

RELEASE NOTES

PACE Packager Hub 1.0.0



Table of Contents

1	PACE Packager Hub Overview	3
2	Requirements	5
2.1	Server-side Requirements	5
2.2	Client-side Requirements	6
2.2.1	Supported browsers	6
2.3	Desktop Client Requirements	6
3	Release Notes	7
3.1	About This Release	7
3.2	Track orders, statuses, assignments	8
3.3	Create new order	9
3.4	Move order through its workflow	10
3.5	View/change order attributes (fields)	12
3.6	Communicate about order (messages)	14
3.7	Upload/download order artifacts (files)	15
3.8	Manage order links (references)	16
3.9	Review, find packages	17
3.10	Create new package	18
3.11	View/change package attributes (fields)	19
3.12	Manage package content (files)	20
3.13	Use automation for packaging	21
3.14	Deliver package to Customer	22
3.15	View/download delivered package	23

1 PACE Packager Hub Overview

PACE Packager Hub is an end-to-end solution that is fully dedicated to customer servicing, teamwork, and management of packaging tasks in teams of any size. It is designed to significantly ease the fulfillment of packaging quality standards and increase the productivity of each member of the packaging team while providing an efficient tool for the management of the teams and the application packaging lifecycle. It operates as a joining link between customers, supervisors, packaging engineers, and their tools.

The solution is already pre-configured for the packaging processes out of the box, provides the best practices approaches, and has extensive customization options.

PACE Packager Hub consists of the **Task Manager** (Workflow) and **Repository Manager** modules.

Highlights:

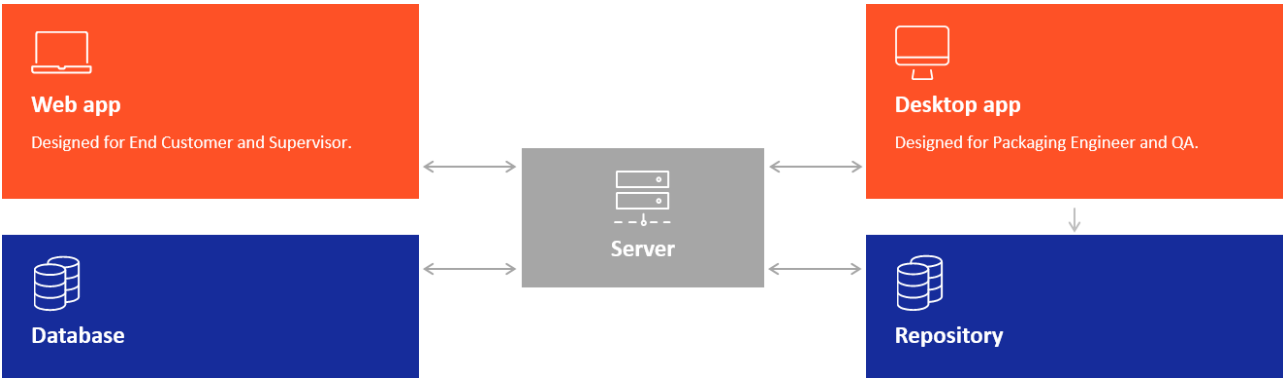
- A central store for all orders for packaging and packages themselves
- Provides statuses of all packaging orders/tasks
- Automates the routine operations and stores the results
- Controls the assignments and completion of orders/tasks
- Balances the workload of team members
- Manages permissions based on the assigned roles
- Highly customizable in adding/configuring fields, workflows, and scripts
- Manages packages and files on various repositories
- Supports direct upload/download of files of any sizes
- Controls the integrity of files
- Supports multiple isolated projects

PACE Packager Hub solution consists of the on-premise Server that has the Web interface and Desktop app client.

Release Notes

PACE Packager Hub 1.0.0

On-premise Location



2 Requirements

2.1 Server-side Requirements

OS [Windows or Windows Server OS that supports ASP.NET Core Runtime 3.1.10](#) or higher.

Manually install and configure the following:

- [IIS Web Server](#)
(for production purposes only and if built-in [Kestrel](#) is not suitable)
- [Microsoft SQL Server 2016](#) or higher
(for production purposes only)
- [ASP.NET Core Runtime 3.1.10](#) or higher
(for production purposes only; as post-requisite if IIS Web Server is used)

NOTE: Please ensure that you are downloading and installing the **Hosting Bundle**.

Middleware

Release information	Build apps - SDK	Run apps - Runtime																																	
v3.1.11 Security patch Release notes Released 2021-01-12	<p>ⓘ This release contains multiple SDKs. If you're using Visual Studio, look for the SDK that supports the version you're using. If you're not using Visual Studio, install the first SDK listed.</p> <p>SDK 3.1.405</p> <p>Visual Studio support Visual Studio 2019 (v16.7) Visual Studio 2019 for Mac (v8.8)</p> <p>Included in Visual Studio 16.7.10</p> <p>Included runtimes .NET Core Runtime 3.1.11 ASP.NET Core Runtime 3.1.11 .NET Core Desktop Runtime 3.1.11</p> <p>Language support C# 8.0 F# 4.7 Visual Basic 15.9</p> <table> <tr> <th>OS</th><th>Installers</th><th>Binaries</th></tr> <tr> <td>Linux</td><td>Package manager instructions</td><td>Arm32 Arm64 RHEL 6 x64 x64 x64 Alpine</td></tr> <tr> <td>macOS</td><td>x64</td><td>x64</td></tr> <tr> <td>Windows</td><td>x64 x86</td><td>Arm32 x64 x86</td></tr> <tr> <td>All</td><td>dotnet-install scripts</td><td></td></tr> </table>	OS	Installers	Binaries	Linux	Package manager instructions	Arm32 Arm64 RHEL 6 x64 x64 x64 Alpine	macOS	x64	x64	Windows	x64 x86	Arm32 x64 x86	All	dotnet-install scripts		<p>ASP.NET Core Runtime 3.1.11</p> <p>The ASP.NET Core Runtime enables you to run existing web/server applications. On Windows, we recommended installing the Hosting Bundle, which includes the .NET Core Runtime and IIS support.</p> <p>IIS runtime support (ASP.NET Core Module v2) 13.1.20325.11</p> <table> <tr> <th>OS</th><th>Installers</th><th>Binaries</th></tr> <tr> <td>Linux</td><td>Package manager instructions</td><td>Arm32 Arm64 Arm64 Alpine x64 x64 Alpine</td></tr> <tr> <td>macOS</td><td></td><td>x64</td></tr> <tr> <td>Windows</td><td>Hosting Bundle x64 x86</td><td>Arm32 x64 x86</td></tr> </table> <p>.NET Core Desktop Runtime 3.1.11</p> <p>The .NET Core Desktop Runtime enables you to run existing Windows desktop applications. This release includes the .NET Core Runtime, you do not need to install it separately.</p> <table> <tr> <th>OS</th><th>Installers</th><th>Binaries</th></tr> <tr> <td>Windows</td><td>x64 x86</td><td></td></tr> </table> <p>.NET Core Runtime 3.1.11</p> <p>The .NET Core Runtime contains just the components needed to run a console app. Typically you'd also install either the ASP.NET Core Runtime or</p>	OS	Installers	Binaries	Linux	Package manager instructions	Arm32 Arm64 Arm64 Alpine x64 x64 Alpine	macOS		x64	Windows	Hosting Bundle x64 x86	Arm32 x64 x86	OS	Installers	Binaries	Windows	x64 x86	
OS	Installers	Binaries																																	
Linux	Package manager instructions	Arm32 Arm64 RHEL 6 x64 x64 x64 Alpine																																	
macOS	x64	x64																																	
Windows	x64 x86	Arm32 x64 x86																																	
All	dotnet-install scripts																																		
OS	Installers	Binaries																																	
Linux	Package manager instructions	Arm32 Arm64 Arm64 Alpine x64 x64 Alpine																																	
macOS		x64																																	
Windows	Hosting Bundle x64 x86	Arm32 x64 x86																																	
OS	Installers	Binaries																																	
Windows	x64 x86																																		

Privileges Administrative privileges on the system.

