

**Packing List Builder** is a tool used to generate Hypercom XML Packing List files.

An HXP file is a zipped archive that contains:

- A set of files that make-up an Optimum terminal application.
- A packinglist.xml file that describes the application.

HXP files are set-up within Term-master to download the application files to an Optimum terminal.

HXP file contain only one application.

To create an HXP file using GUI:

1. Press "New" under "Files" menu or user keyboard shortcut Ctrl-N. A new document form appears.
2. Populate the fields in Application properties group box. The Application Title, Company Name and Destination Directory are required.
3. Drag-n-drop files from MS Explorer or any other file manager onto the Files group box document surface or use the "Add file" button and browse to your desired files.
4. Mark the application's main file using "Set main" button. That file will change it's color in the files grid.

Note: You can not create a valid HXP file without marking one of the files as the "Main File".

5. Mark the application's icon file using "Icon file" button. That file will change it's color in the files grid.

Note: If no icon file is set, Packing List Builder will generate a warning.

6. Save the HXP file using "Save" or "Save As" under the "File" menu.

For more information on the format of the HXP and descriptions of the application properties please see [PACKINGLIST.XML specification](#).

## Packing List Builder command line interface.

Command line syntax will be following:

```
PackingListBuilder -<options> hxp_file app_file1 app_file2 ...
```

**hxp\_file** is a name of HXP file to operate with.

**app\_file1, app\_file2...** is a space-separated list of files to add to hxp.

**options** are:

Short form	Long form	Description
<b>-t</b>	<b>--title</b>	This is the Application Title. Also this is associated to the application in the Shell
<b>-v</b>	<b>--version</b>	This is the application version number
<b>-c</b>	<b>--company</b>	Company name. This attribute is mandatory
<b>-d</b>	<b>--destination</b>	Destination directory attribute.
<b>-m</b>	<b>mode</b>	Main attribute. specified as a string, not an ordinal value. :[ "Desktop Application"   "No Desktop launch Desktop Application"   "Auto-launch System Application"]
<b>-o</b>	<b>--config</b>	Configuration file name.(“Hypercom.C

<b>-a</b>	<b>--main</b>	Main application file name. It is a filename (For example "main.app")
<b>-i</b>	<b>--icon</b>	Icon file name. It is a filename without extension
<b>-s</b>	<b>--silent</b>	Silent mode ( no warning messages will be displayed)
<b>-l</b>	<b>--log &lt;filename&gt;</b>	Log errors to a specified file.
<b>-u</b>	<b>--update</b>	Update an HXP file. This key is used to update an HXP file.
<b>-x</b>	<b>--template</b>	Use XML template. XML template is just a document describing HXP file. In fact it is PackingLst.XML. To generate XML template user "Export XML" menu item under "File" menu.
<b>-n</b>	<b>--newfile</b>	Create new file. This attribute is needed when PLB processes an XML template file. By default PLB updates specified HXP file and if an HXP file name is absent on the command line, PLB will create a new file.
<b>-g</b>	<b>--nogui</b>	Creates or updates a file without starting the application. This switch is used only with -n or -u switches only.
<b>-h</b>	<b>--help</b>	Shows a command-line syntax
	<b>--dependencylist</b>	The Dependency List is used by Appli to start the listed application (and any applications listed applications are dependent on)

		target application. The maximum Dep is 512 characters.
	<b>--name</b>	“Application Unique Name”. The Appli by Application Manager to compare n; dependency list. An application may n dependency list but its name may be l application’s dependency list. The App used by Application Manager to uniqu application. No two applications will h; This is a mandatory field. HXP wouldr specifying this field. The Application U maximum length is 128 characters.

You may also check [command line samples](#)

### Application Property Values

Below is a table of parameters, which should or might be specified during HXP file creation and their default values for a case of omitting.

Property Name	Property Type	Default
Application Title	Mandatory	None
Application Unique Name	Mandatory	None
Version	Optional	None

Company name	Mandatory	None
Destination directory	Mandatory	None
Main attribute	Mandatory	None
Configuration file name	Optional	“Hypercom.CFG”
Application file name(s)	Mandatory	None
Main application file name	Mandatory	None
Dependency List	Optional	None
Icon file name	Optional	None
Release Date	Automatic	PLB will set the date of file creation

You can use a command-line interface to Packing List Builder to update the content of HXP files.

