# Sterling Mobile Store Channel

## Customization Guide

**Release 1.0**

July 2010

**Sterling Commerce**

*An IBM Company*

**WARRANTY DISCLAIMER**

The Framework and the related documentation are licensed either "AS IS", as described in the Eclipse Public License v 1.0. NO WARRANTIES ARES EXPRESSED AND NONE SHALL BE IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE OR FOR A PARTICULAR PURPOSE. The applicable Sterling Commerce entity reserves the right to revise this publication from time to time and to make changes in the content hereof without the obligation to notify any person or entity of such revisions or changes.

**GLYPHISH ICONS**

The Framework is distributed with or on the same storage media as certain of the Glyphish icons created by Joseph Wain (see http://penandthink.com). ("Glyphish Icons"). The Glyphish Icons are subject to the Creative Commons Attribution License v.3.0, a copy of which is located in the readme file, <client_source>/SterlingMobileStore/Sample_App/Library/sms_framework_readme_v1.DOC and the information set forth below. This information and the terms and conditions apply only to the Glyphish Icons and not the Framework or other Third Party Software.

**Information:**

Created by Joseph Wain (see http://penandthink.com) at and probably downloaded from http://glyphish.com.This work is licensed under the Creative Commons Attribution 3.0 United States License. To view a copy of this license, visit http://creativecommons.org/licenses/by/3.0/us/ or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

You are free to share it and to remix it remix under the following conditions:

✦ You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

✦ For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to http://creativecommons.org/licenses/by/3.0/us/

✦ Any of the above conditions can be waived if you get permission from the copyright holder (send me an email!).

✦ Apart from the remix rights granted under this license, nothing in this license impairs or restricts the author's moral rights.

**ATTRIBUTION** -- a note reading "icons by Joseph Wain / glyphish.com" or similar, plus a link back to glyphish.com from your app's website, is the preferred form of attribution. Also acceptable would be, like, a link from within your iPhone application, or from the iTunes store page, but those aren't as useful to other people. If none of these work for you, please contact hello@glyphish.com and we can work something out.

**USE WITHOUT ATTRIBUTION** -- If attribution is not possible, workable or desirable for your application, contact hello@glyphish.com for commercial non-attributed licensing terms.

## NAKED SOFTWARE

The Framework is distributed on the same storage media as the Naked Software code. The Naked Software is a free software product which is distributed subject to the following license (which applies only to the Naked Software and not to the Framework or any other Third Party Software):

Copyright (c) 2002 Naked Software. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

✦ Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

✦ Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

✦ Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## STIG BRAUTASET

The Framework is distributed on the same storage media as the Stig Brautaset Software. The Stig Brautaset Software is a free software product which is distributed subject to the following license (which applies only to the Stig Brautaset Software and not to the Framework or any other Third Party Software):

Copyright (C) 2007-2009 Stig Brautaset. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

✦ Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- ✦ Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- ✦ Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**WILLIAM WOODY**

The Framework is distributed on the same storage media as the William Woody Software. The William Woody Software is a free software product which is distributed subject to the following license (which applies only to the William Woody Software and not to the Framework or any other Third Party Software):

Copyright 2008 William Woody, All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Neither the name of Chaos In Motion nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Contact William Woody at woody@alumni.caltech.edu or at woody@chaosinmotion.com. Chaos In Motion is at http://www.chaosinmotion.com.

# Contents

# Introduction to Sterling Mobile Store Channel Customization

Sterling Mobile Store Channel enables enterprises to build the client application that can be customized to adapt to changing business requirements. As part of Sterling Mobile Store Channel, a sample application is provided to you to serve as a reference to customize the client application based on your business requirements.

The Sterling Mobile Store Channel client application can be customized as follows:

✦ Replacing icons or images.

✦ Modifying or replacing the existing screens.

✦ Extending the application behavior.

✦ Introducing new screens.

## Intended Audience for the *Sterling Mobile Store Channel Customization Guide*

This guide is intended for use by individuals who are responsible for customizing the Sterling Mobile Store Channel solution.

# Prerequisites for Customizing Sterling Mobile Store Channel

The Sterling Mobile Store Channel application is an iPhone and iPod touch client application. To customize the application, you must be comfortable using an Apple MacBook. It is also important that you are familiar with Objective C (the language used for coding) and Xcode (an Integrated Development Environment (IDE) used for development).

You can register at the iPhone Dev Center (http://developer.apple.com) to access the iPhone Software Development Kit (SDK).

# Sterling Mobile Store Channel Architecture

The following figure illustrates the Sterling Mobile Store Channel architecture:
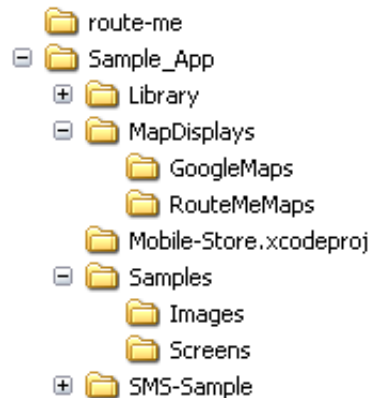


✦ `route-me` – The `route-me` folder is an empty folder that serves as a placeholder for the Route-Me map library.

