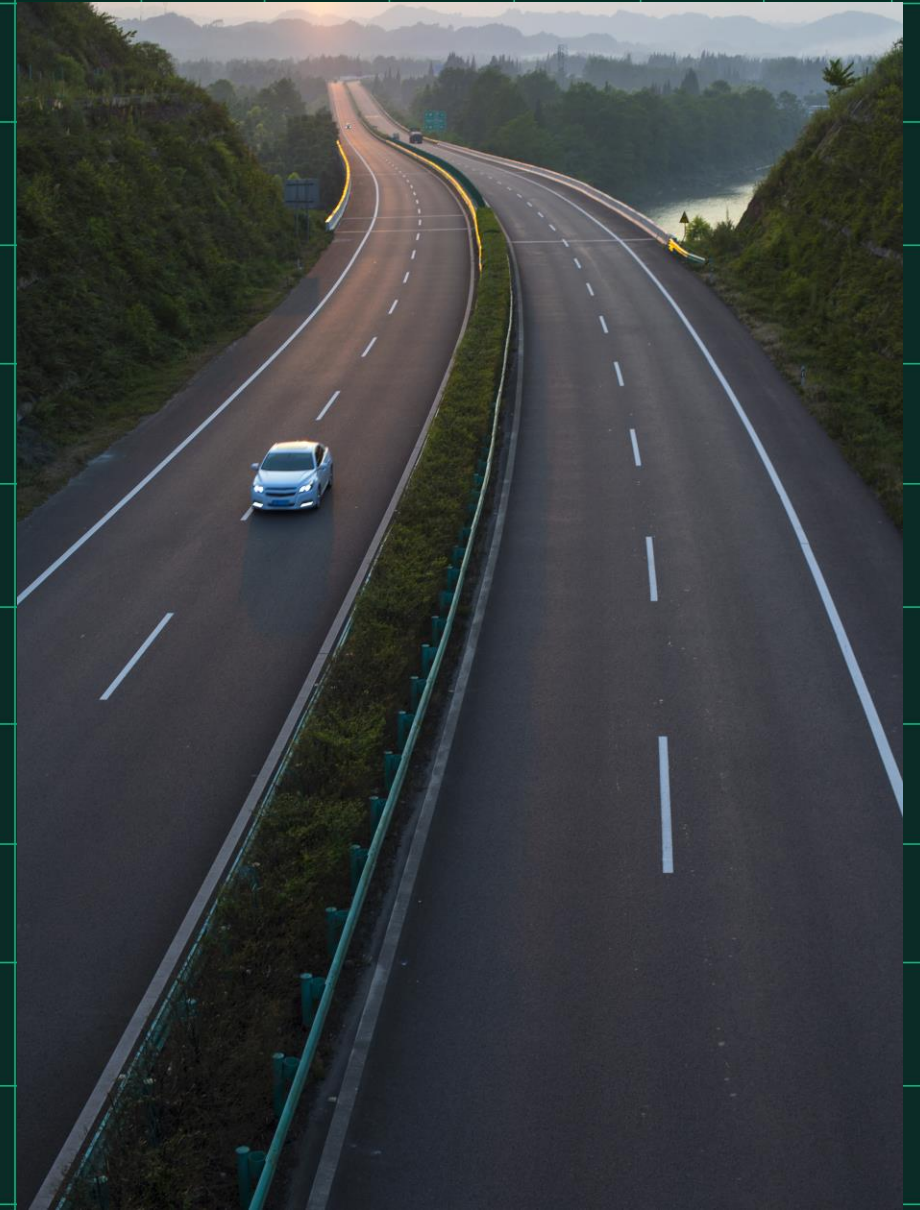


Best Practices for Distributing IP

Niki Budgell

Scott Richardson

Jim Kring



Agenda

- Historical goals for distributing IP
- NI platform solutions for distributing your IP
- What's next: roadmap and exploring your needs

The Goals of Distributing IP?

What is Software IP?

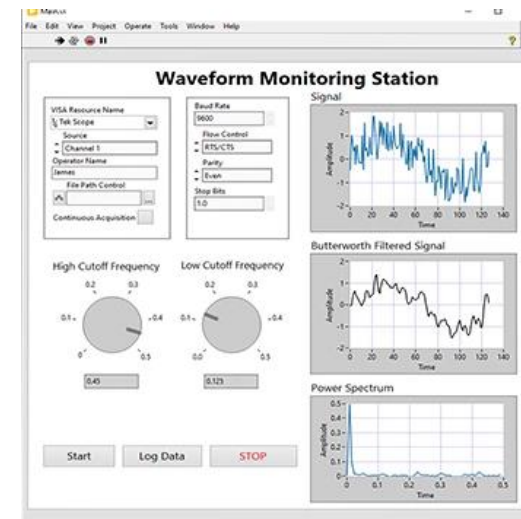
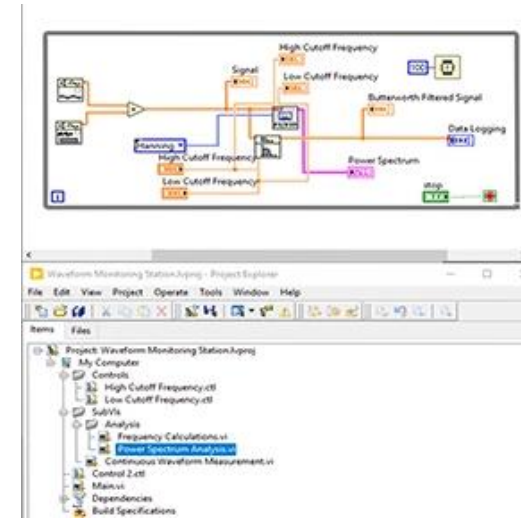
in·tel·lec·tu·al prop·er·ty (noun)

Definition – a work or invention that is the result of creativity, such as a manuscript or a design, to which one has rights and for which one may apply for a patent, copyright, trademark, etc.

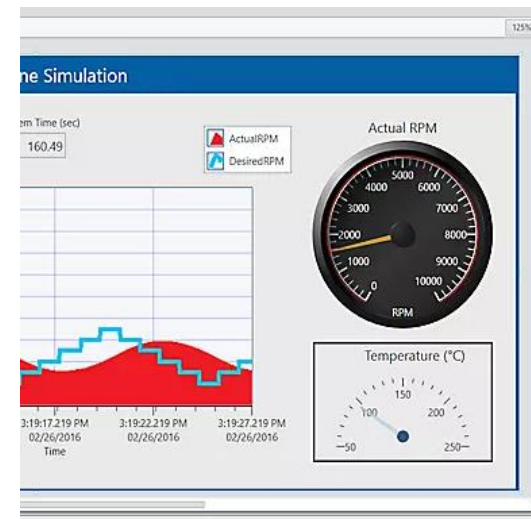
(From Oxford Languages)

For software,

- the source of IP is typically code, algorithms, models, data, etc.
- we might share our IP in the forms of source code, libraries, and applications
- how we protect our IP varies from open-source collaboration to protection by compiling into binaries, or even just providing a web service



