**DATA PROCESSING PROCEDURES 01:**

**CAPI TABLET SETUP**

**Table of Contents**

[Introduction 3](#_Toc52626987)

[1. Using the Inno DHS Setup and Update Templates 3](#_Toc52626988)

[1.1 Initial Steps 3](#_Toc52626989)

[1.2 Modifying the Setup Script Template 4](#_Toc52626990)

[1.2.1 Setup 4](#_Toc52626991)

[1.2.2 Files 6](#_Toc52626992)

[1.2.3 InstallDelete 6](#_Toc52626993)

[1.2.4 Dirs 6](#_Toc52626994)

[1.2.5 Run 7](#_Toc52626995)

[1.2.6 Icons 7](#_Toc52626996)

[2. Creating an Executable Installation File 8](#_Toc52626997)

[Appendix A: Tips for Creating Silent Install Commands 9](#_Toc52626998)

[Appendix B: Template Setup Script 10](#_Toc52626999)

# Introduction

This document provides procedures and instructions to assist with setting up Windows tablets in preparation for training and fieldwork.

For DHS and MIS CAPI surveys, in country IT staff and DHS DP staff must work together to set up Windows tablets, including installation of CSPro, the CAPI system and other utilities such as Adobe PDF viewer. This process typically involves setting up large numbers of tablets, which can be any number from 20 to over 200, in a relatively short period of time. At the same time, any missing component of the system, whether it be missing files, folders or applications, can cause difficulties for interviewers and supervisors, causing disruption and lost time during training and fieldwork.

For these reasons, it is essential for the DHS Program to make use of automated tools and standardized procedures for setting up tablets prior to and during DHS training and fieldwork. Using automation for repetitive tasks such as software installation both reduces the time taken for setup and minimizes the likelihood of key components of the system being unavailable to users following a faulty setup.

Currently, the preferred tool for use by the DHS Data Processing group in automating setup for Windows tablets is [Inno Setup](https://jrsoftware.org/isinfo.php). This utility is a free application which allows easy development of executable installation files using scripts. It replaces the use of WinRAR executable files in conjunction with Windows batch files, which was the method used in most DHS7 surveys. The main advantage of Inno Setup over WinRAR+batch is that it allows both file extraction, installation of software and other changes to the system such as password setting to be controlled and executed from a single reusable script.

The following steps should be automated using Inno setup wherever possible:

1. Installation of CSPro
2. Installation of Windows anti-virus together with an up to date virus database.
3. Installation of any other needed software such as GIS tools, Reader and Teamviewer
4. Installation of the DHS CAPI system including all “helper” files such as interviewer database
5. Creation of a user password (this will be the same for all users)

In addition to the above steps, consideration should be given to the following additional steps which are important to

# 1. Using the Inno DHS Setup and Update Templates

## Initial Steps

To use Inno Setup, the latest version should be downloaded and installed. Once installed, open the file .\setup.iss. This is a script file which acts as input to Inno Setup to create a self-executing installation file. A copy of this script is contained in Appendix A to this document.

Inno Setup Scripts are arranged into sections. Each section controls a different aspect of the installation. A section is defined by specifying the name of the section enclosed in square brackets []. Sections can contain any number of entries.

There are two different types of sections: those such as [Setup] whose entries contain directive names and values (in the form Directive=Value), and those such as [Files] whose entries are divided into [parameters](mk:@MSITStore:C:\Program%20Files%20(x86)\Inno%20Setup%206\isetup.chm::/topic_params.htm).

Here is an example:

[Setup]

AppName=HH Listing CAPI system

[Files]

Source: "Entry\\*.\*";DestDir: "{app}\Entry"

Multiple sections with the same name can be specified.

Comments are denoted by placing a semicolon at the beginning of a line. For example:

set location where installation .exe file is output to OutputDir=C:\CC81\Utility\inno

The template setup.iss should be modified to fit the requirements of the specific installation for the country survey and installation context, such as pretest, main training or main fieldwork. At a minimum, occurrences of the generic “CC81” should be replaced with the name of the project folder throughout the setup.iss file.

Inno Setup allows for considerable flexibility and power in setting up installation scripts. More information about syntax and features is available in the [Inno Setup help documentation](https://jrsoftware.org/ishelp/).

## Modifying the Setup Script Template

### Setup

The **setup** section is used to define global settings used by the installer. The settings used for the setup.iss template are shown in the table below, along with default values and comments indicating where changes should be made.

