



Insite for Sitecore 3.3

Installation Guide

A developer's guide to installing Insite for Sitecore

Table of Contents

Chapter 1	Introduction.....	3
Chapter 2	Preparation.....	4
2.1	Packages Overview	5
2.2	Requirements	6
2.3	Prerequisites	7
Microsoft Unity 2.1.505.0	7	
Sitecore software	7	
2.4	Upgrading from a Previous Version	8
Chapter 3	Installing Insite for Sitecore	9
3.1	Connecting the Database.....	10
3.2	Installing the Package	11
3.3	Connecting the Database.....	12
3.4	Modifying web.config.....	13

Chapter 1

Introduction

Insite for Sitecore is an extension to Sitecore E-Commerce Services.

This document explains how you prepare for installation and the steps you need to take to install Insite for Sitecore.

This document contains the following chapters:

- **Chapter 1 — Introduction**
This chapter gives a summary of the content contained in each chapter.
- **Chapter 2 — Preparation**
This chapter outlines the requirements and prerequisites you need before you can install Insite for Sitecore.
- **Chapter 3 — Installing Insite for Sitecore**
This chapter explains how to install Insite for Sitecore.

Chapter 2

Preparation

Use the information in this section to prepare yourself for the installation of Insite for Sitecore. This chapter outlines the Insite for Sitecore packages you can install and specifies the hardware and software requirements you need before proceeding any further with the installation process.

This chapter contains the following sections:

- **Error! Reference source not found.**
- Requirements
- Prerequisites

2.1 Packages Overview

Before installing Insite for Sitecore, familiarize yourself with the contents of the Insite for Sitecore packages and the installation prerequisites.

There is one Insite for Sitecore packages that you can install:

- Insite for Sitecore 3.3.0.2417 rev. 20120521.

2.2 Requirements

Insite for Sitecore runs on Sitecore E-Commerce Services with DMS, so it requires the same hardware and software.

Note

For more information about installation requirements for the Sitecore E-Commerce Services and DMS, see the Sitecore Developers Network (SDN).

2.3 Prerequisites

This section lists the prerequisites you need for Insite for Sitecore:

Microsoft Unity 2.1.505.0

Insite for Sitecore requires this version of Microsoft Unity.

Sitecore software

The following Sitecore software must be installed prior to installing Insite for Sitecore:

- Sitecore CMS 6.5.0 Update-4 or later.
- Sitecore DMS 2.0.1 or later (optional).
- Sitecore E-Commerce Services 1.2 (or later)
- Web Forms For Marketers 2.3 (or later)

Important

You must have a Sitecore license that includes *Sitecore.EcommerceEnterprise*. If you are unsure whether you have the right license, search your license file for the words '*Sitecore.EcommerceEnterprise*'.

2.4 Upgrading from a Previous Version

There is no upgrade package for Insite for Sitecore. In order to upgrade from a previous version you must manually remove the old items from the Sitecore databases.

The commerce database will automatically update itself.

Chapter 3

Installing Insite for Sitecore

This chapter outlines the steps you need to take to install the Insite for Sitecore package.

This chapter includes the following sections:

- Installing the Package

3.1 Connecting the Database

1. Download Insite for Sitecore database files from SDN.
2. Unzip the *InSite.Commerce* database to the *Databases* folder in your site root:
`wwwroot\<CMS-build-name>\Databases`
3. Edit the `ConnectionStrings.config` file to point to the *InSite.Commerce* database.

Note

Insite for Sitecore requires the connection string name be `InSite.Commerce`.

4. In SQL Server, attach the *InSite.Commerce* database.

- Modifying `web.config`

3.2 Installing the Package

Before you install Insite for Sitecore read Chapter 2, Preparation and then follow the steps in this section.

Note

It is important that the prerequisites be met before you perform these steps.

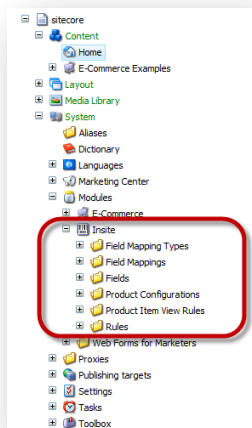
Install Insite for Sitecore Package

To Install the Insite for Sitecore Package:

5. Download Insite for Sitecore and save the zip file to the packages folder in your CMS build folder:
wwwroot\- 6. In the Sitecore CMS Desktop, click **Sitecore, Development Tools, Installation Wizard** to install the Insite for Sitecore package.
- 7. In the **Installation Wizard**, click **Next** and click **Browse** to locate the Insite for Sitecore package.
- 8. Select the Insite for Sitecore package you want to install and click **Open**.
- 9. Study the *ReadMe* file. If necessary make a copy of the text and click **Next**.
- 10. Click **Install**.

