

Streamlining workflow in architectural visualization with Blender and SketchUp



Lintunen, Jani

Laurea University of Applied Sciences
Laurea Leppävaara

Streamlining workflow in architectural visualization with Blender and SketchUp

Jani Lintunen
Information Technology Programme
Thesis
September, 2010

Jani Lintunen

Tehokkaampi työnkulku arkkitehtuurialisessa visualisoinnissa Blender ja SketchUp -ohjelmilla

Vuosi 2010

Sivumäärä 52

Opinnäytetyön tarkoituksena on tutkia avoimeen lähdekoodiin perustuvien ohjelmien käyttöä yrityksissä ja selvittää, kuinka niitä hyödynnetään nykyisessä työskentelyssä. Tarkoituksena on myös oppia ymmärtämään, mitkä asiat vaikuttavat itse päätökseen tukea ilmaisia 3D-ohjelmia, ja vastoin tavallista käytäntöä kertoa siitä avoimesti esimerkiksi yrityksen omille asiakkaille. Avointa lähdekoodia hyödynnetään laajalti kaikilla toimialoilla, mutta opinnäytetyö pyrkii löytämään vastauksia siihen, miksi näin ei ole käynyt esimerkiksi Blenderin osalta.

Aihetta pyritään tutkimaan pääasiassa ammattilaisten antaman tiedon kautta, mikä koostuu keskusteluista, työnkulun dokumentoinnista ja yhteistyöstä varsinaisessa projektissa. Kyseinen projekti toteutettiin yrityksen tarpeiden mukaan ja tehtiin yhteistyössä kokeneen arkkitehdin kanssa, hyödyntäen ennalta kerättyjä tietoja. Opinnäytteen painoarvo kohdistuu työnkulun tutkimiseen, minkä kautta tuodaan esille Blenderin ja SketchUpin tarjoamat edut yrityksille.

Tutkielman tulokset onnistuvat selkeästi osoittamaan, kuinka hyvin avoimeen lähdekoodiin perustuvat ohjelmat soveltuvat yritysympäristöön ja miten niitä voidaan hyödyntää työskentelyssä ilman, että haluttu lopputulos menettää laadullisia ominaisuuksia.

Jani Lintunen

Streamlining workflow in architectural visualization with Blender and SketchUp

Year 2010

Pages 52

The main purpose of this thesis is to research how companies use software based on open source and how they are utilized in current workflows. The target is also to learn the key issues influencing decision to support free 3D programs and against the usual practice openly communicate that decision for example to their own customers. Open source is widely utilized on all business areas, but the thesis aims to find answers to the question why this has not happened with Blender.

The topic is approached by using information gathered from professional consisting of discussions, workflow documentation and collaboration in an actual project. The project was completed according to the company's needs, utilizing previously collected material, and in close collaboration with an experienced architect, Oliver Walter. The emphasis was on researching real workflow and lead to the discovery of the underlying potential of both Blender and SketchUp in production use.

The thesis research clearly demonstrates how programs based on open source code are suitable for commercial environment and how they can be used without the final product losing in quality standards.

Key words: Architecture, Blender, SketchUp, Renderfarm.fi, open source, workflow

Contents

1	Introduction.....	7
2	Background.....	8
	2.1 Initial situation	8
	2.2 Objective	8
	2.3 Research method	9
	2.4 Resources and collaboration groups.....	9
	2.5 Connections to other projects	10
3	Focal concepts	11
	3.1 Blender	11
	3.2 SketchUp	14
	3.3 realXtend	15
	3.4 GNU as license	17
	3.5 Open Rendering Environment (ORE)	17
4	Different workflows in professional fields of architecture	19
	4.1 Architectural Office Oliver Walter Ltd (basic workflow)	19
	4.1.1 Concept	19
	4.1.2 Preliminary Design	19
	4.1.3 Accurate Studies.....	20
	4.1.4 Presentation to Client	21
	4.1.5 Building permits and detailed drawings	22
	4.1.6 Sales	22
	4.2 Toolsets in the lighting design process (additional workflow)	24
	4.3 Evocativi Ltd (basic workflow)	27
	4.3.1 Drawings from the client	28
	4.3.2 Preliminary SketchUp modeling	29
	4.3.3 Production	33
	4.3.4 Project finalizing and delivery	34
	4.4 realXtend (additional workflow)	35
	4.4.1 Designing a game model.....	35
5	Adaptation to a new workflow	37
	5.1 Delivering the SketchUp model	37
	5.2 Adding furniture and details into the SketchUp model	38
	5.3 Importing objects into Blender	39
	5.4 Creating the scene	41
	5.5 Lighting setups.....	41
	5.6 Camera animations	42
	5.7 Final adjustments	43

5.8	Rendering.....	44
6	Benefits of using open source 3D tools in companies	46
7	Conclusions	47
	References	48
	Attachment 1 File formats	50
	Attachment 2 Autodesk Revit to SkechUp	52
	Attachment 3 SketchUp to Blender	53

1 Introduction

The purpose of this thesis is to introduce Blender - the world's most popular open source 3D suite - to companies and organizations, research ways of implementing it to their existing workflow and explain why this particular application represents a serious alternative to its expensive proprietary competitors. By achieving a higher number of commercial Blender users, the software is expected to get a bigger foothold on the market and is more likely to spread into the general workflow of professionals. At the moment, freelancers and small companies most commonly use Blender, where as medium to large sized companies still remain largely unaware of the software's capabilities.

The subject is currently relevant because Blender Foundation has recently released a new version of Blender (2.53 beta), which includes a complete rewrite of the code core structure and a new graphical user interface. In addition, the ORE (Open Rendering Environment) project (see chapter 3.5), which recently got finished at Laurea University of Applied Sciences, produced a new distributed volunteer rendering service called Renderfarm.fi - a service which is very interested in finding companies to make use of its computing power. This is the time and chance to highlight Blender and get companies, professional artists and other 3D software users involved.

