

**Easy! Software LLC**

---

**Easy!Flow**  
**Installation and Configuration**  
**Guide**  
**(Windows Edition)**

**Author(s):** Easy! Software LLC  
**Version:** 1.4  
**Status:** Released May 1, 2009

Easy! Software LLC  
**General Business Information**

This document contains information, which is under protection of copyright. Any distribution, revision and publication of this document are prohibited. The usage of this document must be under the authorization from Easy! Software LLC,

## Contents

<b>1. Quick Installation.....</b>	<b>3</b>
1.1 Prerequisite .....	3
1.2 Run installation.....	3
1.3 Verify your installation .....	4
1.4 Enable Email Notification (optional).....	6
<b>2. Creating your own product .....</b>	<b>7</b>
<b>3. Deep dive into Easy!Flow system configuration.....</b>	<b>9</b>
3.1 Configuring Easy!Flow Tracking Ticket Subsystem (Easy!TTS) .....	10
3.2 Configuring Easy!Review, Peer Review Subsystem .....	11
3.3 Configuring SCM Client.....	11
3.4 Add new repositories to Easy!Flow .....	12
<b>4. A comprehensive configuration example.....</b>	<b>13</b>

## 1. Quick Installation

This document covers the installation and configuration of Easy!Flow, which is an integrated process management solution. The examples used in this document are from a server running Windows XP operating system unless otherwise specifically mentioned.

### 1.1 Prerequisite

Xampp for Windows 1.6.6a, which includes apache and mysql, is required to be installed under C:\Easy\xampp (If you want to install Easy!Flow on disk partition D, the Xampp installation path should be D:\Easy\xampp). Xampp for Windows is a freeware which user can download from <http://www.apachefriends.org/en/xampp-windows.html>

Direct download path:

<http://sourceforge.net/projects/xampp/files/XAMPP%20Windows/1.6.6a/xampp-win32-1.6.6a.exe/download>

The contents of this package are in a folder named xampp. Extract this folder and put it into <drive>:\Easy.

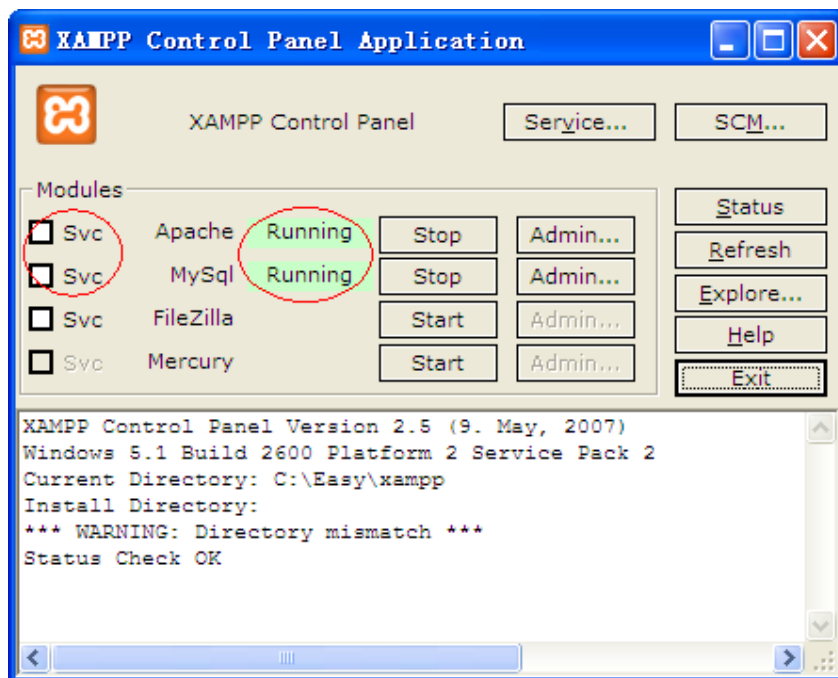
We can also help to provide free Xampp for Windows 1.6.6a installation package separately per user's request.

### 1.2 Run installation

Easy!Flow has been packaged into an executable file. Run EasyFlow\_Win.exe to install Easy!Flow on your machine. Easy!Flow has to be installed in <drive>:\Easy; for example C:\Easy.

After the installation is finished, Apache and MySQL are started automatically and Easy!Flow Control Panel is shown. If the control panel does not open, you can open it manually by opening <drive>:\Easy\xampp\xampp-control.exe

Control Panel shown below is used to start/stop Apache and MySQL processes. Please make sure they are both in Running state. If not, click "Start" to start them both. If you want Apache and MySQL to run automatically after each system reboot, stop them in Control Panel, check both 'Svc' on, and start them again.



Now you have installed Easy!Flow on your hard disk "C:\". Apache and MySQL, which are required by Easy!Flow, both reside in C:\Easy\xampp, and Easy!Flow certified Perl is located at C:\Easy\Perl. Original Password of mysql root user is 'root', which should be changed immediately after installation.

