



Руководство для начинающих по созданию надстроек для SolidWorks



Создание коммерческого программного инструмента стало для меня ценным опытом, а также хорошей возможностью поделиться некоторыми трудностями, с которыми я столкнулся, в надежде помочь другим начинающим разработчикам API для SolidWorks, независимо от того, создаете ли вы инструменты для себя, для команды пользователей SolidWorks или являетесь предпринимателем с идеей. Хотя я пишу приложения для SolidWorks уже более десяти лет, я помню, насколько сложным казалось начало, и какие безграничные возможности открывались благодаря небольшим знаниям.

В этой, первой статье блога CAD Innovations, я решил начать с самого начала и подробно рассказать о шагах по созданию и выпуску надстройки для SolidWorks. В интернете много отличных ресурсов, охватывающих различные аспекты этого процесса, но я хотел бы собрать их воедино и добавить несколько советов, основанных на моем опыте, в надежде создать всеобъемлющее руководство «от начала до конца». Если у вас возникнут проблемы или вы захотите, чтобы я подробнее рассказал о какой-либо теме, оставьте комментарий ниже, и я добавлю больше информации или расскажу об этом в будущей статье блога.

Если вы когда-либо создавали, редактировали или даже просто запускали макросы SolidWorks (существуют тысячи бесплатных макросов, например, популярные макросы SolidWorks от Ленни), вы, вероятно, знакомы с их простотой – напишите несколько строк кода или скачайте чужой, запустите макрос и наблюдайте за волшебством. Создание и развертывание надстройки SolidWorks требует нескольких дополнительных шагов, но предоставляет решение, которое может легко интегрироваться с SolidWorks, включая возможность добавления пользовательских меню, значков, панелей инструментов, страниц диспетчера свойств, вкладок панели задач, выполнения действий по событиям и многого другого.

Давайте сравним некоторые сходства и различия между макросами и надстройками SolidWorks:

	Макросы	Надстройки
Язык(и) программирования	VBA*	VB/C# .NET, C++/CLI
Легко интегрируется в пользовательский интерфейс SolidWorks.	Нет	Да

	Макросы	Надстройки
Поддерживает асинхронные/ многопоточные процессы.	Нет	Да
Код скрыт от пользователей	Нет	Да
Поддерживается системами контроля версий**	Нет	Да

* Хотя использование .NET с макросами VSTA возможно, это не рекомендуется

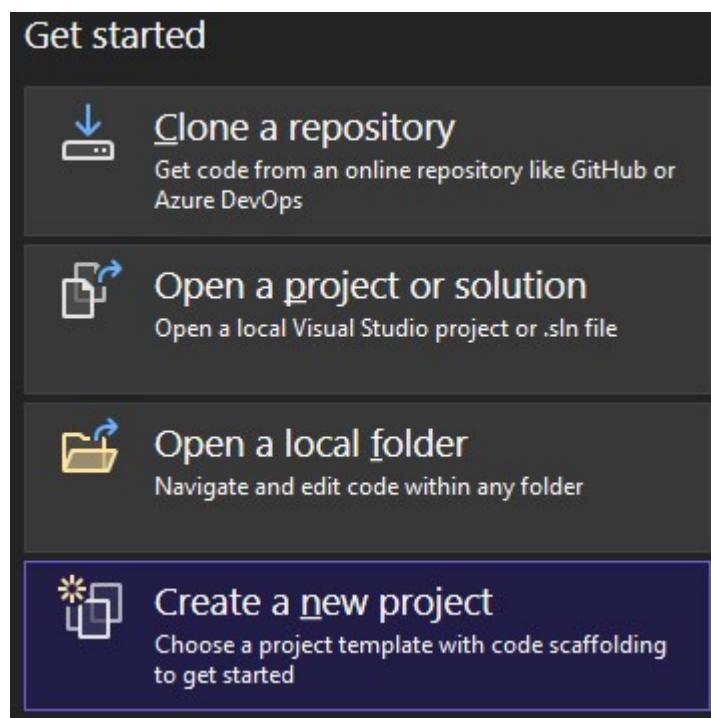
. ** Об этом будет рассказано в будущей публикации в блоге.

Для тех, кто читает это и имеет опыт написания макросов VBA (для SolidWorks или даже Excel), VB .NET, вероятно, является самым простым языком для освоения надстроек SolidWorks. Синтаксис похож на VBA, но с гораздо большим количеством функций и без некоторых запутанных и, казалось бы, лишних особенностей VBA. С другой стороны, если вы уже знаете (или готовы изучить) C# или C++, есть некоторые преимущества: благодаря повсеместному распространению, например, C#, в интернете значительно больше ресурсов, которые помогут, если вы столкнетесь с трудностями (особенно с более сложными проблемами). Visual Studio также поддерживает автозаполнение для C# (очень полезно), в то время как в VB .NET такой поддержки нет.

Для начала вам потребуется загрузить SolidWorks SDK с клиентского портала SolidWorks и Microsoft Visual Studio. Разрешенное использование версии Community Edition имеет некоторые ограничения, поэтому внимательно ознакомьтесь с условиями лицензии (в основном, компании с численностью сотрудников более 250 человек или доходом более 1 млн долларов не могут использовать бесплатную версию), но, по моему опыту, лицензия Professional имеет разумную цену. После установки обоих компонентов вы готовы... возможно. Шаблоны SDK редко появляются в списке шаблонов нового проекта Visual Studio сами по себе, но, к счастью, есть простое решение .

