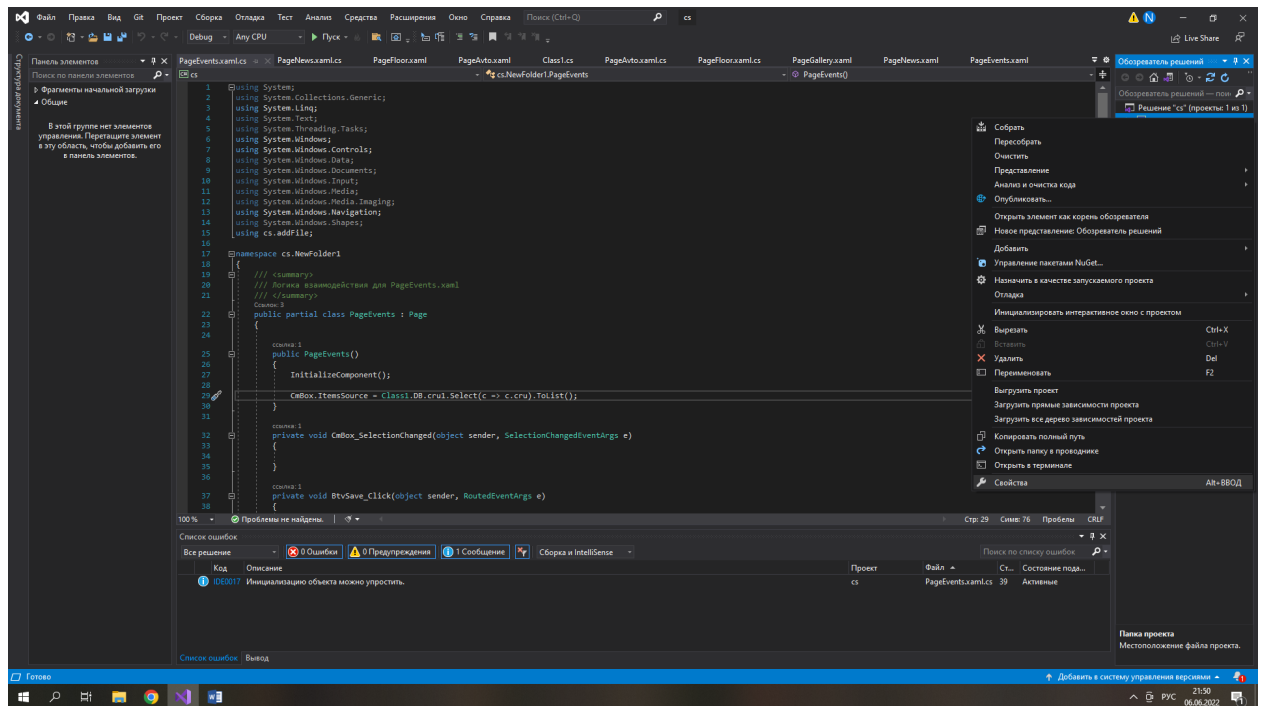
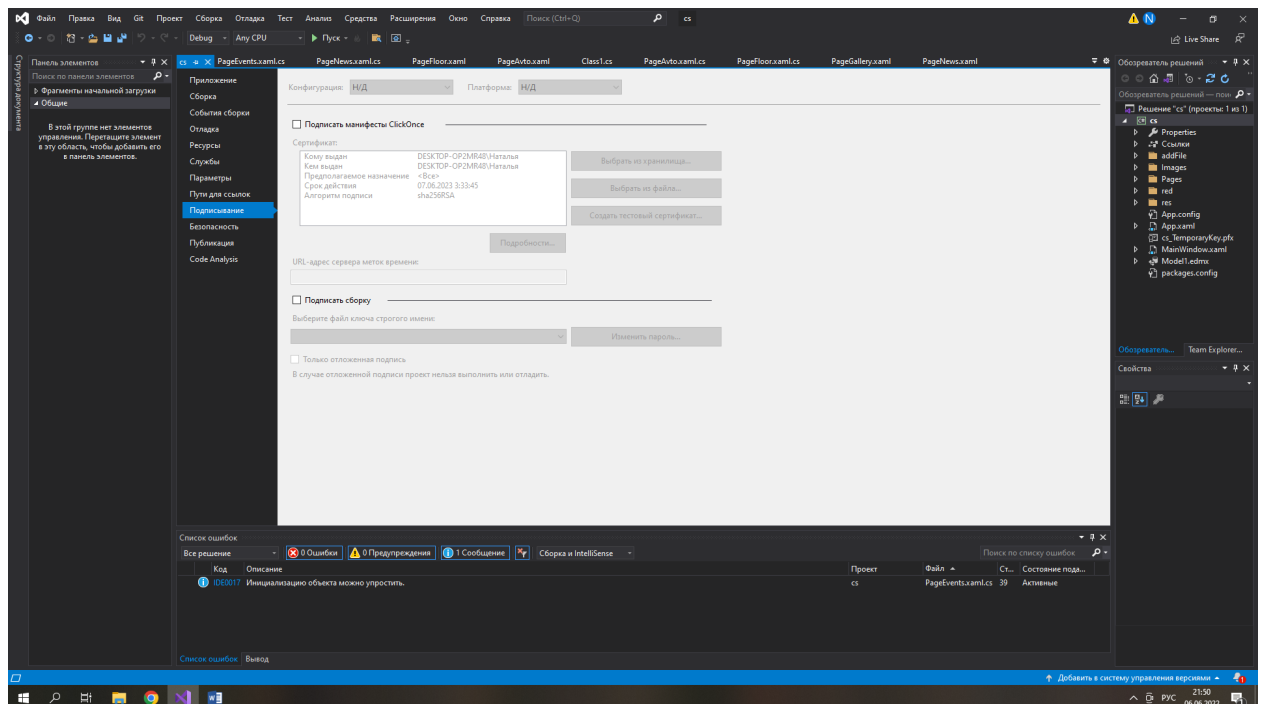


Инструкция по созданию инсталлятора с помощью встроенной технологии ClickOnce

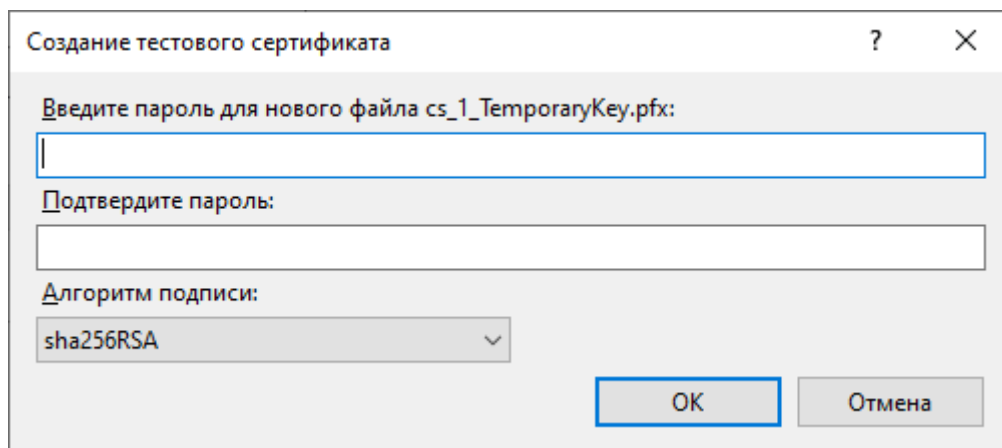
1. Нажимаем правой кнопкой мыши на название проекта в Обозревателе решений, находим «Свойства».



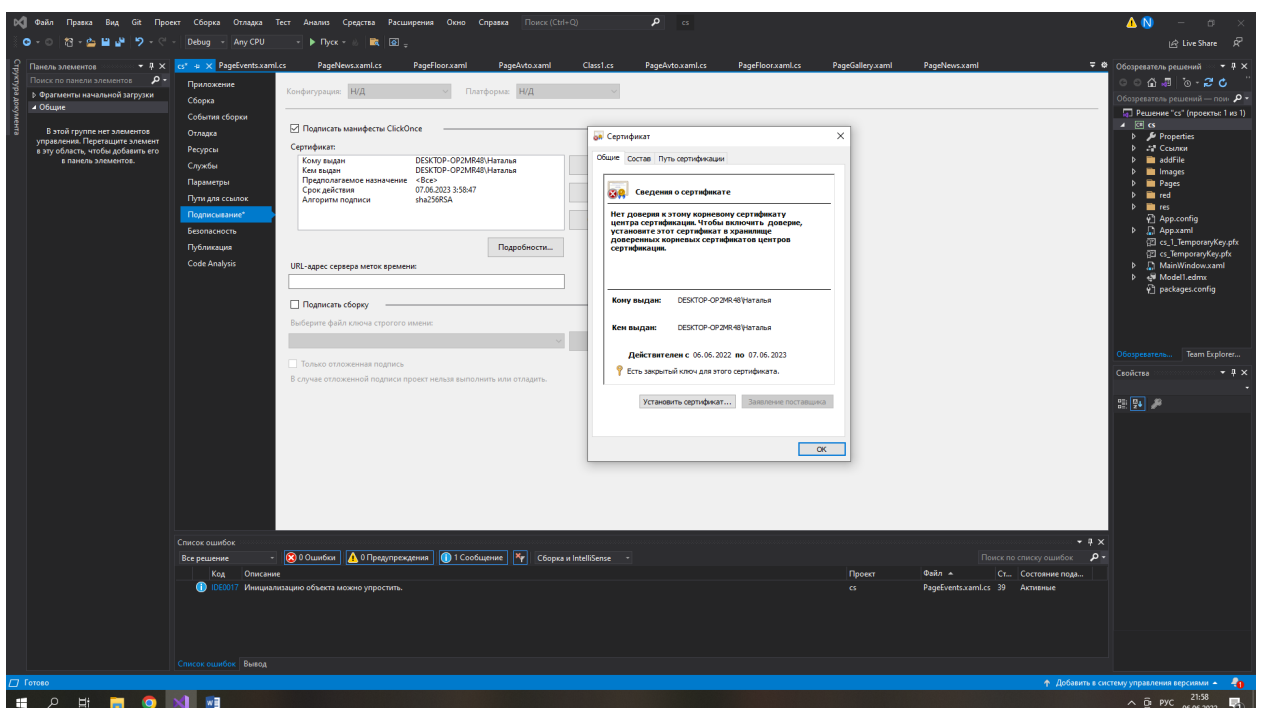
2. В открывшемся окне выбираем раздел «Подписывание» (Signing)



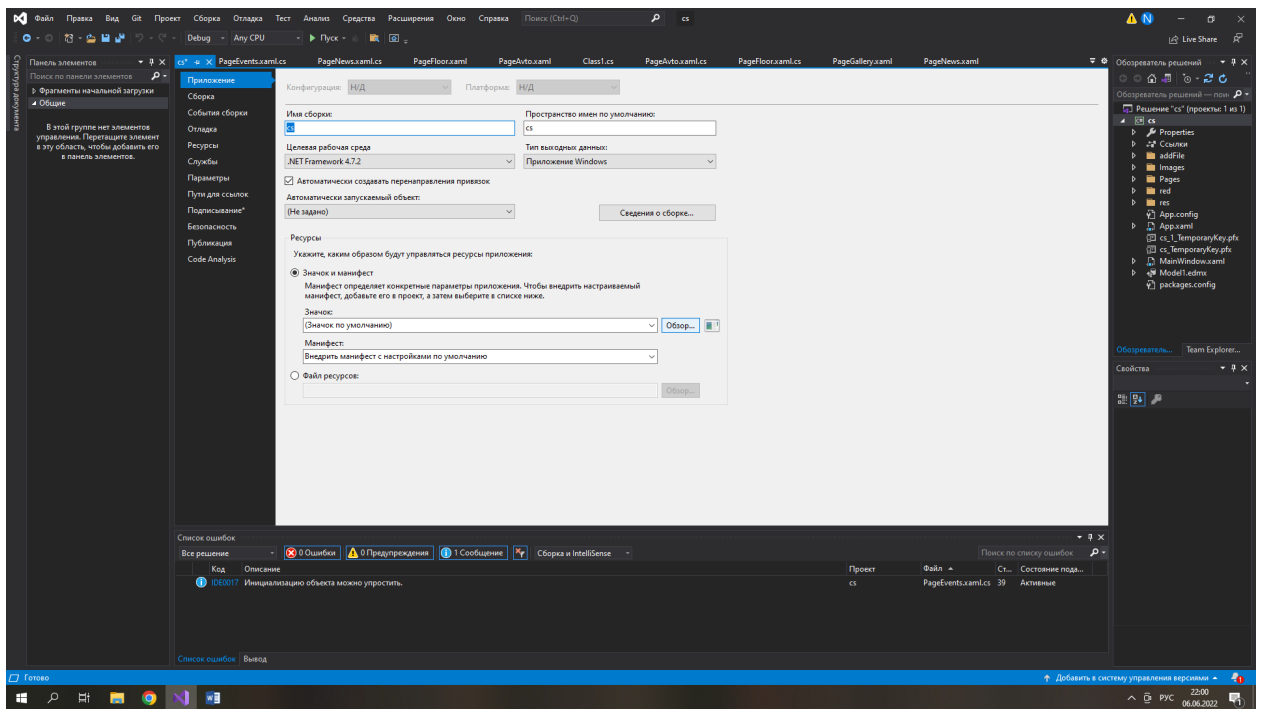
3. Ставим галочку в пункте «Подписать манифест ClickOnce». В этом же пункте меню есть возможность создать тестовый сертификат с паролем.



