



Autodesk Revit API – Plug-ins

Martin Taurer

LivingRoomCraftZ

livingroomcraftz.com



Goals of this presentation:

To give an overview of how to programmatically interact with a Revit project based on practical needs and requirements.

Topics:

- Definition of terms
- History of the API
- Urban Myths
- Methods to utilize programmatic interaction
- Structure of a Revit project
- Practical aspects and use-cases
- Do's and don'ts
- Introduction to the development process

Autodesk Revit API plug-ins



Definition of terms:

An application programming interface (API) is a source code interface that an operating system, library or service provides to support requests made by computer programs.

(Source: http://en.wikipedia.org/wiki/Application_programming_interface)

Application Programming Interface (API) is an older technology that facilitates exchanging messages or data between two or more different software applications. API is the virtual interface between two interworking software functions, such as a word processor and a spreadsheet. This technology has been expanded from simple subroutine calls to include features that provide for interoperability and system modifiability in support of the requirement for data sharing between multiple applications...

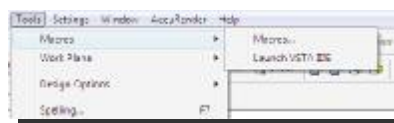
(Source: <http://www.sei.cmu.edu/str/descriptions/api.html>)

Autodesk Revit API plug-ins



History of the Revit API:

- 2004: Revit branches into verticals (Revit Structure). A first limited API for selected Structural Analysis packages/developers is opened up. (Revit 7)
- 2005: The first public API for Revit launches (Revit 8), API centers around access to structural elements in the project.
- 2006: API support is extended in the Revit Architecture, more elements become read/write-able through the API (Revit 9)
- 2007: API introduces a separation into platform API and verticals API (Revit 2008)
- 2008: Customer level Macros are introduced. Revit now ships with Microsoft VSTA IDE bundled (Revit 2009)



Autodesk Revit API plug-ins



Which API – Methods of programmatic interaction with Autodesk Revit:

Revit API:

Offers programmatic interaction with Revit projects on utilizing VB-NET and/or C# as a programming language
Requires significant knowledge of programming techniques in either IDE
Requires understanding of object-oriented programming to access exposed classes and methods with Revit
Centers around access to project data
Provides limited support for user-level process automation

Revit Macros:

Practically offer the same possibilities/limitations as the Revit API
Has a built in IDE
Gives easier access to integrating the macro on a project/session level
Supports Application Level Macros
Supports Project Level Macros



Autodesk Revit API plug-ins



Which API – Methods of programmatic interaction with Autodesk Revit:

...what else is there?

Revit Journals:

- Revit journals are automatically created with each Revit session
- Revit journals are extended VB Script
- Journals can be manipulated and offer a quick access to automate repetitive tasks in Revit
- Journals require less programming experience

Microsoft Windows API:

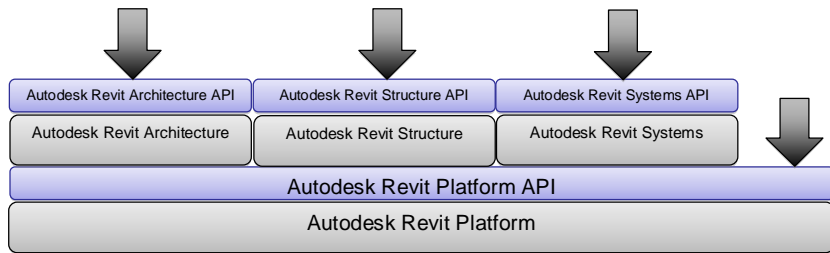
- Microsoft Windows API offers an interface to programmatically interact with virtually every application on a User Interface level
- Programming the Microsoft Windows API per-se requires deep technical understanding of the operating system platform
- 3rd party toolkits are available to facilitate interaction though the Microsoft Windows API on a user level, using either scripting tools or VB NET and C#.

Autodesk Revit API plug-ins



The Autodesk Revit API Model:

The Revit API is separated into a platform API and vertical APIs building on top of the platform API
 An external application can access either a platform class/method or a class/method belonging to a specific vertical.

