



## **MaxCS Release 8.6**

# System Center Configuration Manager Client Remote Deployment Guide

June 24, 2019



## Contents

Introduction .....	3
Requirements .....	3
Before you Begin .....	3
Step 1: SCCM Preparation .....	3
Step 2: Create Packages and Applications .....	4
1. Create the Package for ISScript .....	4
2. Create the Package for VS2005 .....	7
3. Create the Package for UCCApi .....	7
4. Create the Application for AltigenJLIB .....	8
5. Create the Application for MaxAgent .....	14
6. Create the Application for MaxSupervisor .....	16
7. Create the Package for MaxCommunicator .....	16
Step 3: Deploy Applications to the Client Systems .....	17
Fresh Installations .....	17
Upgrade Installations .....	22
Example: How to Uninstall AltigenJlib .....	22
Known Issues .....	22

Altigen Communications, Inc.  
679 River Oaks Parkway, San Jose, CA 95134  
Telephone: 888-Altigen (258-4436) | Fax: 408-597-9020  
E-mail: [info@altigen.com](mailto:info@altigen.com) Web site: [www.altigen.com](http://www.altigen.com)

All product and company names herein may be trademarks of their registered owners.  
Copyright © Altigen Communications, Inc. 2019. All rights reserved.



---

## Introduction

System Center Configuration Manager (SCCM) is a systems management software product available from Microsoft. It helps administrators manage large groups of computers running versions of Windows operating systems.

This document explains how perform remote deployment of the MaxCS clients (specifically, MaxAgent, MaxSupervisor, and MaxInSight). These instructions assume that you are familiar with using SCCM to deploy applications within your environment.

During these procedures, you will first create the deployment packages in SCCM. Then you will deploy those package to the client systems.

## Requirements

- SCCM 1802 or above
- A SQL database

## Before you Begin

1. First, obtain from your Altigen representative:
  - The Remote Deployment package (this should include the ISScriptEngine, MicrosoftUCCAP, and AltigenJLIB folders)
  - The installation programs for each MaxCS client you want to install
2. Make sure that you have SCCM set up and functioning properly.
  - If you have successfully deployed .msi applications in the past, for other applications, then the procedures in this document will work correctly and you can skip the procedures in Step 1.
  - If you have never deployed .msi files with SCCM in your current environment, then we recommend that you follow the procedures in Step 1 to configure SCCM.
3. Once you have confirmed that SCCM is set up properly, then create collections for the client systems that need MaxAgent, MaxCommunicator, and MaxSupervisor.

---

## Step 1: SCCM Preparation

If you have never deployed .msi files with SCCM in your current environment, follow the steps in this section to configure SCCM. Then confirm that SCCM works properly by testing it on another application before you create Altigen packages and deploy them. One option is to do a test deployment of Google Chrome. Once the test deployment has completed successfully, continue with the remainder of this guide.

Detailed instructions for these steps can be found on the Microsoft Technet site at (<https://blogs.technet.microsoft.com/jchalfant/how-to-install-microsoft-sccm-current-branch-step-by-step-guide/>).

1. Prepare Active Directory for the Configuration Manager.
  - a. Create the System Management Container in Active Directory.
  - b. Extend the schema.

2. Install the following components on the server:

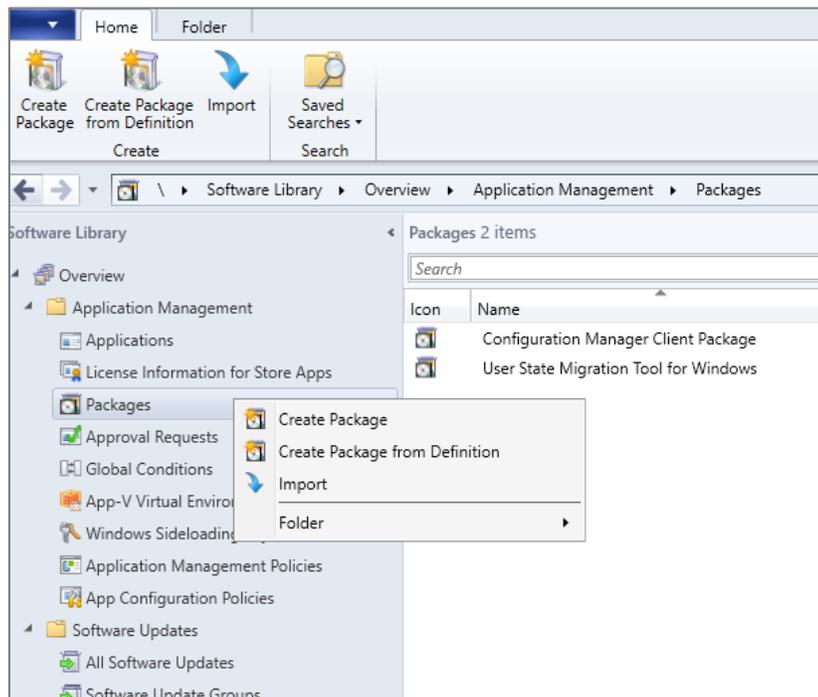
- Create the recommended disk drives for Configuration Manager
- Install IIS, RDC, BITS
- Install and configure Microsoft SQL Server 2017
- Install and configure WSUS
- Install latest Windows 10 ADK
- Extend the Active Directory schema for SCCM
- Create the System Management container in Active Directory
- Pre-create the SCCM database in SQL so we can set the location, size, and number of files
- Install Microsoft SCCM Current Branch 1802

## Step 2: Create Packages and Applications

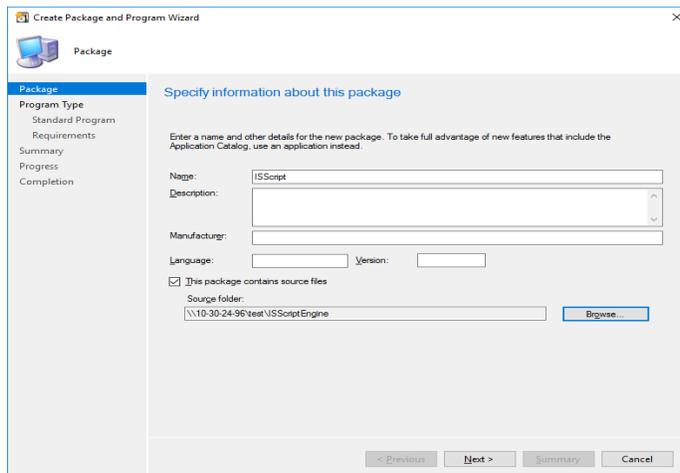
Create the deployment packages in the following order.

### 1. Create the Package for ISScript

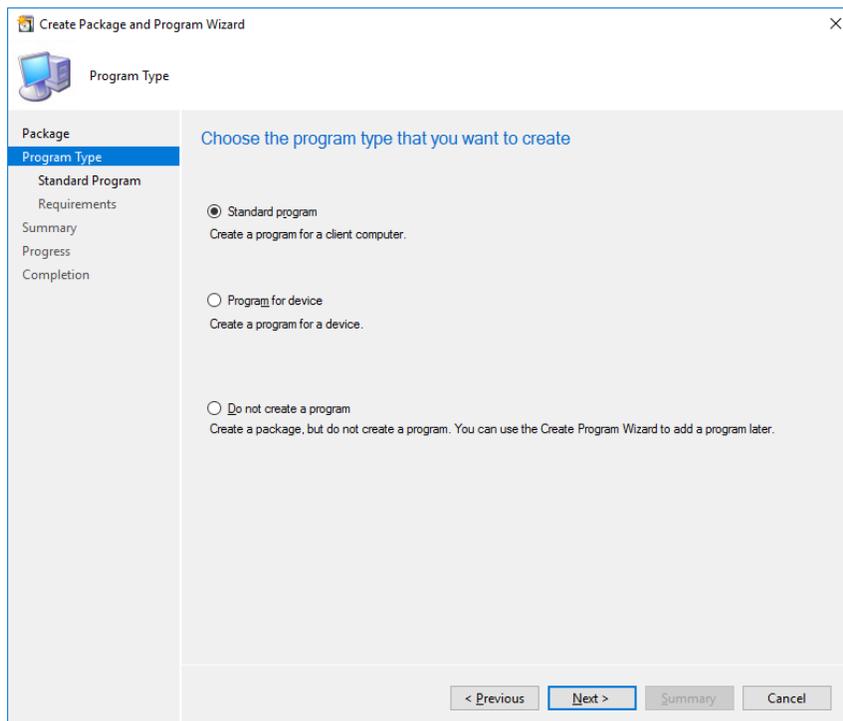
- A. In SCCM, select **Software Library > Overview > Application Management > Packages > Create Package**.