Category	Unit	Jan-15	Feb-15	Mar-15	Apr-15	May-15	Jun-15	Jul-15	Aug-15	Sep-15	Oct-15	Nov-15	Dec-15	Jan-16	Feb-16	Mar-16	Apr-16	May-16
1-Personal	0	12,034	13,565	10,874	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390
2-Marketing	1	521	347	178	519	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557
3-Commu	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4-Other	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5-Web Advertising	1	12,900	16,646	11,195	15,657	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390
6-Personnel Total	2	6,000	2,300	3,000	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100
7-Web Research	1	2,000	4,500	5,000	8,000	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100
8-Independent Research	2	8,200	12,620	10,000	14,600	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390
9-Firm Research Fees	3	16,200	19,000	1,244	573	323	612	612	612	612	612	612	612	612	612	612	612	612
10-Market Research Total	2	622	431	1,244	14,993	12,880	234	34	34	34	34	34	34	34	34	34	34	34
11-Promotions	3	532	10,432	136	587	13,900	13,900	13,900	13,900	13,900	13,900	13,900	13,900	13,900	13,900	13,900	13,900	13,900
12-Web Advertising	1	1,243	12,418	16,505	12,000	12,000	12,000	12,000	12,000	12,000	12,000	12,000	12,000	12,000	12,000	12,000	12,000	12,000
13-Direct Marketing	3	12,682	19,333	15,000	19,333	15,000	19,333	15,000	19,333	15,000	19,333	15,000	19,333	15,000	19,333	15,000	19,333	15,000
14-Newspaper Advertising	4	19,300	150	150	200	200	200	200	200	200	200	200	200	200	200	200	200	200
15-Communication Total	0	200	500	100	251	251	251	251	251	251	251	251	251	251	251	251	251	251
16-Travel	4	683	153	153	15,611	13,995	1	1	1	1	1	1	1	1	1	1	1	1
17-Computer/Office Equipment	2	20,583	13,585	10,874	1,953	1,953	1,953	1,953	1,953	1,953	1,953	1,953	1,953	1,953	1,953	1,953	1,953	1,953
18-Postage	4	12,034	347	178	519	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557
19-Personnel	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20-Other Total	1	521	347	178	519	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557	1,557
21-Benefits	1	12,900	16,646	11,195	15,657	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390
22-Payroll taxes	1	2,000	4,500	5,000	8,000	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100
23-Salaries	1	6,000	2,300	3,000	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100	2,100
24-Commissions and bonuses	2	8,200	12,620	10,000	14,600	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390
25-Personnel Total	1	2,000	4,500	5,000	8,000	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100	10,100
26-Web Research	2	8,200	12,620	10,000	14,600	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390	12,390	13,599	18,390
27-Independent Research	3	16,200	19,000	1,244	573	323	612	612	612	612	612	612	612	612	612	612	612	612
28-Total	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



Early needs when distributing IP

In 200x, developers and end-use were not highly connected; most Windows systems were not very connected to internet.

Basic needs were to help **developers** and **end-users** be **successful** with software by giving **simple steps** for:

- Installing software (both interactive and silent) using media
- Licensing and activating software
- Building projects and installers, including automation



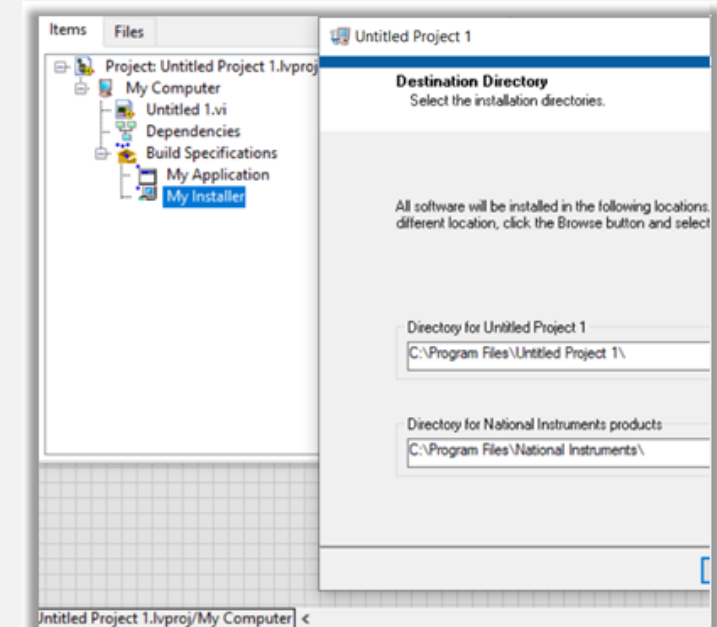
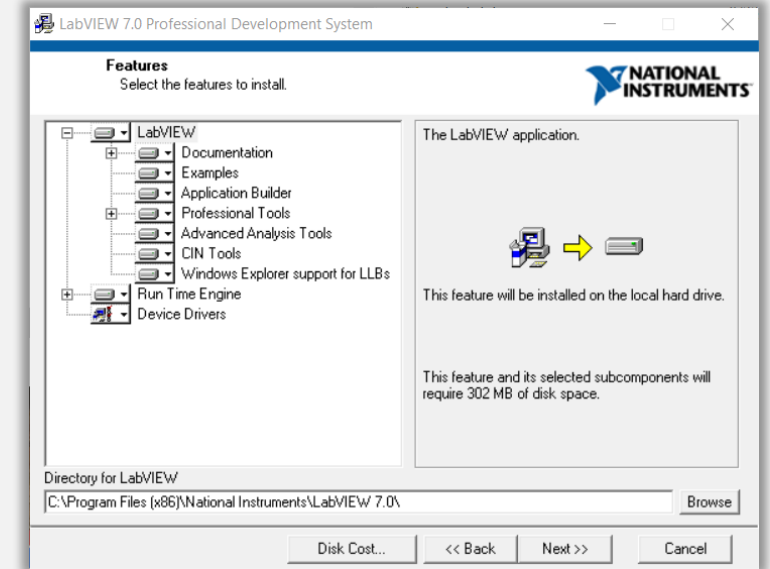
NI solution introduced “NI Installers”

In 2003, NI created an MSI-based installer framework

- Presented a consistent UX and reliability for installation operations
- Used for all NI software installers
- Supported componentization to allow user to select optional components to install
- Supported simple dependency relationships, displaying EULAs and activation

In 2005, NI customers needed to deploy custom and NI components via installers

- NI added support for building custom installers using the same base framework:
 - LabVIEW 8.0
 - Later added to CVI, TestStand, Measurement Studio
- Both NI and customers benefited from this commonality when NI introducing new features



Best Practices from 2003+ using NI Installers

Need: Installing software (both interactive and silent) using media

- Use NI built installers and suites to target the software and drivers you need
- Use installer CLI and “specification” file to automate and customize installation
- Use Batch Installer Builder and Volume License Installer tools to create custom “suite” installers

Need: Licensing and activating software

- Activate NI software inline with installation, using NILM GUI, or use NILM CLI to automate
- Use Volume License Manager to manage NI software licensing internally
- Use Third-Party Licensing and Activation Toolkit with LabVIEW to add licensing to your software

Need: Building projects and installers, including automation

- Use LabVIEW, LabWindows/CVI, TestStand, and Measurement Studio installer builders to create custom installer, and use builder CLI or API to automate builds
- You can add dependencies from your installer to required NI SW components

Fast-forward to present day needs...

Systems more connected than ever, end-users have higher expectations for efficient development and delivery of software updates, and end users are just savvier technically

Additional needs are to help **developers** and **end-users** be **successful** with:

- Installation of software delivered within open and closed network connected systems
- Sharing reuse components for development
- Managing project dependencies for development
- Automated build and test development processes
- Updating and managing remote systems and deployments
- Automated delivery of software updates



NI Platform Solutions for Distributing Your IP

NI Platform Solutions for Distributing IP

Use Cases and NI Platform Solutions	NI Package Management	JKI VIPM*	NI SystemLink	NI License Manager
Building and installing your applications including dependencies on end-user and target systems	✓			
Automate your installer build and test processes	✓			
Installing LabVIEW reuse libraries and tools into your LabVIEW projects		✓		
Updating and managing remote systems and deployments			✓	
Licensing and activation of your applications and reuse libraries				✓

** NI has had a partnership with JKI since 2010, and as a valued partner in handling dependency management in LabVIEW, NI provides VIPM Community Edition with LabVIEW.*



NI Package Management

Use to build packages and install **system-level software**

“Legacy” framework shortcomings...

