Flex Profile Kit 5.0

Flex Framework 2.0

By
Dennis Damen
Magnar Johnsen
Jeroen van de Kamp

Gouda 2005
Log•in Consultants

Author: J. P. van de Kamp

j.kamp@loginconsultants.nl

Version: 5.0

Index

1. Introduction

In a typical Windows NT environment, whether it is based on a Client-Server or a Server-Based Computing architecture, choosing the right profile strategy is never easy.

Many organizations prefer the concept of roaming profile for those essential personal settings. But, as always, managing roaming profiles consumes a lot of resources. Because the profile content is controlled by the users themselves, roaming profiles remain a fragile component to administer and manage.

In many MetaFrame environments applications are pooled in application silos. (Separate sets of servers host different applications.) In environments like these, users simultaneously logon on to different servers to use their applications. When using a single roaming profile, profile corruption or loss of personal settings is bound to happen. When the user's roaming profile is updated and loaded on different servers at the same time, profile-related problems increase dramatically.

Windows 2003 allows a separate profile path for terminal servers via a GPO. This is a fine solution in small environments. However, separate profile paths for each silo have a big impact on the number of profiles you must manage. If a site has three separate application silos and 5000 users, that could possibly mean that 15000 terminal server profiles are created.

Mandatory profiles seem to be the appropriate answer to issues mentioned. Properly configured mandatory profiles are very fast, easy to manage, and cannot be corrupted. Mandatory profiles are (from an administrator's point of view) a very robust component in a Windows NT environment.

There is only one big disadvantage of mandatory profiles—no personal registry settings are saved. It's easy to redirect the profile folder content (such as My Documents and Application Data) to the user's homedrive. This allows personal files that normally reside in the profile folder to be saved in the user's homedrive. However, personal registry information cannot be redirected or saved when using a mandatory profile.

Nowadays, personal settings are considered essential. It is almost impossible to create a user-friendly working environment with mandatory profiles. Until now, mandatory profiles were only suitable for users with generic activities.

This document explains how to setup a "Flex Profile" solution. Several strategies will be highlighted, from very simple to more advanced solutions for finer control. The focus of this document lies in profiles part of Server-Based Computing environments (like Citrix MetaFrame Farms), but the Flex Profiles concept also works on workstations in a traditional client-server environment. To setup a Flex Profile solution effectively, only very basic scripting skills are needed (to edit the logon and logoff scripts). Please take your time to digest this document first. All the information you need is here to get you started with Flex Profiles.

1.1 Flex Framework 2.0

Flex Profile Kit 4.0 introduces the **Flex Framework 2.0**. This is a complete and fully functional framework based on several tools (like Dennis Damen's FlexRefresh) and the valuable feedback posted after the release of FPK 3.0 in the Login Consultants tooling forum; http://portal.loginconsultants.nl/forum/index.php?board=16.

One of the biggest advantages of the Flex Framework is that the setup and configuration process of Flex Profiles is dramatically simplified. In addition, many known issues reported in FKP 3.0 and 4.0 are fixed and brand new features are introduced to improve performance, flexibility and ease of deployment.

With the introduction of the Flex Framework, the Flex Profile concept has become mature and a serious alternative to any other standard or commercial profile management solution. Here is a quick run-down on the key features of FPK 5.0 & Flex Framework 2.0:

- **No license costs.** FPK and the Framework are completely free to use in any environment. The FPK is a nothing more than a concept, the Framework is based on some "intelligent" vbscript and freeware/opensource tooling.
- **No backend servers required.** There is no need for expensive SQL or IIS servers to run Flex Profiles. Since the profile data is stored in the users homedrive and the configuration of Flex Profiles is done with a few INI files, Flex can be introduced in any Windows environment without investments in additional hard- or software.
- **Any application is supported.** A common misconception is that the FPK is limited to Office 2003, since it is based on the profile wizard from the 2003 Office Resource Kit. This is *not* true; Flex Profiles can be configured for every Office version or any other type of application.
- **FPK is lightning FAST.** When you use the Framework all requirements for optimal performance are automatically met. Using a mandatory profile makes a big difference in loading times in comparison to roaming profiles, and FKP loads or saves within seconds!
- **Support for profile files & folders.** Not only registry information, also files and folders from the profile can be managed with Flex Profiles. Folders like "Application Data" can be processed though the Flex Profile Kit; this will increase loading times but will still outperform a roaming profile.
- **Robust operation.** FPK has proven to be a very stable profile solution, only when configured incorrectly problems may arise. The documentation will provide some basic guidelines to help setting up a problem free and reliable configuration for Flex Profiles.
- **Simplified configuration.** Before the introduction of the Flex Framework, some basic scripting skills were required. Flex Framework 1.0 lets you tailor a fully managed profile solution without scripting in no-time. 95% of the Flex Framework is configured through INI files!
- **Configuration per application.** The Flex Framework makes separate configuration of application not only easier, performance is much better guaranteed. The big advantage of a modular setup is that it allows granular control over application settings. This adds the possibility to reset settings for a single application and a new application can be introduced without changing the existing configuration for other applications.
- **Support for Silo's or Server Groups.** Configuration for separate application silos is now straightforward and does not require additional scripting. The configuration of server or even workstation groups is stored in a single location and can be managed without effort.
- **Easy & lightweight deployment.** The Framework is packaged in a MSI format to make deployment easy; without a reboot, "install-mode" or the need to install a service. Only few registry keys and files are added. The MSI is just 500KB.
- **Compression.** The Framework can be set to use the 7z compression algorithm for the storage of personal settings. This *dramatically* improves transfer times through WAN environments or copying of large OPS files. A 15x compression ratio is realistic for registry settings. Files and folders from the profile are compressed up to 7 times!
- **Windows appearance support.** Dennis Damen has released FlexRefresh.exe to re-activate Windows colors, sizes and fonts especially for Flex environments. Users can now customise their desktop to their own needs, even when they are using a mandatory profile!
- **Certificates support.** Thanks to feedback of "Jakobo" and many others in the Login Consultants forum, there is now an option to store and load certificates available in the Flex Framework. Root certificates are not supported, these can be easily configured and distributed through Group Policies however.
- **Support for Passwords.** During the release of FPK 3.0, Windows and IE passwords were not supported. The Framework now introduces the possibility to store Windows related passwords with a "flick of the switch".
- **Error Handling.** Configuration and troubleshooting of the Flex Framework is simplified with new options to generate clear warning messages when an error occurs.

In chapter 8 the configuration of the Flex Framework is explained in detail. If you are experienced with FPK 3.0, go ahead and checkout the Framework immediately. It is very easy to administer and deploy, as you will find out.

2. New Features and Fixes in 5.0

Release 5.0 has the following new features:

- **Support for Windows MUI.** Multilanguage user interface within Windows is now supported.
- **Variable support for StoreRoot & StoreFolder.** In Framework.ini it is now possible to specify variables in the path.
- **Redundant OPS file removal.** When an OPS file is not used anymore, it will automatically be removed.
- **Backup optional.** The backup of the OPS file can now be disabled within the Framework.ini
- **Improved FlexRefresh.** No more annoying FlexRefresh message during logon, no more IconSize bug, and FlexRefresh will now also update cursor settings.
- **More Flexible Configuration.** It is now possible to specify the full path to the other Framework.ini as a third commandline option; this allows the possibility to have multiple configurations.
- **2x Faster Logon.** Although there have been not too many complaints ☺, a bug in the logon process caused to load the all the OPS file twice (oops!). This has now been fixed, reducing the logon time by 50%!
- **A GUI!!!.** The Flex Framework can now be configured with the FlexConfig.exe tool by Magnar Johnsen!

3. Upgrade from FPK 1.x

The main difference to older versions of the FPK is that versions 3.0, 4.0 and 5.0 are based on the Profile Wizard from the Office 2003 resource kit. The core functionality in FKP 5.0 is exactly the same as older FPK releases.