The thesis will concentrate on Blender in use for architectural visualization. The idea is to provide usable information for companies that are unfamiliar with Blender, to give them a perspective how existing professionals use the software and to show the achievable results to get them interested to try it out.

Programs used in commercial work are generally the same in companies, since they often have very standardized workflows and rarely diverge from them. Regardless of architectural-, virtual-or game environments having different types of workflows, the market leading software have kept their place. Although Blender has been doing well over the years, it is hard to compete with large software companies with strong trademark brands, which often matters a great deal in the business world. This is the basic reason why most organizations have not adopted Blender. The workers might not be against it, but for the companies it is above all matter of business image.

2 Background

2.1 Initial situation

The present situation with Blender looks promising, since companies like Oulu, Finland based Evocativi have started to adopt Blender as a replacement for their former 3D tool, 3Ds Max. According to Helsinki, Finland based lighting architect Oliver Walter, the question often asked by clients is: "Can Blender offer as good results as commercial programs?" Evocativi has tackled this question. Their response was that 3D modeling and rendering results can be exactly the same in both Blender and 3Ds Max. This indicates that the price of the software is an unessential factor when comparing two software products.

Among Blender professionals the software is often compared to commercial programs costing thousands of dollars. Blender does have several good features, but it lacks recognition as a professional tool. It is fairly rare to see organizations using Blender as their main tool for 3D work or hear them speak openly about it, regardless of Blender having a solid reputation for being a serious open source alternative.

Schools and universities around the world should use programs that meet industry standards and requirements, but in fact most organizations like to hold on to their old traditions. It can be argued that Blender is an excellent base for learning, providing a real alternative to its expensive competitors, such as Autodesk's 3Ds Max and Maya that cost thousands of euros each. It would seem that schools and Universities are hesitant to admit the fact that open source could meet their needs, something that can be partly credited to open source software having an image of being less accessible than proprietary alternatives (Tuomisto, 2010). When discussing open software, the decision makers seem to think that the positive hype around Blender is based on the price tag and not in the actual content. This is an unfortunate misconception that can be proven wrong by studying the works of companies and creative professionals like at Evocativi Ltd. and Architectural Office Oliver Walter (see projects presented in chapter 4.1 and 4.2).

2.2 Objective

The thesis subject was commissioned by Laurea UAS and the purpose was to research the workflow of two different architecturally oriented companies, both using Blender as their main 3D tool and draw conclusions on how well Blender integrates into commercial use in actual business environment. The results are then presented in a simple level, so anyone interested should be able to understand the concept and draw own conclusions. The final mate-

rials are meant to be usable by any company or professional that is either thinking of or is already taking Blender into use.

The fundamental research questions that the thesis tries to answer:

- What does Blender offer to companies/organizations?
- Can Blender replace already existing 3D software in companies/organizations?
- What are the main reasons for not adopting Blender into companies/organizations?
- How can companies/organizations actually benefit from using Blender?
- How well does Blender fulfill requirements from an architectural aspect?
- What other possibilities does Blender offer to companies/organizations?

2.3 Research method

The subject of this thesis supports using the action research method, which tries to affect target group working habits, but most importantly the attitude to do so. With this method it is possible to add new angles, improve communication between employees and researchers, but above all offer new solutions to existing problems. Because the thesis aims to change the current workflow of professionals, it is clear that action research supports it very well. (Ojasalo, Moilanen & Ritalahti 2009.)

2.4 Resources and collaboration groups

Oliver Walter is a lighting architect who has several years of experience of Blender, Autodesk 3Ds Max and different CAD softwares. Mr. Walter's background is impressive, with work performed in large projects with very complex requirements. He therefore carries a spectacular work portfolio. For these reasons he is an optimal candidate to introduce his previous work and different working methods for creating gorgeous architecture and lighting visualizations in Blender. The thesis has been done in close collaboration with Mr. Walter as well as the chapter 5 case work.

The second partner to share their workflow is Evocativi Ltd. The company portfolio is varied from architectural visualizations to game environments, but the research done in this thesis is focused mainly on the architecture. The whole workflow process will be then demonstrated in an actual work case, done by the thesis author and Oliver Walter.

The Finnish Blender community is also a very important part of the thesis work, but it will be used as a supporting resource. Communicating with this group is mainly done through IRC (Internet Relay Chat) channel #blender.fi in freenode server, and face-to-face meetings. The

reason for mentioning this is that the community has more information than all Blender books together, which means that they are able to answer basically any questions in hand.

2.5 Connections to other projects

The essential connection to other thesis' is the ORE project, which will be explained in chapter 3.1 in more detail. It has allowed researchers, trainees and the Finnish Blender community to work together, share knowledge about common topics of interest and provide many research topics for thesis writers from different universities. In return, these theses represent the body of the final report that will be made available to the public during 2010. The following thesis subjects that are connected to ORE:

Thesis topic: Integrating open source distributed rendering solutions in public and closed networking environments

Authors: Niko Suomalainen and Heikki Seppälä

Short description: The purpose of the thesis is to model a render farm to give resources for rendering 3D-animations in a closed environment, and analyze the possibilities of its commercial use. It also ponders the technical side of distributed computing and how it could be used in a profitable way.

Thesis topic: A more intriguing volunteer computing experience through Drupal and social technology

Authors: Lauri Viitala

Short description: The purpose of the thesis is studying the possible benefits of using a web content management system alongside the BOINC provided website template. The empirical section of the thesis examines the current implementation of Renderfarm.fi, and focuses on how the Drupal component of the web site was built, what features it has and how it was integrated with BOINC and BURP. Furthermore, it examines the possibilities of further implementation of groundswell technologies for creating a community site.

Thesis topic: 3D short animation utilizing distributed computing

Authors: Lassi Haaranen

Short description: This thesis focuses on creation of a short animation with Blender for promotion purposes for the ORE project. It approaches the subject from three different points of view. It firstly focuses on the creation of the animation from the traditional modeling and animation point of view. After that, attention is given to how the project was rendered using this new service and how the service in itself works. Finally it discusses the social side of this rendering rendering platform. Also included in this part is a small questionnaire about interest in this kind of communal creativity.