Теперь вы готовы создать свою первую надстройку для SolidWorks.

Откройте Visual Studio, выберите «Создать новый проект».



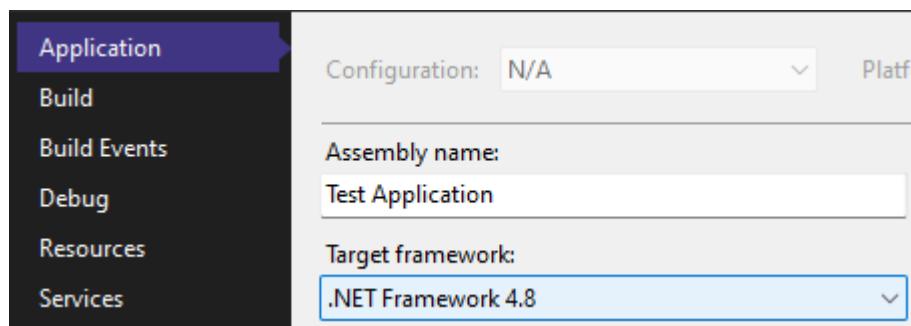
Прокрутите вниз до шаблонов надстроек SolidWorks (опять же, для тех, кто имеет опыт работы с VBA, рекомендуется использовать VB .NET) и нажмите «Далее». Присвойте проекту имя и нажмите «Создать». Пример надстройки, предоставляемый SDK SolidWorks, включает весь необходимый код для регистрации надстройки, подключения к SolidWorks, добавления примеров пунктов меню и настройки страницы диспетчера свойств. Пример также включает класс обработчика событий с доступом к часто используемым событиям SolidWorks (таким как открытие, закрытие и сохранение документов).

Если вы когда-либо копались в реестре Windows, вы могли заметить, что у каждого приложения есть буквенно-цифровой идентификатор. Эта случайно сгенерированная строка из 36 символов называется глобально уникальным идентификатором (GUID) и гарантирует, что двум приложениям не будет присвоено одинаковое значение (отсюда и название «глобально уникальный»). Вам нужно будет сгенерировать собственный GUID для вашего приложения, что легко сделать благодаря веб-технологиям. После того, как вы создали GUID и скопировали его в буфер обмена, замените GUID в

верхней части файла `SwAddin.vb`(или, `SwAddin.cs`если вы создали надстройку на C#) в вашем проекте Visual Studio, как в этом примере на C# из надстройки Versa Note:

```
[Guid("BB75C0FD-3720-45E5-8B80-2ACB34900F7E"),  
ComVisible(true)]  
[SwAddin(  
    Description = "Versa Note AddIn",  
    Title = "Versa Note",  
    LoadAtStartup = true  
)]
```

В зависимости от версии Windows и другого установленного программного обеспечения проект может быть не настроен для установленной версии .NET. Чтобы проверить это, щелкните правой кнопкой мыши по имени проекта в обозревателе решений Visual Studio и выберите «Свойства». На вкладке «Приложение» убедитесь, что в раскрывающемся списке выбрана «Целевая платформа».



Последняя версия .NET Framework – 4.8, и, как правило, это самый безопасный выбор. Если вы хотите обеспечить максимальную совместимость между системами в вашей организации без необходимости обновления .NET Framework на каждой системе, вы можете выбрать самую низкую доступную версию Framework (или даже установить более старую версию Framework на вашу систему разработки). После выбора целевой платформы вам будет предложено подтвердить изменение; нажмите «Да», чтобы продолжить.

Теперь вы готовы создать и протестировать свой первый надстройку для SolidWorks. Если на компьютере, на котором вы установили Visual Studio, также установлена SolidWorks, вам просто нужно выбрать конфигурацию решения «Release» (которая должна находиться на панели инструментов «Standard» в Visual Studio) и в меню «Build» выбрать «Build Solution». Если сборка завершится без ошибок, откройте SolidWorks и откройте меню «Tools». Вы должны увидеть надстройку с именем «VB Addin» или «C# Addin» (отображаемое имя надстройки можно изменить в файле SwAddin.vb(или SwAddin.cs)):

```
#region UI Methods  
public void AddCommandMgr()  
{  
    ICommandGroup cmdGroup;  
    if (iBmp == null)  
        iBmp = new BitmapHandler();  
    Assembly thisAssembly;  
    int[] cmdIndex = new int[4]; //, cmdIndex1;  
    string Title = "C# Addin", ToolTip = "C# Addin";
```

На этом пока всё. Буду рад услышать ваши отзывы и предложения по темам для будущих публикаций. Учитывая бесконечное количество применений надстроек SolidWorks, ответить на каждый возможный вопрос сложно, но я продолжу отбирать интересные моменты из процесса разработки Versa Note, уделяя особое внимание аспектам, специфичным для SolidWorks (ведь именно поэтому вы здесь, верно?).

Блог



Ли

[СЛЕДУЮЩАЯ ЗАПИСЬ >](#)

Повышение качества пользовательского опыта и демонстрация возможностей разработчиков.



Обратная нота

Надстройка SolidWorks Drawing & MBD Notes



Быстрые ссылки

[Дом](#)[Скачать](#)[Цены](#)[Бета](#)

Поддерживать

support@cadinnovations.ca

Часто задаваемые вопросы

Руководство пользователя

Помощь в настройке

**CERTIFIED
Solution
Partner**



Авторские права © Versa Note принадлежат CAD Innovations . Все права защищены.