After you have installed the package, an Insite folder appears in the content tree

/sitecore/system/Modules/Insite:



3.3 Connecting the Database

11. Download Insite for Sitecore database files from SDN.
12. Unzip the *InSite.Commerce* database to the *Databases* folder in your site root:

```
wwwroot\<<CMS-build-name>\Databases
```
13. Edit the `ConnectionStrings.config` file to point to the *InSite.Commerce* database.

Note

Insite for Sitecore requires the connection string name be `InSite.Commerce`.

14. In SQL Server, attach the *InSite.Commerce* database.

3.4 Modifying web.config

After installing the Insite for Sitecore package you must make the following changes to web.config.

1. Add the following to configuration > configSections:

```
<!-- Insite for Sitecore -->
<section name="hibernate-configuration" type="NHibernate.Cfg.ConfigurationSectionHandler,
NHibernate"/>
<sectionGroup name="fluorinefx">
  <section name="settings" type="FluorineFx.Configuration.XmlConfigurator, FluorineFx"
requirePermission="false"/>
</sectionGroup>
<section name="unity"
type="Microsoft.Practices.Unity.Configuration.UnityConfigurationSection,
Microsoft.Practices.Unity.Configuration" />
<!-- Insite for Sitecore -->
```

2. Add the following to configuration > system.webServer > modules:

```
<!-- Insite for Sitecore -->
<add name="FluorineGateway" type="FluorineFx.FluorineGateway, FluorineFx"/>
<add name="NHibernateModule" type="NHibernate.Helper.Module, NHibernate.Helper"/>
<!-- Insite for Sitecore -->
```

3. Add the following to configuration > system.web > httpModules:

```
<!-- Insite for Sitecore -->
<add name="FluorineGateway" type="FluorineFx.FluorineGateway, FluorineFx"/>
<add name="NHibernateModule" type="NHibernate.Helper.Module, NHibernate.Helper"/>
<!-- Insite for Sitecore -->
```

4. Add the following to configuration:

```
<!-- Insite for Sitecore -->
<hibernate-configuration xmlns="urn:hibernate-configuration-2.2">
  <session-factory>
    <!-- properties -->
    <property
name="connection.provider">NHibernate.Connection.DriverConnectionProvider</property>
    <property name="connection.driver_class">NHibernate.Driver.SqlClientDriver</property>
    <property name="connection.connection_string">InSite.Commerce</property>
    <property name="connection.isolation">ReadUncommitted</property>
    <property name="show_sql">>false</property>
    <property name="dialect">NHibernate.Dialect.MsSql2008Dialect</property>
    <property name="query.substitutions">>true 1, false 0, yes 'Y', no 'N'</property>
    <!-- mapping files -->
    <mapping assembly="InSite.Model"/>
  </session-factory>
</hibernate-configuration>
<fluorinefx>
  <settings>
    <importNamespaces>
      <add namespace="InSite.Model" assembly="InSite.Model"/>
    </importNamespaces>
    <nullable>
      <type name="System.Int32" value="MinValue"/>
      <type name="System.Double" value="MinValue"/>
      <type name="System.DateTime" value="MinValue"/>
      <type name="System.Guid" value="Empty"/>
    </nullable>
  </settings>
</fluorinefx>
<unity xmlns="http://schemas.microsoft.com/practices/2010/unity">
  <sectionExtension
type="Microsoft.Practices.Unity.InterceptionExtension.Configuration.InterceptionConfigurationExtens
ion, Microsoft.Practices.Unity.Interception.Configuration" />
  <assembly name="InSite.Model" />
</unity>
```

```
<container>
  <register type="InSite.Model.BootStrapper.IBootStrapper"
mapTo="InSite.Model.BootStrapper.BootStrapper Generic" />
  <!--
  <extension type="Interception"/>
  <interception>
    <policy name="InterceptManagers">
      <matchingRule name="ManagerMatch" type="NamespaceMatchingRule">
        <constructor>
          <param name="namespaceName" value="InSite.Mvc.Domain.Manager"/>
        </constructor>
      </matchingRule>
      <callHandler name="LogHandler" type="InSite.Model.AOP.LogHandler, InSite.Model"/>
    </policy>
    <policy name="InterceptModel">
      <matchingRule name="ModelMatch" type="NamespaceMatchingRule">
        <constructor>
          <param name="namespaceName" value="InSite.Model"/>
        </constructor>
      </matchingRule>
      <callHandler name="LogHandler" type="InSite.Model.AOP.LogHandler, InSite.Model"/>
    </policy>
  </interception>
  -->
</container>
</unity>
<!-- Insite for Sitecore -->
```