



Developers' Guide

Framework

Version 2.0.3.1

20th January 2012

© 2011 51Degrees.mobi Limited. All rights reserved.

The copyright in and title to the document "Developers' Guide – Framework" belongs to 51Degrees.mobi Limited. No part of it whatsoever may be reproduced in any form without the prior authority of 51Degrees.mobi Limited and/or any original source as appropriate. Any agreed copy or extract must be marked with all proprietary notices which appear on the original and will be subject to the requirement that you will acknowledge on the face of each part of the reproduced material that it belongs to 51Degrees.mobi Limited.

9 Mobile Controls

Many 51Degrees.mobi controls either inherit from standard ASP.NET controls, or implement the same interface. If a control is of this type a link to the MSDN documentation is included at the beginning of each chapter.

If the 51Degrees.mobi control alters the implementation of properties described in the MSDN documentation the following chapters will explain the differences. New methods are described separately.

If the control is new to 51Degrees.mobi the control is documented in full within this guide.

9.1 Common Issues

The following properties may exhibit unpredictable behaviour if used on a mobile web site accessed on some mobile devices. This chapter describes issues and presents possible work arounds.

9.1.1 PostbackUrl Property

The PostbackUrl property of the LinkButton and Button controls cannot be relied upon to work in all situations when used with mobile devices. If the property contains a value JavaScript is needed to alter the Url post back data is sent to when these buttons are selected. Some mobile devices will either not implement JavaScript, or the implementation is not complete, and the JavaScript will not execute as intended.

We strongly suggest the PostbackUrl property is not used within mobile web applications. It is available if your application can be certain a full JavaScript implementation will be present on the target mobile devices.

A work around could be to listen for the Command event of the control and then use the [Response.Redirect](#) or [HttpContext.RewritePath](#) to redirect processing to another page.

9.1.2 Security Permissions

The resource paths I.axd and P.axd need to be accessible to the web site in all situations to manage the generation of includes and images. If a blanket access permission is applied to the web site these resources will not be available until after the user has been authenticated. For this reason the following configuration should be included in the web.config file to enable access to these resources.

Web.config

```
<location path="I.axd">
  <system.web>
    <authorization>
      <allow users="?" />
    </authorization>
  </system.web>
</location>
<location path="P.axd">
  <system.web>
    <authorization>
      <allow users="?" />
    </authorization>
  </system.web>
</location>
```

9.2 AdRotator

See [MSDN Documentation](#) for base details.

The AdRotator control has been extended to enable multiple images sources to be provided without changing the structure of the XML file used to provide advert data. The AltImages collection of the control can optionally be configured to provide alternative source images to the ones defined in the XML file.

The Image file name provided in the XML file should be prefixed AdX_, where X is the numeric number of the advert. Any image source defined in the AltImages collection with the same prefix before the underscore will be associated with the advert. Consider the following XML file where two adverts are defined with the ImageUrl prefix Ad1 and Ad2.

XML

```
<Advertisements>
  <Ad>
    <ImageUrl>~/Images/AdRotator/Ad1_48.png</ImageUrl>
    <AlternateText>This is your first advert.</AlternateText>
    <Impressions>10</Impressions>
    <NavigateUrl>~/Default.aspx</NavigateUrl>
  </Ad>
  <Ad>
    <ImageUrl>~/Images/AdRotator/Ad2_48.png</ImageUrl>
    <AlternateText>This is your second advert.</AlternateText>
    <Impressions>10</Impressions>
    <NavigateUrl>~/Default.aspx</NavigateUrl>
  </Ad>
</Advertisements/>
```

The corresponding AltImages collection requires the ImageUrl to use the same prefix. The following ASP.NET example shows how the controls collection should be configured.

ASP.NET

```
<mob:AdRotator ID="AdRotator1" runat="server" AdvertisementFile="Ads.xml"
Target="_blank" StyleID="Style1">
  <!-- Advert 1 Image Sources -->
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad1_48.gif" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad1_72.gif" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad1_128.gif" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad1_48.png" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad1_128.png" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad1_72.png" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad1_48.jpg" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad1_128.jpg" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad1_72.jpg" />
  <!-- Advert 2 Image Sources -->
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad2_48.gif" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad2_72.gif" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad2_128.gif" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad2_48.jpg" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad2_128.jpg" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad2_72.jpg" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad2_48.png" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad2_128.png" />
  <mob:AltImage ImageUrl="~/Images/AdRotator/Ad2_72.png" />
</mob:AdRotator>
```

Other than this minor alteration and the following new or enhanced properties the AdRotator is interface compatible with the control provided by ASP.NET.

Property	Description
<code>AltImages</code>	The list of of different size versions of the same image. Used to improve the quality of the final image rendered on mobile device.
<code>CalculateSizeMode</code>	The method used to determine the size of the optimum size of the image in pixels.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.3 BulletedList

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.4 Button

See [MSDN Documentation](#) for base details.

Property	Description
<code>AutoSize</code>	If set to true and no height information is provided explicitly for the control the height will automatically be set to a size that is appropriate for the mobile device displaying the page. This ensures selectable controls are sized appropriately on touch screen devices. Defaults to true.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of

Property	Description
	the same page.
StyleID	The StyleID of the Style control to use when calculating style properties for the control.

9.5 Calendar

See [MSDN Documentation](#) for base details. See online [tutorial](#) for walk through of control.

Whilst we have maintained interface compatibility with the original calendar control provided by Microsoft we've made some significant enhancements. The main difference between our implementation and the original one is that the control renders in three different ways depending on the capabilities of the mobile device it is being displayed on. The following rendering methods are used.

Mode	Description	Example Screen Shot
Wide Screen	Where a mobile device has a screen wide enough to display a classic calendar in grid view, similar to the one used by the original calendar control, this format will be used. Every interaction with the calendar will result in a post back to the web server.	
Wide Screen – JavaScript & DOM	<p>If the mobile device supports JavaScript and Document Object Model (DOM) a wide screen calendar identical in appearance to the previous mode will be displayed. However there are two important differences.</p> <ol style="list-style-type: none"> 1. Only interactions with the calendar that require a post back to the web server will result in one. For example; the month can be changed without requiring a complete refresh. 2. Where the mobile device supports gestures the month can be changed with a swipe horizontally through any part of the grid. 	
Basic	If the screen width is too narrow to display a calendar in grid view then a drop down list based selected method will be used. Day, month and year can be selected and then added to a list of selected dates. Dates added to the selected list can be removed. The user interface is more cumbersome than a grid but date validation is enforced and the output received by the underlying mobile web application is identical.	

The mode used to display the calendar control is automatically calculated by 51Degrees.mobi and cannot be controlled by the developer.

When displayed on a touch screen phone the calendar control provides an exceptionally simple solution for capturing multiple dates. Consider the following reservation system scenario:

1. Dates within the following 3 months can be selected.
2. Some dates have already been taken and these are not available for selection.
3. Weekends are not available.

The calendar control represents this situation intuitively and clearly enabling the user to choose a date easily. Different elements of the control can be styled to meet the needs to your application using CSS or ASP.NET style attributes and the 51Degrees.mobi style control. See the Styles chapter for more information.

The following tables describe the available properties where they're new or differ in implementation to the standard calendar control.

Property	Description
<code>CaptionStyleID</code>	StyleID to use to apply styling to the Caption section.
<code>ClearButtonStyleID</code>	StyleID to use to apply styling to the Clear button.
<code>ClientID</code>	Overridden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>DayHeaderStyleID</code>	StyleID to use to apply styling to the day header.
<code>DayStyleID</code>	StyleID to use to apply styling to the day cell.
<code>DisableDates</code>	A collection of dates which are not available for selection.
<code>DisableDays</code>	A collection of days of the week which are not available for selection.
<code>DisabledDayStyle</code>	Style to apply to the disabled days of the calendar.
<code>DisabledDayStyleID</code>	The ID of the Style control to use to provide additional style information to the disabled days.
<code>DisabledNextMonthImage</code>	Image sources used to provide image information to the next month disabled button .
<code>DisabledPreviousMonthImage</code>	Image sources used to provide image information to the previous month disabled button .

