



AUTODESK UNIVERSITY 2015

SD9662

Creating Professional Plug-ins for Autodesk Exchange Store

Fernando Malard
ofcdesk, LLC

Learning Objectives

- Learn how to create good-looking and professional plug-ins
- Discover the Autodesk Exchange Store market
- Discover how important it is to take care of small details
- Learn how to improve the design and capabilities of your plug-ins

Description

This class will explain how developers can create stable, successful, useful, and professional-looking plug-ins for the Autodesk Exchange store. We will explore basic elements, such as well-designed icons, friendly dialogs, and consistent user feedback and support. Finally, we will explain the Autodesk Exchange store publishing workflow, and we will cover a few tricks about how to market your plug-in.

Your AU Experts

Fernando Malard is a civil engineer who has worked with AutoCAD software and ObjectARX technology since 1996 and with Revit software since 2009. He has also been an Autodesk Developer Network member since 1997. He has worked on several AutoCAD and Revit applications for civil engineering, architecture, interior design, and geographic information system (GIS) using ObjectARX technology, C++, Microsoft .NET, JavaScript, Cocoa, and databases. Fernando has had extensive experience teaching AutoCAD, Revit, C++, Microsoft Foundation Class, Microsoft .NET, and ObjectARX technology over the last years. Today he continues to apply his skills to the design and implementation of complex Industry Solutions. Fernando has also worked with Autodesk User Group International (AUGI) communities, and he maintains a blog about ObjectARX technology. He holds a master's degree in structural engineering from the Federal University of Minas Gerais, Brazil.

Learning Objectives	1
Description	1
Your AU Experts	1
Introduction	3
App Store strategy.....	3
Market Analysis.....	4
.....	5
.....	5
Product Requirements.....	6
Company and Product Identities.....	6
Product Category: Trial, Free, Paid and Subscription	6
Professional Looking and UI Experience.....	6
Strong Platform Integration	7
Stability and Robustness	8
Product Deployment	8
The Bundle Package	8
Package Contents file	9
Documentation	11
Payment policies	12
Payment options – BlueSnap or Paypal.....	13
Product Support	14
Videos and Training	14
Protection.....	14
Hear the Community	15
Updates and new versions	16
Conclusion.....	16
Noah's Ark Syndrome.....	16
What's next?	17



Introduction

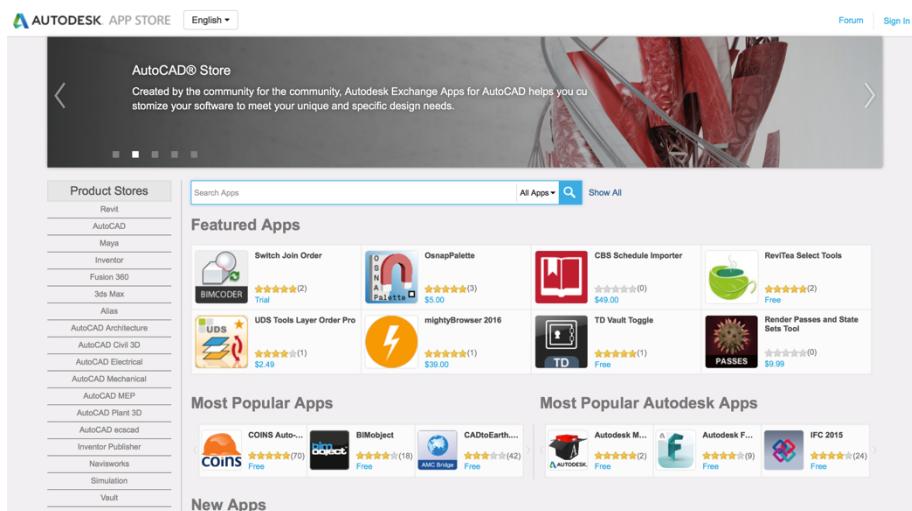
Autodesk launched **Autodesk Exchange Store** Apps back in **2011** to create a website where Developers can sell and distribute apps, content, training and any other digital material to the Autodesk user community. It was inspired by Apple App Store created in 2008 when Apple decided to create a centralized place to distribute Developer created applications.

The concept of an **Application Store (App Store)** is based on a convenience aspect of providing a large collection of applications through the same delivery pattern, presentation and technical information like compatibility, features, screenshots and any other further information relevant to the user's decision.

As any other market, the Exchange Store relies on the quality and variety of published applications to attract more users to browser through the store web pages and research more information about the products. From the user's perspective it is indeed very convenient to have a single trusted place to acquire or test applications.

At the beginning, Autodesk decided to prioritize **AutoCAD** as the Store's showcase section because it is by far the most known product Autodesk distribute. Later other widely used platforms like **Revit**, **Inventor** and **3ds Max** was added to the store. Today the Store has about **18** different Autodesk product categories where you can publish your application or content. Autodesk also decided to follow the same principles Apple did with the App Store, every single application submitted is audited by a **Developer Team** to assure its minimum quality, proper work and minimum stability.

The Store itself is today a great and popular place to showcase your plugins targeting a rich and selected audience where your sale success is way higher when compared to ordinary advertisement mechanisms like **Google**, **Bing** or **Yahoo** because they target any user browsing through their website then applying filters that aren't necessarily useful for you. In 2015 the Exchange Store was rebranded as "**Autodesk App Store**".