By 2018, NI recognize the following missing needs:

- × Limited visibility and flexibility to manage installed components
- × Framework did not support web-based distribution
- × Installation could not download only required components
- × Limited support to upgrade and patch components
- × Limited dependency model between components
- × Ill-suited to support remote system management
- × No low-level access for custom workflows

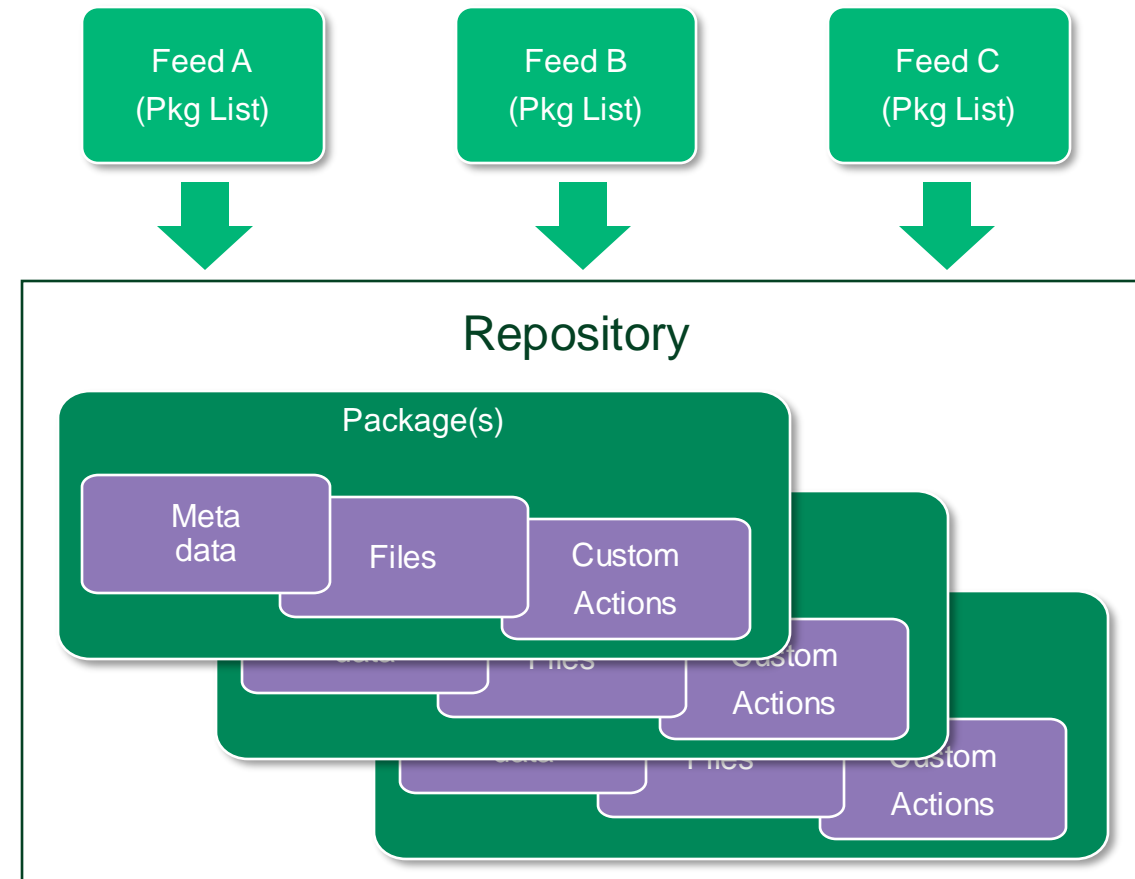


What is NI Package Management?

In 2019, NI released framework to package, distribute, and manage NI software on Windows

A package is an installable software component that represents a product, tool, driver, library, or just files

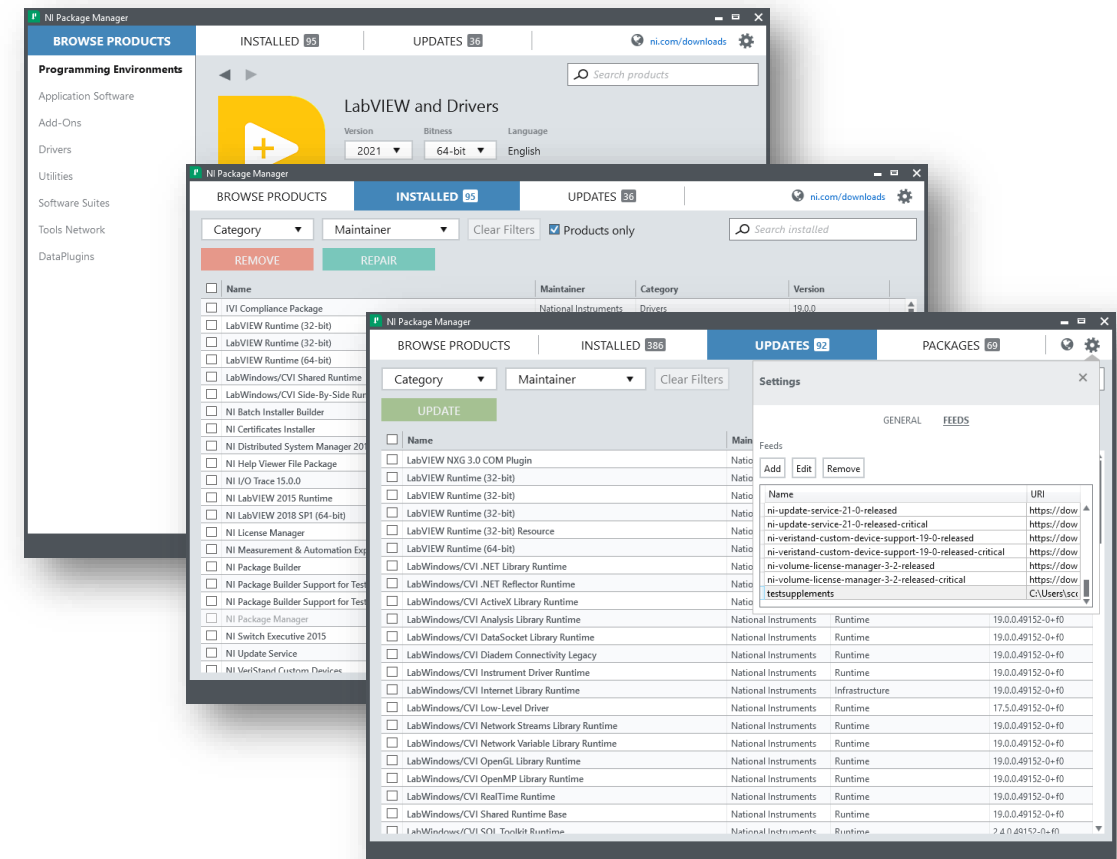
- Contains:
 - Files & directory information for installing
 - Metadata: name, description, version, etc.
 - Dependency information to other packages
- Packages are typically distributed as a feed (a list of packages) that points to a repository (or pool) of packages
- All NI software built and deployed using this framework, while still supporting legacy installers



What is NI Package Manager GUI?