Property	Description
DoneButtonStyleID	StyleID to use to apply styling to the Done button.
MaxDate	The largest date that can be selected.
MinDate	The smallest date that can be selected.
NextMonthImage	An image to use to navigate to the next month.
NextPrevStyleID	StyleID to use to apply styling to the next and previous elements.
NumberOfSelectableDays	If set a value greater than 1 to indicate the number of days that can be selected by the end user.
OtherMonthDayStyleID	The ID of the Style control to use to provide additional style information to the other months days.
PersistState	If set to true the control will persist its state in the mobile profile. The previous state will be used in subsequent page requests. This property has no effect if MobileProfile is disabled. Defaults to true.
PreviousMonthImage	An image to use to navigate to the previous month.
PrevMonthText	Text to be associated with the navigation button used to move to the previous month.
ProfileDataKey	An optional shared key that can be used across to enable mobile profile data related to the control to be shared with other controls of the same type using the same key.
SelectedDayStyleID	StyleID to use to apply styling to selected dates.
SelectorStyleID	StyleID to use to apply styling to the week and month selectors.
StyleID	The StyleID of the Style control to use when calculating style properties for the control.
TitleStyleID	StyleID to use to apply styling to the title text.
TodayButtonStyleID	StyleID to use to apply styling to the Today button.
TodayDayStyleID	StyleID to use to apply styling to the current date.
WeekendDayStyleID	StyleID to use to apply styling to the weekend labels.

9.6 ChangePassword

See [MSDN Documentation](#) for base details.

Property	Description
CancelButtonStyleID	The ID of the Style control to use to provide additional style

Property	Description
	information to the cancel button.
<code>ChangePasswordButtonStyleID</code>	The ID of the Style control to use to provide additional style information to the change password button.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>ContinueButtonStyleID</code>	The ID of the Style control to use to provide additional style information to the continue button.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>FailureTextStyleID</code>	The ID of the Style control to use to provide additional style information to the hyperlinks
<code>HyperLinkStyleID</code>	The ID of the Style control to use to provide additional style information to the hyperlinks.
<code>InstructionTextStyleID</code>	The ID of the Style control to use to provide additional style information to the instruction text.
<code>LabelStyleID</code>	The ID of the Style control to use to provide additional style information to the labels.
<code>PasswordHintStyleID</code>	The ID of the Style control to use to provide additional style information to the password hint text.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.
<code>SuccessTextStyleID</code>	The ID of the Style control to use to provide additional style information to the success text.
<code>TextBoxStyleID</code>	The ID of the Style control to use to provide additional style information to the textboxes.
<code>ValidatorTextStyleID</code>	The ID of the Style control to use to provide additional style information to the validator text.

9.7 CheckBox

See [MSDN Documentation](#) for base details.

Property	Description
----------	-------------

Property	Description
<code>AutoPostBack</code>	If the mobile device does not support javascript and the EnableAutoPostBackButton property is true a button will appear next to the control to enable the user to manual postback the page.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>DisabledAutoPostBackImage</code>	The list of disabled images to use for the expand button when displayed on non-javascript devices.
<code>EnableAutoPostBackButton</code>	If set to false auto post back will be disabled when displayed on mobile devices that do not support javascript. Note: AutoPost back must not be a critical part of the mobile web applications process.
<code>EnabledAutoPostBackImage</code>	The list of enabled images to use for the autopostback button displayed on non-javascript devices.
<code>PersistState</code>	If set to true the control will persist its state in the mobile profile. The previous state will be used in subsequent page requests. This property has no effect if MobileProfile is disabled. Defaults to true.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.8 CheckBoxList

See [MSDN Documentation](#) for base details.

Property	Description
<code>AutoPostBack</code>	If the mobile device does not support javascript and the EnableAutoPostBackButton property is true a button will appear next to the control to enable the user to manual postback the page.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>DisabledAutoPostBackImage</code>	The list of disabled images to use for the expand button when displayed

Property	Description
	on non-javascript devices.
<code>EnableAutoPostBackButton</code>	If set to false auto post back will be disabled when displayed on mobile devices that do not support javascript. Note: AutoPost back must not be a critical part of the mobile web applications process.
<code>EnabledAutoPostBackImage</code>	The list of enabled images to use for the autopostback button displayed on non-javascript devices.
<code>PersistState</code>	If set to true the control will persist its state in the mobile profile. The previous state will be used in subsequent page requests. This property has no effect if MobileProfile is disabled. Defaults to true.
<code>ProfileDataKey</code>	An optional shared key that can be used across to enable mobile profile data related to the control to be shared with other controls of the same type using the same key.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.9 CompareValidator

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.10 CustomValidator

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.

<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.
----------------------	--

9.11 DataList

See [MSDN Documentation](#) for base details.

Property	Description
<code>AlternateItemStyleID</code>	The ID of the Style control to use to provide additional style information to the alternative item in the items list.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>EditItemStyleID</code>	The ID of the Style control to use to provide additional style information to the selected item for editing item in the items list.
<code>FooterStyleID</code>	The ID of the Style control to use to provide additional style information to the footer.
<code>HeaderStyleID</code>	The ID of the Style control to use to provide additional style information to the header.
<code>ItemStyleID</code>	The ID of the Style control to use to provide additional style information to the item in the items list.
<code>PageIndex</code>	The current active page index.
<code>PageSize</code>	The number of records displayed per page.
<code>PagingMode</code>	Whether the number of records is calculated using a fixed number of rows or screen heights.
<code>SelectedItemStyle</code>	Style to apply to the selected item in the items list.
<code>SeparatorStyleID</code>	The ID of the Style control to use to provide additional style information to the saperartor in the items list.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

Event	Description
<code>PageIndexChanged</code>	Fired after the page index has been changed.

Event	Description
<code>PageIndexChanging</code>	Fired before the page index is changed.

9.12 DropDownList

See [MSDN Documentation](#) for base details.

Property	Description
<code>AutoPostBack</code>	If the mobile device does not support javascript and the EnableAutoPostBackButton property is true a button will appear next to the control to enable the user to manual postback the page.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>DisabledAutoPostBackImage</code>	The list of disabled images to use for the expand button when displayed on non-javascript devices.
<code>EnableAutoPostBackButton</code>	If set to false auto post back will be disabled when displayed on mobile devices that do not support javascript. Note: AutoPost back must not be a critical part of the mobile web applications process.
<code>EnabledAutoPostBackImage</code>	The list of enabled images to use for the autopostback button displayed on non-javascript devices.
<code>PersistState</code>	If set to true the control will persist its state in the mobile profile. The previous state will be used in subsequent page requests. This property has no effect if MobileProfile is disabled. Defaults to true.
<code>ProfileDataKey</code>	An optional shared key that can be used across to enable mobile profile data related to the control to be shared with other controls of the same type using the same key.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.13 FileUpload

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.

Property	Description
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>NotSupportedText</code>	The text that should be displayed if file upload is not supported by the mobile device.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.14 HyperLink

See [MSDN Documentation](#) for base details.

Property	Description
<code>AltImages</code>	The list of of different size versions of the same image. Usedto improve the quality of the final image rendered on mobile device.
<code>CalculateSizeMode</code>	The method used to determine the size of the optimum size of the image in pixels.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.15 Image

See [MSDN Documentation](#) for base details. See the online images [tutorial](#) for a walk through of this control.

Property	Description
<code>AltImages</code>	The list of of different size versions of the same image. Usedto improve the quality of the final image rendered on mobile device.
<code>CalculateSizeMode</code>	The method used to determine the size of the optimum size of the image in pixels.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between

	different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.16 ImageButton

See [MSDN Documentation](#) for base details. See the online images [tutorial](#) for a walk through of this control.

Property	Description
<code>AltImages</code>	The list of of different size versions of the same image. Used to improve the quality of the final image rendered on mobile device.
<code>CalculateSizeMode</code>	The method used to determine the size of the optimum size of the image in pixels.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.17 Label

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.18 LinkButton

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.