App Store strategy

The Store website primarily created for selling applications can also work as a powerful marketing tool for you. Applications today may need to prove their efficiency, quality and deliverables before the customer agree to pay anything for it. The easiest way to do it is by providing **Free or Trial** versions of



your product where features and be **test-driven**, “tasted” and also put to test into real world scenarios. If they succeeded the chances your potential customer will agree to buy it will increase. A hidden outcome from the very act of looking and testing your product is the attention brought to you or your company when the user is visiting your page, installing and testing your application.

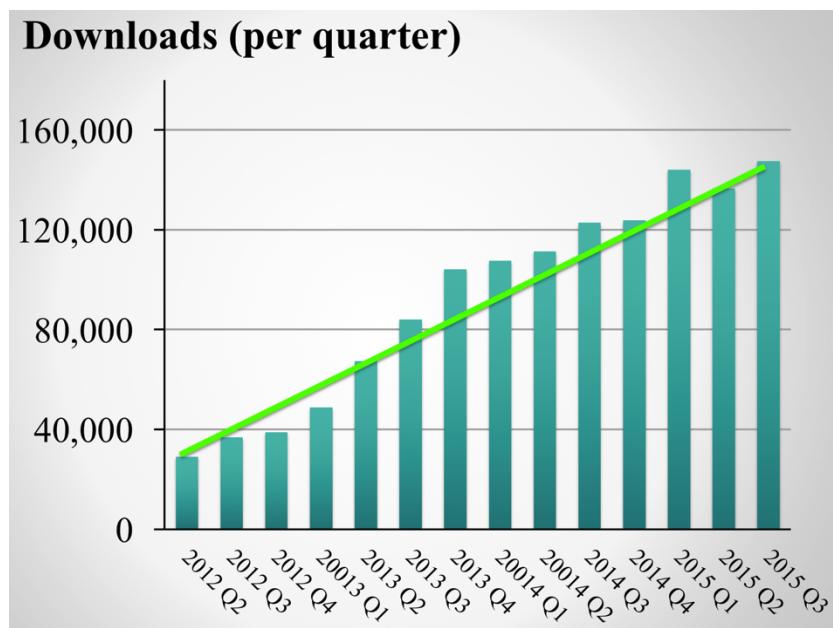
The leads generated by user downloads are made available by the Store through simple spreadsheet reports available by published product. You can use those emails to directly communicate with the users and evolve your contact by providing better communication experiences like phone calls, live demos or even a visit to their company or office.

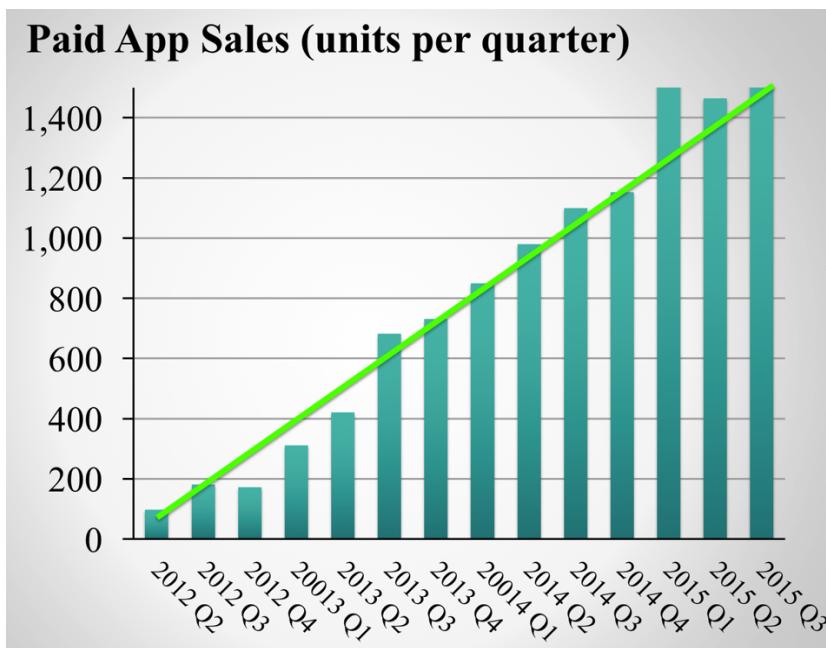
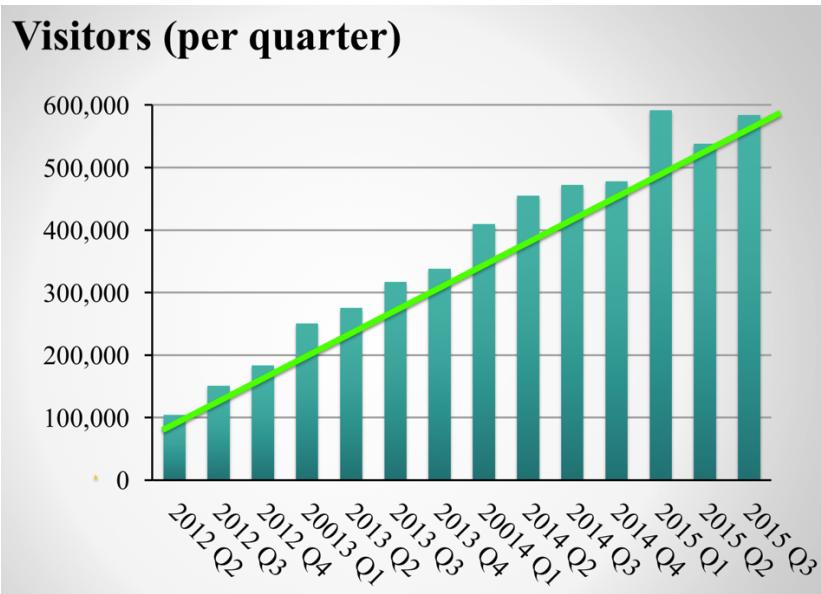
It is very important to understand there is no miracle recipe for the success but you need to work hard to increase your chances to achieve it by following the Store suggested procedures and adopting a minimum level of quality, good looking and efficiency of your published application or content.