Use GUI to install software delivered from network connected systems

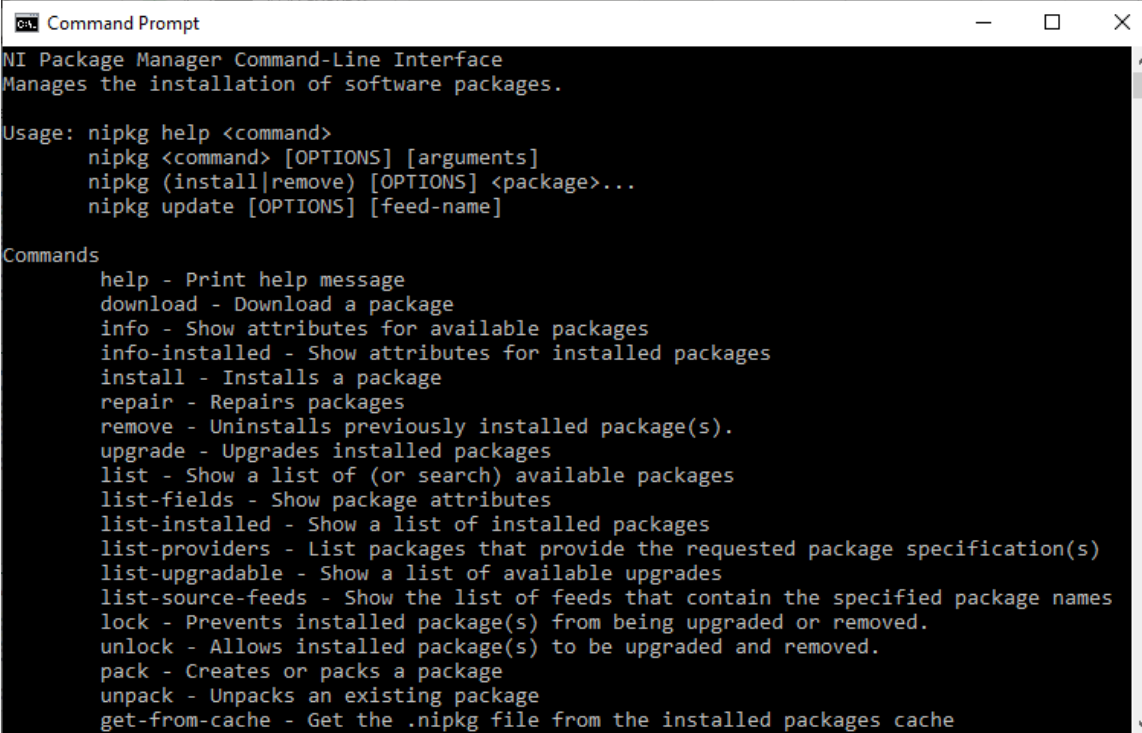
- Browse for software products from ni.com
- Lists installed packages and discover recommendations not yet installed
- List available updates (patches) to install
- Repair or remove installed packages
- Configuration settings
 - Flexible filtering of list of packages
 - Update registered feeds for system
 - Options to show more package information



What is NIPM CLI?

Use nipkg.exe to perform and automate low-level commands for managing packages

- Install, update, repair, and remove packages
- Update registered feeds for system
- List installed and available packages
- Download packages from registered feeds
- Pack (create) and unpack packages
- Create and edit feeds, and list feed contents
- Full help available, “nipkg help <command>”



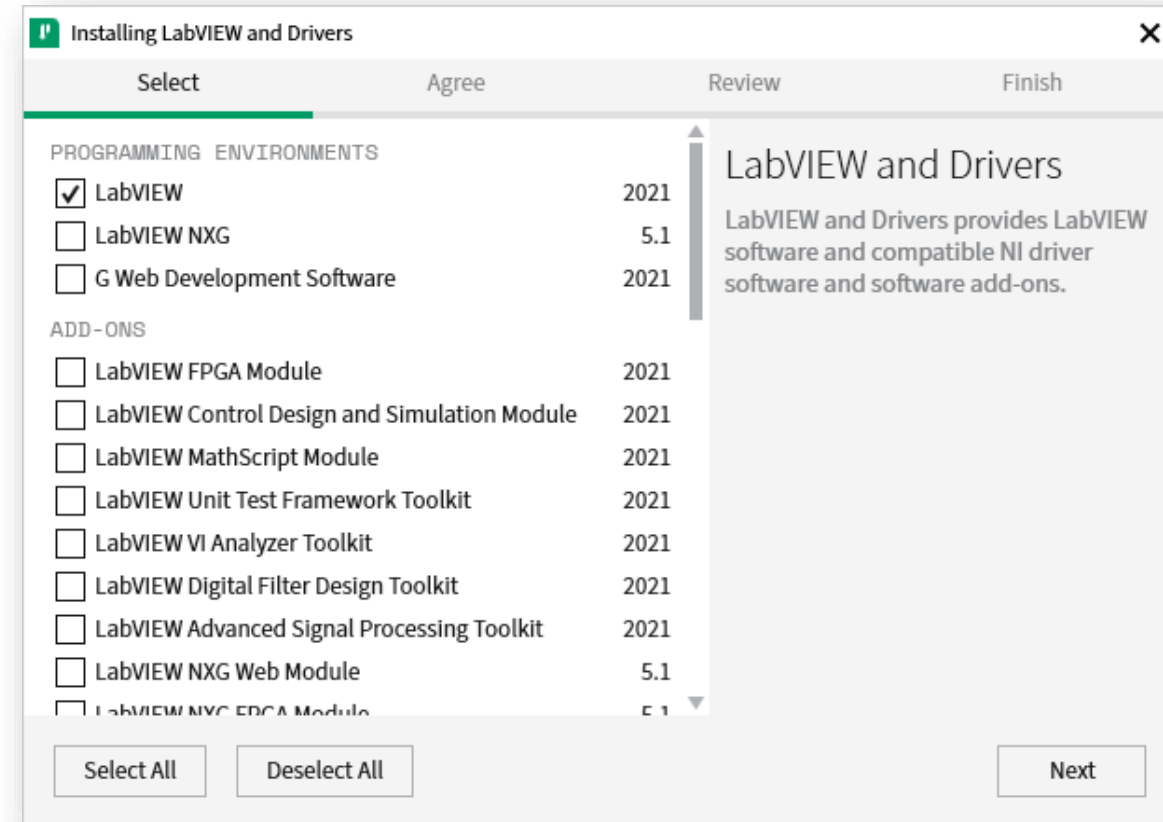
```
Command Prompt
NI Package Manager Command-Line Interface
Manages the installation of software packages.

Usage: nipkg help <command>
       nipkg <command> [OPTIONS] [arguments]
       nipkg (install|remove) [OPTIONS] <package>...
       nipkg update [OPTIONS] [feed-name]

Commands
  help - Print help message
  download - Download a package
  info - Show attributes for available packages
  info-installed - Show attributes for installed packages
  install - Installs a package
  repair - Repairs packages
  remove - Uninstalls previously installed package(s).
  upgrade - Upgrades installed packages
  list - Show a list of (or search) available packages
  list-fields - Show package attributes
  list-installed - Show a list of installed packages
  list-providers - List packages that provide the requested package specification(s)
  list-upgradable - Show a list of available upgrades
  list-source-feeds - Show the list of feeds that contain the specified package names
  lock - Prevents installed package(s) from being upgraded or removed.
  unlock - Allows installed package(s) to be upgraded and removed.
  pack - Creates or packs a package
  unpack - Unpacks an existing package
  get-from-cache - Get the .nipkg file from the installed packages cache
```

Packages might be cool, but are installers still supported?

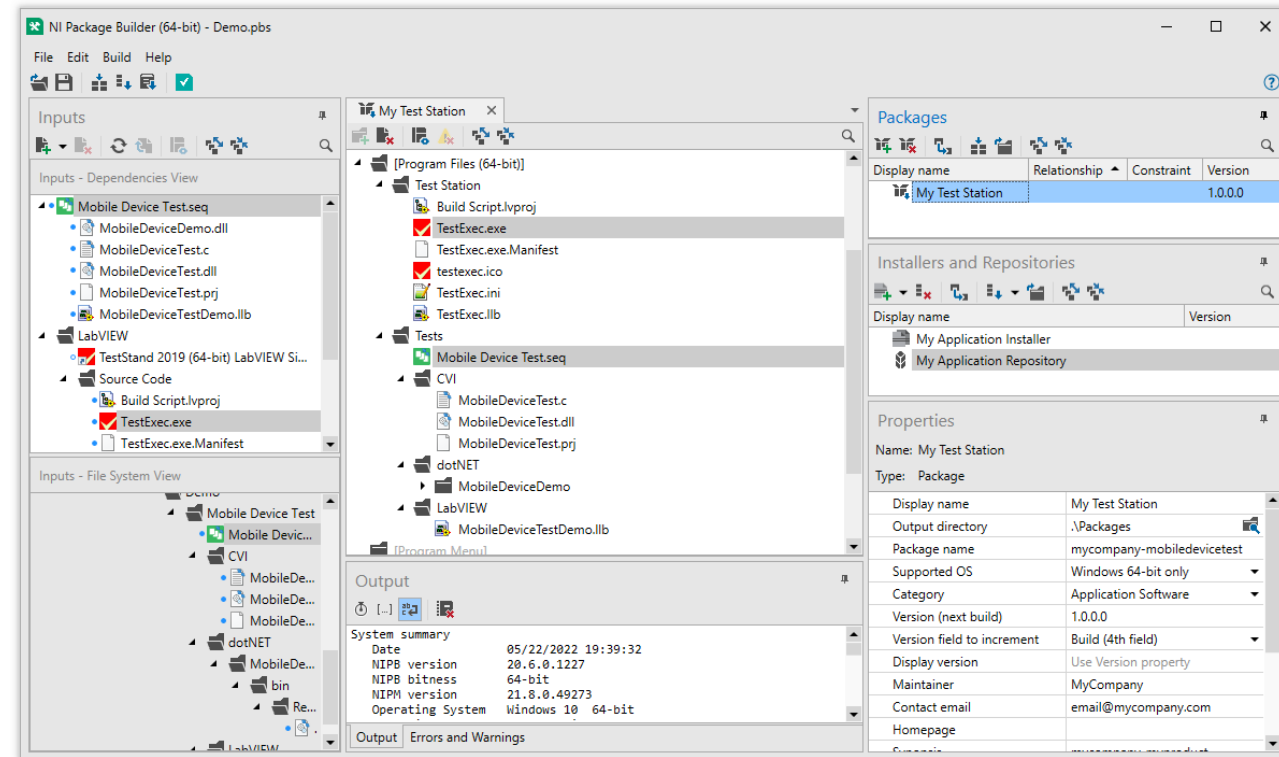
- **Yes:** NI Package Management includes capability to create a "Package Installer", which is a "wizard" like GUI that wraps packages and feeds.
- Support a one-click install EXE to setup the system, register feeds, and prompts for "top-level" packages to install and any additional recommendations.
 - Online installer is small and downloads files from ni.com
 - Offline installer contains all packages required for installing without a network
- Non-interactive & silent modes supported
- A "suite" installer displays multiple top-level packages