Hardware CPU: 64-bit (x64) processor at 2.4 GHz or greater
RAM: 4 GB

Release Notes

PACE Packager Hub 1.0.0

HDD: 1 GB free space

Network: 100 Mbit

2.2 Client-side Requirements

2.2.1 Supported browsers

Desktop	<ul style="list-style-type: none">• (Recommended) Google Chrome (Desktop) v. 87.0.4280.141 (https://chromeenterprise.google/browser/download/) or higher stable release.• Latest stable version of Web Browsers based on Chromium. <p>NOTE: Please ensure that JS support is enabled.</p>
---------	--

Display	1280 x 800 resolution or higher
---------	---------------------------------

2.3 Desktop Client Requirements

OS	Windows or Windows Server OS that supports .NET 4.7.2 or higher.
Middleware	Microsoft .NET Framework 4.7.2 or higher.
Privileges	Regular user privileges on the system.
Hardware	CPU: 32-bit (x86) or 64-bit (x64) processor at 2 GHz or greater RAM: 2 GB HDD: 100 MB free space Display: 1280 x 800 resolution or higher. Network: 100 Mbit

3 Release Notes

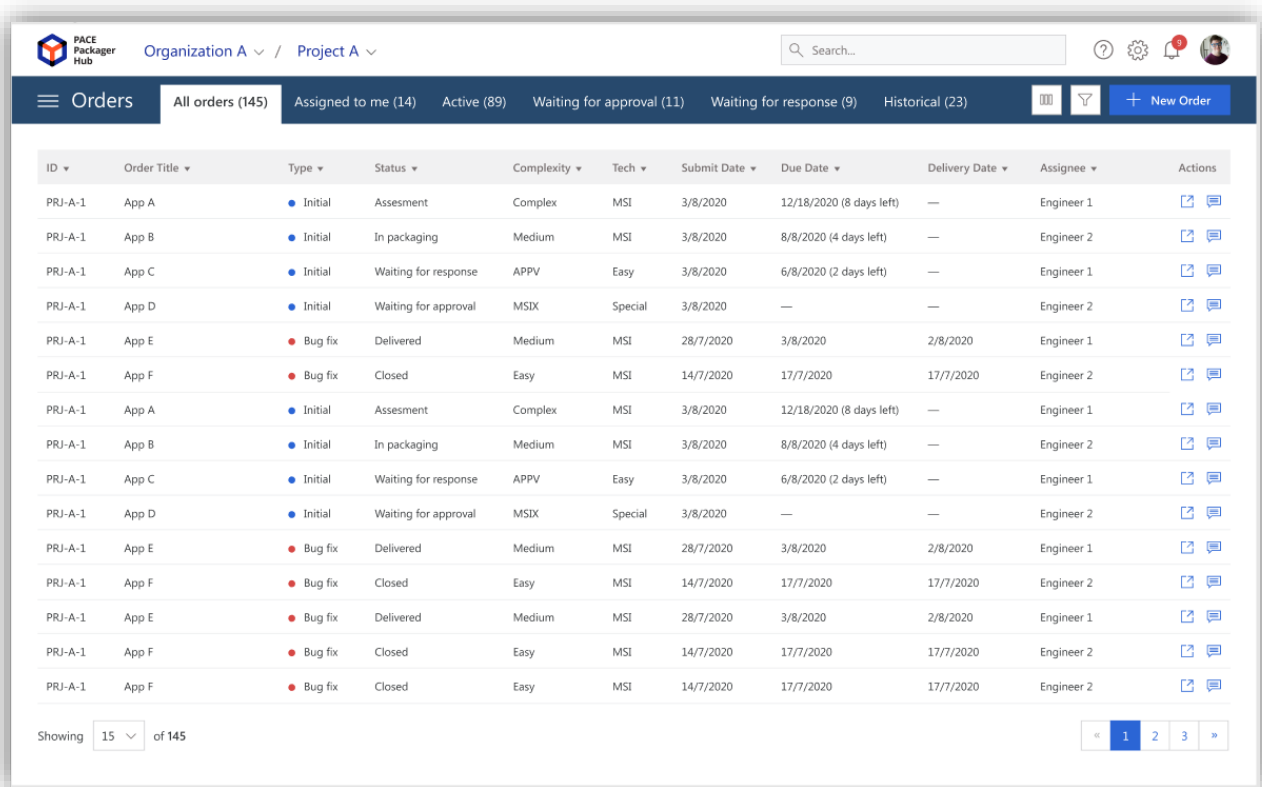
3.1 About This Release

This is the initial version of the **PACE Packager Hub** solution. In this version, we focused on the core functionality that is required to start and complete the application packaging process. We have considered the process that begins with the submission of a new packaging order by the customer and completes with the download of the delivered output package and includes all that is between these two points.

We are very grateful to everyone who has made a significant contribution to this product by responding to surveys and providing feedbacks. The information provided became the basis of the product roadmap.

Share your further feedbacks and suggestions [with us](#) and get the product sharpened for your needs.

3.2 Track orders, statuses, assignments



Organization A / Project A

Search...

Orders | All orders (145) | Assigned to me (14) | Active (89) | Waiting for approval (11) | Waiting for response (9) | Historical (23) | + New Order

ID	Order Title	Type	Status	Complexity	Tech	Submit Date	Due Date	Delivery Date	Assignee	Actions
PRJ-A-1	App A	Initial	Assesment	Complex	MSI	3/8/2020	12/18/2020 (8 days left)	—	Engineer 1	View Comment
PRJ-A-1	App B	Initial	In packaging	Medium	MSI	3/8/2020	8/8/2020 (4 days left)	—	Engineer 2	View Comment
PRJ-A-1	App C	Initial	Waiting for response	APPV	Easy	3/8/2020	6/8/2020 (2 days left)	—	Engineer 1	View Comment
PRJ-A-1	App D	Initial	Waiting for approval	MSIX	Special	3/8/2020	—	—	Engineer 2	View Comment
PRJ-A-1	App E	Bug fix	Delivered	Medium	MSI	28/7/2020	3/8/2020	2/8/2020	Engineer 1	View Comment
PRJ-A-1	App F	Bug fix	Closed	Easy	MSI	14/7/2020	17/7/2020	17/7/2020	Engineer 2	View Comment
PRJ-A-1	App A	Initial	Assesment	Complex	MSI	3/8/2020	12/18/2020 (8 days left)	—	Engineer 1	View Comment
PRJ-A-1	App B	Initial	In packaging	Medium	MSI	3/8/2020	8/8/2020 (4 days left)	—	Engineer 2	View Comment
PRJ-A-1	App C	Initial	Waiting for response	APPV	Easy	3/8/2020	6/8/2020 (2 days left)	—	Engineer 1	View Comment
PRJ-A-1	App D	Initial	Waiting for approval	MSIX	Special	3/8/2020	—	—	Engineer 2	View Comment
PRJ-A-1	App E	Bug fix	Delivered	Medium	MSI	28/7/2020	3/8/2020	2/8/2020	Engineer 1	View Comment
PRJ-A-1	App F	Bug fix	Closed	Easy	MSI	14/7/2020	17/7/2020	17/7/2020	Engineer 2	View Comment
PRJ-A-1	App E	Bug fix	Delivered	Medium	MSI	28/7/2020	3/8/2020	2/8/2020	Engineer 1	View Comment
PRJ-A-1	App F	Bug fix	Closed	Easy	MSI	14/7/2020	17/7/2020	17/7/2020	Engineer 2	View Comment
PRJ-A-1	App F	Bug fix	Closed	Easy	MSI	14/7/2020	17/7/2020	17/7/2020	Engineer 2	View Comment

Showing 15 of 145

« 1 2 3 »

Review orders and their details in the intelligible and configurable list. The list displays **orders of the selected project or multiple projects**. The user access is configured separately for each project.

Every project can contain a **unique list of Order Types**, where type is a packaging task (e.g., app packaging; bug fix; change request; support request; etc.). An order type can have a **unique workflow** (statuses, transitions between them, and permissions to invoke); and a **unique set of order attributes (fields)**. The system has (1) pre-defined and (2) user-defined attributes. The first group of attributes can only be adjusted, and user-defined attributes can be completely arbitrary for each type of order and project.