The App Store can be also used to reach entirely new markets and languages, which would be very hard to do your own. The App Store does support Multilanguage products when specific **localized Stores** will show your application or content specifically tuned for that audience. At some countries it is very important to deploy products made on their specific language so they feel more natural and intuitive for them. The additional work necessary to create and deploy Multilanguage application or content may pay off if you reach the right market for your product thus provide a strong and fast revenue. Current supporter languages at the App Store are: **Czech, Deutsch, English, Spanish, French, Hungarian, Italian, Japanese, Korean, Polish, Portuguese, Russian, Simplified Chinese and Traditional Chinese**.

Market Analysis

The App Store is increasing on aspects like number of visitors, applications, downloads and purchases. We can consider it today as the first source of Autodesk plugins around the world. Here are some statistics about the Store and it is very important for you to pay attention to the details and chart trends so you can properly build your product launch strategy:





Product Requirements

So far we talked about the market and this can feed your brain with lots of ideas. To be able to benefit from this eager market you will need to accomplish some basic requirements and pay attention to some very important details.

Company and Product Identities

If you plan to not only publish a single product at the App Store but also to keep doing that for a while, with more products focusing different audience and running at the top of other **Autodesk** platforms, get off on the right foot. Users in general appreciate organized companies and the very first impression is very important. A strong and well-designed company logo and also a consistent brand for the products is the minimum effort you must do to please users and look at you as a trustworthy company or individual.

You can create your company and products brand images alone or hire a professional to do that for you. **Technical** people tend to see **Design** services cost as something expensive and a simple embellishment but at the end it is your image! Don't be afraid to sit and spend some valuable time thinking about your brand, it will pay off for sure! If you have problems finding such a professional individual or company, go to websites like Freelancer.com and hire professionals already rated and reviewed by several other contractors.

Make sure you don't use comics or colorful identities because they are hard to stick at the user's minds. Go for something simple, clear, sharp and strong. Users will keep that in mind and will eventually identify your company as soon as they see a simple product icon. They will feel good about it, psychologically speaking, and will be interested to discover more about your products.

Product Category: Trial, Free, Paid and Subscription

So far we talked a lot about market, users, locations and your brand but what exactly you are trying to sell? Do you want to sell volume, high-priced products and subscriptions? The App Store does support any of these Business Models and you can start with something quick and easy, measure results and adapt. Software market is somehow ungrateful. We can see innumerable companies creating really fancy applications but not having best selling products.

Sometimes users need to test pilot your product before deciding they really need them and they are ready to buy. It is a crazy science and each user will have his mature timing affected by corporate decision process, bureaucracy, cash flow or even personal emotions. It is very important to let them taste your product as soon as possible. Don't be greedy allowing them to test just a few features of your product for a small period of time. You should provide a useful Trial or Free (limited) version that can at least help the user a little bit for a while.

Software subscription is the new trend and we can see companies like Autodesk and Microsoft investing hard on this market approach. If your product belongs to a Business Model like this, go for it. We can clearly see that Software business itself is passing through a serious change for the next 10 years or so. Don't stick with a strategy, be dynamic and adapt fast.

Professional Looking and UI Experience

It won't help if you have a beautiful company logo, product identity and good marketing material if your UI (user interface) is really bad. You should not reinvent the wheel. Just follow your host Platform style while creating your **UI** elements so it will smoothly integrate into the host UI thus providing the users a



fluid and consistent experience. Further, don't be negligent with your UI layouts, control and picture alignments, good written text, proper language translations and any other small aspect you tend to skip. It seems stupidity to spend time on those aspects but they are strong pillars of your product as a whole successful user experience.

Be clear when communicating to the user through well written messages, relevant icon identities to communicate error events, warning and succeeded operations. Be consistent throughout your other products so users don't have any extra challenge adapting to your product style. Make sure you don't use **automated language translators** to create localized versions of your products. Hire someone capable of creating good quality translations consistent and into the context of your product technical details. Literal translations are in general very poor and not easy to understand by the native-language speaker.

