



Studio/Robot 2018.3 アクティビティパッケージ 依存の解決について

2018年10月

UiPath 株式会社 津田義史

目次

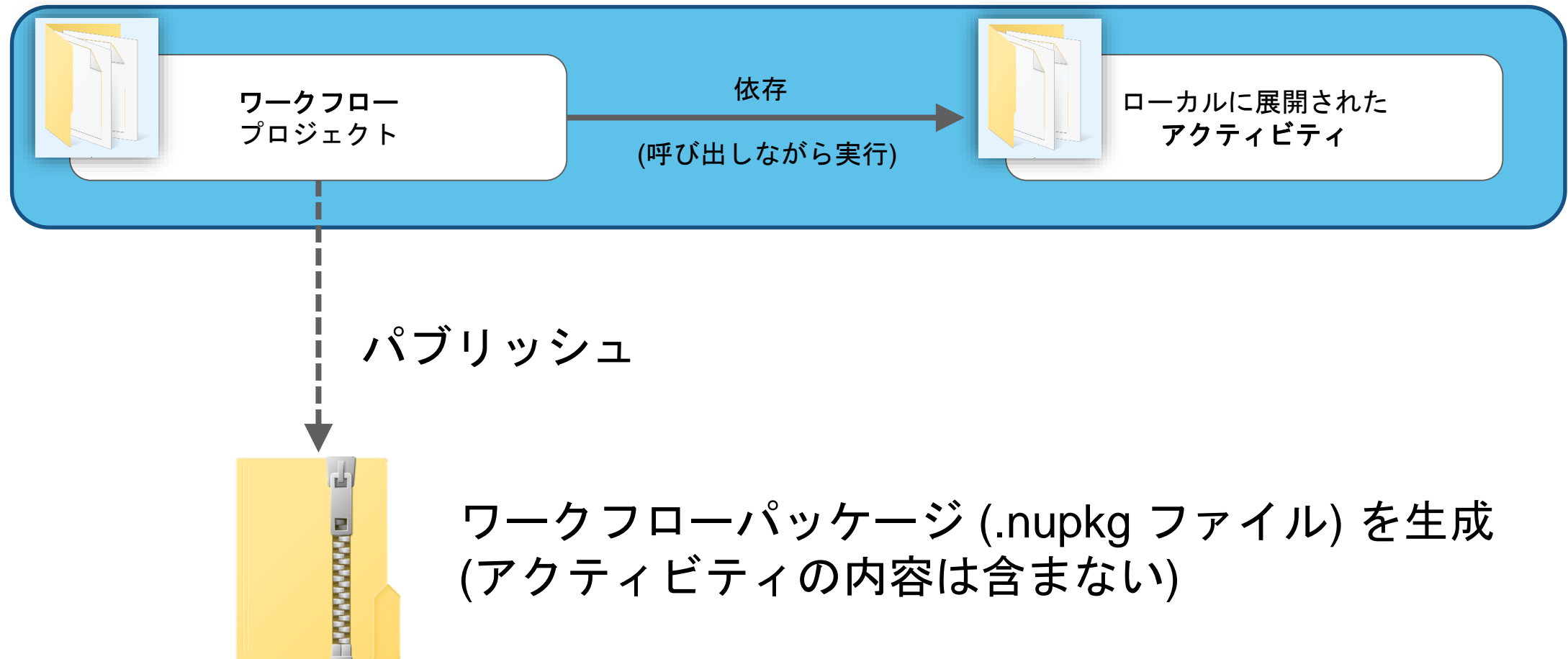
- パッケージファイルの展開と実行について
- Dependency per Project について
- Core アクティビティのデカップリングについて



パッケージファイルの展開と実行について

Studio でワークフローパッケージを作成

(Studio)



Robot でワークフローパッケージを実行

(Orchestrator)

パブリッシュした
ワークフローパッケージ(.nupkg) の
ソースフォルダ

アクティビティパッケージ (.nupkg) の
ソースフォルダ

取得・展開

取得・展開

(Robot)

