoSync Architecture



RhoSync – Source Adapter

D:\BitlaSoft\address_book>**rho**gen source contacts

```
1 class Contacts < SourceAdapter
2   def initialize(source, credential)
3     super(source, credential)
4   end
5
6   def login
7     #TODO: Write some code here
8     # use the variable @source.login and @source.password
9     raise "Please provide some code to perform an authenticated login to the backend application"
10  end
11
12  def query
13    # TODO: write some code here, Query your backend for objects. Put into some variable that is used in sync metho
14    raise "Please provide some code to read records from the backend application"
15  end
16
17  def sync
18    # TODO: write code here that converts the data you got back from query into an @result object
19    # where @result is a hash of hashes, each array item representing an object
20    # for example: @result={1=>{"name">"Acme", "industry">"Electronics"}, 2=>{"name">"Best", "industry">"Softw
21    # if you have such a hash of hashes, then you can just call "super" as shown below
22    super # this creates object value triples from an @result variable that contains a hash of hashes
23  end
24
25  def create(name_value_list, blob=nil)
26    #TODO: write some code here
27    # the backend application will provide the object hash key and corresponding value
28    raise "Please provide some code to create a single object in the backend application using the hash values in n
29  end
30
31  def update(name_value_list)
32    #TODO: write some code here
33    # be sure to have a hash key and value for "object"
34    raise "Please provide some code to update a single object in the backend application using the hash values in n
35  end
36
37  def delete(name_value_list)
38    #TODO: write some code here if applicable
39    # be sure to have a hash key and value for "object"
40    # for now, we'll say that its OK to not have a delete operation
41    # raise "Please provide some code to delete a single object in the backend application using the hash values in
42  end
43
44  def logoff
45    #TODO: write some code here if applicable
46    # no need to do a raise here
47  end
48 end
```

Sync Methods

- Login
- Query
- Sync
- Create
- Update
- Delete
- logoff

RhoHub

- First and only “Development as a Service for mobile”
- Hosted IDE app development (edit online a la Heroku)
- Hosted Central build (don't need to install all development environment) for all SmartPhones online
- Hosted provisioning (user downloading from URL)
- Hosted runtime (sync server)
- Write apps without installing SDKs locally
- Diagnostic Tools
- Lot more ...





AddressBook

[Overview](#)[Settings](#)[Collaborators](#)[Editor](#)[Builds](#)

Select an App

[Console](#) [Clear Console](#) Server Time: Mar 17, 2010, 11:22[Download Rhodes Desktop Client](#)[Visit the Rhomobile Wiki](#)

Client

Ser Find in project

[Find](#) ist[Reset Database](#)[Save](#)[Quick Links](#)[open all](#) | [close all](#) |
[refresh](#)

- AddressBook
 - app
 - Settings
 - cont
 - index
 - login
 - rese
 - wait
 - helpers
 - appli
 - applicati
 - index.erl
 - layout.er
 - loading.t
 - icon
 - icon.ico
 - icon.png
 - public
 - css
 - base

* controller.rb

Delete

Download

Rename

```
1 require 'rho'
2 require 'rho/rhocontroller'
3 require 'rho/rhoerror'
4
5 class SettingsController < Rho::RhoController
6
7   def index
8     @msg = @params['msg']
9     render
10  end
11
12  def login
13    @msg = @params['msg']
14    render :action => :login
15  end
16
17  def login_callback
18    errCode = @params['error_code'].to_i
19    if errCode == 0
20      # run sync if we were successful
21      WebView.navigate Rho::RhoConfig.start_path
22      SyncEngine.dosync
23    else
24      if errCode == Rho::RhoError::ERR_CUSTOMSYNCSEVER
25        @msg = @params['error_message']
26      end
27    end
28  end
29 end
```


Questions?

Dasharatham Bitla (Dash)
Founder & CEO, [BitlaSoft](http://BitlaSoft.com)
dash@bitlasoft.com

<http://mobilog.bitlasoft.com>
www.BitlaSoft.com

ticket  **Simply** *Simply*  **BUS**.com
Simply Manage Travels

mobee.in
Simplified Email Messaging Solution