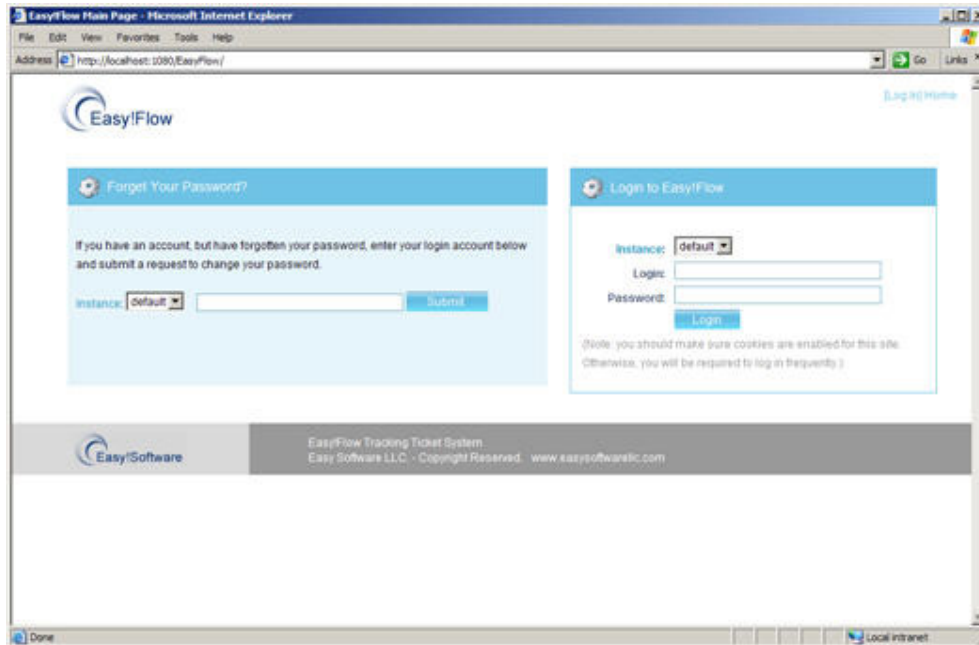
If you want to enable ticket dependency graph, run vcredist\_x86.exe under C:\Easy root directory to install Microsoft Visual C++ Redistribution Package.

### 1.3 Verify your installation

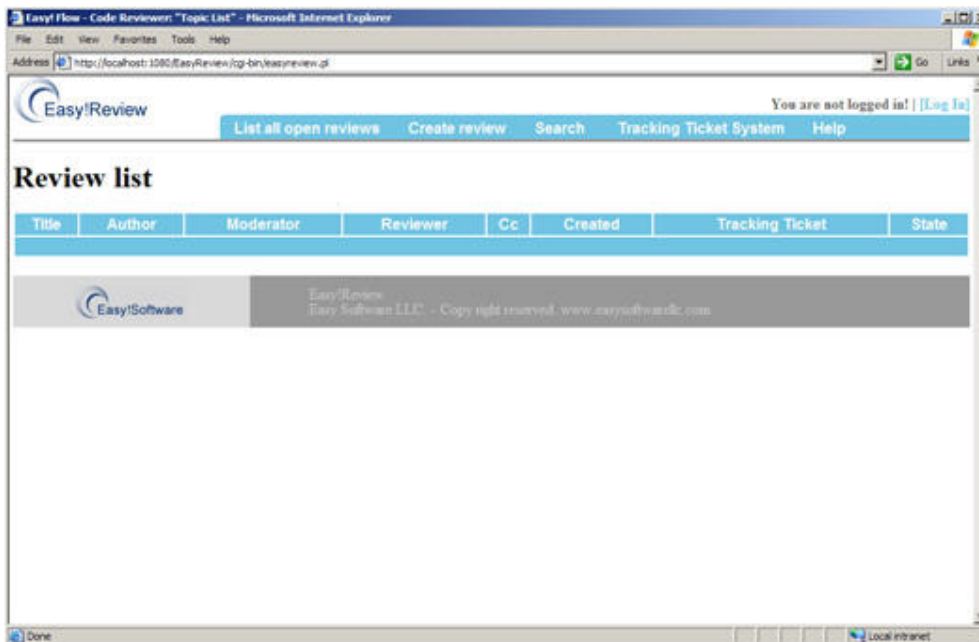
After the installation, open Easy!Flow login page via <http://localhost:1080/EasyFlow/>. See below. (Note: If you are not on the local machine of your Easy!Flow installation, please use its url instead, such as <http://abc.somecompany.com:1080/EasyFlow/>)

The default Admin account is  
Username: easyflow@easysoftwarellc.com  
Password: easyflow