Property	Description
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.19 ListBox

See [MSDN Documentation](#) for base details.

Property	Description
<code>AutoPostBack</code>	If the mobile device does not support javascript and the EnableAutoPostBackButton property is true a button will appear next to the control to enable the user to manual postback the page.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>DisabledAutoPostBackImage</code>	The list of disabled images to use for the expand button when displayed on non-javascript devices.
<code>EnableAutoPostBackButton</code>	If set to false auto post back will be disabled when displayed on mobile devices that do not support javascript. Note: AutoPost back must not be a critical part of the mobile web applications process.
<code>EnabledAutoPostBackImage</code>	The list of enabled images to use for the autopostback button displayed on non-javascript devices.
<code>PersistState</code>	If set to true the control will persist its state in the mobile profile. The previous state will be used in subsequent page requests. This property has no effect if MobileProfile is disabled. Defaults to true.
<code>ProfileDataKey</code>	An optional shared key that can be used across to enable mobile profile data related to the control to be shared with other controls of the same type using the same key.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.20 Location

The 51Degrees.mobi location control provides all the services developers need in manage geographic location in a single control. It enables questions such as “Where am I now?” or “What points of interest are there where I’m going?” to be answered smoothly without the developer needing to become concerned about the technology implementations and platforms involved.

See this [tutorial](#) for an online guide walking through the implementation of the location control.

9.20.1 Methods

The location control supports two types of geographic location request:

1. Geocoding determines the location using text entered by the user such as a zip code or city name.
2. Determine the current location of the mobile device without requiring user input.

The two types involve very different technology each of which is described in this chapter. The different types can be combined to support common business scenarios such as a point of interest locator, or a continuous location tracking service.

A third method of locating mobile devices is also available via some Mobile Network Operators (MNOs) and is typically part of a commercial service. This method is explained in this chapter but is not supported within the 51Degrees.mobi framework product at this time. Developers wishing to use this method should contact 51Degrees.mobi via the web site to discuss how it could be applied to their mobile web application, target countries and MNOs.

9.20.1.1 Geocoding

3rd parties provide services to translate partial addresses or place names into geographic location described using latitude and longitude. Web servers running 51Degrees.mobi can access these 3rd parties without requiring bespoke code. The location control will capture the data from the user and then facilitate a dialogue between the 3rd party and the user. See Figure 1.

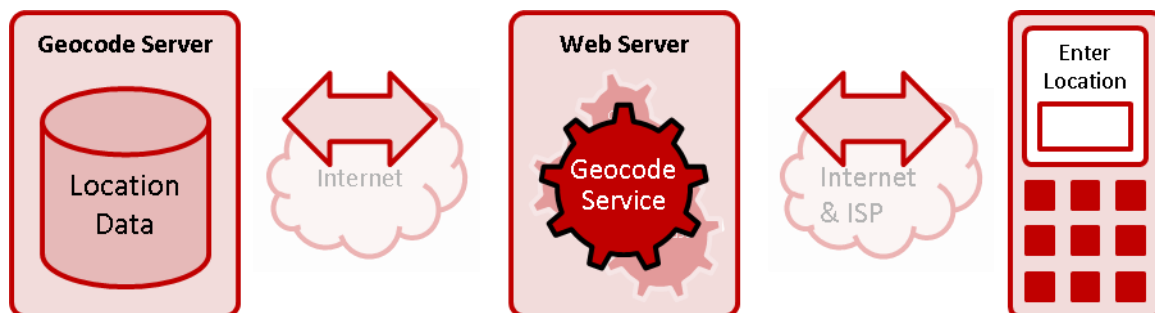


Figure 1 - 3rd party geocoding

The user will provide location text and then one of 3 scenario's will be returned from the 3rd party geocode server.

1. One location was found.
2. Many possible locations were found.

3. No locations found.

If more than one location is found the user will be presented with a list of locations to choose from. See Figure 2 for a summary flow diagram.

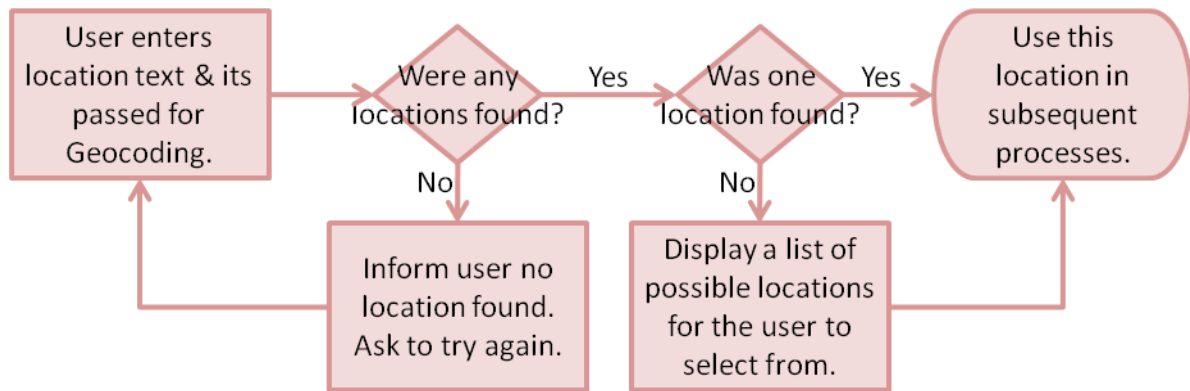


Figure 2 - Geocoding process flow

Once a location has been chosen through any route the location control will fire an event informing the developer that a location has been found providing details about the location selected. This information can then be used within subsequent processes.

51Degrees.mobi links to [Google’s geocoding web service](#) to determine the location. 51Degrees.mobi does not provide a licence to use Google’s service and you must obtain a licence for your purposes. In many situations such a licence will be provided freely by Google. An important consideration when using a free licence is the requirement to use the service in conjunction with Google maps. See this [link](#). 51Degrees.mobi provide a separate mapping control for this purpose. If you would like to use an alternative Geocoding service with the location control please contact us via the <http://51Degrees.mobi> web site.

9.20.1.2 No User Input

Some mobile devices allow a web page to determine the geographic location of the mobile device via JavaScript. This method determines an accurate location without the need for the user to enter information. A second web server based method using the public IP address of the device can also be used, but is considerably less accurate. Both methods are available via the location control.

9.20.1.2.1 Device Based

Many modern mobile devices will know their position through the use of the Global Position System (GPS) using satellites, WiFi or knowledge of mobile network cell site information. See Figure 3.