ローカルに展開した
ワークフロー

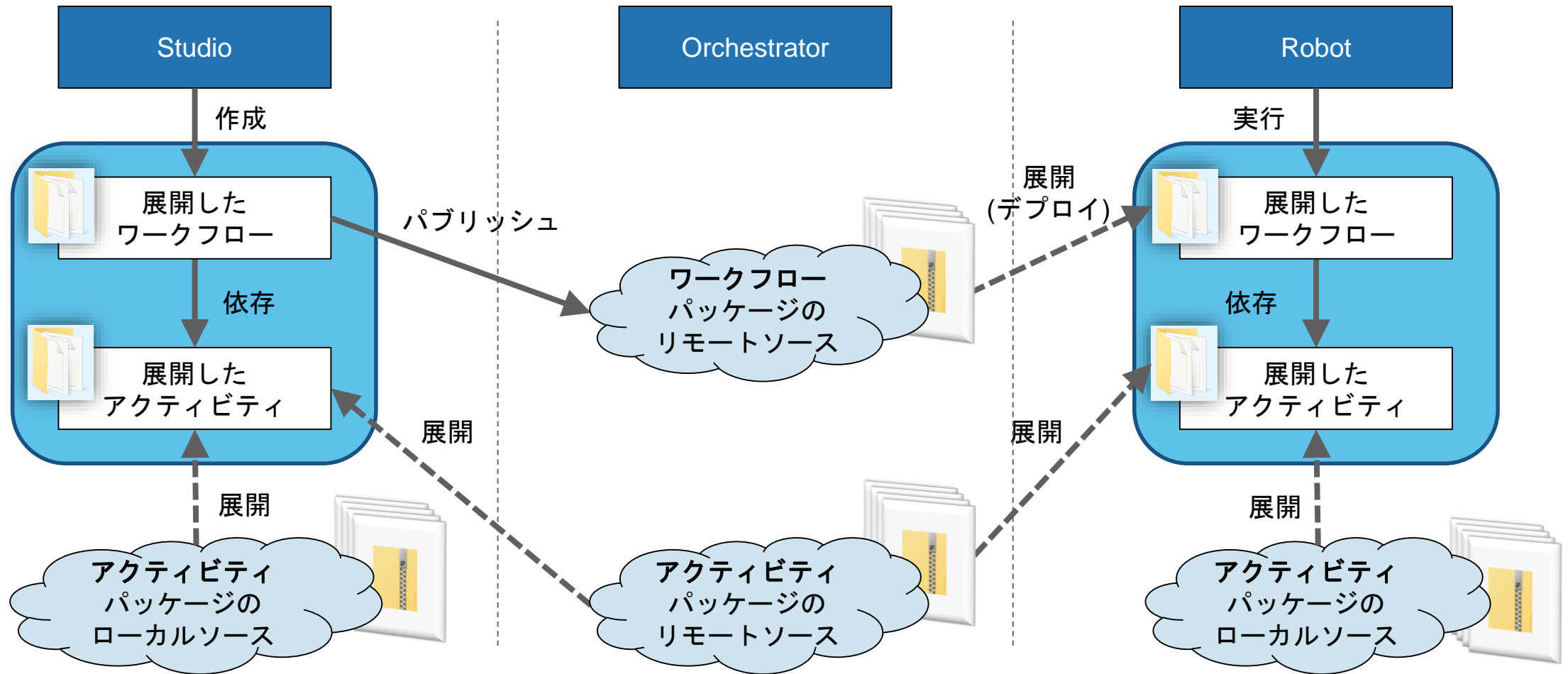
依存

(呼び出しながら実行)

ローカルに展開した
アクティビティ

パッケージの展開と実行

パッケージの解決の関係図



Studio/Robot のパッケージソースを設定する画面

Ui パッケージを管理

設定

プロジェクト依存関係

すべてのパッケージ

ローカル

オフィシャル

コミュニティ

nuget.org

既定のパッケージソース

- ☒ ローカル
C:\Program Files (x86)\UiPath\Studio*.Packages
- ☒ オフィシャル
<https://www.myget.org/F/workflow/>
- ☒ コミュニティ
<https://gallery.uipath.com/api/v2>