# Easy!Flow Administration Documentation

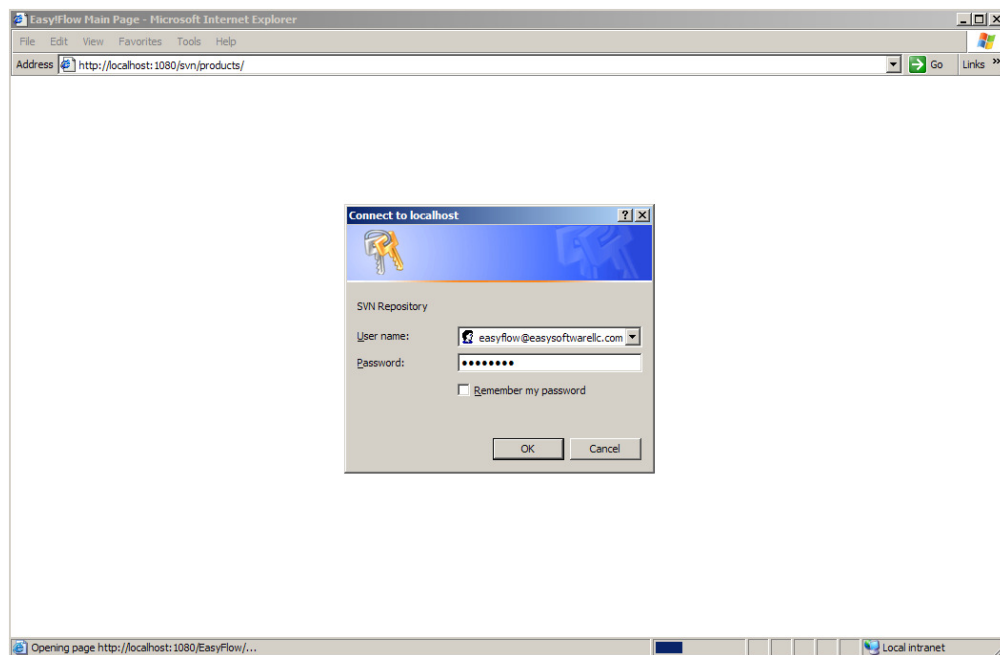


Likewise, you can open the Easy!Review home page at <http://localhost:1080/EasyReview/cgi-bin/easyreview.pl>. See the screenshot shown below.



Easy!Flow creates a default Subversion repository for you. You can access it through <http://localhost:1080/svn/products> with an Easy!Flow account (username/password). See the screenshot below.

## Easy!Flow Administration Documentation



Apache runs at port 1080 by default to avoid conflict with other applications. You can change it in `C:\Easyxampp\apache\conf\httpd.conf`. (by editing line: “Listen 1080”)

Now, you have finished a quick installation with default settings of Easy!Flow. The next chapter will guide you through the steps of creating your products in Easy!Flow.

### 1.4 Enable Email Notification (optional)

By default, Easy!Flow Tracking Ticket System disables Email Notification. If you have an SMTP server running on port 25 at server side, you can enable it by setting Site Admin->Parameters->Email->mail\_delivery\_method to “smtp”. If you have an SMTP server running on another machine, you must set the address of that machine in Site Admin->Parameters->Email->smtpserver.

We recommend the SMTP virtual server that is part of IIS, which is a standard component on Windows XP and Windows Servers. You will need your Windows installation CD. Use Add or Remove Programs in the Windows Control Panel. Select Internet Information Services (IIS) and then click Details. Check SMTP Service and then click OK. The installation process will prompt you for your Windows CD and will install the SMTP virtual server.

After installing the SMTP server (and possibly rebooting), configure it by following these steps:

- 1) In the Control Panel, choose Administrative Tools, and then Internet Information Services.
- 2) Open the node for your computer, right-click the Default SMTP Virtual Server node and choose Properties.
- 3) In the Access tab, click Connection.

- 4) Select only the list below, click Add and add the IP 127.0.0.1. This restricts connections to just the local computer, i.e., localhost. Close the Connection dialog box.
- 5) Click Relay and repeat Step 5 to allow only localhost to relay through this server.

For troubleshooting, you might want to turn on logging. In the General tab, check Enable logging.

Note: You may need to manually start SMTP service in above Administrative Window after installation.