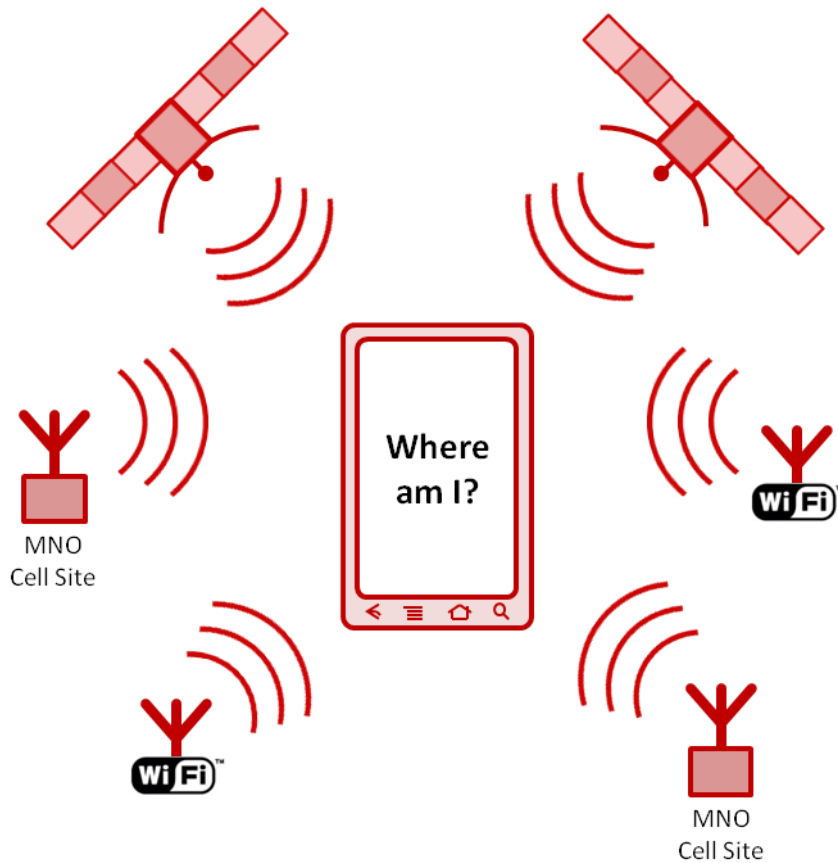


Figure 3 - Device based location sources

JavaScript running within the web browser of the mobile device can request the current location from the device. The mobile device is responsible for determining which source is the most accurate at the time. For example; GPS coverage within a building is not likely to be very accurate, but a nearby WiFi hot spot will be very accurate. The mobile device chooses the source.

Some mobile devices do not implement all of these location sources. The accuracy of the different sources will also vary by platform. As developers there is little we can do to improve the accuracy of the location the device returns.

When a web site requests geographic location information via JavaScript a message will typically appear on the mobile device screen informing the user the web site would like to know the current position. The user can then accept, reject or permanently enable location access for the web site. The precise implementation will vary between device platforms.

9.20.1.2.2 IP Address

Several companies provided services that map version 4 IP addresses to geographic locations. One such company is [MaxMind](#) who provide a small database of typically 30mbs in size containing the necessary data.

51Degrees.mobi enables MaxMind databases to be used to provide a geographic location for the public IP address of the mobile device accessing the mobile web application. The location returned can be inaccurate depending on the configuration of the MNO, WiFi service or handset. Typically WiFi connections will produce in a more accurate result. The method can be relied upon to locate a mobile device to a country, or even to a region, but should not be relied upon for more accurate requirements.

51Degrees.mobi does not distribute or licence MaxMind databases or services. These need to be acquired and updated separately. The terms of any licence required by MaxMind to use their open source or commercial product will need to be accepted if using their databases with 51Degrees.mobi.

9.20.1.3 Mobile Network Operator

Some MNO's will provide a commercial service to 3rd parties which enables the location of a mobile device to be determined using the mobile phone number or the Subscriber Identification Number (SIM). The availability of such services will depend on individual MNOs and countries.

The 3rd party application will need to know the SIM or mobile phone number to be queried. This will typically have been gained via text message or web site activity prior to the location request being made. A secure communication method, typically carried over the internet, will be used to request the location from the MNO. The MNO's cell sites are in constant contact with all mobile devices that are currently switched on and in the local area. See Figure 4.



Figure 4 - Mobile network operator location based services

Different technologies are used by the MNO to determine the location of the mobile device when the request is received. The accuracy of the result varies depending on the MNO and mobile device. For example; a MNO that receives location information via a built in GPS within the device will be more accurate than one that uses the signal strength between the mobile device and surrounding cell sites to determine an approximate location.

Many MNOs will work together with aggregators to avoid dealing directly with application developers. The advantage to developers is that they have a single relationship with one aggregator rather than multiple relationships with each MNO and do not need to know which MNO a mobile number or SIM is with. MNOs benefit by avoiding the need to deal with many smaller application developers.

MNO location based services cost anywhere from \$0.16 per request depending on volumes and minimums.

9.20.2 Implementation

A location control can be added to the page using ASP.NET code similar to the following.

ASP.NET

```
<mob:Location ID="Location1" runat="server" Mode="Client"
HighAccuracy="true" AutoPostBack="True" OnLocationFound="LocationFound">
</mob:Location>
```

The mode property of the control is used to determine the method(s) of obtaining a location to use. See Table 1 - Location Modes for valid mode values.

Value	Description
MaxMind	The IP Address database held on the server to determine an approximate location.
Client	Location services on mobile device are accessed to determine the handsets location using WiFi, GPS and other methods dependent on the device.
GeoCode	Google’s geocode service is used to obtain a location based on partial address information entered via a text box.
Server	MaxMind & GeoCode used together.
Precise	GeoCode & Client are used together.
All	All three methods are used together.

Table 1 - Location Modes

Modes such as Server, Precise and All which utilise a combination of user interface and automatic methods to determine a location present a user interface similar to those shown in Table 2 - Location Control Examples.

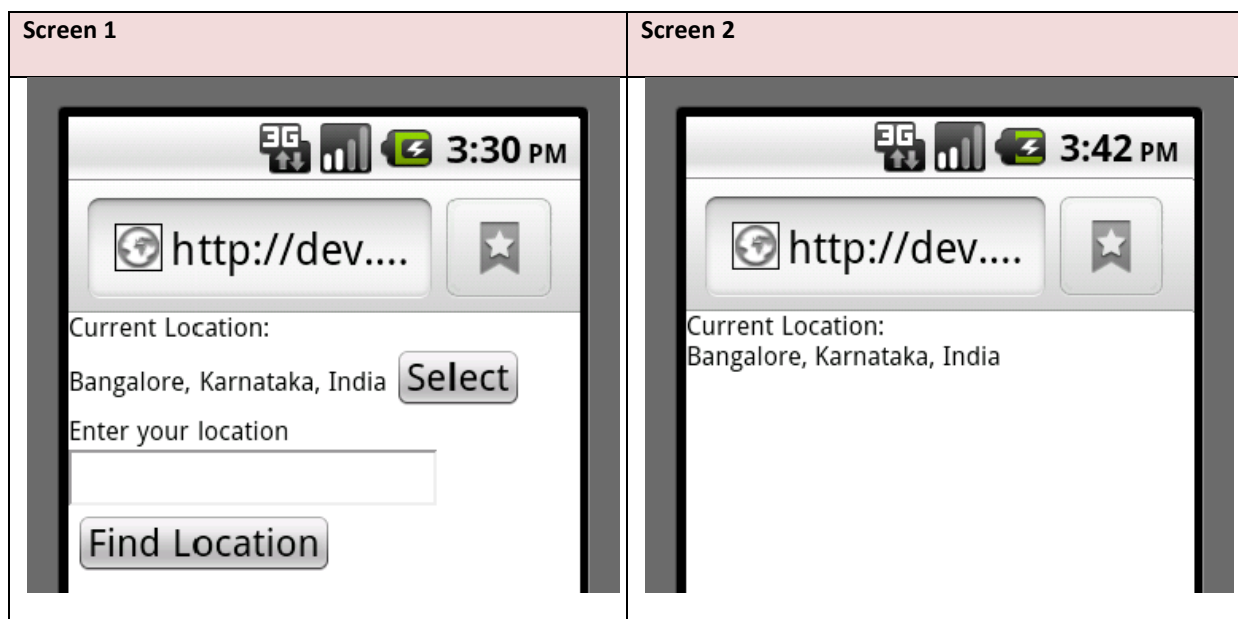


Table 2 - Location Control Examples

The control presents both the location found via the automatic location service, and the option to enter a location via a textbox for geocoding. Either the automatic location can be chosen via a button or the geocode process can be used. Whichever method is used the LocationFound event will be fired when the post back occurs to provide the chosen location details. A method similar to the following can be used to process the event and utilise the location found.

C#

```
protected void LocationFound(object sender, LocationFoundEventArgs e)
{
    Label1.Text = e.IsAvailable ? e.LocationInfo.Position.ToDisplay() :
    "Not Found";
}
```

The following tables explain the properties available to control and style the location control to fit into any mobile web application.

A tutorial explaining how to use the location control can be found [here](#).