Pay attention to small things like font family, font size, and button size, use simple and easy to use UI controls. Don't try to revolutionize the market presenting a new clunky user control that deviates attention from the product itself to the control operation. If you consider all those aspects the chances you have a good user audience are really high.

Strong Platform Integration

The idea behind any Plugin is a tight and smooth integration with the host platform so it extends the native features thus providing more capabilities to the end user supporting more specific user needs. That said, your product should not only be consistent with the host platform graphical appearance but also work seamlessly with the host features by extending them.

Following information only applies to version(s) compatible with Windows.

Autodesk Product Compatibility
Select each product and version that is compatible with your app.

<input type="checkbox"/> M&E (Select All)	2016	2015	<input checked="" type="checkbox"/> 2014	<input type="checkbox"/> 2013
<input type="checkbox"/> Alias (Select All)	2016	2015	<input checked="" type="checkbox"/> 2014	<input type="checkbox"/> 2013
<input type="checkbox"/> AutoCAD (Select All)	2016	2015	<input checked="" type="checkbox"/> 2014	<input type="checkbox"/> 2013
Autodesk AutoCAD	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Autodesk AutoCAD Architecture	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Autodesk AutoCAD Civil 3D	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Autodesk AutoCAD Electrical	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Autodesk AutoCAD Mechanical	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Autodesk AutoCAD MEP	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Autodesk AutoCAD P&ID	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Autodesk AutoCAD Plant 3D	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Manufacturing (Select All)
 Inventor (Select All)
 Revit (Select All)
 Simulation (Select All)
 Vault (Select All)
 AEC (Select All)

If your Plugin requires a different work process when compared to the host you may end up confusing the users and make the adoption of your product a little harder. Further, it is recommended that your Plugin support the same latest versions Autodesk does support at the portal. Usually most of Autodesk products are supported through their latest 3 versions so it is highly recommended you do the same. Host platforms working in both 32 and 64-bit machines would also require your Plugin to support them.

When you publish your Product it can be compatible to totally different Autodesk products because sometimes the features it carries on are way beyond the Host product itself. In this case, the App Store allows you to select multiple product Families, versions and Platforms so you can make the product publishing work just once.



Stability and Robustness

When creating your Plugin make sure you adopt the most known good coding practices and take into account extreme use cases your users may face in real scenarios. We tend to assume absurd things won't ever happen but I can assure users will be very creative hitting buttons or keys when you never expect them to do so. By supporting more Host Platform versions, different **System Architectures** and also different **Operating System** version, the work of test-proofing your Plugin will be hard and time consuming. Make sure your test your Plugin into those different scenarios as many times as possible. I know it is really hard to do that but with a systematic approach and discipline you can do that.

It is almost impossible to assure stability and robustness if you don't play with the product on those different scenarios. In other hand, having all those different scenarios configured and ready to perform your tests without people behind is very challenging. During the years I've been trying to achieve the ideal solutions for this problem but today I think it was solved quite well. What you should do is to use **Virtual Machines** (VM). VMs are ideal to solve this problem because you can have as many as your local (internal and external) storage allows you. Once you build you several template machines with your Autodesk Host Platform installed, you will have literally a playground at your service. I can mention here two different options depending on your computer's operating system:

- **Windows:** VMWare Workstation (<http://www.vmware.com/products/workstation/>)
- **Mac OSX:** Parallels Desktop (<http://www.parallels.com/products/desktop/>)

Either solution will be your redemption but I personally prefer Parallels running into the Mac OS hardware due its high performance, stability and average price range. I've been using Parallels for a long time and it pays off. A great feature of those VMs is the **Snapshot** where you can save your machine state just before doing something may affect it. So before performing the action you generate a Snapshot, do your tests as harmful they might be, and then restore it to its original state. It is recommended you keep those machines, as clean as possible and without any developer tool your users probably won't have.

Product Deployment

The App Store requires you to prepare your product to be delivered to different machine configurations, languages, Host Applications and to be secure to install.

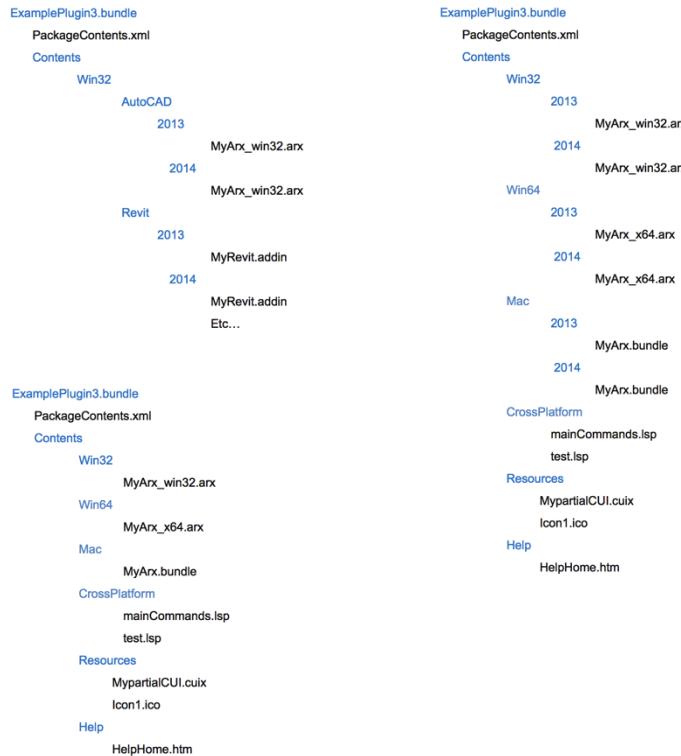
The Bundle Package

To simplify and unify the software deployment process Autodesk adopted the "bundle" approach to distribute your Plugin feeding different Autodesk Products at once. All the necessary files like DLLs, images, documents, media or any other required file will be packed into this bundle. The Bundle structure can have nested folders and you can structure it the way you need to properly store your application files.

