



What's new in GPM

GPM

It's been a year, let's talk



Derek Trepanier
Derek@mooregoodideas.com



What is G Package Manager

- An Open Source tool to build and consume LabVIEW packages
- Installs packages locally
- Keeps information about package dependencies
- Easily restore package configuration
- Public centralized registry that anyone can post to in seconds



NIPM vs GPM

NIPM

- Based on Linux Packages, which distribute Linux Applications
- Closed source, maintained by National Instruments
- Installs runtime code
- Installs packages to a system

Great for Application
deployment!

GPM

- Based on NPM, which distributes JavaScript source code
- Open Source maintained by LabVIEW community
- Installs source code
- Installs packages to a project

Great for reuse library
distribution!



How to use GPM

- **Install the application**

- Instructions at: <http://gpackage.io>

- **Initialize your project**

- Run `gpm init` from the working directory
- Open the GPM browser to the working directory

- **Find packages**

- Search <http://gpackage.io> and use the command `gpm install <package>
- Use the GPM Browser application

- **Publish a package**

- Create an account on <http://gpackage.io>
- Install other packages (gpm automatically keeps track of your dependent pkgs)
- Develop your code
- Create a readme to tell people what your package does
- Publish to the registry with the command `gpm publish`

Your package is now accessible by anyone instantly!



New Commands

- ``gpm relink``
 - Manually force a relink of gpm packages using the package dependency tree
- ``gpm recompile``
 - Manually force a recompile of you entire project
- ``gpm upublish``
 - Remove a package from the registry
 - Must be preformed within 36hrs of the original publish
- ``gpm deprecate``
 - Remove a package from search results with out removing the package
 - Ensure new projects don't use the package without breaking old projects

Special thanks to the Composed Systems teams for the relink and recompile commands!

New: Dev Dependencies

- Ability to add dependencies to your project that help with development, but aren't required for consumption of the packages
- When the package is installed by a consumer, dev dependencies are not installed
- Use cases:
 - Unit test suite
 - Monitored Actor Window

```
Panel Actors > | gpackage.json > {} devDependencies
1  {
2    "author": {
3      "email": "support@mooregoodideas.com",
4      "name": "Moore Good Ideas, Inc",
5      "url": "https://www.mooregoodideas.com"
6    },
7    "bugs": "https://gitlab.com/mgi/gpm-packages/panel-actors\\issues",
8    "contributors": [
9      {
10       "email": "derek@mooregoodideas.com",
11       "name": "Derek Trepanier",
12       "url": "https://www.mooregoodideas.com/"
13     }
14   ],
15   "dependencies": {
16     "@mgi/monitored-actor-core": "^3.0.2",
17     "@mgi/panel-manager": "^2.0.0"
18   },
19   "description": "An Actor Framework implementation of MGI's Panel Manager Framework",
20   "devDependencies": {
21     "@mgi/monitored-actor": "^3.0.1"
22   },
23   "files": [
24     "Source/"
25   ],
26   "keywords": [
27     "UI"
28   ],
29   "license": "BSD-3-Clause",
30   "lvversion": ">=17",
31   "name": "@mgi/panel-actors",
32   "repository": "https://gitlab.com/mgi/gpm-packages/panel-actors",
33   "version": "3.1.1"
34 }
```



New: File System Registries

- Registries that can be hosted on the local (or network) file system
- Perfect for hosting “internal” packages
- GPM manages the file system operations
- Seamlessly link to packages hosted both on the file system and publicly
- Demo

New: Bug fixes

- Multiple user-reported bugs were fixed
- Thanks to: [Ethan Stern](#) , [Jon McBee](#), [Steve Ball](#), [Michael Lambert](#), [S Bates](#) and [Matt Baker](#) for reporting bugs
- Special shout out to Steve Ball (Composed Systems) and Quentin Aldridge (QSI) for fixing bugs and submitting merge requests





New: Packages on gpackage.io

Moore Good Ideas

- [@mgi/json](#) – Quickly parse and generate JSON strings from LabVIEW data types
- [@mgi/library-core](#) – Super simple, but super useful VIs that I use on *every* project
- [@mgi/standard-windows-error-reporter](#) – A flexible and extendable general error reporter

Composed Systems

- [@cs/lookup-table](#) - A malleable, by-reference, lookup table API with in-placeness optimization and type abilities
- [@cs/event-logger](#) - an extendable, ready to use LabVIEW event logger.
- [@cs/test-executive](#) - A test executive library for adding sequencing with flow control to your application.

I need help!

- This is beta software
- The repo's issue tracker is populated with desired features and bugs
- **Submit your merge requests NOW!**
- <https://gitlab.com/mgi/gpm/>





Todo List

Open Questions

- Pallets?
- Favorites?
- Quickdrop Integration?
- “Global” package installation?

Improvements

- Better browse abilities
- Error Reporting
- More Documentation
- Icons/Logos

Fill Up the Registry!