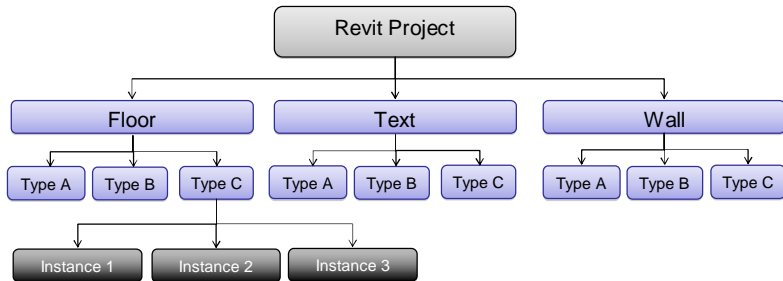


Autodesk Revit API plug-ins



The Autodesk Revit Object Model:

A Revit project is hierarchically structured along the following object model:
 Project – Element – Symbol – Instance



Autodesk Revit API plug-ins



What Can I Do With the Autodesk Autodesk Revit API?

The following are general areas where the API is suitable:

- Creating add-ins to automate repetitive tasks in the Autodesk Revit user interface
- Enforcing project design standards by checking for errors automatically
- Extracting project data for analysis and to generate reports
- Importing external data to create new elements or parameter values
- Integrating other applications, including analysis applications, into Autodesk Revit products
- Creating Autodesk Revit project documentation automatically

(Source: Autodesk Revit SDK - Getting Started Autodesk Revit API 2009.doc)

Autodesk Revit API plug-ins



Getting Started with the Autodesk Autodesk Revit API?

You will require:

- A working understanding of Autodesk Revit Architecture, Autodesk Revit Structure or Autodesk Revit MEP 2009
- An installation of an Autodesk Revit-based product, including the Software Development Kit
- MS Visual Studio 2005 or MS Visual Studio 2005 Express Edition
- Some experience in a .NET based development language (Autodesk Revit API examples are provided in C# and Visual Basic.NET.)
- 2 days of free time

(Source: Autodesk Revit SDK - Getting Started Autodesk Revit API 2009.doc)

Autodesk Revit API plug-ins



Autodesk Revit API in-depth:

Deployment:

- The Autodesk Revit API supports in-process DLLs only. This means that your API application will be compiled as a DLL loaded into the Autodesk Revit process.
- The Autodesk Revit API supports single threaded access only. This means that your API application must perform all Autodesk Revit API calls in the main thread (which is called by the Autodesk Revit process at various API entry points), and your API application cannot maintain operations in other threads and expect them to be able to make calls to Autodesk Revit at any time.
- **External commands**
- The Autodesk Revit API enables you to add new commands to the user interface of Autodesk Revit. These commands will appear in the 'Tools' menu, under a submenu labeled 'External Tools'. Through the API, external tool commands have access to the Autodesk Revit database, as well as the currently selected elements.
- **External applications**
- The Autodesk Revit API enables you to also add external applications. These applications are invoked during Autodesk Revit startup and shutdown. They can create new top level menus, toolbars, and buttons, as well as react to events occurring in the Autodesk Revit user interface.

(Source: Autodesk Revit SDK - Getting Started Autodesk Revit API 2009.doc)

Autodesk Revit API plug-ins



The Autodesk Revit API Object Model:

The Autodesk Autodesk Revit API is designed to reflect the same user interaction paradigms as the program's Graphical User Interface. Therefore, the first step to understanding the API is to learn how to use the program.

(Source: Autodesk Revit SDK - Getting Started Autodesk Revit API 2009.doc)

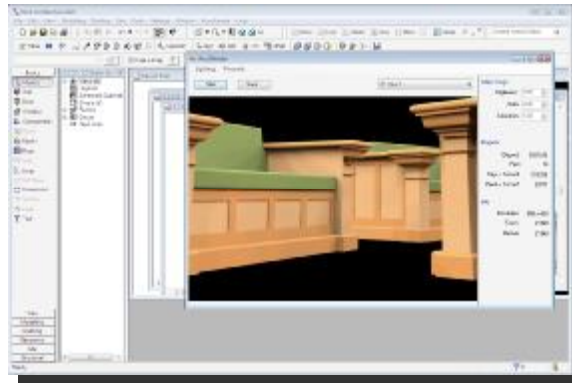


Autodesk Revit API plug-ins



Autodesk Revit API - what does it look like:

Below is an example for an external application interacting with Revit Architecture 2009



Autodesk Revit API plug-ins



Autodesk Revit Macros:

“What are macros, and why would you use them”?

- Macros are programs that are designed to help you save time, by automating repetitive tasks. Each macro performs a series of predefined steps to accomplish a particular task.
- The steps should be repeatable and the actions predictable. For example, you might define a macro to add a grid to your project, to rotate a selected object, or to collect information about the square footage of all the rooms in your structure.

Other general examples include:

- • Locating and extracting Revit content to external files
- • Automatically tweaking geometry or parameters
- • Automatic creation of many types of elements
- • Importing/exporting external file formats.