The proflwiz.exe is modified differently than version 1.x, to improve performance considerably, especially during logons on busy servers. Some cases of instability were reported with 1.x on 2003 servers; the new proflwiz.exe is fully tested and reported to be very reliable in Windows 2000 and 2003 environments.

Important!

Critical information when you used a previous version (1.x) of proflwiz.exe and want to upgrade to FPK 4.0:

The new proflwiz.exe works the same as version 1.0. It is perfectly possible to use the .INI files from the previous version. You only need to modify the header section of the .INI files you used with the 1.0 version! In every .INI file you used, change the [Header] section from:

```
[Header]
Version  = 10.0
Product  = Microsoft Office 10.0
CheckSum = 77960994
```

Into:

```
[Header]
Version  = 11.0
Product  = Microsoft Office 11.0
```

Existing .OPS files from users can be loaded without problems. When you want to use the new proflwiz.exe, ensure you update the header in all the .INI files before you begin using the new ProflWiz.exe.

The OPSviewer from the 1.1 kit cannot read .OPS files generated by the new ProflWiz.exe. Please use the OPSview.exe included in the Flex Profile Kit v4.0.

4. Introducing the Profile Wizard

If there was a way to redirect HKEY_CURRENT_USER registry components to the homedrive (just like My Documents and Application Data), personal settings could be saved and used in a mandatory profile environment.

Even though policies allow folder redirection, there has never been an easy way to redirect registry information. Fortunately, there is a very robust and ***easy*** solution to implement this from Microsoft: The (Office) Profile Wizard. The Office Profile Wizard (from here referred to as PW) is freely available as a component from the Office Resource Kit: http://www.microsoft.com/office/ork/2003/tools/BoxA08.htm

The Profile Wizard was originally built to save Office information from the registry and profile folders into an .OPS file. This lets you move around and import your personal Office settings on any computer.

Important!

The ProflWiz.exe in the Flex Profile Kit is modified for running in a Terminal Services environment. The modification prevents the ProflWiz.exe from looking for active Office processes. Ordinarily the Proflwiz.exe will exit when it finds an active Office process. This is a problem in Terminal Server environments since the unmodified ProflWiz.exe sometimes sees Office processes from other users. This modification also improves performance since no running processes are queried during operation.

The proflwiz.exe was modified with a hex editor. Proflwiz.exe uses the psapi.dll to query the system for active office processes. With the hex editor, the reference to psapi.dll was changed to qqqqi.dll. Since qqqqi.dll does not exist the proflwiz.exe continues without scanning for active office processes.

You can identify the newly modified ProflWiz.exe by checking the file version, It is set to 22.2.2222.0 instead of the original 10.0.2609.0. The new ProflWiz.exe is 111kb. The original ProflWiz.exe is 214kb and tagged as version 11.1.1111.0.

The Profile Wizard consists of three components:

1. The executable ***Proflwiz.exe***, which saves and restores the profile information.
2. An .INI file like ***Example.ini*** that's used to define which registry keys or profile files should be saved and restored. *Spaces in the are NOT supported in the filename!*
3. An .OPS file like ***MySettings.OPS*** where personal settings are saved to and restored from, is automatically created.

The executable (***Proflwiz.exe***) checks an .INI-file (***Example.INI***) to identify which registry trees and keys needs to be stored in the configuration file (***MySettings.OPS***). When run without any parameters, Proflwiz.exe acts as a wizard. Using Proflwiz.exe with parameters can automatically save and restore personal settings. The relevant parameters are:

/i <INI file>	Instructs ProflWiz.exe to use a custom INI file instead of the standard OPW11usr.ini, always use this for saving settings!
/s <OPS file>	Save user configuration settings from the current computer to the OPS file. The wizard displays progress indicators and error messages while it is running.
/r <OPS file>	Restores the application settings from the specified OPS file to the computer. The wizard displays progress indicators and error messages while it is running.
/q	Run in quiet mode; do not display progress indicators or error messages. **Recommended!**
/e	Display error messages; suppress progress indicators. Can be combined with **/p**. Cannot be used with **/q**.
/p	Display progress indicators; suppress error messages. Can be combined with **/e**. Cannot be used with **/q**.

5. Flex Basics

You can configure a Flex Profile Solution in seven easy steps:

1. *Configure a single empty mandatory profile for all users.*
2. *Configure policies for profile folder redirection.*
3. *Create INI file(s) from the Template.INI file(s) to designate which registry keys should be saved as personal settings.*
4. *For optimal performance, copy the ProflWiz.exe and new .INI files locally on all TS or Citrix servers.*
5. *Configure a logoff script to save the designated personal settings with proflwiz.exe.*
6. *Edit the logon script to load personal settings with proflwiz.exe.*
7. *Configure the user's accounts terminal server profile to be the newly created mandatory profile.*

These seven simple steps are all what is needed to setup Flex Profiles! Each step is discussed in detail. Once you understand the basics of the Flex Profile concept; checkout the Flex Framework and see how easy it is to configure and deploy.

Important!

It must be noted that proper understanding on how profile & policies are used in general is really recommended before you start deploying Flex Profiles.

5.1 Step 1: Preparation Mandatory Profile

You'll first need to create a mandatory profile for your users. For ease of administration, create a single mandatory profile for all users on all servers.

Important!

Create the new mandatory profile on a fresh server/workstation without any applications and policies!! It is vital that the mandatory profile is "clean" without any specific settings since this profile is used by all users on all servers or workstation.

A very nice step by step guide on mandatory profiles can be found here:
http://support.microsoft.com/default.aspx?scid=kb;en-us;323368

Important!

With the introduction of Windows 2003, the "*Everyone*" group is crippled to enhance security. Do not assign "*Everyone*" in the "Permitted to use" option while copying the template profile you created for the mandatory profile. Preferably, select the "*Authenticated Users*" group when setting the permission for the new mandatory profile.

After you create the mandatory profile, get rid of all the folders and files that are not needed. You can strip all unwanted application and user-specific registry information using regedt32.exe (or regedit.exe in Windows 2003):

1. Make a backup copy of the ntuser.man!
2. Open regedt32 (or regedit.exe when using Windows 2003)
3. Select *HKEY_Users* window (in regedt32 only)
4. Select the root of *HKEY_Users*
5. Choose "Load Hive" from the file menu
6. Select the copy ntuser.man or ntuser.dat
7. Choose a name, e.g. **ManProfile**
8. Select the (e.g. **ManProfile**) tree to expand it
9. Now you can manipulate registry settings from the mandatory profile without having to logon with that profile.
10. Do not forget to unload the ntuser.man when you are finished! Select the tree and choose "Unload Hive" from the file menu.

5.2 Step 2: Folder Redirection Preparation

Since Mandatory profiles are not saved when a user logs off, you'll want to redirect the essential folder content (like the Desktop, My Documents, Favorites and Application Data) to the user's homedrive. These personal files are then stored directly in the user's homedrive regardless what happens to the profile. Since the files are no longer part off the profile, loading times during logon and logoff will decrease significantly opposed to standard roaming profiles.

You can use the limited folder redirection options available in the standard group policies. Group policy folder redirection is restricted to using UNC paths. It can be preferable to use profile folder redirection based on a driveletter. In fact, some applications function better when using a driveletter for folder redirection. Doing this also prevents file open dialogs from being cluttered by lengthy UNC paths.

Application Data	T:\AppData
Temp. Internet Files	%USERPROFILE%\local Settings\Temporary Internet Files
Cookies	%USERPROFILE%\Cookies
Desktop	H:\Desktop
Favorites	H:\Favorites
History	%USERPROFILE%\local settings\history
Local AppData	%USERPROFILE%\Local Settings\Application Data
Local settings	%USERPROFILE%\Local Settings
My Pictures	H:\My Documents\My Pictures
Nethood	%USERPROFILE%\Nethood
Personal / My Docs	H:\My Documents
Printhood	%USERPROFILE%\Printhood
Programs	%USERPROFILE%\Start Menu\Programs
Recent	T:\Recent
SendTo	%USERPROFILE%\SendTo
Start Menu	%USERPROFILE%\Start Menu
Startup	%USERPROFILE%\Start Menu\Programs\Startup
Templates	H:\Templates

Profile Folder Redirection Example

You can use any drive letter for folder redirection, in this example **H:** & **T:** are used. The Profile Folder Redirection example shows where the "Terminal Server Homedrive" of the user account property at the domain level is mapped to **T:**. It's recommended that you hide this Terminal Server Home drive in explorer since it only contains configuration data.