ユーザーが定義したパッケージソース

- ☒ nuget.org
<https://api.nuget.org/v3/index.json>

名前: 新しいパッケージソース

ソース: <http://newpackagesource> ... 追加

保存 キャンセル

アクティビティパッケージのソースについて

.nupkg パッケージファイルは、パッケージソースに配置されます。

先にローカルソースが検索されますが、それ以外の機能はローカルソースとリモートソースで同一です。

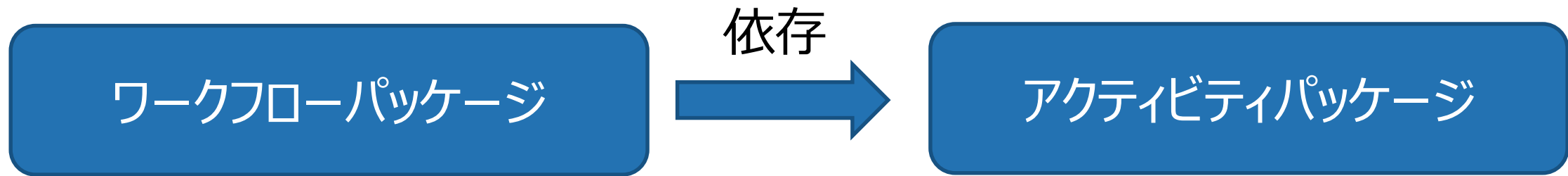
#	パスの種別	分類	概要	例
1	ローカルパス	ローカルソース	ローカルPCのパスです。ネットワークフォルダをローカルにマウントしたパスも使えます。	C:\Program Files\UiPath\Studio\Packages
2	UNCパス	リモートソース	ファイルサーバー上に共有したフォルダをソースとして利用できます。	\\FileServer\YourPackages
3	NuGet サーバーの URL	リモートソース	NuGet サーバーがホストするソースです。Orchestrator は NuGet サーバーを内包するため、Orchestrator の URL もソースとして利用できます。	https://platform.uipath.com/



Dependency per Project について

Dependency per Project について

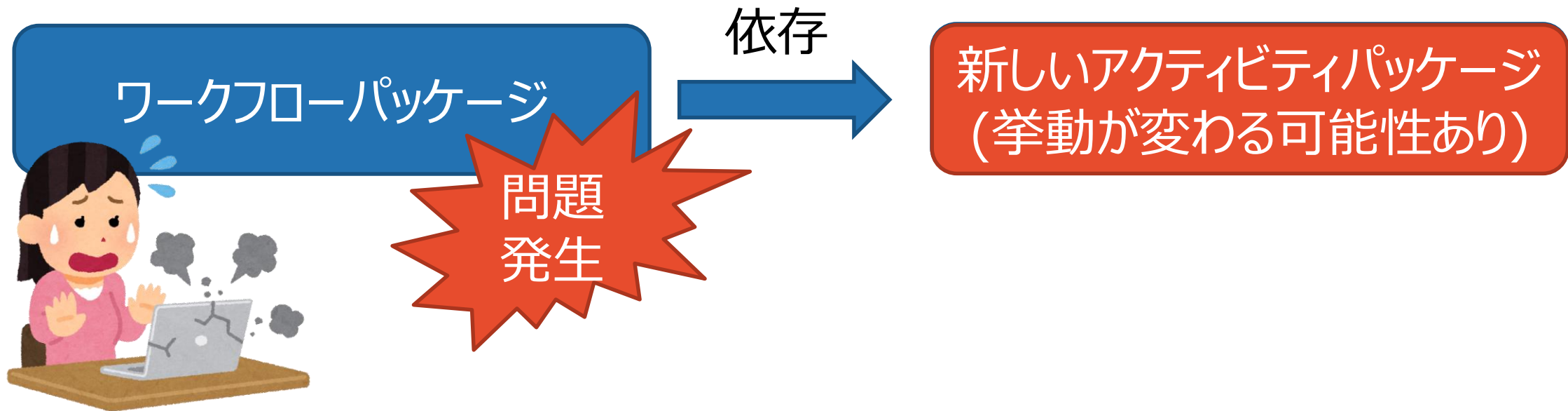
この機能の導入前 (18.2):



Dependency per Project について

この機能の導入前 (18.2):

