



Compelson SyncML Server Administrator's Guide

revision 1.26

1 Table of contents

1	Table of contents	2
2	Introduction.....	4
2.1	Features	4
2.2	Device compatibility list	5
3	Installation	6
3.1	Prerequisites	6
3.2	Copying files.....	7
3.3	Setting up application account	8
3.4	Creating and populating databases.....	9
3.4.1	SQL Permissions	9
3.5	Creating event log	10
3.6	Configuring IIS	11
3.7	Configuring SyncML.....	14
3.7.1	Outgoing e-mail configuration	15
4	Activation.....	16
5	Management	17
5.1	Configuration files.....	17
5.2	Web manager.....	18
5.3	Event log	21
5.4	Communication dump folder	22
6	Localizing public web.....	23
7	Support.....	24
8	Appendix	25
8.1	SQL Permission Fix Commands	25

2 Introduction

Compelson SyncML Server is a remote synchronization solution for mobile devices. Currently it supports synchronization of contacts (phonebook), events (calendar), tasks, and notes – the PIM features most commonly found in today's mobile phones and personal digital assistants.

Over-the-air synchronization avoids the need for a cable, IR, or Bluetooth connection, which is vital to any local synchronization solution. In addition to the SyncML protocol support itself, mobile devices are only required to have a working Internet connection, such as GPRS, HSCSD, EDGE, UMTS, etc. Users don't have to install specialized software on their machines since a web browser is all they need to access their data stored on the server. Advantages of web-based access naturally include platform independence and simple centralized management. With the recent Internet ubiquity, data can be accessed almost instantly from virtually any place at any time. Losing one's mobile phone while traveling should no longer be a catastrophe. Visiting an Internet café is all it takes to access its latest content. Furthermore, loading all the contacts and meetings to a new phone is just a matter of a few clicks.

2.1 Features

Compelson SyncML Server consists of a synchronization engine distributed in binary form and a web interface (public and administrative web) distributed in source code form. The web interface is an ASP.NET 2.0 application and as its source files are available, it can be easily customized to fit specific needs. Minimum effort is required to translate the web to another language or change its overall visual appearance.

The engine is a .NET 2.0 application that integrates with the Microsoft IIS server at the `IHttpHandler` level, taking full advantage of the ASP.NET infrastructure. Microsoft SQL Server 2000 or 2005 is used as storage for all user data, as well as for internal structures related to synchronization.

Contrary to most SyncML implementations around, Compelson SyncML Server is proud to provide excellent support for national character sets (Cyrillic, diacritic characters, etc.) Other distinguished features include recurrent event support in the calendar. As for data formats, the engine accepts and generates contacts in the vCard 2.1 format (content type `text/x-vcard`), events and tasks in the vCalendar 1.0 format (content type `text/x-vcalendar`), and notes in vNote 1.1 (content type `text/x-vnote`) and plain text (content type `text/plain`) format.

Two-way as well as one-way synchronization is supported. More devices can synchronize their data with one server store, which results in their mutual synchronization with the server database acting as an intermediary.

2.2 Device compatibility list

We have implemented the solution according OMA SyncML DS 1.2 specifications. All phones and clients compliant with the specification are compatible with our server. The exception is only if there is some bug in the phone. The number of compliant phones is growing every day as manufacturers are releasing phones to the market, so we don't have a list.

3 Installation

The installation program performs several complex tasks in order to fully set up Compelson SyncML Server on the target machine. Aside from copying all the files, the necessary databases are created and populated, IIS is properly configured, a new event log is created, and the SyncML server is configured. The following paragraphs describe in detail all the steps involved in setting up the server.

3.1 Prerequisites

The server can be installed on any machine with:

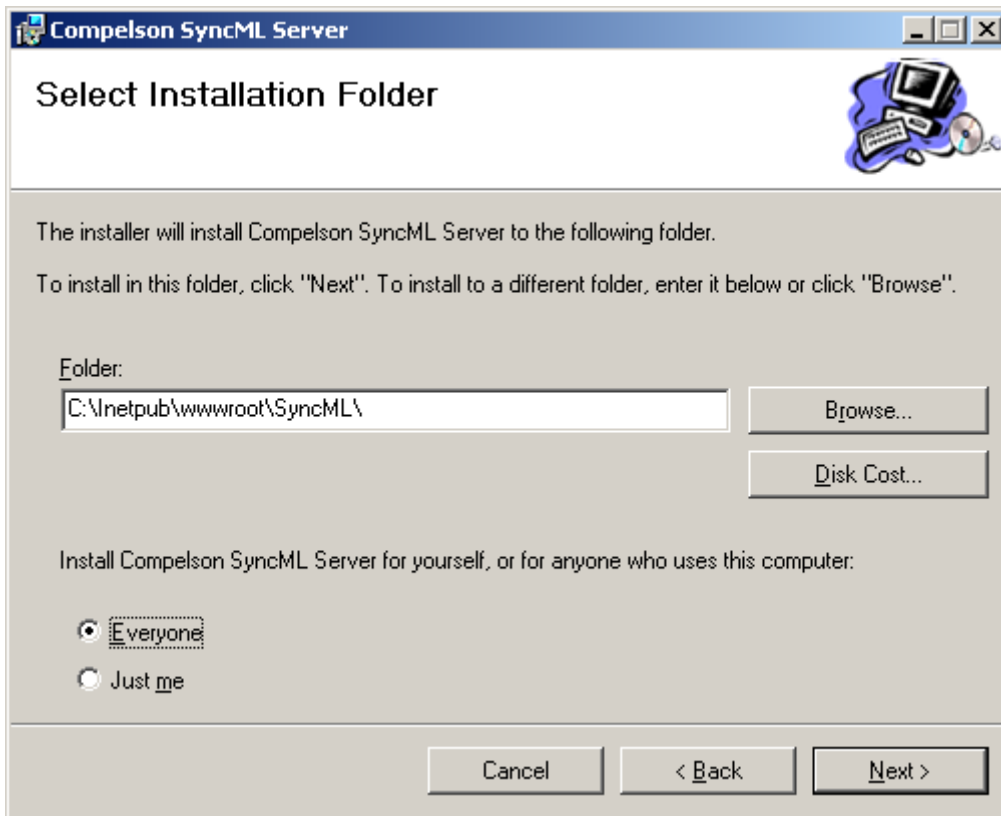
- Windows 2003 Server with IIS 6.0
- Microsoft .NET Framework 2.0 (distributed via Windows Update)
- Microsoft SQL Server 2000 (or 2005), the free Express (MSDE) editions are also supported (MSDE 2000 is not recommended, due to its limitations, which can cause problems with multiple simultaneous user connections)
- SMTP service installed on server or accessible through network (this is optionally required for sending informative e-mails from server application)