- B. Enter ISScript as the package Name, check the option *This package contains source files*, and specify the *ISScriptEngine* folder as the Source folder location.



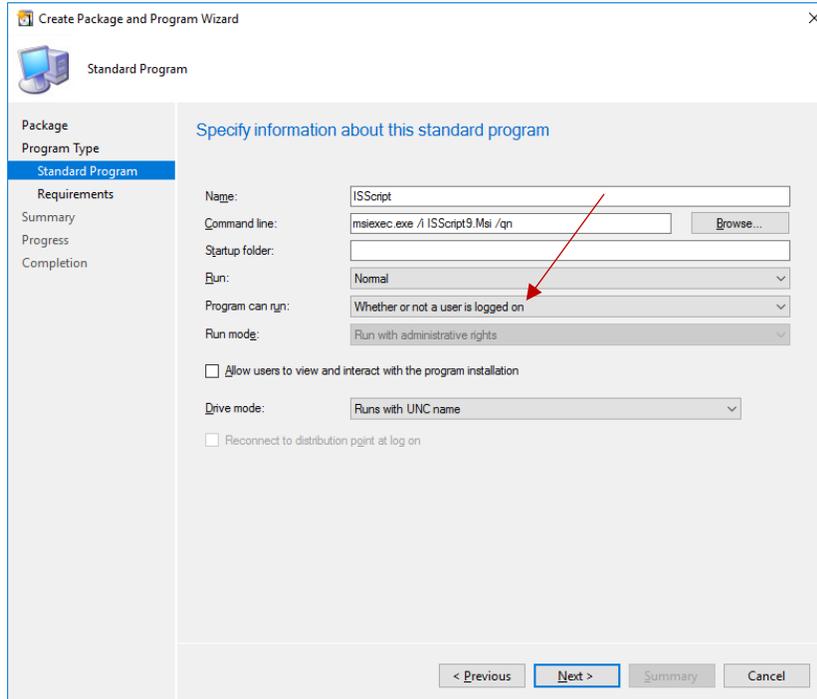
C. In the next panel, choose *Standard program* and click **Next**.



D. Enter the name **ISScript** in the *Name* field. Enter the following string into the Command line field:

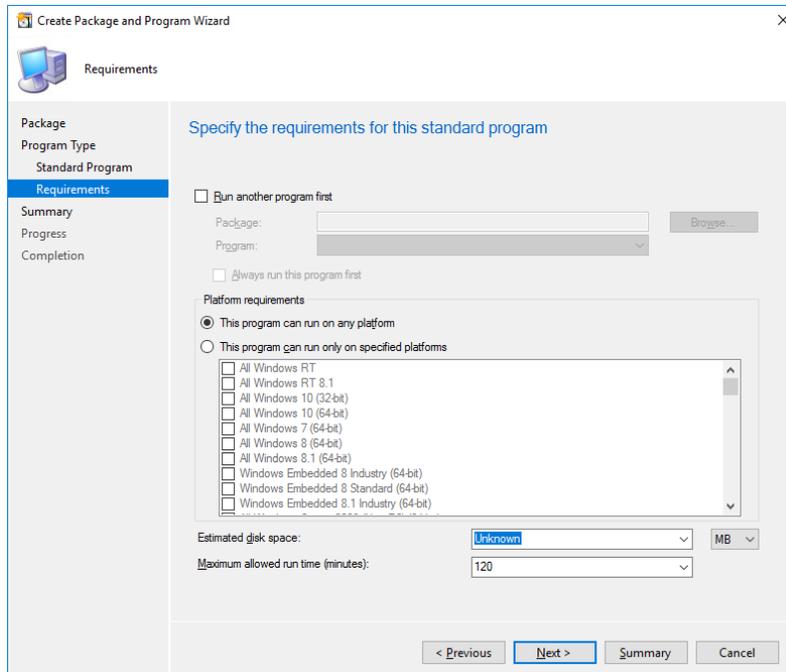
```
msiexec.exe /i ISScript9.Msi /qn
```

E. For the option *Program can run*, select *Whether or not a user is logged on*. Click **Next**.



The screenshot shows the 'Create Package and Program Wizard' dialog box, specifically the 'Standard Program' step. The left sidebar shows the progression: Package, Program Type, Standard Program, Requirements, Summary, Progress, and Completion. The main area is titled 'Specify information about this standard program'. The 'Name' field contains 'ISScript'. The 'Command line' field contains 'msiexec.exe /I ISScript9.Msi /qn'. The 'Startup folder' field is empty. The 'Run:' dropdown is set to 'Normal'. The 'Program can run:' dropdown is set to 'Whether or not a user is logged on', with a red arrow pointing to it. The 'Run mode:' dropdown is set to 'Run with administrative rights'. There are checkboxes for 'Allow users to view and interact with the program installation' (unchecked) and 'Reconnect to distribution point at log on' (unchecked). The 'Drive mode:' dropdown is set to 'Runs with UNC name'. At the bottom, there are buttons for '< Previous', 'Next >', 'Summary', and 'Cancel'.

F. There are no requirements for this package. Therefore, in this panel, click **Next**.



The screenshot shows the 'Create Package and Program Wizard' dialog box, specifically the 'Requirements' step. The left sidebar shows the progression: Package, Program Type, Standard Program, Requirements, Summary, Progress, and Completion. The main area is titled 'Specify the requirements for this standard program'. There are checkboxes for 'Run another program first' (unchecked) and 'Always run this program first' (unchecked). The 'Platform requirements' section is expanded, showing two radio buttons: 'This program can run on any platform' (selected) and 'This program can run only on specified platforms'. Below the second radio button is a list of platform options, all of which are unchecked: All Windows RT, All Windows RT 8.1, All Windows 10 (32-bit), All Windows 10 (64-bit), All Windows 7 (64-bit), All Windows 8 (64-bit), All Windows 8.1 (64-bit), Windows Embedded 8 Industry (64-bit), Windows Embedded 8 Standard (64-bit), and Windows Embedded 8.1 Industry (64-bit). At the bottom, there are dropdowns for 'Estimated disk space:' (set to 'Unknown') and 'Maximum allowed run time (minutes):' (set to '120'). At the bottom, there are buttons for '< Previous', 'Next >', 'Summary', and 'Cancel'.

G. Click the **Next** button until you have finished creating this package.

## 2. Create the Package for VS2005

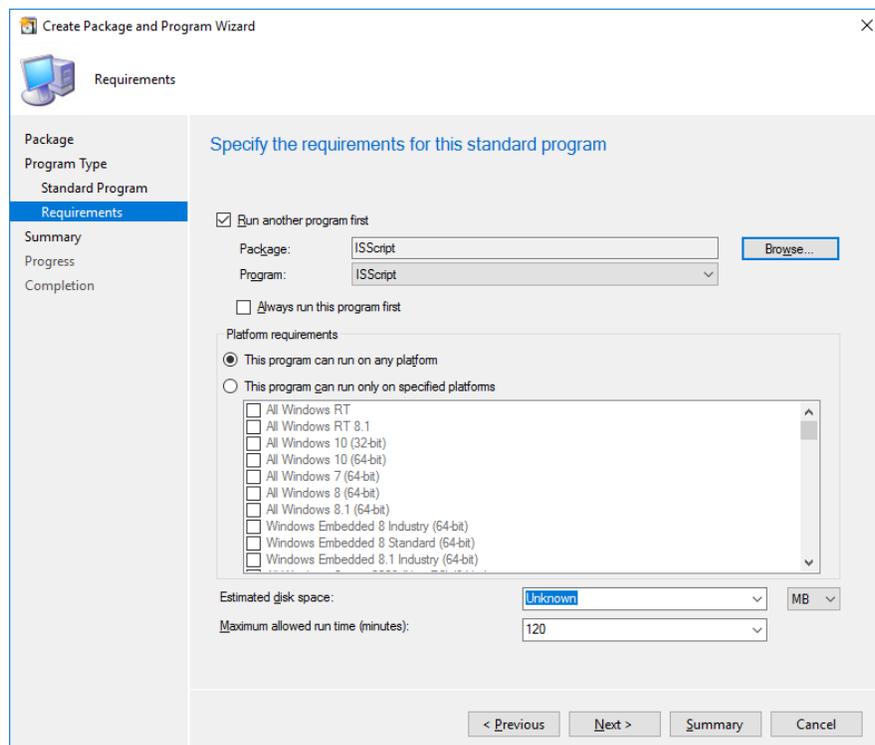
Follow the same procedures as in step 1 to create a new package *VS2005*:

- Select **Software Library > Overview > Application Management > Packages > Create Package**.
- Enter **VS2005** as the package name in the *Name* field. Check the option *This package contains source files*, and then specify *VS2005PIA* as the Source Folder location.
- In the next panel, choose *Standard program* and click **Next**.
- Enter the name **VS2005** in the Name field. Enter the following string into the Command Line field:

```
msiexec.exe /i VS_2005_PIA.msi /qn
```

Select *Whether or not a user is logged on* and click **Next**.