9.20.3 Reference

Property	Default	Description	Relevant Modes
<code>AutoFoundPrefixText</code>	Current Location:	The text which appears before the automatically found location. Only shown when DisplayLocation is set to true.	Client MaxMind
<code>AutoPostBack</code>	False	An auto post back to the server will occur when the location is found.	Client
<code>AutoRefreshInterval</code>	0	How often to auto post back the location to the server.	Client
<code>CurrentLocationText</code>	Current Location	The name of the current location. Can be set by the user. Only shown when DisplayLocation is set to true.	Client, GeoSearchM axMind
<code>DisplayLocation</code>	False	When set to true the location that has been found will be displayed automatically. If false the control will have no User Interface or display once the location has been identified.	Client GeoSearchM axMind
<code>EnterLocationLabelText</code>	Enter your location	The text displayed above the GeoCode location search text box.	GeoCode
<code>FindLocationButtonText</code>	Find Location	The text displayed on the button used to search for a location once the textbox has been completed.	GeoCode
<code>GeoCodeMaxItems</code>	0	The maximum number of previously entered values in the textbox.	GeoCode
<code>GeoCodeTimeout</code>	10	Number of seconds before the GeoCode request times out.	GeoCode
<code>GeoSearchCountryCode</code>	NotSet	When GeoSearchCountryMode is set to Custom a country to limit geocode searches to. Eliminates similar locations from other countries from appearing. See the following link for a full list. http://code.google.com/apis/maps/documentation/geocoding/#RegionCodes	GeoCode
<code>GeoSearchCountryMode</code>	None	Possible values are: None : Uses Google's default country code. At the time of writing this is the US. Custom: Uses the country code specified by the <code>GeoSearchCountryCode</code> property. Auto: MaxMind is used to identify the country the device is in and will limit the geocode search options to this country.	GeoCode
<code>GeoSearchFoundPrefix</code>	Found	The text appearing before the location	GeoCode

Property	Default	Description	Relevant Modes														
Text	Location:	returned from a GeoCode request. Only shown when DisplayLocation is set to true.															
HighAccuracy	False	Where location service APIs, such as Google Gears, have the option to provide more detailed location information the HighAccuracy property determines if it should be used.	Client														
LocatingText	Locating mobile device...	The text displayed whilst the location service of the mobile device is locating the mobile device. Only displayed if DisplayLocation is set to true.	Client														
LocationInfo	Null	If a location has been found this property will provide details of the found location via the															
LocationNotFoundText	Sorry. We're unable to locate you at the moment.	If a location cannot be found automatically the text that will be displayed when DisplayLocation is set to true.	Client, MaxMind														
Mode	All	<table border="0"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>Client</td> <td>Location services on the mobile device are used to provide a location.</td> </tr> <tr> <td>MaxMind</td> <td>The GeoLiteCity database provided by MaxMind is used to determine a location.</td> </tr> <tr> <td>Geocode</td> <td>The user is asked to provide a location via a free text control and Google's GeoCode services are used to determine a precise location.</td> </tr> <tr> <td>Server</td> <td>MaxMind and Geocode are used together.</td> </tr> <tr> <td>Precise</td> <td>GeoCode and Client are used together.</td> </tr> <tr> <td>All</td> <td>Client, MaxMind and GeoCode are used together. If Client does not return a location the MaxMind location will be used.</td> </tr> </tbody> </table>	Value	Description	Client	Location services on the mobile device are used to provide a location.	MaxMind	The GeoLiteCity database provided by MaxMind is used to determine a location.	Geocode	The user is asked to provide a location via a free text control and Google's GeoCode services are used to determine a precise location.	Server	MaxMind and Geocode are used together.	Precise	GeoCode and Client are used together.	All	Client, MaxMind and GeoCode are used together. If Client does not return a location the MaxMind location will be used.	N/A
Value	Description																
Client	Location services on the mobile device are used to provide a location.																
MaxMind	The GeoLiteCity database provided by MaxMind is used to determine a location.																
Geocode	The user is asked to provide a location via a free text control and Google's GeoCode services are used to determine a precise location.																
Server	MaxMind and Geocode are used together.																
Precise	GeoCode and Client are used together.																
All	Client, MaxMind and GeoCode are used together. If Client does not return a location the MaxMind location will be used.																
MultipleLocationFoundLabelText	Choose a location	If the GeoCode process returns more than one location the text that is displayed above the list box of possible locations to choose from.	GeoCode														
PersistState	True	If set to true the control will persist its state in the mobile profile. The previous state will be used in subsequent page requests. This property has no effect if MobileProfile is disabled.	All														
RequestAddress	False	If set to true the control will attempt to convert a latitude and longitude into a readable address or point of interest name using Google's reverse geocoding web	Client GeoCodeMax Mind														

Property	Default	Description	Relevant Modes
		service. If set to false the control will display locations as Latitude and Longitude in the format 57°18'22.5"N 4°27'32.7"W. The control will require longer to return a location if this property is set to true.	
TextFieldAltItemStyle	Null	The style information used when rendering the alternative item in the items list of the location textbox.	
TextFieldAltItemStyleID	Null	The ID of the Style control to use to provide additional style information to the alternative item in the items list of the location textbox if displayed.	
TextFieldItemsStyle	Null	The style information used when rendering the items container of the location textbox.	
TextFieldItemsStyleID	Null	The ID of the Style control to use to provide additional style information to the items container of the location textbox if displayed.	
TextFieldItemStyle	Null	The style information used when rendering the item in the items list of the location textbox.	
TextFieldItemStyleID	Null	The ID of the Style control to use to provide additional style information to the item in the items list of the location textbox if displayed.	

Event	Description
LocationFound	<p>Fired when the location control has found an event.</p> <p><i>C#</i></p> <pre>protected void LocationFound(object sender, LocationFoundEventArgs e) { }</pre>

Class / Enumeration	Property	Description
LocationFoundEventArgs	IsAvailable	True if a location was found. False if a timeout occurred and no location could be obtained from any of the sources available as defined by mode property of the Location control.
	LocationInfo	A LocationInfo class.

Class / Enumeration	Property	Description
	LocationSource	The source used to obtain the location information. Valid values include: NotSet = Unknown Maxmind = MaxMind database Geocode = Google Geocode service If the control is configured in client mode values of GoogleGears, BlackBerry, iPhone, Android could be returned depending on the type of client.
<code>LocationInfo</code>	Latitude	The latitude of the centre of the location found.
	Longitude	The longitude of the centre of the location found.
	Position	A position object for the centre of the location found.
	Circle	The circular area the location found is within.
	Area	Similar to Circle but includes a region, city and country information about the area if available.
<code>Position</code>	Latitude	The latitude of the position.
	Longitude	The longitude of the position.
<code>Position - > Circle</code>	Accuracy	The radius of the circular area in metres. Could also be described as the accuracy of the position in metres.
<code>Circle -> Area</code>	City	The city located by the service if available.
	Region	The region located by the service if available.
	Country	The country located by the service if available.

9.21 Login

See [MSDN Documentation](#) for base details.

Property	Description
<code>AutoSize</code>	If set to true and no height information is provided explicitly for the control the height will automatically be set to a size that is appropriate for the mobile device displaying the page. This ensures selectable controls are sized appropriately on touch screen devices. Defaults to true.
<code>CheckBoxStyleID</code>	The ID of the Style control to use to provide additional style information to the Checkbox.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>FailureTextStyleID</code>	The ID of the Style control to use to provide additional style information to the failure text.
<code>HyperLinkStyleID</code>	The ID of the Style control to use to provide additional style information to the hyperlinks.
<code>InstructionStyleID</code>	The ID of the Style control to use to provide additional style information to the instruction text.
<code>LabelStyleID</code>	The ID of the Style control to use to provide additional style information to the labels.
<code>LoginButtonStyleID</code>	The ID of the Style control to use to provide additional style information to the login button.
<code>MaxHistoryItems</code>	The maximum number of items to display under the text box.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.
<code>TextBoxStyleID</code>	The ID of the Style control to use to provide additional style information to the textboxes.
<code>TitleTextStyleID</code>	The ID of the Style control to use to provide additional style information to the title text.
<code>UserNameAltItemStyle</code>	The style information used when rendering the alternative item in the items list of the username textbox.
<code>UserNameAltItemStyleID</code>	The ID of the Style control to use to provide additional style information to the alternative item in the items list of the username textbox if displayed.
<code>UserNameItemsStyle</code>	The style information used when rendering the items container of the username textbox.
<code>UserNameItemsStyleID</code>	The ID of the Style control to use to provide additional style information to the items container of the username textbox if displayed.
<code>UserNameItemStyle</code>	The style information used when rendering the item in the items list of the username textbox.
<code>UserNameItemStyleID</code>	The ID of the Style control to use to provide additional style information to the item in the items list of the username textbox if displayed.
<code>ValidatorTextStyleID</code>	The ID of the Style control to use to provide additional style information to the validator text.

