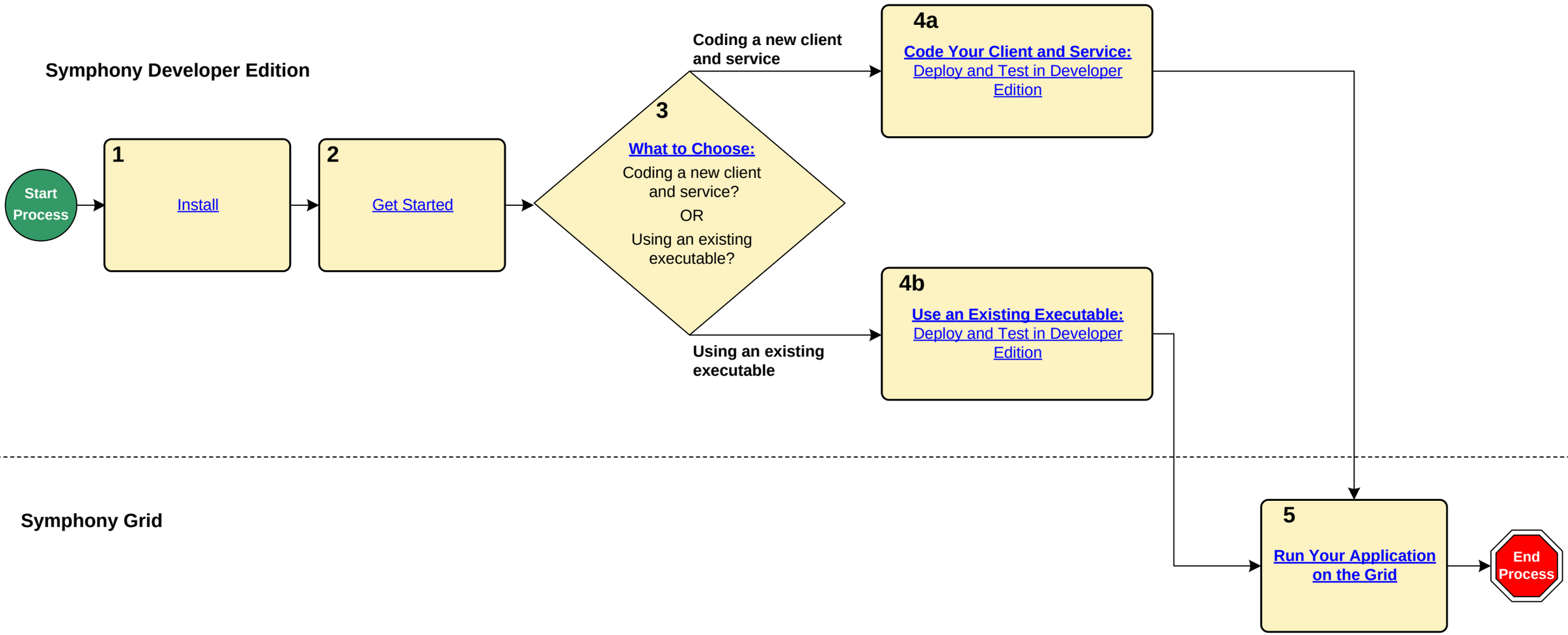
Getting Started: Developer Overview



About this document

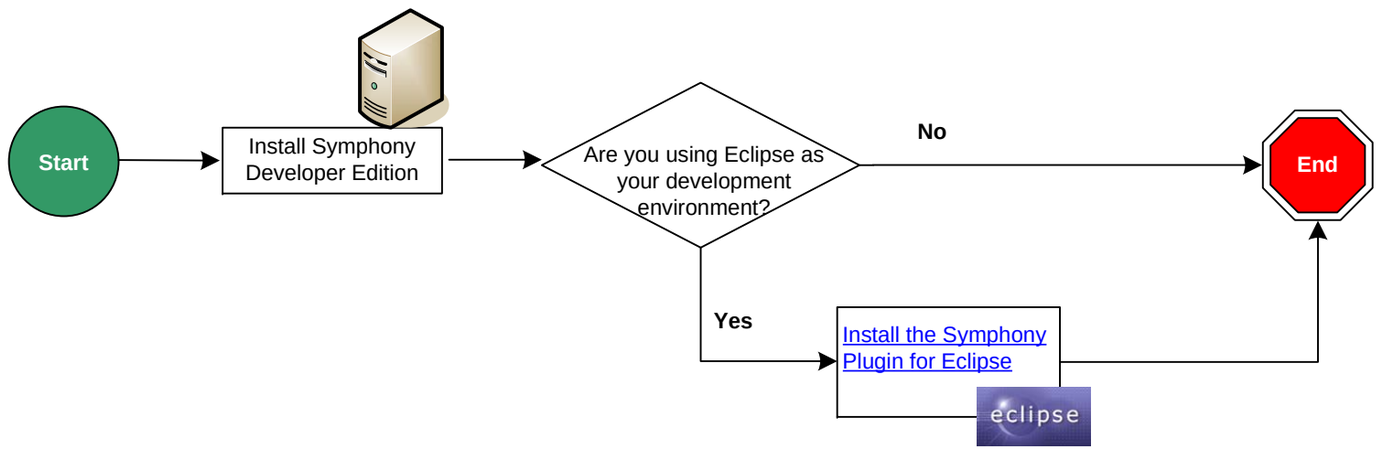
This document gives you an overview of the steps you need to go through to develop and put your application on the grid. This document also highlights steps you can take through the Platform Management Console.