Package and Installer Build Tooling

Tools to manage project dependencies for development and automate build processes

- LabVIEW build specifications
 - Build package, feed, and/or package installer that contains LabVIEW applications/libraries and dependencies
 - Upload built packages to SystemLink feed
 - Use CLI/API to automate building spec
- TestStand Deployment Utility
 - Build a package, and feed or installer, that contains TestStand sequence files and dependencies
 - Use CLI to automate building a solution
- NI Package Builder
 - Builds multiple packages, feeds, and installers
 - Supports TestStand deployment workflows
 - Build “suite” installer from other installers
 - Simple CLI to automate building solutions



Best Practices using NI Package Management

Need: Installation of software delivered within open and closed network connected systems

- Use NI Package Manager GUI or nipkg CLI

Need: Automated build and test development processes

- Use nipkg CLI or any of the package and installer builder tools



JKI VI Package Manager

Find and install LabVIEW code reuse and tools.

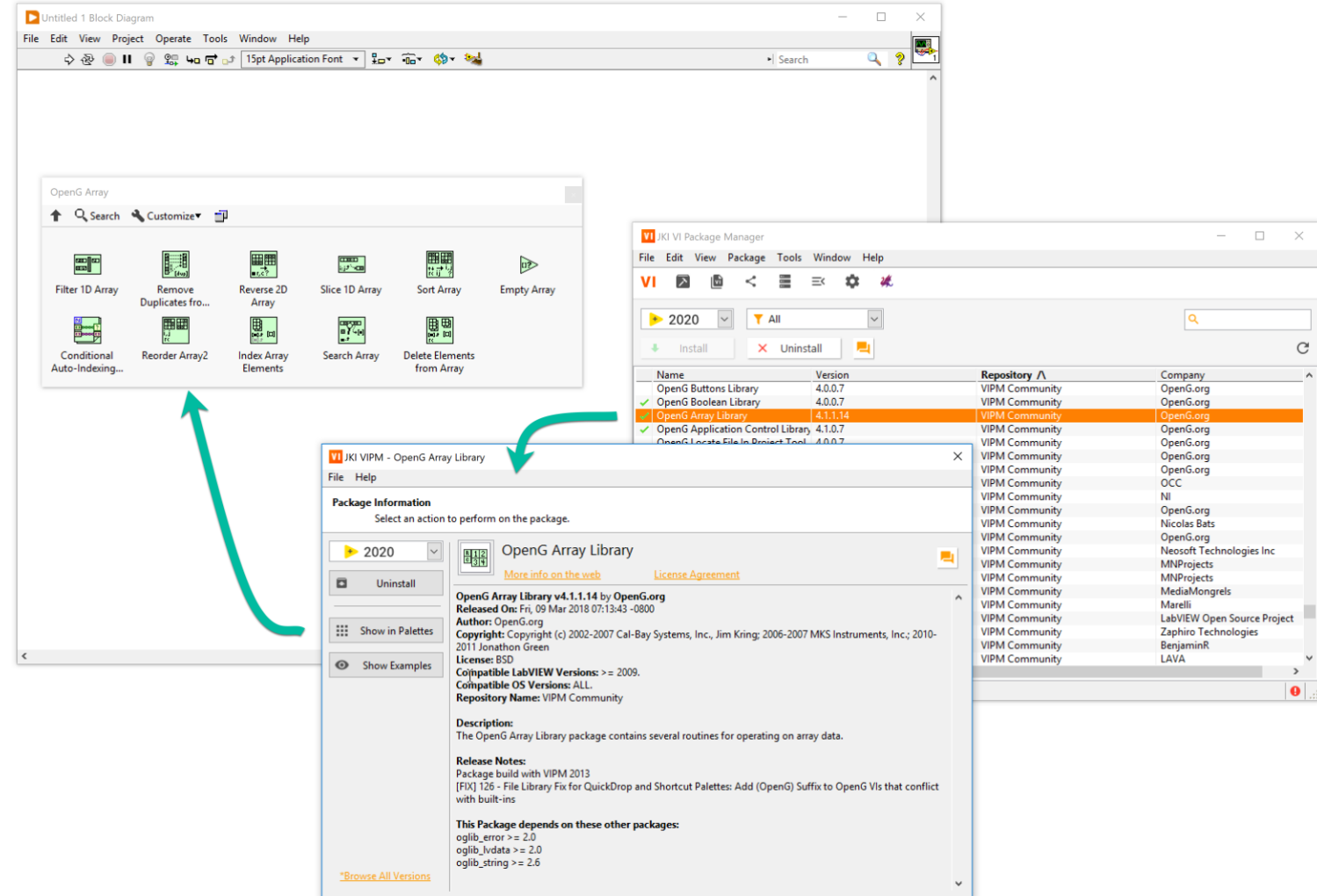
Manage your project's package configuration.

Build and share packages with the community and teams.