(Source: Autodesk Revit SDK - Autodesk Revit VSTA User Manual.pdf)

Autodesk Revit API plug-ins



Creating Autodesk Revit Macros:

To create macros, you must install Revit VSTA on top of your existing Revit installation. The Autodesk Revit product and Autodesk Revit VSTA versions must be the same.

You can use the Autodesk Revit VSTA macro features in all Autodesk Revit products: Autodesk Revit Architecture, Autodesk Revit Structure, and Autodesk Revit MEP., however, a separate Revit VSTA install, described below is required for each Revit product that you want to be able to run macros.

Once installed, Autodesk Revit VSTA provides:

- New items on the toolbar's Tools menu:
 - Tools > Macros > Macros...
 - Tools > Macros > Launch VSTA IDE
- Macro Manager, a user interface launched by the Tools > Macros > Macros... menu option. Macro Manager presents a list of macros you built previously that you can run, edit, or debug (StepInto). Macro Manager also provides options to create new macros using different types of templates.
- An Integrated Development Environment (IDE) built into the product, the Autodesk Revit VSTA IDE. You can launch it several ways:
 - Tools > Macros > Launch VSTA IDE
 - From the Macro Manager, by selecting the New, Edit, or StepInto buttons
- Full access to the Autodesk Revit API.

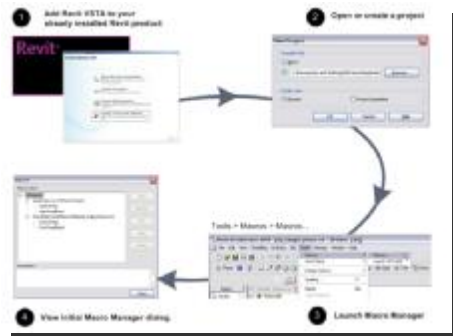
(Source: Autodesk Revit SDK - Autodesk Revit VSTA User Manual.pdf)

Autodesk Revit API plug-ins



Autodesk Revit Macros:

First Steps to Revit Macros



(Source: Autodesk Revit SDK - Autodesk Revit VSTA User Manual.pdf)

Autodesk Revit API plug-ins



Autodesk Revit Macros:

Revit API differences:

Feature or Capability	ExternalCommand	Macro
Declaration	Must implement IExternalCommand interface and its Execute method.	Declare a public method with no parameters and void return type in the ThisApplication or ThisDocument class.
Application object	Access the Application object through externalCommandData.Application.	As noted earlier, the this keyword in C# points to the Application object for Application-level macros, and points to the Document object for Document-level macros. Use the Me keyword in VB.NET for Document-level macros. this.Application points to the Application object (Me.Application in VB.NET).
new operator	Some Revit API objects can be created directly by new operator, such as Geometry.XYZ. For example: XYZ a = new XYZ(1.0, 0.0, 0.0)	The Revit API objects that can be created by new operator in external commands cannot be created by new operator in macros. Instead, they must be created by Application.Create.NewXXX methods.

(Source: Autodesk Revit SDK - Autodesk Revit VSTA User Manual.pdf)

Autodesk Revit API plug-ins



Autodesk Revit Macros:

Revit API differences:

Feature or Capability	ExternalCommand	Macro
		For example, in a C# Document-level macro: <pre>XYZ a = this.Application.Create.NewXYZ(1.0, 0.0, 0.0);</pre> In a VB.NET Document-level macro: <pre>Dim a As Autodesk.Revit.Geometry.XYZ = Me.Application.Create.NewXYZ(1.0, 0.0, 0.0)</pre> See the MyFirstMacro* examples in this topic.
Menus and toolbars	API external applications can create menus and toolbars for each external command through an external application.	Not supported.

(Source: Autodesk Revit SDK - Autodesk Revit VSTA User Manual.pdf)

Autodesk Revit API plug-ins



Practical aspects and use cases for API/Macro development:

Revit API based applications:

Commercial software development, large company-wide applications

Well suited to tie the Building Information Model into a process chain with other applications accessing the Building Information Models data

Suitable to access and reuse-/re-factor information stored in the Building Information Model for downstream use

Requires software development approach (Design Implementation, Testing and Deployment)

Revit Macros:

Easier to deploy and maintain

Better suited for small and medium sized organizations

Suitable for intra office standardization and process integration

Autodesk Revit API plug-ins



Technical aspects around Autodesk Revit API/Macro development:

Revit API based applications:

Consider the target audience and possible distribution size

Even a trivially small application requires software development procedures and methodologies once it will be distributed to a large pool of users

Professional software development at least encompasses the following steps:

- General Design and use cases
- Implementation specific design
- Implementation
- Testing/QA
- Deployment

Revit Macros:

As above, but due to potentially smaller target audience, the requirements regarding software development procedure are lower.