3.2 Copying files

The installer creates in the target folder a directory structure. Some important subdirectories are described in the following table (target path is the directory chosen during installation – C:\inetpub\wwwroot\SyncML\ by default):

Path	Description
TARGET_PATH	Represents the root of the public web application.
TARGET_PATH\WebManager	Represents the root of the web manager application.
TARGET_PATH\sync	Contains the “synchronization document”, with which client devices communicate.

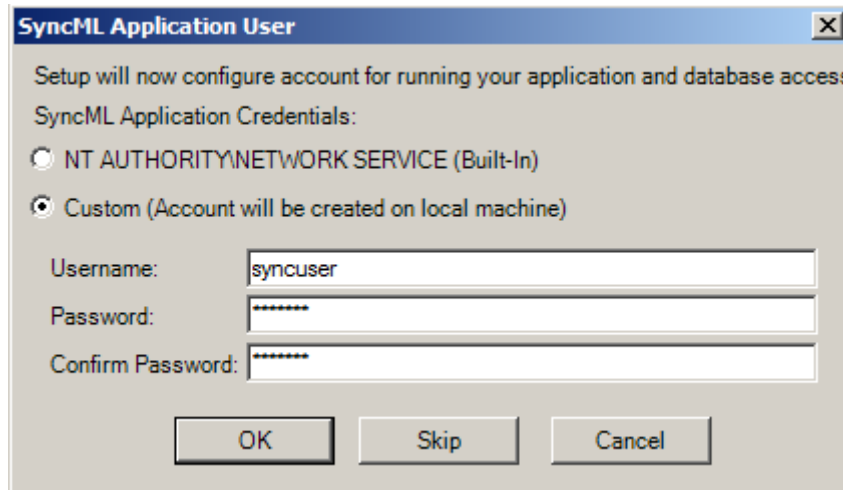
Make sure that the target folder is accessible for the account under which the applications will run. By default, it is the NT AUTHORITY\NETWORK SERVICE account.



It is rarely necessary to change the directory structure created by the installer. If the applications are required to run at different URLs, you can conveniently achieve this goal by reconfiguring the appropriate IIS virtual directories (see below).

3.3 Setting up application account

Installer can setup application either to run with under own account or use system 'NT AUTHORITY\NETWORK SERVICE' account.



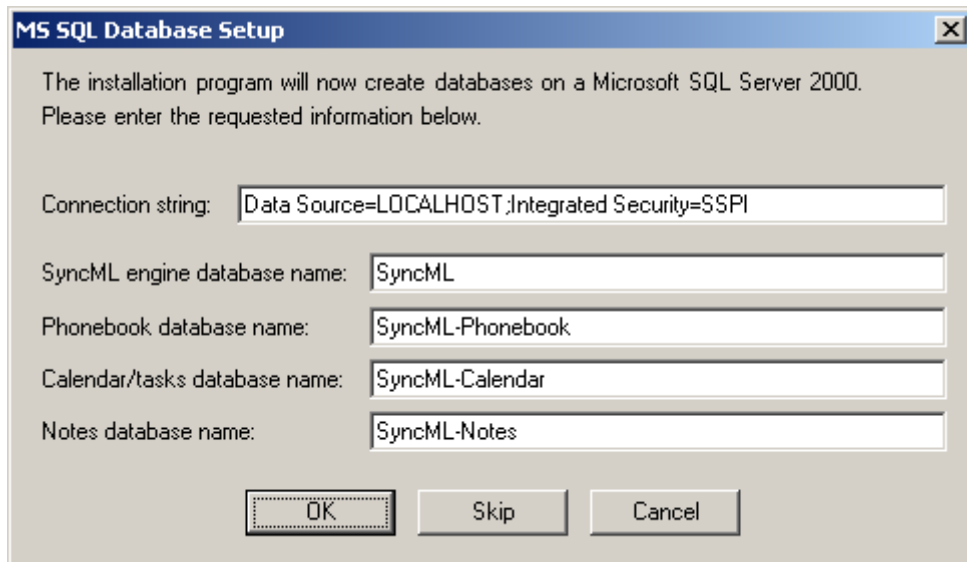
This account will be used as identity for web application pool and also for accessing databases. This account can be also used to access WebManager page (normally it can be accessed only by Administrators group members).

Account will be created on machine as local user account, and added to IIS_WPG group (group which has permission to run IIS application pool). Also read access for SyncML installation directory will be given to this account.