VIPM

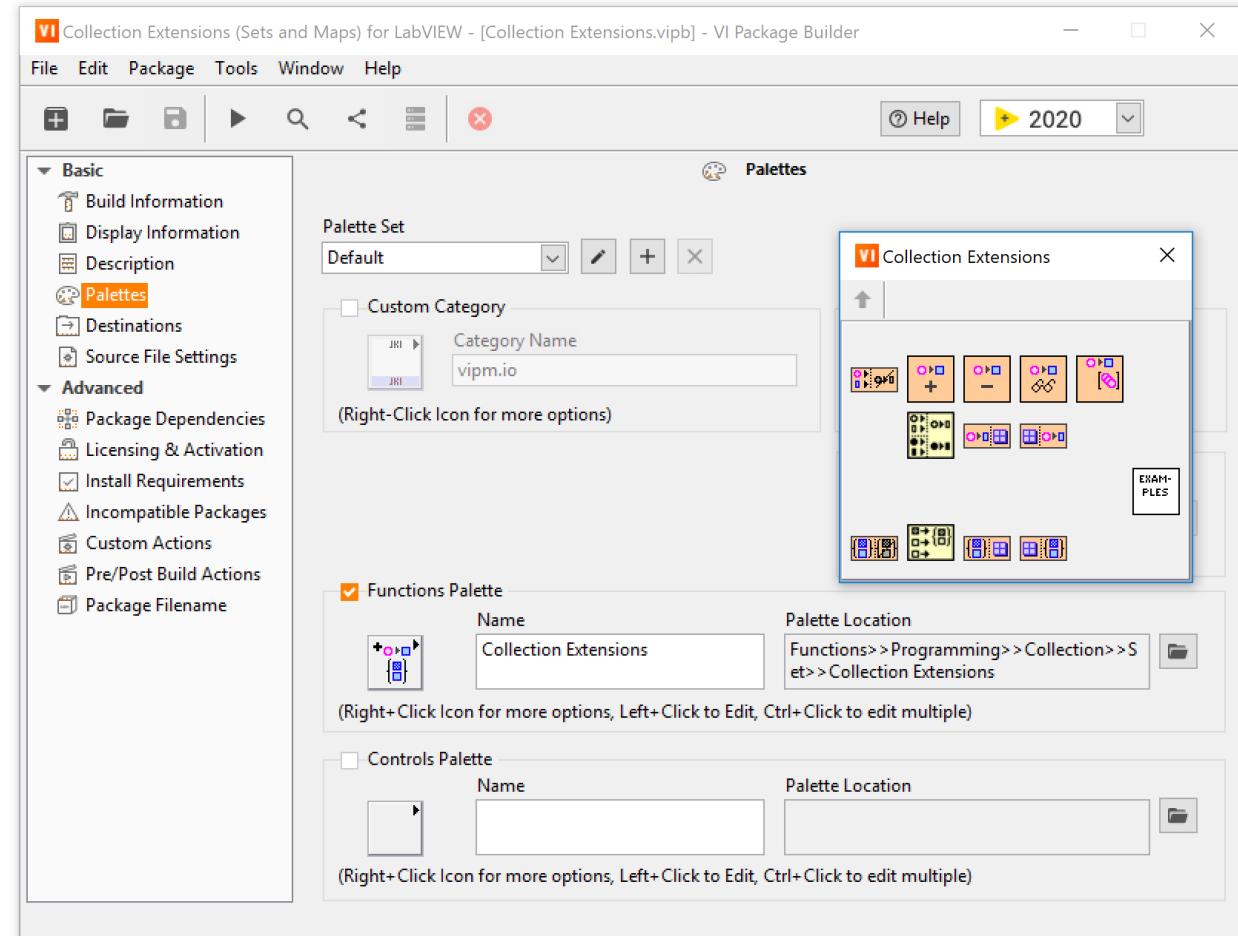
Product Summary

- Included with LabVIEW
- Search for LabVIEW tools and install them right into your palettes
- Build packages of your reusable code
- Manage configurations of packages used by your project (dependencies)
- Share packages privately with your team or publicly the LabVIEW community.



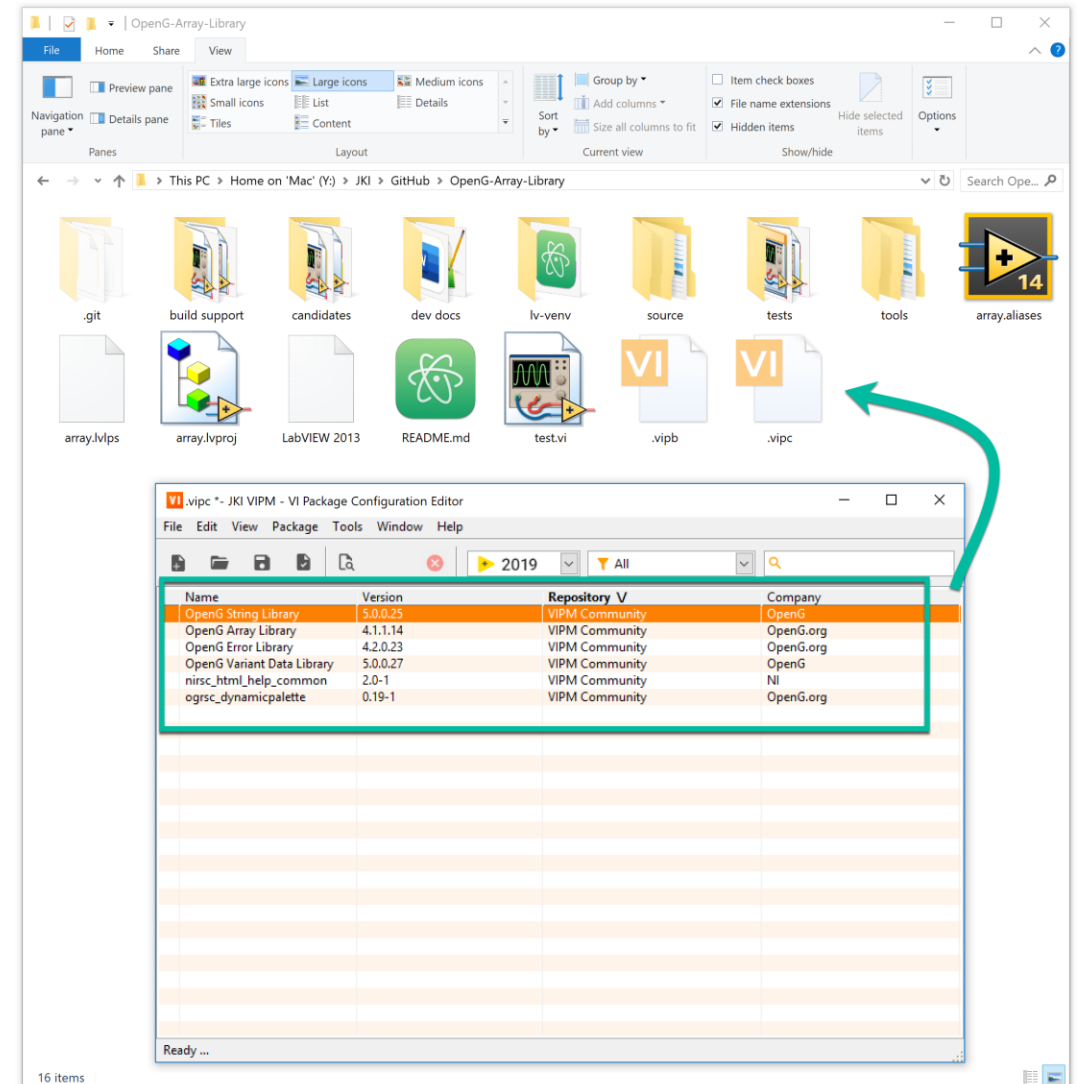
Building VI Packages

- Build packages of your reusable code
- Define package dependencies on other packages
- Add password protection or remove block diagrams during packaging process
- Add licensing and activation with NI's TPLAT toolkit



Managing Project Package Dependencies

- Manage project package configurations (using a VIPC file)
- Store copies of the packages inside the VIPC file (for off-line systems)
- Use in conjunction with source code control (like Git and SVN)
- Use API to automate package installation and builds (Continuous Integration)



Share Packages

Privately

Share packages with your team via a Shared Package Repository (VIPM Pro and Enterprise)

Publicly

Share packages with the community on vipm.io (include source code)

★ anyone can publish on vipm.io ★

The screenshot shows a web browser window with the URL `vipm.io/search/?q=modbus`. The page displays search results for "modbus" on the VIPM (Vendor Independent Package Manager) website. The results are listed in a table with columns for Package, Summary, Stars, and Installs. The packages are sorted by the number of installs in descending order.

Package	Summary	Stars	Installs
NI Modbus Library by NI	This Modbus library provides low level customizable communication over the Modbus standard.	☆ 26	17081
Predix Modbus TCP Connectivity for NI Linux RT by Ovak Technologies	Predix Modbus TCP Connectivity for NI Linux RT toolkit is designed to allow users send and store sensors data .	☆ 1	251
DCAF Modbus Module by NI		☆ 0	1345
Plasmionique Modbus Master by Plasmionique Inc	Open source Modbus master library for LabVIEW	☆ 6	3429
LB Drive by Lenord, Bauer & Co. GmbH	Control LB Drive systems over Modbus.	☆ 0	31
GModBus over Serial Line by SAPHIR	Implement ModBus protocol over serial line in your LabVIEW application	☆ 0	902
GModBus over TCP by SAPHIR	Implement ModBus protocol over TCP in your LabVIEW application	☆ 0	1060
KELLER RS485 Library by KELLER AG für Druckmesstechnik	Communication library for digital pressure transmitters, pressure gauges and converters.	☆ 0	500
Instrument Control Module by Original Code Consulting	Instrument Communication Module for LabVIEW ICM is an open-source LabVIEW toolkit for acquiring and writing to "slow" data, such as that coming from instrument...	☆ 1	104

Page 1 of 1



SystemLinkTM

Use to manage distributed systems and much more...