✦ `Sample_app` folder – The `Sample_app` folder contains the Mobile-Store Xcode project and the following resources:

   ◆ Library - Contains the core classes and view controllers.

   ◆ MapDisplays – Contains the implementations for Route-Me Maps and Google Maps.

   **Note:** For more information about license issues pertaining to Route-Me Maps, refer to the appropriate documentation provided with the corresponding mapping sources.

   ◆ Samples – Contains the images and screens used by the sample Sterling Mobile Store Channel application.

   ◆ SMS-Sample – Contains the sample Sterling Mobile Store Channel application resources.

# Launching the Sample Sterling Mobile Store Channel Application

To launch the sample Sterling Mobile Store Channel application, perform the following tasks:

1. Download and extract the contents of route-me (version 0.5) map library to the empty `route-me` folder.

2. Open the Mobile-Store project in Xcode.

3. From the Build menu, select Build and Run.

The sample Sterling Mobile Store Channel application is launched.

# Customizing the Sample Sterling Mobile Store Channel Application

You can customize the sample Sterling Mobile Store Channel application to suit your business needs. The following topics describe how to perform some basic customizations on the sample Sterling Mobile Store Channel application.

**Modify the Product Name**

To modify the product name:

1. Open the Mobile-Store project in Xcode.

2. Double-click the SMS-Sample target of the Mobile-Store project.

3. Navigate to the Build tab and modify the Product Name property.

**Modify the Properties**

To modify the Sterling Mobile Store Channel application properties such as Bundle identifier, Bundle version, Icon file, and so on:

1. Open the Mobile-Store project in Xcode.

2. Select the Mobile_Store-info.plist and modify the appropriate properties.

**Modify the Settings**

To modify the Sterling Mobile Store Channel application settings:

1. Open the Mobile-Store project in Xcode.

2. Select the Settings.bundle and modify the appropriate settings.

For more information about the application settings, refer to the *iPhone Application Programming Guide* that is available on the Apple Web site.

**Display Maps Using Route-Me**

To display maps using Route-Me, refer to the topic "*Using Maps to Locate a Store* ".

# Customizing Sterling Mobile Store Channel

Following is the list of Sterling Mobile Store Channel customizations that can be performed:

✦ Replacing icons, or images, or both
✦ Modifying the existing screens
✦ Extending application behavior using user exits

## Replacing an Icon or an Image

Following are the methods that can be used for replacing icons or images:

- ✦ To replace icons or images requested from a framework class, replace an existing image stored under the `Samples/Images` directory with a new image.
- ✦ To replace the icons or images used within the Interface Builder, open the interface builder and select the appropriate image from the **Image** drop-down list.

   **Note:** The **Image** drop-down list contains only those images that have been added to `The Project`.

## Limitations

None.

# Modifying or Replacing an User Interface

To modify or replace an user interface, copy the screen files that are present in the `Samples/Screens` directory, and add these files to `The Project`. Perform the required modifications using the Interface Builder, and save the file.

## Limitations

This approach is applicable only to the screens that are present in the framework. Therefore, the mapping entries that are provided in the `Nib-Name-Map.plist` file are the only valid entries that can be used for mapping.

# Extending the Sterling Mobile Store Channel Behavior

The following user exits are exposed in the Sterling Mobile Store Channel framework:

- ✦ `kPlatformUtilsUserExit`
- ✦ `kAppUtilsUserExit`
- ✦ `kGeoCodingUserExit`
- ✦ `kMapControllerUserExit`
- ✦ `kShipmentTrackingUserExit`

When the client application, which is built using the Sterling Mobile Store Channel framework is launched, the application delegate must register the user exits with the corresponding handlers. Otherwise, default implementation of methods takes place. For example, the following code snippet registers the platform utils user exit for the self class:

```
[GlobalMemory registerClassForUserExitName:kPlatformUtilsUserExit class:self]
```

## Limitations

You can modify the behavior of only those methods that are exposed as user exits.

# Adding a New Screen in Sterling Mobile Store Channel

If you add a new screen whose functionality is similar to a sample screen present in the `Samples/Screens` directory, and save the new screen with a different name, add the `Nib-Name-Map.plist` file present in the `Samples` directory to `The Project.`

The `Nib-Name-Map.plist` file contains a list of existing screen names without the file extension. By default, the existing screen names are mapped to an empty string (`" "`). Ensure that the new screen is mapped appropriately in the `Nib-Name-Map.plist` file, and added to `The Project.`

# Adding Reviews for an Item

The Reviews screen provided in the sample application does not populate reviews for an item. To display the reviews for an item, extend the Reviews screen by using the `Nib-Name-Map.plist` file, and populate the item reviews from a third-party source.