The "normal" homedrive could be mapped during the logon script as **H:**. This is the personal network drive where users store their documents, and should be visible.

To setup extensive profile folder redirection to a mapped network drive like the example above, use the AllinOne (NT4) or True Control (AD) policy templates. These templates are available from http://portal.loginconsultants.nl/forum/index.php?board=16. It is also possible to hardcode folder redirection in the mandatory profile.

Important!

Hardcode the redirection path of **Application Data** in the mandatory profile itself when using folder redirection based on drive letters. (Load and edit the ntuser.man of the profile in regedt32, as explained in the "Mandatory Profile Preparation.") This guarantees folder redirection of Application Data to a drive letter is working properly. Set the path to application data in the key below (example value could be T:\Appdata):
HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\UserShellFolders\App Data

More information on folder redirection can be found in this Microsoft article: http://support.microsoft.com/default.aspx?scid=kb;en-us;242557

Important!

The redirection of **Application Data** to a file server can introduce performance problems. Some applications constantly write or read to files in Application data, this can strain the file-server considerably. As an alternative use FPK to store the Application Data content. This will increase loading times, but it is always faster than using a roaming profile.

Always exclude the locked files **Local Settings\Application Data\Microsoft\Windows\ UsrClass.dat & UsrClass.dat.LOG** when data from Local Settings are saved. These are locked, FPK cannot access them and would stop saving or loading. This can be achieved by adding the following entry in the INI file:

```
[ExcludeFiles]
<UserProfile>\Local Settings\Application Data\Microsoft\Windows\UsrClass.dat
<UserProfile>\Local Settings\Application Data\Microsoft\Windows\UsrClass.dat.LOG
```

5.3 Step 3: Customizing the TEMPLATE.ini

Before configuring the logon and logoff scripts, the TEMPLATE.ini needs to be customized to reflect what trees or keys of the profile are actually roaming.

Important!

Office does *not* need to be installed or configured; it is possible to customize the TEMPLATE.ini to save *any* registry value(s) you want to be personal (roaming) for your users!!

Assuming that the profile folders are redirected to the user's home directory, only registry information needs to redirected by the Profile Wizard. It is possible to include files from the profile folder in the .OPS file, but with complete folder redirection configured this will not be necessary.

Important!

Only use the file (re-)store capacity of the Profile Wizard when folder redirection is not sufficient or possible. A typical folder to save with FPK is "Application Data" when the redirection of Application Data results in performance issues.

By default, an OPW11ADM.INI is included with the Profile Wizard to save and restore global Office 2003 settings (files and registry settings). Never use this original template .INI file! These extensive details are not needed in mandatory profile environments. It's easier and more transparent to administer simplified definition of these settings.

Be sure to review the **Help.INI** included with the Kit, as all sections are explained in detail. An empty TEMPLATE.ini is included for customization. Use the TEMPLATE.ini to define in detail what components of the registry should be roaming.

Template.ini explanation:

```
ALWAYS USE THE TEMPLATE.INI TO CONFIGURE THE PROFILE KIT
REMOVE UNUSED SECTIONS!!


Microsoft Office Save My Settings/Profile Wizard INI file

Edit this file to change which files and registry keys are included into
the OPS file.

Syntax is documented in each section.
All include and exclude strings are case insensitive, except the
entries in the [IncludeFolderTrees] section.
Comments are denoted with # at the beginning of the line.



[Header]
Version  = 11.0
Product  = Microsoft Office 11.0


************************* File/Folder Sections ****************************

[IncludeFolderTrees]
List folder trees to be included into the OPS file.
Syntax is one folder per line; no trailing backslash.
Includes all subfolders in specified tree.
Wildcards are not supported.
These entries are CASE SENSITIVE and must begin with one of
 the following Folder tokens:
  <AppData>, <Desktop>, <Favorites>, <NetHood>, <Personal>,
  <PrintHood>, <ProgramsMenu>, <RecentFiles>, <SendTo>,
  <StartMenu>, <StartupMenu>, <UserProfile>.


[IncludeIndividualFolders]
List individual folders to be included into the OPS file.
Syntax same as [IncludeFolderTrees] but does not include subfolders.
Wildcards are not supported.


[IncludeIndividualFiles]
List individual files to be included into the OPS file.
Syntax is one path\filename per line.
Entries must begin with one of the Folder tokens listed under
[IncludeFolderTrees].
Wildcards are not supported.
Example for including Normal.dot:
  <AppData>\Microsoft\<SubFolder_Templates>\Normal.dot


[ExcludeFiles]
List files to not include into the OPS file.
Syntax is one filename or path\filename per line.
Folder-token (e.g. <AppData>) is optional.
Path relative to folder-token is optional.
Wildcards are supported in the filename.
Wildcards are not supported in the path.
Examples for excluding Normal.dot:
  Normal.dot
  Normal.*
  Norm??.dot
  <AppData>\Microsoft\<SubFolder_Templates>\Normal.dot
```

```
***************************** Registry Sections *****************************

[IncludeRegistryTrees]
List registry trees to include.
All values and subkeys within the specified tree are included.
Syntax is one key per line.
Wildcards are not supported.


[IncludeIndividualRegistryKeys]
List individual registry keys to include.
Syntax is same as [IncludeRegistryTrees] but includes only values
  in the specified key, not subkeys.
Wildcards are not supported.


[IncludeIndividualRegistryValues]
List individual registry values to include.
Same as [IncludeIndividualRegistryKeys] but includes only specific named
  value, not subkeys.
Syntax is key\valuename.
Wildcards are not supported.
Name can be blank to denote the default value (use a trailing backslash).


[ExcludeRegistryTrees]
List registry trees to exclude.
All values and subkeys within the specified tree are excluded.
Syntax is one key per line.
Wildcards are not supported.


[ExcludeIndividualRegistryKeys]
List individual registry keys to exclude.
Syntax is same as [ExcludeRegistryTrees] but excludes only values
  in the specified key, not subkeys.
Wildcards are not supported.


[ExcludeIndividualRegistryValues]
List individual registry values to exclude.
Same as [ExcludeIndividualRegistryKeys] but excludes only
specific named  value, not subkeys.
Syntax is key\valuename.
Wildcards are not supported.
Name can be blank to denote the default value (use a trailing backslash).
```

Important!

In the HELP.INI also `[…….ResetToDefaults]` sections are described; do not use them with FPK, an mandatory profile should be empty so there is no need to reset settings.