Thesis topic: Social media as business supporter and part of ORE-project

Authors: Juha Laakso

Short description: The purpose of this thesis was to research what companies know about social media and open licenses and also how companies have used them in tasks. This thesis introduces examples of social media services and procedures how to take advantage of them in marketing and business in general.

3 Focal concepts

3.1 Blender

The history of Blender began in 1988 when Ton Roosendaal founded a company called Neo-Geo. It was the leading animation studio at that time and the largest one in Europe. In 1995, Roosendaal became the responsible person for Blender development - after the company came to the conclusion that other 3D softwares were too old and difficult to update. In 2002, Blender was released under GNU General Public License after creditors agreed to invest €100,000 to finally free the software as open source. After that the Blender Foundation became the responsible developer and Ton Roosendaal its chairman. (History of Blender. Blender.org 2010.)

The Blender we know these days was very different before, since it has not always been free and open source. According to Evocativi's experience, Blender is often mistaken to be a less efficient tool and not able to produce high quality results. The reality of Blender is that the software has a lot smaller amount of users and especially professionals, which then reflects Blender's capabilities and might give false impressions. Figure 1. indicates the aberrancy of different 3D programs, which can also be used to demonstrate the previous fact. When software has a large amount of users, like 3Ds Max, it clearly means that there are also more professionals working with it. The work these people produce is then evaluated by everyone and most of the credits are given to the software, as if it would only be because it. To have a better understanding of the Blender capabilities, visit official gallery at:

<http://www.blender.org/features-gallery/gallery/art-gallery/>

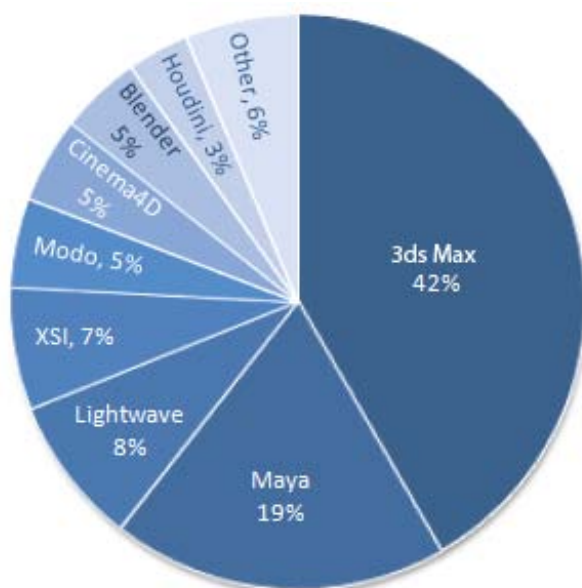


Figure 1: Community survey made by CGenie. (CGenie survey 2009.)

Blender was originally designed to be internal tool for commercial production. The requirements for such software are still very high. The tools must allow users and artist to work within tight schedules, fast, and above all: be as productive as possible. History of Blender actually indicates that the program was originally designed as professional software, since it was operating as comprehensive in-house tool for years. Its main goal is still to serve expert needs and provide them with a high quality program to work with, able to produce same- or better results than its competitors.

Current status of Blender is that the code is being rewritten from ground up. This is due to the limitations of the old version, and the current needs of development to meet the required standards of the users. The versions below 2.5 represent the old Blender, which was originally created in the 90's. The new versions from 2.50 and above are the new era for Blender, in the development. The finished product will be called Blender 2.6, containing all the functions that the old one had, and much more. Figure 2. below will describe the phases more clearly and in detail.

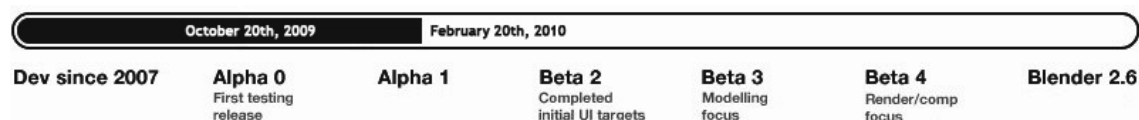


Figure 2: Blender 2.5 alpha development roadmap. (Blender Foundation 2010.)

The tools and functions included in Blender cover all necessary needs of personal and professional user. In common use all 3D packages have good coverage, but for very specific use they tend to lack features. Some are missing fluid or smoke simulation, where others have them, but lack something else. This is where the third party developers step in and sell plug-ins worth thousands of dollars, but the case is different with Blender. Like mentioned before, Blender is based on open source code and anyone can edit it or create scripts easily by using Python language. The core development makes fast leaps forward, thanks to volunteers and Blender Foundation enthusiasm to improve the software. Needed functions are often added on demand and users can influence the actual development heavily. This way the package is usable by default and the need for plug-ins is minimized. Ton Roosendaal manages to describe Blender development in CGenie interview in 2009 very well: "No matter how much influence I might have, I can't make the software move in any particular direction without a very solid and well covered consensus by its active volunteers. A good example is the period shortly after Blender became open source, in 2003. Instead of picking up all the beautiful visions I had for exploring real-time interactive 3D creation, the developers just worked on fixing and improving the modeling tools first." (CGenie 2009 survey, Blender developer interview. 2009.)

Survey made by TDT 3D compared competing software by their features and functions, as well as the cost of necessary add-ons for film production, freelance- and student work. The results displayed in figure 3. show the strong- and weak points of Blender. Even though the survey was made in 2007, the results can still give good overview. One thing that stands out clear is that all of the programs get similar results and none of them have a clear advantage over the others. The same thing can be seen in any survey available.