When the user downloads it from the Autodesk App Store the host product will properly install it thus sharing the entire application with all compatible products. If you are targeting too many products at once you need to consider common features that are platform independent and pack them into a single shared module so all your specific Plugin versions can consume it. This is a good approach because you won't have to duplicate the same code or content across the platforms. In other hand, once you change this shared portion and publish an update it can affect all other front ends into different Autodesk



Products. Here we can see 3 different types of Bundle: Simple AutoCAD ARX Bundle, AutoCAD + Revit Bundle and AutoCAD (Mac OSX + Windows) Bundle.



Package Contents file

Each Bundle is drive by its description file always named as “**PackageContents.xml**”. This file, a **XML** (Extensible Markup Language), describes how the Autodesk Host application will load the Plugin and its companion files. It does support language options, menus, DLLs, content files and any other file you want to pack into the Bundle.

There isn’t currently any application to create this file so you need to create it manually through a text editor such as **Komodo**, **Notepad++** or even **Notepad**. If the editor does support XML formatted files you will get the color support for markups.



```

<?xml version="1.0" encoding="utf-8" standalone="no"?>
<ApplicationPackage SchemaVersion="1.0" AppVersion="3.32.3004" AutodeskProduct="AutoCAD"
    Description="Autodesk BIM 360 Glue AutoCAD 2015 Add-in 64 bit" Author="Autodesk"
    ProductCode="{9D589081-AFC2-4932-9071-AC585AC1EA83}"
    UpgradeCode="{52C3255E-42DC-4437-A4D2-B81E458F0960}"
    Name="Autodesk BIM 360 Glue AutoCAD 2015 Add-in 64 bit">
<RuntimeRequirements OS="Win32|Win64" Platform="AutoCAD|AutoCAD*" SeriesMin="20.0" SeriesMax="20.0" />
<CompanyDetails Name="Autodesk" Email="milin.trivedi@autodesk.com" />
<Components Description="Win32 parts">
    <RuntimeRequirements OS="Win32" Platform="AutoCAD|AutoCAD*" SeriesMin="20.0" SeriesMax="20.0" />
    <ComponentEntry AppName="BIM360Glue" AppType=".Net" Version="3.32.3004" ModuleName=".Contents\Win32\BIM360GlueAutoCAD2015Addin.dll"
        AppDescription="Main .NET Module" LoadOnCommandInvocation="True">
        <Commands GroupName="ADESK_BIM360GLUE">
            <Command Local="ShowBIM360" Global="ShowBIM360" Description="command to show BIM360 plugin" />
            <Command Local="HideBIM360" Global="HideBIM360" Description="Command to hide BIM360 plugin" />
            <Command Local="Glue" Global="Glue" Description="Glue" />
            <Command Local="ClashPinpoint" Global="ClashPinpoint" Description="ClashPinpoint" />
        </Commands>
    </ComponentEntry>
    <ComponentEntry AppName="BIM360Glue" Version="1.0.0" ModuleName=".Contents\Win32\BIM360GlueAddinsCommon.dll"
        AppDescription="BIM360GlueAddinsCommon" AppType="Dependency" />
    <ComponentEntry AppName="BIM360Glue" Version="1.0.0" ModuleName=".bim360.cuix" />
</Components>
<Components Description="Win64 parts">
    <RuntimeRequirements OS="Win64" Platform="AutoCAD|AutoCAD*" SeriesMin="20.0" SeriesMax="20.0" />
    <ComponentEntry AppName="BIM360Glue" AppType=".Net" Version="3.32.3004" ModuleName=".Contents\Win64\BIM360GlueAutoCAD2015Addin.dll"
        AppDescription="Main .NET Module" LoadOnCommandInvocation="True">
        <Commands GroupName="ADESK_BIM360GLUE">
            <Command Local="addHorizontalRibbon" Global="addHorizontalRibbon" Description="addHorizontalRibbon" />
            <Command Local="removeHorizontalRibbon" Global="removeHorizontalRibbon" Description="removeHorizontalRibbon" />
            <Command Local="Glue" Global="Glue" Description="Glue" />
            <Command Local="ClashPinpoint" Global="ClashPinpoint" Description="ClashPinpoint" />
        </Commands>
    </ComponentEntry>
    <ComponentEntry AppName="BIM360Glue" Version="1.0.0" ModuleName=".Contents\Win64\BIM360GlueAddinsCommon.dll"
        AppDescription="BIM360GlueAddinsCommon" AppType="Dependency" />
    <ComponentEntry AppName="BIM360Glue" Version="1.0.0" ModuleName=".bim360.cuix" />
</Components>
</ApplicationPackage>

```

This example shows how modules, platforms, commands, files, menus and versions are well defined so the Host Application can easily find its appropriate set of files to load. Note that file paths beginning with “./” refer to files placed inside the Bundle subfolders so it doesn’t matter where the Bundle is. The contents should be also separated by platform so your product may provide the appropriate binary file to **32 or 64-bit** systems.