Important: Installer does **not support domain accounts, only local accounts.**

3.4 Creating and populating databases

The installer creates and populates four databases on a specified SQL server and grants needed access right to application account on them. The connection string as well as database names are entered during installation by the user. Make sure that the SQL server is running and that the databases do not already exist. If the databases already exist (i.e. you are repairing an existing installation), use the *Skip* button to go directly to the next step.



Note that it is not necessary to backup the databases before uninstalling Compelson SyncML Server because the user is asked during uninstall whether to keep them or remove them.

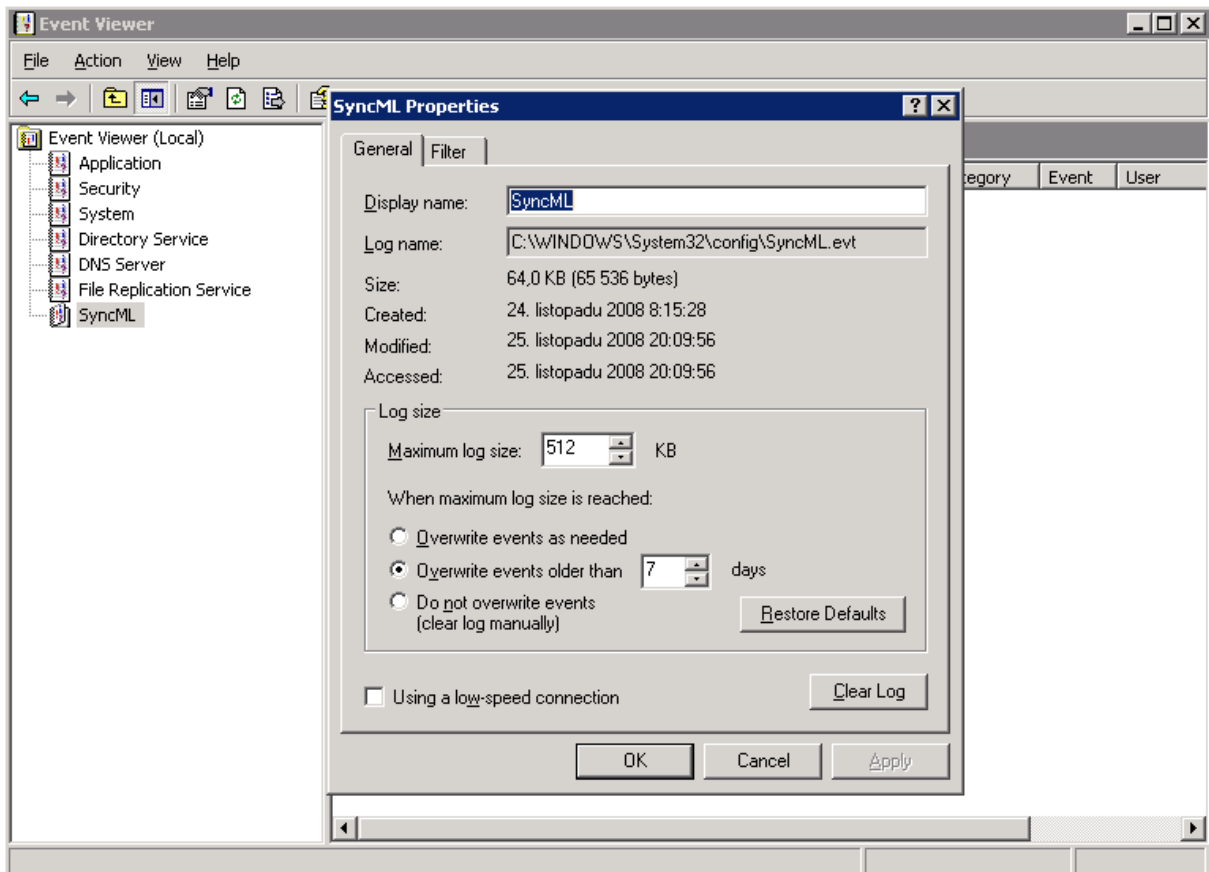
The connection string entered at this stage will also be used at application run-time (unless changed manually, of course). If you decide to use Integrated Authentication, make sure that the account under which the application will be running has sufficient SQL permissions.

3.4.1 SQL Permissions

Application needs to have `db_datawriter` and `db_datareader` permission to created databases for account, which is running application pool, also for SyncML database is needed to grant `Execute` permission for stored procedures. Standard installation of SQL server includes management tools where account and its right can be assigned, for SQL Express 2005 you need to install SQL Express Management Studio, which can be downloaded from Microsoft download website. For SQL commands, see end of this document.

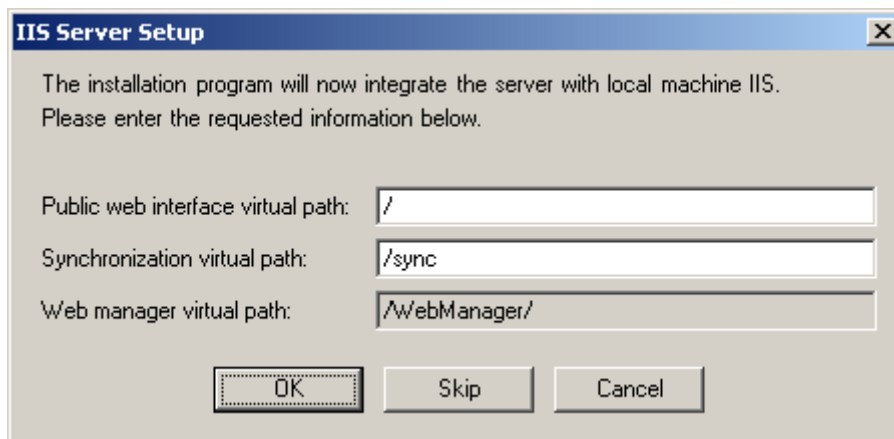
3.5 Creating event log

A new event log named `SyncML` is automatically created by the installer. The log can be accessed using standard tools, e.g. the Event Viewer located in Administrative Tools. It might be necessary to adjust the event log so that it fits your needs. For example the maximum log size, which is by default only 512 kB, and the overwriting policy might need adjustments.