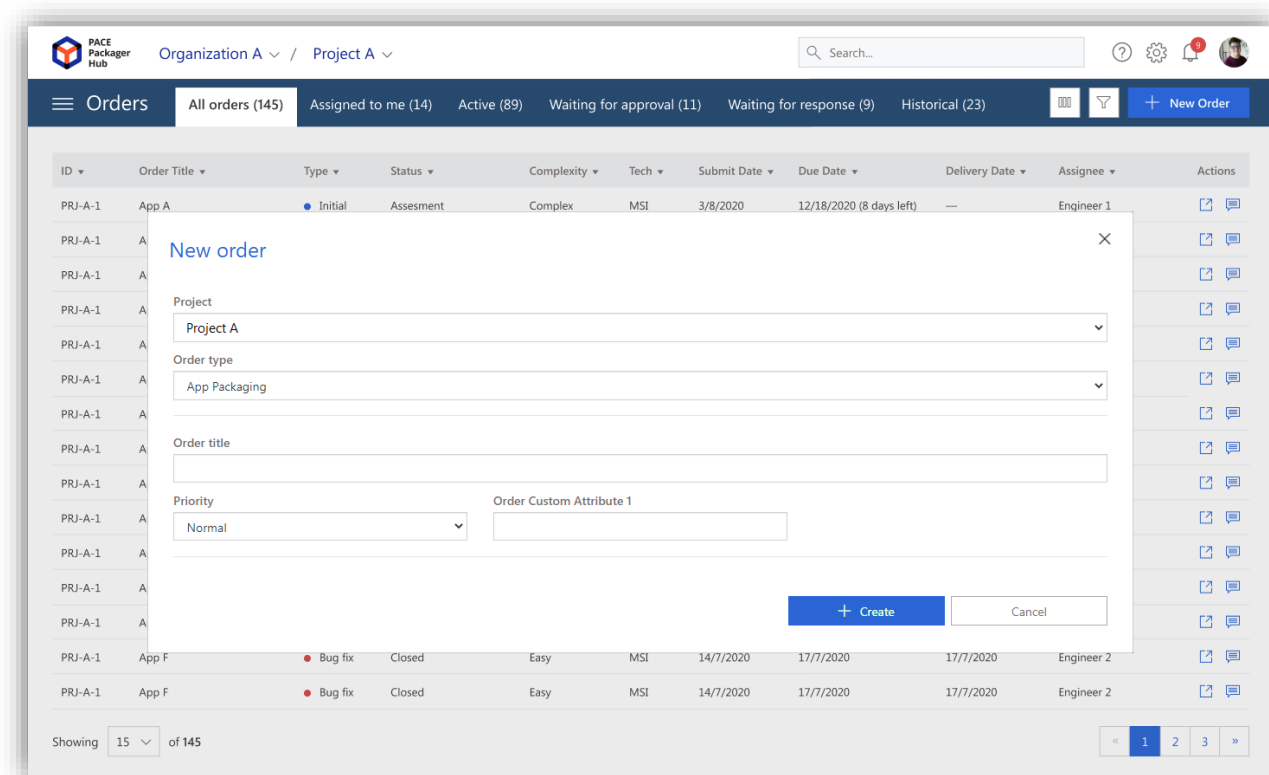
In addition to the initial 'All orders' view, other tabs allow to **quickly find orders assigned to the current user**; active orders of the project(s) that are not completed; and historical orders that are already completed. The order completion state is defined in the workflow settings for every order type and project. Also, **custom filters** will allow to find the needed order in any of these lists.

Release Notes

PACE Packager Hub 1.0.0

The default list **appearance can be customized** for each user (order details to display; columns width and sequence; data sorting). By default, the list displays the following: *Order ID, Title, Order Type, Status, Complexity Level, Priority, Creation Date, Due Date, Completion Date, Assignee*.

3.3 Create new order



Create a new order for packaging if you have the **Customer or Supervisor** role. The system strictly controls permissions depending on the user's role and access to a project.

During order creation, the user should **select a type of this order** (e.g., app packaging; bug fix; change request; support request; etc.). The type defines the workflow of the order (statuses and transitions between them) and the set of order attributes (fields). Available order types, their child-parent hierarchy, and other settings are configured in the project settings. In accordance with the order types hierarchy, some order types can be created from scratch, while others can only be created in response to other existing orders (e.g., 'bug fix' can only be created to the

Release Notes

PACE Packager Hub 1.0.0

initial 'app packaging' order). Most of the project settings can be saved as templates and reused in other projects.

Moreover, to create a new order the user must **specify the required order attributes (fields)** defined in the project settings for the selected order type.

This is only a small part of how the system can be modified to meet the needs of each project.

3.4 Move order through its workflow

The screenshot displays the PACE Packager Hub interface for 'Organization A' and 'Project A'. The main header shows the organization and project names, along with a search bar and user profile. The 'Order #3' page is divided into 'Order' and 'Package' tabs. The 'Order' tab is active, showing a table with columns for Order ID, Type, and Status. The 'Order ID' is #3, 'Type' is Initial order, and 'Status' is In packaging. A 'Primary transition' dropdown menu is highlighted with a red box. Below the table, there are sections for 'General', 'Messages', 'Artifacts', 'History', and 'Links'. The 'General' section contains fields for Order Title, App Name, App Version, App Vendor, Complexity, Due Date, Packaging Technology, Packager, QA Engineer, Assignee, and Description. A 'Save' button is at the bottom left.

Each **order has a workflow**. The workflow is a set of statuses and transitions (actions) between them. A **status can have multiple transitions**, and only one of them can be set as the primary (default) transition. A transition has **permissions that define who can run it**.

A **workflow depends on the order type** and may **differ between various types of orders** and projects. A workflow can be **saved as a template and reused** in other order types of the same or other projects.

The user sees only the transitions available to him/her depending on the role.

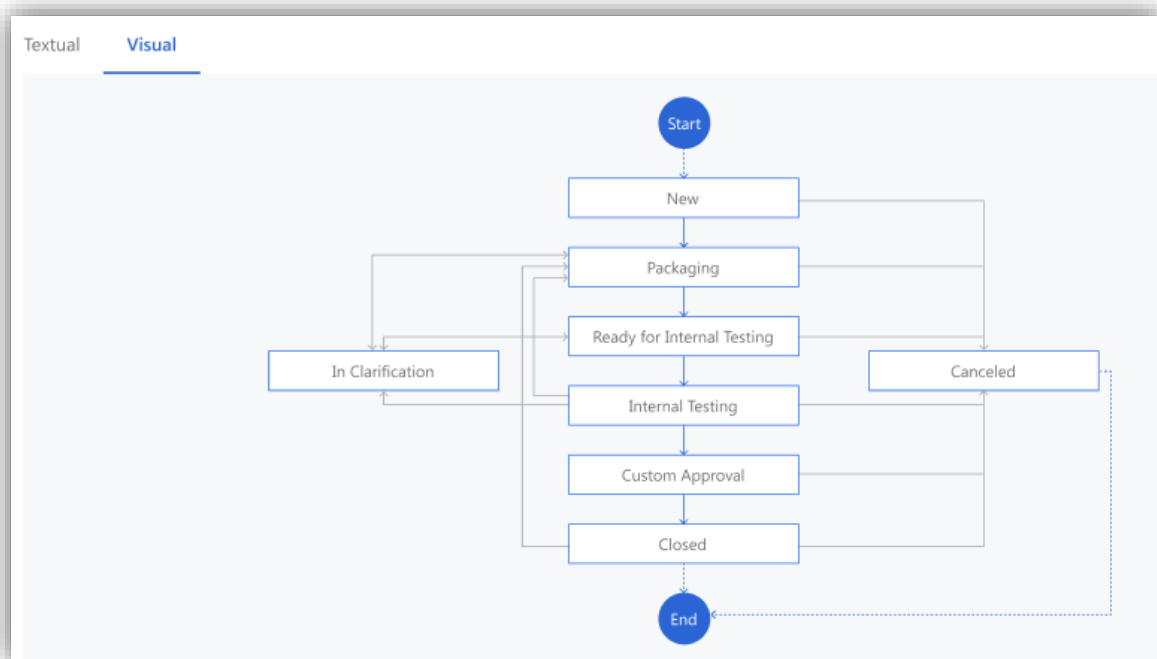
Release Notes

PACE Packager Hub 1.0.0

The workflow is configured in the project settings and has two types of representations: textual and visual.

The screenshot shows the 'Project A' configuration page in the PACE Packager Hub. The left sidebar contains navigation links: General, User Access, Package Settings, Order Settings, SLA, Order artifacts storage, and Notifications. The 'Workflow' section is active, showing a table of workflow steps. The 'Order type' is set to 'App Packaging'. The table lists steps like 'New', 'Submitted', 'Incoming check', and 'Ready for packaging', each with a 'Flag', 'Description', 'Permissions', and 'Actions'.