[Setup]

AppName=HH Listing CAPI system

AppVersion=1.0

DisableDirPage=yes

DisableProgramGroupPage=yes

DefaultDirName=C:\CC81

DefaultGroupName=HH Listing CAPI System

Compression=lzma2

SolidCompression=yes

OutputDir=C:\CC81\Utility\inno

SourceDir="C:\CC81"

OutputBaseFilename=CC81\_Setup

Uninstallable=no

WizardImageFile=C:\CC81\Utility\inno\dhs.bmp

WizardSmallImageFile=C:\CC81\Utility\inno\dhs.bmp

|  |  |  |
| --- | --- | --- |
| **Directive** | **Default Value** | **Comment** |
| AppName | DHS CAPI system | Name of system shown in installation window and in Windows start menu. |
| AppVersion | 1.0 | Can be changed if desired |
| DisableDirPage | Yes | Should be left as ‘yes’ to prevent the installation user from changing the default location of installed programs |
| DisableProgramGroupPage | Yes | Leave as ‘yes’ to prevent installation user from changing the program group in Windows |
| DefaultDirName | C:\CC81 | Folder where installation files will be copied. Normally this will be the project root. Change to the country specific project directory. Note that this sets the value of the {app} internal variable used in the sections following Setup |
| DefaultGroupName | DHS CAPI System | Program group in Windows. Can be left as default. |
| Compression | lzma2 | Leave as default. |
| SolidCompression | Yes | Leave as default. |
| OutputDir | C:\CC81\inno | Location where the installation executable file will be output to. Note: make sure the executable file is not located in one of the folders that contains files that are part of the installation, otherwise |
| SourceDir | "C:\CC81" | Sets default path for source folders in the [Files] section |
| OutputBaseFilename | CC81\_Setup | Name of installation executable |
| Uninstallable | No | Leave as ‘no’ to disable default creation of a file for uninstalling the system (uninstall.exe). Normally having such a file available to users is not desirable. |
| WizardImageFile | C:\CC81\Utility\inno\dhs.bmp | Location of DHS logo image file for display in installation window. |
| WizardSmallImageFile | C:\CC81\Utility\inno\dhs.bmp | Location of DHS logo image file for display in installation window |

### Files

The [Files] section is used to specify which files and folders need to be installed on the target machine. The general syntax is as follows:

**Source: “<file path>”; DestDir “<target file path>”**

An example is shown below. Note that {app} is an internal variable created by the Inno system that contains the value of the DefaultDirName assigned in the Setup section.

By default, any subdirectories under the folders specified as source folders will not be included in the installation. To ensure that subdirectories are copied it is necessary to add the recursesubdirs flag after the Destdir keyword as shown below for the PII folder.

[Files]

Source: "Entry\\*.\*"; DestDir: "{app}\Entry"

Source: "Library\\*.\*"; DestDir: "{app}\Library"

Source: "Superv\\*.\*"; DestDir: "{app}\Superv"

Source: "PII\\*.\*"; DestDir: "{app}\PII";Flags:recursesubdirs

### InstallDelete

This optional section defines any additional files or directories you want the installer to delete. This section is useful as a way to ensure that old files or applications from previous setups are automatically deleted from the target machine during installation. This is most useful when installing an updated system for a new phase of survey activity, for example if installing a main training version of the system on tablets that may have been used for the pretest. The general syntax is as follows:

[InstallDelete]

Name: <file or folder to be deleted>; Type: [filesandordirs | files | dirifempty]

An example is shown below. This syntax specifies that the installation will delete all existing files and folders, including subfolders, in the PII folder. Note that setting the Type flag to filesandordirs will delete all subfolders recursively.

[InstallDelete]

Name: {app}\PII; Type: filesandordirs

### Dirs

This section is used to create empty folders. This is useful to ensure that folders required by the system exist. The general syntax is as follows:

[Dirs]

Name: "<file path>"

An example is shown below:

[Dirs]

;create empty folders

Name: "{app}\PII\data\ "

Name: "{app}\PII\work"

Name: "{app}\PII\temp"

### Run

The [Run] section specifies programs to be executed after the system has been successfully installed, but before the Setup program displays the final dialog.

Programs are executed in the order they appear in the script. By default, when processing a [Run] entry, Setup will wait until the program has terminated before proceeding to the next one, unless the nowait, shellexec, or waituntilidle flags are used.

Basic syntax is as follows:

[Run]

Filename: "<path and file name of executable"; Parameters:"<command line parameters>"; StatusMsg:"<text for optional status message>";

All command line parameters needed for a given executable should be placed in the Parameters section between double quotes.

An example is shown below for installation of Adobe Reader, Windows anti-virus and CSPro 6.3.