In addition, the section [SubstituteEnvironmentVariables] is not recommended to use anymore; earlier editions of FPK did originally suggest to use it. However, this setting has no added value and increases the risk of problems. If you have this section in your INI files, remove it completely:

```
[SubstituteEnvironmentVariables]
%USERPROFILE%
%USERNAME%
```

A customized TEMPLATE.ini could look like this;

```
[Header]
Version  = 11.0
Product  = Microsoft Office 11.0

# ****************** Files Sections ********************

[IncludeFolderTrees]
# When the Application Data folder is not redirected include these folders
<AppData>\Microsoft\Office
<AppData>\Microsoft\Outlook
<AppData>\Microsoft\PowerPoint
<AppData>\Microsoft\Proof
<AppData>\Microsoft\Templates
<AppData>\Microsoft\Word
<AppData>\Adobe


# ****************** Registry Sections ********************

[SubstituteEnvironmentVariables]
%USERPROFILE%
%USERNAME%

[IncludeRegistryTrees]
HKCU\Software\Microsoft\Office\
HKCU\Software\Microsoft\Windows NT\CurrentVersion\Windows Messaging Subsystem
HKCU\Software\Microsoft\Windows Messaging Subsystem\Profiles\Outlook
HKCU\Software\Adobe\Acrobat
HKCU\Printers

[IncludeIndividualRegistryValues]
HKCU\Software\Microsoft\Windows NT\CurrentVersion\Windows\Device
```

Explanation section **[IncludeRegistryTrees]**;

- `HKCU\Software\Microsoft\Office`
  The root office key is included for simplified administration in the first example line. Review the original OPW1ADM.ini for information on how to set detailed and specific Office settings.
- `HKCU\Software\Microsoft\Windows NT\CurrentVersion\Windows Messaging Subsystem`
  `HKCU\Software\Microsoft\Windows Messaging Subsystem\Profiles\Outlook`
  The second and third lines can be included to allow users to configure and save additional mailboxes.
  `HKCU\Software\Adobe\Acrobat`
  Since the Profile Wizard accepts any registry key, it can support any application. The fourth and fifth lines from the example above can be used for Adobe Acrobat.
- `HKCU\Printers`
  Windows information such as printer settings can be also be included as shown in the last example. Add the value below in the [IncludeIndividualRegistryValues] section to save the default printer information:
  `HKCU\Software\Microsoft\Windows NT\CurrentVersion\Windows\Device`

Important!

Like any other INI file, ensure there are **no trailing spaces or backslashes** after the entries! The spaces will be included; the registry key or file you have intended to save will not be found. Although used in the original profile wizard from Microsoft, it is not recommended to add the "# all", "#common" or other flags at the end of each line.

5.4 Step 4, 5 & 6: Configuring the Profile Wizard

To save personal settings, add the following command to the logoff script. When you do this, *the OPS file is automatically created*:

```
C:\PROFLWIZ\proflwiz.exe /s "%temp%\MySettings.ops" /i C:\PROFLWIZ\example.INI /q
copy /Y "%temp%\MySettings.ops" H:\appdata\MySettings.ops
```

Have a directory configured with the Proflwiz.exe and a customized TEMPLATE.ini (which is included in the Kit). Rename the .INI file and use the /I switch to configure your logoff script.

Important!

The profile Wizard is considerably faster when saving or restoring to the local hard drive. Improve performance dramatically by saving the OPS file locally in %temp% or %userprofile% first. Then copy the OPS file to the home drive afterwards.

To configure the logoff script in Windows 2000, run the Group Policy Editor (Use the local policies in an NT4 domain: Start > Run > GPedit.msc) and set a logoff script. On NT4 workstations or NT4 TSE, disable all the standard logoff buttons and create a shortcut that executes a logoff script.

Use the parameter **/q** (quiet mode) to hide the process completely. This will speed up the process considerably and is recommended in almost all situations. When using the /q switch you can reduce the whole process to less then half a second when you save to the OPS file locally first and copy the .ops file to the home drive afterwards.

Next, you'll find an example for loading (restoring) settings with the Profile Wizard for a configuration file stored in the home drive. Before you restore personal settings, first copy the .OPS file locally (in %userprofile%) to dramatically improve performance. You do not need to specify an .INI file while restoring settings, and if you do the .INI file is completely ignored. These commands should be added to the logon script, preferably as one of the first items to be executed:

```
Copy /Y H:\appdata\MySettings.ops "%temp%\MySettings.ops"
C:\PROFLWIZ\proflwiz.exe /r "%temp%\MySettings.ops" /q
```

Using the **/p** parameter, the Profile Wizard runs during logon and logoff, displaying the actions being performed. No interaction is required but the progress indicator ensures the user is aware of the fact that his settings are being saved or configured. You should only use this switch only to check whether the Profile Wizard is actually running, as it is considerably slower than the **/q** switch.

Maximize performance by running all three Profile Wizard components from the local hard drive (ProflWiz.exe, Example.ini & MySettings.ops). The Profile Wizard runs fine from a network share, although it is considerably slower (but still very acceptable in most cases).

You can open the .OPS file with OPS File Viewer (OPSVIEW.exe, also part of the Office Resource kit) to review and troubleshoot saved registry and file settings.

Important!

Proper timing is essential! It is recommended to run the profile wizard in the logon script after the drives are mapped and before other application settings are configured. Also enable the policy "run logon scripts synchronously" to prevent applications or the desktop from starting while the logon script is still running.

In a NT4 Domain you can't refresh policies manually, so be sure to investigate settings from policies that could be overwritten by the Profile Wizard. This can be prevented in Active Directory environments by running the "**Secedit /Refreshpolicy User_Policy /Enforce**" command in Windows 2000 or "**gpupdate /target:user /force**" in Windows 2003 just after the Profile Wizard has finished in the logon script. This way you re-enforce policies, preventing personal settings from conflicting with the mandatory group policies.

Important!

To maximize performance, always use the **/q** switch, place a copy of the ProflWiz.exe and .INI files on the server's hard drive. Load and save a copy of the .OPS file is in the users profile (%USERPROFILE%) or temp (%TEMP%) path on the hard drive (as opposed to directly into to users home drive where the .OPS files are stored).

5.5 Step 7: Configure the profile path

The last step is the most simple of all. In the domain or AD configure all user accounts to use the mandatory profile. Just change the Terminal Server Profile path of each user to the location of the newly created base mandatory profile. When setting up the Flex Profiles for workstations edit the normal roaming profile path in the users properties.

It is possible to configure with Windows 2003 a separate terminal server profile path through GPO, only mandatory profiles can normally not be set this way. A workaround is to configure the Terminal Server profile path in the GPO, copy the mandatory profile in "Documents and Settings\Default User" and rename the ntuser.man back to ntuser.dat. Configure the "Prevent Roaming Profile changes from propagating to the server" to prevent the profile from unloading into the share. Since no profile exists in the share, everytime a user logs on a profile is created from the "Default User" profile. Effectively this makes the profile setup a mandatory one.

Important!

When configuring the logon script through GPO's, timing issues can arise when also the logon script is defined within the policies. There is a real chance that policy excuting the logonscript is run first, before the TS homedrive is mapped. To prevent timing issues configure the logon script as user property in the domain or add the logon script in the "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Appsetup".

6. Configuration Considerations

When configuring a flex profile INI files, Keep It Simple! In most cases saving a single registry tree, key, value or foldertree is sufficient. When you're not certain which keys contain the personal settings: 95% of all applications save personal information in **HKCU\Software\Vendor Name\Application name**.

For example, if Adobe Acrobat settings need to be saved, chances are that when you include **HKCU\Software\Adobe\Adobe Acrobat** you have successfully configured most of Adobe Acrobat's settings to be saved.

```
[IncludeRegistryTrees]
HKCU\Software\Adobe\Adobe Acrobat
```

A generic configuration is not recommended however, **do <u>not</u> just enter**:

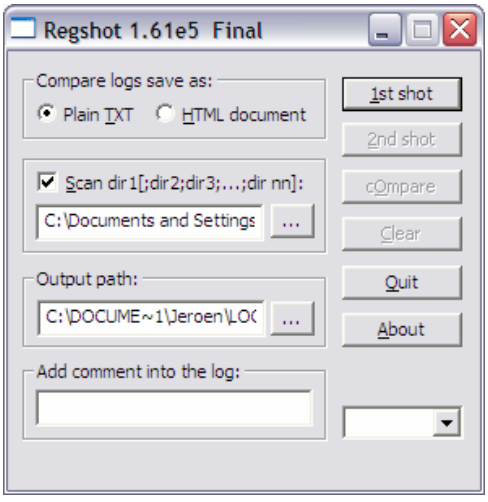
```
[IncludeIndividualFolders]
<UserProfile>
Or
<AppData>

[IncludeRegistryTrees]
HKCU\Software
```

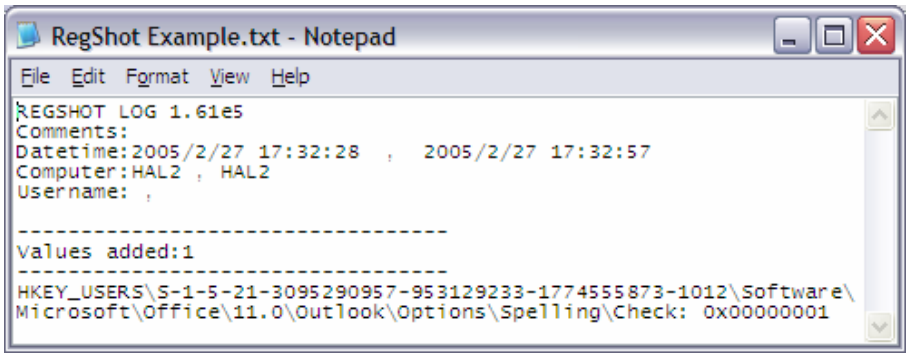
Such a configuration is bound for trouble; some registry keys have restricted access, some files could be locked. Effectively such a configuration seems easy, but would end up functioning like a unmanaged roaming profile.