Name	Flag	Description	Permissions	Actions
New	Initial status	Some description		Edit Delete
→ Submit (New > Submitted)	Primary transition		Supervisor Customer	Edit Delete
→ Cancel (New > Canceled)			Supervisor Customer	Edit Delete
Submitted		Some description		Edit Delete
→ Start incoming check (Submitted > Incoming check)	Primary transition		Supervisor Customer Engineer	Edit Delete
→ Cancel (Submitted > Canceled)			Supervisor Customer Engineer	Edit Delete
Incoming check		Some description		Edit Delete
→ Start incoming check (Incoming check > Ready for packaging)	Primary transition		Supervisor Customer Engineer	Edit Delete
→ Cancel (Incoming check > Canceled)			Supervisor Customer Engineer	Edit Delete
Ready for packaging	Final status	Some description		Edit Delete
→ XXX (Ready for packaging > XXX)	Primary transition		Supervisor Customer Engineer	Edit Delete
→ Cancel (Ready for packaging > Canceled)			Supervisor Customer Engineer	Edit Delete



Release Notes

PACE Packager Hub 1.0.0

3.5 View/change order attributes (fields)

Organization A / Project A

Order #3

Order ID: #3, Type: Initial order, Status: In packaging, View workflow

Primary transition | New sub-order

General | Messages | Artifacts | History | Links

Order Title: ABC

App Name: [Error label], App Version: 1.0.0, App Vendor: Vendor A

Complexity: Medium, Due Date: 5/9/2020 (1 day left), Packaging Technology: MSI

Packager: Arlene McCoy, QA Engineer: Kathryn Murphy, Assignee: Kristin Watson

Description: This marathon is about improving everyday, and building a strong community. Your support is everything to us. Our philosophy is to have fun, help others, and give back to the community. We aim to support the community and encourage the community through the running path. We encourage everyone to come and join us so that we can support each others journey. We will have water, soakers, snacks, a map, and plenty of other supplies.

Save

Order **attributes** are pieces that represent and describe an order and distinguish one order from another.

The system supports (1) **pre-defined** and (2) **user-defined attributes**. The first group of attributes can only be adjusted, and user-defined attributes can be completely arbitrary for each type of order and project.

The following attribute options can be changed: **Name**, **Availability** (enabled/disabled), **Type** (text – single/multiple lines; number; date; dropdown list – single/multiple choice; checkbox; system user), **Obligatoriness** (optional/required), **Default value**; **Field width** (how much space each field will take on display), **Description** and **Sequence** (in what order fields will be displayed).

All of this allows to dramatically customize how the order will look like. Each order type can have a unique set of attributes, and these settings also may differ from project to project.

Below you can see how these configuration settings look like in the project settings:

Release Notes

PACE Packager Hub 1.0.0

Project A

General

User Access

Package Settings

Main operational repository

Delivery repository

Package attributes

Automation

Order Settings

Order types

Order attributes

Workflow

Order artifacts storage

Order attributes (fields)

Some info message about the order of displaying on form

Order type: App Packaging [Jump to order types settings](#)

+ New | Use template

Name	Field type	Obligatoriness	Visibility	Field width	Description	Attribute type	Actions
Order title	Text (single line)	Required	Visible	3/3 view width		Pre-defined	Edit Delete Refresh
App name	Text (single line)	Optional	Visible	1/3 view width		User-defined	Edit Delete Refresh
App version	Text (single line)	Optional	Visible	1/3 view width		User-defined	Edit Delete Refresh
App vendor	Text (single line)	Optional	Visible	1/3 view width		User-defined	Edit Delete Refresh
Complexity	Dropdown list (single choice)	Optional	Visible	1/3 view width		Pre-defined	Edit Delete Refresh
Priority	Dropdown list (single choice)	Required	Visible	1/3 view width		Pre-defined	Edit Delete Refresh
Packaging technology	Dropdown list (single choice)	Optional	Visible	1/3 view width		User-defined	Edit Delete Refresh
Due date	Date	Optional	Visible	1/3 view width		Pre-defined	Edit Delete Refresh
Creation date	Date	—	Visible	1/3 view width		Pre-defined	Edit Delete Refresh
Completion date	Date	—	Visible	1/3 view width		Pre-defined	Edit Delete Refresh
Assignee	System user	Optional	Visible	1/3 view width		User-defined	Edit Delete Refresh
Packaging engineer	System user	Optional	Visible	1/3 view width		User-defined	Edit Delete Refresh
Description	Text (multiple lines)	Optional	Visible	3/3 view width		User-defined	Edit Delete Refresh

User-defined attribute

Attribute name:

Identifier:

☐ Hide

☐ Disable

Type: Text (single line)

Obligatoriness: ☒ Optional (can be empty) ☐ Required (must be filled-in)

Default value for new and existing items:

Field width:

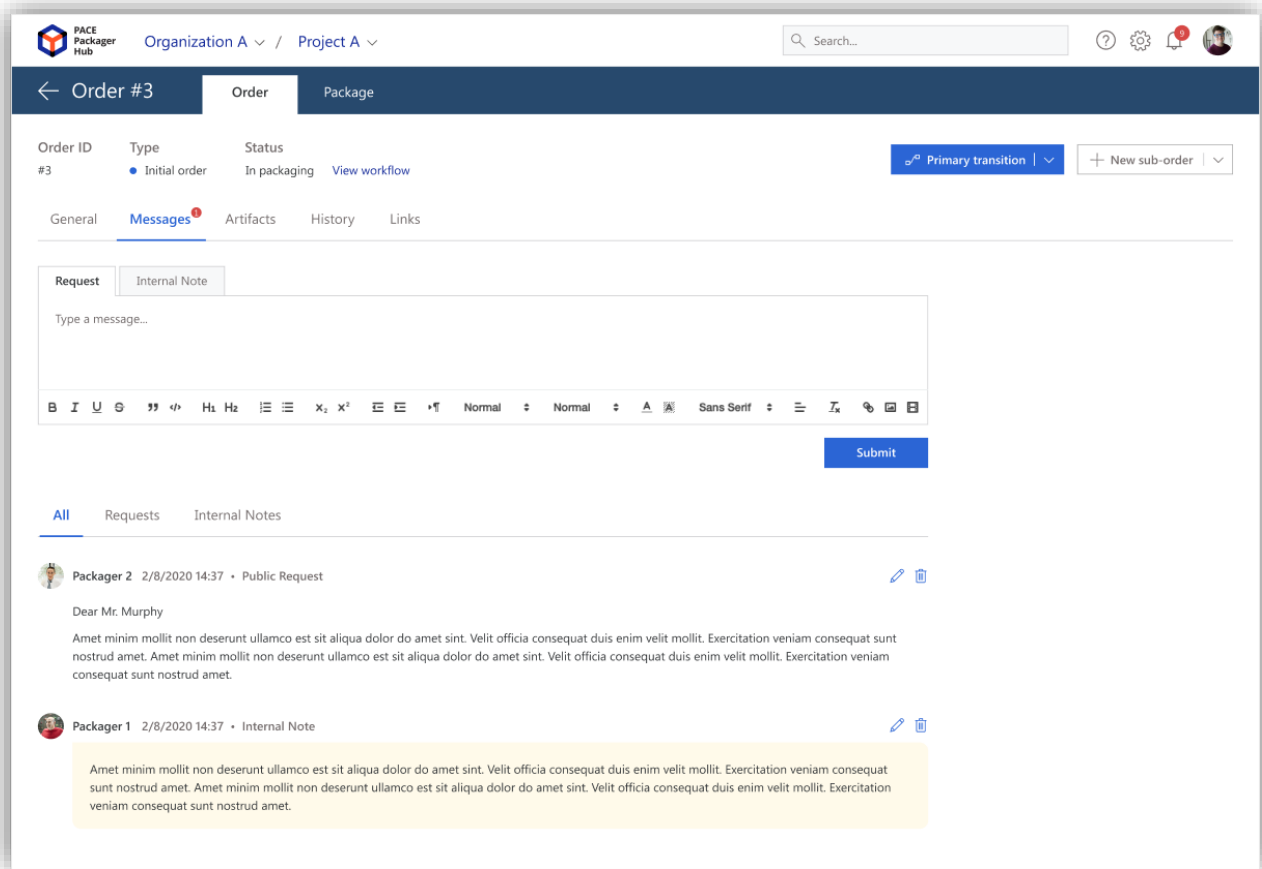
Description:

[Save](#) [Cancel](#)

Release Notes

PACE Packager Hub 1.0.0

3.6 Communicate about order (messages)



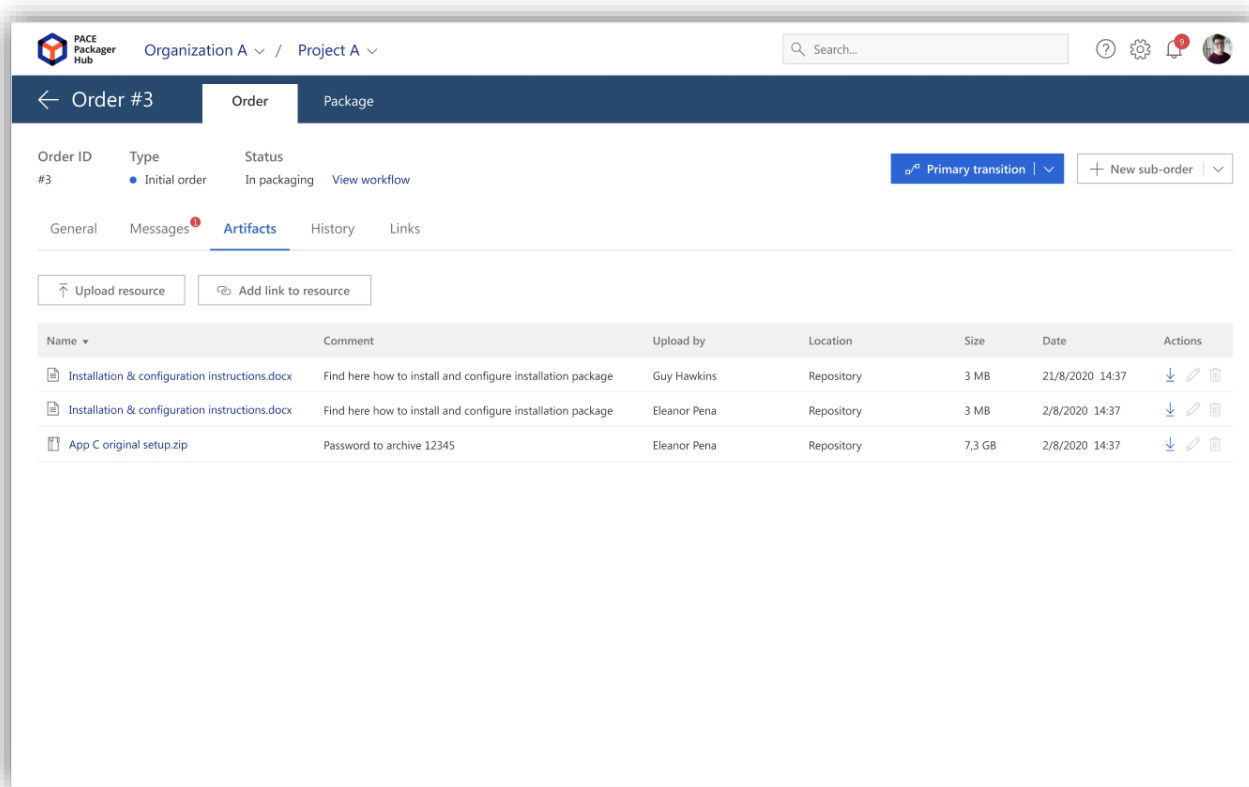
Communicate about the order right in this order. Messages support rich text formatting, inserting images right in the message body, and an advanced module for designing tables of any complexity.

There are two types of messages: (1) **Internal Note** that are used for communication only within the packaging team (Customer will not see these messages); while messages of the (2) **Public Request** type are visible to everyone who has access to this project. The Internal Note messages are additionally highlighted and have respective labels, which makes them easily distinguishable among others.

Each published message has metadata such as the date-time stamp of publication and editing (if edited), and the name of the user who did it.

Users of the Customer and Engineer roles can edit and remove only their own messages, while users of the Supervisor role can edit and remove any messages.

3.7 Upload/download order artifacts (files)



Transfer files between the Customer and Packaging team **right in the order**. It could be an app installer that should be repackaged, installation instruction document, complexity estimation reports, guidelines, and many others. There is no need anymore to manually use FTP or any other solutions.

The solution allows to **upload/download any files from small to big** using single web-portal and **add links to external resources**. The attached files are stored in the configured project repository for the order artifacts. **Connection settings are saved on the server** and are not available to users of the Engineer and Customer roles.

The added order artifact contains (1) comments where the user can describe the purpose of the attached artifact; (2) a name of the user who did it; (3) artifact location (repository/external link); (4) size on disk (only for files on repository); (5) date when the artifact was added. The **artifact metadata** allows to control who and when added the artifact.

Release Notes

PACE Packager Hub 1.0.0

3.8 Manage order links (references)

Organization A / Project A

Order #3

Order ID #3 | Type Initial order | Status In packaging | View workflow

Primary transition | New sub-order

General | Messages | Artifacts | History | Links

+ Add link

Package

Package ID	Package name	State	Actions
PRJ-A-3	VendorA-AppB-2.0.0-de-01	Not delivered	

Orders

Link Type	ID	Order Type	Order Title	Status	Actions
Parent	PRJ-A-1	Type 1	Abc	New	
Child	PRJ-A-1	Type 2	Abc	New	
Prerequisite	PRJ-A-1	Type 3	Abc	New	

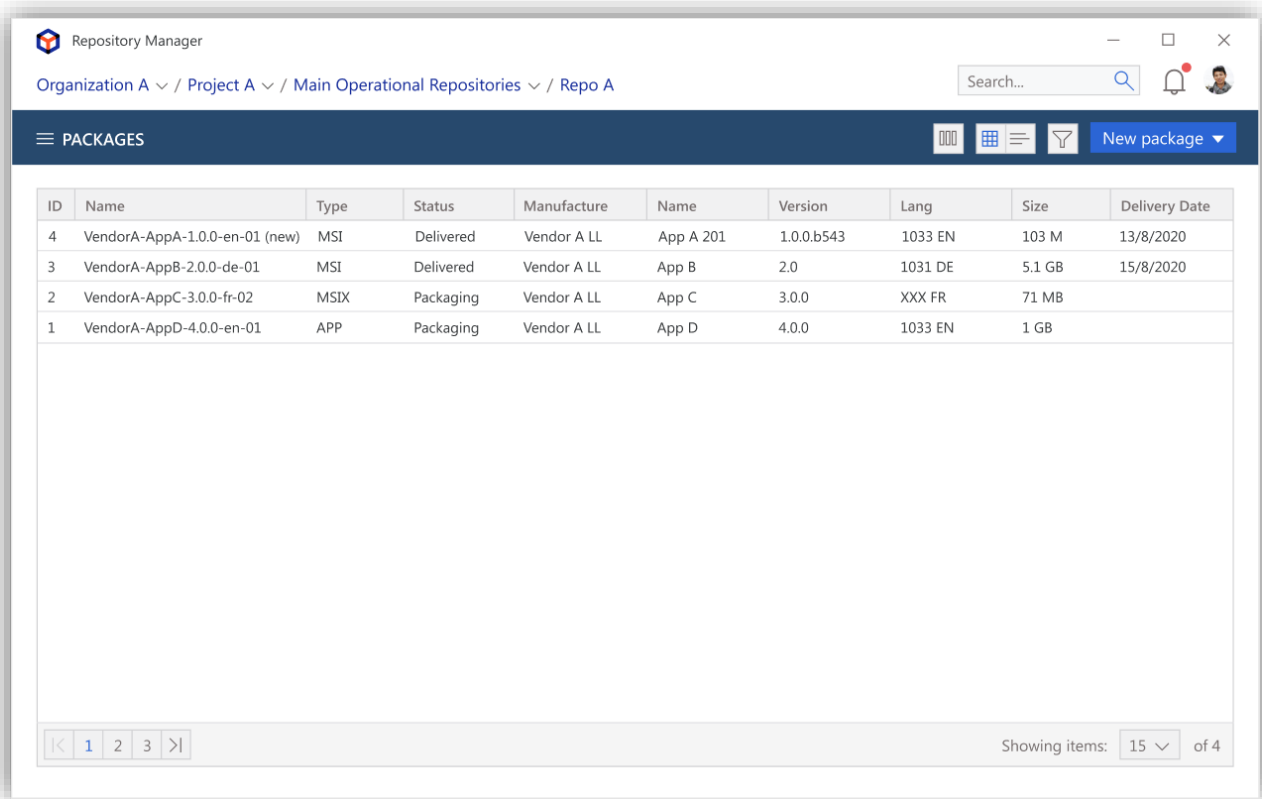
Links show what other orders and package are referenced to the current order. Some links are automatically managed, and all of them the user can edit or remove.