[Run]

Filename: "{app}\utility\others\AcroRdrDC1500820082\_en\_US.exe"; Parameters:"/sPB /rs"; StatusMsg:"Installing Adobe Reader";

Filename: "{app}\utility\others\mpam-fe64.exe"; Parameters:"-q";StatusMsg:"Updating Windows Defender";

Filename: "{app}\utility\others\cspro63.exe";Parameters:"/s /sms /f1{app}\utility\others\cspro63.iss";StatusMsg:"Installing CSPro 6.3";

Note that the Run section can be used to execute any command that could be run from the Windows command line by invoking cmd.exe. This capability can be used to run Windows console commands including commands to create or modify user passwords. An example is shown below. Note that in this example, there are double quotes around the %username% environment variable to ensure that it is surrounded by quotes when the executable is called.

Filename: "{sys}\cmd.exe"; Parameters:"/c net user ""%username%"" cc81password";StatusMsg:"setting windows password";

### Icons

This section is used to create shortcut icons in the Start Menu and/or other locations, such as the desktop. For DHS and MIS CAPI surveys, typically shortcuts to the the DCMENU and SUPMENU pff files are created on the user desktop.

Basic syntax is as follows:

Name "<location of shortcut>"; Filename:"<shortcut target file>";

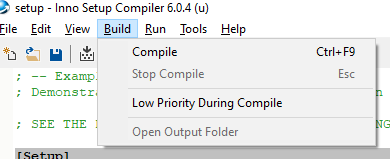
An example is shown below. Note that {commondesktop} is a built-in constant in Inno Setup to denote the common desktop in Windows. More information about constants can be found in the Inno Setup documentation.

[Icons]

Name: "{commondesktop}\Interviewer"; Filename:"c:\CC81\entry\dcmenu.pff";

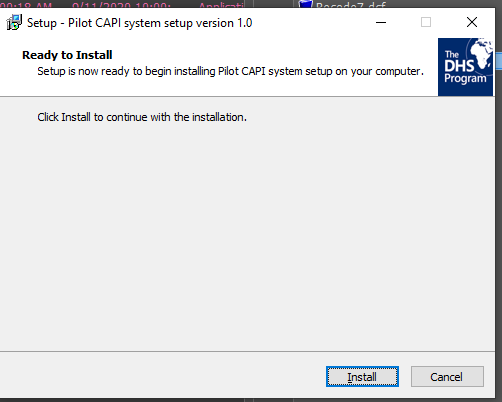
# 2. Creating an Executable Installation File

Once the script has been modified to suit the needs of the specific installation and saved, the next step is to compile it into an executable file. This is done by using the build menu and selecting the ‘Compile’ option. The keyboard shortcut CTRL-F9 can also be used to run this option.



The compiler will run the script and generate an executable installation file, with the file name and output path specified in the OutputBaseFilename and OutputDir parameters specified in the script. This file should then be copied to a test machine and run to ensure that the installation operates correctly and installs all needed files.

To run the installation, just double click on the executable. The system will confirm you wish to make changes to the system, then the following installation window will appear. Note that the title text is user defined based on parameters in the [Setup] section of the script. Once the “Install” button is clicked the installation will begin.



# Appendix A: Tips for Creating Silent Install Commands

The following are instructions and tips for setting up installation programs to run silently, that is, without user intervention, when called in the [Run] section of the Inno setup script.

1. Silent install for Windows Security essentials:

* Command line for MSE = mseinstall.exe /s /q /o /runwgacheck
* Download latest definitions from <http://www.microsoft.com/security/portal/Definitions/HowToMSE.aspx>
* Then  mpam-fe.exe /q to install definitions

1. Silent install for CSPro

Installers created by InstallShield recognize the /r, /s, /sms, /f1, and /f2 switches.

To perform a silent installation, you need an InstallShield "answer file", customarily named setup.iss. Some applications ship with such a file, but if yours does not, you can use the graphical installer itself to create one.

Here is how it works. Run the installer with the /r ("record") switch. Proceed through the dialogs and complete the installation. This will create a setup.iss file and place it in the C:\WINDOWS directory. This file will include all your responses to the InstallShield dialogs, allowing you to perform unattended installations as if you were giving the same answers again. Simply copy setup.iss to the same directory as the installer executable.

Once you have a setup.iss file, run the installer with the /s ("silent") option. This will perform an unattended installation.

Unfortunately, the installer will fork a separate process and exit, meaning it will return immediately even if you run it under start /wait. This makes it useless for scripting purposes. Luckily, there is another switch, /sms, which will cause the installer to pause until the installation completes.