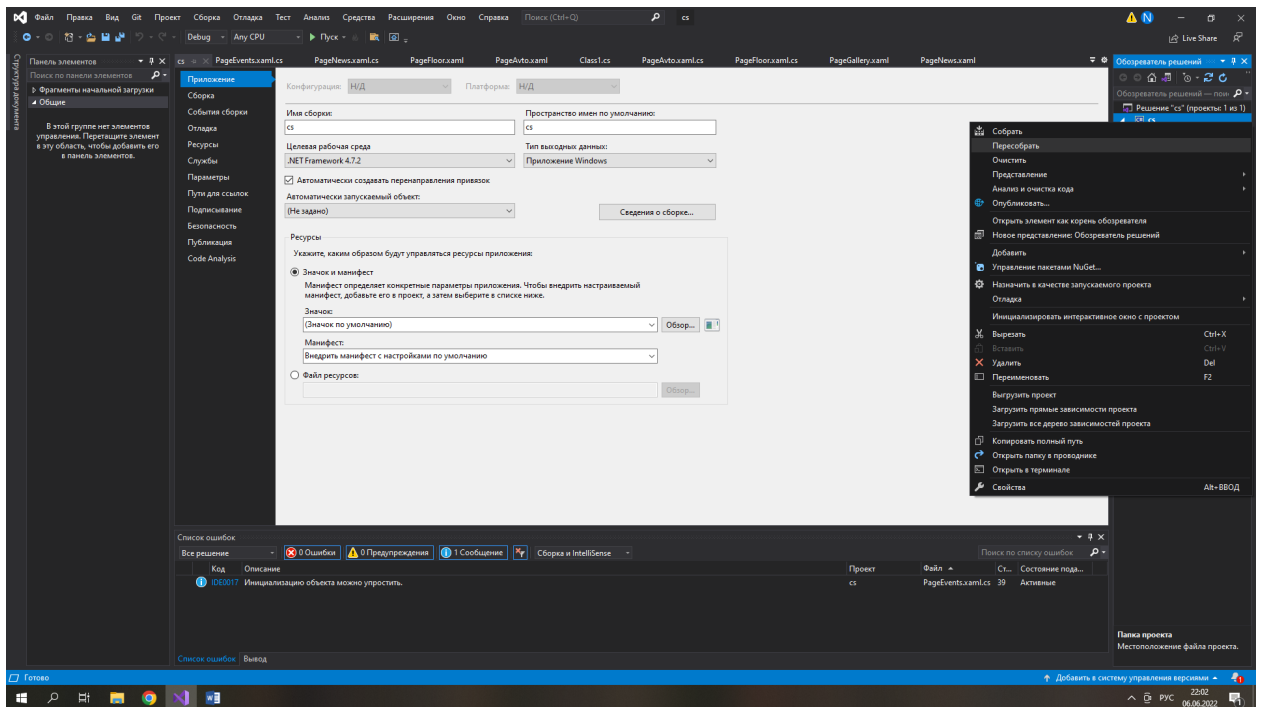
При нажатии на кнопку «Подробности» может посмотреть свойства сертификата и срок его действия.



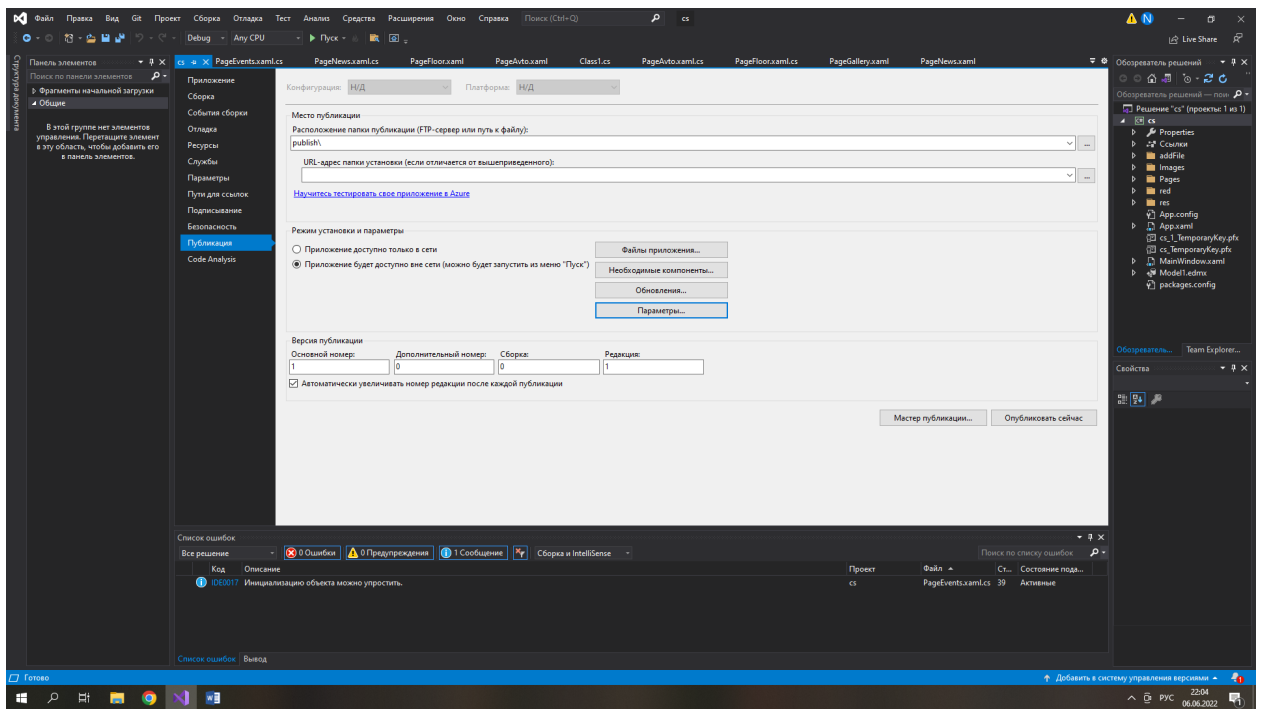
4. На вкладке «Приложение» можно установить иконку приложения:



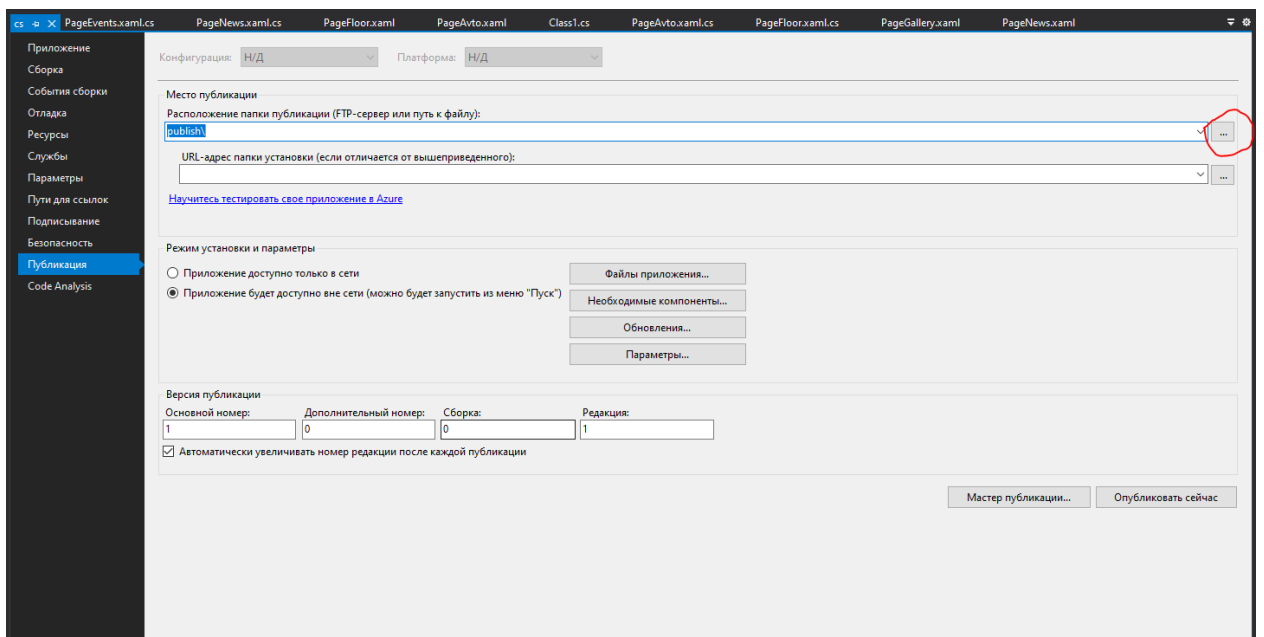
5. Затем следует сохранить изменения и пересобрать решение, нажав правой кнопкой мыши на название вашего решения и выбрав пункт «Пересобрать».



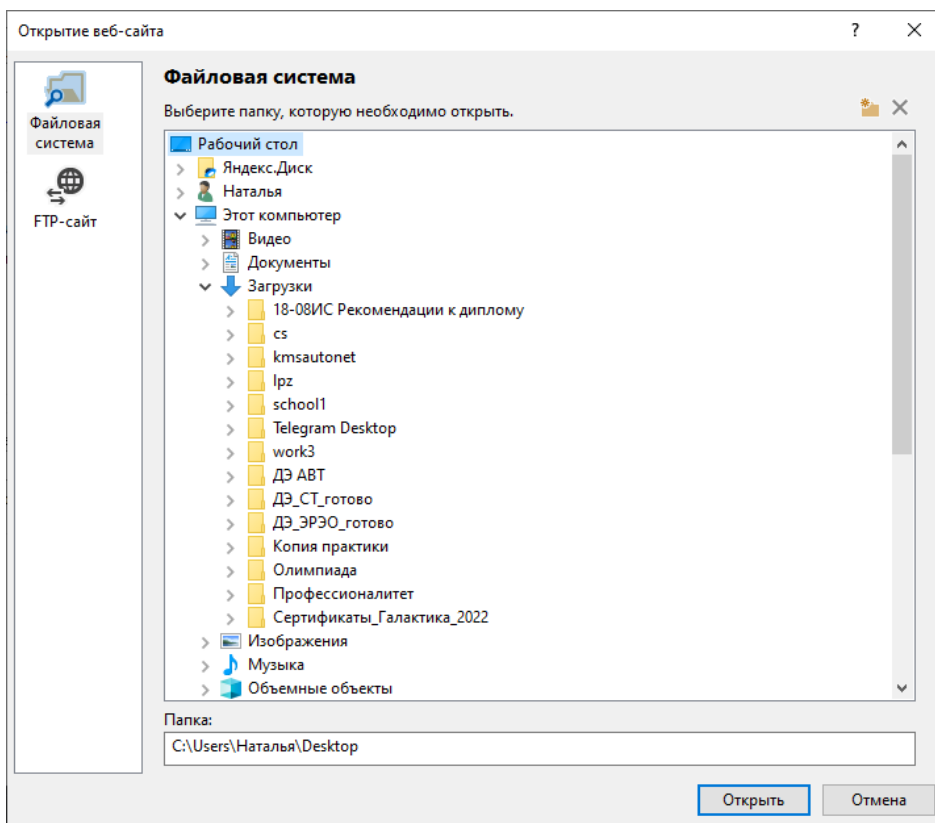
6. Затем в «Свойствах» переходим в пункт меню «Публикация».



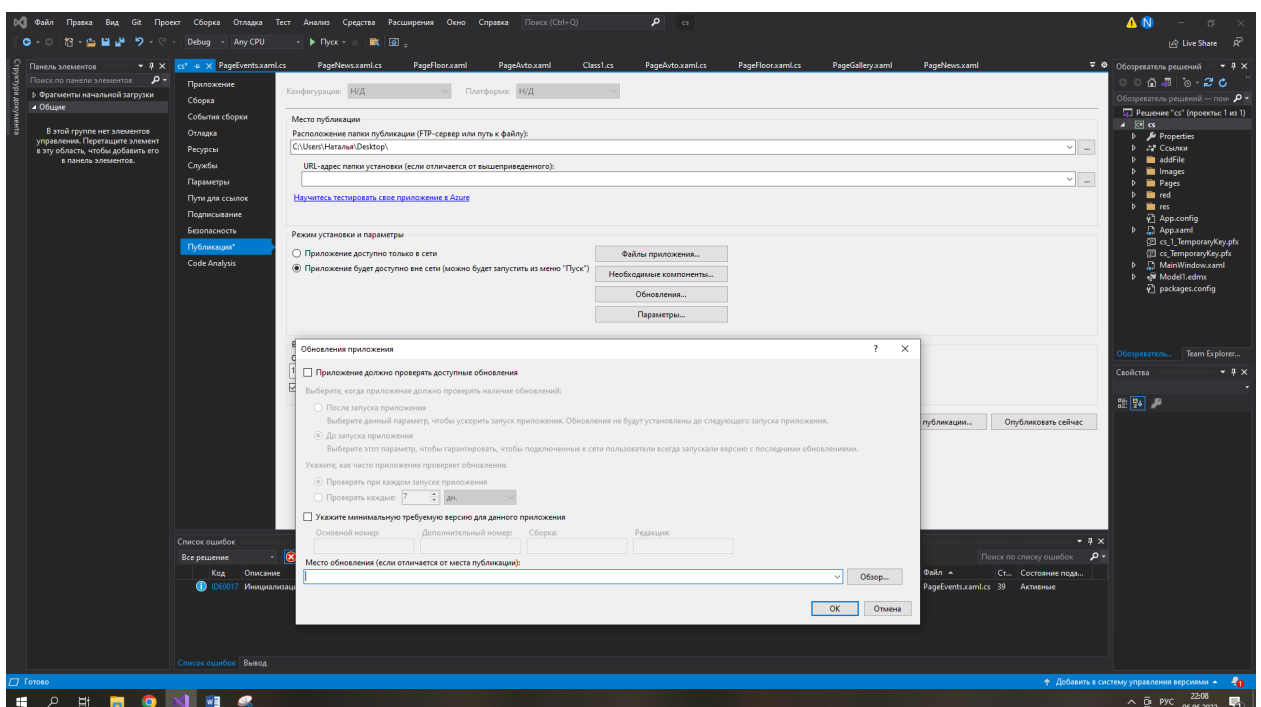
7. В открывшемся окне можно выбрать обзор «Расположения папки публикации»:



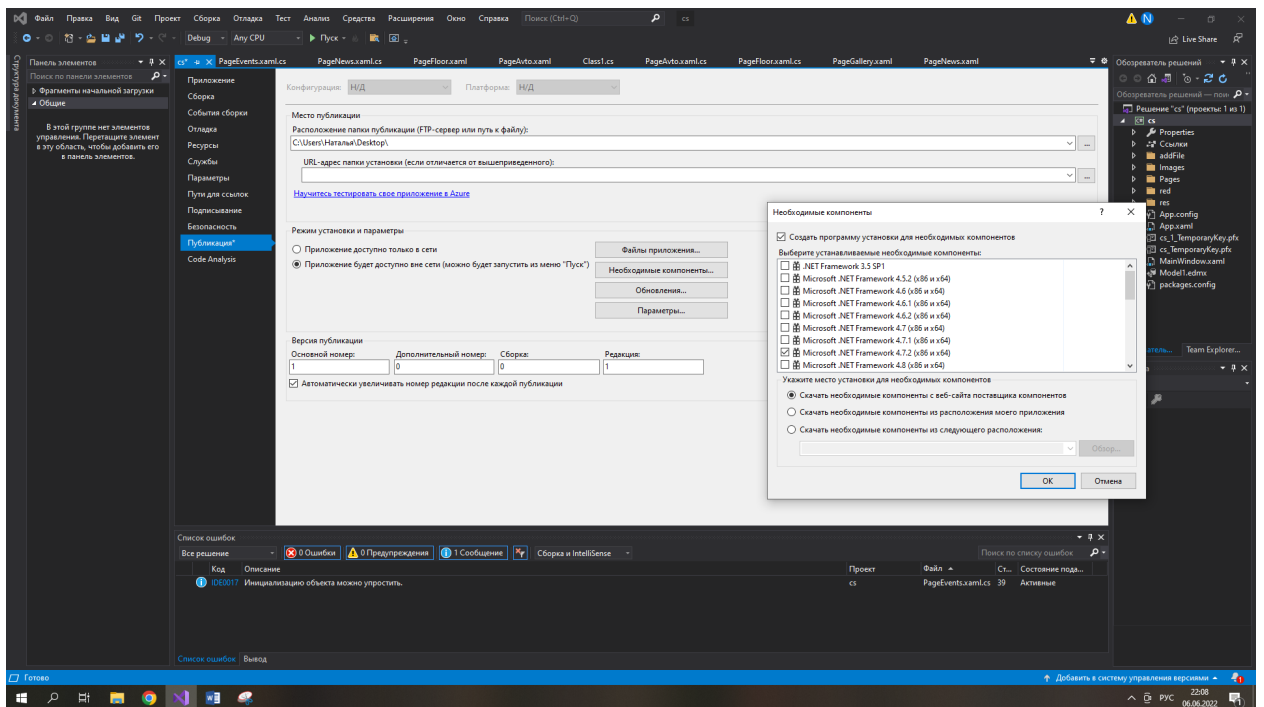
И в открывшемся окне выбрать папку на компьютере или указать адрес файлового сервера.



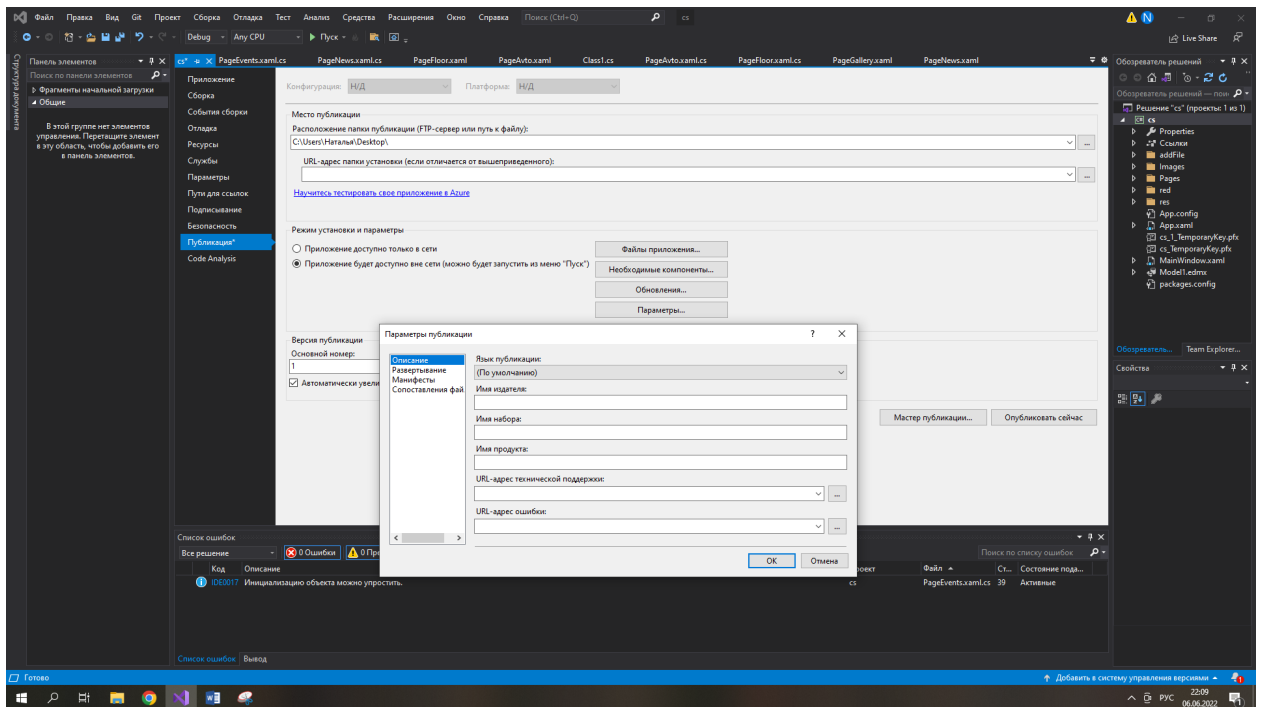
8. При нажатии на кнопку «Обновление» можно указать адрес сайта/сервера, на котором будут располагаться обновления программы.



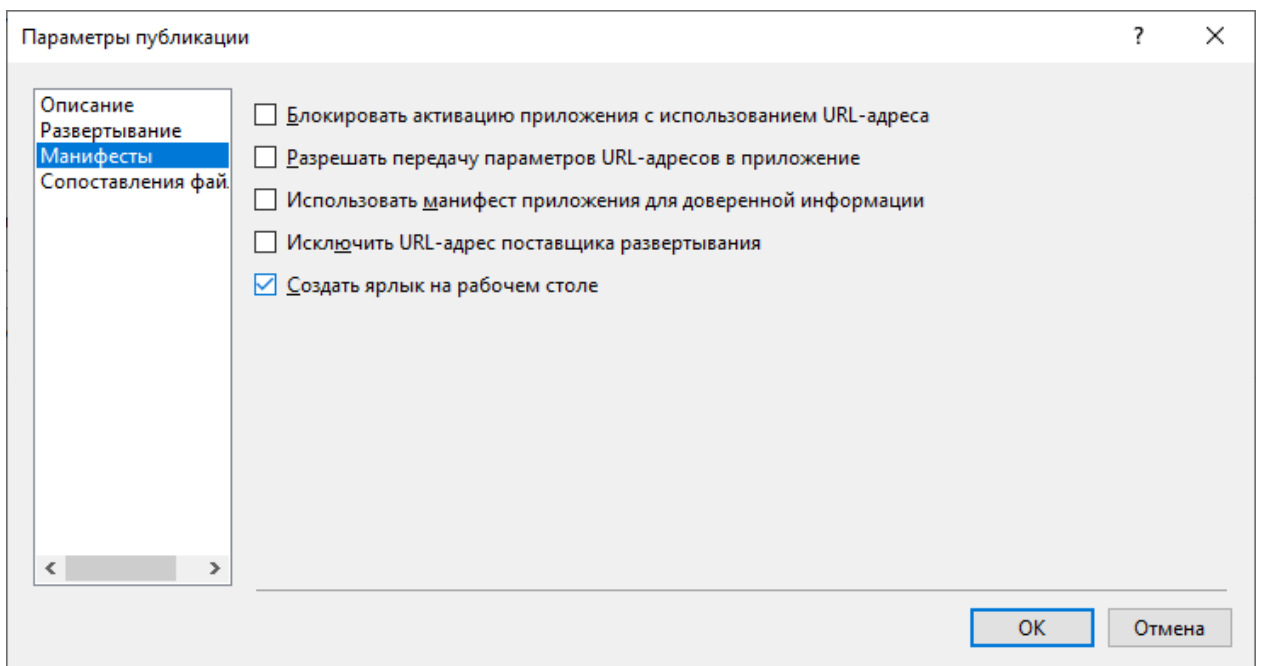
9. В окне «Необходимые компоненты» можно посмотреть или отметить те компоненты, которые должны быть установлены на компьютере для корректной работы программы:



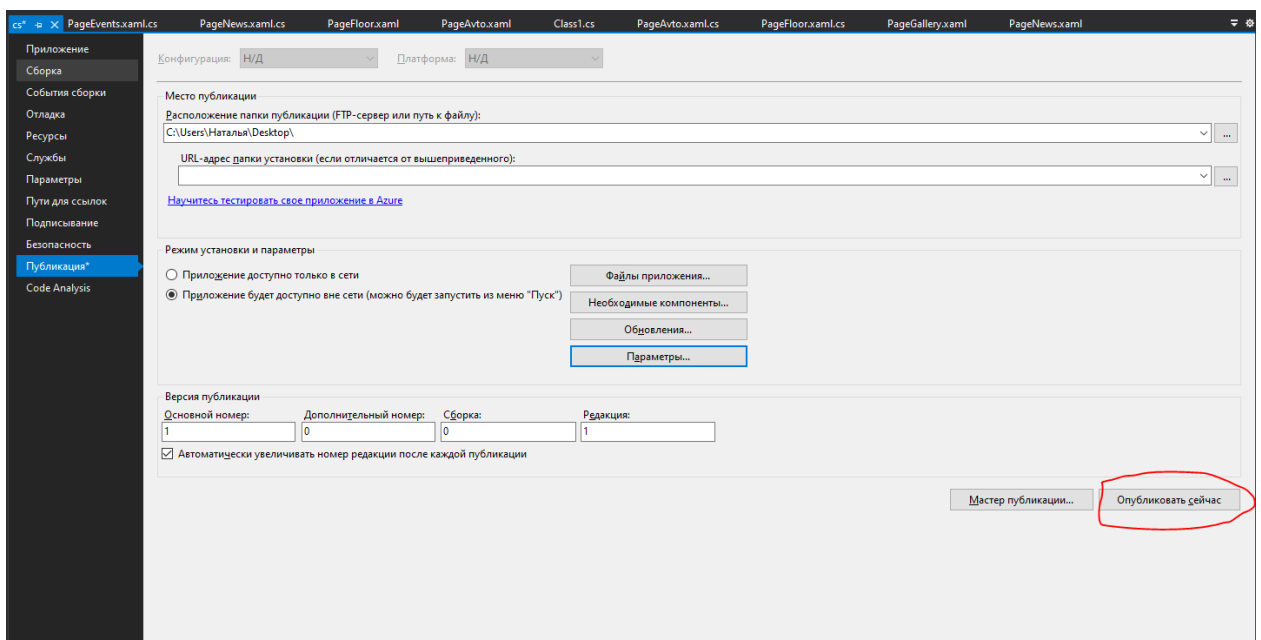
10. В окне «Параметры» необходимо указать Имя издателя, Имя набора и Имя продукта. Все эти параметры повлияют на адрес установки программы в меню Пуск.



11. Затем перейдите на вкладку «Манифест» и в открывшемся окне поставьте галочку в пункте «Создать ярлык на рабочем столе».