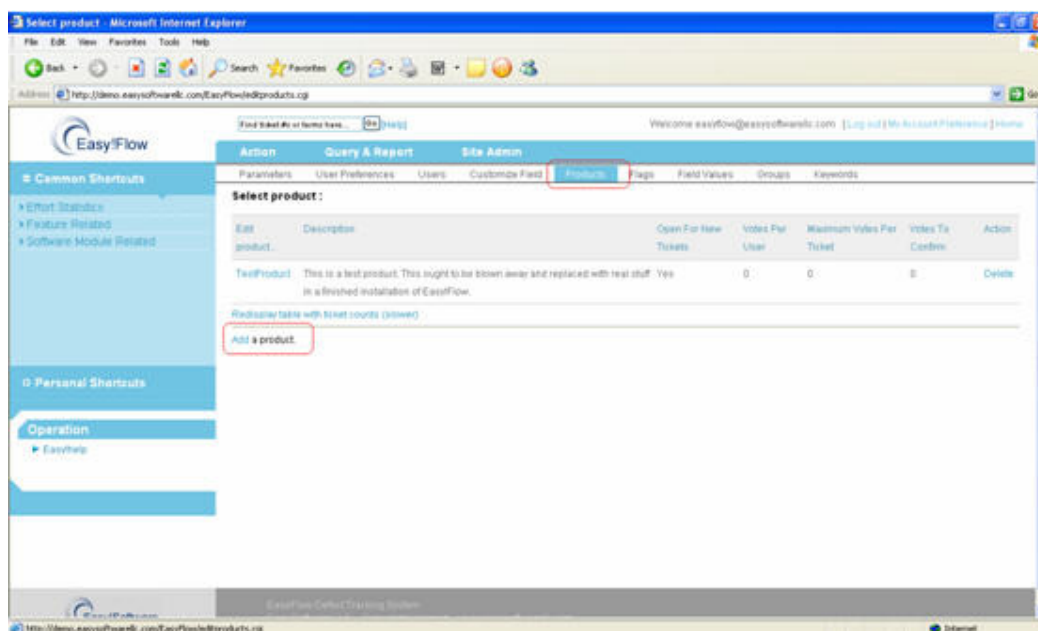
## 2. Creating your own product

A default product, TestProduct, and a default Instance, Default, are created after the default installation. In this chapter we are assuming the new product/project that you are about to create uses the default Instance and SVN repository root directory. The subsequent chapters in this document will guide you through how to create new product(s), SVN repository directories, and new Instance(s).

First, with admin account, you “add” a new product in Easy!Flow, and fill in the product related info.

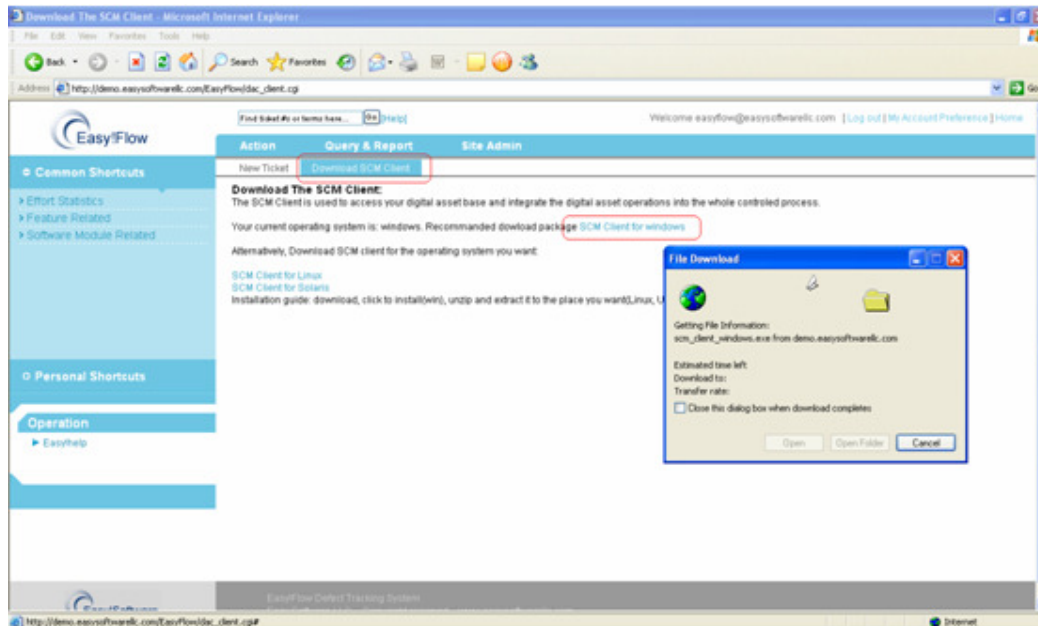
Note:

- 1. If you are not familiar with the system, you may just rename the default product name and then modify the product related info to reflect your product.
- 2. The Easy!help under Site Admin->Products provides detail explanation of each product related info.



Second, download and install SCM Client. With SCM Client, you will be able to import files into

Subversion repository for the product you have just created.



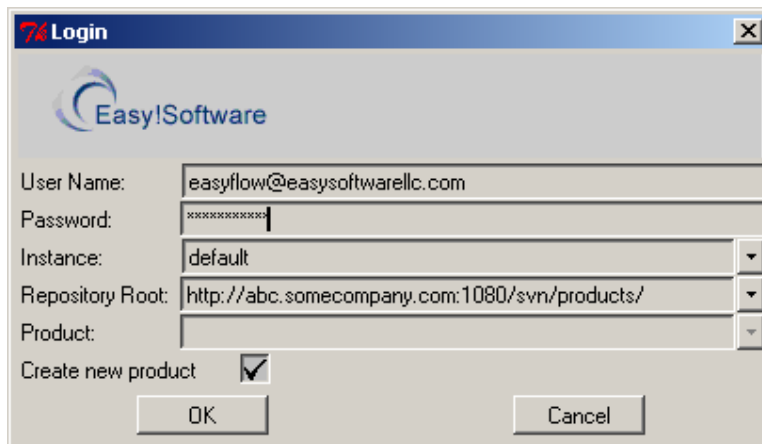
Third, edit C:\Program Files\EasySoftware\Easy Flow SCMClient\site.conf (or in the installation location you specified). Change two configuration items per your actual situation. See example below.