Animation tools	Good	Nodes-Based Compositing	Yes
UV tools	Excellent	Compositing	Yes
Painting	Low	Nodes-Based Workflow	None
Modeling	Good	Nodes-Based Materials	Yes
Modifiers	Good	Fluids	Very good
NURBS	Low	CG shader / Games	Yes
Dynamics / Rigid bodies	Very good	Top unique feature	3DRT Sculpting Video edit
Soft bodies	Good	Scripting	Very good
Hairs	Good	Scripting Dev.	Python
Cloths	Good	C/C++ Dev.	Very good
Particles	Good	The bad side	Interface/documentations

Figure 3: Blender feature review. (TDT 3D 2008.)

The steepness of the learning curve can be argued, but users that have previous experience on other similar softwares can learn Blender within a month. Even though the GUI (graphical user interface) is known to be outdated and disliked in default settings, the new version of Blender has that problem solved. The documentation is nowadays so well done it could even be praised, and the commonly known downsides also apparent in the survey results have been entirely fixed since.

Blender can be downloaded from here:
<http://www.blender.org/download/get-blender/>

3.2 SketchUp

@Last Software is the company that originally developed SketchUp in year 2000. The goal was to create a comprehensive drawing tool that enables easy creation of 3D models and prototypes for various industries, but mostly concentrating on architecture. The program was designed to be lighter than ordinary CAD software and to offer freedom similar to using pen and paper. Having a system that is fun and at the same time easy to learn was very important for the development, where most traditional software have failed to reach this goal. Sometime later Google got interested in the program and bought the software rights in 2006. The software has a free- and the commercial (Pro) version, but is still keeping the cost low as €400 per license. (SketchUp original developer 2009.)

SketchUp is commonly used for pre-production work, meaning that the software is capable of producing complicated models within short period of time. This indicates that disregarding the field of business, SketchUp has a lot to offer for streamlining purposes. Good example is a company called *Massive Black*: a full production artwork and asset outsourcing studio that has its own distinct needs for working, and SketchUp suites it well. Even though most of the artists there are strictly drawing-or painting oriented, they still use SketchUp to create baseline for the art. Architects, like those working with Evocativi, use it for completely different purpose and is addressed in detail in chapter 4.3.

It is commonly known how easy SketchUp is to learn, and even non-professionals can master the software within hours. According to Evocativi, this is the greatest advantage of using software like SketchUp. It can even be given to people with no previous 3D experience and offer new ways of working that have not been possible before.

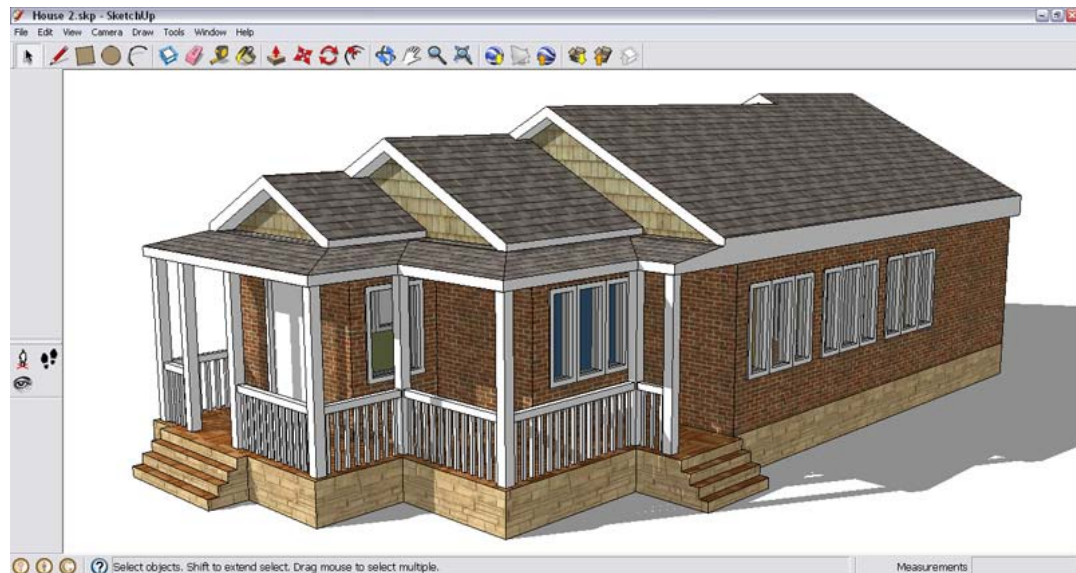


Figure 4: SketchUp screenshot of Google Warehouse -model

Free version of SketchUp can be downloaded from here:

<http://sketchup.google.com/>

3.3 realXtend

realXtend is an open source software being developed by companies like LudoCraft and Admino technologies in Oulu, Finland. The platform offers an opportunity to create virtual 3D environments for "fair and transparent cooperation" between companies, organizations or any interested parties for free (realXtend homepage, 2010). The concept was created with new business models in mind. Even though realXtend is not meant to be the source of the income, the idea is to offer a platform capable of processing third-party applications, and therefore generate new ways of streamlining work or simply adding value to existing services and products.

The platform is a combination of various communication technologies that have been merged into a single software. This way the environment can be used as productive platform for everyday work, presentations, as a teaching environment or simply for entertainment purposes. The idea behind realXtend is to take all modern-day 3D technologies and build a virtual world around them. The ready product will be presented to business owners as a tool that enables new business opportunities from them.

The future plan for realXtend is to develop an client version, capable of running in smaller devices like smartphones and tablets.



Figure 5: realXtend game engine footage 1. (realXtend homepage 2010.)

realXtend also enables users to create custom models to represent themselves, so they can for example visit a housing fair online and see an actual building in virtual representation. Giving users the ability to walk around and see example products, the company selling them can add value to the services offered.



Figure 6: realXtend game engine footage 2. (realXtend homepage 2010.)

3.4 GNU as license

There can be many reasons for companies to use open source licenses. The decision is usually made with business purposes in mind and naturally cost saving is one of the key issues. There for lowering the development costs, exploiting the possibility of new innovations and being independent from costly licenses are often the top three goals of the companies using open source. (Seppä, A. 2006.)