- Check the option *Run another program first* and select *ISScript*. Click **Next**.



- Click the **Next** button until you have finished creating this package.

## 3. Create the Package for UCCApi

- Select **Software Library > Overview > Application Management > Packages > Create Package**.
- Enter **UCCApi** as the package name in the *Name* field. Check the option *This package contains source files*, and specify *MicrosoftUCCAP* as the Source Folder location.
- In the next panel, choose *Standard program* and click **Next**.

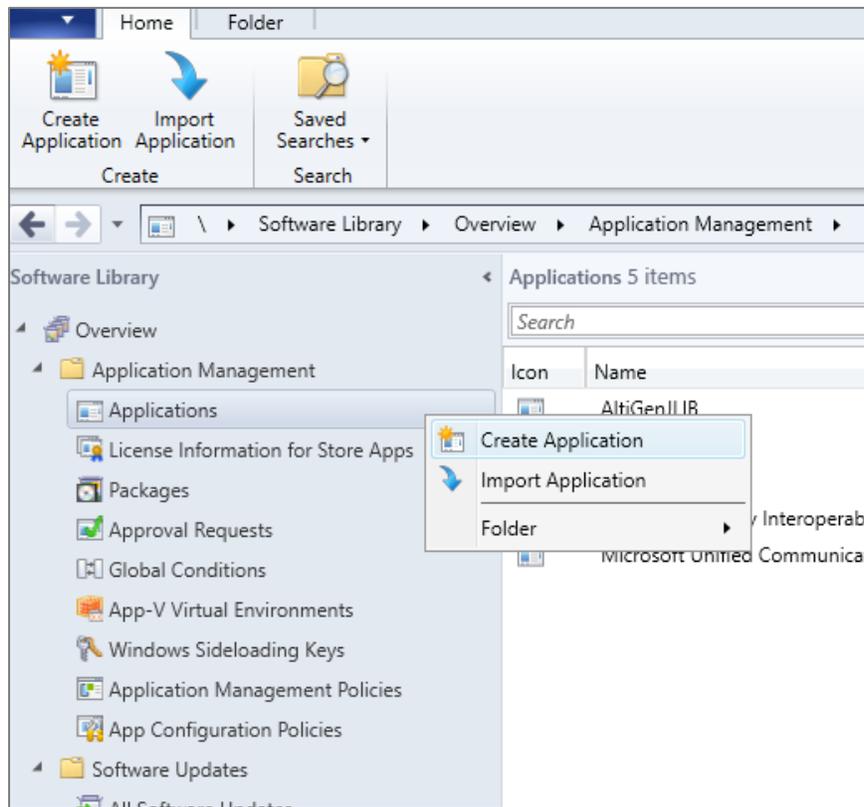
- D. Enter the name **UCCapi** in the Name field. Enter the following string into the Command line field:

```
msiexec.exe /i UccApiSdk.msi /qn
```

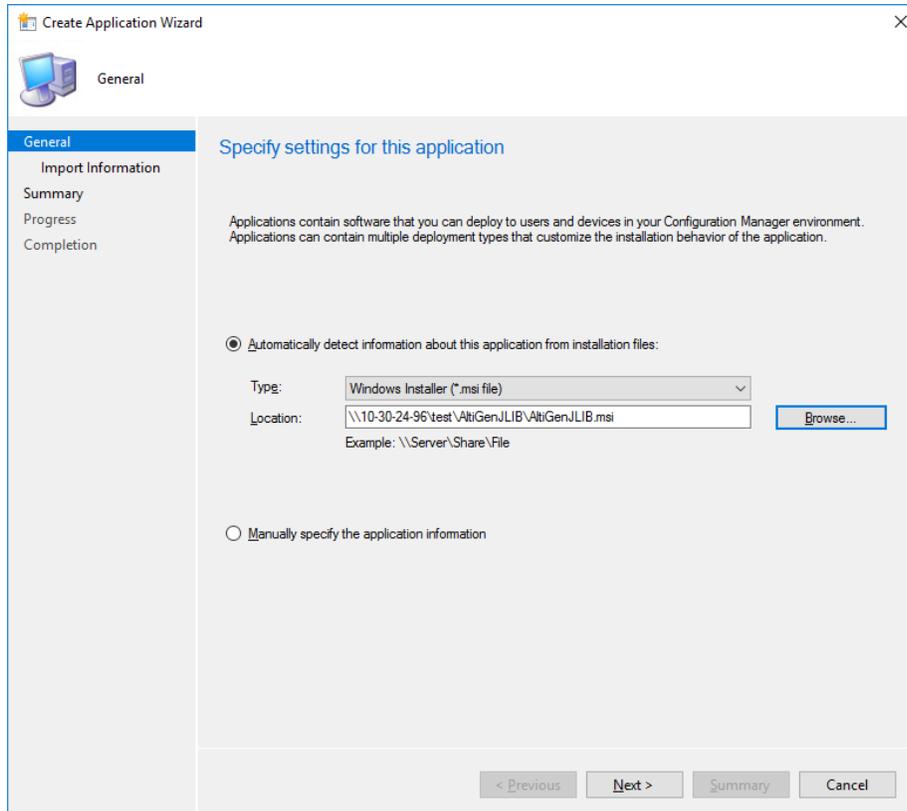
- E. Select *Whether or not a user is logged on* and click **Next**.  
F. Check the option *Run another program first* and select **VS2005**. Click **Next**  
G. Click the **Next** button until you have finished creating this package.

#### 4. Create the Application for AltigenJLIB

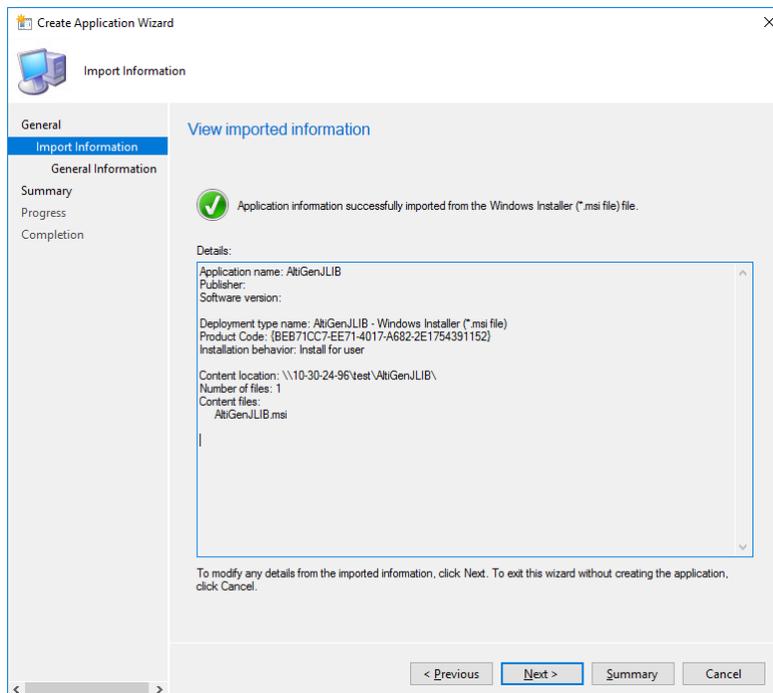
- A. Select **Software Library > Overview > Application Management > Applications > Create Application**.



- B. Enter the path of package **AltigenJLIB** as the location field for option “Automatically detect information about this application from installation files.”