3.6 Configuring IIS

It is possible to let the installer configure IIS so that the server will work out-of-the-box. The user is asked for a virtual path where the public web should be accessible and for the synchronization virtual path (the one that is entered to client devices). Make sure that IIS is properly installed, i.e. Windows has been configured with the Application Server role.

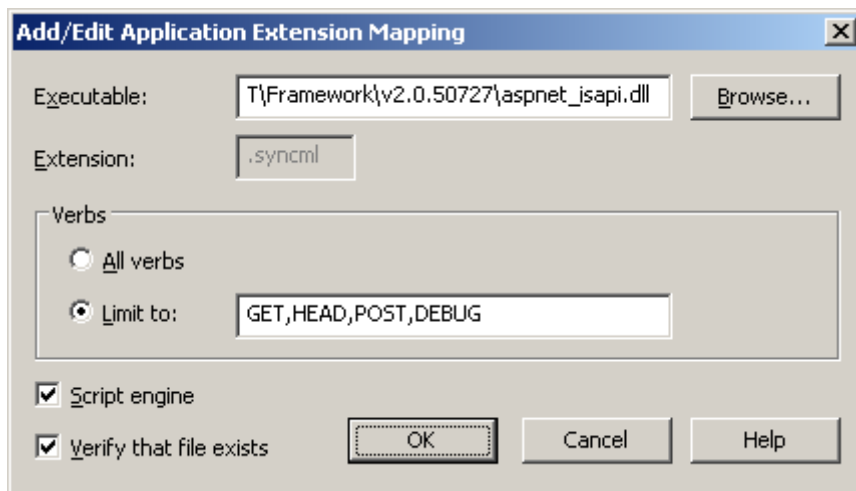


Automatic installation commenced by the OK button creates a new application pool named `SyncMLPool` and a new web site named `SyncMLSite` running in this pool. Important: the site is intentionally set up on port 20304 in order to avoid clashes with other sites that might be running on the machine. Please adjust the port as necessary. The web manager application is accessible only for members of the Administrator group (Integrated Windows Authentication is in effect) and also for application account (if has been created in previous steps). This can be changed in the IIS Manager (in Administrative Tools) by editing `WebManager` directory properties and in addition the fine-grained access control can be adjusted in `TARGET_PATH\WebManager\Web.config` file.

If you decide to configure IIS manually, use the `Skip` button. The following is the list of tasks that should performed:

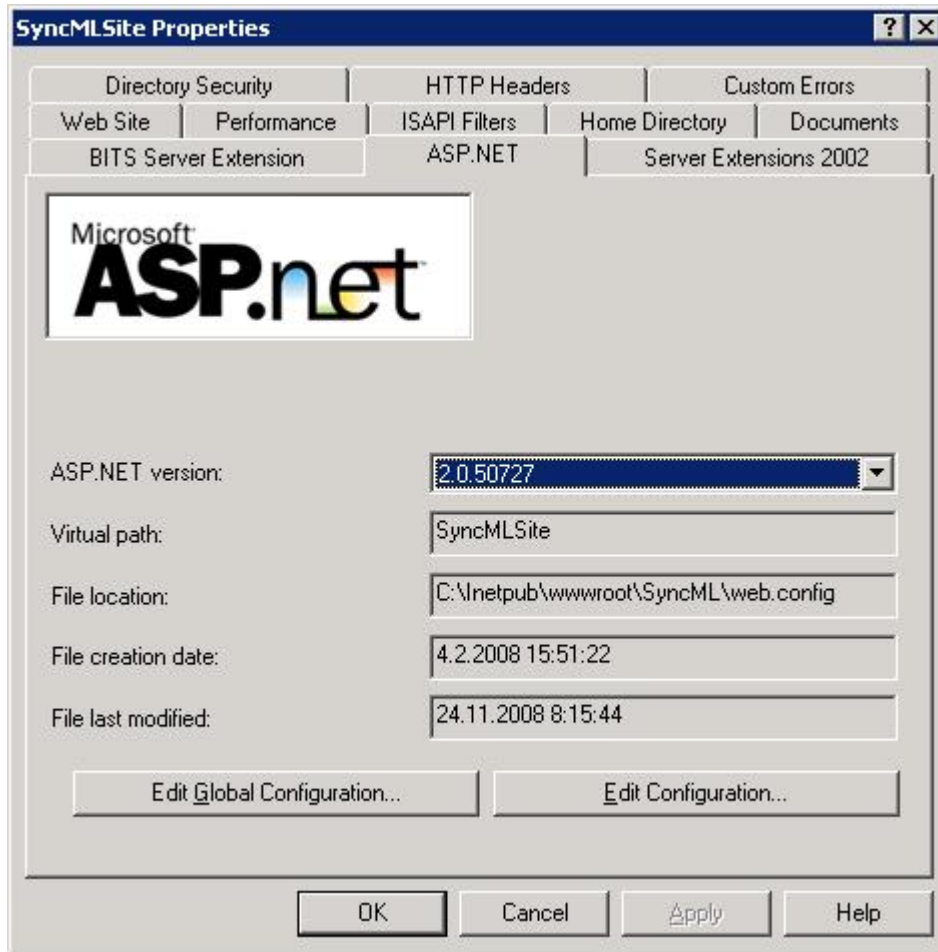
1. A suitable application pool must be chosen for the synchronization server. It can be an existing application pool shared with other applications but keep in mind that it must not be a so called web garden, i.e. the maximum number of worker processes must be set to 1 also all applications in pool must use same version of ASP.NET eg. ASP.NET v2 (which is used by SyncML), otherwise there might unexpected behaviours.
2. A web application must be created for the public web interface, having the `TARGET_PATH` as its root folder, and set to run in the pool chosen in step 1.
3. The ISAPI filter located in `TARGET_PATH\Bin\SyncFilter.dll` must be added to the web site. Furthermore, `SyncFilter.ini` which resides in the same directory should be edited to configure the synchronization virtual path (see the file for details on this).