SystemLink

Managing software and data for test & measurement

Capabilities include:

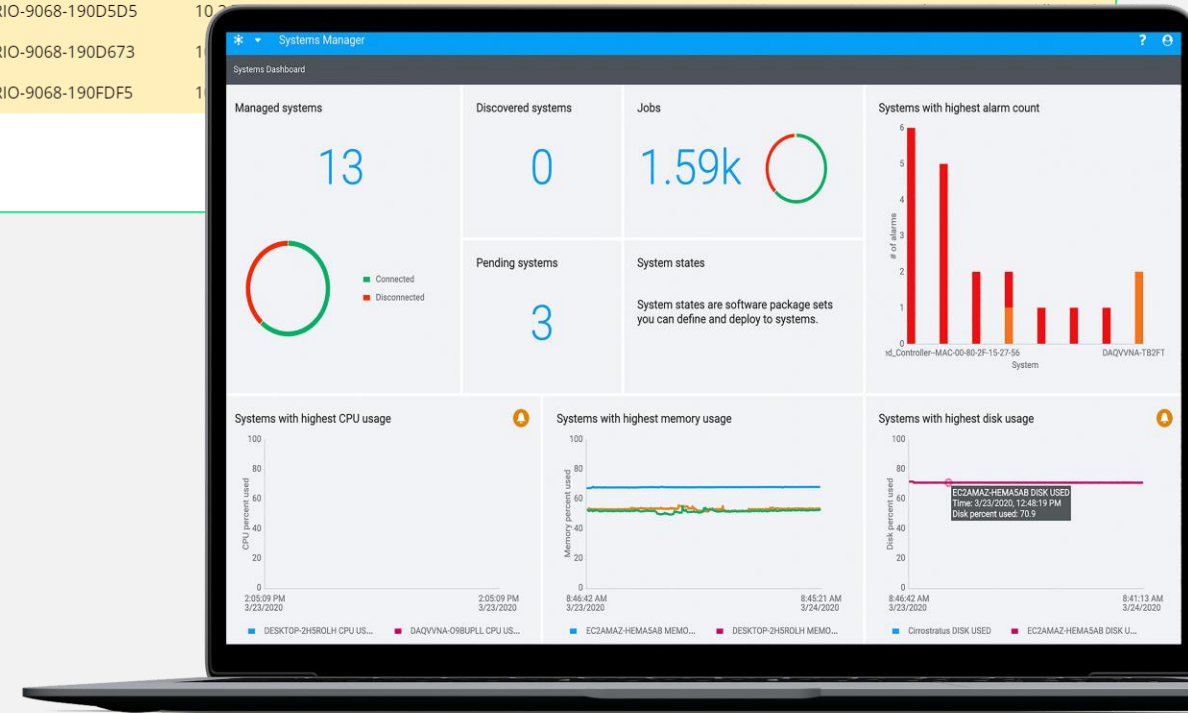
- Remote Systems Management
- Test Asset Management
- Test Monitoring & Results Management
- Measurement Data Management

Systems Manager

Dashboard > Managed Systems >

Groups History Software Restart More 6 of 6 systems Filter

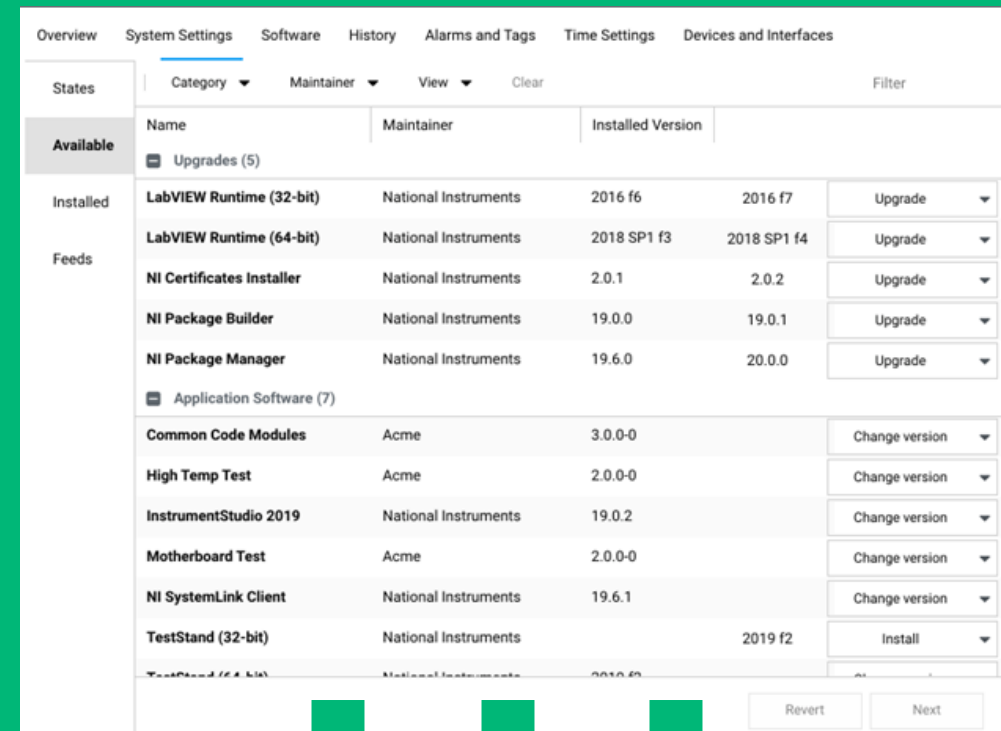
<input type="checkbox"/>	Name	IP Address	Model Name	Operating System	Serial Number	Connection	Comments
<input type="checkbox"/>	Automated Test Systems (2)						
<input type="checkbox"/>	PXIe-8840Quad-1	10.2.74.79	NI PXIe-8840 Quad-Core	Windows 7	030E1626	Connected	Test Station 1
<input type="checkbox"/>	PXIe-8840Quad-2	10.2.74.80	NI PXIe-8840 Quad-Core	Windows 7	030DDB85	Connected	Test Station 2
<input checked="" type="checkbox"/>	Control Systems (4)						
<input checked="" type="checkbox"/>	NI-cRIO-9068-190CB7B	10.2.74.64	cRIO-9068	NI Linux RT 4.1	190CB7B	Connected	Test Cell 1
<input checked="" type="checkbox"/>	NI-cRIO-9068-190D5D5	10.2.74.65	cRIO-9068	NI Linux RT 4.1	190D5D5	Connected	Test Cell 1
<input checked="" type="checkbox"/>	NI-cRIO-9068-190D673	10.2.74.66	cRIO-9068	NI Linux RT 4.1	190D673	Connected	Test Cell 1
<input checked="" type="checkbox"/>	NI-cRIO-9068-190FDF5	10.2.74.67	cRIO-9068	NI Linux RT 4.1	190FDF5	Connected	Test Cell 1