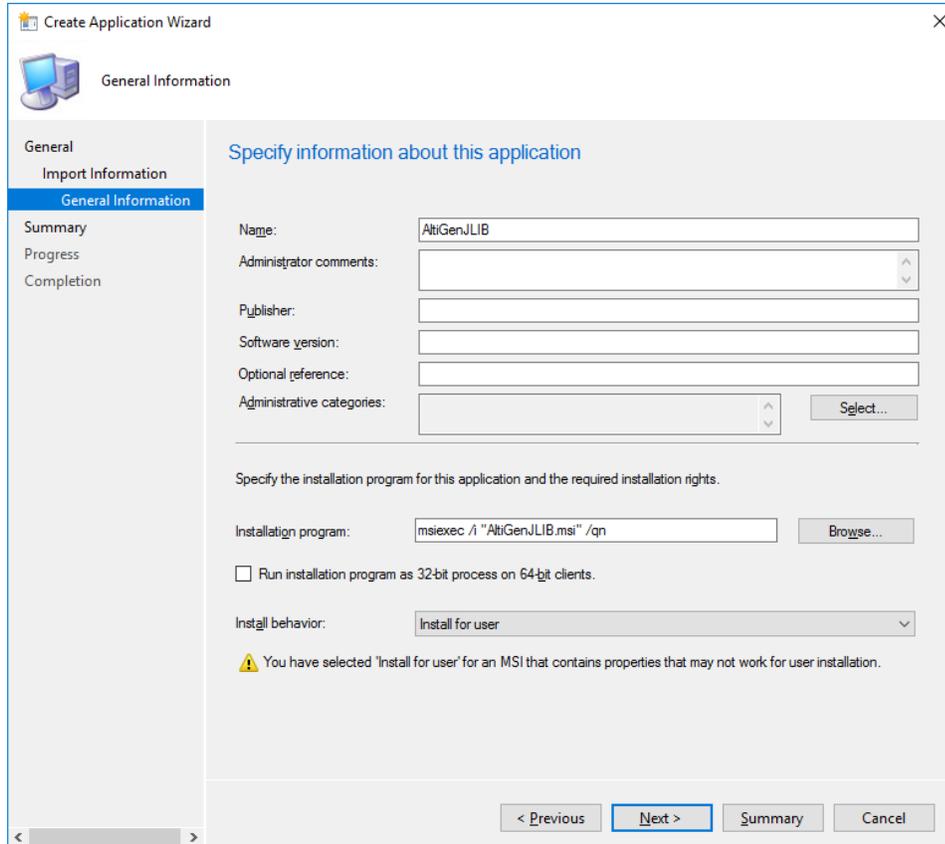


- C. In the next panel, you can see the imported information. Click **Next**.



- D. Enter the name **AltigenJLIB** in the *Name* field. Enter the following string into the *Installation Program* field:

```
msiexec.exe /i "AltigenJLIB.msi" /qn
```



Create Application Wizard

General Information

General Information

Specify information about this application

Name: AltigenJLIB

Administrator comments:

Publisher:

Software version:

Optional reference:

Administrative categories: Select...

Specify the installation program for this application and the required installation rights.

Installation program: msiexec /i "AltigenJLIB.msi" /qn Browse...

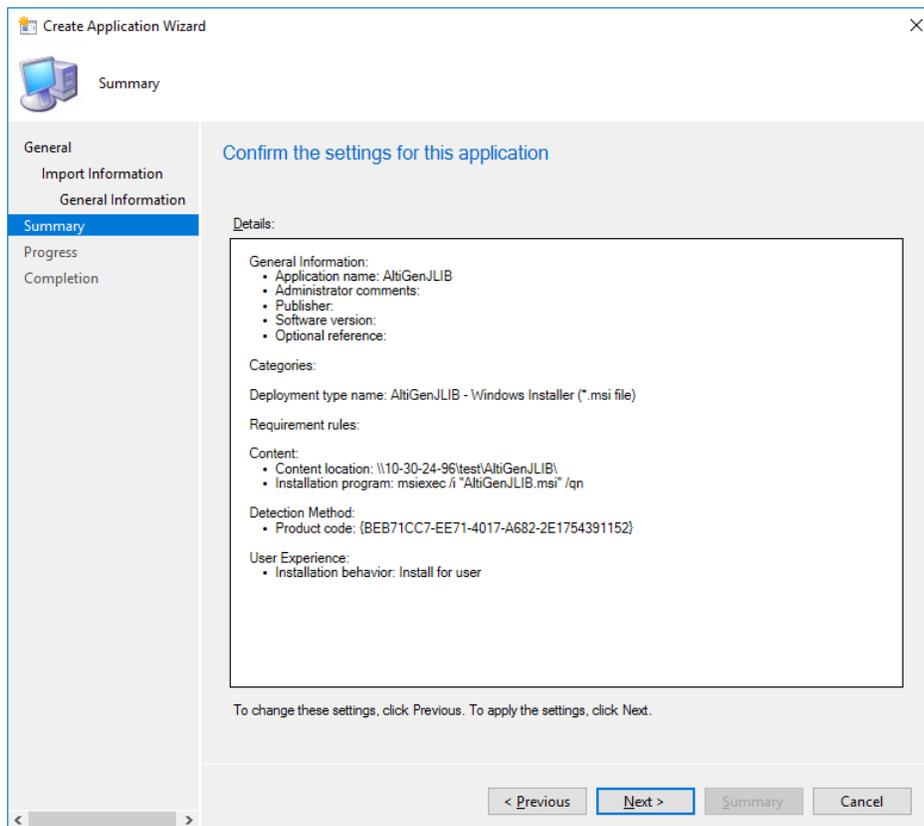
Run installation program as 32-bit process on 64-bit clients.

Install behavior: Install for user

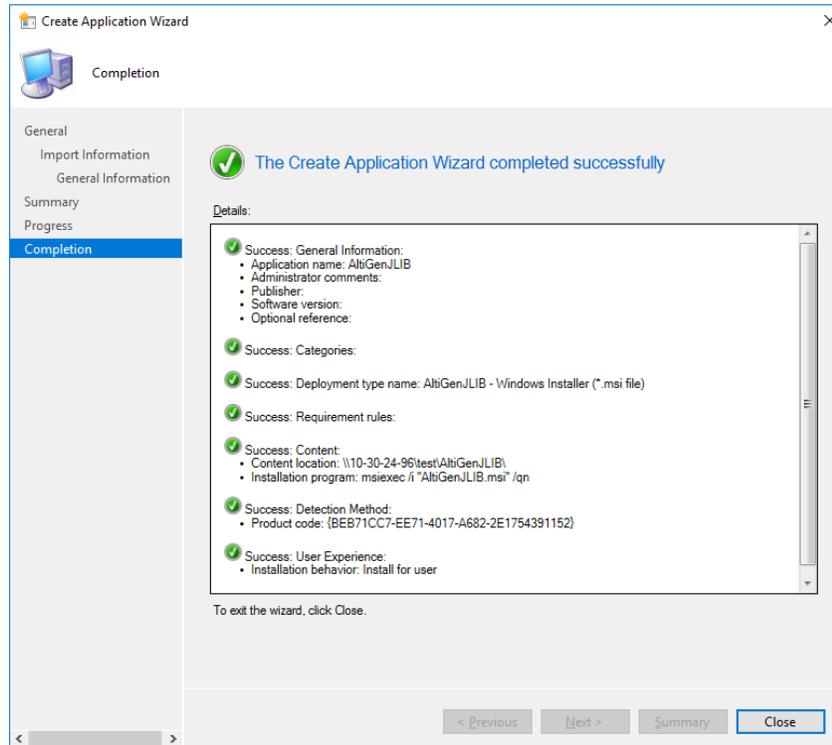
**⚠** You have selected 'Install for user' for an MSI that contains properties that may not work for user installation.