9.22 LoginName

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.23 LoginStatus

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.24 Map

Both Google and Microsoft provide online mapping products which can be used without charge in many situations. The 51Degrees.mobi – Framework provides a simple method of deploying these maps within a mobile application ensuring they work on every type of mobile device.

Users have become familiar with dragging maps around with points of interest and overlays appearing in almost real time when used on the big screen broadband internet experience of a desktop or laptop computer. The small screen and bandwidth constrained environment of a mobile device makes such a solution extremely difficult to achieve consistently across different devices. The map control provided with 51Degrees.mobi makes use of static map services provided by Bing and Google to create an experience that is both consistent and reliable on any mobile device. See Figure 5 - Example Maps for screen shots of different maps from mobile device emulators.



Google map displayed on feature phone.



Bing map displayed on smart phone with selectable points of interest.

Figure 5 - Example Maps

The `Provider` property of the map control determines the mapping provider to use. All other properties of the control remain the same irrespective of the map provider.

Due to the current capabilities of static maps it is not practical to enable map markers to be selectable directly on the map. Instead the map control provides a list of the displayed markers under the map for selection as links. When these marker links are selected the `MarkerSelected` event will be fired and the `CommandEventArgs` object to the event handler will contain the value of the selected marker in the `CommandArgument` property. Using this approach markers can be easily selected.

Navigation control will appear in two different positions depending on the capabilities of the mobile device. Where stylus or touchscreens are supported the controls will appear around the map. Feature phones with different input methods will position the controls towards the bottom left hand corner of the map. We believe these different positions make the map easier to interact with. Navigation controls can be turned on or off using the `ControlsVisible`, `ZoomInEnabled` and `ZoomOutEnabled` properties.

The current terms of service for these products should be checked prior to using them.

- [Google Terms of Service](#)
- [Bing Maps Terms of Use](#)

See this [tutorial](#) for an online guide walking through an implementation of the map control.

The following tables provide details of the properties and events available.

Property	Description
<code>AltItemStyle</code>	The style information used when rendering the alternative item in the items

Property	Description
	list.
<code>AltItemStyleID</code>	The ID of the Style control to use to provide additional style information to the alternative item in the markers list.
<code>BottomRightLatitude</code>	The latitude of the bottom right corner of the displayed map.
<code>BottomRightLongitude</code>	The longitude of the bottom right corner of the displayed map.
<code>CalculateSizeMode</code>	The method used to calculate the size of the buttons that control map navigation.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>ControlsVisible</code>	Set to true to enable navigation controls for the map.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>DownAltText</code>	The alternative text to associate with the down navigation button.
<code>DownImages</code>	Image sources used to provide image information to the down navigation button.
<code>DownLeftAltText</code>	The alternative text to associate with the down and left navigation button.
<code>DownLeftImages</code>	Image sources used to provide image information to the down and left navigation button.
<code>DownLeftShortText</code>	Text to use with devices that do not support images rendered as submit buttons. <i>Down and left navigation text to use with devices that do not support images rendered as submit buttons.</i>
<code>DownRightAltText</code>	The alternative text to associate with the down and right navigation button.
<code>DownRightImages</code>	Image sources used to provide image information to the down and right navigation button.
<code>DownRightShortText</code>	Down and right navigation text to use with devices that do not support images rendered as submit buttons.
<code>DownShortText</code>	Down navigation text to use with devices that do not support images rendered as submit buttons.
<code>FirstPageImage</code>	Image sources used to provide image information to the first page button of markers pagination.

Property	Description
ItemsStyleID	The ID of the Style control to use to provide additional style information to the markers container.
ItemsStyles	The style information used when rendering the items container.
ItemStyle	The style information used when rendering the item in the items list.
ItemStyleID	The ID of the Style control to use to provide additional style information to the item in the markers list.
LastPageImage	Image sources used to provide image information to the last page button of markers pagination.
Latitude	The latitude of the map centre.
LeftAltText	The alternative text to associate with the left navigation button.
LeftImages	Image sources used to provide image information to the left navigation button.
LeftShortText	Left navigation text to use with devices that do not support images rendered as submit buttons.
Longitude	The longitude of the map centre.
MapHeightPixels	The height of the map image in pixels. If not specified the value will automatically be calculated based on the dimensions of the screen.
MapWidthPixels	The width of the map image in pixels. If not specified the value will automatically be calculated based on the dimensions of the screen.
Markers	A collection of markers.
MaxMarkers	The number of map markers to display on a page.
NextPageImage	Image sources used to provide image information to the next page button of markers pagination.
PreviousPageImage	Image sources used to provide image information to the previous page button of markers pagination.
Provider	The mapping provider to use with the control. Can be Google or Bing.
RightAltText	The alternative text to associate with the right navigation button.
RightImages	Image sources used to provide image information to the right navigation button.
RightShortText	<i>Right and left navigation text to use with devices that do not support images rendered as submit buttons.</i>

Property	Description
<code>SelectionEnabled</code>	Set to true to enable markers to be selected from the list. False will render the markers but they cannot be selected.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.
<code>TopLeftLatitude</code>	The latitude of the top left corner of the displayed map.
<code>TopLeftLongitude</code>	The longitude of the top left corner of the displayed map.
<code>UpAltText</code>	The alternative text to associate with the up navigation button.
<code>UpImages</code>	Image sources used to provide image information to the up navigation button.
<code>UpLeftAltText</code>	The alternative text to associate with the up and left navigation button.
<code>UpLeftImages</code>	Image sources used to provide image information to the up and left navigation button.
<code>UpLeftShortText</code>	<i>Up and left navigation text to use with devices that do not support images rendered as submit buttons.</i>
<code>UpRightAltText</code>	The alternative text to associate with the up and right navigation button.
<code>UpRightImages</code>	Image sources used to provide image information to the up and right navigation button.
<code>UpRightShortText</code>	Up and right navigation text to use with devices that do not support images rendered as submit buttons.
<code>UpShortText</code>	Up and left navigation text to use with devices that do not support images rendered as submit buttons.
<code>ZoomInAltText</code>	The alternative text to associate with the zoom-in button.
<code>ZoomInDisabledImages</code>	Image sources used to provide image information to the zoom-in disabled button.
<code>ZoomInEnabled</code>	True to enable the zoom in option. Can be set to false in situations where no points of interest will be visible if the map is zoomed in further.
<code>ZoomInImages</code>	Image sources used to provide image information to the zoom-in button.
<code>ZoomInShortText</code>	Zoom-in text to use with devices that do not support images rendered as submit buttons.
<code>ZoomOutAltText</code>	The alternative text to associate with the zoom-out button.
<code>ZoomOutDisabledImages</code>	Image sources used to provide image information to the zoom-out disabled button.

Property	Description
<code>ZoomOutEnabled</code>	True to enable the zoom out option.
<code>ZoomOutImages</code>	Image sources used to provide image information to the zoom-out button.
<code>ZoomOutShortText</code>	Zoom-out text to use with devices that do not support images rendered as submit buttons.

Event	Description
<code>MarkerSelected</code>	Fired when a map marker is selected. The <code>CommandArgument</code> property of the <code>CommandEventArgs</code> object passed into the event handler will contain the value of the marker selected.