What makes Blender different from other 3D softwares is that it operates fully under GNU General Public License (GPL) and still offer all the industry tools required, for example functions needed in professional animation, modeling, editing and so forth. The program is also freely downloadable from the internet and it enables users to edit the source code legally, as well as giving them opportunity to participate in the actual development process. Blender can also be modified and even be sold forward as product. All this is possible because of the licensing method used.

The GPL license offers Blender developers an excellent base for more flexible and very effective work. It evolves exactly to the direction what users desire, which means that the overall advantage of open source is not cost saving, but rather the license openness and freedom to evolve. (About GPL license 2009.)

3.5 Open Rendering Environment (ORE)

ORE project operates at Laurea University of Applied Sciences in one of the Lab environments in Leppävaara. The project was originally started in early 2008 and in summer 2009 the main produce of the project, the publicly distributed rendering service Renderfarm.fi was opened for public beta phase. Since 2010 the website has gained a large amount of new users, through social media and events like the Assembly computer festival held yearly in Helsinki.

Renderfarm.fi itself is a platform for distributed rendering, done over the Internet. The service is completely free and enables users to render their animations or (optionally) still images by using the computing power of volunteers around the world. Any graphical artists or animators can benefit from the service and use higher standards than normally possible, like bigger resolution and have shorter render times. Equally importantly, it enables its users to participate in the rendering process, regardless of whether or not they themselves are able to or even interested in learning about 3D modeling and animation. The service is basically the same thing as a normal render farm, but in this case it operates over the Internet by using BOINC (Berkeley Open Infrastructure for Network Computing) and BURP (Big and Ugly Rendering Project). All of the technologies used in the project are based on open source and the

service itself is currently offered completely free, as previously mentioned. (About Renderfarm.fi -service 2009.)



Figure 7: Screenshot of Renderfarm.fi -website

Renderfarm.fi supports currently only the Blender internal rendering engine, but there are plans to expand and possibly integrate advanced external engine to the service. The plugin needed to use the service is included with the newest version of Blender and require no additional installation. This makes the service more intuitive, which enhances the accessibility a great deal. The service advocates the use of Creative Commons licensing (read more at www.creativecommons.org), but also offers the user a chance to hold copyright to the materials rendered on the service.

Executing serious rendering in any render farm has its limits, and some of the most common ones are introduced in chapter 5.8 from the Renderfarm.fi perspective. There is no official list about limitations, but the most obvious and common problems are explained there. It is advisable for new users to familiarize themselves with Renderfarm.fi *frequently asked questions* and other materials available.

Rendered animations and images done over Renderfarm.fi are displayed at the service homepage, under Gallery. Please visit www.renderfarm.fi for more information.

4 Different workflows in professional fields of architecture

4.1 Architectural Office Oliver Walter Ltd (basic workflow)

Along with architectural projects, Oliver Walter has been working closely with Julle Oksanen Lighting Design Ltd, finishing a great number of high profile lighting projects in Finland and abroad. The work has required intensive use of 3D modeling and lighting simulations. For this Oliver Walter has previously used expensive programs, like 3Ds max and AutoCAD, which now have been replaced with either free or low cost alternatives with no overall loss of performance.

The following chapters will consist Mr. Walter's typical workflow for common architectural challenges. The different phases are explained in a form that anyone should be able to understand- and use them afterwards. Each step has the necessary information about the software used and other tools needed. These steps also include simple tables that indicate the purpose, strengths and step by step instructions. This is to ensure fast read-through so steps can be followed more easily.

4.1.1 Concept

Process:	Hand sketching
Strength:	Quick, flexible and doable anywhere

The workflow of any architectural task usually begins with hand sketching, which provides the first vital steps for creative work and gives guidelines for continuing. The whole process is comparable to any line of artistic work, since it is the usual way for recording ideas and creating new ones. The purpose of this work phase is to have variations and alternatives to work on, but above all to make them fast. Drawing holds no limits whatsoever and it is doable anywhere and anytime, which is the real strength of it.

4.1.2 Preliminary Design

Tool / Software:	Blender
Process:	Scanning hand sketches and importing them to Blender
Purpose:	Create rough mass studies that can also be presented to the client
Strength:	Quick and flexible

A preliminary study is something that is needed in order to see how the idea works in three dimensional form and within the representation of the environment. Hand sketches are scanned and imported into 3D software, which is in this case Blender. Even though Blender is not very well adapted to have accurate CAD drafting, it is possible to create rough measurements that are handy in later phases. In this case the model is kept in very rough masses, like a 3D sketch. This allows the architect to evaluate how the design appears as a whole and how the different parts interact with each other. It is easy to place virtual cameras to see how the design looks from different key view angles and create fast lighting studies. Even though the rapidly producible 3D sketch (possible in Blender) is essential mainly for the designer's research, the visual material can also be used for presentations purposes.

4.1.3 Accurate Studies

Tool / Software:	CAD / ProgeCAD
Process:	<ol style="list-style-type: none"> 1. Import from Blender into CAD software as DXF file format or in plain image 2. Accurate drafting over background images
Purpose:	Base for CAD drawings and accurate measurements
Strength:	Precise accuracy

Architectural planning requires very accurate dimensional inputs. The design is required to fulfill specific building standards (space requirements) and regulations (such as fire regulations etc.) precisely. According to Walters, this phase can only be done within proper CAD software, such as the high-end programs like Archicad, AutoCAD or the much less costly ProgeCAD for example.

Blender can export 3D models in vector format like DXF, which can later be imported to any CAD software. This method is not recommended because of the inaccurate nature of the Blender model for architectural purpose. It is better to import different viewing angles as images and take them to the chosen CAD program. By doing so, the architect can retrace the original idea to actual CAD drawings and have accurate measurement inputs. The drawings are later used to create the actual 3D model.