In site.conf file:

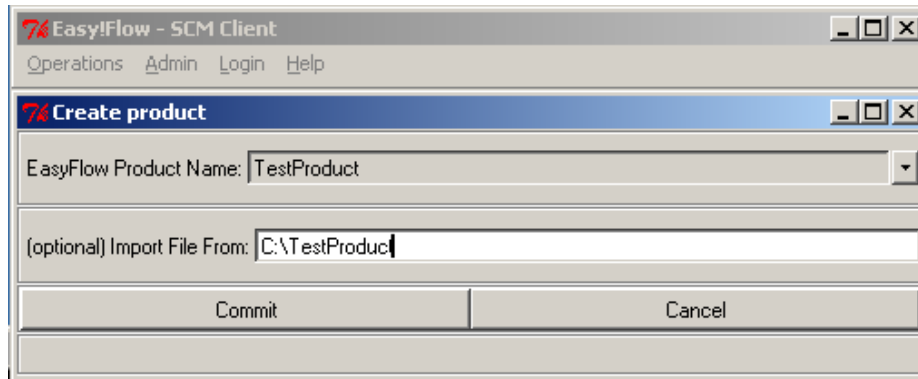
```
# Remote SVN configuration.  
repos_url=http://abc.somecompany.com:1080/svn/products  
# Remote Easy!Flow Tracking Ticket system  
easyflow_url=http://abc.somecompany.com:1080/EasyFlow/
```

Finally, create the new product in SVN repository via SCM Client.

Log in with admin account with “Create new product” option box checked in login form.



Once logged in, click “Create product” in “Admin” menu, choose a product you have created earlier, and specify the location that contains initial files to be imported to the new product in SVN. SCM Client will create the repository schema, import the initial files and construct mapping relationship with the product you have created.



Alternatively, you can create a product in repository using command line on your repository server. See help file of SCM Client for details.

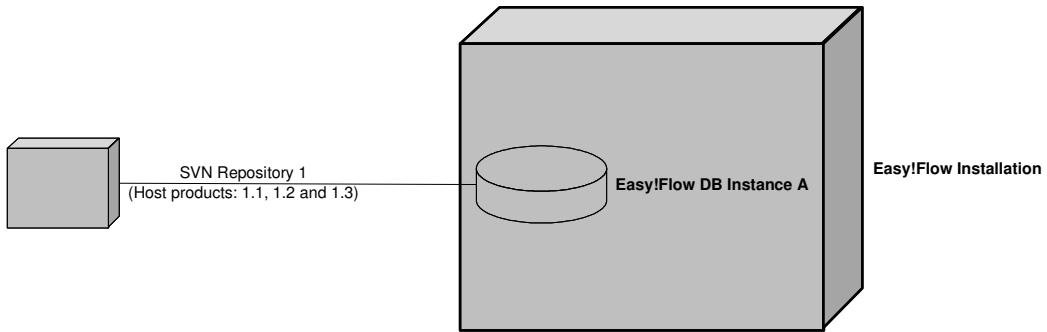
More detailed and sophisticated configuration cases will be shown in subsequent chapters.

### 3. Deep dive into Easy!Flow system configuration

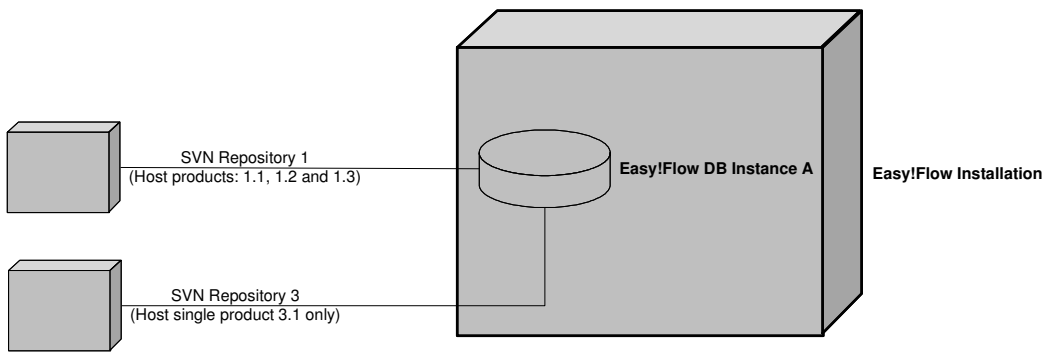
Besides the quick installation with default configuration mentioned above, which is much suitable for a starter, Easy!Flow can support sophisticated application models for advanced users. The diagrams (3.1, 3.2 and 3.3) below are examples, which show some possible Easy!Flow setups. The basic rules are: (1) one single Easy!Flow installation can support one or multiple DB instances; (2) one DB instance can support multiple source-file repositories; (3) one source-file repository can host the source files of one or more products in one DB instance.

Note: Easy!Flow is capable of managing multiple product/project sets in one installation. The Instances are database, which contains one or more products/projects of a similar development process. You can treat an Instance as a methodology of development process. For example, your company may have 10 development projects, of which some may be more mission critical applications than others, yet some may just be small improvements for internal tools. In this example, you may design and configure three Instances; one for those projects require more stringent development process, the 2nd one for those require less stringent process, and the 3rd one for some sort of, say, Agile process.