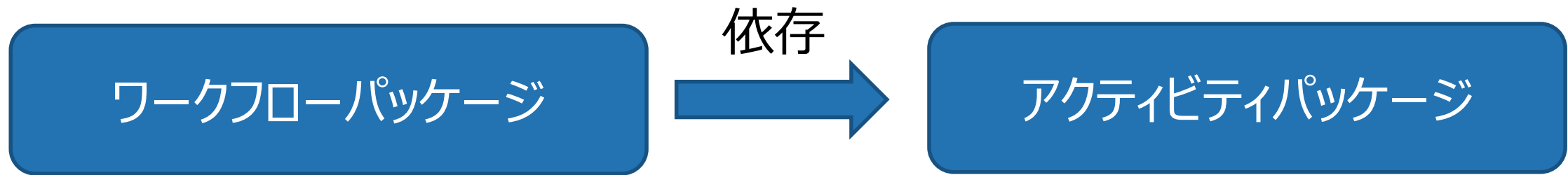
製品バージョンアップ



- 基本的に既存ワークフローは製品バージョンアップ後も問題なく動作するが、まれに問題が発生
- このため、弊社製品のバージョンアップには、特に丁寧な計画・テストが必要だった
- 製品本体のみバージョンアップし、アクティビティはバージョンアップしない選択肢も

Dependency per Project について

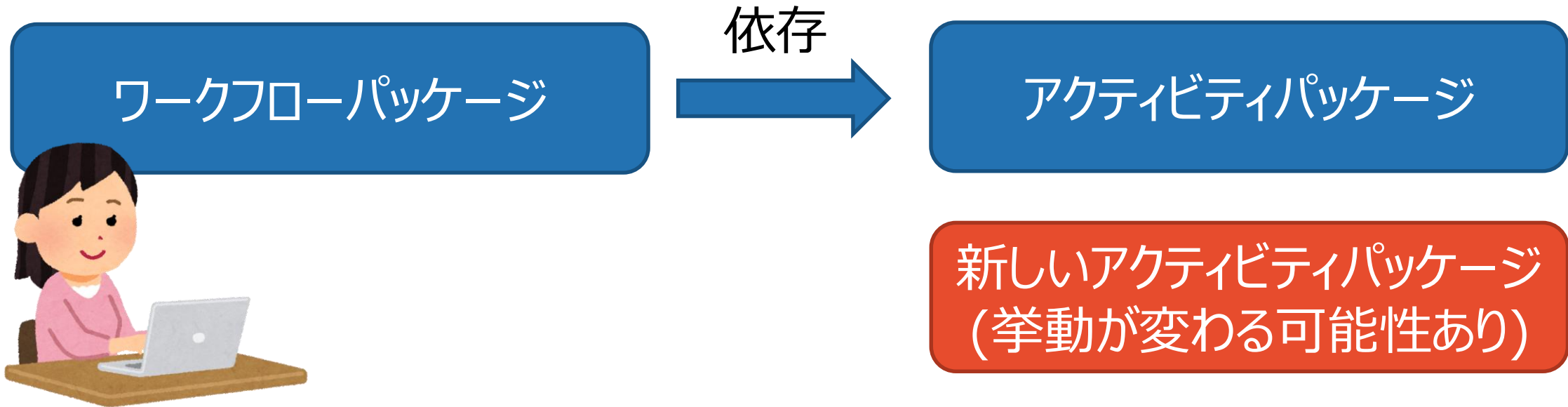
この機能の導入以後 (18.3):



Dependency per Project について

この機能の導入以後 (18.3):

製品バージョンアップ



- 複数のアクティビティバージョンが共存、使用したいバージョンを選択可能
- 既存ワークフローは製品バージョンアップ後も同じバージョンのアクティビティを使用するため、安定した稼働が可能
- 一方で、新規ワークフロー開発では新しいアクティビティを利用可能