< Previous Next > Summary Cancel

E. In the summary panel, click **Next**.

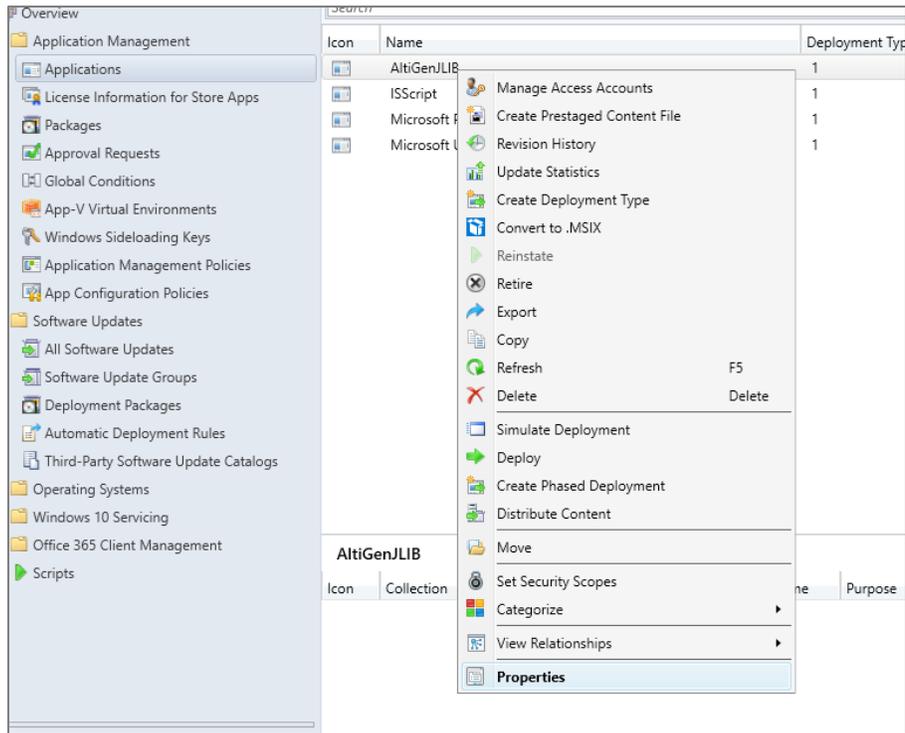


F. Click the **Next** button until you have finished creating this application.

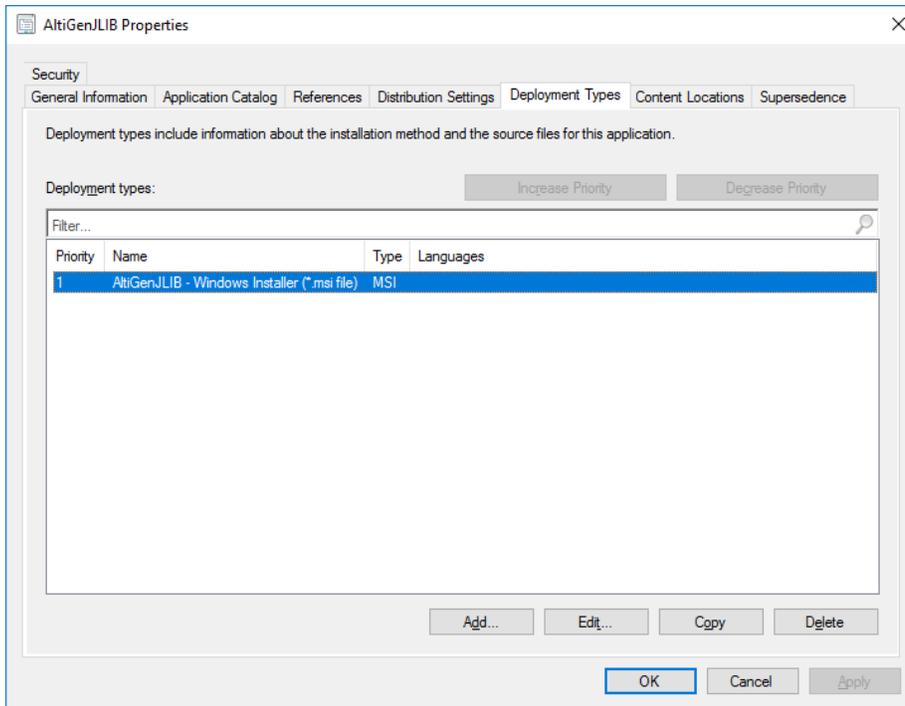


For some customers, SCCM may not be able to add the silent uninstall parameter for AltigenJlib. If this happens, verify the setting as following.

G. Right click the AltigenJlib application and choose **Properties**.

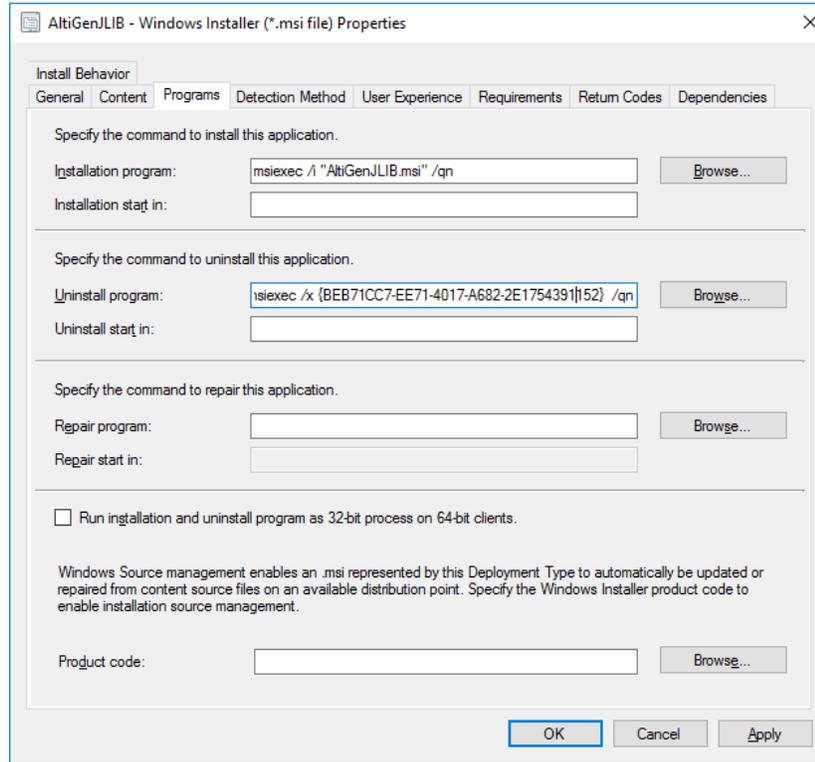


H. In the *Deployment types* panel, select AltigenJlib and click **Edit**.



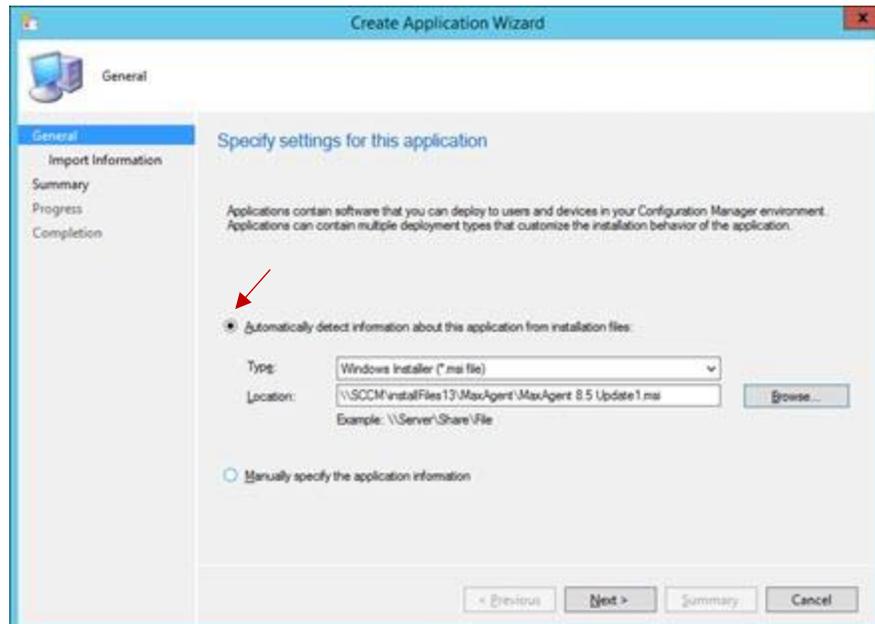
- I. In the AltigenJlib Properties panel, switch to the *Programs* tab.

Make sure the silent parameter “/qn” is there for both Installation program and Uninstallation program. (your product ID could be different from the mine in uninstallation program. It is automatically detected by SCCM). If it is not there you need manually enter them and click Apply.

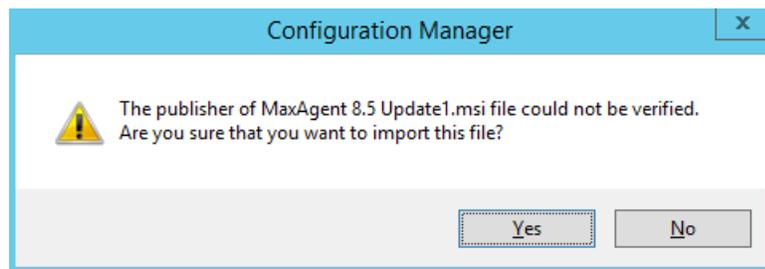


## 5. Create the Application for MaxAgent

- A. Select **Software Library > Overview > Application Management > Applications > Create Application**.
- B. Select the option *Automatically detect information about this application from installation files*, and browse to the .msi file for MaxAgent.