AutoCAD-based products are localized into a wide range of languages and the structure of the ***PackageContents.xml*** file supports these different languages with locale codes. Many of the attributes in the ***PackageContents.xml*** file support localized languages. Append a locale code to the end of an attribute name to define a localized version of the attribute. For example, to define a **Spanish** version of a description in the **ApplicationPackage** element you would create an attribute named **DescriptionEsp**.

The following is a full list of all supported locale codes:

- **Chs** - Chinese Simplified (PRC)
- **Cht** - Chinese Traditional (Taiwan)
- **Csy** - Czech
- **Deu** - German
- **Enu** - English
- **Esp** - Spanish
- **Fra** - French
- **Hun** - Hungarian
- **Ita** - Italian



- **Jpn** - Japanese
- **Kor** - Korean
- **Plk** - Polish
- **Ptb** - Portuguese
- **Rus** - Russian

Documentation

Each Product you create should have its companion documentation. Nowadays most of software documentation is made available online through the author's website. The App Store does require you to provide documentation for each product you publish. It doesn't need to be a book about your product but you need to expose most of its features into the documentation thus allowing users to read and understand your product before downloading or purchasing it. The documentation **needs to match the languages** you plan to support in your application. For each different language a new documentation set (including short descriptions, images, limitations, etc.) will be required.

The Autodesk App Store is sensible to the visitor's browser language so it will display the appropriate version of your documentation. Again, as mentioned before, take care of a good translation from your native language and don't use automatic translators. The documentation will also contain some screenshots of your product so they will need to be taken from the corresponding language version of your UI. Some Developers prefer to deploy only the English version. It is in fact a first good step but if you really want to spread your product through out another country, make sure you deliver everything well done on its native language.

Countries with strong culture like China, Japan, Spain, France and Italy tend to prefer software with everything running with their native languages. Pay attention to the languages Autodesk supports because if they do the chances your product need to do so are really high.

Make sure you capture precise and sharp images of your interfaces. I would recommend you to capture them in high-resolution screens and save them with low compression rates so you avoid the images to get distorted.

The screenshot shows the 'App publishing - English, Win32 and 64' submission page. The form is divided into several sections:

- Step 1 of 2**
- Please fill in following information in English.**
- App Name***: Geometry Merger Preview
- App Short Description**: Merge adjacent 3DFaces into a Mesh; Extract Surfaces and Solids from Meshes – Preview Edition
- App Description**: Description to be displayed for this item on the product page (4000 characters). Includes a rich text editor and note: "Note: this is a Preview edition. It does everything that the full Geometry Merger app does, but wraps the resulting drawing changes in custom objects which display in your edit session, but cannot be saved, copied or plotted."
- App Version***: 1.0.0.6 (Latest)
- Version History**: Help file changes. Includes an 'Edit' button.
- App File***: A file named 'ofcdeskicGeometryMergerPreview.msi' (5.6 MB) is uploaded.
- Buttons at the bottom:**
 - This Submission includes a version change.
 - This Submission includes no version change.
 - Remove (for the uploaded file).



Payment policies

The Autodesk App Store currently supports 4 methods of payment:

- **Free:** Your product does not require any payment and can be freely downloaded;
- **Trial:** Your product works without payment but it may contain limitations while as unpaid version;
- **Paid:** Your product has a fixed price so users will only be able to download it after they pay the amount of money setup;
- **Subscription:** Your product requires a recurring subscription charged monthly or yearly.

The Subscription policy once set cannot be changed as you can with all other payment methods. Sometimes you want to give your product for free on its first release and then decide to start charging for it. Users won't like game rules being changed while playing so I would recommend you stick with the initial price policy.

Price *
Product price, to be displayed on the product page.

Free Trial Paid Subscription

Price (USD)

Note: Please use periods (.) to indicate decimals; for example: 1.99. You will not be able to change the payment type once "Monthly recurring subscription(USD)" is selected.
Notice: Without a PayPal/BlueSnap account, you cannot set a price when publishing an App. Please make sure to use your master PayPal email address or BlueSnap Credentials.

Payment Platform
Please select a payment platform if you want to publish a paid App.

BlueSnap (Recommended) PayPal

Offer free 30-day trial of this app
Your app needs to have Autodesk App Store entitlement API integrated to enable this function.

Price *
Product price, to be displayed on the product page.

Free Trial Paid Subscription

Pricing Plans

1 month (recurring)	<input type="text" value="19.99"/>	USD
1 year (recurring)	<input type="text" value="199.99"/>	USD

Note: Please use periods (.) to indicate decimals; for example: 1.99. You will not be able to change the payment type once "Monthly recurring subscription(USD)" is selected.
Notice: Without a PayPal/BlueSnap account, you cannot set a price when publishing an App. Please make sure to use your master PayPal email address or BlueSnap Credentials.