9.25 Pager

The pager control can be bound to a `DataList` to enable paging similar to the `Grid` control provided with ASP.NET by Microsoft. Multiple pagers can be bound to the same `DataList` to providing paging controls above and below the `DataList`. Use the `ControlToBind` property of the `Pager` control to determine the `DataList` to associate with the pager.

Property	Description
<code>BorderRadius</code>	The corner radius of the buttons used in the pager control.
<code>ButtonCount</code>	The maximum number of buttons to display in the pager control.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>ControlToBind</code>	The ID of the control to associate the pager control with.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>FirstPageImage</code>	Image sources used to provide image information to the first page button image.
<code>ImageItemStyle</code>	The style information used when rendering the pagination images.
<code>ImageItemStyleID</code>	The <code>StyleID</code> of the <code>Style</code> control to use when calculating style properties for the navigation image buttons contained within the pager.
<code>ItemStyle</code>	The style information used when rendering the item in the items list.
<code>ItemStyleID</code>	The <code>StyleID</code> of the <code>Style</code> control to use when calculating style properties for the numeric navigation buttons used within the pager.

Property	Description
<code>LastPageImage</code>	Image sources used to provide image information to the last page button image.
<code>NextPageImage</code>	Image sources used to provide image information to the next page button image.
<code>NextPageSetText</code>	Next page text to use with devices that do not support images rendered as submit buttons.
<code>SelectedItemStyleID</code>	The ID of the Style control to use to provide additional style information to the selected numeric navigation button.
<code>PagerStyle</code>	The style information used when rendering the pager.
<code>PreviousPageImage</code>	Image sources used to provide image information to the previous page button image.
<code>PreviousPagerSetText</code>	Previous page text to use with devices that do not support images rendered as submit buttons.
<code>SelectedItemStyle</code>	The style information used when rendering the selected page index in pagination.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.
<code>TotalPages</code>	The total number of pages to be available.

Event	Description
<code>PageIndexChanging</code>	Fired when the active page changes.

9.26 Panel

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.27 PasswordRecovery

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>FailureTextStyleID</code>	The ID of the Style control to use to provide additional style information to the failure text.
<code>HyperLinkStyleID</code>	The ID of the Style control to use to provide additional style information to the hyperlinks
<code>InstructionStyleID</code>	The ID of the Style control to use to provide additional style information to the instruction text.
<code>LabelStyleID</code>	The ID of the Style control to use to provide additional style information to the labels.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.
<code>SubmitButtonStyleID</code>	The ID of the Style control to use to provide additional style information to the submit button.
<code>SuccessTextStyleID</code>	The ID of the Style control to use to provide additional style information to the success text.
<code>TextBoxStyleID</code>	The ID of the Style control to use to provide additional style information to the textboxes.
<code>TitleTextStyleID</code>	The ID of the Style control to use to provide additional style information to the title text.
<code>ValidatorTextStyleID</code>	The ID of the Style control to use to provide additional style information to the validator text.

9.28 RadioButton

See [MSDN Documentation](#) for base details.

Property	Description
<code>AutoPostBack</code>	If the mobile device does not support javascript and the EnableAutoPostBackButton property is true a button will appear next

Property	Description
	to the control to enable the user to manual postback the page.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>DisabledAutoPostBackImage</code>	The list of disabled images to use for the expand button when displayed on non-javascript devices.
<code>EnableAutoPostBackButton</code>	If set to false auto post back will be disabled when displayed on mobile devices that do not support javascript. Note: AutoPost back must not be a critical part of the mobile web applications process.
<code>EnabledAutoPostBackImage</code>	The list of enabled images to use for the autopostback button displayed on non-javascript devices.
<code>PersistState</code>	If set to true the control will persist its state in the mobile profile. The previous state will be used in subsequent page requests. This property has no effect if MobileProfile is disabled. Default true.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.29 RadioButtonList

See [MSDN Documentation](#) for base details.

Property	Description
<code>AutoPostBack</code>	If the mobile device does not support javascript and the EnableAutoPostBackButton property is true a button will appear next to the control to enable the user to manual postback the page.
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>DisabledAutoPostBackImage</code>	The list of disabled images to use for the expand button when displayed on non-javascript devices.
<code>EnableAutoPostBackButton</code>	If set to false auto post back will be disabled when displayed on mobile

Property	Description
	devices that do not support javascript. Note: AutoPost back must not be a critical part of the mobile web applications process.
<code>EnabledAutoPostBackImage</code>	The images that should be used for the post back button when enabled.
<code>PersistState</code>	If set to true the control will persist its state in the mobile profile. The previous state will be used in subsequent page requests. This property has no effect if MobileProfile is disabled. Default true.
<code>ProfileDataKey</code>	An optional shared key that can be used across to enable mobile profile data related to the control to be shared with other controls of the same type using the same key.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.30 RangeValidator

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.31 RegularExpressionValidator

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.32 RequiredFieldValidator

See [MSDN Documentation](#) for base details.

Property	Description
<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.33 Styles

A Styles control is never displayed directly on the web page. It contains formatting information that will be applied to controls referencing the `StyleID` property depending on the filter criteria specified and the capabilities of the requesting mobile device.

The `Property` attribute is evaluated to determine the value associated with the current request. The `Method` is then used to compare this result for the current request to the `Value` provided. If the result is positive either further child theme elements are evaluated or the style attributes applied to the control.

Table 3 provides a description of all the ASP.NET attributes available. Sometimes the same attribute will be provided for both the Styles control and the referencing control. Ideally this situation should be avoided in development. If it does occur one of two methods are used to determine which attribute value should be used depending on the attribute in question. Either the attribute value in the referencing control will only be updated if it has not been set (Update if empty). Or the attribute value will be overwritten irrespective of its current value. The method used for relevant attributes is shown in the far right hand column of the table.

Property	Mandatory	Description				
<code>Property</code>	Yes	The name of the property from the HttpBaseCapabilities class exposed via HttpRequest.Browser , or the WURFL capability to use when evaluating the element. If using WURFL capabilities ensure the capability has been configured to be loaded when the WURFL device database is parsed. For more information read the capabilitiesWhiteList documentation in the Foundation User Guide .				
<code>Method</code>	Yes	The method to use when comparing the value returned from the <code>property</code> attribute for the current request with the <code>value</code> attributes value. Valid values are: <table border="0" style="margin-left: 20px;"> <tr> <td style="padding-right: 20px;">Equals</td> <td>The value of the property must match the filter exactly.</td> </tr> <tr> <td>Contains</td> <td>The value of the property must contain the</td> </tr> </table>	Equals	The value of the property must match the filter exactly.	Contains	The value of the property must contain the
Equals	The value of the property must match the filter exactly.					
Contains	The value of the property must contain the					

Property	Mandatory	Description
		<p>filter.</p> <p>GreaterThan Filter and the value must both relate to a number and the property value must be greater than the filter value.</p> <p>LessThan Filter and the value must both relate to a number and the property value must be less than the filter value.</p> <p>Regex The filter is a regular expression and the IsMatch property will be used to determine a match.</p> <p>LessThanOrEqual Filter and the value must both relate to a number and the property value must be less than or equal to the filter value.</p> <p>GreaterThanOrEqual Filter and the value must both relate to a number and the property value must be greater than or equal to the filter value.</p>
Value	Yes	The value to compare using the matching method.
StyleID	Yes	The ID used to identify the style within the Page.
Style Attributes		Merge Rules
BackColor	No	background color of the control. Update if empty
BorderColor	No	border color of the control. Update if empty
BorderStyle	No	border style of the control. Update if empty
BorderWidth	No	border width of the control. Update if empty
CssClass	No	cascading style sheet (CSS) class rendered by the control on the client. Update if empty
Font-Bold	No	a value that indicates whether the font is bold. Overwrite
Font-Italic	No	a value that indicates whether the font is italic. Overwrite
Font-Name	No	primary font name. Overwrite
Font-Overline	No	a value that indicates whether the font is overlined. Overwrite
Font-Size	No	font size. Overwrite
Font-Strikeout	No	a value that indicates whether the font is Overwrite

Property	Mandatory	Description	
		strikethrough.	
Font-Underline	No	a value that indicates whether the font is underlined.	Overwrite
ForeColor	No	foreground color (typically the color of the text) of the control.	Update if empty
Height	No	height of the control.	Update if empty
HorizontalAlign	No	horizontal alignment of the contents in a container.	Update if empty
Style	No	Gets or sets	Update if empty
VerticalAlign	No	vertical alignment of the contents in a container.	Update if empty
Width	No	width of the Web server control.	Update if empty
Wrap	No	a value indicating whether the contents of a control will wrap.	Update if empty

Table 3 - Styles ASP.NET attributes

Style attributes are evaluated and applied to referencing controls during the Load event of the control. Therefore the developer should not provide additional style information at runtime until the PreRender event of the referencing control.