A link to the package is **automatically added** when the user creates a new package from the order. A link to the child order is also **automatically added** when the user creates a new **sub-order** (child order based on the order types hierarchy in the project settings). For instance, when 'Bug-fix' is created in response to the initial 'App Packaging' order.

Release Notes

PACE Packager Hub 1.0.0

3.9 Review, find packages



The screenshot shows the 'Repository Manager' application window. The breadcrumb navigation at the top indicates the path: 'Organization A' > 'Project A' > 'Main Operational Repositories' > 'Repo A'. A search bar is located in the top right corner. Below the navigation bar, there is a 'PACKAGES' section with a 'New package' button. The main area contains a table with the following data:

ID	Name	Type	Status	Manufacture	Name	Version	Lang	Size	Delivery Date
4	VendorA-AppA-1.0.0-en-01 (new)	MSI	Delivered	Vendor A LL	App A 201	1.0.0.b543	1033 EN	103 M	13/8/2020
3	VendorA-AppB-2.0.0-de-01	MSI	Delivered	Vendor A LL	App B	2.0	1031 DE	5.1 GB	15/8/2020
2	VendorA-AppC-3.0.0-fr-02	MSIX	Packaging	Vendor A LL	App C	3.0.0	XXX FR	71 MB	
1	VendorA-AppD-4.0.0-en-01	APP	Packaging	Vendor A LL	App D	4.0.0	1033 EN	1 GB	

At the bottom of the table, there is a pagination control showing 'Showing items: 15 of 4'.

View packages of the selected project or multiple projects in the configurable list. The user access to the project is configured separately for every project.

Packages that the user sees may be stored on various repositories and the user can access any of these packages directly from here. Depending on the project settings, the solution automatically connects to the repository either using the 'shared' credentials or asks to specify the personal ones. The user is abstracted from the address of the project's repository, connection details, package naming conventions, etc.

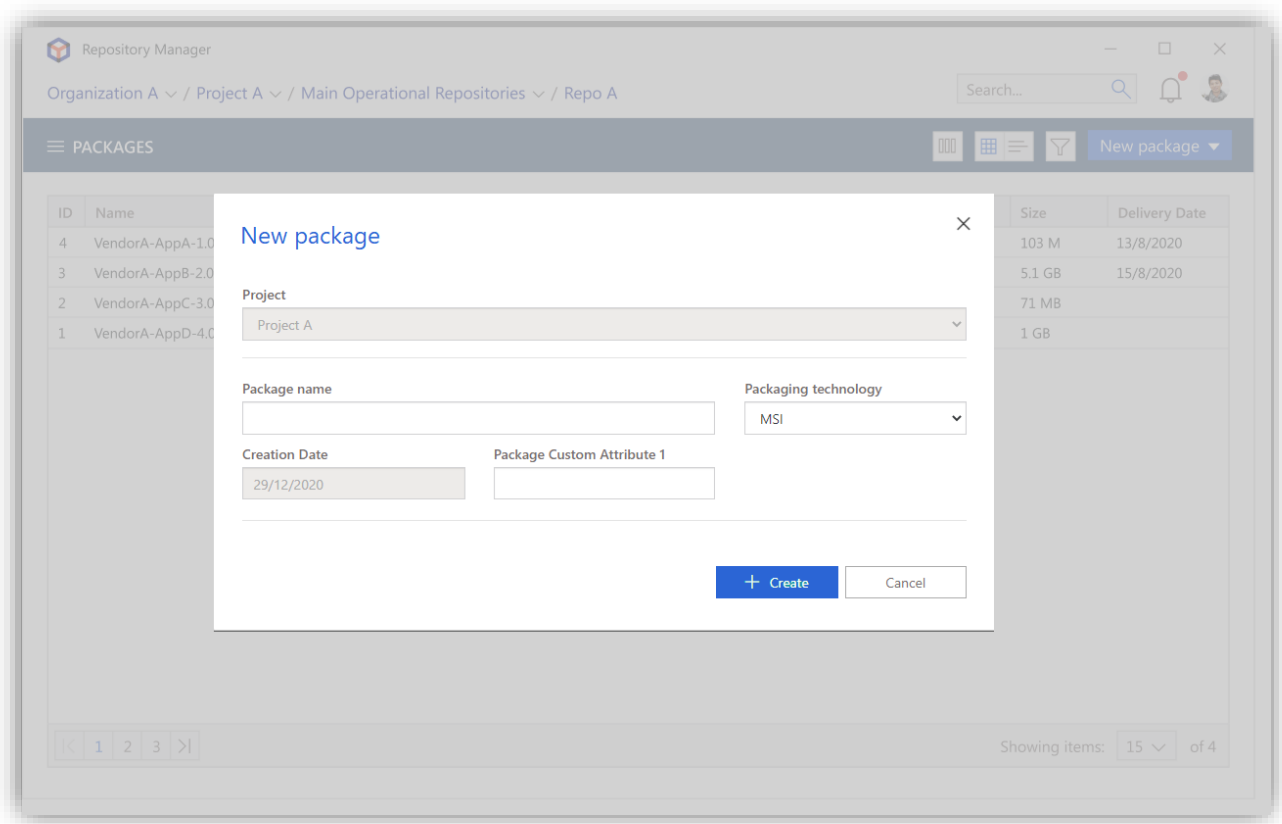
Details of packages that are displayed in the list consist of the pre-defined and user-defined attributes (fields). The set of these attributes are configured for every project separately and respectively may differ from project to project to meet the needs of each project.

The default list appearance can be customized for each user (package details to display; columns width and sequence; data sorting). By default, the list displays the following: *Package ID, Package Name, Packaging Technology, App Name, App Version, App Vendor, App Language, Architecture, Creation Date, Author*.

Release Notes

PACE Packager Hub 1.0.0

3.10 Create new package



Create new packages in the selected project with just a few clicks. The user does not need to know the address of the project repository and set of folders (their hierarchy, name) that should be created on the repository for every new package. All activities related to creating package folders on the repository are performed automatically by the solution to reduce human error, especially if one engineer is working on several projects simultaneously.

The repository address, connection details and package folders template of the new package are defined in each project's settings.

When creating a package, the user must specify all the required package attributes (fields). The list of package attributes and their settings, including the obligatoriness option, are also defined in the project settings and may differ from project to project.

A package can be created from scratch (and then manually linked with an order) or right from an existing order. In this case, values of order attributes are inherited to the package, and the reference between the package and order is also saved. Then, the user can easily switch between the package and order and vice versa.

3.11 View/change package attributes (fields)

The screenshot shows the 'Repository Manager' interface with the 'Package' tab selected. The package ID is 'PROJ-Name-3'. The 'General' tab is active, showing various attributes for the package. The attributes are organized into two columns. The left column includes 'Order title' (text field with 'Abc'), 'Supported OS' (dropdown with 'Windows 10 May 202... and more'), 'App Name' (text field with 'App A'), 'App Vendor' (text field with 'Vendor A'), 'Author' (dropdown with 'Engineer A'), and 'Description' (text area). The right column includes 'Packaging technology' (dropdown with 'MSI'), 'Architecture' (dropdown with 'x64'), 'Reboot' (dropdown with 'Not required'), 'App Version' (text field with '1.0.0'), 'App Lang' (text field with '1033 (English)'), and 'Creation Date' (text field with '1/9/2020 (3 days ago)'). A 'Save' button is at the bottom left. On the right side, there are sections for 'Operations' (with 'Show folder content' and 'More actions' buttons) and 'Automation' (with a 'Purpose' dropdown and a list of scripts: 'Get complexity', 'Check ICE', 'Get deployment CMD', 'Publish to SCCM', 'Script A', 'Script B', 'Script C', 'Script D'). A 'Run' button is at the bottom of the automation section.