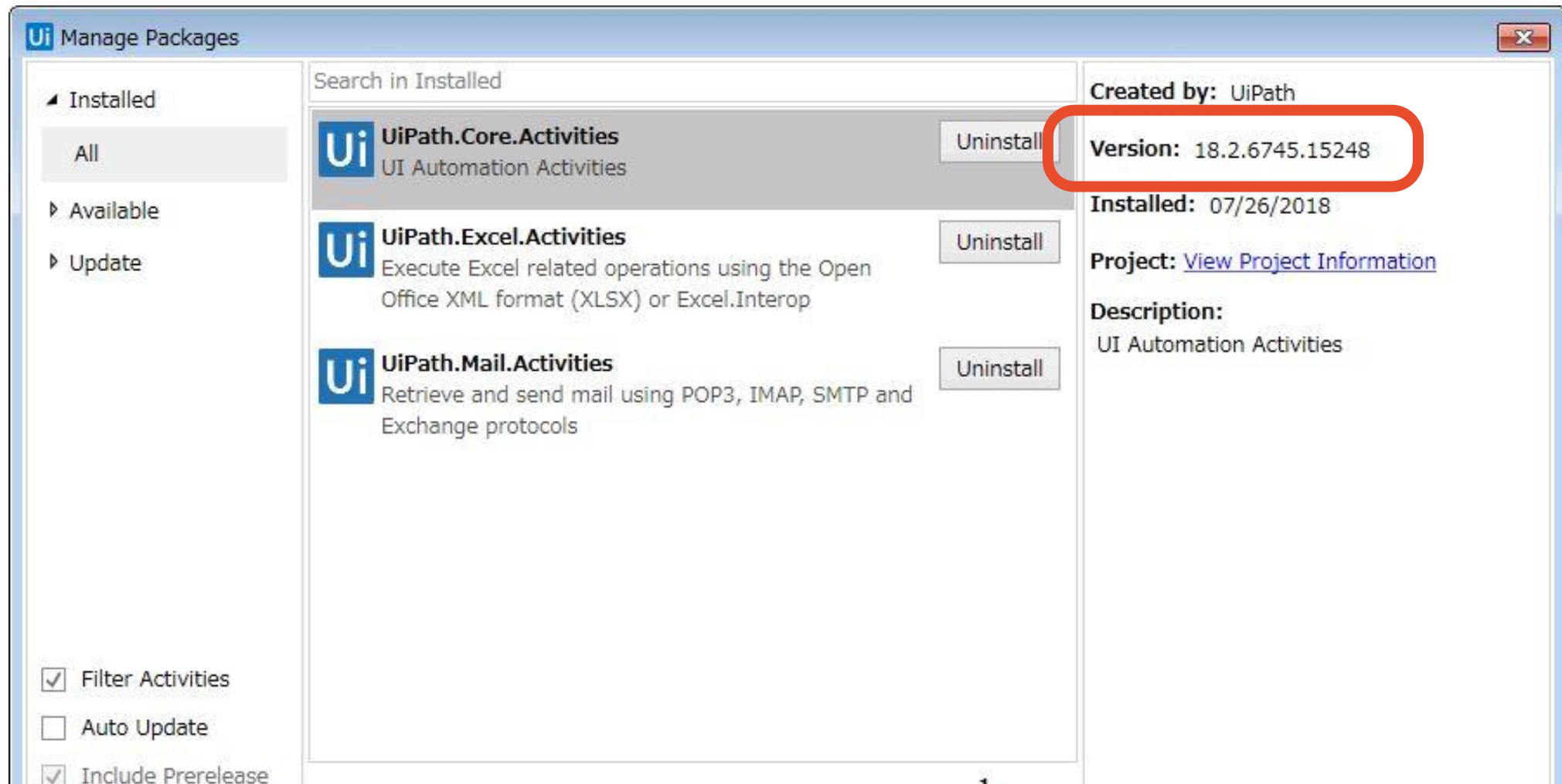
アクティビティパッケージのバージョン解決ルール

18.2 以前には、Applicable Version ルールしかありませんでした。
これは後方互換性の維持に問題があるため、18.3 以降では廃止されます。

#	ランタイムルール	製品バージョン	動作	特徴
1	Applicable Version (同じか、より新しいバージョン)	18.2 以前	ワークフローで指定したバージョンと同じか、より新しいバージョンであれば、任意のバージョンのアクティビティを呼び出して動作	UiPath 製品バージョンアップ後に、既存のワークフローが壊れるリスクあり 18.3以降では廃止
2	Strict (厳密に同じバージョン)	18.3 以後	ワークフローで指定したバージョンのアクティビティパッケージのみを使って動作	UiPath 製品バージョンアップ後にも、既存のワークフローが安定して動作できる安全なルール
3	Lowest Applicable Version (最も低い適用可能バージョン)	18.3 以後	#1 と同じだが、可能な限り低いバージョンのアクティビティを探して動作	後方互換性維持のため、18.2 以前の Studio でパブリッシュしたワークフローを18.3以降のRobotで実行するときに適用されるルール