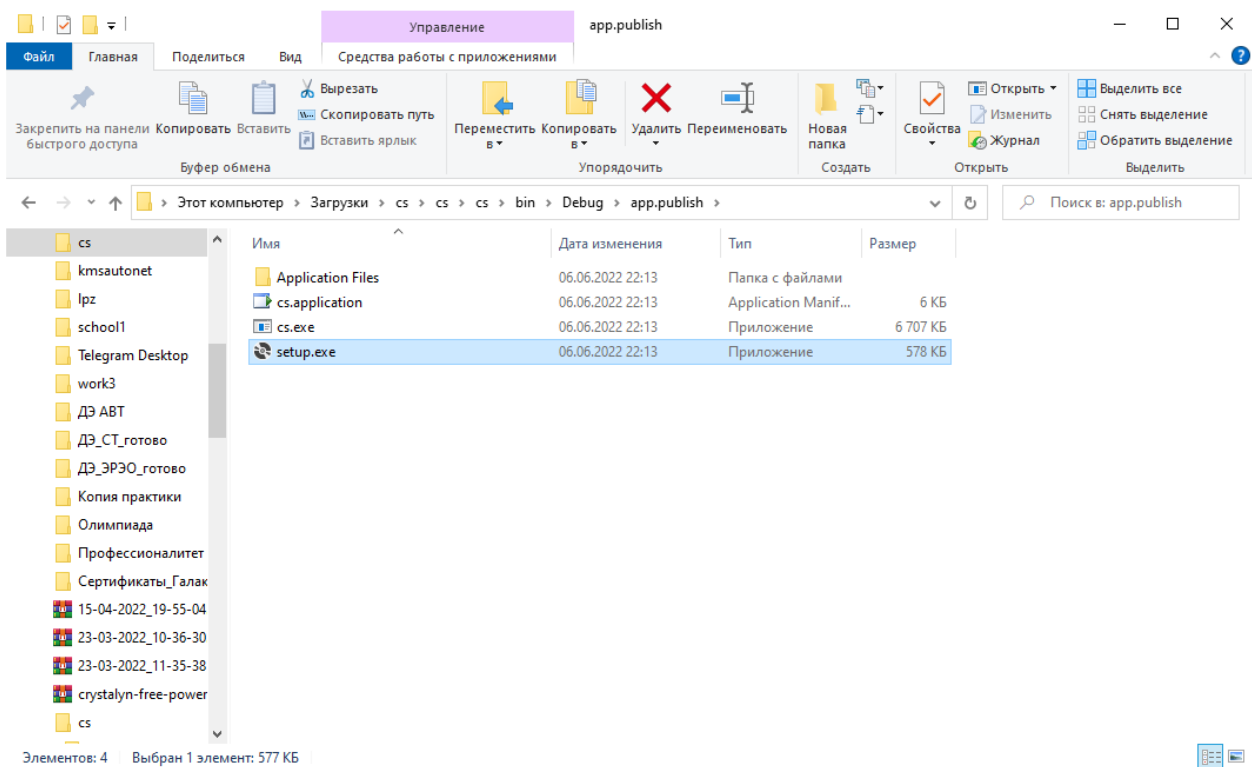
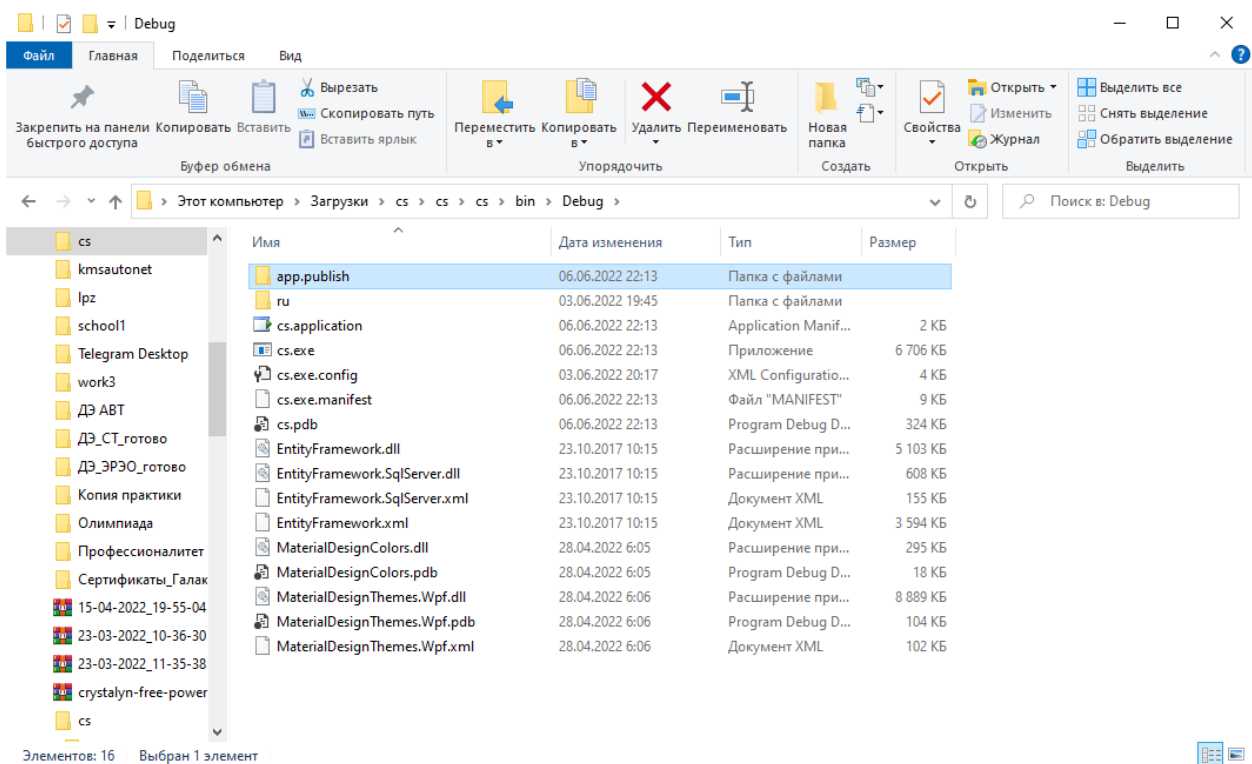


12. После того, как закончите работу с параметрами, нажмите на кнопку «Опубликовать сейчас»:

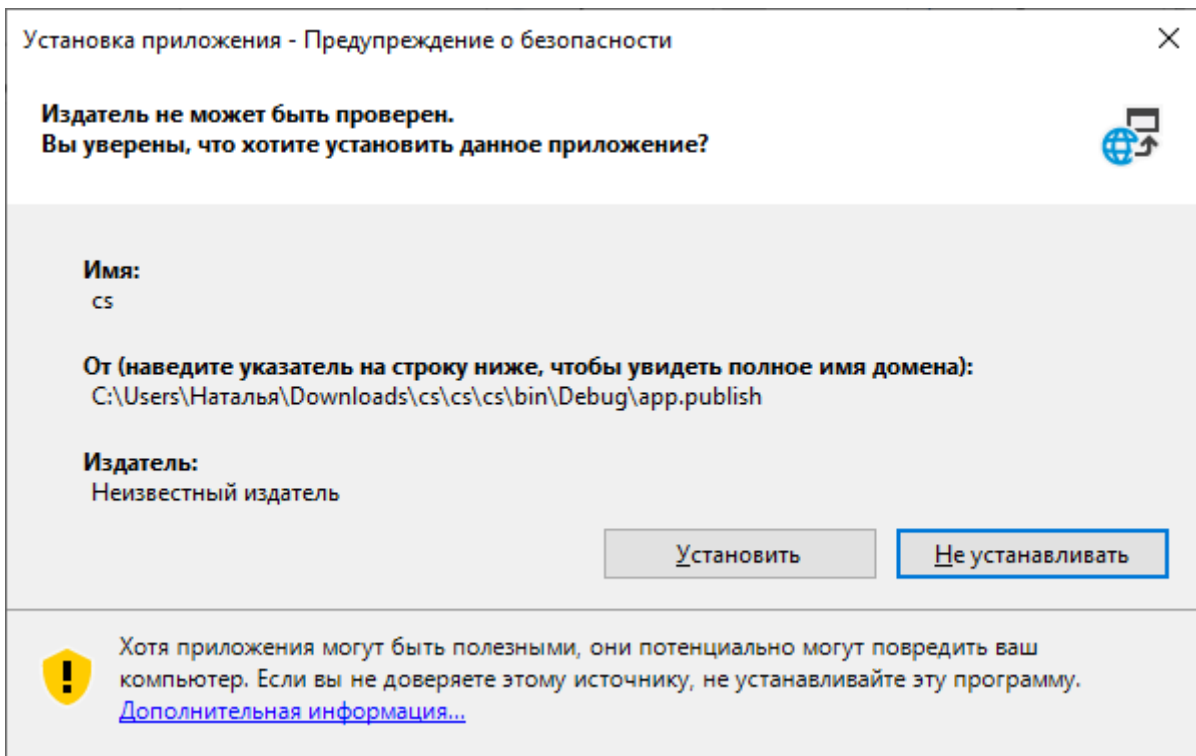


13. Затем, по указанному вами ранее адресу будет создан ярлык «Setup.exe», содержащий в себе минимальный инсталлятор.

14. По умолчанию, в папке проекта создается папка «app.publish», в которой будет создан файл установочника:

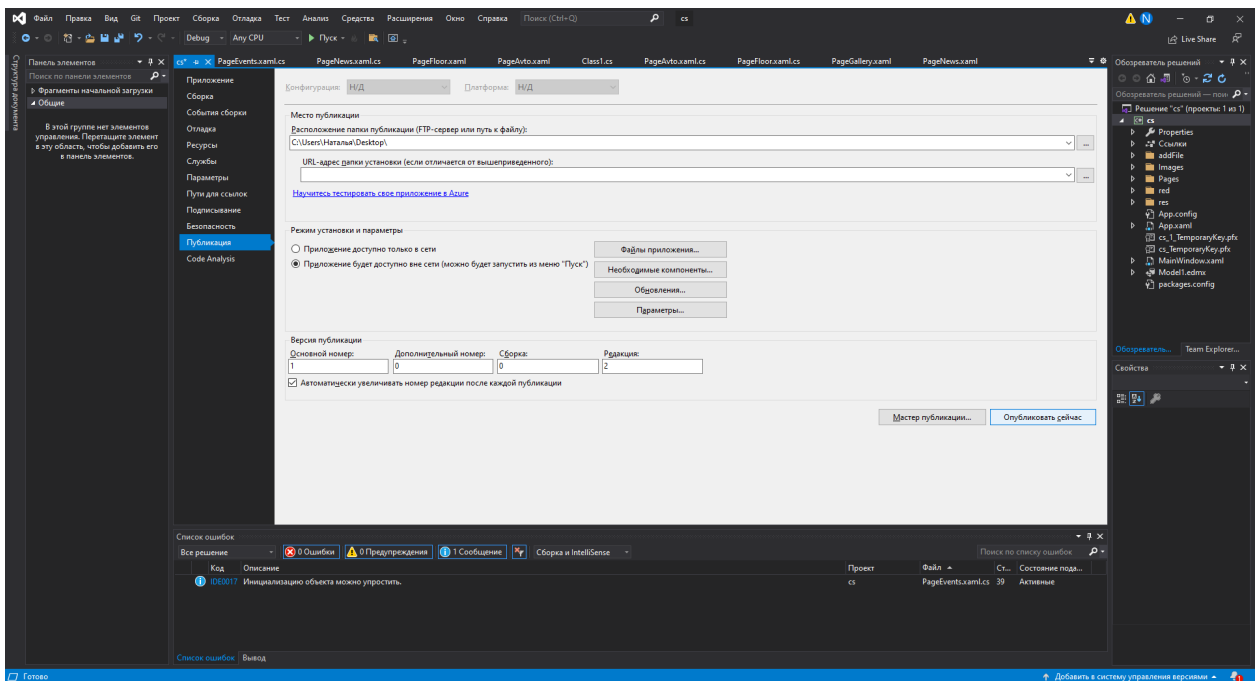


15. При открытии этого файла возникает окно-предупреждение:



Нажимаем кнопку «Установить». Если на компьютере не хватает какого-либо компонента, то перед установкой программы будет предложено доустановить недостающие компоненты.

16. Если вы внесли какие-либо изменения в программу и хотите, чтобы инсталлятор устанавливал обновленную программу, вам необходимо вернуться в «Свойства» решения и опубликовать инсталлятор заново.



17. И затем, пользователь запускает приложение с помощью иконки на Рабочем столе, и в случае внесенных изменений, приложение предложит совершить обновление.
18. Установленная таким образом программа также вносится в раздел Панели управления – «Программы и компоненты». Приложение можно оттуда удалить.

Развертывание WPF-приложений

Рано или поздно вы отпустите свое приложение WPF в "свободное плавание". Хотя существуют десятки разных способов передать готовое приложение с компьютера разработчика на настольный компьютер конечного пользователя, большинство приложений WPF используют одну из описанных ниже стратегий развертывания:

Запуск в браузере

Если создано WPF-приложение, состоящее из веб-страниц, его можно запускать прямо в браузере. Ничего устанавливать не понадобится. Однако приложение должно быть готово к тому, чтобы функционировать с очень небольшим набором привилегий. (Например, нельзя получать доступ к произвольным файлам, работать с системным реестром Windows, отображать всплывающие окна и т.д.)

Развертывание через браузер

WPF-приложения тесно интегрированы со средством установки ClickOnce ("однократный щелчок"), которое позволяет пользователю запустить программу установки со страницы браузера. Лучше всего то, что приложения, установленные с помощью ClickOnce, могут быть сконфигурированы так, чтобы автоматически проверять наличие обновлений. Отрицательной стороной является то, что возможности по настройке установки ограничены, и нет никакого способа выполнить задачи конфигурирования системы (вроде модификации системного реестра Windows, создания базы данных и т.п.).

Развертывание с помощью традиционной программы установки

Этот подход все еще существует в мире WPF. В случае выбора такого варианта придется решить, нужно ли создавать полноценный установочный пакет **Microsoft Installer (MSI)** или же обратиться к более простой (но и ограниченной) установке ClickOnce. Готовый установочный пакет можно распространять на CD-диске, во вложении сообщения электронной почты, через общедоступный сетевой ресурс и т.д.

Хотя ClickOnce была разработана как легковесная технология развертывания, которая не заменит полноценные программы установки, каждая новая версия добавляет к ней ряд новых возможностей. В .NET 4 установка ClickOnce может создавать значки рабочего стола и регистрировать типы файлов. Вдобавок .NET теперь устанавливает дополнение Firefox под названием *Microsoft .NET Framework Assistant*, которое позволяет пользователям запускать установки ClickOnce из браузеров Firefox и Internet Explorer (естественно, если это дополнение включено).

Хотя формально вполне допускается перемещать приложение .NET с одного компьютера на другой простым копированием содержащей его папки, все же профессиональные приложения часто требуют несколько большего. Например, может понадобиться добавить множество ярлыков в меню Start (Пуск), изменить настройки системного реестра либо установить дополнительные ресурсы (такие как специальный журнал регистрации событий или база данных). Для получения этих средств должна быть создана специальная программа установки.

Существует много вариантов создания программ установки. Можно воспользоваться коммерчески распространяемым продуктом типа [InstallShield](#), либо создать установочный файл MSI с помощью шаблона Setup Project (Проект установки) в Visual Studio.

Традиционные программы установки предлагают пользователю хорошо знакомый интерфейс мастера установки, с избытком средств для передачи файлов и выполнения разнообразных конфигурационных действий.

Другой вариант — применение системы развертывания ClickOnce, тесно интегрированной с WPF. Система ClickOnce обладает рядом ограничений (большинство из них связано с положенным в основу ClickOnce проектным решением), но обеспечивает два существенных преимущества:

- поддержка установки со страницы браузера (которая может размещаться как во внутренней сети, так и в Интернете);
- поддержка обновлений автоматической загрузки и установки обновлений.

Этих двух средств может оказаться недостаточно, чтобы соблазнить разработчиков отказаться от использования полноценных программ установки. Но если интересует простое, легковесное развертывание, которое работает через Интернет и поддерживает автоматическое обновление, то в этом случае ClickOnce — просто идеальный вариант.

Модель ClickOnce и частичное доверие

Обычные приложения WPF требуют *полного доверия* (*full trust*), потому что им нужны права доступа к неуправляемому коду для создания окна WPF. Это означает, что установка автономного приложения WPF посредством ClickOnce вызовет такое же препятствие со стороны системы безопасности, как и установка любого другого типа приложения из Интернета — в частности, браузер выдаст предупреждение безопасности. Если пользователь соглашается, устанавливаемое приложение получает возможность делать все, что разрешено делать в системе текущему пользователю.

ClickOnce работает иначе с более старыми приложениями Windows Forms. Приложения Windows Forms могут быть сконфигурированы на использование частичного доверия и затем развернуты с помощью ClickOnce. В лучшем случае это означает, что пользователь может устанавливать частично доверенное приложение Windows Forms через ClickOnce без какого-либо предупреждения безопасности или повышения привилегий.