Payment Platform
Please select a payment platform if you want to publish a paid App.

BlueSnap (Recommended) PayPal

Offer free 30-day trial of this app
Your app needs to have Autodesk App Store entitlement API integrated to enable this function.

From time to time you may want to do **promotions or discounts**. To do that, simply edit your product price at the App Store. Once the promotion / discount ends you can edit the price again. The purchased product is bound to the user's login at the App Store so he will capable of installing (without being charged again) your product into a different machine if he uses the same credentials.



Payment options – BlueSnap or Paypal

The payments are currently made through **BlueSnap** or **PayPal**. The App Store will take care of the payment process (it's a huge time saver) and will communicate with your merchant payment account.

While PayPal is more appropriate for single developers and small business, BlueSnap has more features and works based on **Purchase Orders** (PO). It is important to understand all payment methods are going to charge fees per transaction so it is very important to select the method which is suitable for your current business and marketing plans. The integration of these payment methods are well described on App Store website.

BlueSnap: Is a global payment gateway that allows your customers to pay for your app in the Autodesk App Store marketplace with a wide range of payment options making it easier for customers worldwide to buy your app. You can register your account through the following URL:

<http://home.bluesnap.com/ecommerce/partner-referral-autodesk>

The screenshot shows the BlueSnap website with a blue header bar. On the left, the BlueSnap logo is visible. To the right of the logo are navigation links: Features, Pricing, Snap Center (with a dropdown arrow), About, Support, Docs, Login (with a dropdown arrow), and a 'Get Started' button. Below the header, a large blue section contains the text 'Welcome Autodesk Exchange Apps Partners!' and a sub-instruction: 'Please submit your information below and a BlueSnap representative will contact you.'

PayPal: Is by far the most used payment process across the web. It is primarily target to individuals and small business so the registration process is simple and fast. If you still don't have a good idea where do you plan to go with your app selling strategy it is a cheap and fast payment option. The PayPal service can offer strong integration when we talk about payment options. It also provides several local payment options for several countries thus making the purchase process smooth regardless the user geographical location.

The screenshot shows the PayPal sign-up page. At the top left is the PayPal logo. To the right is a 'Log In' button. Below the logo are three circular profile pictures of smiling people. To the right of these images is a column of text: 'Sign up for free.', 'Choose from 2 types of accounts:', and two radio button options: 'Personal Account' and 'Business Account'. The 'Personal Account' option is selected. Below each option is a brief description. At the bottom right is a 'Continue' button.

See for yourself why 169 million people love PayPal



Optionally, you can setup an **IPN** (Instant Payment Notification) so the provided callback URL will be fired when the purchase occurs. This would allow you to improve the integration of purchases with your own custom procedures and data processing. It is very important to set the price levels accordingly to the store average price for products like yours. If you put a price too high, it will certainly make users to stay away from your product. Sometimes it is better to sell more using a low price than sell less using a high price. It is your decision though; just try to be consistent across the time.

Product Support

A published Product will require you to deliver much more than the product itself. The ultimate user experience goes way beyond the software and requires good documentation, demonstration, training and support.

Videos and Training

The new generations adopted the video as the main source of learning activities. Today some websites like **YouTube**, **Vimeo** and the recent **Autodesk Screencast** are flooded with videos about everything. There isn't any better documentation than a clear and objective video about your software. You can create small video clips emphasizing specific features of your product and exploring all the potential uses of your technology. To enrich the experience, try to couple these videos with real scenario situations proving that what you are trying to sell was created from a real demand.

Videos can be boring or very useful to the end user. Plan a good script and have the appropriate material to be used during your video. Try to be as much professional as you can by eliminating family images from the background, other application icons from your desktop, personal folders and information, etc. A good tip to avoid those distractions is to create a secondary login into your operating system to be fully dedicated to your demo purposes. If you can afford and use two monitors, you can record your videos in full screen at one monitor while running other support tools at another monitor.

If your product requires specialized training, you can also do it via demo videos or even use online training sessions. For small size companies or even individuals, it is sometimes hard to hire a professional to record the narration of your demo videos. In other hand, we can see more and more informal videos where the narrator is the author putting himself at a natural and fluid position. The positive side of this approach is that a person actually known about what he is talking transmits much more confidence to the listener.

Protection

Currently the Autodesk App Store website doesn't provide publishers any direct protection to their products against piracy. As said before, the payment process is secure and you won't face any problems with that. In other hand, an expert user can manually copy your product files after installation from the machine he purchased it to another machine so other user can use it as well without paying you a second license. Again, as a small company or independent developer, you may not have the necessary resources to implement a robust protection system or buy a protection software library to bind to your product. You will have a list of user's emails who purchased your product so you can at least filter them during your training sessions, support communication, etc.

If you have money to invest, your product is expected to sell a lot and you don't want your product to be copied around, go ahead and buy some protection library. Once you empower your product with this



protection system you can perform a post-sale action and send the user who purchased your product, instructions to activate it. Make sure you give him at least a good 7 days to activate the product because he may be in transit on a business trip or during the weekend browsing the App Store.