4.1.4 Presentation to Client

Tool / Software:	Blender +external renderer
Process:	<ol style="list-style-type: none"> 1. Import from CAD to Blender 2. Additional details and lighting 3. Quick rendering
Purpose:	Demonstrate design to the client
Strength:	Created fast, but with high level of detail

CAD programs are not suitable for high quality visualizations, since they lack features to adjust material settings, lighting and shadow types, or simply have little navigation control to move through the 3D space. Blender in other hand has those tools well integrated and it enables architects to create quick, but professional level visualizations.

Once the building measurements are fixed and the overall volumes defined, the model can be taken into Blender. Before exporting anything, there are certain basic requirements to keep in mind: When transferring 2D drawings or 3D models from CAD software, the units need to be in meter format and the content has to be aligned near the point of origin ($x = 0, y = 0, z = 0$). There are also other problematic features involved, which are common to all 3D software when importing DXF files. This is mainly due to the retrained specifications created by Autodesk and how the file format has been coded, and there for is not Blender specific problem.

There are two ways to import 3D content from CAD to Blender:

1. In the chosen CAD software all *blocks* and *solids* have to be turned (exploded) into single faces. The resulting 3D model will then be exported in AutoCAD's DXF (release 11/12 ASCII) file format and imported into Blender with the internal DXF importer. When following this approach, it is important not to accidentally explode the faces into lines.
2. Alternatively the 3D model with *solids* and *blocks* can be exported as AutoCAD's DXF (release 11/12 ASCII) and opened with a program called Accutrans, which is a 3D object conversion software. The model can then be exported in Lightwave own LWO format, which Blender can reads well without any no loss of data.

The process is continued by adding materials, lights and cameras into the scene. Advanced lighting setups are created and used with external rendering engines, like Yafaray and Lu-

xrender. Both of the add-ons offer great results by simulating the physical parameters of the light and can be downloaded for free. This way even the most simple structure can be presented with professional level quality and in a short time period. Creating convincing visualizations is extremely important, since 3D images can illustrate more than line drawings. They also help to persuade the client and assure them that the architect has the project under control.

4.1.5 Building permits and detailed drawings

Tool / Software:	CAD / ProgeCAD
Process:	Accurate measurements and confirmation of building standards
Purpose:	Building elements are done according to standards
Strength:	Accuracy

Chapter 4.1.3 is continued by combining the working drawings with building codes, heights, square meter information and detailed measurement lines that are added for each building part. The accuracy of this information is crucial, where the working drawings will most importantly serve as permission drawings and are submitted to the building authorities to obtain a building permit. The drawings will also serve as base for specialists and engineers to do their complementing design for statics, ventilation, electricity and lighting.

4.1.6 Sales

Tool / Software:	Blender +external renderer
Process:	1. Final touches and camera animation 2. Rendering with Yafaray or Luxrender for example
Purpose:	Persuasion of buyers / clients
Strength:	Very high quality results

The finalizing is done in Blender by adding last touches to the models, materials and overall setups. The same lighting setup can be used as previously, but with higher values to active best quality possible.

Libraries of models, materials and textures help architects to concentrate on the important parts of the working process. Creating custom models is often very time consuming and might delay the project for unnecessary reasons. Still, clients often want to see the houses furnished and environments filled with plants, buildings and people. These problems can be solved with models libraries, which can be downloaded or bought with low costs, and finally just applied to the scene. One of the most popular sites for selling and buying models is *Turbosquid.com*, where they offer high quality 3D models for professional and personal use.

Animations are commonly used to present the architectural aspects of models by using different camera movement and majority of clients require this to be done. The animations can also contain people walking, or driving cars to create illusion that the environment is close to real. These materials can also be attained from various sites, and possibly even for free.

Blender's own node editor allows the renders to be modified within the software. This eliminates the need of mandatory external composition software like Adobe After Effects and enables many functions not possible in any other program. With the node editor all major modifications to be done in professionally, even before the actual rendering is done.

Render farms are usually costly to own and maintain, but to meet deadlines in bigger projects it is highly advisable to utilize them. Outsourcing this task is very common, but not strictly necessary. One option is to use distributed rendering software that operates in a local area network, but optimizing a local render farm can be challenging and it requires several decent computers to run. This is one of the reasons why publicly distributed rendering was created.

The following example (figure 8.) was rendered as demonstration in Corefarm, and with a resolution not possible to be done in one computer or ordinary render farm. For animations Mr. Walter has tried Renderfarm.fi with partial success, with hopes of using the technology more in the future in order to have a more efficient workflow since much less time is spent rendering high quality animations on his own machine.



Figure 8: Hof-r-haus, north (Oliver Walter, 2009.)

4.2 Toolsets in the lighting design process (additional workflow)

For each lighting project the production steps vary and there is no typical outline for the workflow. Since the lighting design is a fairly new discipline, it has only a handful of professionals and therefore lack competition. This also has an impact on the workflow itself, which remains very individual for architects. This is partly because there are no institutions that would have a regulatory impact on the design process or the detailed content of each design step. This would be good in order to ensure more equality among the designing studios, as well as to guarantee that the various other engineering activities beside the architectural design can interact seamlessly.

Oliver Walter himself considers the lighting design process more an artistic than a technical task. Even so, it is clear that proper design need to be backed up with calculations to prove that the lighting values fulfill the required recommendations.

Instead of identifying a typical workflow for lighting design process, Walter shared his work approach as in two design toolsets that are currently utilized. These two techniques complement each other and reassure that the visual representation of the design corresponds also in level of empirical effectiveness.

Toolset 1	CAD, 3D modeling software, rendering software
Purpose	Visual representation of the design idea

Often the lighting solution has a structural impact on the building, as well as the environment. In other words, lighting installations need to be attached or hidden behind construction elements. Accurate measured drawings make sure that this happens in accordance with the space and structure available.

Architects need to represent the design proposal as a three dimensional model to test lighting alternatives and to create presentation images for the client. In many cases, standard renderers that Blender is able to produce are enough. Architects can also use the internal renderer when the representation does not need to be physically correct but instead to give an approximate impression.