Remote System Management

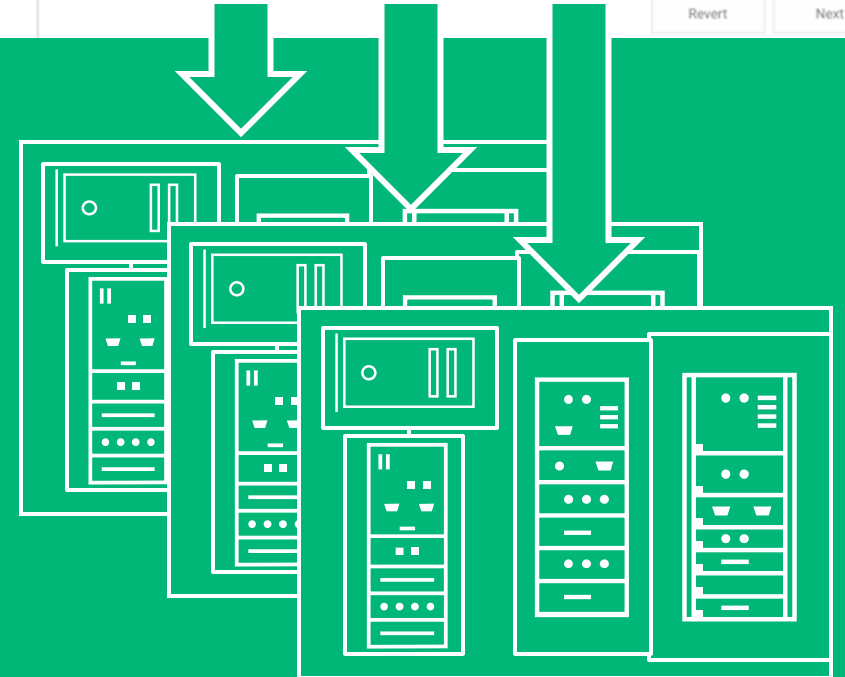
Discover, and efficiently manage and configure the software on your distributed systems using the following capabilities

- Manage which client systems can connect to server
- Upload packages and configure feeds that client systems can use
- Execute remote software installs, upgrades, and downgrades on systems using NI Package Manager as a service
- Capture, compare, and edit system “states” that represent software collections, and deploy to systems
- Deploy changes to multiple systems simultaneously using “jobs”
- Manage client system availability for updates using a lock/unlock feature



The screenshot shows the NI Package Manager web interface. At the top, there are tabs for Overview, System Settings, Software, History, Alarms and Tags, Time Settings, and Devices and Interfaces. Below these is a filter section with buttons for States, Category, Maintainer, View, and Clear, along with a Filter button. The main content area displays a table of software feeds. The table has columns for Name, Maintainer, and Installed Version. The feeds are categorized by state: Available, Installed, and Feeds. The Available section shows a list of feeds with their names, maintainers, and installed versions, along with an Upgrade button. The Installed section shows a list of feeds with their names, maintainers, and installed versions, along with an Upgrade button. The Feeds section shows a list of feeds with their names, maintainers, and installed versions, along with an Upgrade button. The table also includes a 'Revert' button and a 'Next' button at the bottom right.

States	Category	Maintainer	View	Clear	Filter
Available					
Installed					
Feeds					
	Name	Maintainer	Installed Version		
	Upgrades (5)				
	LabVIEW Runtime (32-bit)	National Instruments	2016 f6	2016 f7	Upgrade
	LabVIEW Runtime (64-bit)	National Instruments	2018 SP1 f3	2018 SP1 f4	Upgrade
	NI Certificates Installer	National Instruments	2.0.1	2.0.2	Upgrade
	NI Package Builder	National Instruments	19.0.0	19.0.1	Upgrade
	NI Package Manager	National Instruments	19.6.0	20.0.0	Upgrade
	Application Software (7)				
	Common Code Modules	Acme	3.0.0-0		Change version
	High Temp Test	Acme	2.0.0-0		Change version
	InstrumentStudio 2019	National Instruments	19.0.2		Change version
	Motherboard Test	Acme	2.0.0-0		Change version
	NI SystemLink Client	National Instruments	19.6.1		Change version
	TestStand (32-bit)	National Instruments		2019 f2	Install
	TestStand (64-bit)	National Instruments	2019 f2		Install



Best Practices using SystemLink

Need: Updating and managing remote systems and deployments

- Configure and install software on a connected remote system, and/or force systems to adhere to known software states

Need: Automated delivery of software updates (continuous delivery)

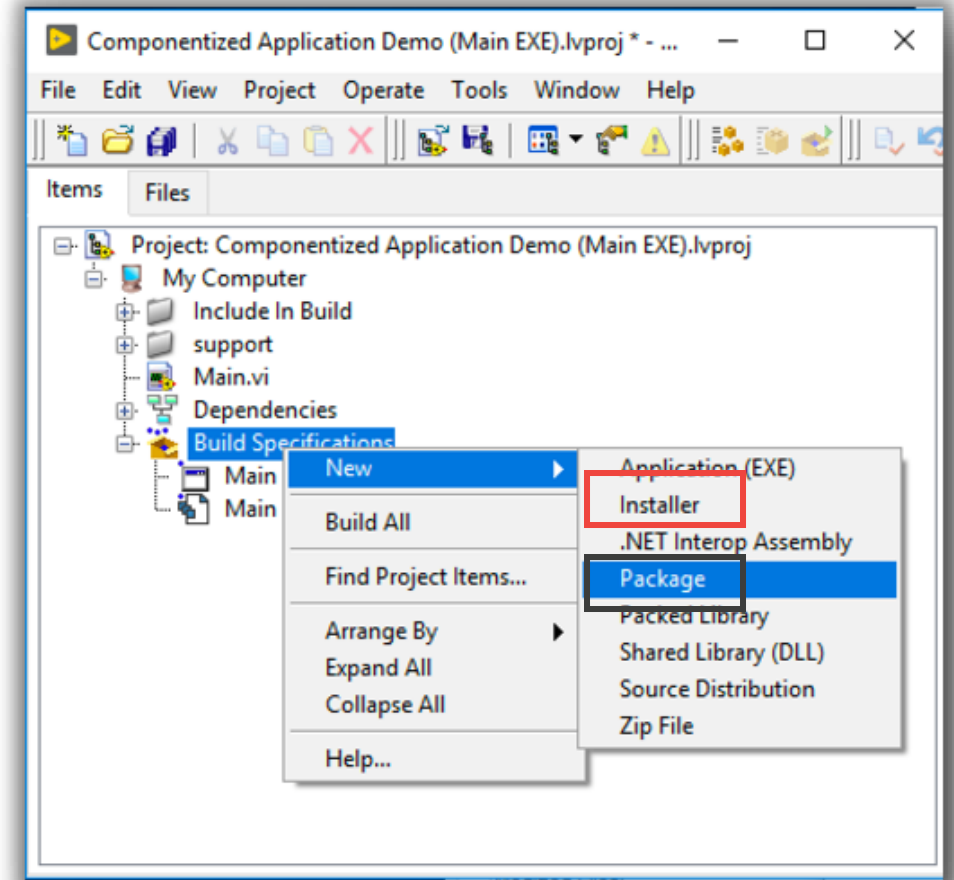
- Use the SystemLink APIs to perform operations, such as,
 - upload new packages to the server
 - create/update available feeds with new packages
 - access client system information
 - initiate jobs to install packages and push states to client systems

What's next?

Roadmap and exploring your needs

Package Management is replacing “legacy” installers

- As of 2019, new NI software distributed using packages
 - LabVIEW and TestStand support both legacy and package technology
- NI is planning to remove legacy support from LabVIEW and TestStand
- Package technology offers similar capabilities to legacy installers, but we know there are some gaps
- NI is gathering feedback about which capabilities are essential for your workflows
 - Consider using NI package technology for new projects
 - Consider prototyping port of existing projects to identify gaps and challenges





Customer Feedback



Installer Builders in LabVIEW Application Builder

This survey asks questions to better understand how you use NI tools (i.e. LabVIEW, LabVIEW Application Builder and NIPM) to package and distribute your applications.

If interested in participating in a similar TestStand survey, please leave us your name and email.

Linux Exploration

One other area we are currently exploring for package management, is improved support for Linux. NI currently supports installation of certain NI software (mainly drivers and LabVIEW) on common desktop Linux distributions, but the installation is not intuitive.

We are currently exploring ways to improve the installation process. A subsequent step in this improved support will be exploring deployment support.



Offline Installation of NI Linux Device Drivers

AE KB

Updated Apr 12, 2022



Environment ⓘ

Driver

- NI Linux Device Drivers

Operating System

- Linux

Questions and Discussion