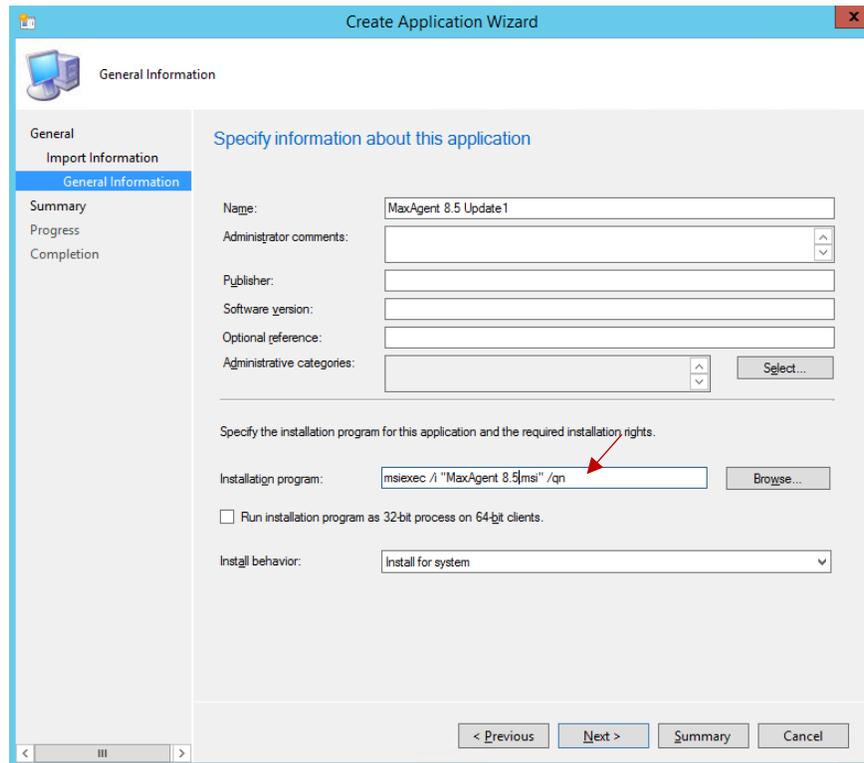
- C. When you see the popup window indicating that the publisher of the .msi file could not be verified, click **Yes**.



- D. The Name field should automatically populate. Type the following string into the *Installation program* field:

```
msiexec.exe /i "MaxAgent 8.5.msi" /qn
```

Click **Next**.



- E. Click the **Next** button until you have finished creating this package.

## 6. Create the Application for MaxSupervisor

- Select **Software Library > Overview > Application Management > Applications > Create Application**.
- Select the option *Automatically detect information about this application from installation files*, and browse to the .msi file for MaxSupervisor.
- When you see the popup window indicating that the publisher of the .msi file could not be verified, click **Yes**.
- The Name field should be automatically filled in. Type the following string into the *Installation program* field:

```
msiexec.exe /i "MaxSupervisor 8.5.msi" /qn
```

Click **Next**.

- E. Click the **Next** button until you have finished creating this package.

## 7. Create the Package for MaxCommunicator

- Select **Software Library > Overview > Application Management > Applications > Create Application**.
- Select the option *Automatically detect information about this application from installation files*, and browse to the .msi file for MaxCommunicator.

- C. When you see the popup window indicating that the publisher of the .msi file could not be verified, click **Yes**.
- D. The Name field should be automatically filled in. Type the following string into the *Installation program* field:

```
msiexec.exe /i "MaxCommunicator 8.5.msi" /qn
```

Click **Next**.

- E. Click the **Next** button until you have finished creating this package.

## Step 3: Deploy Applications to the Client Systems

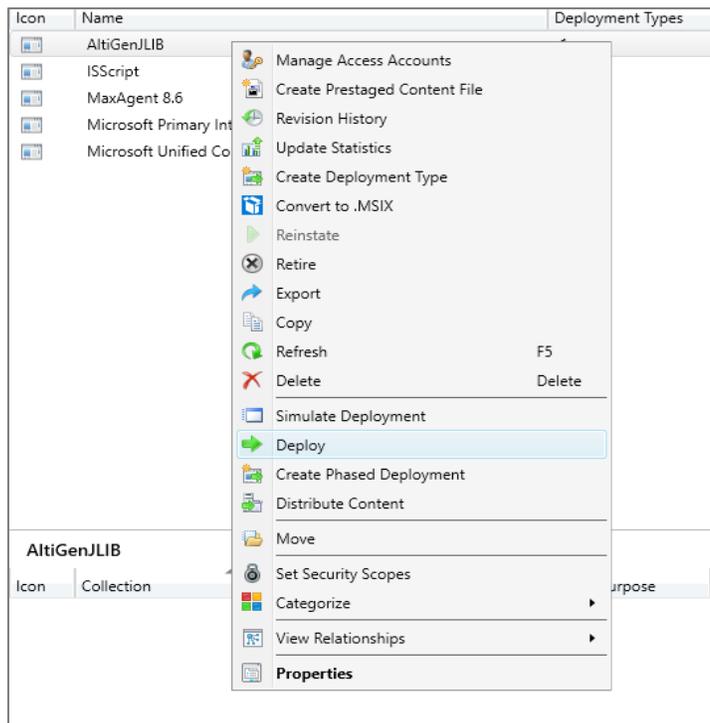
### Fresh Installations

Once you have created all of the applications, you are ready to deploy the applications to the client systems. Deploy packages in the following order:

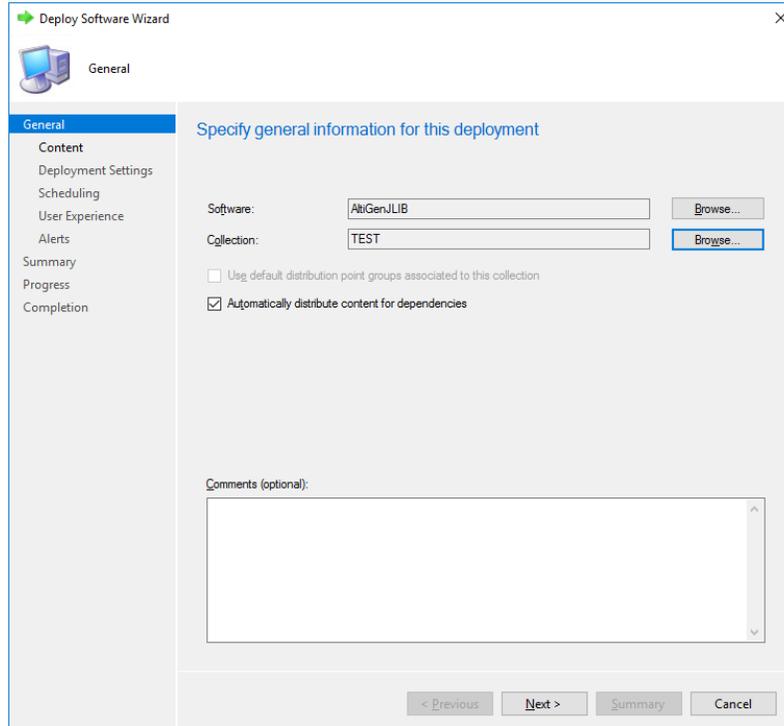
1. Deploy ISScript, VS2005, UCCApi and AltijLIB packages to the MaxAgent, MaxSupervisor, and MaxCommunicator containers that you created in the section [Before you Begin](#).
2. Deploy MaxAgent application to the MaxAgent container after the first step is done.
3. Deploy the MaxSupervisor application to the MaxSupervisor container.
4. Deploy MaxCommunicator application to the MaxCommunicator container.

Here is the one example for deploying AltigenJlib.