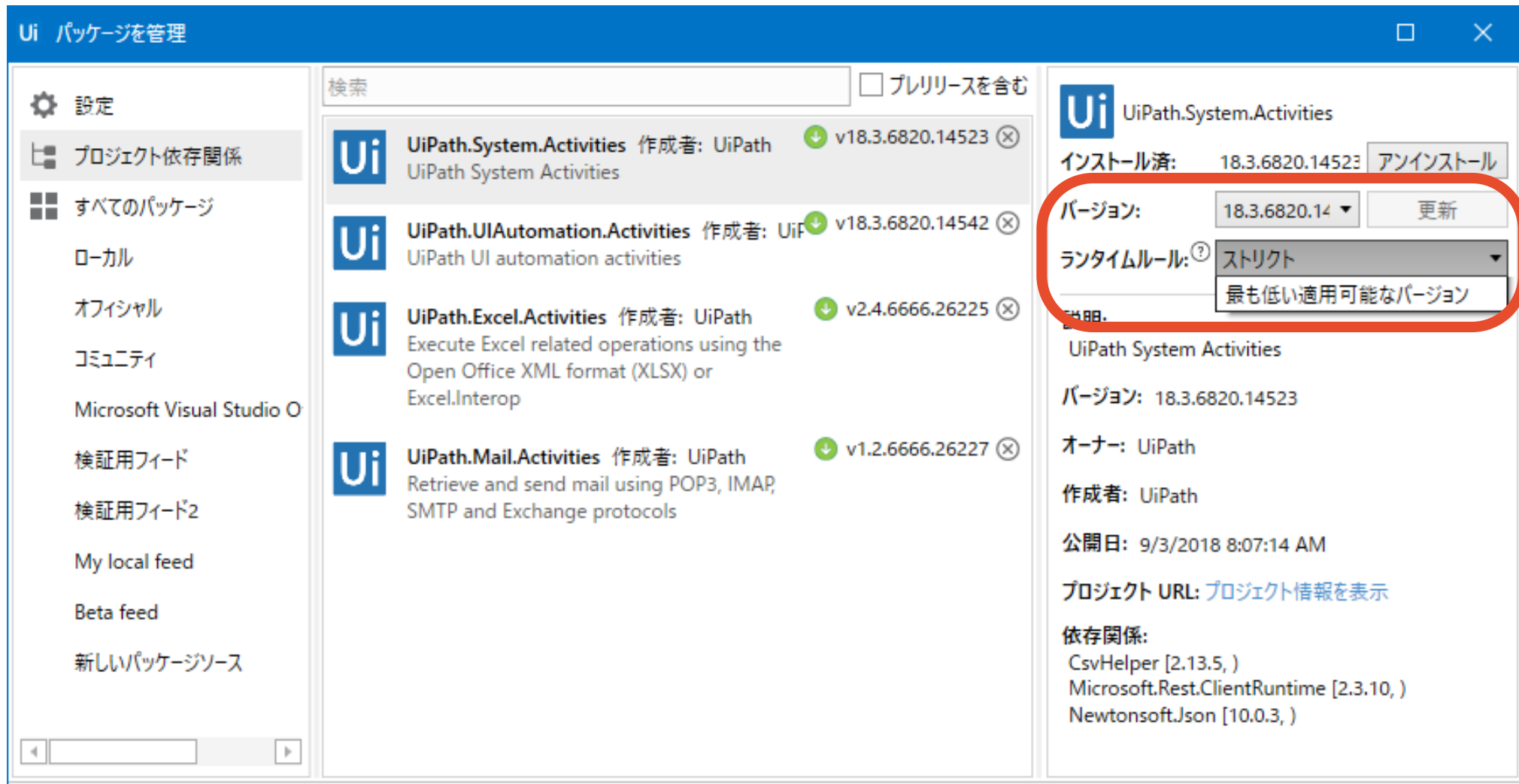
18.2 以前でのアクティビティパッケージのバージョン指定

アクティビティパッケージは、Studio にインストールされます。この環境で作成したワークフローは、ここで指定されたバージョンのパッケージに対して Applicable Version で動作します。