Может показаться, что подход Windows Forms лучше, но WPF предусматривает возможность комбинировать частично доверенное программирование и технологию установки ClickOnce. Фокус заключается в использовании [модели XBAP](#). В такой ситуации приложение запускается в браузере, поэтому ему не нужно создавать каких-либо окон, и не нужны права доступа к неуправляемому коду. Более того, поскольку приложение доступно через URL (и затем кэшируется локально), пользователь всегда запускает последнюю, самую свежую версию.

При проектировании ClickOnce предполагалось иметь дело с простыми, прямолинейными приложениями. Развертывание ClickOnce, в частности, подходит для производственных приложений и внутреннего программного обеспечения компании. Обычно такие приложения выполняют свою работу, опираясь на данные и службы, предоставляемые серверами приложений среднего уровня. Как следствие, им не требуется привилегированный доступ к локальному компьютеру.

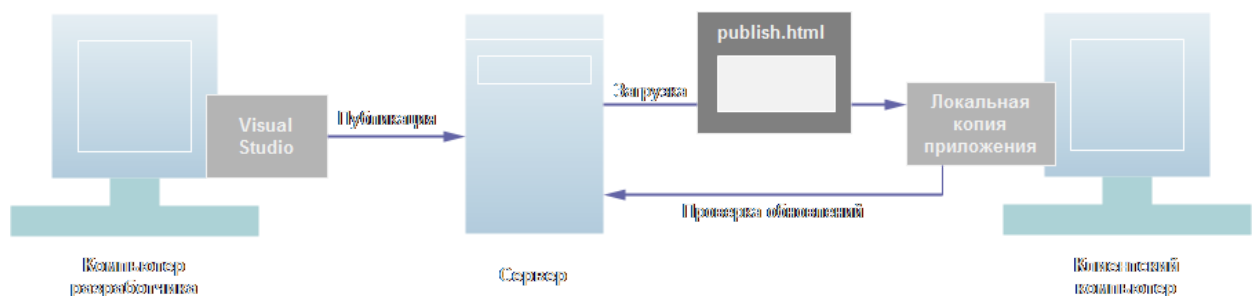
Эти приложения также развертываются в среде масштаба предприятия, которая может включать тысячи рабочих станций. В таких средах стоимость развертывания приложений и их обновления довольно значительна, особенно если все это должно поддерживаться администратором. В результате более важно предложить простой и прямолинейный процесс, нежели развитый пакет средств.

Модель ClickOnce также может иметь смысл для прикладных приложений, которые распространяются через Интернет, в частности, если эти приложения требуют частого обновления и не предъявляют строгих требований к установке. Однако ограничения ClickOnce (такие как недостаток гибкости в настройке мастера установки) делают эту модель непрактичной для сложных прикладных приложений, которые выдвигают детализированные требования к установке или нуждаются во взаимодействии с пользователем при выполнении ряда тонких

конфигурационных шагов. В таких случаях понадобится создавать специальное приложение установки.

Чтобы установить приложение WPF с помощью ClickOnce, на компьютере уже должна быть установлена исполняющая среда .NET Framework. Это требование проверяется при первом запуске установки ClickOnce. Если исполняющая среда .NET Framework не установлена, отображается окно сообщения, которое поясняет суть проблемы и предлагает установить .NET с веб-сайта Microsoft.

Хотя ClickOnce поддерживает несколько типов развертывания, общая модель разработана так, чтобы сделать веб-развертывание практичным и легким. Вот как это работает. Среда Visual Studio используется для публикации приложения ClickOnce на вебсервере. Затем пользователь переходит на автоматически сгенерированную страницу (по имени publish.htm), которая предлагает ссылку для установки приложения. Когда пользователь щелкает на этой ссылке, приложение загружается, устанавливается и добавляется в меню Пуск. На рисунке показан этот процесс:



Хотя ClickOnce — идеальный выбор для веб-развертывания, та же базовая модель подходит и для других сценариев, включая следующие:

- развертывание приложения из общедоступного сетевого ресурса;
- развертывание приложения с CD- или DVD-диска;
- развертывание приложения на веб-сервере или общедоступном сетевом ресурсе с последующей отправкой по электронной почте ссылки на программу установки.

Веб-страница установки не создается на общедоступном сетевом ресурсе, CD- или DVD-диске. Вместо этого пользователи устанавливают приложение непосредственным запуском программы setup.exe.

Наиболее интересная часть развертывания ClickOnce заключается в его способе поддержки обновления. По сути дела, вы (как разработчик) управляете несколькими установками обновления.

Например, вы можете сконфигурировать приложение для автоматической проверки наличия обновлений через определенные промежутки времени. Когда пользователь запускает приложение, в действительности он сначала запускает тонкую прослойку, которая проверяет наличие новой версии и предлагает пользователю загрузить ее.

Можно даже сконфигурировать приложение для использования веб-подобного интерактивного режима. В этом случае приложение должно запускаться со специальной веб-страницы ClickOnce. Приложение по-прежнему кэшируется локально для достижения оптимальной производительности, но пользователи не смогут запустить его до тех пор, пока не подключатся к сайту, на котором это приложение опубликовано. Это гарантирует, что они всегда будут запускать последнюю, самую современную версию приложения.

Ограничения ClickOnce

Развертывание ClickOnce не позволяет повсеместное конфигурирование. Многие аспекты его поведения жестко фиксированы — либо чтобы гарантировать согласованное восприятие пользователем, либо для обеспечения политик безопасности, подходящих для предприятий.

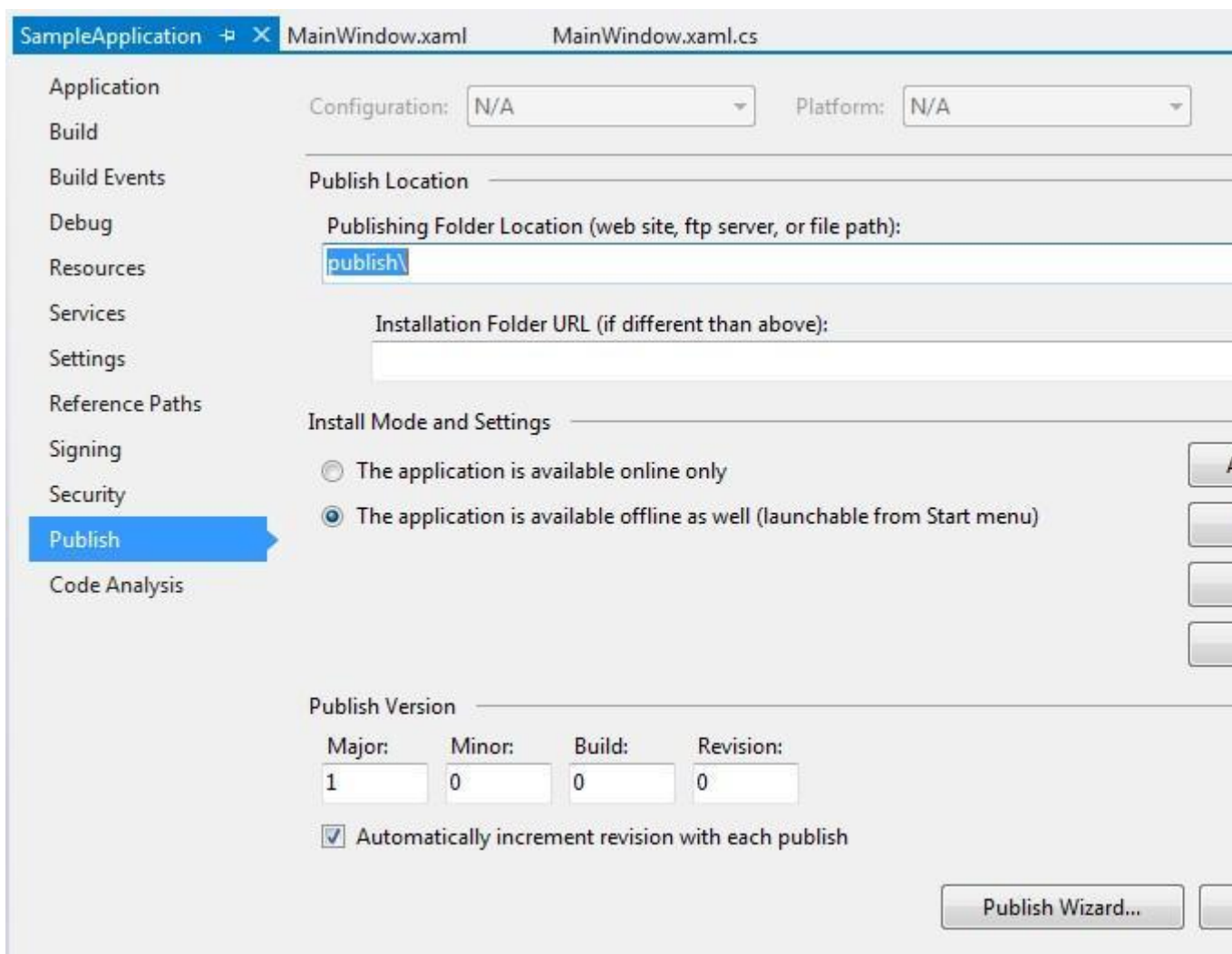
Ниже перечислены ограничения технологии ClickOnce:

- Приложения ClickOnce устанавливаются для единственного пользователя. Установка приложения для всех пользователей рабочей станции невозможна.
- Приложения ClickOnce всегда устанавливаются в управляемую системой папку, специфичную для пользователя. Изменить или повлиять на выбор папки, куда устанавливается приложение, нельзя.
- Если приложения ClickOnce устанавливаются в меню Пуск, они получают всего два ярлыка: один запускает приложение, а другой — открывает в браузере справочную веб-страницу. Изменить это не удастся, равно как не получится добавить приложение ClickOnce в группу Startup (Автозагрузка), меню Favorites (Избранное) и т.п.
- Нельзя изменять пользовательский интерфейс мастера установки. Это значит, что добавлять новые диалоговые окна, изменять текст в существующих окнах и т.п. невозможно.
- Нельзя изменять страницу установки, сгенерированную приложениями ClickOnce. Однако после генерации можно отредактировать HTML-разметку вручную.
- Развертывание ClickOnce не позволяет устанавливать разделенные компоненты в кэш глобальных сборок (GAC).
- Развертывание ClickOnce не дает возможность выполнять специализированные действия (такие как создание базы данных или конфигурирование настроек реестра).

Некоторые из перечисленных ограничений можно обойти. Например, приложение может быть сконфигурировано на добавление настроек реестра при его первом запуске на новом компьютере. Однако если предъявлены сложные требования к установке, то намного лучше обратиться к созданию полноценной специальной программы установки. Можно воспользоваться инструментами от независимых разработчиков вроде InstallShield или же создать проект установки в Visual Studio.

Наконец, следует отметить, что .NET позволяет построить специальную программу установки, использующую технологию развертывания ClickOnce. Это дает возможность проектировать сложные приложения установки, не жертвуя при этом средствами автоматического обновления, которые предлагаются ClickOnce. Однако есть и некоторые недостатки. Этот подход не только заставляет писать (и отлаживать) большой объем кода, но также требует использования унаследованных классов из комплекта инструментов Windows Forms для построения пользовательского интерфейса установки.

Прежде чем приступить к публикации ClickOnce, понадобится некоторая базовая информация о проекте. Дважды щелкните на узле Properties (Свойства) в окне Solution Explorer и перейдите на вкладку Publish (Публикация). Вы увидите настройки, показанные на рисунке:



Смысл всех настроек будет объяснен далее, а сначала следует указать некоторые базовые детали публикации.

Настройка издателя и продукта

Приложению для установки понадобится базовая идентичность, включающая имя издателя и название продукта, которые могут быть использованы в запросах процесса установки и в ярлыках меню Пуск.

Чтобы указать эту информацию, щелкните на кнопке Options (Параметры). Откроется диалоговое окно Publish Options (Параметры публикации) с набором дополнительных настроек, разнесенных по нескольким группам. Как показано на рисунке ниже, в списке слева выбран элемент Description (Описание), а справа находятся текстовые поля, которые позволяют указать ключевые детали: Publisher Name (Имя издателя), Suite Name (Имя комплекта) и Product Name (Имя продукта):

Publish Options

Description
Deployment
Manifests
File Associations

Publish language:
(Default)

Publisher name:
Alex Company

Suite name:

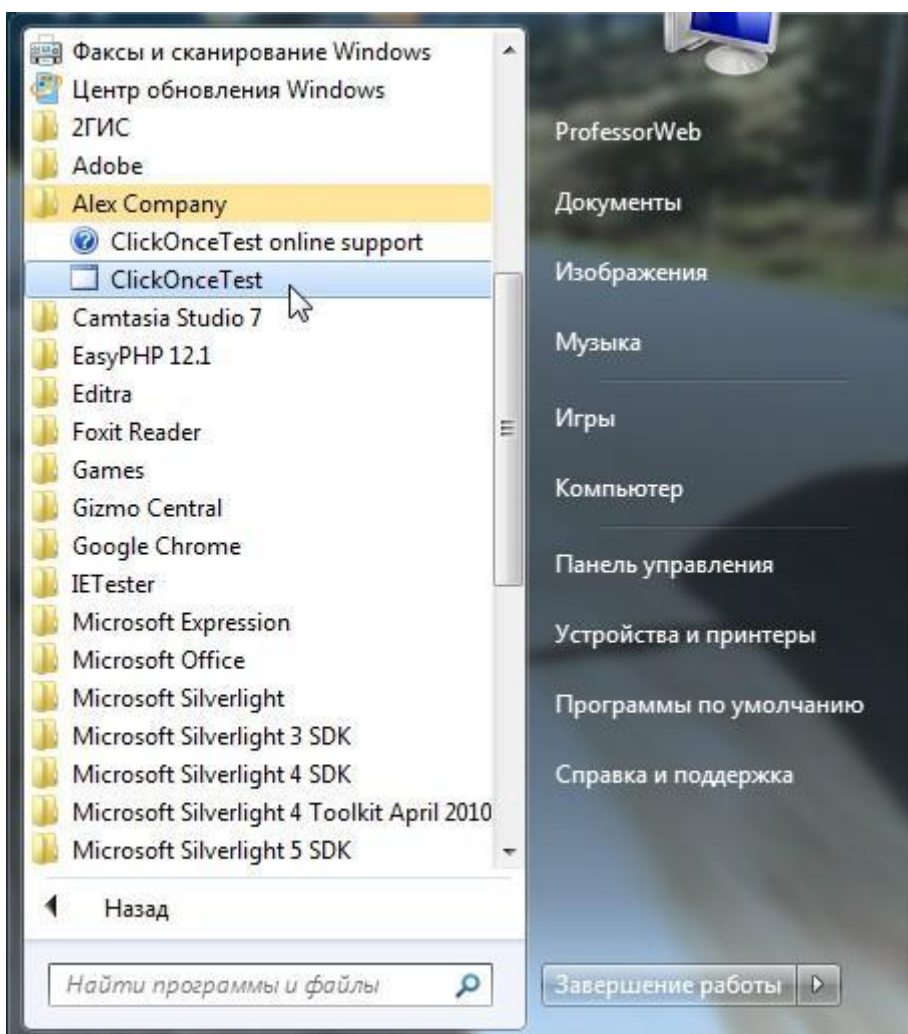
Product name:
ClickOnceTest

Support URL:
http://www.mycompany.com

Error URL:

OK Cancel

Эти детали важны, потому что они используются для создания иерархии меню Пуск. Если задается необязательное имя комплекта, ClickOnce создаст ярлык приложения в форме [Имя издателя] --> [Имя комплекта] --> [Имя продукта]. Если имя комплекта не указано, ClickOnce создаст ярлык [Имя издателя] --> [Имя продукта]. Ниже показано как будет выглядеть ярлык программы в меню Пуск:



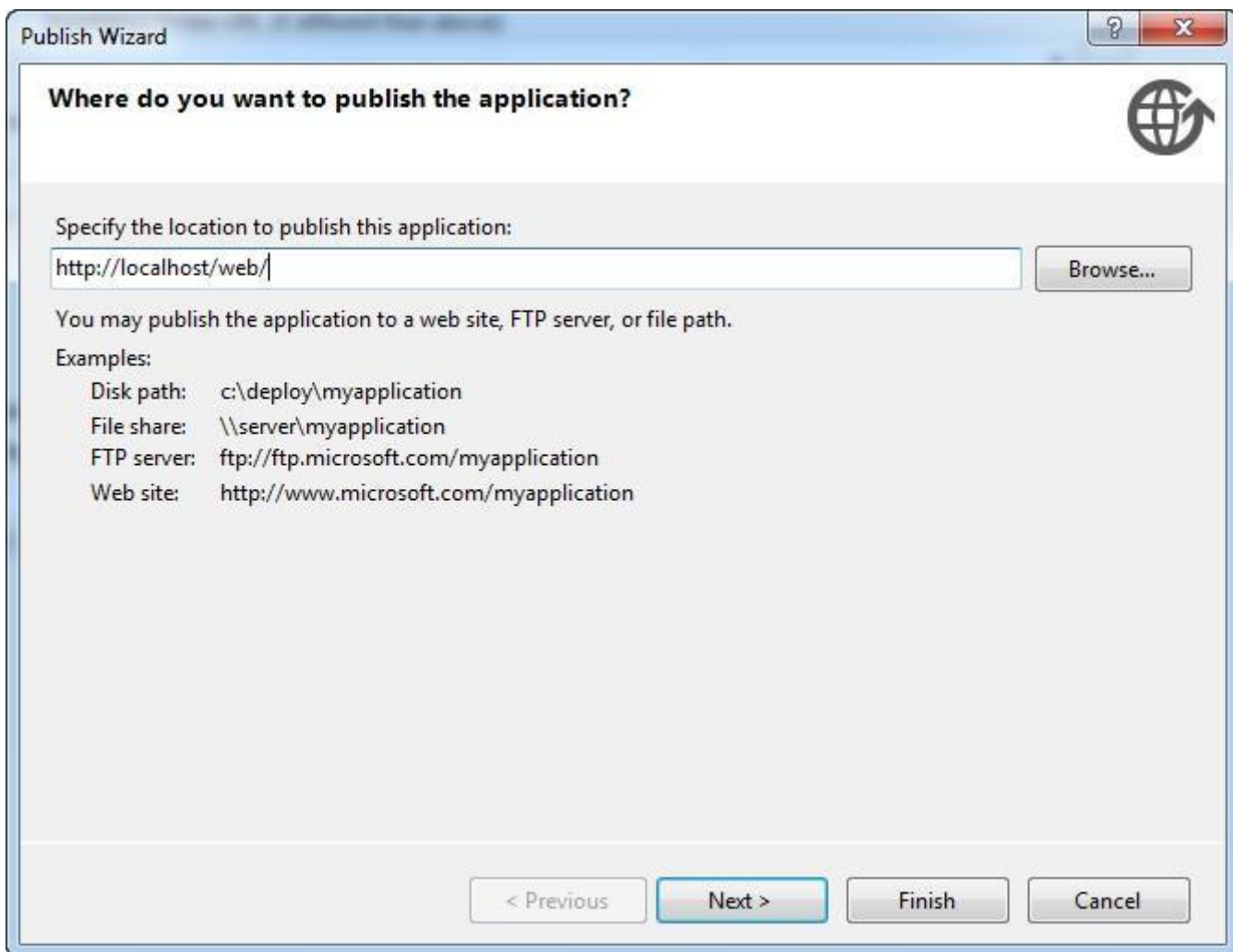
Если указан URL-адрес поддержки, ClickOnce создаст дополнительный ярлык [Имя продукта] online support. Когда пользователь щелкнет на нем, запустится браузер и загрузит указанную страницу.

URL-адрес ошибки — это ссылка на веб-сайт, который будет показан (в диалоговом окне), если при попытке установки приложения возникнет ошибка.

Запуск мастера публикации

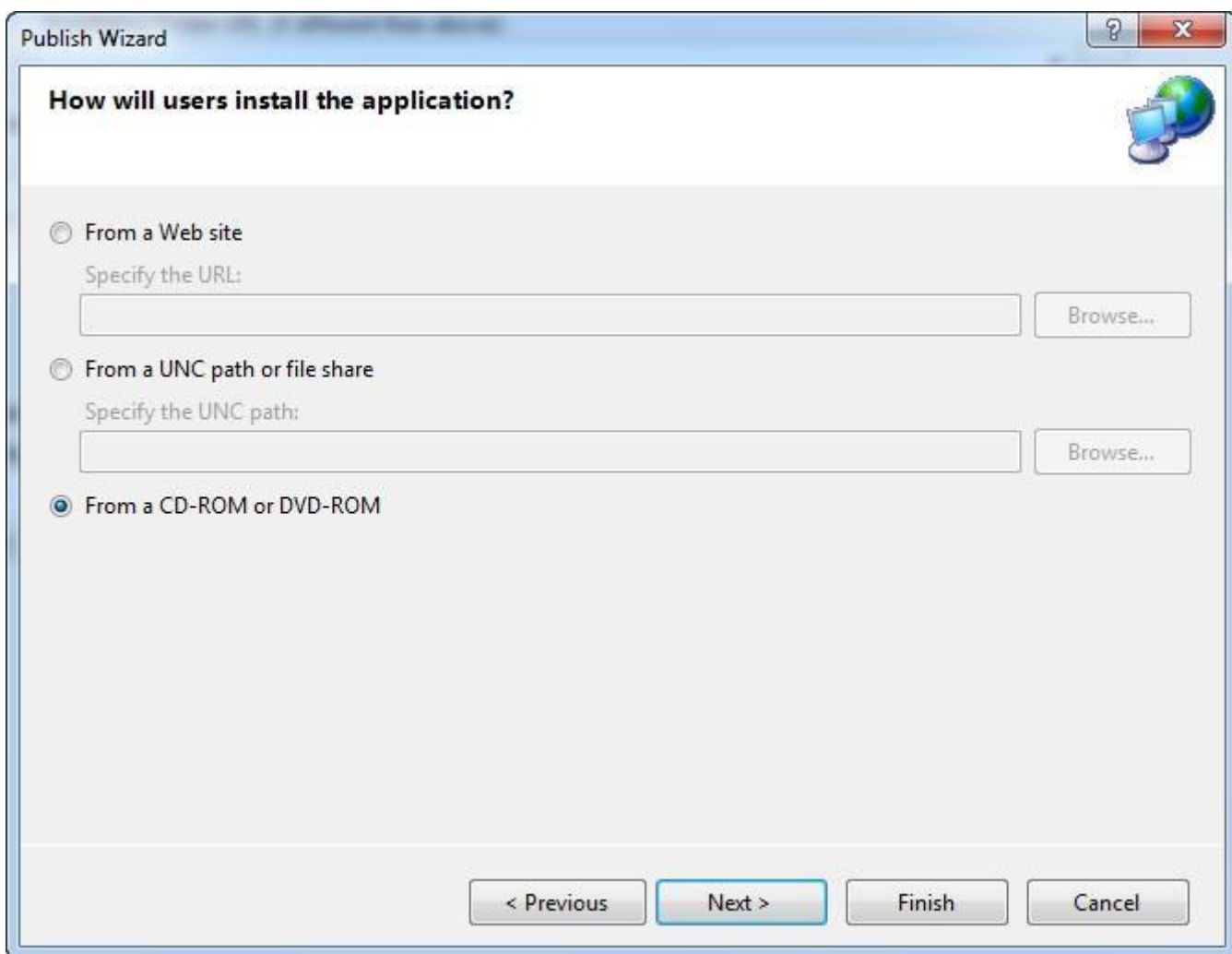
Простейший способ сконфигурировать настройки ClickOnce состоит в том, чтобы щелкнуть на кнопке Publish Wizard (Мастер публикации) внизу окна свойств, показанного на первом рисунке. В результате запустится мастер, который проведет через несколько простых шагов, чтобы собрать важную информацию. Этот мастер не предоставляет доступ ко всем средствам ClickOnce, однако это самый быстрый способ начать.

Первым выбором в мастере публикации будет выбор местоположения для публикации приложения:



В выборе местоположения первой публикации приложения нет ничего особо важного, поскольку это не обязательно то же самое место, где позже будут размещаться установочные файлы. Другими словами, можно опубликовать в локальном каталоге и затем передать файлы на веб-сервер. Единственный нюанс — перед запуском мастера публикации необходимо знать конечное место расположения файлов, поскольку эту информацию придется указывать. Без нее не будет работать функция автоматического обновления.

Разумеется, можно публиковать приложение непосредственно в конечное расположение, но это не обязательно. Фактически, локальное построение установки часто представляет собой простейший путь. Чтобы понять, как это работает, начните с выбора локального пути (например, C:\Temp\ClickOnceApp). Затем щелкните на кнопке Next (Далее). После этого возникает реальный вопрос — как пользователи будут устанавливать это приложение?



Выбор важен, потому что он влияет на стратегию обновления. Произведенный выбор будет зафиксирован в файле манифеста, поставляемом вместе с приложением.

Есть один случай, когда диалоговое окно, показанное на рисунке выше, не отображается. Если в качестве местоположения публикации вводится виртуальный каталог на веб-сервере (другими словами, URL-адрес, начинающийся с `http://`), то мастер предположит, что это окончательное местоположение установки.

На рисунке выше показаны три доступных выбора. Можно создать установку на сетевом файловом ресурсе, на веб-сервере или же на CD- либо DVD-диске. Ниже рассматриваются все варианты.

Публикация на сетевом файловом ресурсе

В этом случае все пользователи сети получают доступ к установке, перейдя по определенному пути UNC и запустив находящийся там файл по имени `setup.exe`.

Пусть UNC — это сетевой путь в форме `\\ИмяКомпьютера\ИмяОбщегоРесурса`. Сетевой диск использовать нельзя, поскольку он зависит от настроек системы (так что разные пользователи могут иметь разные настройки сетевых дисков). Чтобы обеспечить автоматические обновления, инфраструктура ClickOnce должна знать точно, где она сможет найти установочные файлы, и это то же самое место, где будут развернуты обновления.

Публикация для веб-сервера

Установку для веб-сервера можно создать в локальной корпоративной сети или в Интернете. Visual Studio сгенерирует HTML-файл по имени `publish.htm`, который упростит процесс. Пользователь

запросит эту страницу в своем браузере и щелкнет на ссылке для загрузки и установки приложения.