4. A new extension mapping must be added to the web application created in step 2. Go to the properties of the web application and press the Configuration... button. In the Application Configuration go to the Mappings tab, hit Add... , and fill in the following:
 - o Executable:
C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\aspnet_isapi.dll
(change the path if you've installed Windows to a different folder)
 - o Extension: .syncml
 - o Limit verbs to: GET, HEAD, POST, DEBUG
 - o Script engine: YES, Verify that file exists: YES



5. Default document should be added to the sync directory located under the web application created in step 2. In the sync directory properties go to the Documents tab and add a new Default.syncml default document (you can safely delete the others).
6. The web manager application should be set up. Its root physical directory must be TARGET_PATH\WebManager and it can use either own application pool, or same pool as public interface. You can add SSL or any other security setting to keep this administrative application from prying eyes.
7. User running application pool must have read access to TARGET_PATH (and subdirectories of course) in default environment it only needs to be member of IIS_WPG group. In case of different pools for public part and webmanager part, identity of pools should be the same, or you might need to setup additional database right and directory access right.

Important: If there are more ASP.NET versions installed on the system, it may be necessary to manually switch the web site to ASP.NET version 2.0 in the web site's properties (the ASP.NET tab). This applies to the automatic installation as well!



3.7 Configuring SyncML

In its last step, the installer configures the SyncML engine and the public web interface using the provided information.

SyncML Server Setup

The installation program will now configure the SyncML Server.
You can change these settings any time by editing the 'Web.config' file in the install dir.

SMTP (outgoing) server address:

Server e-mail address (From):

Error reporting e-mail (To):

Tech feedback e-mail (To):

Biz feedback e-mail (To):

Dump communication with devices

Communication dump folder:

Web interface host-to-culture mapping:

Host	Culture
*	en-US

Buttons: Add..., Edit..., Remove

Buttons: OK, Skip, Cancel

The SMTP server address is used by the server when sending out any e-mail. If your SMTP server requires authentication, you will have to configure this manually (refer to chapter [Outgoing e-mail configuration 3.7.1](#) for details). The server e-mail address is the source address used by the server for all outgoing e-mails. The three destination addresses are used when informing about an error, a technical feedback entered via the web, and a business feedback respectively.

If you check the `Dump communication with devices` checkbox, all communication is logged in the specified folder, which must exist and be writable for the account under which the engine is running (NT AUTHORITY\NETWORK SERVICE by default).

Important: The installation program does not create the dump folder. It has to be created manually with appropriate permissions!

The host-to-culture mapping allows you to assign cultures to specific host names. For example your server may have English user interface when accessed via `www.myserver.com` and Czech user interface when accessed via

`www.myserver.cz`. Currently only English and Czech resources are distributed with the server, but adding a new language is just a matter of translating resource files that are located in the `App_GlobalResources` and `App_LocalResources` directories under the `TARGET_PATH`. Refer to the management chapter for more information about localization.

`*` is alias for all undefined languages, so be sure, that there one `*` defined (but it can be only one, if there are multiple entries with `*`, first will be used). Example.: You want the web to be in czech, therefore you add `cs-CZ` with address `*` and next remove `en-US` with address `*`.

If you choose to skip this step or simply wish to change the settings after the installation, edit the `TARGET_PATH\Web.config` file. The file is fairly commented so you should not get lost.

If everything went smooth during the installation, you should be able to navigate to the public and administrative webs, activate the installation, create user accounts, and synchronize devices. Just keep in mind that the automatic IIS installation starts the web site on port 20304 and the server does not work until properly activated (see the next chapter).

3.7.1 Outgoing e-mail configuration

This server requires a working SMTP relay server in order to send out e-mails. If your machine acts as a SMTP relay, i.e. has the Mail server role (the most common setup), the SMTP server address entered during setup or later in the `Web.config` file should be `127.0.0.1` or `localhost`. Just make sure that relaying is permitted at least for IP address `127.0.0.1` because the SMTP Virtual Server service restricts relaying by default.

If you use your ISP's SMTP server, you'll have to know its host name or IP address. Furthermore, if your ISP's server requires authentication, the correct credentials must be given in the `Web.config` file. Locate the `TARGET_PATH\Web.config` file and add the `userName` and `password` attributes as follows:

```
<system.net>
  <mailSettings>
    <smtp deliveryMethod="Network">
      <network
        userName="isp-username"
        password="isp-password"
        host="smtp.your-isp.com"
        port="25" />
      </smtp>
    </mailSettings>
  </system.net>
```

For detailed description of the `mailSettings` element, please refer to MSDN: [http://msdn2.microsoft.com/en-us/library/ms164242\(VS.80\).aspx](http://msdn2.microsoft.com/en-us/library/ms164242(VS.80).aspx)

4 Activation

Compelson SyncML Server must be activated in order to create users, access data stores, and unlock its entire synchronization functionality.