Command-line syntax is:

PackingListBuilder.exe [ [/update] [/log] filename]

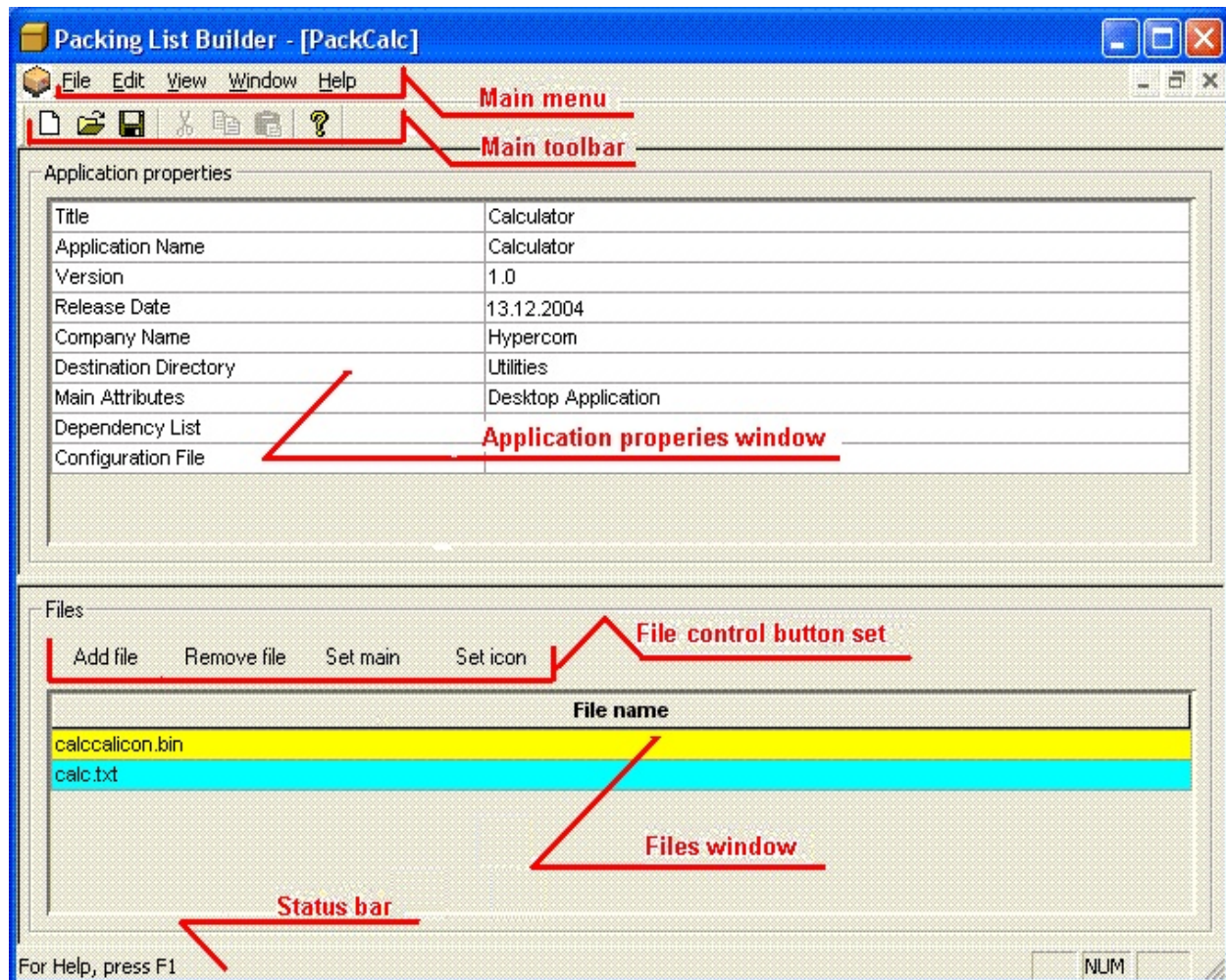
<b>Key</b>	<b>Description</b>
/update	The HXP file <code>_filename_</code> should be updated. In case of any error a message box will appear.
/log	In the case of a failed update an error log file named as <code>_filename_.err</code> is placed in the same directory as <code>.</code> . The error message box is not displayed and error messages will be written to <code>_filename_.err</code> .
filename	Name and path of an HXP file to operate with.

Packing List Builder will return a -1 if any error occurs and 0 if the HXP update is successful.

Using only `_filename_` will open the HXP file in Packing List Builder.

## Using main screen GUI:

Main screen looks like it shown below:



## Application properties window:

Used to manage application properties. Please refer to [Packing List file format](#) for details.

**Main menu** commands are described [here](#)

**Files window** shows the files which will be included into HXP package.

**File control button set** used to operate with files within HXP package.



Add file	Open a browser to allow add files to an HXP package.
Remove file	Remove selected file from HXP package.
Set main	Toggle function, which mark selected file as "Main" for the application or clear this file attribute.
Set icon	Toggle function, which mark selected file as "Icon" for the application or clear this file attribute.

## File menu commands

The **File** menu offers the following commands:

New	Creates a new document.
Open	Opens an existing document.
Close	Closes an opened document.
Save	Saves an opened document using the same file name.
Save As	Saves an opened document to a specified file name.
Export XML	Exports XML from opened HXP file. To see this option, at least one HXP file must
Exit	Exits Packing List Builder.