For more information about extending screens, refer to the topic "Adding a New Screen in Sterling Mobile Store Channel".

# Using Maps to Locate a Store

The Sterling Mobile Store Channel framework exposes the following user exits that enable you to implement maps to locate stores:

✦   `kGeoCodingUserExit`
✦   `kMapControllerUserExit`

The `MapDisplays` folder that is present in the `Sample_app` folder contains an implementation for Route-Me Maps and Google Maps.

To register an implementation to the respective user exits, perform the following tasks within the `applicationDidFinishLaunching` method of the application delegate:

✦   Register the geo coder class to the kGeoCodingUserExit as follows:

   `[GlobalMemory registerClassForUserExitName:kGeoCodingUserExit class:@"<geo coder class name>"];`

   For example, if you are implementing Route-Me Maps, you must replace the *<geo coder class name>* attribute with WSGeoNames, which is present in the `Sample_app/MapDisplays/RouteMeMaps` folder.

- ✦ Register the map controller class to the kMapControllerUserExit as follows:

```
[GlobalMemory registerClassForUserExitName:kMapControllerUserExit
class:@"<map controller>"];
```

For example, if you are implementing Route-Me Maps, you must replace the *<map controller>* attribute with RouteMeMapController, which is present in the `Sample_app/MapDisplays/RouteMeMaps` folder.

**Important:** If you use the RouteMeMapController, add the following code in the `getTileSource` method of the RouteMeViewController.

```
return [[[RMCloudMadeMapSource alloc]
initWithAccessKey:<CloudMadeAccessKey> styleNumber:<CloudMadeStyleKey>]
autorelease];
```

To obtain *<CloudeMadeAccessKey>* and *<CloudMadeStyleKey>*, register online at the CloudMade Web site.

# Payment Card Industry (PCI) Considerations

Payment Card Industry Data Security Standard (PCI-DSS) and Payment Application Data Security Standard (PA-DSS) are standards established by the Payment Card Industry to promote secure payment processing. PA-DSS is a standard for commercially off-the-shelf payment applications, which are defined as applications that capture, process, store, or transmit information pertaining to cards, such as credit cards.

The Sterling Order Management Mobile Framework, which is the server-side pack is not a PCI PA-DSS validated payment application. As a result, you should not transmit Primary Account Numbers (PANs) to the Sterling Order Management Mobile Framework. Otherwise, it will contravene Visa's Payment Application Security Mandate (see http://usa.visa.com/merchants/risk_management/cisp_payment_applications.html) that states "Acquirers must ensure their merchants, VisaNet Processors (VNPs), and agents use only PA-DSS compliant applications". You can use tokenization, which is the process of storing credit card numbers or stored value card numbers in a vault system that associates a token to a securely stored credit card number or stored value card number. Credit card numbers and stored value card numbers should be stored only in a vault system and not in the Sterling Selling and Fulfillment Suite. It is recommended that you use your organization's

vault solution along with Sterling Order Management Mobile Framework to implement tokenization as shown in the following figure:
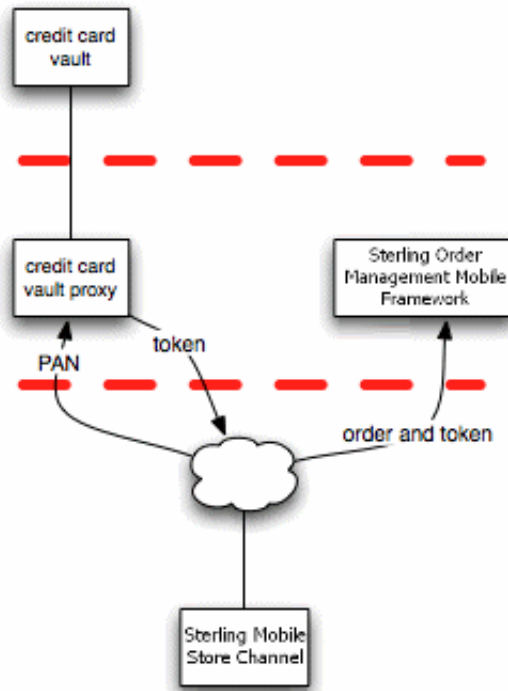


## Developing, Testing, and Distributing an iPhone or iPod touch Application

The iPhone Developer Program enables you to develop, test, and distribute iPhone or iPod touch applications. The program offers a Standard Program that enables you to deploy an iPhone or iPod touch application on the App Store. For more information about enrolling in the Standard Program, refer to the iPhone Developer Program, details of which are available in the Apple Web site: http://developer.apple.com.

# Index