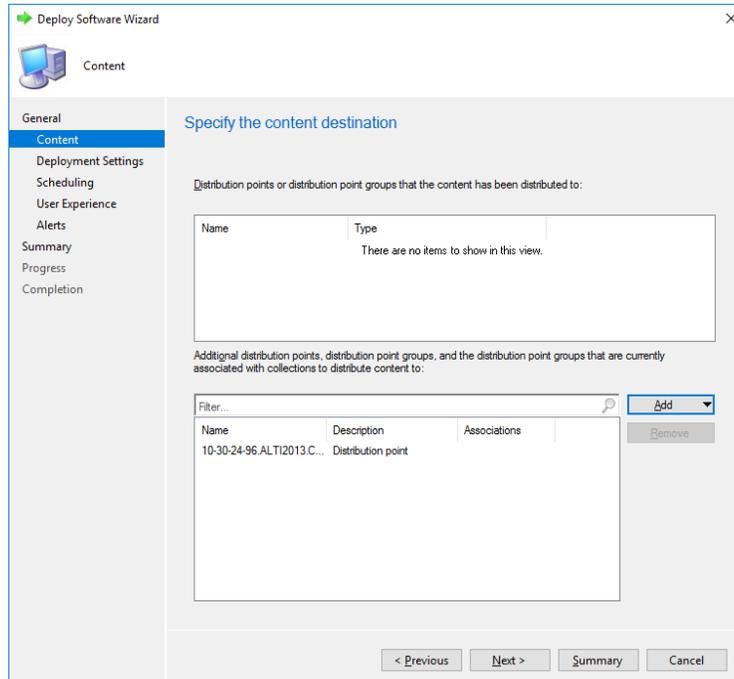
- A. Right click AltigenJlib -> Deploy



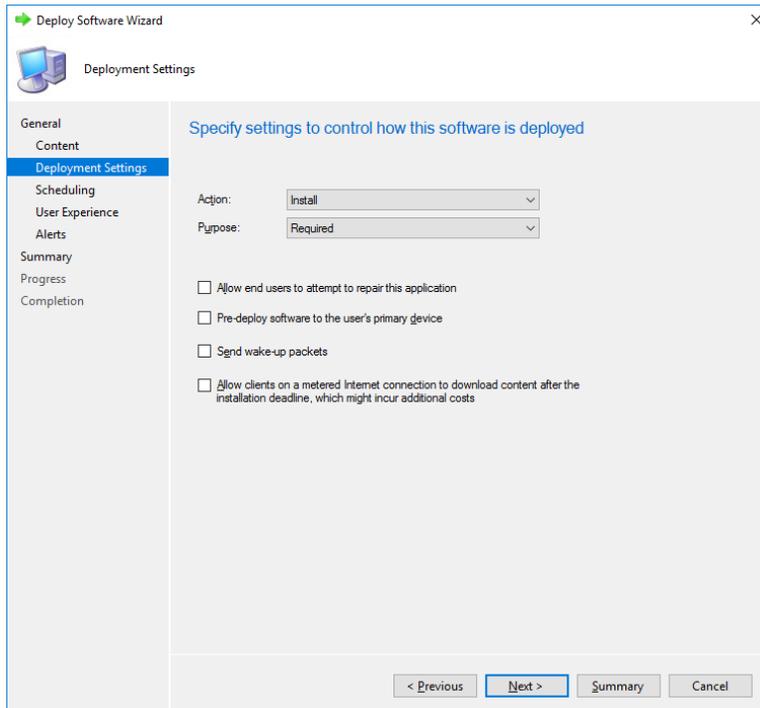
B. Choose your Collection Target machines.



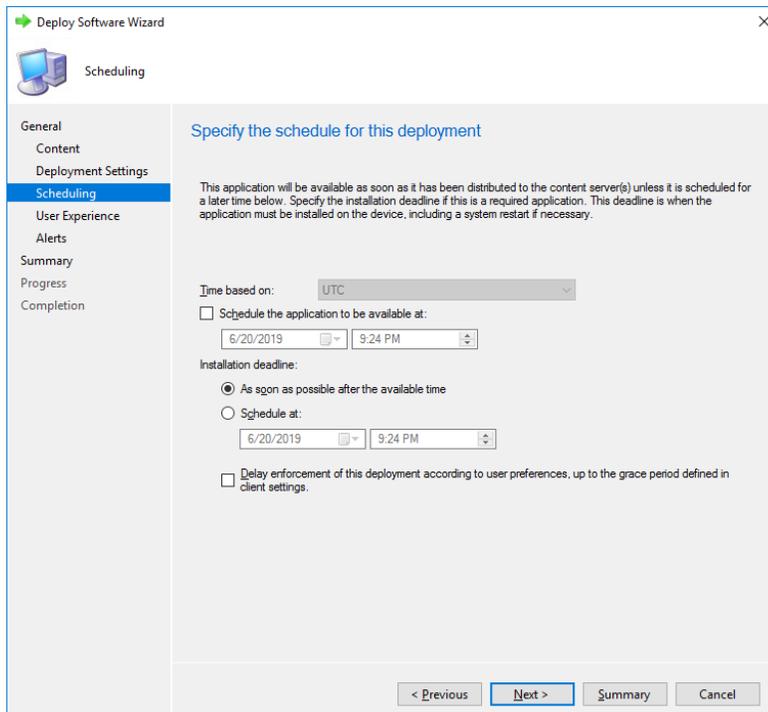
C. Choose your content destination.



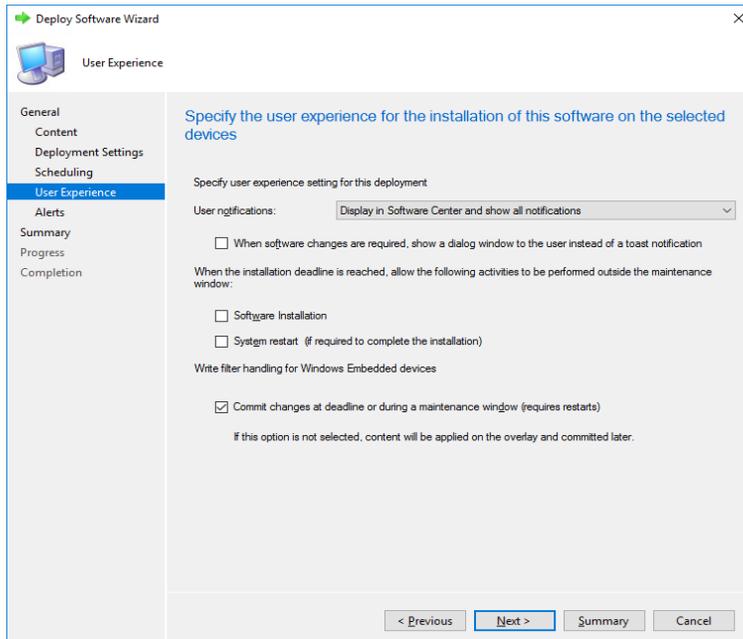
D. Specify how the software is deployed.



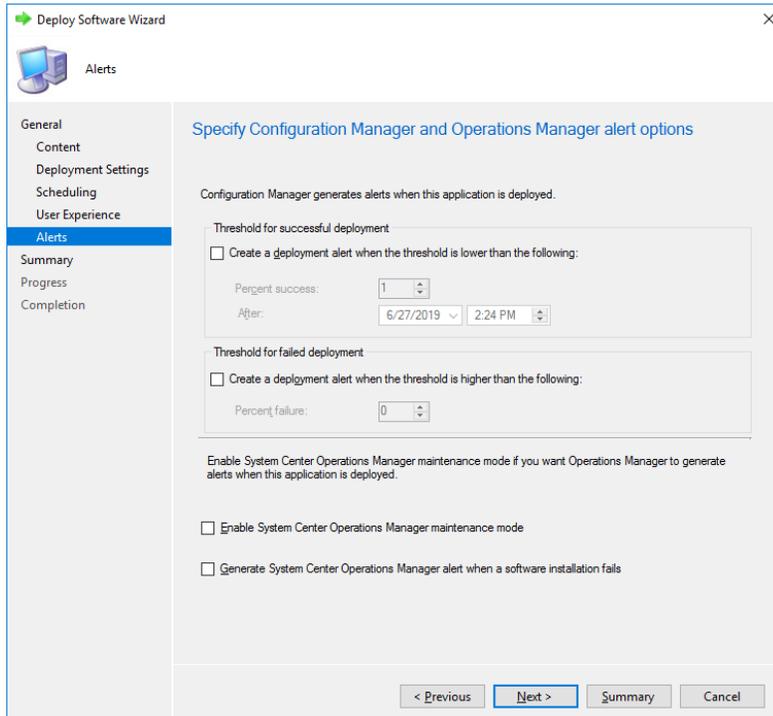
E. Specify a schedule for the deployment.



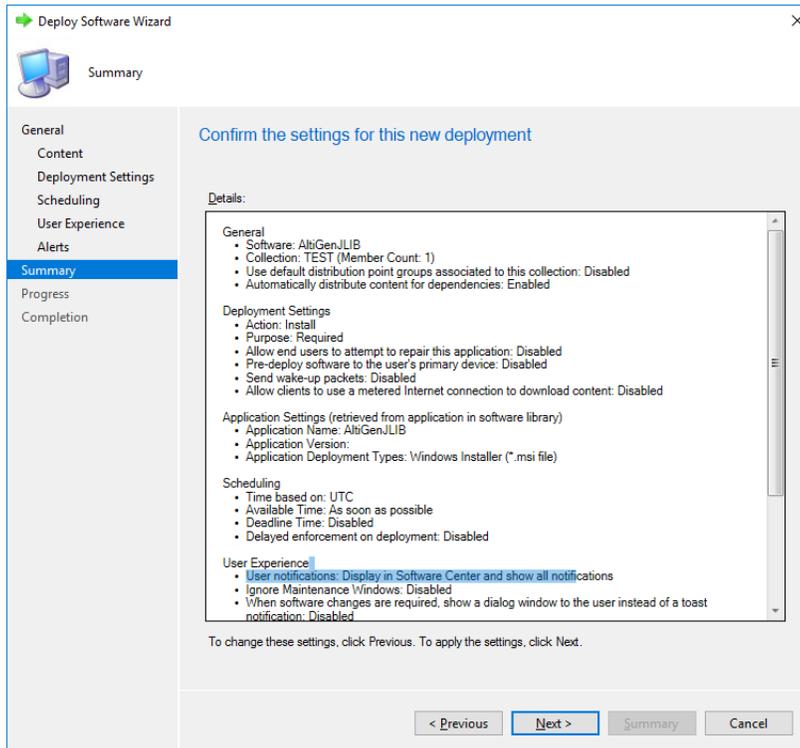
F. Specify the user experience.