Important: The server does not work unless properly activated!

After installation, point your web browser to the WebManager URL, which is <http://localhost:20304/WebManager/> by default. Click on the `Product activation` link and enter the activation key, user, and company as obtained from Compelson Labs. Clicking the `Activate` button initiates download of an activation file named `SyncML.actfile`. The file will be downloaded from Compelson Labs servers over the Internet so make sure that your server is permitted to make outbound HTTPS connections to <https://secure.mobiledit.com>.

In order to complete the activation, please save the file to the directory indicated on the activation page. It is always the same directory, where the server binaries are located.

You should see your activation information after pressing the `Refresh` button. If the server still claims that it's not activated, verify that `SyncML.actfile` is placed in the server binaries directory (`TARGET_PATH\Bin`). It is the same directory where files like `SyncML.dll` are located.

5 Management

5.1 Configuration files

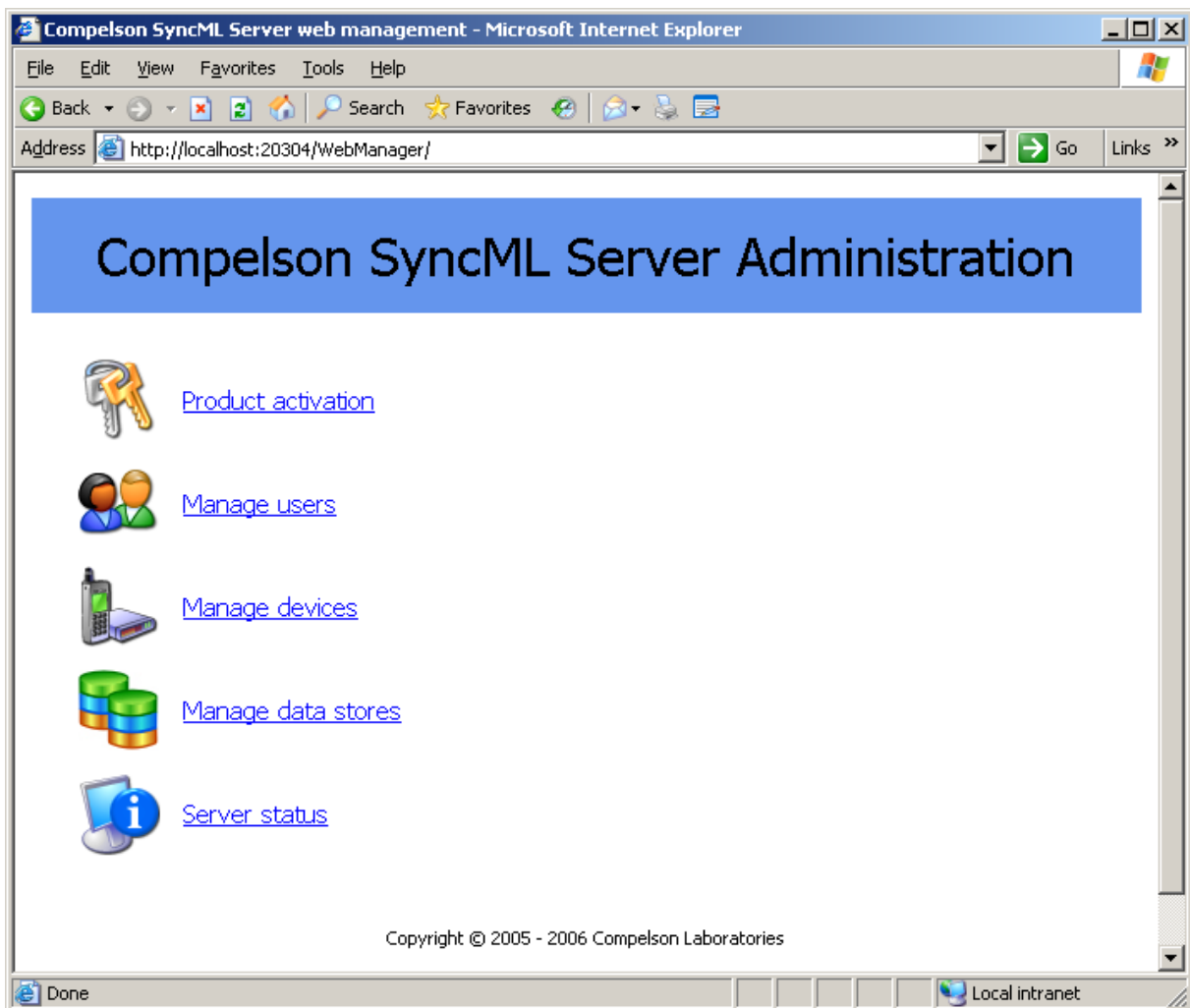
The following is the list of configuration files along with their roles:

- `TARGET_PATH\Web.config`
Contains configuration settings that is common for the synchronization engine and both web applications plus some public web specific settings.
 - e-mail addresses, SMTP server settings
 - database connection strings
 - public web access control
 - public web host-to-culture mapping
- `TARGET_PATH\WebManager\Web.config`
Contains web manager access control (authorization and authentication settings).
- `TARGET_PATH\sync\Web.config`
Contains HTTP handler mapping for the synchronization virtual path (unlikely to be edited).
- `TARGET_PATH\Bin\SyncFilter.ini`
Contains ISAPI filter settings, mainly the synchronization virtual path. The IIS makes it possible to map handlers to file extensions but unfortunately it is not possible to map a handler to a single file. Usually, users expect the synchronization URL to be similar to `http://somehost/sync`, i.e. without the trailing slash. When IIS receives a request for such a URL, it responds with a redirect to `http://somehost/sync/`. This is absolutely OK under normal circumstances, but many mobile phones do not implement the redirect and the synchronization fails. The primary purpose of the ISAPI filter is to translate `http://somehost/sync` to `http://somehost/sync/Default.syncml` before it reaches IIS, thus making it transparent for the device that it in fact posts its request to the modified path. The other purpose of the filter is filtering `100-Continue` HTTP responses that are new to HTTP 1.1 and cause problems in some devices.