Attribute	Value
Order title	Abc
Packaging technology	MSI
Supported OS	Windows 10 May 202... and more
Architecture	x64
Reboot	Not required
App Name	App A
App Version	1.0.0
App Vendor	Vendor A
App Lang	1033 (English)
Author	Engineer A
Creation Date	1/9/2020 (3 days ago)
Description	

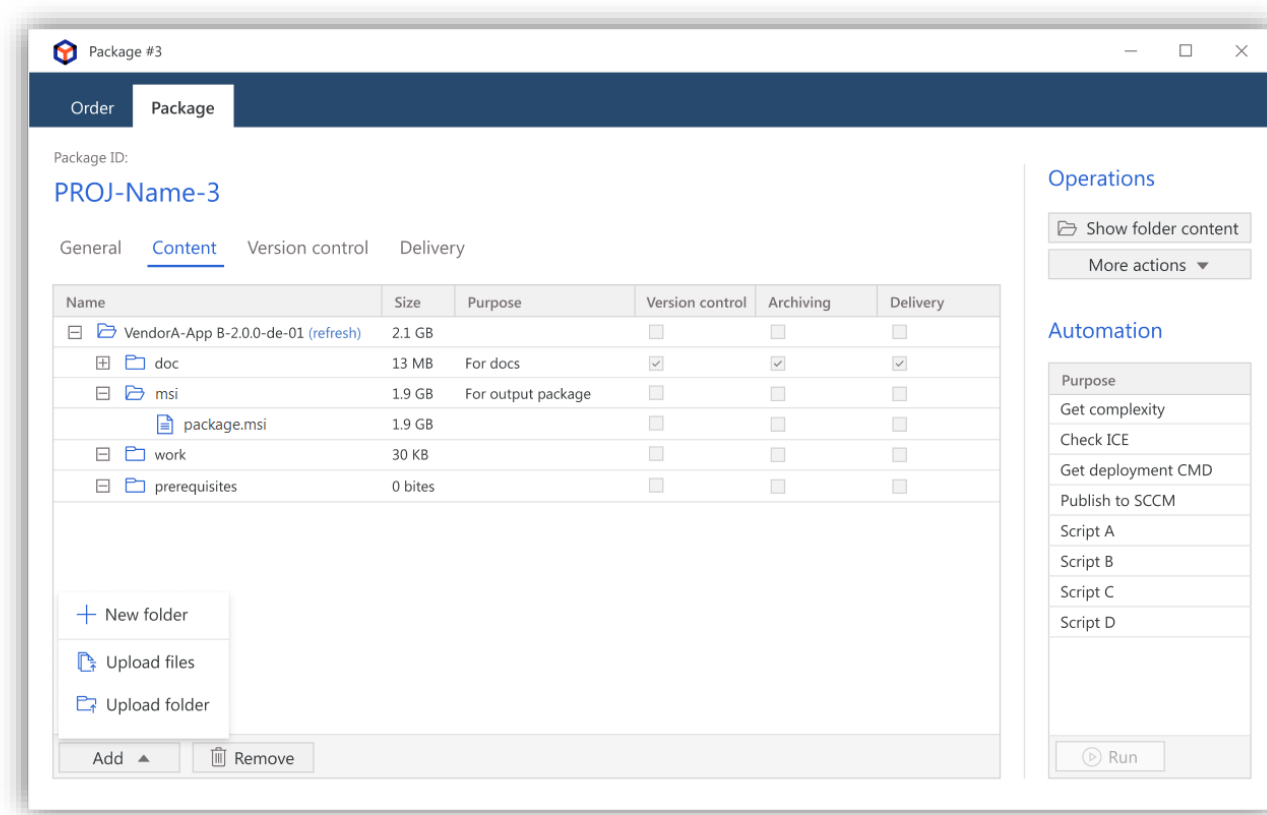
Package **attributes** are used to describe the package and distinguish it from other packages.

The system supports (1) **pre-defined** and (2) **user-defined attributes**. The first group of attributes can only be adjusted, while the user-defined attributes can be completely arbitrary for each project.

The following attribute options can be changed: **Name**, **Availability** (enabled/disabled), **Type** (text – single/multiple lines; number; date; dropdown list – single/multiple choice; checkbox; system user), **Obligatoriness** (optional/required), **Default value**; **Field width** (how much space each field will take on display), **Description** and **Sequence** (in what order fields will be displayed).

All this allows to dramatically customize how the package will look like. Set of package attributes may differ from project to project.

3.12 Manage package content (files)

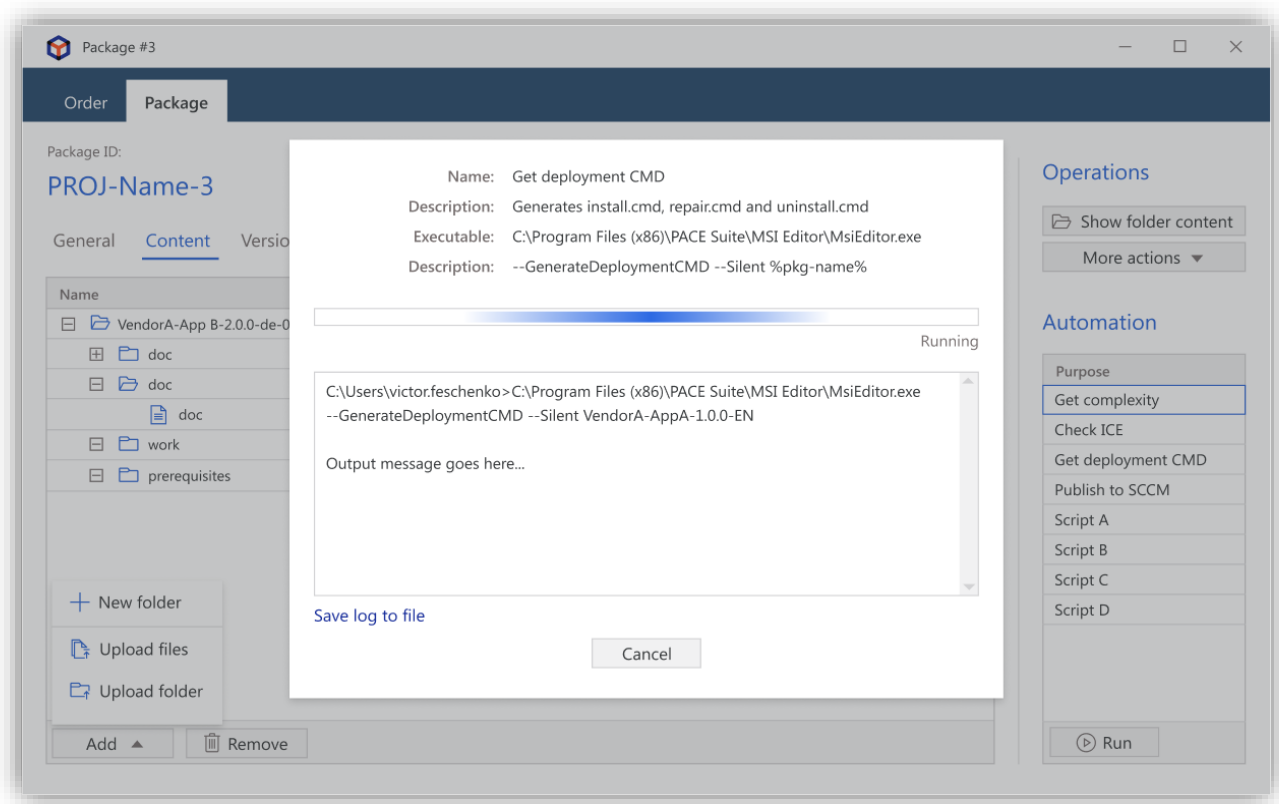


Package content is a collection of files and folders that represent this package. This set can vary significantly for different package types (msi, msix, appv, etc.) and at different stages of package creation.

The user is **abstracted from the connection details** and can easily work with package files no matter where and how they are stored. The solution **automatically connects to the project repository** and displays the content of the necessary package.

Here in the list, the user can **create, upload/download, and move package resources** or open the package root folder in Windows Explorer. In addition to basic operations, the user can **open MSI** directly in PACE Suite – MSI Editor or **run installation repackaging** on remote VM with the help of PACE Suite –Setup Capture.

3.13 Use automation for packaging



The solution allows to configure and run the project **automation** that **can consist of multiple scripts and command lines**. This automation is totally custom and may include running external tools, testing the package, integration with deployment systems, and so on. All this is stored in the project settings and may vary from project to project.