Printing this document

Print using 8.5 in. x14 in. (legal size) paper, landscape orientation.
Be sure to print all pages.

1. Install

Symphony Developer Edition



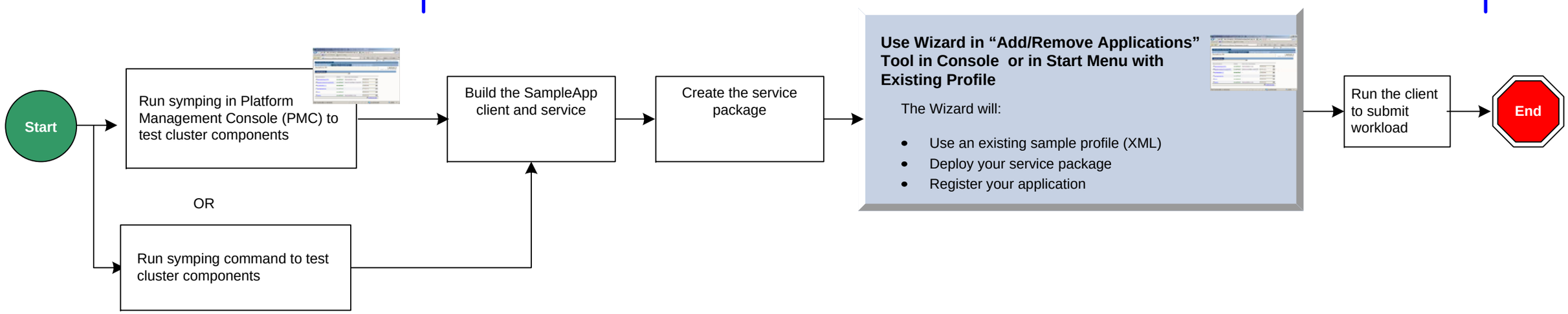
2. Get Started

Symphony Developer Edition

Familiarize yourself with the Symphony API. Follow the tutorial for your language:

- C++: Your First Symphony C++ Service
- Java: Your First Java Service
- .NET: Your First Synchronous C# Client and Service

Detailed instructions for the steps below are in the tutorials:



3. What to Choose:
Code Your Client and Service or Use an Existing Executable

Symphony Developer Edition

Use an Existing Executable

Reasons to choose:

- You have an existing executable and want to reuse it without recoding it or reimplementing it through the Symphony API.
- Your client may be a script, or you start the executable from the client.
- You have no access to the source code and cannot recode it.

Characteristics of this option:

- Data cannot be passed from the client to the service. Input, output is local to compute hosts.
- Only exit codes from the executable are returned to the client. Note that the symexec service which manages the executable can still return exceptions.
- The executable process is started and stopped for every invoke, adding system overhead.
- Inability to set, per executable invocation, remote values for OS limits, execution user and permissions.

OR

Code Your Client and Service

Reasons to choose:

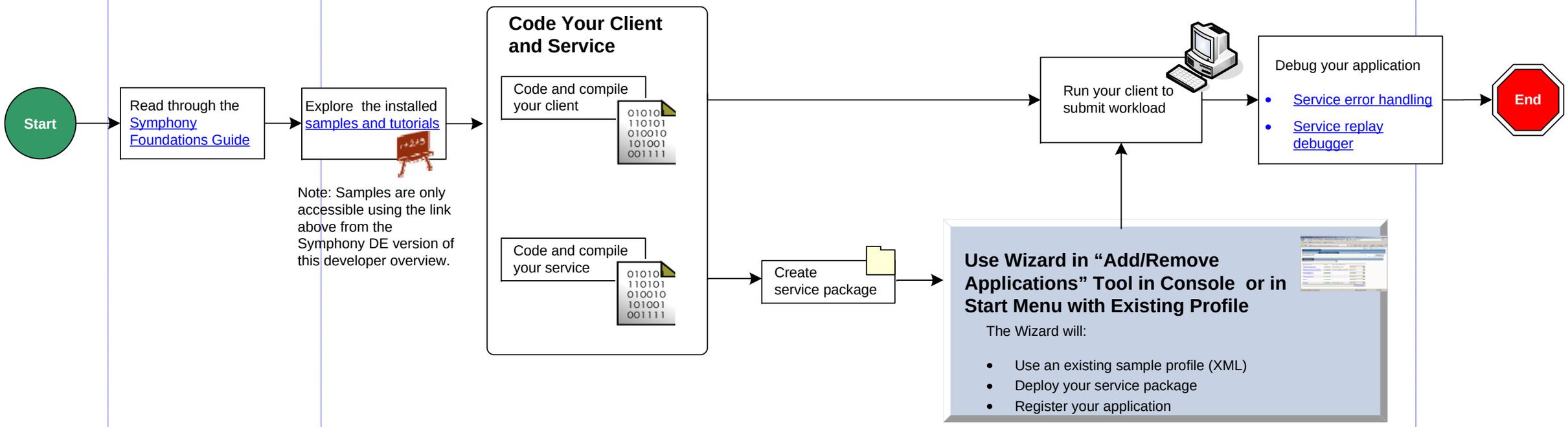
- You are coding a service from scratch.
- You want your service to be prestarted.
- You have data that you want to pass.