5.2 Web manager

The web manager application available by default at the `ROOT/WebManager` URL, where `ROOT` is the public web interface virtual path, can be used to perform various administrative tasks like activating the server, adding and removing users, managing server data stores, and seeing the synchronization engine internals.

The `Server status` link displays a page with the number of currently registered users, both total and per country, and ten most recently registered users. In addition, the number of client devices which have made at least one connection to the server is displayed – total and per manufacturer.

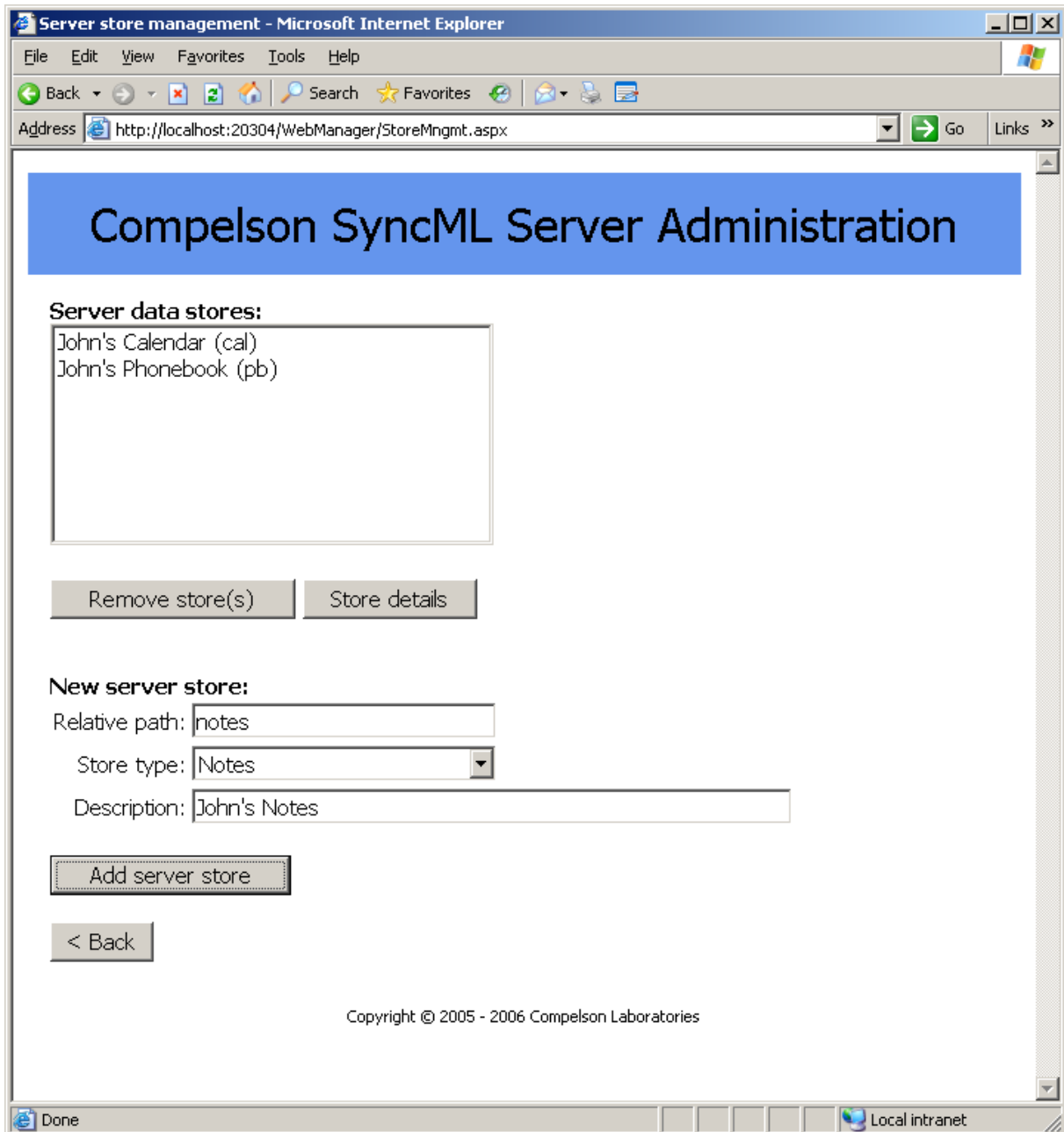


Setting up an account manually involves the following steps:

- Go to `Manage data stores` and create new server data stores of the required types (`Phonebook`, `Calendar`, `Notes`). Note that events (`calendar`) and tasks both use the same data store type, which is named `Calendar`. Relative path is the path that users will have to enter to their devices.

- o Go to `Manage users` and create a new user. Go to user details and assign the new server data stores to him/her (so called bindings). The same server data stores can be assigned to more users simultaneously – in that case they all share the same data and mutually synchronize their devices. Do not forget to check the `User is enabled` checkbox.

At this moment, the new user is all set and can synchronize with his/her client device. Note that when the user registers via the public web, all these steps are automatically performed by the public web application and no administrative intervention is necessary. A few screen-shots illustrating the steps follow.



