

# Collaboration Program for Scripted Gallery 2 Installers

As an alternative to the official Gallery 2 Installation wizard there are providers of scripted and automated Gallery 2 installation tools.

The **goal** of the *Collaboration Program for Scripted Gallery 2 Installers* is to improve customer satisfaction by solving common challenges of scripted Gallery 2 installation tools.

As a provider of Gallery 2 installation products **you** are asked to follow the specified recommendations and invited to collaborate with the Gallery Project and other installation tool providers to work towards the declared goal.

---

<b>1</b>	<b>The Nature of Scripted Gallery 2 Installers.....</b>	<b>2</b>
<b>2</b>	<b>Recommendations .....</b>	<b>2</b>
<b>3</b>	<b>Implementation Design .....</b>	<b>4</b>
<b>4</b>	<b>Collaboration and Feedback .....</b>	<b>6</b>
<b>5</b>	<b>Summary .....</b>	<b>6</b>
<b>6</b>	<b>List of Scripted Gallery 2 Installation Products.....</b>	<b>6</b>
<b>7</b>	<b>About Gallery .....</b>	<b>7</b>

---

# 1 The Nature of Scripted Gallery 2 Installers

## DEF 1 Scripted Gallery 2 Installation

“A scripted Gallery 2 installer allows the installation of Gallery 2 with limited or no user interaction.”

The **typical properties** of scripted Gallery 2 installers are:

- **Easier and faster** installation of Gallery 2, often with a single mouse-click or a single step with fewer input fields.
- **Hiding complexity** by offering no or very few customer-facing configuration options.
- **Pre-configuration** of the database connection information, the storage folder and other data is usually making the above possible.

The **typical problems** with scripted Gallery 2 installers are:

- Often there is **no upgrade path** provided.
- Versions are often **out of date** and new releases and more importantly **security** patch releases are offered with a large delay after the official Gallery 2 releases.
- **Unclear escalation path** and **lack of documentation** in case support is needed.

## 2 Recommendations

The following recommendations are key to customer satisfaction throughout the life-cycle of a Gallery 2 installation and are direct consequences of the previously identified problems.

### 2.1 Keep It Up To Date

Gallery 2 is actively developed. Major releases appear every year and important security patch releases can be announced any day.

It is imperative to **protect customers** by providing the latest stable release with your scripted Gallery 2 installer ideally as a co-launch with the official Gallery 2 releases.

Furthermore, **customers clearly prefer the latest release** of Gallery 2 over obsolete versions that are available through some automated installation tools.

To help you keeping your scripted Gallery 2 installer up to date with the latest stable Gallery 2 release, we highly recommend the following actions.



**RECOMMENDATION:** Subscribe to the gallery-announce mailing list ([http://codex.gallery2.org/index.php/Mailing\\_Lists](http://codex.gallery2.org/index.php/Mailing_Lists)) to get notified about Gallery releases as they happen.



**RECOMMENDATION:** Release a new version of your scripted Gallery 2 installer as soon as possible. You can start development and QA as early as the announcement of the first release candidate of a new Gallery 2 release.

## 2.2 Provide an Upgrade Path

The installation is just the start of the life-cycle of a Gallery 2 instance. Customers are either tempted by **new features** or forced due to **security patch releases** to keep their own Gallery 2 installation up to date.



**RECOMMENDATION:** Include an upgrade path for Gallery 2 in the design of your product. In case you offer a replacement for the official upgrade wizard of Gallery 2, please consider the challenges outlined in chapter 3 (Implementation Design).



**RECOMMENDATION:** **Inform the end-user** of the designed upgrade path. Do you offer an automated upgrade or should the user follow the official Gallery 2 upgrade instructions? Provide links to the *Gallery codex* (<http://codex.gallery2.org>), the home of the Gallery 2 documentation.

## 2.3 Plan for Support and Maintenance

When using your installation tool, a customer might be confused about some options or even worse, there could be an unexpected problem.



**RECOMMENDATION:** Provide **inline documentation** and links to further information in user-facing interfaces.

Who should the customer contact if there is a problem with Gallery 2 or with its installation? Where to check for updates?



**RECOMMENDATION:** Add **contact information** and explain who is providing what software component.

What changes have been made compared to an official Gallery 2 release? Who is responsible for the support and the maintenance? As a distributor of an adapted Gallery 2 release you have certain responsibilities. The Gallery Project cannot support an unknown product. Your redistributed Gallery 2 version is a black box for the Gallery community, unless it's open or clearly documented.



**RECOMMENDATION:** Publicly **document and communicate your changes** compared to the official Gallery 2 release to the Gallery Project *or* document that you or a 3<sup>rd</sup> party is responsible for support and maintenance.



**RECOMMENDATION:** **Provide an escalation path** in case the Gallery Support Team needs additional information or help when supporting customers of your re-distributed Gallery 2 version.

## 2.4 Provide Added Value

The above mentioned typical properties of scripted Gallery 2 installers are excellent examples of how added value can be achieved - Make the installation of Gallery 2 easier and faster by pre-configuration and hiding configuration that is rendered unnecessary for specific server environments.

