

Applying Hot Fix 34 for Sage SalesLogix Version 7.5.1

Version 7.5.1.34

Developed by Sage SalesLogix User Assistance



Your business in mind.

Applying Hot Fix 34 for Sage SalesLogix Version 7.5.1

| | |
|-------------------------------|---|
| Documentation Comments | This documentation was developed by Sage SalesLogix User Assistance. For content revisions, questions, or comments, contact the writers at saleslogix.techpubs@sage.com . |
| Copyright | Copyright © 1997-2009, Sage Software, Inc. All Rights Reserved. This product and related documentation are protected by copyright and are distributed under licenses restricting their use, copying, distribution, and decompilation. No part of this product or related documentation may be reproduced in any form by any means without prior written authorization of Sage Software and its licensors, if any. |
| Version | Version 7.5.1.34 2009 |
| Trademarks | SalesLogix is a registered trademark of Sage Software, Inc. Other product names may be trademarks or registered trademarks of their respective companies and are hereby acknowledged. |
| Disclaimer | Sage Software has thoroughly reviewed this manual. All statements, technical information, and recommendations in this manual and in any guides or related documents are believed reliable, but the accuracy and completeness thereof are not guaranteed or warranted, and they are not intended to be, nor should they be understood to be, representations or warranties concerning the products described. Sage Software assumes no responsibility or liability for errors or inaccuracies with respect to this publication or usage of information. Further, Sage Software reserves the right to make changes to the information described in this manual at any time without notice and without obligation to notify any person of such changes. |
| Technical Support | Technical Support is available to customers with support contracts directly from Sage Software and to Certified Business Partners. Calls are answered during business hours, Monday through Friday, excluding holidays. Current contact information is available on the Sage SalesLogix Web site. Customers with a valid technical support contract and a Web access code can request technical support electronically on the Sage SalesLogix SupportOnline/Sage Software Online Support and Services Web site. Sage SalesLogix Web site www.saleslogix.com Sage SalesLogix SupportOnline/Sage Software Online Support and Services Web site http://www.sagesoftwareonline.com Revisions to this book are posted on the Sage SalesLogix SupportOnline/Sage Software Online Support and Services Web site. Check this site regularly for current Sage SalesLogix product documentation. |

Applying Hot Fix 34 for Sage SalesLogix Version 7.5.1

This hot fix addresses the following issues:

| Defect | Description |
|---------|---|
| 1-70630 | Performance: Port back HQL Data Source code in support of Notes/Activities/History performance fix. |
| 1-71251 | Performance: Address tab performs an unconstrained query. |

This hot fix requires version 7.5.1 and 7.5.1 Hot Fix 6 and 7.5.1 Hot Fix 24. Do not install this hot fix on any other Sage SalesLogix version. Contact your support representative or go to <http://support.saleslogix.com> to obtain the hot fix.

File Information

| File Name | File Contents |
|---|---|
| SLX_v751_HF34.zip | Sage SalesLogix v7.5.1 HF34 VFS Upgrade.zip |
| Sage SalesLogix v7.5.1 HF34 VFS Upgrade.zip | AddressList.ascx, AddressList.ascx.cs, ActivityList.ascx, ActivityList.ascx.cs, HistoryList.ascx, HistoryList.ascx.cs, NotesList.ascx, NotesList.ascx.cs, Sage.Platform.WebPortal.dll |

Applying the Hot Fix

Ensure you apply 7.5.1 Hot Fix 6 and 7.5.1 Hot Fix 24 prior to this hot fix. Then, install the bundle using the Application Architect and build and deploy your Web site(s).

Note Before installing the bundle, review the files included in the hot fix. Back up any customized files that may be affected, or back up the whole project if there are many files. Then apply the hot fix bundle in one of the following ways:

- Manually merge the hot fix items with the customized items.
- Apply the bundle (overwriting all existing items), and then manually merge the customized items.