## **Edit menu commands**

The **Edit** menu offers the following commands:

Cut	Deletes data from the document and moves it to the clipboard.
Copy	Copies data from the document to the clipboard.
Paste	Pastes data from the clipboard into the document.

## View menu commands

The **View** menu offers the following commands:

Toolbar	Shows or hides the toolbar.
Status Bar	Shows or hides the status bar.

## Window menu commands

The Window menu offers the following commands, which enable you to arrange multiple views of multiple documents in the application window:

New Window	Creates a new window that views the same document.
Cascade	Arranges windows in an overlapped fashion.
Tile	Arranges windows in non-overlapped tiles.
Arrange Icons	Arranges icons of closed windows.
Window 1, 2, ...	Goes to specified window.

## Help menu commands

The **Help** menu offers the following commands, which provide you assistance with this application:

[Help Topics](#)

Offers you an index to topics on which you can get help.

[About](#)

Displays the version number of this application.

## **Toolbar**

The toolbar is displayed across the top of the application window, below the menu bar. The toolbar provides quick mouse access to many tools used in Packing List Builder,

To hide or display the toolbar, click **Toolbar** from the **View** menu.

## Status Bar

The status bar is displayed at the bottom of the Packing List Builder window. To display or hide the status bar, use the **Status Bar** command in the **View** menu.

The left area of the status bar describes actions of menu items as you use the arrow keys to navigate through menus. This area similarly shows messages that describe the actions of toolbar buttons as you press them, before releasing them. If after viewing the description of the toolbar button command you wish not to execute the command, then release the mouse button while the pointer is off the toolbar button.

The right areas of the status bar indicate which of the following keys are latched down:

<b>Indicator</b>	<b>Description</b>
------------------	--------------------

CAP	The Caps Lock key is latched down.
-----	------------------------------------

NUM	The Num Lock key is latched down.
-----	-----------------------------------

SCRL	The Scroll Lock key is latched down.
------	--------------------------------------



Packlist.xml is a file that describes an application within the HXP file.

## **PACKING LIST FORMAT.**

< XML Header >

< Application

```
Title = " ... "  
UniqueName="..."  
Version = " ... "  
Release Date = " ... "  
Icon = "File"  
Company = " ... "  
DestinationDir = " ... "  
Main = "MainFile "  
MainAttributes = "..."  
DependencyList="..."  
<Files>
```

< File

```
URL = " ... "  
Size = " ... "  
MD5 = " ... "  
SHA1 = " ... "  
Date = " ... "  
Compression = "Y" or "N"  
Uncompress = "Y" or "N"  
UserName = "..."  
Password = "..."
```

>

```
</File>
```

```
...
```

```
<File
```

```
.../>
```

```
</Files>
```

```
<StartUp
```

```
CommandLine = " ... ">
```

```
</StartUp>
```

```
</Application>
```

## **TAGS AND ATTRIBUTES**

## < Application> Tag attributes:

**Title:** This is the application Title. Also this is the name associated to the application in the Shell.

**UniqueName:** The Application Name is used by Application Manager to compare names in the dependency list. An application may not have a dependency list but its name may be listed in another application's dependency list. The Application Name is used by Application Manager to uniquely identify an application. No two applications will have the same name. This field can contain maximum 128 characters.

**Version:** This is the application version number. This number should be displayed during uploading. This is information for the User/Help desk.

**ReleaseDate:** This is the application release date. This is information for the User/Help desk. The format should be according to ISO 8601 standard.

**Icon:** This is an icon file name. This icon file is used in Term Master and Terminal Shell to depict the application. The attribute value is a file name described as one of the File Tag.

**Company:** This is the company name of the application. This name will be used as a top destination directory that is located above the Application destination directory. It can also be used together with the Company Program Key for secure downloading.

**DestinationDir:** This is the destination application directory that should be under the company directory (Second level directory).

**Main:** Program file name to run the application. The Shell calls this file.

**MainAttr:** This field contains the attributes for the current application. It can contain the following attributes:

Attribute	Value	Description
Desktop Application	0	The application will appear on the main desktop screen (Shell). The icon will be used to depict the application.
No Desktop Application	1	The application will be downloaded and used for other applications. It will not appear on the desktop screen.
Auto-Launch Desktop Application	2	Desktop application is automatically launched after a successful download.
Auto-Launch Application	3	This attribute describes the applications that need to be launched in memory automatically after being successfully downloaded by the Application Manager.
System application	4	Application Manager program

**DependencyList:** The Dependency List is used by Application Manager to start the listed application (and any applications that the listed applications are dependent on) before starting the target application. This field can contain 512 characters maximum.