The selected **scripts** are stored in the database in order to **protect them from tampering**.

Each automation script or cmd can have an arbitrary set of **arguments, which can be static and dynamic**. **Dynamic** arguments are system variables that **are replaced with values** during runtime.

Example of system variables:

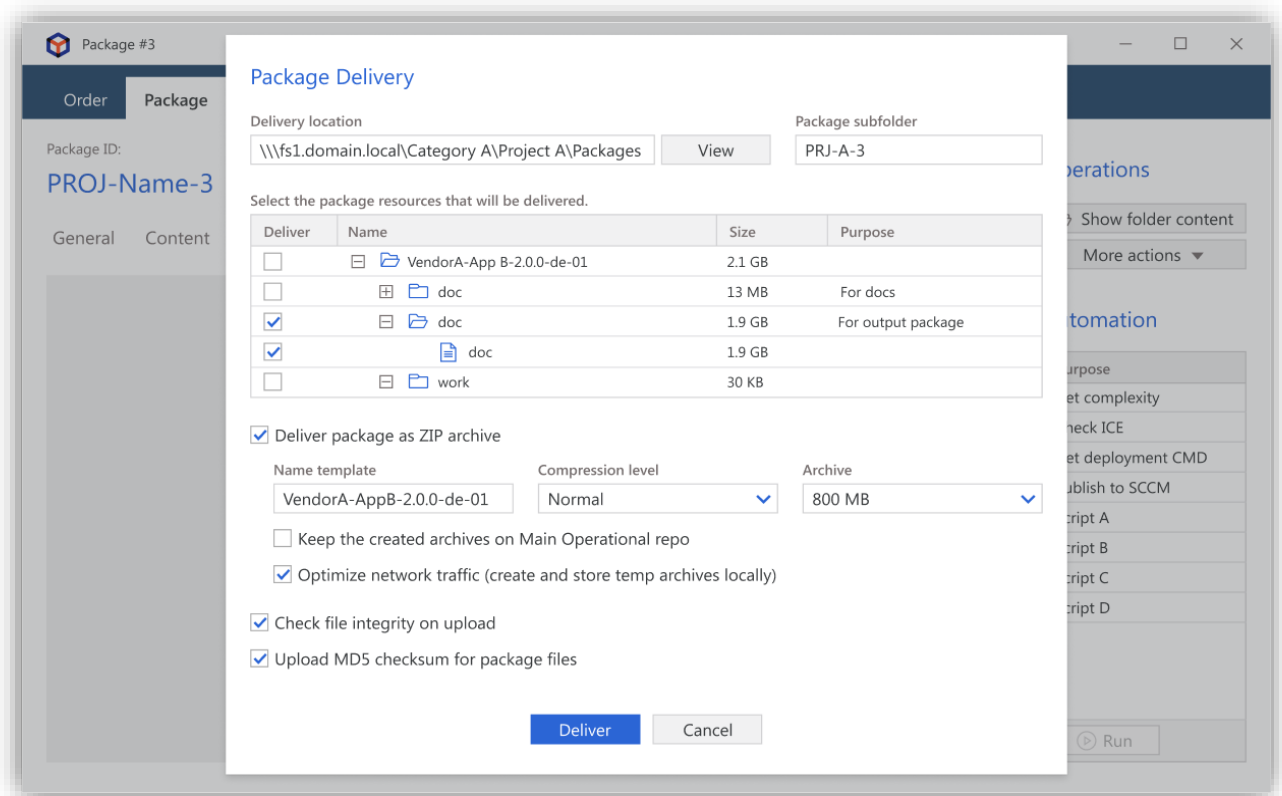
- %path-to-output-pkg% - a path to the folder of the 'For output package' purpose;
- %path-to-pkg-doc% - a path to the folder of the 'For package documentation' purpose;
- %path-to-in-resources% - a path to the folder of the 'For incoming resources' purpose;
- %ask-for-path-in-user% - the user will be prompted to select a custom path via UI;
- %pkg-name% - a value of the "Package Name" attribute of the Package entity;
- %pkg-id% - a value of the "ID" attribute of the Package entity;

Release Notes

PACE Packager Hub 1.0.0

- %pkg-tech% - a value of the "Packaging technology" attribute of the Package entity.

3.14 Deliver package to Customer



The process of delivering the package to the customer is greatly simplified and automated.

There is no need to remember the address and connection credentials to the server, where the package must be uploaded in order to deliver it to the customer. Server location and other delivery settings are automatically inherited from the project settings.

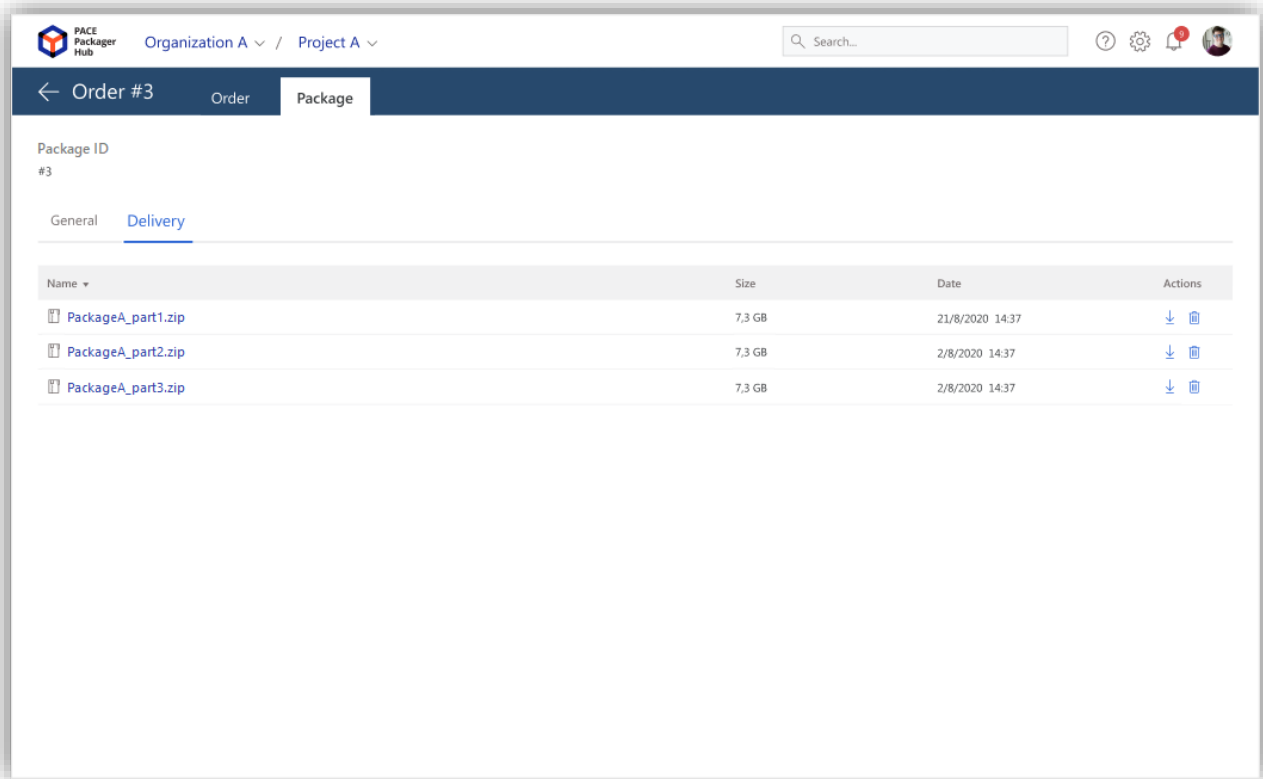
When delivering a package, the user can set/change folders and files that should be delivered; choose the compression settings such ZIP name template, compression level, and archive splitting by size; select network traffic optimization and others. One of the greatest things is that the solution checks the integrity of files when uploading them to the server.

It is possible to deliver packages to FTP and Network Share repositories. The engineer can fully control and observe the process of package delivery.

Release Notes

PACE Packager Hub 1.0.0

3.15 View/download delivered package



After opening the initial packaging order customer can switch to the package tab and then easily **download the output package right from the web portal**. There is no need anymore to connect to the FTP server manually and browse for the necessary files. The **solution identifies what files belong to the package**, allows to download and (if needed) delete them using the friendly interface. It uses the connection settings stored in the project settings and establishes the connection automatically.