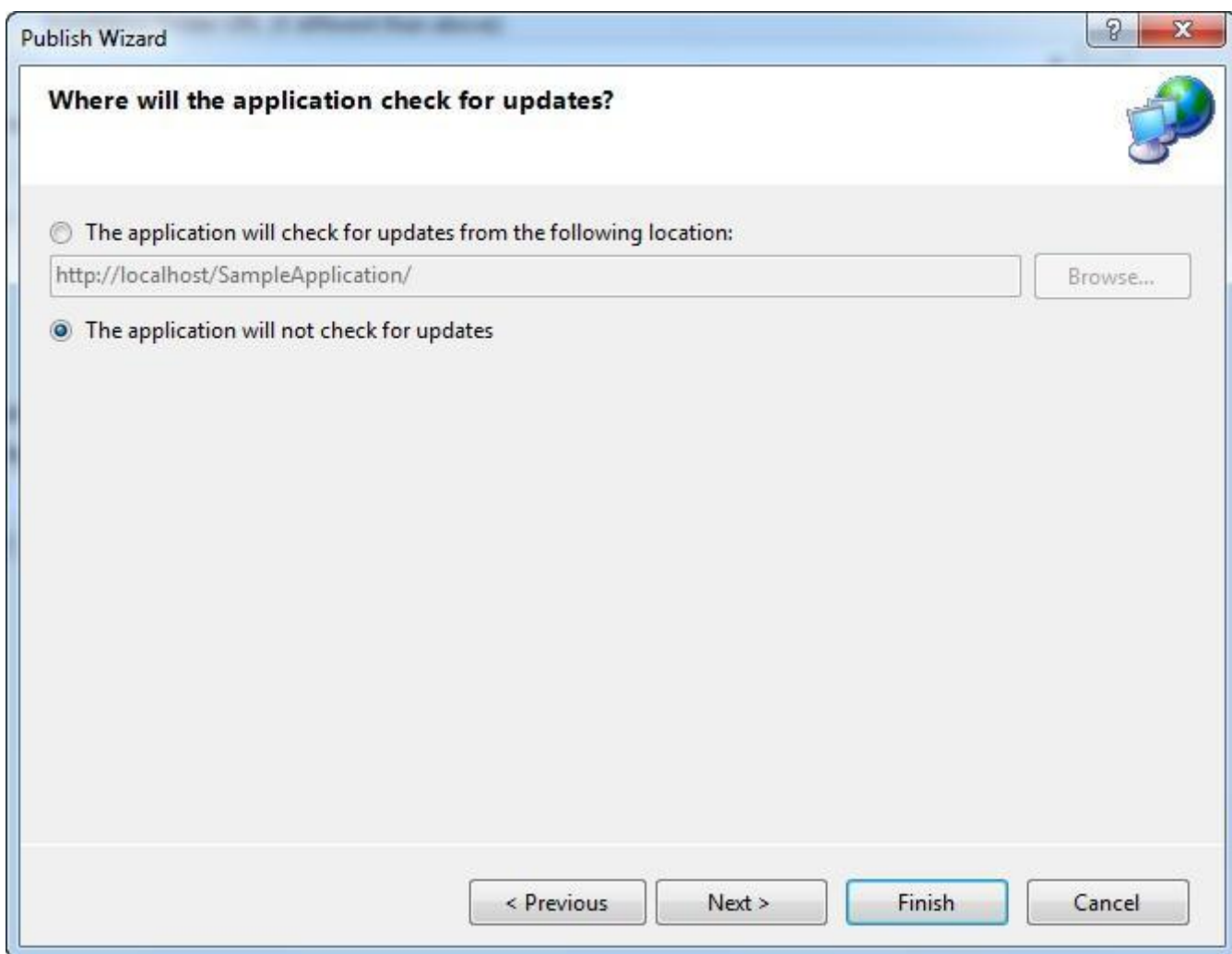
Существует несколько способов передачи файлов на веб-сервер. Чтобы воспользоваться двухшаговым подходом (опубликовать файлы локально, а затем передать их в нужное место), просто следует скопировать файлы из локального каталога на веб-сервер, используя соответствующий механизм (вроде FTP). Удостоверьтесь, что при этом сохраняется структура каталогов.

Для опубликования файлов непосредственно на веб-сервере, минуя предварительное тестирование, есть два выбора. Если применяется IIS, и текущая учетная запись пользователя обладает достаточными привилегиями для создания нового виртуального каталога на веб-сервере (или загрузки файлов в существующий виртуальный каталог), можно опубликовать файлы непосредственно на веб-сервер. Просто укажите путь к виртуальному каталогу на первом шаге мастера. Например, в качестве местоположения публикации можно использовать `http://ИмяКомпьютера/ИмяВиртуальногоКаталога` (в случае корпоративной сети) или `http://ДоменноеИмя/ИмяВиртуальногоКаталога` (для сервера, находящегося в Интернете).

Также можно публиковать непосредственно на веб-сервер через FTP. В Интернете это часто бывает обязательным требованием (в отличие от корпоративной сети). В этом случае Visual Studio установит соединение с веб-сервером и передаст туда файлы ClickOnce по FTP. При этом для установки соединения будет запрошено имя пользователя и пароль.

Публикация для CD- и DVD-диска

В случае выбора публикации на установочный носитель вроде CD- или DVD-диска все равно понадобится решить, планируются ли автоматические обновления. Некоторые организации используют исключительно развертывание с помощью CD-дисков, в то время как другие применяют его в дополнение к веб-развертыванию или развертыванию через сетевые ресурсы. На третьем шаге мастера выбираются опции для этого конкретного случая:

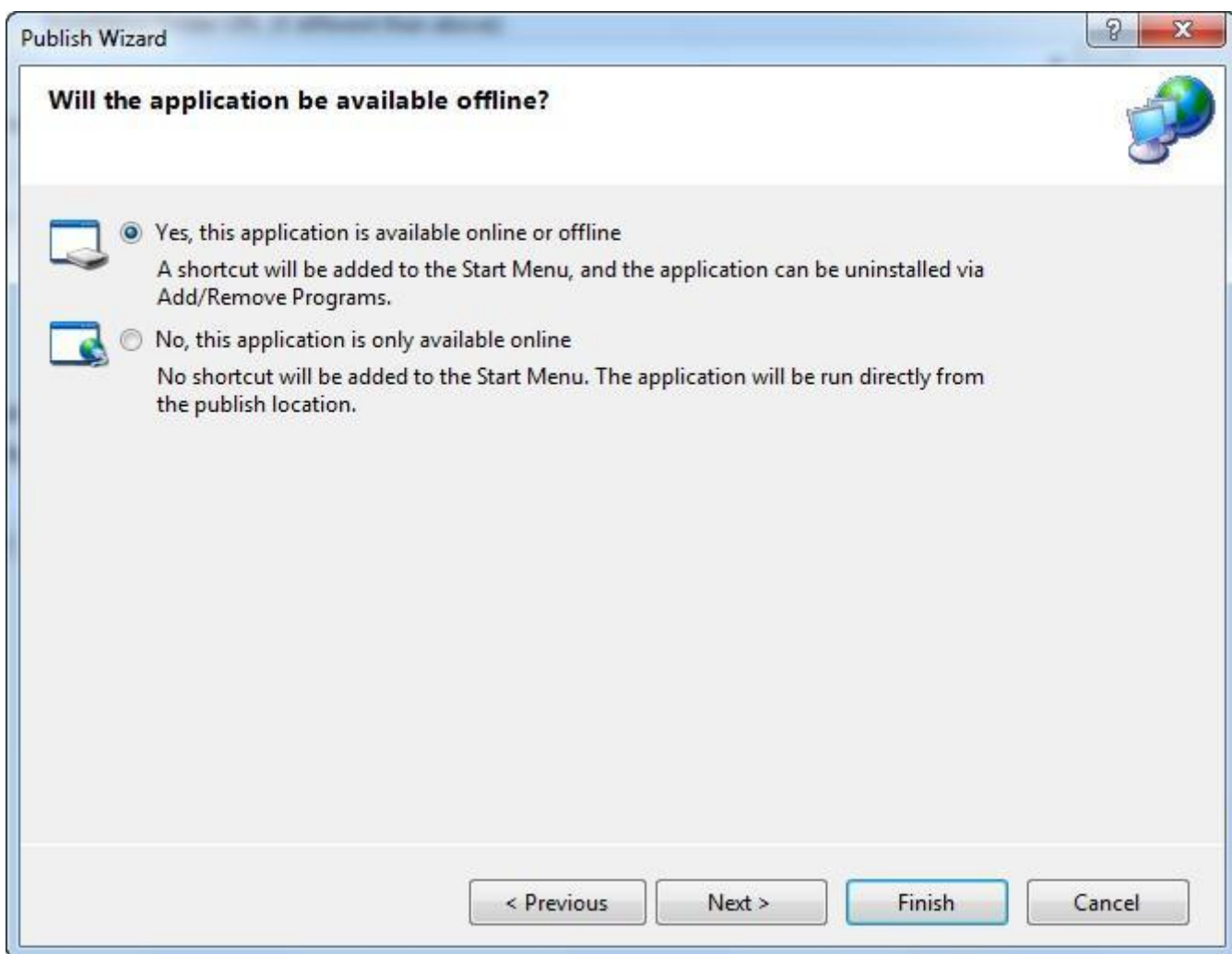


Здесь доступны следующие варианты:

- Можно указать URL-адрес или путь UNC, по которому приложение будет искать обновления. Это предполагает, что приложение будет опубликовано в этом месте.
- Можно опустить эту информацию и тем самым вообще отключить функцию автоматического обновления.
- Можно опустить эту информацию, но заставить приложение ClickOnce использовать местоположение установки в качестве местоположения обновления.

Интерактивный и автономный режимы

В случае создания развертывания для веб-сервера или сетевого ресурса должна быть установлена дополнительная опция:



Выбор по умолчанию заключается в создании приложения, доступного в интерактивном и автономном режиме, которое запускается независимо от того, может пользователь подключиться к месту публикации или нет. В этом случае ярлык для запуска приложения добавляется в меню Пуск.

Если выбрано создание приложения, доступного только в интерактивном режиме, то пользователю для запуска приложения будет требоваться каждый раз обращаться к месту публикации. (На веб-странице `publish.htm` будет отображаться кнопка Run (Запуск) вместо Install (Установка).) Это исключит вероятность запуска старых версий приложения после применения обновлений. Данная часть модели развертывания аналогична веб-приложениям.

Когда создается приложение, доступное только в интерактивном режиме, оно по-прежнему будет загружаться (в локально кэшируемое местоположение) при первом запуске.

Таким образом, хотя первоначальное время запуска может быть больше (из-за первоначальной загрузки), приложение будет работать так же быстро, как любое обычное установленное Windows-приложение. Однако приложение не может быть запущено, если пользователь не подключен к сети или Интернету, что делает его неподходящим для мобильных пользователей (например, пользователей ноутбуков, которым не всегда доступно подключение к Интернету).

Если выбрано создание приложения, которое поддерживает автономный режим, то программа установки добавит ярлык в меню Пуск. Пользователь может запустить приложение с помощью этого ярлыка, независимо от того, подключен компьютер к сети или нет.

Если компьютер находится в интерактивном режиме, то приложение проверит наличие новых версий в месте, где опубликовано приложение. При наличии обновлений приложение предложит

пользователю установить их. Позже будет показано, как сконфигурировать эту политику. При публикации для CD-диска отсутствует выбор для создания приложения, доступного только в интерактивном режиме.

Это — последний выбор в мастере публикации. Щелкните на кнопке Next, чтобы просмотреть итоговую информацию и затем на кнопке Finish (Готово) для генерации файлов развертывания и копирования их в место, указанное на первом шаге. В этот момент можно быстро переопубликовать приложение, щелкнув на кнопке Publish Now (Опубликовать сейчас) либо выбрав в меню пункт Build --> Publish [Имя приложения] (Сборка --> Опубликовать [Имя приложения]).

Развернутая файловая структура

Технология ClickOnce использует довольно простую структуру каталогов. Она создает файл setup.exe в указанном месте и подкаталог для приложения. Например, при развертывании приложения по имени ClickOnceText в каталоге c:\ClickOnceTest появляются следующие файлы:

c:\ClickOnceTest\setup.exe

c:\ClickOnceTest\publish.htm

c:\ClickOnceTest\ClickOnceTest.application

c:\ClickOnceTest\ClickOnceTest_1_0_0_0.application

c:\ClickOnceTest\ClickOnceTest_1_0_0_0\ClickOnceTest.exe.deploy

c:\ClickOnceTest\ClickOnceTest_1_0_0_0\ClickOnceTest.exe.manifest

Файл publish.htm присутствует только в случае публикации на веб-сервере. Файлы .manifest и .application хранят информацию о необходимых файлах, настройках обновлений и прочие детали. (С подробной информацией об этих файлах и их XML-файлом можно ознакомиться в справочной системе MSDN.) Файлы .manifest и .application снабжаются электронной подписью во время публикации, поэтому не могут модифицироваться вручную. В случае внесения изменений ClickOnce заметит разницу и откажется устанавливать приложение.

По мере публикации новых версий приложения ClickOnce будет добавлять новые подкаталоги для каждой новой версии. Например, изменение опубликованной версии приложения на 1.0.0.1 приводит к появлению нового каталога:

c:\ClickOnceTest\ClickOnceTest_1_0_0_1\ClickOnceTest.exe.deploy

c:\ClickOnceTest\ClickOnceTest_1_0_0_1\ClickOnceTest.exe.manifest

После запуска программа setup.exe отработает процесс установки всего обязательного программного обеспечения (такого как .NET Framework) и затем установит наиболее свежую версию приложения.

Мастер публикации представляет собой быстрый способ создания комплекта развертывания ClickOnce, но он не позволяет настроить все возможные параметры. Эти параметры доступны через вкладку Publish окна свойств приложения.

Многие из этих настроек дублируют детали, которые вы уже видели в мастере. Например, первые два текстовых поля позволяют выбрать местоположение публикации (место, куда будут помещены файлы ClickOnce) и местоположение установки (место, из которого пользователь запустит установку). Настройка *Install Mode (Режим установки)* позволяет выбрать, должно приложение устанавливаться на локальном компьютере или запускаться только в интерактивном режиме. Однако есть некоторые установки, которые в мастере не доступны.

Версия публикации

В разделе Publish Version (Версия публикации) устанавливается версия приложения, которая сохраняется в файле манифеста ClickOnce. Это не то же самое, что версия сборки, которую можно указать на вкладке Application (Приложение), хотя можно установить их одинаковыми.

Ключевое отличие в том, что версия публикации является критерием, служащим для определения доступности нового обновления. Если пользователь запускает версию 1.5.0.0 приложения, а при этом доступна версия 1.5.0.1, то инфраструктура ClickOnce отобразит диалоговое окно обновления.