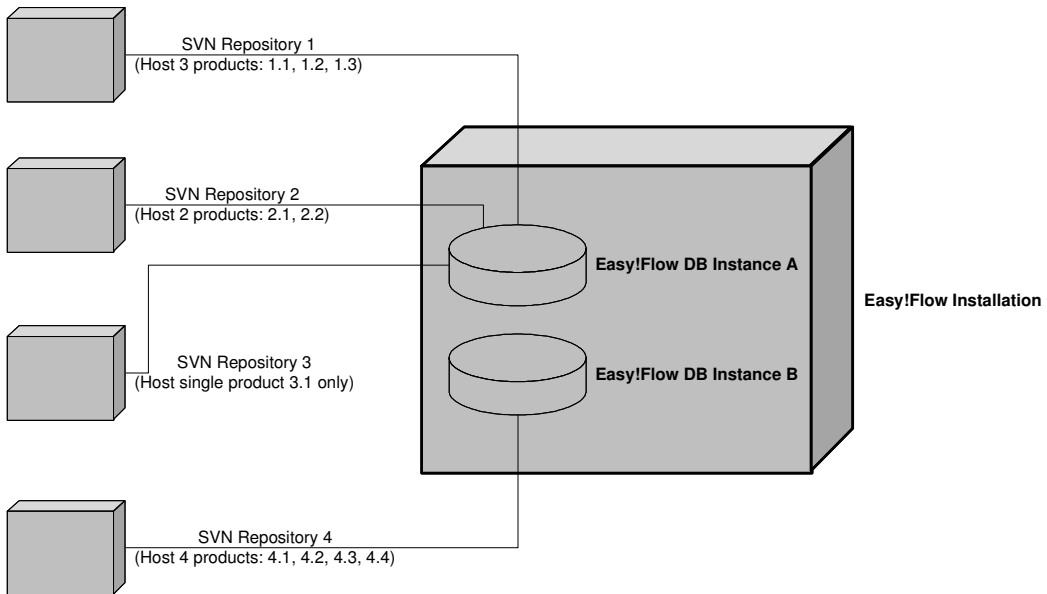
# Easy!Flow Administration Documentation



3.1 -- One repository using one Easy!Flow DB instance



3.2 -- Two repositories using one Easy!Flow DB instance



3.3 – Three repositories using two Easy!Flow DB instances

## 3.1 Configuring Easy!Flow Tracking Ticket Subsystem (Easy!TTS)

The configuration files for Easy!TTS are C:\Easy\EasyFlow\localconfig.\*. For example, you may

## Easy!Flow Administration Documentation

have localconfig.instanceA and localconfig.instanceB for Instances A and B, respectively.

If you'd like to create a new Instance, you can follow the steps below:

1. Create database instance and related user accounts for your new Instance.
2. `set PROJECT=instancename`
3. `cd \D C:\Easy\EasyFlow`
4. Run "`\Easy\Perl\bin\perl checksetup.pl`" (Note: You need ActivePerl installed to run .pl file.)
5. Edit localconfig.instanceName to change database related info.
6. Run "`\Easy\Perl\bin\perl checksetup.pl`" again.

Chapter 4, step 5 provides more details about creating a DB instance.

If you don't want a particular Instance to be shown on Easy!Flow login page, just remove its corresponding flag file show below:

```
C:\Easy\EasyFlow\EasyFlow_Instance_<instance name>
```

### 3.2 Configuring Easy!Review, Peer Review Subsystem

The configuration file for the peer review subsystem is

C:\Easy\EasyReview\easyreview.conf.<instance name>, if you have created a DB instance named "instance1" as referred in previous chapter 3.1, then it's related Easy!Review configuration file will be easyreview.conf.instance1. You can add the repositories related to "instance1" to this configuration file as follows:

Edit C:\Easy\EasyReview\easyreview.conf.instance1 file to add repository URL.

```
@valid_repositories =
{
    # This example is the syntax used for specifying a Subversion
    # repository, which is simply the subversion repository URL
    # prefixed by svn:
    'svn:http://abc.somecompany.com:1080/svn/products',
    ...
}
```

### 3.3 Configuring SCM Client

To configure Easy!Flow SCM Client application, just edit the configuration file "site.conf" in the SCM Client installation directory. You need to specify three items there: repos\_url, easyflow\_url and instance so that SCM Client knows where to find the repository, where to locate your tracking ticket system and what is the instance in tracking ticket system which you are using. Below is an example of the configuration file.

```
#####
# Common site configuration file
```

```
#####

# Remote Easy!Flow Tracking Ticket System.
easyflow_url=http://abc.company.com:1080/EasyFlow/

# The DB instance used in Easy!Flow Tracking Ticket System.
# Separate instances with ';'.
# For example: instance=default;instance1
instance=instance1

# Remote Repository.
# Separate repositories for different instances with '|'.
# Separate repositories in the same instance with ';'.
# For example: repos_url=<inst1_repos1>;<inst1_repos2>|<inst2_repos1>;<inst2_repos2>
repos_url=http://abc.company.com:1080/multi-repos-root/repos1/

# Log files
svn_stderr_log=svn_error_file
svn_stdout_log=svn_log_file

# site.conf configure file end here.
#####
```

Some environment variables listed below are also supported allowing user to define some personal default setting.

- REPOS\_ROOT: default repository.
- PROD\_NAME: default product name in repository.
- INSTANCE: the Easy!Flow DB instance currently used.