18.3でのアクティビティパッケージバージョンの指定

アクティビティは、ワークフロープロジェクトにインストールされます。
パッケージごとに、どのランタイムルールを適用するかを設定できます。



WFを実行時に適用されるランタイムルールまとめ

既存のワークフローを 18.3 で実行する際には、LAV ルールが適用されます。

WFを実行する Robot のバージョン WFを作成した Studio のバージョン	18.2 以前	18.3 以後
	18.2 以前	18.3 以後
18.2 以前	Applicable Version (equals or higher)	Lowest Applicable Version (※1)
18.3 以後	Not supported	Strict (※2)

※1. 18.2 以前の Studio で作成した WF プロジェクトを Studio 18.3 で開くと自動マイグレーションが実行されます。マイグレーションされた WF には、Strict が適用されるようになります。

※2. 18.3 以降の Studio で作成した WF プロジェクトでは、既定で Strict が適用されますが、この設定を手動で LAV に変更することもできます。(既定の Strict のままで運用されることをお勧めします)

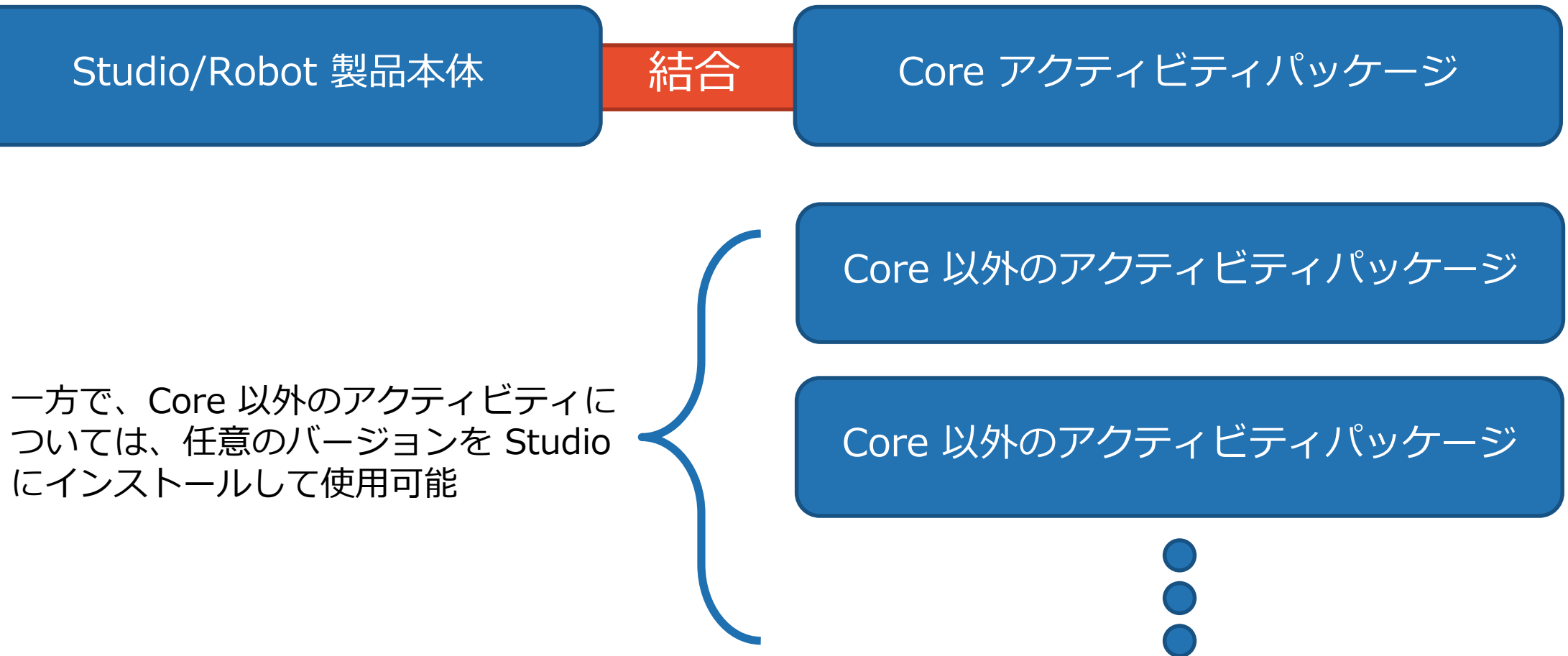


Core アクティビティのデカップリングについて

Core アクティビティのデカップリングについて

18.2 以前:

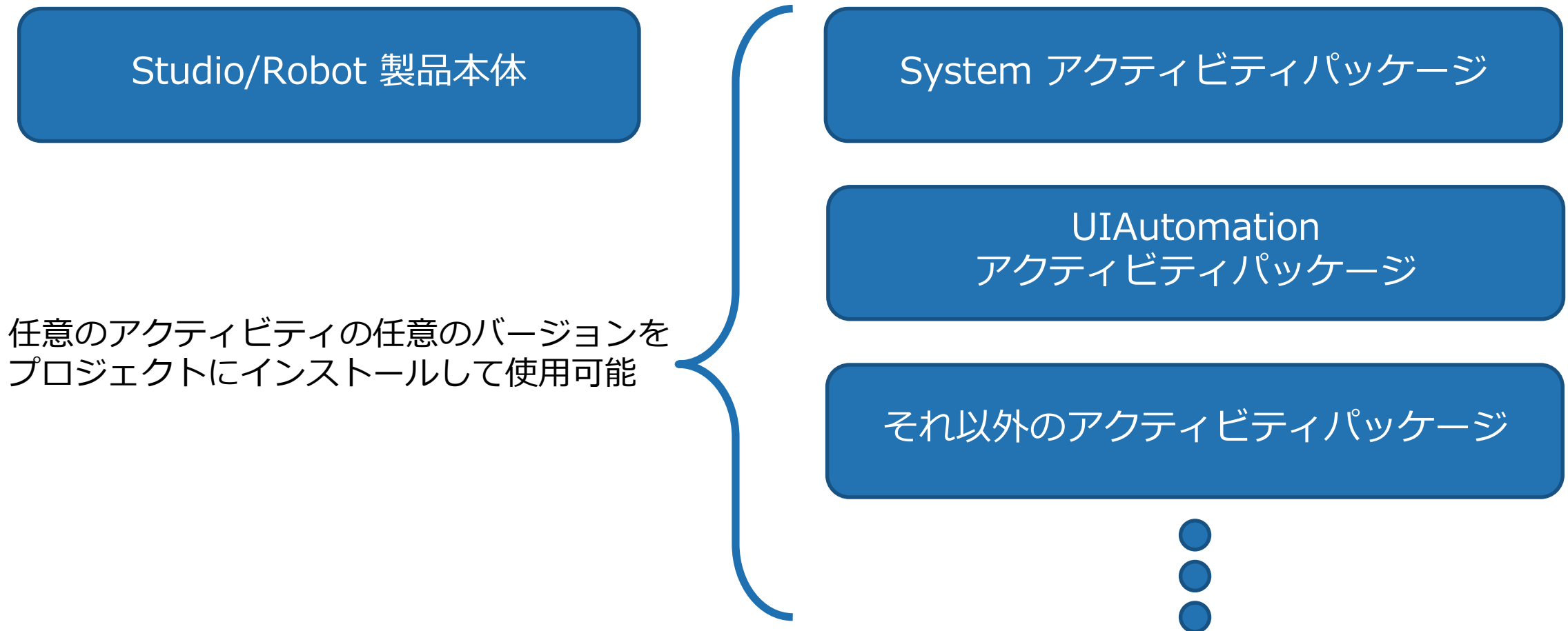
Core アクティビティは製品本体と密に結合していたため、Core アクティビティについては Studio/Robot に同梱されたバージョンのものしか利用できなかった



Core アクティビティのデカップリングについて

18.3 以後:

Core アクティビティは System と UIAutomation に分割され、**製品本体と分離された**
このため、プロジェクトごとに任意のバージョンを指定できる





Thank you!