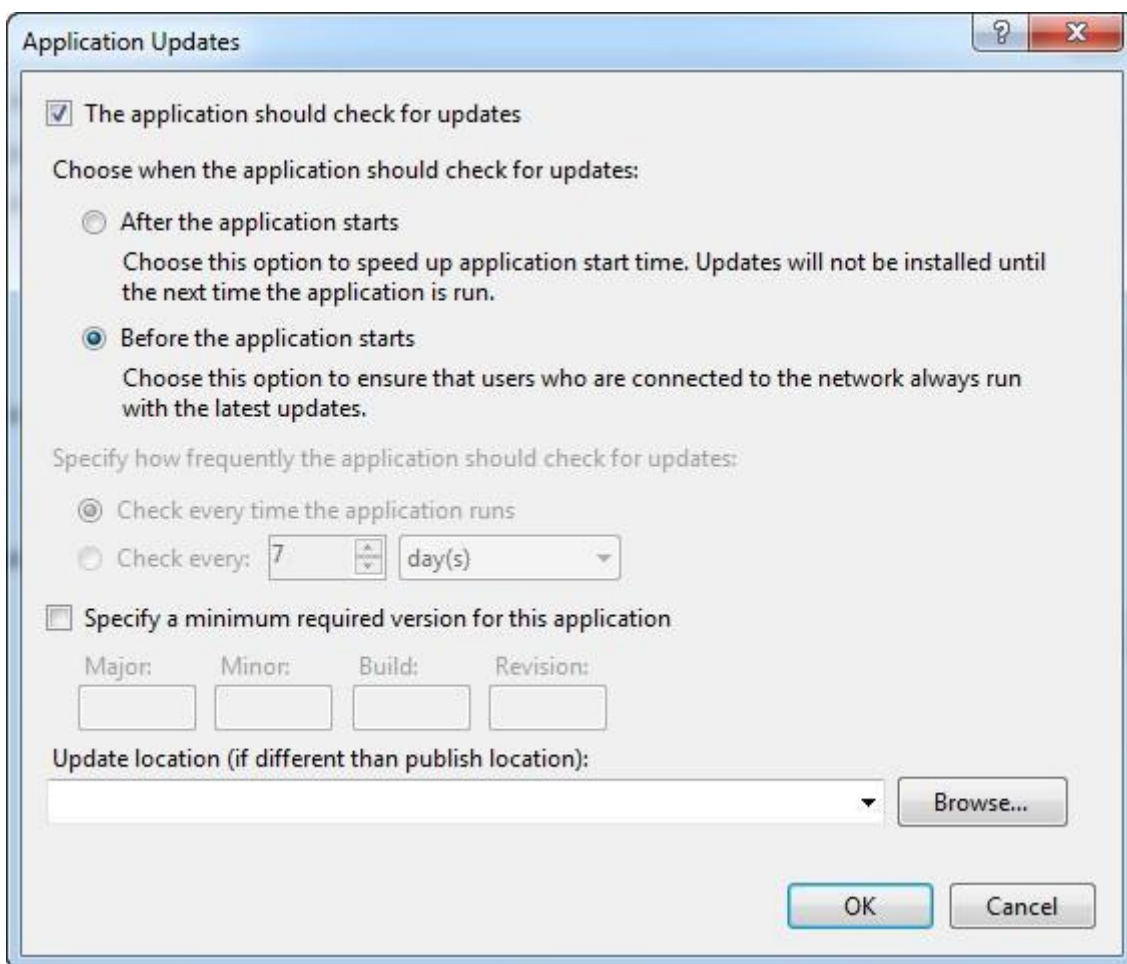
По умолчанию флажок Automatically Increment Revision with Each Publish (Автоматически увеличивать номер редакции с каждой публикацией) отмечен, и в этом случае финальная часть версии публикации (номер редакции) увеличивается на 1 после каждой публикации, так что 1.0.0.0 превращается в 1.0.0.1, затем в 1.0.0.2 и т.д.

Если нужно опубликовать одну и ту же версию приложения во многих местоположениях, следует снять отметку с этого флажка. Однако имейте в виду, что средство автоматического обновления вступает в действие только тогда, когда обнаруживает более высокий номер версии. Отметка даты развернутых файлов влияния не оказывает (и ненадежна).

Может показаться, что совершенно незелегантно отслеживать отдельные номера версий сборки и публикации. Однако иногда это имеет смысл. Например, при тестировании приложения может понадобиться сохранить фиксированный номер версии сборки, чтобы предотвратить попадание к тестировщикам самой последней версии. В этом случае можно использовать ту же версию сборки, но оставить автоматически увеличиваемым номер версии публикации. Когда все готово к выпуску официального обновления, можно установить версию сборки и версию публикации одинаковыми.

Обновления

Щелкните на кнопке Updates (Обновления), чтобы отобразить диалоговое окно Application Updates (Обновления приложения). В этом окне можно выбрать стратегию обновлений:



Сначала необходимо указать, должно ли приложение проверять наличие обновлений, отметив флажок **The application should check for updates** (Приложение должно проверять наличие обновлений). Затем следует выбрать, когда должно происходить обновление. Для этого доступны два переключателя:

Before the application starts

Если использовать эту модель, то инфраструктура ClickOnce проверяет наличие обновлений приложения (на веб-сайте или сетевом ресурсе) при каждом запуске приложения пользователем. Если обновление обнаружено, оно устанавливается и затем запускается приложение. Эта опция — хороший выбор, если нужно гарантировать получение пользователем обновлений немедленно после их появления.

After the application starts

Если выбрать эту модель, то инфраструктура ClickOnce проверяет новые обновления после запуска приложения. Если обнаруживается обновленная версия, она устанавливается при следующем запуске приложения пользователем. Это — рекомендованная опция для большинства приложений, поскольку сокращает время загрузки.

Если отдается предпочтение проверке после запуска приложения, то такая проверка осуществляется в фоновом режиме. Можно выбрать выполнение проверки при каждом выполнении приложения (по умолчанию так и есть) или проверять реже, через какой-то интервал времени, например, один раз в несколько часов, дней или недель.

Можно также указать необходимую минимальную версию. Это можно использовать для того, чтобы сделать обновления обязательными. Например, если установить версию публикации 1.5.0.1 и минимальную версию 1.5.0.0, а затем опубликовать приложение, то любой пользователь,

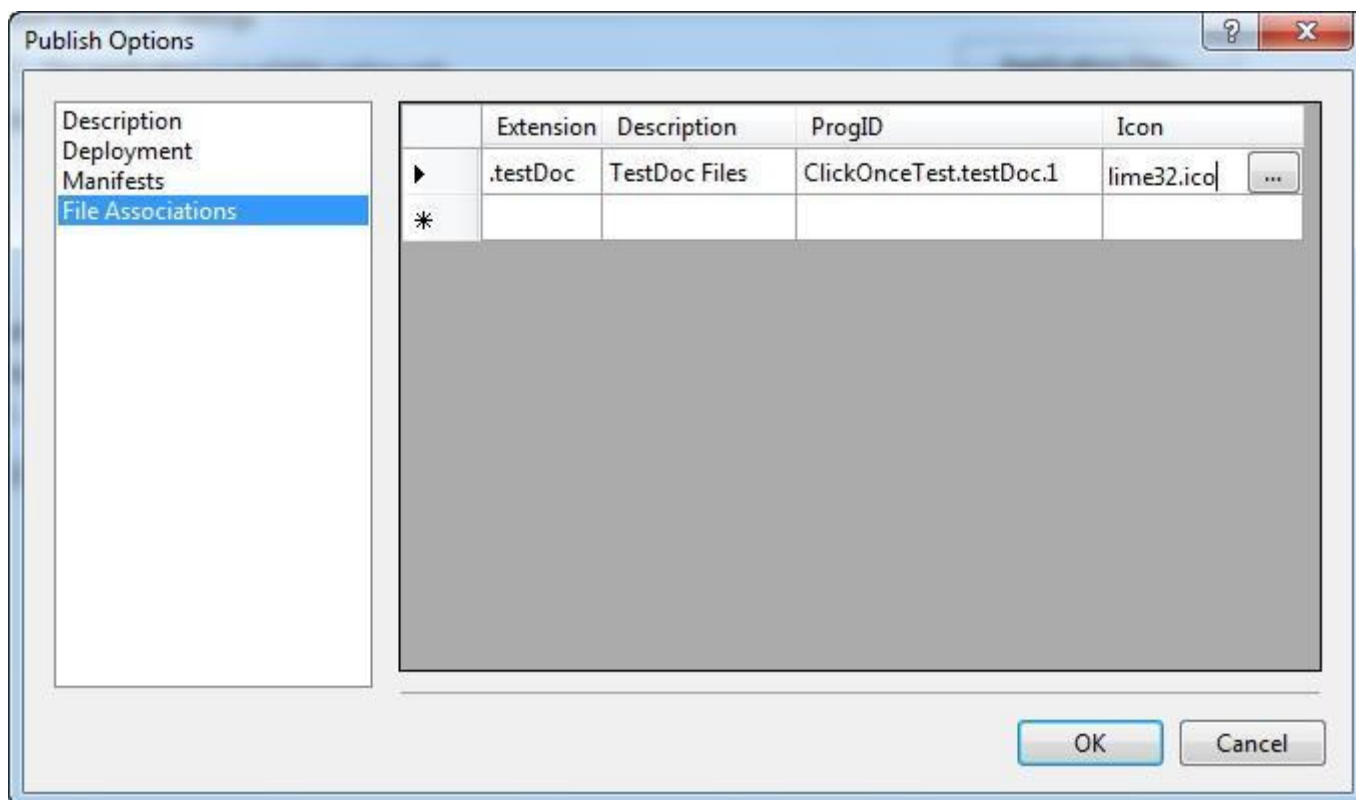
располагающий версией старше 1.5.0.0, будет вынужден выполнить обновление, прежде чем сможет запустить приложение (по умолчанию минимальная версия не устанавливается и все обновления необязательны).

Даже если указать минимальную версию и заставить приложение проверять наличие обновлений перед запуском, то пользователь сможет запустить старую версию приложения. Это происходит, когда он находится в автономном режиме — при этом проверка обновлений завершается сбоем, не сообщая об ошибке. Единственный способ обойти это ограничение — создать приложение, доступное только в интерактивном режиме.

Ассоциации файлов

Технология ClickOnce позволяет установить до восьми ассоциаций файлов. Это типы файлов, которые будут привязаны к приложению, чтобы двойной щелчок на файле соответствующего типа в Windows Explorer приводил к автоматическому запуску приложения.

Чтобы создать ассоциацию файла, щелкните на кнопке Options (Параметры) внутри вкладки Publish (Публикация). Откроется диалоговое окно Publish Options (Параметры публикации). В списке слева выберите элемент File Associations (Ассоциации файлов). Появится таблица, в которой можно ввести информацию об ассоциации файла:



Каждая такая ассоциация требует четырех порций информации: расширение файла, текстовое описание, идентификатор программы (ProgID) и значок. ProgID — это текстовый код, уникально идентифицирующий тип файла. По соглашению он должен базироваться на имени и версии приложения, хотя на самом деле формат не имеет значения, если он уникален. Значок указывает на файл в проекте. Для того чтобы этот файл был включен в установку ClickOnce, его понадобится выбрать в окне Solution Explorer и установить Build Action (Действие сборки) в Content (Содержимое).

Для ассоциации файла не нужно указывать только одну деталь — имя или путь программы. Дело в том, что ClickOnce уже располагает этой информацией.

Используя ассоциации файлов с ClickOnce, следует помнить об одном важном моменте. В противоположность ожиданиям, когда пользователь дважды щелкает на зарегистрированном файле, он не передается приложению в качестве параметра командной строки. Вместо этого файл потребуется извлечь из текущего домена приложения, как показано ниже:

```
string commandLineFile =
```

```
    AppDomain.CurrentDomain.SetupInformation.ActivationArguments.ActivationData[0];
```

Другой нюанс заключается в том, что местоположение файла передается в формате URI, как в file:///c:\MyApp\MyFile.testDoc. Это значит, что для получения действительного пути к файлу и очистке от защищенных пробелов (которые в URI транслируются в символ %20) понадобится следующий код:

```
Uri fileUri = new Uri(commandLineFile);
```

```
string filePath = Uri.UnescapeDataString(fileUri.AbsolutePath);
```

После этого можно проверить существование файла и открыть его, как обычно.

Параметры публикации

Как уже упоминалось, щелчок на кнопке Options приводит к открытию диалогового окна Publish Options с дополнительными параметрами. Список слева позволяет выбрать необходимую группу настроек. Ранее были описаны настройки из группы Description и File Associations. В таблицах ниже перечислены параметры из групп Deployment (Развертывание) и Manifests (Манифесты):

Параметры из группы Deployment

Параметр

Описание

*Deployment web page
(Веб-страница
развертывания)*

Устанавливает имя страницы установки при веб-развертывании (по умолчанию — publish.htm).

*Automatically
generate deployment
web page after every
publish
(Автоматически
генерировать
веб-страницу
развертывания после
каждой публикации)*

Если установлен (по умолчанию), веб-страница пересоздается при каждой публикации.

*Open deployment web
page after publish
(Открывать
веб-страницу)*

Если установлен (по умолчанию), Visual Studio запускает страницу установки в браузере после успешной публикации, чтобы можно было ее протестировать.

развертывания после публикации)

Use ".deploy" file extension
(Использовать файловое расширение .deploy)

Если установлен (по умолчанию), веб-страница установки всегда имеет расширение файла .deploy. Изменять этот параметр не следует, т.к. это расширение файла зарегистрировано на веб-сервере IIS и заблокировано, чтобы предотвратить подделку злоумышленниками.

For CD installations, automatically start Setup when CD is inserted
(Для установок из CD-дисков автоматически запускать установку при вставке CD-диска)

Если установлен, Visual Studio генерирует файл autorun.inf, чтобы обеспечить автоматический запуск программы установки немедленно после вставки CD-диска в привод.

Verify files uploaded to a web server
(Верифицировать файлы, загружаемые на веб-сервер)

Если установлен, процесс публикации загружает каждый файл после публикации для проверки возможности его загрузки. Если файл не может быть загружен, выдается уведомление, объясняющее проблему.

Параметры из группы Manifests

Параметр

Описание

Block application from being activated via a URL
(Блокировать активизацию приложения через URL)

Если установлен, пользователь сможет запускать приложение из меню Пуск только после установки, а не из веб-браузера.

Allow URL parameters to be passed to application
(Разрешить передачу

Установка этой опции позволяет приложению получать информацию URL от браузера, который запустил его, такую как аргументы строки запроса. Извлечь URI можно через класс ApplicationDeployment из пространства имен System.Deployment.Application. Для этого необходимо

параметров URL приложения) воспользоваться свойством ApplicationDeployment.CurrentDeployment.ActivationUri.

Use application manifest for trust information (Использовать манифест приложения для подтверждения информации) Если установлен, можно заново подписать манифест приложения после публикации. Обычно это делается для использования сертификата с именем компании. Данная информация появится в сообщении о доверии, которое пользователь видит при установке приложения.

Exclude deployment provider URL (Исключить URL поставщика развертывания) Если установлен, приложение будет автоматически проверять свое местоположение установки на предмет обновлений. Этот параметр можно использовать, если точное местоположение развертывания не известно, но, тем не менее, требуется автоматическое обновление ClickOnce.

Create desktop shortcut (Создать ярлык на рабочем столе) Если установлен, в дополнение к значку в меню Пуск процесс установки создаст значок на рабочем столе.