### 3.4 Add new repositories to Easy!Flow

In order to enforce a user defined process, the Easy!Flow is tightly integrated with Subversion repositories. This is achieved via SVN server side trigger, which is located in the SVN repository directory. The trigger files are located in the subdirectory “hooks”. In the default installation, Easy!Flow server-side trigger files are in the following location:

C:\Easy\SVN\_REPOSITORY\products\hooks\

For each new repository added, you need to copy those trigger files from C:\Easy\SVN\_REPOSITORY\products\hooks to the “hooks” subdirectory of the new repository.

Below is the content of the trigger configuration file (trigger.conf). The mandatory items you need to specify are “repos\_url”, “easyflow\_url”, and Instance name; these are the same as SCM Client configuration file referred previously in chapter 3.3.

```
#####
```

```
# Site common site configuration file for repository triggers.
#####

# Local SVN configuration.
local_svn= \Easy\svn\bin\svn
local_svnlook=\Easy\svn\bin\svn

# Remote Repository
repos_url=http://abc.company.com:1080/multi-repos-root/repos1

# Easy!Flow Tracking Ticket System
easyflow_url=http://abc.company.com:1080/EasyFlow/

# Easy!Flow instance
instance=instance1

# auth_mode can be integration or standalone
# integration: used if there is Easy!Flow Tracking Ticket integration
# standalone: used if no Easy!Flow Tracking Ticket integration
auth_mode=integration

# used when auth_mode is set to integration, to specify the scm group
# in Tracking Ticket system.
scm_user_group=admin

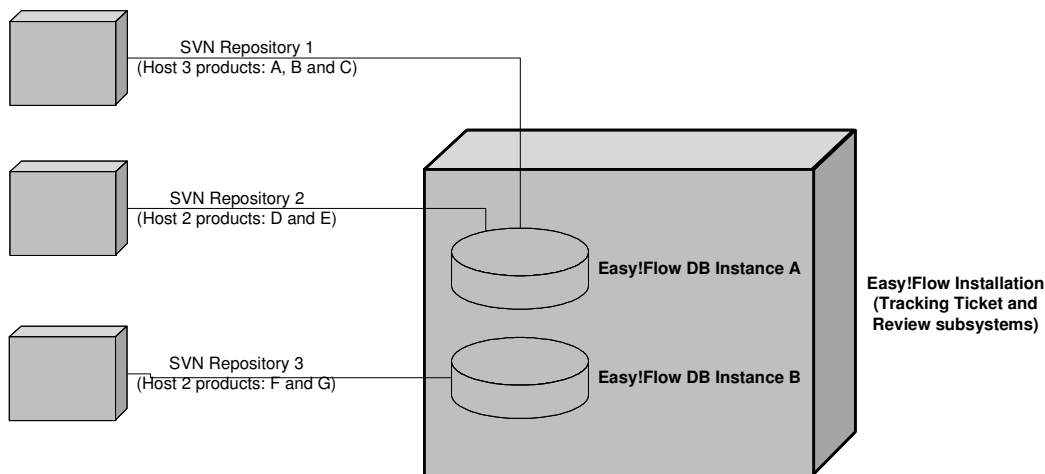
# used when auth_mode is set to standalone, to specify the scm users here.
# such as, scm_users=larry,john,william the user accounts are SVN login account.
scm_users=

# trigger configuration file ends here
#####
```

#### 4. A comprehensive configuration example

In this chapter you will be guided through the steps for a comprehensive Easy!Flow configuration example. Although not all organizations or development teams need such a sophisticated application model, it nevertheless covers all points in system configuration. Of course, you just skip the steps that don't apply to your specific configuration.

Assuming that your organization creates three repositories, say repos1, repos2 and repos3, with repos1 hosts products A, B and C, repos2 hosts products D and E, and repos3 hosts products F and G. We further assume that products A through E use similar development process, therefore, similar field definitions, and developed by the same group of developers, whereas products F and G use a different process. In this case, repos1 and repos2 will map to Easy!Flow DB instanceA, and repos3 maps to DB instanceB. Please refer to the diagram below.



The following steps will configure such an application model. (Assuming that you have finished Easy!Flow default installation on your server and your server has a domain name called abc.company.com)

**1. Create the three repositories**

```
mkdir C:\svnroot\multi-repos-root
mkdir C:\svnroot\single-repos-root
C:\Easy\svn\bin\svnadmin create C:\svnroot\multi-repos-root\repos1
C:\Easy\svn\bin\svnadmin create C:\svnroot\multi-repos-root\repos2
C:\Easy\svn\bin\svnadmin create C:\svnroot\single-repos-root\repos3
```