Figure 9: Stockmann lighting project. (Oliver Walter and Julle Oksanen, 2005.)

In many cases architects need a more accurate picture to see how light affects the context. There are a variety of programs that are based on accurate raytrace calculation and are able to simulate the light under real-world conditions. Luxrender is an open source program that is

closely linked with the Blender interface and Mr. Walter favors. Lighting objects like luminaries can be associated with photometric IES files and the photometry of a light fixture describes the way it distributes its light into space. The rendering process is considered to be slow process, but often high quality images are not required when the pathways of lightning rays are investigated.

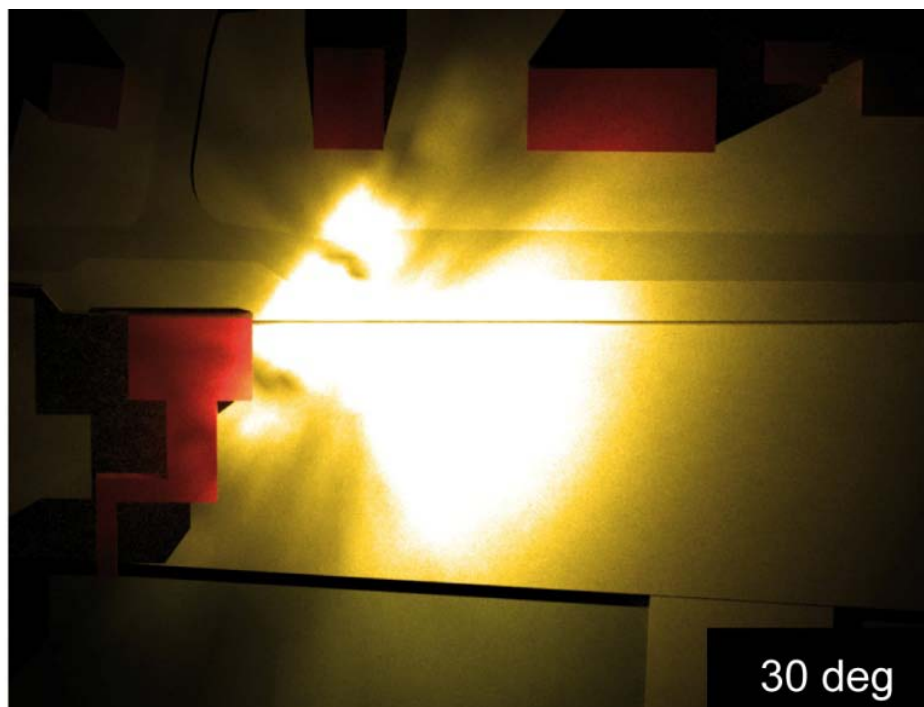


Figure 10: Blender light pollution render. (Oliver Walter, 2009.)

Toolset 2	Lighting level calculation software, Dialux and hand calculations
Purpose	Visual representation of the design idea

Toolset 2 enables architects to get real lighting values, luminances and lux (light that is being projected onto a surface) values. Professor Julle Oksanen has long experienced in calculating the lighting values by hand and Mr. Walter has been working closely with him over the years.

Dialux is a free program which reads manufacturers data of lighting fixtures and it is able to imports CAD files in DXF formats. Luminarie types of diverse manufacturers can be placed directly into the scene, the program then calculates the lighting values and finally presents the results in colorful charts and diagrams. These can then be presented to the client and used for light designing purposes.