It is very simple to retrieve the keys or files needed for an application. It is recommended to run **Regshot** on a machine without other users. Regshot is included with the Flex Profile Kit:

1. Start Regshot.exe
2. Configure the "Scan Dir" to the profile folder of the current user to be included in the scan if you also want to monitor files changes.



3. Click "1st Shot"
4. Change the setting you want to "capture"
5. Click "2nd Shot"
6. Select "Compare"
7. A notepad window with all changes will appear



In this example the "Always check spelling before sending" option in Outlook was enabled.

Important!

Never try to configure LOCAL_MACHINE settings or files outside the user's profile with FPK.

6.1 Flex Profile Strategies

Basically there are two strategies for saving settings:

1. The fastest and most simple method of configuration is a single .INI and .OPS file for all settings. This is also very fast during logon and logoff. However, when using application silos (separate servers for different applications) you need to configure a separate .INI and .OPS file for each silo. This prevents the .OPS file from being overwritten by settings from different servers. This method has some disadvantages though. For example, it's only possible to reset settings for users by completely removing the .OPS file, and it's much less flexible to configure and manage.

2. Alternately, you can store settings per application in separate .OPS files. Just use different .INI files for each application. When using separate .OPS files to store personal settings on a application basis, only the .OPS file with "faulty" settings needs to be removed in case of problems with an application. Other personal settings are not affected. Separate OPS files take longer to process during logon and logoff, but working locally on the server hard drive with the profile kit speeds up operation tremendously (about 0.10 - 0.20 seconds per application). A lot of applications do not require personal settings from the registry to be saved, and you obviously do not configure the profile kit for such applications.

Important!

Flex Profiles have a golden rule in Citrix application silo environments; in most cases it's not a problem to load and use the same .OPS file with personal settings on different silos. However, it is vital to always save personal settings to that .OPS file in single silo! If the same .OPS file is saved from servers in different silos, personal settings will be lost and the same problems with roaming profiles are introduced.

It's also perfectly possibly to use the profile kit during **platform migrations**. Most complaints and frustration of users during migrations is caused by loss of personal settings. Use the proflwiz.exe to save personal settings in the old environment in the home drive. When users logon first time in the new environment, just restore these settings with proflwiz.exe. Just check very carefully the settings you are migrating do not conflict with the new environment. The MIG_OUTL.INI is a great example included in the profile kit of how to migrate users' mailbox settings into a new environment.

7. Flex Framework

The installation of the Flex framework is easy and straightforward:

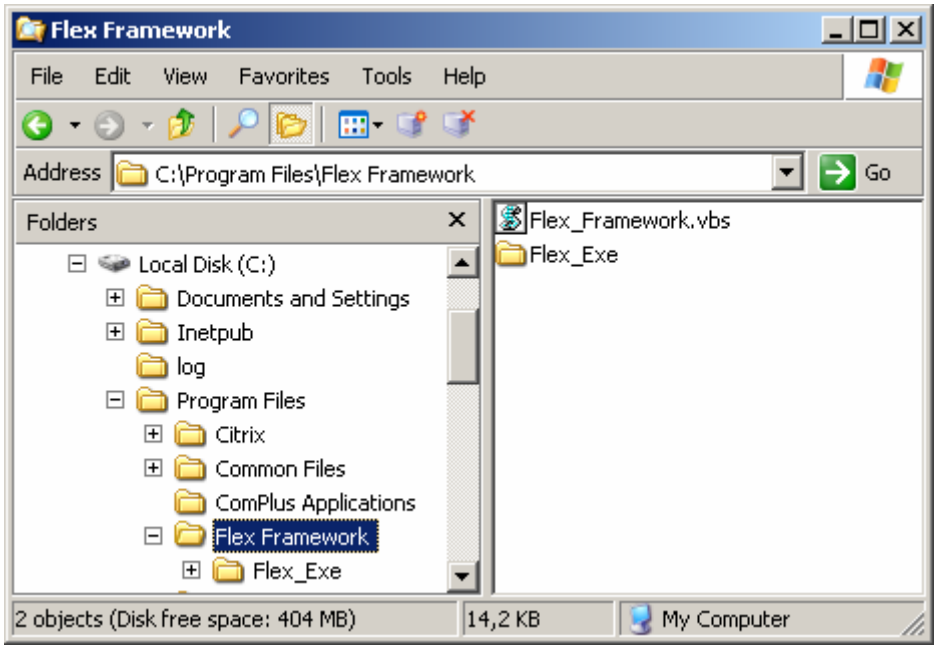
1. Deploy the Framework on all terminal servers or workstations.
2. Unpack "Flex_Config.zip" in a central (redundant) share.
3. Configure the Framework.INI.
4. Configure the "ProfileSettings" folder
5. Add the command for the Flex Framework in the logon and logoff scripts.

7.1 Installing the Framework runtimes

Before you can use the Flex Framework, the tools (the Flex_EXE folder) and main script (Flex_Framework.vbs) need to be installed on every terminal server or workstation. The Framework is packaged in a 500kb MSI installation file. There are no services installed, no reboot is required and for basic usage no additional configuration is needed. Only administrative privileges are required, since a few registry keys are added to Local_Machine.

To install the Framework unattended, just run;

```
msiexec /i "\\Server\Share\FlexFramework.msi" /qn
```



This will install the Framework runtimes automatically in "Program Files\Flex Framework".

To install the Framework in a different location run;

```
msiexec /i "\\Server\Share\FlexFramework.msi" INSTALLDIR="InstallLocation" /qn
```

This allows you to install in any location, installation on a network share is technically possible but not recommended however; this would degrade performance considerably.

7.2 Unpack Flex_Config.zip

In the Flex Profile Kit 4.0 you will also find a zipped Flex_Config.zip file. Unpack this in a central and fault tolerant network share. This can be share of a file cluster, in the NETLOGON share or the SYSVOL folder on a domain controller. From here you can centrally administer the Flex Framework completely.