Deployment is handled by The Revit VSTA IDE/Revit itself

Better suited for small and medium sized organizations

Autodesk Revit API plug-ins



Technical aspects around Autodesk Revit API/Macro development:

Revit API based applications/Revit Macros:

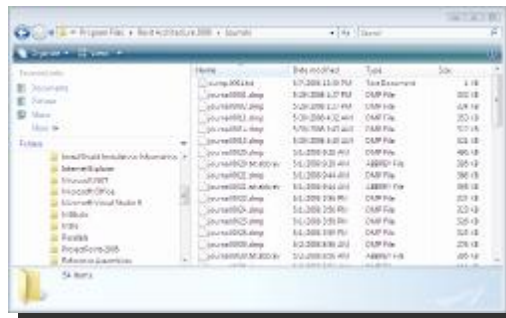
- Remember, that API based application/Macro will strictly run in-process.
- This makes error handling as well as testing mandatory
- Consider the impact of your application to projects ranging in size
- Think conservatively and rely on methodologies proven to work. What will not work reliably when done interactively in Revit will most likely get amplified in it's negative consequences when done programmatically

Autodesk Revit API plug-ins



Scripts and Journals:

- Revit creates so-called journal files every time a Revit session is started. These journal files enable replay of the Revit session
- The location of the journal files is in the journals subfolder of the program installation directory

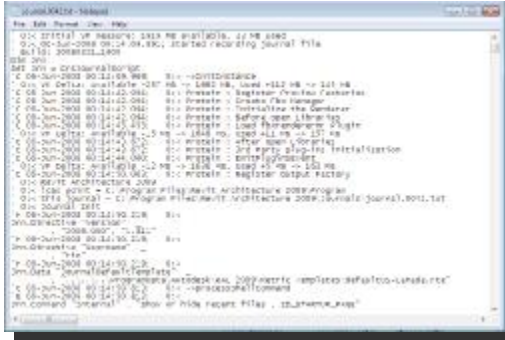


Autodesk Revit API plug-ins



Scripts and Journals:

Journal files are VB-script based scripting files utilizing Revit extensions for scripting. Originally developed as a QA and debugging tool, they can be used to quickly script repetitive tasks in Revit



Autodesk Revit API plug-ins



When to use journals and scripting:

- Given the comparatively low requirements in understanding programming techniques, journals are mainly recommended for power users who wish to automate repetitive tasks in Revit in a quick and easy way.
- Whenever you find yourself in a situation where you do the same sequence of user interaction multiple times over and over again, journal scripting may be a low-key alternative to API/Macro programming
- Journals (VB Script) support rudimentary operations and user interactions such as loops, variables and message boxes

Main use cases:

- Content management and update, project maintenance tasks

Drawbacks:

- There is no documentation of the scripting extensions that Revit offers
- Revit is sensitive to syntax issues as well as callback values in journals which makes complex operations difficult.
- The clipboard cannot be accessed though scripting

Autodesk Revit API plug-ins



Scripting – Microsoft Windows shell scripts (WSH scripting):

This is the lowest possible way of programmatically accessing Revit (and any other Microsoft Windows based application)

WSH scripting allows direct UI access to Revit – a shell script can basically mimic a subset of possible user interactions with the software

WSH scripting also allows manipulating the application window itself and tying multiple applications into a shell script based workflow (e.g. copy-and-paste from Notepad to Revit,...)

```

setlocal
cd /d %~dp0
call %~dp0\Scripts\RevitAPI.vbs

```

Autodesk Revit API plug-ins



Scripting/Journaling – Use cases and technical considerations:

Typical use cases for scripting/journaling include:

- Library maintenance, project maintenance
- Automation of simple repetitive tasks
- Automating repetitive user input
- Automating UI actions across multiple applications (WSH, VB Scripting)
- Power-user/CAD manager level access to automation without adding large software development overhead.
- Scripts can be written on the spot.