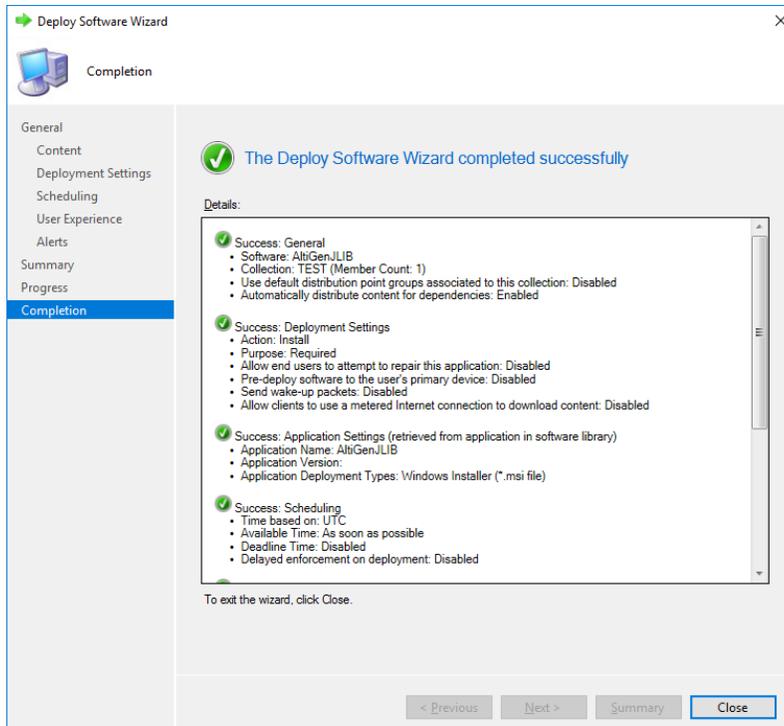
G. Specify Configuration Manager and Operations Manager alert options.



H. Confirm the settings.



I. The Deployment should complete successfully.



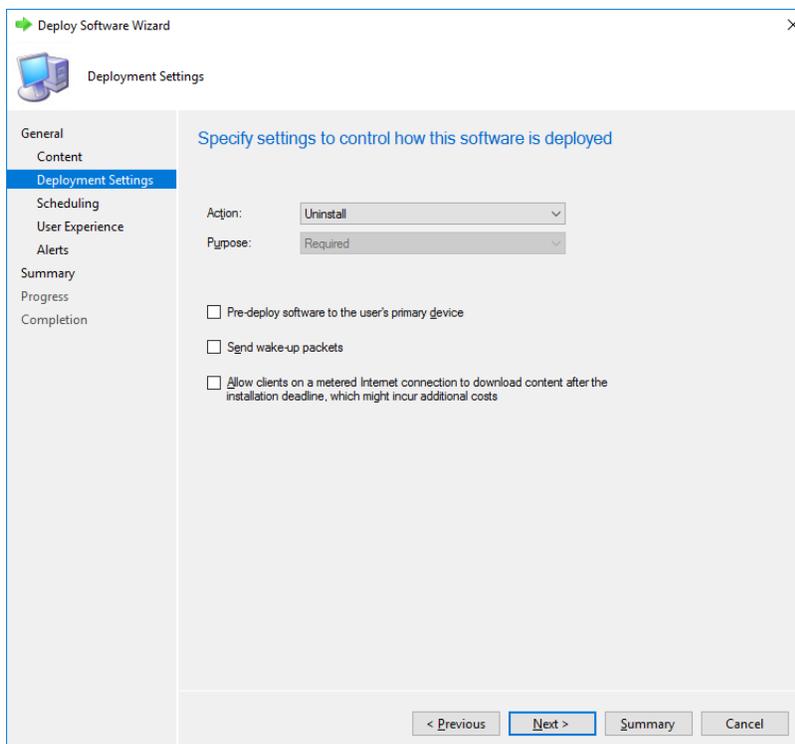
## Upgrade Installations

Usually Altigen users use the auto-upgrade service to do the upgrade. But if user want to use SCCM they also can do the same as following steps.

1. Uninstall AltigenJlib from SCCM.
2. Uninstall MaxAgent (or MaxCommunicator , MaxSupervisor) from SCCM.
3. Install the new version AltigenJlib.
4. Install the new version of MaxAgent (or MaxCommunicator, or MaxSupervisor).

### Example: How to Uninstall AltigenJlib

All the steps are the same as deploy “AltigenJlib” steps, except in the section “Specify setting to control how this software is deployed,” choose “Uninstall.”



## Known Issues

Some customers may have difficulty uninstalling AltigenJlib. If this is the case, you can create batch files and deploy them to client machines to run it. The content of the batch file are just one command:

```
msiexec.exe /X{GUID} /qn
```

For example:

```
msiexec.exe /X{BEB71CC7-EE71-4017-A682-2E1754391152} /qn
```

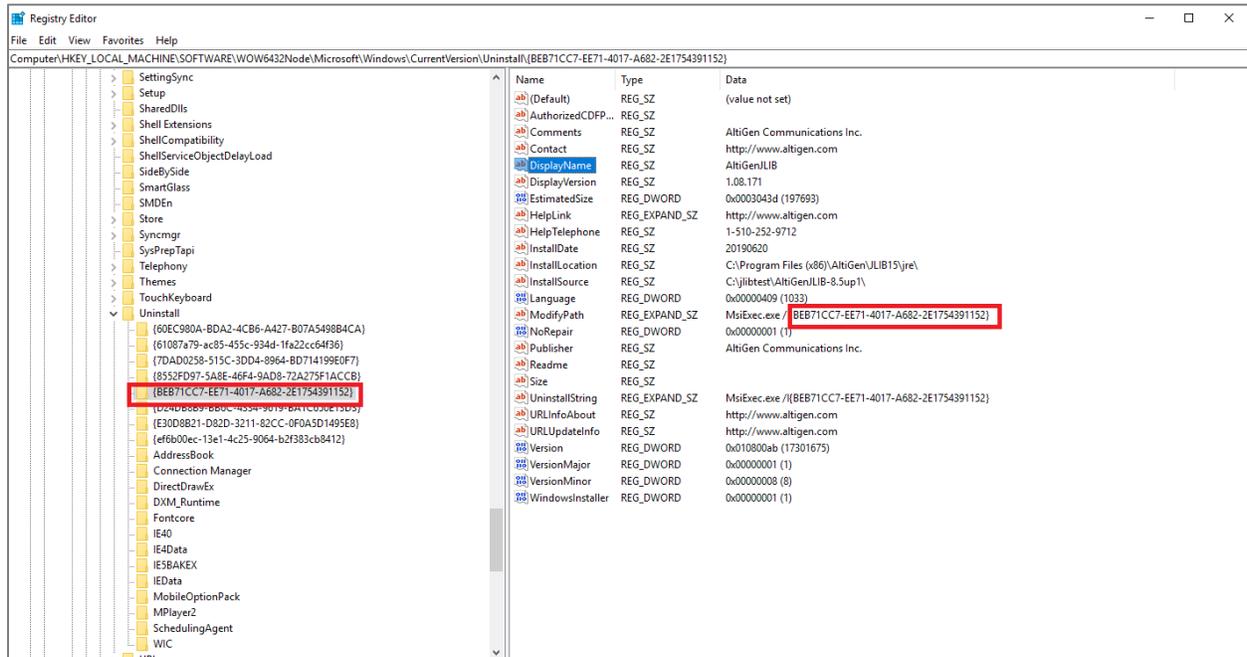
Where: {BEB71CC7-EE71-4017-A682-2E1754391152} is the GUID of the AltigenJlib installation.

There are 2 ways to identify the GUID of AltigenJlib.

### Method 1:

On a system which already has AltigenJlib installed, you can search for the keyword “AltigenJlib” in the registry path.

HKEY\_LOCAL\_MACHINE\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Uninstall\



### Method 2:

From the SCCM server,

1. Select **Software Library > Overview > Application Management > Applications > Create Application.**
2. Select the option *Automatically detect information about this application from installation files*, and browse to the .msi file for AltigenJlib. Click **Next.**