## <Files> Tag attributes:

**File:** This Tag describes a file that belongs to the Application. This Tag contains a list of attributes distinguishing the file.

**URL:** Full (Absolute) URL to the resource file (file can be compressed if compression Tag is "Y"). Example: http://manifests/[TerminalID]/somefile.ext  
Currently domain names present in the URL are not accepted. Request Manager will never serve requests for Absolute Path Files (this may create a severe security risk). Request Manager will provide the root directory for file storage and all application files will be located in that directory sub-tree. If a request for a file from the root of the hard drive were allowed, a path to password files would become clear. Therefore, no servers have such a possibility by default (IIS = wwwroot, ftproot, etc.). Access to Term-Master will be protected by a secure certificate, but when one certificate for one POS is discredited, another layer of protection will keep all systems stable and safe. It is applied and is up to the user to get files from another server, either FTP or HTTP, but it will be done through another server with its own security.

**MD5:** 128-bit message digest value of the file. This attribute is used by the Terminal to compare to what is already present in the Terminal File System. If this value differs then this file will be downloaded and replace the one that currently is present in the File System. This parameter works together with the size to check if it is necessary to update this file.

**SHA1:** Another digest value of the file.

**Size:** File size in bytes. This attribute is used by the Terminal to compare to what is already present in the File System. It is used together with the Checksum and Date to check if it is necessary to update this file.

**Date:** Date of the file in YYYYMMDD format.

**Compression:** File is sent in compressed format ("Y" or "N")

**Uncompress:** File is to be stored uncompressed in terminal file system ("Y" or "N")

**UserName:** This field is used for FTP server connections as the user name to login to the server.

**Password:** This field is used for FTP server connections as the password to login to the server.

**<StartUp> Tag attributes:**

**Command Line:** This is the command line executed when the Application runs.

## Command line usage examples:

1. Create an HXP file with GUI. When this command line is started, PLB opens a main window with populated properties. You can edit them and press "Save file".

```
PackingListBuilder.exe -n --name  
"12345678901234567890123456789012345678901234567890123456" --title  
"GDBSVR rs232" --company Hypercom --destination GDB -m  
"Desktop Application" --main Load.vbs CorrectCreateFile.hxp  
..\Files\DownloadFileAll.vbs ..\Files\Init.vbs ..\Files\Load.vbs  
..\Files\Load_00000001.vbs
```

Please note that in command line input files are allowed with relative paths.

2. Create an HXP file without showing GUI

```
PackingListBuilder.exe -n -g -l myfile.err --name  
"123456789012345678901234567890123456789012345678901234" --title  
"GDBSVR rs232" --company Hypercom --destination GDB -m  
"Desktop Application" --main Load.vbs CorrectCreateFile.hxp  
..\Files\DownloadFileAll.vbs ..\Files\Init.vbs ..\Files\Load.vbs  
..\Files\Load_00000001.vbs
```

Please note that no error messages will appear on the screen. In case of any errors (PLB will return 1), please review the log file. In this example, the log file name is "myfile.err".

To avoid questions like "Icon file not specified, save file anyway?" please specify **-s** ("silent mode") command line option.

3. Create an HXP file using template XML file.

```
PackingListBuilder.exe -n -g -l "Template.err" -x TestTemplates.xml  
ByTemplates.hxp
```



To create a template file from any HXP file, open HXP file with PLB tool and choose "Export XML" option under "File" menu.

4. Create an HXP file with Dependency List and not showing GUI.

```
PackingListBuilder.exe -n -g -l myfile.err --dependencylist "12345" -  
-name "12345678901234567890123456789" --title "GDBSVR rs232"  
--company Hypercom --destination GDB -m "Desktop Application" --  
main Load.vbs CorrectCreateFile.hxp ..\Files\DownloadFileAll.vbs  
..\Files\Init.vbs ..\Files\Load.vbs ..\Files\Load_00000001.vbs
```

5. To update existing HXP file

```
PackingListBuilder.exe /update -l test.err CorrectCreateFile.hxp  
or  
PackingListBuilder.exe -u -g -l test.err CorrectCreateFile.hxp
```

Please note that the "/update" option still exists to have compability with previous versions of PLB.

## **Packing List Builder Help Index**

[What is Packing List Builder](#)

[How do I create an HXP file](#)

[How do i create and update an HXP file form the command fle](#) Also please check command line [samples](#)

[HXP file format](#)

### **Commands**

[File menu](#)

[Edit menu](#)

[View menu](#)

[Window menu](#)

[Help menu](#)

## **Index command (Help menu)**

Use this command to display the opening screen of help. From the opening screen, you can jump to step-by-step instructions for using Packing List Builder and various types of reference information.

Once you open help, you can click the **Contents** button whenever you want to return to the opening screen.

## **About command (Help menu)**

Use this command to display the copyright notice and version number of your copy of Packing List Builder.