# 3 Implementation Design

When offering a wrapper or a even a replacement for the official Gallery 2 installation and upgrade wizards, it is important to know the anatomy of these components and what common challenges have to be considered.

## 3.1 Timeouts and Resource Limits

The installation and upgrade process of Gallery 2 can take longer than one, two minutes, depending on the server. The official installation and upgrade wizards of Gallery employ the following measures to mitigate related problems:

- To combat the **PHP execution time limit**, Gallery 2 dynamically extends the limit during operation. Removing the time limit before control is given to Gallery code does not solve the problem since the limit is set multiple times in Gallery's installation, upgrade and API code overriding previous values.
- Usually, there is a **web server request time limit** (typically 1, 5 or 10 minutes). After this limit, the PHP process is usually killed and the TCP/IP connection closed. To mitigate this limit, Gallery periodically sends a few bytes to the client until the long running operation is complete.
- The same measure also helps to prevent **user-agent connection timeouts**. Periodically sending a few bytes keeps the connection alive.
- And finally, the same measure also helps to **keep the customer from closing the connection** and thus killing the upgrade process. The few bytes that are periodically transmitted are in fact the building blocks of a progress bar that is displayed in the web browser. Thereby the user has an estimate of the remaining time and knows that it's working correctly.

- Be prepared to allow for increased **resource usage** during Gallery 2 installation and upgrade processes.

## 3.2 Dynamic Dependencies

While the *System Checks* step of the installation and upgrade wizards checks for all necessary dependencies for the core module, each module can have additional requirements. A module might depend on a specific PHP extension (e.g. XML) or on some external applications.

If an installation works on one server, you cannot just replicate the application image on other servers without guaranteeing that the setup is truly identical.

Alternatively, by using the correct Gallery 2 API methods during your scripted installation you can guarantee that all dependencies are checked for dynamically at installation time as in the official wizards.

## 3.3 Non-Trivial Changes between Releases

The upgrade code of Gallery 2 can change the database schema, data in the storage directory and in the database can be transformed and the configuration file can be changed as well. Everything can change between releases and only running the official upgrade code guarantees correct behavior.

## 3.4 Missing and Incompatible Plugins on Upgrade

Customers can install additional Gallery 2 modules and themes by uploading files to their `modules/` and `themes/` folder. Since Gallery 2.2, this is even easier - no FTP access is required anymore thanks to the *downloadable plugins* feature.

When providing a customized upgrade path you have to be aware that the customer might have deleted or added plugins. The official upgrade wizard tries to gracefully deactivate plugins if there is no new version available on the server.

## 3.5 Clearing Cached Data on Upgrade

- After putting the new PHP code into place and before starting the upgrade process, you should clear the **PHP opcode cache** (Zend, eAccelerator, APC, etc) if appropriate.
- The **CacheMap database table** should be truncated before upgrading the core module since MySQL tends to have problems with this large table which could result in upgrade errors.
- **Gallery's database and template caches** should be cleared after the core module upgrade.

## 4 Collaboration and Feedback

The Gallery team is not only looking forward to learning more about the details of your installer and to collaborating on end-user support, but would also like to support you in implementing the above mentioned recommendations.

Please let us know what kind of hooks in Gallery 2 would simplify your own implementation. Depending on development resources, there will be an official scripted installer for Gallery 2 which could serve as the base for your Gallery 2 package.

We are happy to answer your questions and ready to implement changes in Gallery 2 to resolve common challenges. You can contact the Gallery Project Team at: [gallery-devel@lists.sourceforge.net](mailto:gallery-devel@lists.sourceforge.net).

## 5 Summary

The *Collaboration Program for Scripted Gallery 2 Installers* can be summarized as follows:

- Keep your scripted installer up to date. Subscribe to the gallery-announce mailing list to get notified about new releases and strive to co-launch your products with the official Gallery 2 releases.
- Provide an upgrade path or clearly document that the official upgrade wizard should be used.
- Plan for support and maintenance. Provide an escalation path to the Gallery Support Team, document your changes and communicate the contact information and responsibilities to the end user.
- Know the official Gallery 2 installation and upgrade wizards and how they solve common challenges.
- Let us know how we can help you achieve the above mentioned goals.

## 6 List of Scripted Gallery 2 Installation Products

Providers of scripted installers for Gallery 2:

- 4PSA-Integrator - <http://www.4psa.com/products/4psaintegrator.php>
- CPSkins cPanel Auto-Installer and Plesk PSA-Installer - <http://cpskins.com>
- Fantastico De Luxe - <http://netenberg.com/fantastico.php>
- Installatron - <http://www.installatron.com>
- Plesk Application Vault / Application Pack - <http://www.swsoft.com/plesk>

In-house solutions:

- DreamHost - <http://www.dreamhost.com/>
- PowWeb - <http://www.powweb.com>
- Yahoo! - <http://smallbusiness.yahoo.com/webhosting/>

## 7 **About Gallery**

Gallery is an open source project with the goal to develop and support leading photo sharing web application solutions.

Website: <http://gallery.menalto.com>

Further information: <http://codex.gallery2.org/index.php/Gallery:About>