Characteristics of this option:

- Data is passed remotely from client to service. Input is passed from client to service and output and exceptions passed back to the client from the service.
- The service process can be prestarted with no need to start and stop the service process on every invoke, thus saving on system overhead.
- Ability to set, per application, remote values for shell, OS limits, execution users, and permissions.

4a Code Your Client and Service: Deploy and Test in Developer Edition

Symphony Developer Edition



4b Use an Existing Executable: Deploy and Test in Developer Edition

Symphony Developer Edition

Start

Read through the Symphony Foundations Guide

Explore the installed samples and tutorials

Note: Samples are only accessible using the link above from the Symphony DE version of this developer overview.

Use the cluster administrator account to run executable binaries?

No

Yes

Are executable binaries accessible from all compute hosts?

No

Yes

Executable binaries will run under the same application and same user account?

No

Yes

Add a new symexec application

Create a package:

- Put your executable in it
- Include any dependent libraries and files

Use the symexec application profile template to create your profile:

1. Open the symexec file in SOAM_HOME\4.1\samples\Templates\SymexecApplicationProfileTemplate.xml

2. Modify the application name to your executable name.
Do not change any other parameters.

3. Save your XML with a new file name.

Use Wizard in "Add/Remove Applications" Tool in Console or in Start Menu with Existing Profile

For this **typical** option, the Wizard will:

- *Use the profile you modified*
- *Deploy your package*
- *Register your application*

For more details, see Executable feature.

Use out-of-box symexec application: Select Symphony Workload > Configure Applications in Console

Enable symexec application.
See Executable feature.

Submit Work

symexec command

Specify executable with symexec:

- Specify executable if package was deployed
- Specify path and executable if executable is on shared filesystem or copied to compute hosts

OR

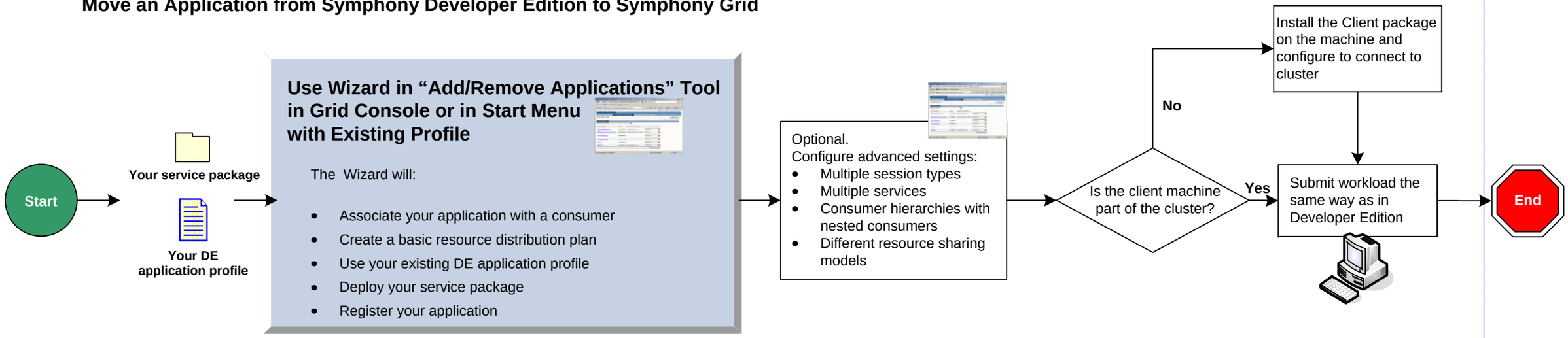
Run executable

Select **Monitor Workload >Run Executable**

OR

Write your own client and integrate with execution API

End

Symphony Grid

Move an Application from Symphony Developer Edition to Symphony Grid