Autodesk has a simple protection system called “**Entitlement API**”. It is basically a web REST API that respond to your request to check if a specific user is entitled to use your application. This API will basically check the user credentials with your product identification. Here is a basic example:

Base URL: <https://apps.autodesk.com/>
End Point: webservices/checkentitlement
Http Method: GET
Parameters: ?userid=***&appid=***
Return: JSON object.

Here is a sample request URL:

<https://apps.autodesk.com/webservices/checkentitlement?userid=2N5FMZW9CCED&appid=appstore.exchange.autodesk.com%3ascreenshot%3aen>

The userid is “**2N5FMZW9CCED**”, appId is “**appstore.exchange.autodesk.com%3ascreenshot%3aen**”.
The return JSON is:

```
{
  "UserId" : "2N5FMZW9CCED",
  "AppId" : "appstore.exchange.autodesk.com:screenshot:en",
  "IsValid" : false,
  "Message" : "Ok"
}
```

The ‘**IsValid**’ value will be true if the user has entitlement to the App. Otherwise, IsValid will be false.
Possible return messages can be any one of the following):

- “OK” - current call is correct
- “Invalid parameters(s)” - userid or appid is not set
- “Please use https” - the request is not using https

Inside AutoCAD you can get the user ID from the “**ONLINEUSERNAME**” or “**ONLINEUSERID**” variables.
Inside other Autodesk products you will need to use Autodesk **OAuth** to get it. The **appid** is your product ID extracted from its page URL at the App Store.

Hear the Community

We all know that a Product is nothing without its users. When we decide to create an application to complete and extend an existing Autodesk product we certainly know exactly what needs to be done. In other hand, all we know is a result about our specific perspective about a problem and the measures to solve or mitigate this problem. Sometimes other users may have a slightly different perspective about the same problem and they may also have a better idea than yours.

The humble ability to hear your users is great advantage to shape the future about your product. Autodesk itself changed the approach a long time ago and started to valuate the user community doing recurring Researches about user demands and needs. Once you publish your product and users start to



consume it you will begin to receive some feedback. Some will criticize your product; others will give you compliments and request more and more.

The Autodesk App Store does provide a **user review panel** inside your product's page where they will be able to post comments, rate your product and also see other user's posts. First, you need to respond any question they submit as much polite you can. Secondly, you need to be smart and collect valuable information from their feedback like common requests, common complaints and any other relevant information regarding your product. If you systematically do that and consider the relevant requests as your main requests for the next version the chances of captivate those users and start a strong community around your product are really high.

Updates and new versions

We tend to consider our particular problems the most important in the world. This is how your product's user will feel when they find a bug or problem with your product. They will be affected emotionally and will respond sometimes with roughness. You need to respond as quick as you can with a temporary remedy for the problem and promise an update as soon as you can. If your communication is constant and honest with the users, they will be confident and loyal to you and your products.

It is fair to say the users see updates today with a little lack of confidence because more and more cases of an update fixing one issue and creating two new are happening. You don't want to do that so be careful while fixing something and do a complete test to validate your product again. Depending on the audience, you can plan regular updates, say one each month or even two each month. A completely new version must have something that justifies that. Make sure you control your product version by using 4 or 3 digits' group number so you can represent main updates, major fixes and minor fixes.

Autodesk does promote early product launches when they are about to release updates of products like AutoCAD, Revit, Inventor, etc. You have an opportunity to early test your product before they launch the Host Product and be able to prepare to launch your update at the same time. The marketing impact is really good and you may leverage your sales by being an earlier adopter of the new version. Check Autodesk Beta Programs and sign up.

Conclusion

To wrap up all the concepts and ideas presented here, we need to emphasize some important key points that will motivate you to create amazing products but all seasoned with a little bit of reality.

Noah's Ark Syndrome

The notion of "**If you build it, they will come,**" started and ended with Noah. If you just publish your app and seat at your desk waiting for 100,000 downloads it will never happen. You need to also work hard at the market side. Look for partnerships with specialized magazines, give a copy of your product to a professor you know so he can use it during his classes, consider posting about it into the social media network. Invest some time creating pages into Facebook, send tweets to your friends and community, recover your old business contacts and let them know what you've just created.

The App Store is tiny when we compare to Apple Store or Google Play. The App Store just started and will have more and more users as fast as you can note. So act now, be quick, set your strategy and start today! Once you build your reputation and are known as a top App Store Publisher you can use these credentials to launch more products that will be quickly noted and consumed by the users.



What's next?

The very first product you create will be really tough and will require a lot of work to make things happen. Once you publish the first product and keep it up and running the next will be way easier to do. I would encourage you to visit and browser carefully through the App Store and read about other products, check their features, download them and play with them.

A good research about what currently exists and also about the quality, level and price of everything being sold into the App Store, will give you the necessary push to launch your product. Don't be afraid about the negative reviews you may receive. As humans and perfectly imperfect creatures we need to constantly evolve and turn all bad reviews into new great ideas for the future improvements.

This Lecture was directed to talented people wanting to sell (or even distribute for free) amazing products created with their long hard working hours during years. The ability of translate the talent and brilliant ideas into a product is not easy to achieve but if you plan yourself, look around, stay humble and concentrate into your goals, the success is at the corner!