**2. Configure Apache configuration file to add repository access points**

Easy!Flow Apache configuration file is located at C:\Easy\xampp\apache\conf\extra\Easy.conf  
 Edit the content of this file:

```
<Location /multi-repos-root/>
DAV svn
SVNParentPath C:/svnroot/multi-repos-root/
AuthzSVNAccessFile C:/svnroot/multi-repos-root/accessfile

AuthName "SVN Zone"
AuthType Basic

AuthMySQLUser easy1
AuthMySQLPassword easy1
AuthMySQLDB instance1
AuthMySQLUserTable profiles
AuthMySQLNameField login_name
```

```

AuthMySQLPasswordField cryptpassword
AuthMySQLPwEncryption none
AuthMySQLUserCondition "disabledtext=""
#Satisfy Any
require valid-user
</Location>

<Location /single-repos-root/>
DAV svn
SVNParentPath C:/svnroot/single-repos-root/
AuthzSVNAccessFile C:/svnroot/single-repos-root/repos3/accessfile

AuthName "SVN Zone"
AuthType Basic

AuthMySQLUser easy2
AuthMySQLPassword easy2
AuthMySQLDB instance2
AuthMySQLUserTable profiles
AuthMySQLNameField login_name
AuthMySQLPasswordField cryptpassword
AuthMySQLPwEncryption none
AuthMySQLUserCondition "disabledtext=""
#Satisfy Any
require valid-user
</Location>

```

Note: Apache needs to be restarted for the above setting to take effect. For example, you can restart it in EasyFlow Control-Panel or invoke “C:\Easy\xampp\bin\apachectl -k restart” to restart it.

### 3. Create access files needed for accessing files specified in step2

The access files are used in fine-grained permission control in the directory level. Normally, you can just set to “\*=rw”, which means every Easy!Flow registered user has read/write permission. Easy!Flow triggers will protect common baseline branch and official tags in repository from being interpolated by individuals even “\*=rw” is specified. Step 4 below will show you how to setup those triggers.

Edit the content of C:\svnroot\multi-repos-root\accessfile (shared by repos1 and repos2):

```

[repos1:/]
* = rw
[repos2:/]
* = rw

```

## Easy!Flow Administration Documentation

Edit the content of C:\svnroot\single-repos-root\repos3\accessfile is solely used by repos3.

```
[/]  
* = rw
```

### 4. Setup and configure Easy!Flow repository triggers

Copy Easy!Flow default triggers to the repository directories created in step 1:

```
copy C:\Easy\SVN_REPOSITORY\products\hooks\* C:\svnroot\multi-repos-root\repos1\hooks\  
copy C:\Easy\SVN_REPOSITORY\products\hooks\* C:\svnroot\multi-repos-root\repos2\hooks\  
copy C:\Easy\SVN_REPOSITORY\products\hooks\* C:\svnroot\single-repos-root\repos3\hooks\
```

Edit trigger.conf in the above three “hooks” directories. Take repos1 as an example:

```
# Remote Repository  
repos_url=http://abc.company.com/multi-repos-root/repos1  
  
# Easy!Flow Tracking Ticket System  
easyflow_url=http://abc.company.com/EasyFlow/  
  
# Easy!Flow instance  
instance=instanceA
```

### 5. Create Easy!Flow DB instances

```
C:\Easy\xampp\mysql\bin\mysql -u root -p
```

```
mysql> create database instanceA;  
Query OK, 1 row affected (0.04 sec)
```

```
mysql> create database instanceB;  
Query OK, 1 row affected (0.00 sec)
```

```
mysql> grant all privileges on instanceA.* to easy1@'%' identified by 'easy1' with grant option;  
Query OK, 0 rows affected (0.12 sec)
```

```
mysql> grant all privileges on instanceB.* to easy2@'%' identified by 'easy2' with grant option;  
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> flush privileges;  
Query OK, 0 rows affected (0.16 sec)
```

### 6. Setup the DB instances in Easy!Flow

```
set PROJECT=instanceA  
cd \D C:\Easy\EasyFlow
```

```
\Easy\Per\bin\perl checksetup.pl
```

Edit localconfig.instanceA

```
$db_host = 'localhost';      # where is the database
$db_name = 'instanceA';     # name of the SQL DB instance
$db_user = 'easy1';        # user to attach to the SQL DB instance
$db_pass = 'easy1';
$db_driver = 'mysql';
```

Run `\Easy\Per\bin\perl checksetup.pl` again for the above configuration to take effect.

The setup for instanceB is similar to the above procedure for instanceA.

### 7. *Configure Easy!Review subsystem for the newly create repositories*

Edit C:\Easy\EasyReview\easyreview.conf.instanceA and

C:\Easy\EasyReview\easyreview.conf.instanceB to add corresponding repositories so that review subsystem knows where to fetch source files to generate review package.

In C:\Easy\EasyReview\easyreview.conf.instanceA

```
@valid_repositories =
(
  'svn:http://abc.company.com/multi-repos-root/repos1',
  'svn:http:// abc.company.com /multi-repos-root/repos2',
);
```

In C:\Easy\EasyReview\easyreview.conf.instanceB

```
@valid_repositories =
(
  'svn:http:// abc.company.com /single-repos-root/repos3',
);
```

### 8 *Checkup*

You should be able to access the newly created three repositories through the URLs below:

```
http://abc.company.com/multi-repos-root/repos1/
http://abc.company.com/multi-repos-root/repos2/
http://abc.company.com/single-repos-root/repos3/
```

Note: You access the integrated repositories with the unified user accounts in Easy!Flow. That means you need only one account to log into tracking ticket system, conduct review and access repository directly or through SCM Client. That account is created and managed by Easy!Flow tracking ticket system. A default administrator account is created during default installation described in chapter 1.