Properties of the Styles controls should not be changed at runtime. Unpredictable behaviour may result.

9.34 Table

See [MSDN Documentation](#) for base details.

Property	Description
ClientID	Overriden to provide shorter unique client IDs within markup. Can not be set.
CssIncludeGroup	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who’s style attributes are likely to vary between different requests of the same page.
StyleID	The StyleID of the Style control to use when calculating style properties for the control.

9.35 TextBox

The aim of any good application should be to reduce the barriers between the user and achieving the result they desire. Mobile device keyboards are often such a barrier when text entry is required. Where it is not

possible to avoid text entry the 51Degrees.mobi enhanced text box enables common text, or previous text to be populated in a list for the user to choose from rather than using the keyboard.

The following code example shows how to use the Items collection of the text box to pre-populate airport names.

ASP.NET

```
<mob:TextBox runat="server" ID="TextBoxAirport" AutoPostBack="true"
MaxItems="5">
  <Items>
    <mob:TextBoxItem Text="New York" />
    <mob:TextBoxItem Text="Los Angeles" />
    <mob:TextBoxItem Text="San Francisco" />
  </Items>
</mob:TextBox>
```

When the text box is first displayed to the user it will look like a normal text box. When the text box receives focus the list of possible options will appear underneath and can be selected from. See Figure 6 for an example of a textbox displayed on a touch screen mobile device with JavaScript support.

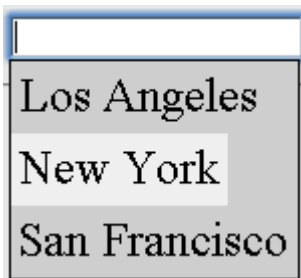


Figure 6 - JavaScript Textbox

In the above example the `MaxItems` property of the text box is configured to allow up to 5 items to be displayed. In addition to pre-populated items the textbox will remember previous entries if Mobile Profile is enabled. If the user were to enter another airport, perhaps Dallas, the next time the text box was displayed to them Dallas would become an available option. See Figure 7 for an example of the same textbox displayed on a mobile device without JavaScript support.

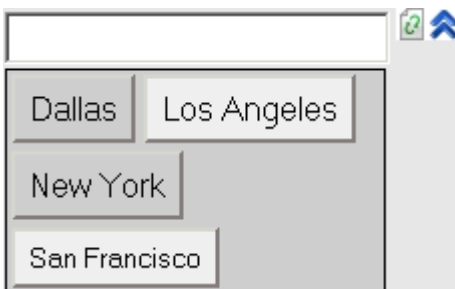


Figure 7 - Non JavaScript Textbox

Notice in Figure 7 how two icons appear to the right of the textbox to enable the user to generate the postback and expand/collapse events that cannot be implemented in JavaScript.

The default behaviour of the control is to replicate the base TextBox. Setting MaxItems to a positive value will enable these features.

9.35.1 Reference

See [MSDN Documentation](#) for base details.

Property	Description
<code>AltItemStyle</code>	The style information used when rendering the alternative item in the items list.
<code>AltItemStyleID</code>	The ID of the Style control to use to provide additional style information to the alternative item in the items list.
<code>AutoPostBack</code>	If enabled and the control is displayed on a mobile device that does not support javascript a postback button will appear next to the text box. 51Degrees.mobi recommends autopostback is not enabled within mobile web applications.
<code>ClientID</code>	Overriden in 51Degrees.mobi to support short Ids.
<code>CollapseImages</code>	The list of images to use for the collapse button when displayed on non-javascript devices.
<code>CssIncludeGroup</code>	The include group this control should be part of.
<code>DisabledAutoPostBackImage</code>	The list of disabled images to use for the autopostback button displayed on non-javascript devices.
<code>DisabledCollapseImages</code>	The list of disabled images to use for the collapse button when displayed on non-javascript devices.
<code>DisabledExpandImages</code>	The list of disabled images to use for the expand button when displayed on non-javascript devices.
<code>EnableAutoPostBackButton</code>	If set to false auto post back will be disabled when displayed on mobile devices that do not support javascript. Note: AutoPost back must not be a critical part of the mobile web applications process.
<code>EnabledAutoPostBackImage</code>	The list of enabled images to use for the autopostback button displayed on non-javascript devices.
<code>ExpandImages</code>	The list of images to use for the expand button when displayed on non-javascript devices.
<code>Items</code>	A list of pre-populated items that can be displayed under the text box. <i>ASP.NET</i> <code><Items></code> <pre> <mob:TextBoxItem Text="New York" /> <mob:TextBoxItem Text="Los Angeles" /> </pre>

Property	Description
	<pre><mob:TextBoxItem Text="San Francisco" /> </Items></pre> <p>The mob:TextBoxItem element also provides the property Priority. The higher the value the more important the item. The value of the MaxItems property will be used to determine the number of items to display.</p> <p>Items can be added to the control at runtime from your code through this collection.</p>
ItemsButtonHeight	The height of the expand/collapse button used to toggle the display of items on non-javascript enabled devices.
ItemsButtonWidth	The width of the expand/collapse button used to toggle the display of items on non-javascript enabled devices.
ItemsStyle	The style information used when rendering the items container.
ItemsStyleID	The ID of the Style control to use to provide additional style information to the items container.
ItemStyle	The style information used when rendering the item in the items list.
ItemStyleID	The ID of the Style control to use to provide additional style information to the item in the items list.
MaxItems	The maximum number of items to display under the text box.
PersistState	If set to true the control will persist its state in the mobile profile. The previous state will be used in subsequent page requests. This property has no effect if MobileProfile is disabled. Default true.
ProfileDataKey	Setting a string value enables previous entries to be shared between text box controls on different pages. For example; a user control containing a text box which captures search queries. The user control appears on multiple pages. This value will need to be set to a value unique across the mobile web application to enable the textbox to share previous values.
StyleID	The ID of the Style control to use with the text box control.

9.36 ValidationSummary

See [MSDN Documentation](#) for base details.

Property	Description
----------	-------------

<code>ClientID</code>	Overriden to provide shorter unique client IDs within markup. Can not be set.
<code>CssIncludeGroup</code>	Used to group control style attributes into stylesheet includes. If left empty the pages stylesheet will be used for the controls style attributes. This property should only be set for controls who's style attributes are likely to vary between different requests of the same page.
<code>StyleID</code>	The StyleID of the Style control to use when calculating style properties for the control.

9.37 Page

Property	Description
<code>MobileProfile</code>	If Mobile Profile is enabled returns a reference to the Mobile Profile associated with the currently requesting mobile device. Returns null if Mobile Profile is disabled.
<code>MobileSmartNavigation</code>	If set to true the page will automatically return to the position it was in before a post back request happened. Set this property to true for mobile web applications that are likely to use postback events with controls that will appear outside the initial visible page.
The following page properties will always return zero if Mobile Profile is disabled.	
<code>Bandwidth</code>	Returns the average round trip bandwidth in kilobits per second (Kbs). Zero is returned if insufficient data is available or the mobile device does not support the necessary javascript.
<code>ResponseTime</code>	Returns the average response time experienced by the end user. Zero is returned if insufficient data is available or the mobile device does not support the necessary javascript.