Compelson SyncML Server Administration

User details:

Name:

Password:

Full name:

E-mail:

Country:

UTC offset:

Date format:

Events and tasks share one store

User is enabled (can log in and synchronize)

Store bindings:

Server store	Client store	Sync type	Last sync anchors			
John's Phonebook (pb)	N/A (template)	TwoWay	N/A N/A	<input type="button" value="Details"/>	<input type="button" value="Delete"/>	<input type="button" value="Reset anchors"/>
John's Notes (notes)	N/A (template)	TwoWay	N/A N/A	<input type="button" value="Details"/>	<input type="button" value="Delete"/>	<input type="button" value="Reset anchors"/>
John's Calendar (cal)	N/A (template)	TwoWay	N/A N/A	<input type="button" value="Details"/>	<input type="button" value="Delete"/>	<input type="button" value="Reset anchors"/>

Note that a valid last sync anchor makes it possible to perform 'fast' synchronization of the respective data store. Resetting the anchor results in falling back to 'slow' synchronization, in which all items are sent by the client and compared one by one with the items present in the server store.

New binding:

Server store:

Sync type:

Copyright © 2005 - 2006 Compelson Laboratories

5.3 Event log

During synchronization, Compelson SyncML Server creates entries in the `SyncML` event log. These entries are of three levels: Information, Warning, and Error.

Information entries inform about the progress and results of synchronization sessions. You can determine e.g. the exact time and date when a specific user started the sync, the data stores involved, number of items transferred etc.

When the server receives incorrect data, such as a malformed command sent by a device, it creates a Warning entry with detailed description of what happened. You can use these entries for example to inform the user that his/her device is not set up correctly.

Should an unexpected internal error occur, the server creates an Error entry and immediately sends an e-mail containing this entry to the address specified during installation. Error entries generally indicate problems with the server itself. They may be a result of misconfiguration or a bug in the server. Please forward the entry to Compelson Labs if you are in doubt and unable to resolve it yourself.

5.4 Communication dump folder

If the server is configured to log all communication with devices to a folder (see chapter 3.7 Configuring SyncML), every message received and sent will be automatically saved into the specified folder. Its structure is organized as follows. Files stored directly in the folder are unrecognized requests. Their names are randomly generated (sometimes referred to by the event log entries) and they come in pairs – one `.txt` file containing HTTP request headers, and one `.raw` file containing HTTP posted data.

When the server is able to decode the request message, it saves it to a file named:

```
<device-id>\<session-id>_<random-number>\rx_<msg-id>.wbxml
```

or

```
<device-id>\<session-id>_<random-number>\rx_<msg-id>.xml
```

depending on the message format (either plain-text XML or binary encoded WBXML). Similarly, request headers are saved in files named:

```
<device-id>\<session-id>_<random-number>\rxhdr_<msg-id>.txt
```

and response messages in files named:

```
<device-id>\<session-id>_<random-number>\tx_<msg-id>.{wb}xml
```

The `<device-id>` is a string used by the device to identify itself – usually its IMEI number, the `<session-id>` is a string assigned to the current session by the device – usually it is a number that is incremented with every sync session, and finally the `<msg-id>` is a 1-based sequence number.

6 Localizing public web

Adding a new language to the public web comprises of translating the resources and adjusting the configuration. Resources are located in `.resx` files in the entire subdirectory tree of `TARGET_PATH`. To add a new language, you should first determine the culture name according to the table in MSDN:

<http://msdn2.microsoft.com/en-us/library/system.globalization.cultureinfo.aspx>.

Then create a copy of each and every `.resx` file named `<page-name>.resx` and name it `<page-name>.<culture-name>.resx`. As an example, consider the resources for the user registration page, which are stored in:

- o `TARGET_PATH\Account\App_LocalResources\New.aspx.resx`
(English)
- o `TARGET_PATH\Account\App_LocalResources\New.aspx.cs-CZ.resx`
(Czech)

In order to add French, take the `New.aspx.resx` file, make a copy of it, and name it `New.aspx.fr-FR.resx`. Now open this copy with Visual Studio and translate all texts in the `Value` column to the target language. Repeat this for every `.resx` file in the entire directory tree.

Visual Studio Express Edition can be downloaded from MSDN for free. The URL is: <http://msdn.microsoft.com/vstudio/express/>.

After all `.resx` files have been added, change the host-to-culture mapping, which is done by editing the `TARGET_PATH\Web.config` file. Details about the host-to-culture mapping can be found in chapter 3.7 Configuring SyncML.

7 Support

Compelson SyncML Server support can be reached at syncml@mobiledit.com. We would be more than happy to provide you with assistance related to setting up the server.

8 Appendix

8.1 SQL Permission Fix Commands

SQL command for granting database permissions (*replace \$DBNAME\$ with appropriate database name*):

```
USE $DBNAME

IF NOT EXISTS (SELECT * FROM master.dbo.syslogins WHERE loginname = 'NT
Authority\Network Service')

    EXEC sp_grantlogin 'NT Authority\Network Service'

IF NOT EXISTS (SELECT * FROM dbo.sysusers WHERE name = 'NT
Authority\Network Service')

    EXEC sp_grantdbaccess 'NT Authority\Network Service'

EXEC sp_addrolemember N'db_datareader', 'NT Authority\Network Service'
EXEC sp_addrolemember N'db_datawriter', 'NT Authority\Network Service'

GO
```

SQL command for granting execute permission

```
USE SyncML

GRANT EXEC ON SCHEMA::dbo to [NT AUTHORITY\NETWORK SERVICE]

GO
```