Example where the **Flex_Config** folder is located in the SYSVOL share

7.3 Configuring Framework.ini

First the Framework.ini file in the root of the "Flex_Config" folder needs to be configured. The Framework.INI configures the Flex Framework, it is pretty much self-explanatory:

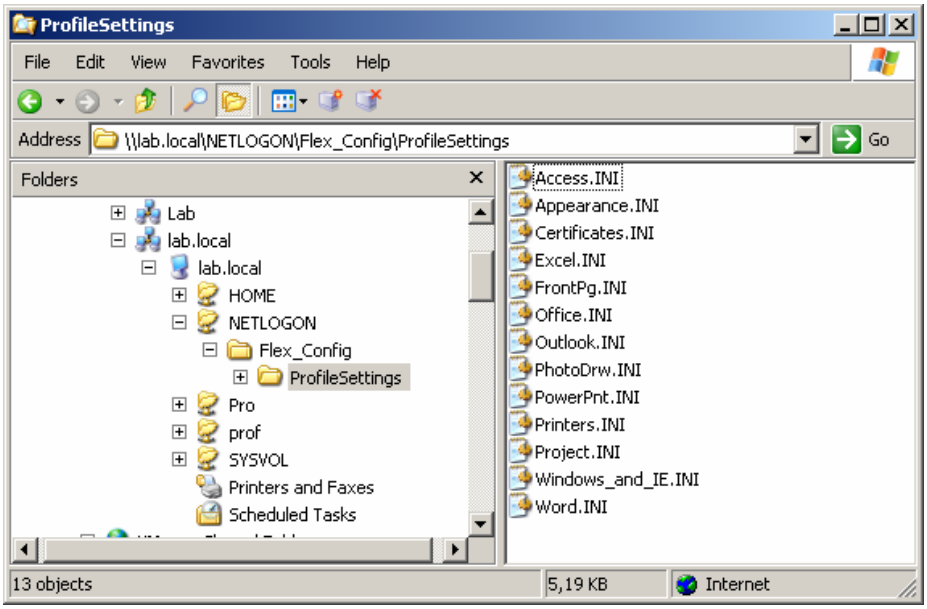
Framework.ini	>>> FLEX_FRAMEWORK CONFIGURATION FILE <<< Like any other INI file, ensure there are no trailing spaces or backslashes at the end of each setting!
[MAIN]	In the section [MAIN] features of the framework can be enabled or disabled with a "1" or "0".
REFRESH_WINDOWS_APPEARANCE=0	To enable the use of Windows appearance settings with Flex Profiles configure REFRESH_WINDOWS_APPEARANCE=1. This will launch Dennis Damen's FlexRefresh and personal appearance settings are activated.
REFRESH_KEYBOARD=0 **REFRESH_MOUSE=0**	Like Windows appearance settings the keyboard and mouse settings need to be activated by a refresh. With these options the Keyboard & Mouse settings become user specific instead of client specific.
ENABLE_PASSWORDS=0	With the ENABLE_PASSWORDS setting the Flex Framework will create a key in *HKCU\Software\Microsoft\Protected Storage System Provider\[SID of the User]*. Since it already exists Windows does not need to create this key with only permissions for System account.
ENABLE_CERTIFICATES=0	To enable the use Certificates set ENABLE_CERTIFICATES=1. In addition, it is **essential** to configure permissions to "*HKLM\Software\Microsoft\Windows NT\Currentversion\Profilelist*" for users. Normally, users only have read access here. Enable the special permission "set value" for "Authenticated Users" on this key. This allows the Framework to spoof the profile state to a "Roaming Profile" during logon. Only Roaming or Local Profiles are allowed to store a certificate. During Logoff the Framework configures the profile back to a "Mandatory Profile" to prevent Windows from trying to save the profile. Root certificates are not supported. This is a typical limitation amongst profile alternatives, and not only of FPK. It is possible to distribute a root certificate through group policies.
COMPRESSION=0	To enable compression of the OPS files configure COMPRESSION=1. The ZevenZip "7z" commandline tool is used (freeware & open source). 7z has by far the best compression algorithm, 10x - 15x compression factor for registry settings, 5x - 7x compression factor for files. 7z is very fast and less demanding on CPU the other compression methods or tools. Go to http://sourceforge.net/projects/sevenzip/ for more information. This setting is only recommended when in addition to registry settings also files and folders from the profile are saved. Use the compression setting where it >clearly< improves loading/saving times or when the OPS files are stored over a WAN connections. In typical 100 MB LAN environments compression is only interesting when the total of OPS files per user-session exceeds 5-10MB. Since the 7z Commandline tool is used, the option to "Disable the command script processing also?" must NOT be set "Yes" in [User Config > Administrative Templates > System > Prevent Access to the Command Prompt]. It is OK to configure Prevent Access to the Command Prompt console.
COMPRESSION_PRIORITY=NORMAL	When (de-)compressing large files 7z can put a full load on the CPU. Optionally configure the priority to BELOWNORMAL of the

	process so other users are not affected. The COMPRESSION_PRIORITY can be set to LOW, BELOWNORMAL, NORMAL, ABOVENORMAL or HIGH. Setting the priority higher than NORMAL is not recommended!
[LOCATIONS]	The [LOCATIONS] section configures the location where the OPS files are stored.
STOREROOT=1	STOREROOT configures the root of the path where the (compressed) ops files are stored. Its usage is very flexible and straightforward. When the logon script is defined through GPO, chances are high that the Terminal Server Homedrive is not mapped when the logon script runs. To prevent these timing issues configure the logon script as a user property in the domain or add the logon script in the *"HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Appsetup".* STOREROOT=1 -> AUTO, will set the store root path automatically to the mapped homedrive by resolving %HOMEDRIVE%%HOMEPATH% from the environment. When TS home drives are configured via W2003 GPO's, this setting is also recommendable. STOREROOT=2 -> Query the user's homedrive from the domain. Configure this option if there is a reason not to use the TS Homedrive as a save location. STOREROOT=3 -> Use the user's Terminal Server Homedrive directly from AD (Only supported on W2003 server in an AD Enviroment!) STOREROOT=[FullPath] -> It is also possible to define a drive letter (STOREROOT=T:) or use a UNC path (STOREROOT=\\Client\C$). Variables like %username% are now supported, configure the path without a trailing backslash "\".
STOREFOLDER=SETTINGS\FLEX	The value of STOREFOLDER will be appended to the STOREROOT value. This setting defines in what folder the (compressed) OPS files are stored. At least one folder needs to be configured, it is possible to specify a tree of folders (folder1\folder2\etc..).
DISABLE_BACKUP=1	By default a backup copy of the OPS file is created in the store location before it is overwritten. To disable this backup configure DISABLE_BACKUP=1.
[ERRORMESSAGES]	When the Framework needs troubleshooting, the options in the [ERRORMESSAGES] section can assist.
DISPLAY_FRAMEWORK_ERROR=1	To create a messagebox when an error has occurred within the configuration of the Flex Framework configure DISPLAY_FRAMEWORK_ERROR. This option will not show error messages by the Proflwiz.exe. DISPLAY_FRAMEWORK_ERROR=0 -> Disable Flex Framework Error Messages. DISPLAY_FRAMEWORK_ERROR=1 -> Show only error messages during the Logon process. DISPLAY_FRAMEWORK_ERROR=2 -> Show only error messages during Logon and Logoff, this setting should only be used for troubleshooting. When an error occurs during logoff, the Framework will not stop until the users as selected "Ok". Since users mostly do not observe of what happens in the logoff process, this option increases the risk of orphaned sessions.
DISPLAY_PROFLWIZ_ERROR=0	To enable error messages from the ProflWiz.exe, enable the

	DISPLAY_PROFLWIZ_ERROR setting. DISPLAY_PROFLWIZ_ERROR=0 -> Run in quiet mode; do not display progress indicators or error messages. This setting is highly recommended for normal operation! DISPLAY_PROFLWIZ_ERROR=1 -> Display error messages; suppress progress indicators. DISPLAY_PROFLWIZ_ERROR=2 -> Display error messages and progress indicators. Use this setting only for troublshooting, the Profile Wizard will run much slower. DISPLAY_PROFLWIZ_ERROR=3 -> Display progress indicators; suppress error messages. The Profile Wizard will run much slower.

7.4 Configuring Profile Settings

In the Flex_Config folder you will find a "ProfileSettings" folder next to the Framework.ini. This is where the actual settings for the Profile wizard are configured.



Example where the **Flex_Config** folder is located in the NETLOGON share.

In the Framework you will find separate example INI files per application. For every INI file an OPS file will be automatically created with the same name. If you create a SAP.ini, the Flex Framework will automatically save it in the storefolder (the storefolder is set up in the Framework.ini) as SAP.ops.

It is perfectly possible to merge all INI files into one (example: All.INI), in that case all settings will be stored in a single OPS file (example: All.OPS). This is completely your decision, separate INI files are more flexible to use and manage however. Since settings are stored in separate OPS files, resetting the configuration of a specific application for a user does not affect other applications.