To take advantage of the HQL Data Source code, you must make changes to the form. See ["To use the HQL Data Source Code" on page 2](#) for more information.

To install the bundle

1. Copy **SLX_v751_HF34.zip** to your local machine and unzip it.

Note There is a folder path character limitation that requires the path where you save or copy bundles that are created in the Application Architect to be 57 characters or less (including the drive name).

-
2. Ensure you have **Write** permissions to the bundle installation folder. Check permissions on the Security tab in the folder properties.
 3. Open the **Application Architect**.
 4. In the **Project Explorer**, right-click the project, and then click **Install Bundle**.
 5. Browse to the folder where you unzipped the hot fix files, click **Sage SalesLogix v7.5.1 HF34 VFS Upgrade.zip**, and then click **Open**.
 6. On the **Select Bundle** screen, click **Next**.
 7. On the **Select Items** screen, ensure the **Portals** option is selected.
 8. Click **Next**, and then click **Finish**.

To build and deploy the Web site

1. In the **Project Explorer**, click the project.
2. Press and hold the **CTRL** key, and then on the **Build** menu, click **Build Web Platform**. All the files in the entire assembly for the current project are compiled. Status for the build is displayed in the Output Window at the bottom of the screen.
3. On the **View** menu, click **Deployment Explorer**.
4. Expand **Deployments**.
5. Double-click the portal to deploy.
6. Under **Deployment Targets**, right-click the target portal, and click **Deploy Portal**. A status box appears indicating status for the deployment.
7. Repeat these steps for the each of the affected portals.

Note By default, the Sage SalesLogix Web Client portal is set to deploy to the localhost, but you can change these settings to fit your environment.

To use the HQL Data Source Code

1. In the markup of the form, create or modify an existing gridview control.

```
<SalesLogix:SlxGridView ID="TestGrid" runat="server" AutoGenerateColumns="False"
EnableViewState="false" CssClass="datagrid" GridLines="None" AllowPaging="true"
PageSize="10" AllowSorting="true">

<Columns>

    <asp:BoundField DataField="Name" HeaderText="Name" SortExpression="Name" />
    <asp:BoundField DataField="Birthday" HeaderText="Birthdate" SortExpression="Birthday"
/>
    <asp:BoundField DataField="Phone" HeaderText="Main Phone" SortExpression="Phone" />

</Columns>
</SalesLogix:SlxGridView>
```

2. In the code-behind for the form, create a read-only public (or protected) property (and corresponding private property) to reference the local WebHqlListBindingSource. In the getter of this property, create the bindingsource object and give it the "select" and "from" portions of the hql to be used in the grid for this form.

```
private WebHqlListBindingSource _hqlBindingSource;
protected WebHqlListBindingSource HqlBindingSource
{
    get
    {
        if (_hqlBindingSource == null)
```

```

        {
            List<HqlSelectField> select = new List<HqlSelectField>();
            // "Name" is the property on the entity and the alias we used in the first
            column of the grid.
            select.Add(new HqlSelectField("c.Name", "Name"));
            select.Add(new HqlSelectField("c.Birthdate", "Birthday"));
            select.Add(new HqlSelectField("c.Phone", "Phone"));
            // create the bindingsource object and pass select and from parts:
            _hqlBindingSource = new WebHqlListBindingSource(select, "Contact c");
        }
        return _hqlBindingSource;
    }
}

```

3. In the OnFormBound method (or PreRender) do the following:
 - a. Determine the parameters for the where clause and pass that to the bindingsource.
 - b. Provide the name of the grid to which the bindingsource is providing data.
 - c. Call DataBind() on the grid.

```

protected override void OnFormBound()
{
    base.OnFormBound();
    string entityId = EntityService.EntityID.ToString();
    // add any other 'where' criteria here...
    HqlBindingSource.Where = string.Format("c.Account.Id = '{0}' and c.Type = 'Customer'",
    entityId);
    HqlBindingSource.BoundGrid = TestGrid;
    TestGrid.DataBind();
}

```