Hence, for an InstallShield application, you want to provide both the /s and the /sms switches.

The /f1 filename switch allows you to specify a fully-qualified alternate name for the setup.iss file. Note that there must be no space between the /f1switch and the file name. This switch works both with /r to create the file and with /s to read it.

The /f2 filename switch specifies a log file. Once again, there must be no space between the switch and the file name.

WARNING: Be careful what characters you use in these file names, because InstallShield silently strips certain non-alphanumerics (like hyphens).

# Appendix B: Template Setup Script

; -- Example1.iss --

; Demonstrates copying 3 files and creating an icon.

; SEE THE DOCUMENTATION FOR DETAILS ON CREATING .ISS SCRIPT FILES!

[Setup]

AppName=HH Listing CAPI system

AppVersion=1.0

;disable prompt to allow user to choose installation folder

DisableDirPage=yes

DisableProgramGroupPage=yes

;set folder for installation files

;note this sets the value of the {app} internal variable used in the sections below

DefaultDirName=C:\CC81

DefaultGroupName=HH Listing CAPI System

Compression=lzma2

SolidCompression=yes

;set location where installation .exe file is output to

OutputDir=C:\CC81\Utility\inno

;set default path for source folders in the [Files] section

SourceDir="C:\CC81"

;set the name of the installation file

OutputBaseFilename=CC81\_Setup

;disable default creation of uninstall.exe since we don't usually want the user

;to be able to uninstall the system

Uninstallable=no

;locations of image files used for the install windows

WizardImageFile=C:\CC81\Utility\inno\dhs.bmp

WizardSmallImageFile=C:\CC81\Utility\inno\dhs.bmp

;Use this section to specify which files and folders need to be installed on the target machine

;Note that by default folders are NOT copied recursively

;To make this happen use the recursesubdirs flag as shown for the entries for Utility and PII below

[Files]

Source: "Entry\\*.\*"; DestDir: "{app}\Entry"

Source: "Library\\*.\*"; DestDir: "{app}\Library"

Source: "Superv\\*.\*"; DestDir: "{app}\Superv"

Source: "PII\\*.\*"; DestDir: "{app}\PII";Flags:recursesubdirs

Source: "SampList\\*.\*";DestDir: "{app}\SampList"

Source: "Dicts\\*.\*"; DestDir: "{app}\Dicts"

Source: "Utility\\*.\*"; DestDir: "{app}\Utility"; Flags:recursesubdirs

Source: "Docs\\*.\*"; DestDir: "{app}\Docs"

[InstallDelete]

; deletes any existing files/directories from previous installations

; note that if the Type is filesandordirs, this will delete all subfolders in

; a specified folder

Name: {app}\PII; Type: filesandordirs

Name: {app}\zipdata; Type: filesandordirs

Name: {app}\upgrades; Type: filesandordirs

[Dirs]

;create empty folders

Name: "{app}\PII\data"

Name: "{app}\PII\work"

Name: "{app}\PII\temp"

Name: "{app}\PII\final"

Name: "{app}\PII\work"

Name: "{app}\Sampdata"

Name: "{app}\zipdata"

Name: "{app}\upgrades"

[Run]

;Use this section to run installation programs for CSPro and any other programs that need to be installed

;Ideally all setup lines for software installation should use the silent install parameters if available or known

Filename: "{app}\utility\others\AcroRdrDC1500820082\_en\_US.exe"; Parameters:"/sPB /rs"; StatusMsg:"Installing Adobe Reader";

; Windows Defender Update

Filename: "{app}\utility\others\mpam-fe64.exe"; Parameters:"-q";StatusMsg:"Updating Windows Defender";

;teamviewer

Filename: "{app}\utility\others\teamviewer\_setup.exe"; Parameters:"/S";StatusMsg:"Installing Teamviewer";

;cspro

Filename: "{app}\utility\others\cspro63.exe";Parameters:"/s /sms /f1{app}\utility\others\cspro63.iss";StatusMsg:"Installing CSPro 6.3";

; set up user account password on tablets

; need double quotes around %username% for user names that have spaces

; need the "" to escape the quotes for the inno compiler

; !! don't forget to change cc81password to the actual password for the project !!

Filename: "{sys}\cmd.exe"; Parameters:"/c net user ""%username%"" cc81password";StatusMsg:"setting windows password";

[Icons]

;create icons for menu apps and put on desktop

Name: "{commondesktop}\Interviewer"; Filename:"c:\CC81\entry\dcmenu.pff";

Name: "{commondesktop}\Supervisor"; Filename:"c:\CC81\superv\supmenu.pff";