When a new application is introduced, just add an INI file with the settings for that application. When an application is not used any more, remove the corresponding INI file and these settings will not be loaded anymore. **Spaces in the name of the INI file are NOT supported!**

Important!

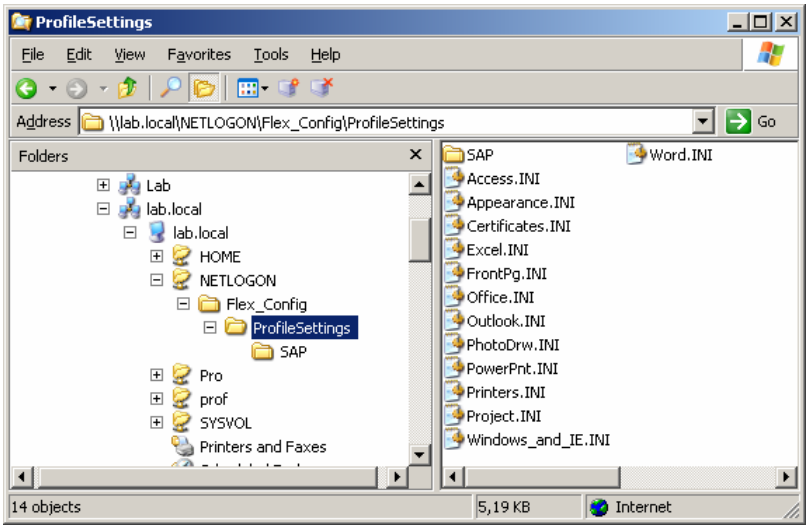
It is possible to configure NTFS permission on the INI files. Configure per INI file SYSTEM and ADMINISTRATORS with "Full-Control" and grant the corresponding application group "Read" permissions. This way, when a user is not authorised for an application, the Framework will not attempt to save or restore settings for that program anymore.

Even when hundreds of separate INI files are configured, through NTFS permissions only settings for authorised applications are configured. This is a very flexible and scalable solution. In practise, about 30-40% of all applications require personal settings and users are normally not authorised to use them all. It is highly unlikely users end with too many separate ops files that would slowdown logon or logoff.

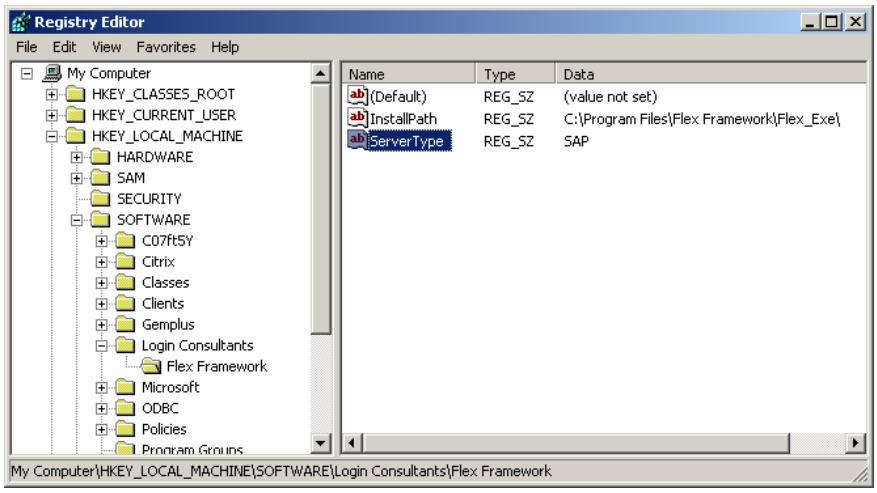
To summarize: You got a new application that requires personal settings? Add an INI file for that application in the "ProfileSettings" folder, and configure the permission to ensure the Framework will only manage settings of authorised applications. That is all there is to it…

7.5 Configuring ServerType

Having a separate configuration for each application silo's or workstation type, is made easy with the Flex Framework. To accomplish this, add a folder in "Flex_Config\ProfileSettings" with the name of that server group or workstation type. For instance, if you introduce a dedicated silo for the SAP application, just create a "SAP" folder in "Flex_Config\ProfileSettings" and add a SAP.ini file (and possibly other INI files for that server group).



Once you have created a separate folder in "Flex_Config\ProfileSettings" you must configure the Servertype value on the corresponding servers or workstations. This can be done by configuring the registry "Servertype" value (empty by default) on each server or workstation in **"HKLM\Software\Login Consultants\Flex Framework"**. In the SAP silo example mentioned a SAP folder was created; so the "Servetype" value data should be "SAP". When the "Servertype" value is configured, OPS files will be stored in: StoreRoot\StoreFolder\Servertype.



Setting the corresponding "Servertype" value in the registry is probably the easiest method, but the Framework also allows the ServerType to be set as a System variable: SERVERTYPE. When both a SERVERTYPE system variable and the registry value "Servertype" are configured, the registry value of "ServerType" will be used by the Framework.

To summarise; for every server group or workstation type you need a different configuration; create a folder in "Flex_Config\ProfileSettings" and configure the "Servertype" value with the name of that folder on the corresponding computers. The "Servertype" value can be configured in the registry or as a System variable.

7.6 Logon/Logoff script configuration

Configuring the Flex Framework in the logon and logoff script does not require rocket science. The syntax to activate the Framework is:

```
CSCRIPT /NOLOGO [PATH_TO_FRAMEWORK.VBS] [LOGON / LOGOFF] [PATH_TO_FLEX_CONFIG]
```

In practise the command above could be configured in the logon script as this example;

```
CSCRIPT /NOLOGO "%PROGRAMFILES%\Flex Framework\Flex_Framework.vbs" LOGON
    "\\lab.local\SYSVOL\LAB.local\Flex_Config"
```

The logoff script should be configured the same, expect "LOGOFF" needs to be configured:

```
CSCRIPT /NOLOGO "%PROGRAMFILES%\Flex Framework\Flex_Framework.vbs" LOGOFF
    "\\lab.local\SYSVOL\LAB.local\Flex_Config"
```

Important!

Proper timing is essential when setting up the FrameWork in the logonscript. It is recommended to run the Framework in the logon script *after* the home drives are mapped and before other application settings are configured. Also enable the policy "run logon scripts synchronously" to prevent applications or the desktop from starting while the logon script is still running.

When the logon script is defined through GPO, chances are high that the Terminal Server Homedrive is not mapped yet when the logon script runs. To prevent these timing issues configure the logon script as a user property in the domain or add the logon script in the *"HKLM\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\Winlogon\Appsetup".*

Optional

With Flex Framework 2.0 it is possible to specify a different framework.ini file on the commandline, this way it is possible to specify an separate configuration for specific servers or workstations:

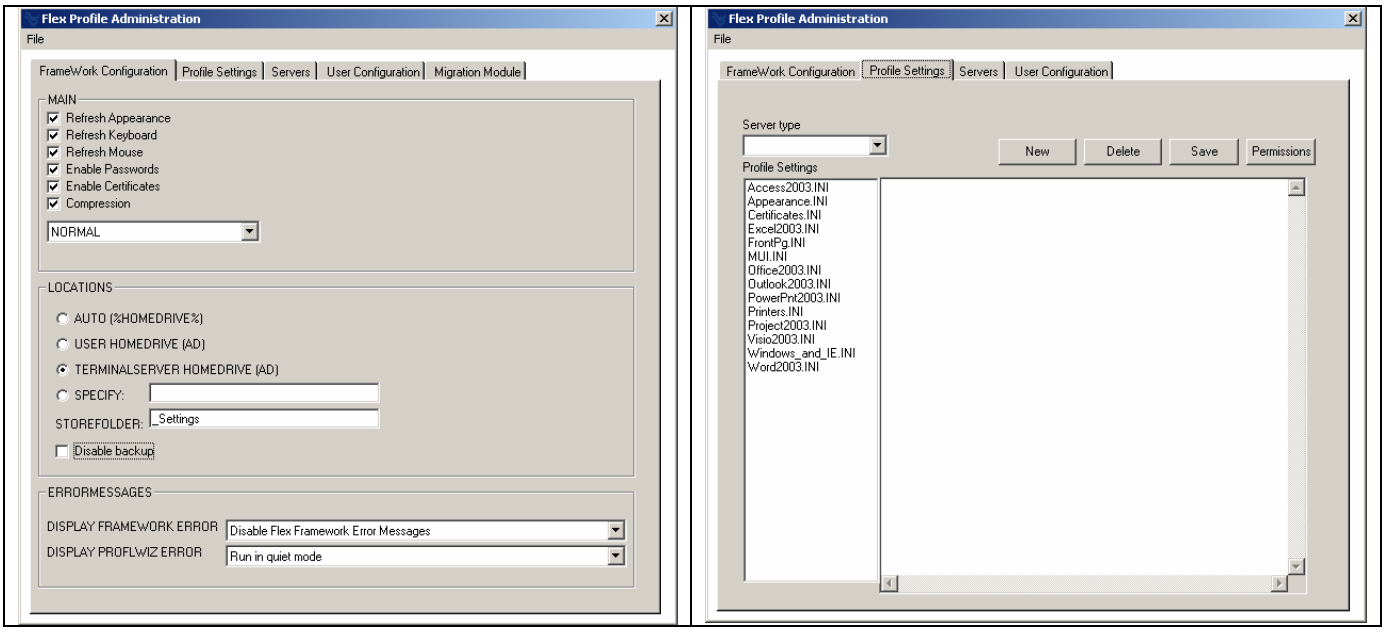
```
CSCRIPT /NOLOGO [PATH_TO_FRAMEWORK.VBS] [LOGON / LOGOFF] [PATH_TO_FLEX_CONFIG]
[PATH_TO_FRAMEWORK.INI]
```

8. FlexConfig.exe

Magnar Johnsen has created the Flex Framework configurator: **FlexConfig.exe**. This tool makes configuration of the framework even easier:

- Full configuration of the Framework.ini settings (Framework Configuration Tab)
- Configuration of the Flex Profile configuration INI files, including security (Profile Settings Tab)
- Configuration of additional servertypes (Servers Tab)
- Configuration of user settings in Active Directory (User Configuration)

It is very easy to use, when you start FlexConfig.exe, browse to your Framework.ini. After this you can easily configure the Framework and additional settings.



After you have configured the Framework settings, select File > Save to store the settings in the INI file.

Important!

You need TABCTL32.OCX to run the tool. Run the Install_TABCTL32.CMD to copy and register the file automatically, but only do this when you get the error message about a missing TABCTL32.OCX!

9. Additional Information

Please remember usage of the Flex Profile Kit and Flex Framework is completely at-your-own risk. You are allowed to change the content for your own purposes. It is explicitly not allowed to sell the Kit in any form or shape, it is a freeware solution/concept.

Webmasters: Feel free to add the Kit in your download section. You are allowed to add a referral plain-text file with info about your site to root of the FKPv4.zip file. It is not allowed to change or alter the original contents!

9.1 Troubleshooting Checklist

When troubleshooting Flex Profiles please use this checklist:

1. Ensure no trailing spaces or backslashes are typed in the INI files
2. Verify there are no spaces in the name of the INI files
3. Make sure the [Header] is in the INI file is configured to:
   ```
   [Header]
   Version  = 11.0
   Product  = Microsoft Office 11.0
   ```
4. Verify you have configured "Authenticed Users" to have Full-Control in the mandatory profile
5. Check settings stored in the OPS file with OPSview.exe
6. Configure DISPLAY_FRAMEWORK_ERROR=1 to check for Framework error messages in the Framework.ini
7. Configure DISPLAY_PROFLWIZ_ERROR=1 to check for Profile Wizard error messagesin the Framework.ini
8. Verify the policy "run logon scripts synchronously" is enabled and working
9. Verify the Flex runs in the logon script after the homedrive is available and before applications are configured
10. Configure the logon script in the "winlogon\appsetup" registry key or in the user's domain logon script property instead through GPO's .
11. Use Regshot to confirm you have configured the proper keys and folders in the INI file
12. Run the FKP scripts while you are logged on a desktop and see what happens
13. Remove the section [SubstituteEnvironmentVariables] completely from the INI file
14. Check if the Profile wizard is trying to save or overwrite locked files in the profile with Sysinternals Filemon
15. Verify the Appdata redirection is straining the file-server when experiencing performance problems (this can be a problem of folder redirection in general)
16. Read this document carefully ☺
17. Do a search on Login Consultants forum (set "*Max Age since last post*" on 999 days); http://portal.loginconsultants.nl/forum/index.php?board=16;action=search
18. If you still have troubles, *please don't mail me personally,* but ask your question or put a suggestion on the forum.
19. When you post on the forum post your INI files and the logon/logoff scripts, and please give as much information you can share.

9.2 Updates & addons

Always refer to our forum http://portal.loginconsultants.nl/forum/index.php?board=16 for the latest updates, bugfixes and addons. Here you will find other tools like the "DADE Logonthrottler" (by Dennis Damen and Daniel Nikolic), "Reconnact!" (by Dennis Damen) and the True Control Policy templates. Feel free to share ideas, solutions or suggestions.

9.3 Support services

On several occasions, Login Consultants has received inquiries from around the world for official support on Flex Profiles. With FKP 3.0 this was not possible, there is no license agreement and the Flex Profile Kit 3.0 was just a concept.

With FPK 4.0 Login Consultants can offer limited support services on the Flex Framework itself. It is also possible to hire our international consultancy services for a one day on-site "Flex

Profile and Framework" workshop. Please contact "Login Consultants International" for more information: info@loginconsultants.com.

9.4 Briforum

"When talking about the "experience" of BriForum, we have to be clear about one thing. All of us of attended countless vendor conferences, hungry for good, practical, and relevant technical content. We've often been disappointed at commercial conferences due to the lack of in-depth technical content (so called "advanced" sessions that are really very basic), off-topic content, or content that is total vendor propoganda.

BriForum traces its roots to conversations between industry experts who longed for a truly technical and indpendent Citrix and Terminal Server conference. We sketched the idea on the back of a napkin years ago, and then we went out and built the conference that we wanted. All by ourselves, without and vendor control.

BriForum 2006 will have an unreal amount of solid technical content. There are almost 30 presenters who will lead over 60 technical breakout sessions. The only problem is with so many great sessions, which ones do you want to attend? Don't worry. We're video-recording all of the sessions, and the videos will be made available on the BriForum website for attendees just after they're presented.

BriForum is a technical conference, but there is plenty of content for everyone from true heavy-hitters to less experienced admins who are trying to wrap their heads around what's going on in this industry."

For more information on Briforum 2006, please visit http://www.briforum.com.

9.5 Links

Please checkout the following and very recommended sites:

- http://www.brianmadden.com with loads of interesting rumors, opinions, and quality technical background information.
- www.dabcc.com from Douglas Brown, home of the famous "Methodology in a Box". The site for quality in-depth information on Citrix (related) technologies and strategies for building server based solutions.
- www.pubforum.net by Dr. Conti, he organises the BRILJANT pubforum, a highly enjoyable informal event for SBC geeks. Here you can also download all Terminal Server related hotfixes from Microsoft!
- www.thincomputing.net is run by Michel Roth, a great SBC related news site that is regularly updated.
- www.loginconsultants.com is the company I work for. Check out the download section on our Forum! It contains many useful tools we use on a daily basis, like the Flex Profile Kit, ReconnAct! and the latest True Control or AllinOne Templates.

9.6 Credits

Without the feedback of posters in our forum, Flex Framework would not have existed. Special thanks go to: Erwin Vollering, Jakobo, Brianoserous, Citrix44u, Simon Bond, Stefan Baerthel, NeilH and the others.

And then there is my colleague Dennis Damen. He has invested many late night spare time hours in FlexRefresh.exe on my request. Cheers Mate! It is truly appreciated!

Many thanks go to Magnar Johnsen, he has created the brilliant FlexConfig.exe!