Technical considerations:

- WSH and Journals offer only very limited callback and error handling functionality
- You need to do your own process management and exception handling
- WSH based scripts work on the entire Microsoft Windows API – you need to make sure that your scripted input goes to the right window

Autodesk Revit API plug-ins



A lookout beyond:

- An advanced method to access any Microsoft Windows-based application other than through scripting and the applications API is to directly utilize the Microsoft Windows API.
- Rule of thumb is that anything that can be done interactively in an application can also be done programmatically through the Microsoft Windows API
- This may however require extreme efforts to effectively implement and therefore may have only limited relevance in a practical sense
- The Microsoft Windows API is open to most modern programming languages based on Microsoft Windows
- A number of 3rd party libraries and extensions is available that simplify the process of accessing an application through the Microsoft Windows API

Autodesk Revit API plug-ins



Appendix – The software development process:

Domain Analysis

Often the first step in attempting to design a new piece of software, whether it be an addition to an existing software, a new application, a new subsystem or a whole new system, is, what is generally referred to as "Domain Analysis". Assuming that the developers (including the analysts) are not sufficiently knowledgeable in the subject area of the new software, the first task is to investigate the so-called "domain" of the software. The more knowledgeable they are about the domain already, the less work required. Another objective of this work is to make the analysts, who will later try to elicit and gather the requirements from the area experts, speak with them in the domain's own terminology, facilitating a better understanding of what is being said by these experts. If the analyst does not use the proper terminology it is likely that they will not be taken seriously, thus this phase is an important prelude to extracting and gathering the requirements. If an analyst hasn't done the appropriate work confusion may ensue: "I know you believe you understood what you think I said, but I am not sure you realize what you heard is not what I meant." [1]

Software Elements Analysis

The most important task in creating a software product is extracting the requirements. Customers typically have an abstract idea of what they want as an end result, but not what software should do. Incomplete, ambiguous, or even contradictory requirements are recognized by skilled and experienced software engineers at this point. Frequently demonstrating live code may help reduce the risk that the requirements are incorrect.

Autodesk Revit API plug-ins



Appendix – The software development process:

Requirements Analysis

Once the general requirements are gleaned from the client, an analysis of the scope of the development should be determined and clearly stated. This is often called a scope document. Certain functionality may be out of scope of the project as a function of cost or as a result of unclear requirements at the start of development. If the development is done externally, this document can be considered a legal document so that if there are ever disputes, any ambiguity of what was promised to the client can be clarified.

Specification

Specification is the task of precisely describing the software to be written, possibly in a rigorous way. In practice, most successful specifications are written to understand and fine-tune applications that were already well-developed, although safety-critical software systems are often carefully specified prior to application development. Specifications are most important for external interfaces that must remain stable. A good way to determine whether the specifications are sufficiently precise is to have a third party review the documents making sure that the requirements are logically sound.

Autodesk Revit API plug-ins



Software architecture

The architecture of a software system refers to an abstract representation of that system. Architecture is concerned with making sure the software system will meet the requirements of the product, as well as ensuring that future requirements can be addressed. The architecture step also addresses interfaces between the software system and other software products, as well as the underlying hardware or the host operating system.

Implementation

This is the part of the process where software engineers actually program the code for the project.

Testing

Testing software is an integral and important part of the software development process. This part of the process ensures that bugs are recognized as early as possible

Deployment

After the code is appropriately tested, it is approved for release and sold or otherwise distributed into a production environment.

Autodesk Revit API plug-ins



Documentation

Documenting the internal design of software for the purpose of future maintenance and enhancement is done throughout development. This may also include the authoring of an API, be it external or internal.

Software Training and Support

A large percentage of software projects fail because the developers fail to realize that it doesn't matter how much time and planning a development team puts into creating software if nobody in an organization ends up using it. People are often resistant to change and avoid venturing into an unfamiliar area, so as a part of the deployment phase, it is very important to have training classes for new clients of your software.

Maintenance

Maintaining and enhancing software to cope with newly discovered problems or new requirements can take far more time than the initial development of the software. It may be necessary to add code that does not fit the original design to correct an unforeseen problem or it may be that a customer is requesting more functionality and code can be added to accommodate their requests. It is during this phase that customer calls come in and you see whether your testing was extensive enough to uncover the problems before customers do.

Autodesk Revit API plug-ins



Links and Sources:

Recommended resources are:

Autodesk Resources:

<http://www.autodesk.com/revitbuilding/>
<http://www.autodesk.com/revitstructure/>
<http://www.autodesk.com/revitsystems/>
<http://www.autodesk.com/bim/>
<http://discussion.autodesk.com/>

External Resources:

<http://www.revitcity.com/>
<http://www.augi.com/revit/>
<http://www.revitinside.com/>

Autodesk Revit API plug-ins



Thank you
Questions ?

Martin Taurer

livingroomcraftz.com