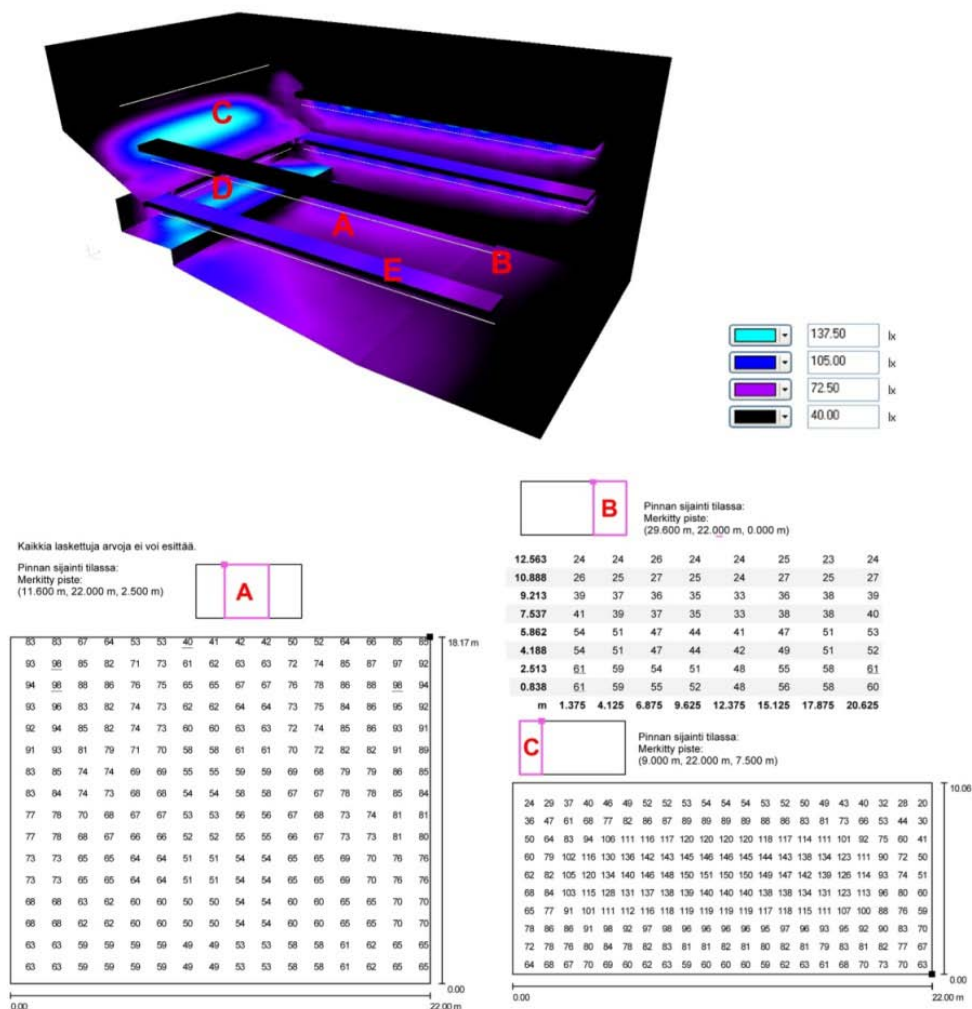


Figure 11: Concerthall lighting and value calculations. (Oliver Walter, 2006.)

4.3 Evocativi Ltd (basic workflow)

Evocativi is a medium sized company that functions worldwide, creating photorealistic 3D environments mainly for use in architectural visualizations - according to drawings and blue-prints delivered. The idea behind the business is to have 24-hour production cycle that can offer high quality and cost-efficient results for all tasks. By having subcontractors, freelancers and employees scattered around the world, the production is potentially more efficient than having them only in one place. Blender and SketchUp also enable that almost anyone can be hired without need for them to own or buy expensive software licenses. This way the concept can be applied even to the countries with low income levels, where the cost of buying an expensive 3D suite license could lead either to piracy or diminishing profit margins.

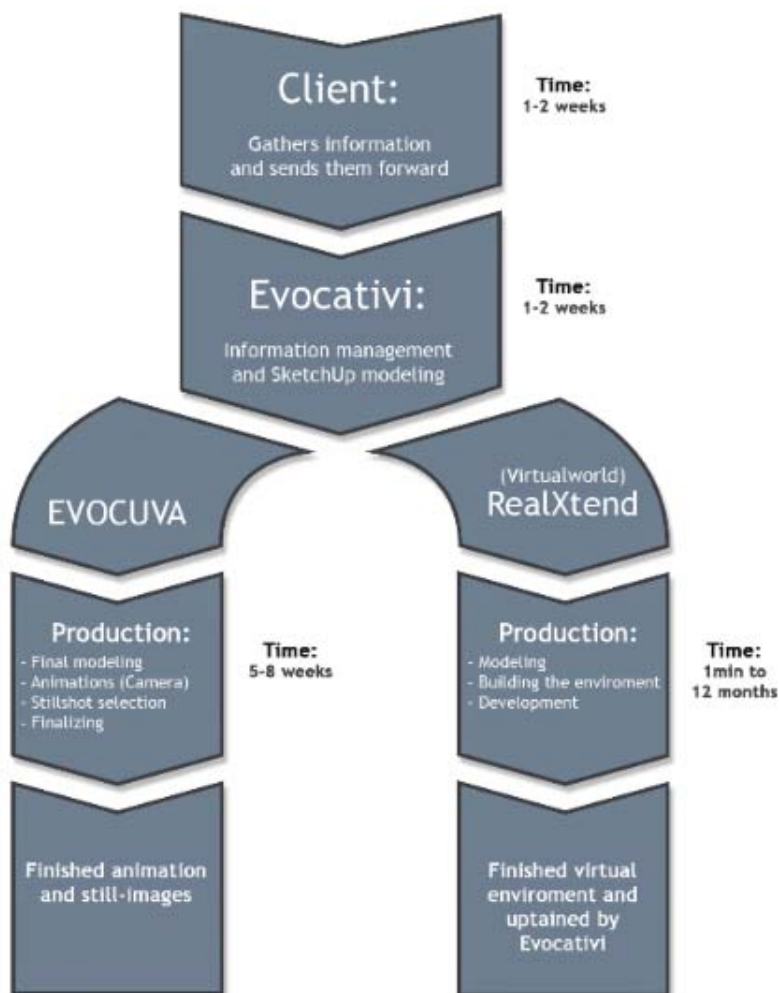


Figure 12: Evocativi's complete workflow diagram

Evocativi utilize mainly two different workflows that can both include collaboration between the client and possibly other involved parties. The first phases of the work have been planned so that the co-operation is maximized at the beginning, leaving smaller margin for errors and to get larger input from the client. After this the workflow diverges (see the figure 12. cross-road) and follow either one of the paths, depending on what results are required by the client. This chapter will introduce the basic methods for producing realistic 3D visualizations, which is Evocativi's main focus. The additional workflow is explained in its own chapter 4.4 and it concentrates on building virtual environments, like realXtend.

4.3.1 Drawings from the client

The process begins when the technical drawings are acquired from the client. The commissioning architectural company can provide them either by plain paper drawings or alternatively in any CAD format. The drawings are then used to create the SketchUp model, which is

explained in the next chapter. The drawings work as a blueprint for the 3d model, but do not need to be finalized, since they can be used simply to visualize different options for example.

4.3.2 Preliminary SketchUp modeling

Tool / Software:	SketchUp +Google warehouse
Process:	<ol style="list-style-type: none"> 1. The model is created inside SketchUp by using CAD-drawings or imported straight as 3Ds or DWG -file 2. SketchUp model is forwarded to partners and/or clients 3. Needed changes are made 4. Screenshots and animations are demonstrated inside SketchUp 5. Final and approved concept is then sent to actual production
Purpose:	To have exact information of the architectural concept and to create basic model structure
Strength:	SketchUp offers the tools and possibilities to have efficient workflow between all involved parties

In case the architectural drawings are in 2D format (figure 13.), the 3D model needs to be created by hand. The time required to do so corresponds directly to the complexity of the drawings. Thanks to SketchUp measurement system the task is relatively easy perform, it can even be done by non-professional and the results can be still good. The accuracy itself also helps in later stages of the production, since Blender is unable to deliver such precise results in equally short time.

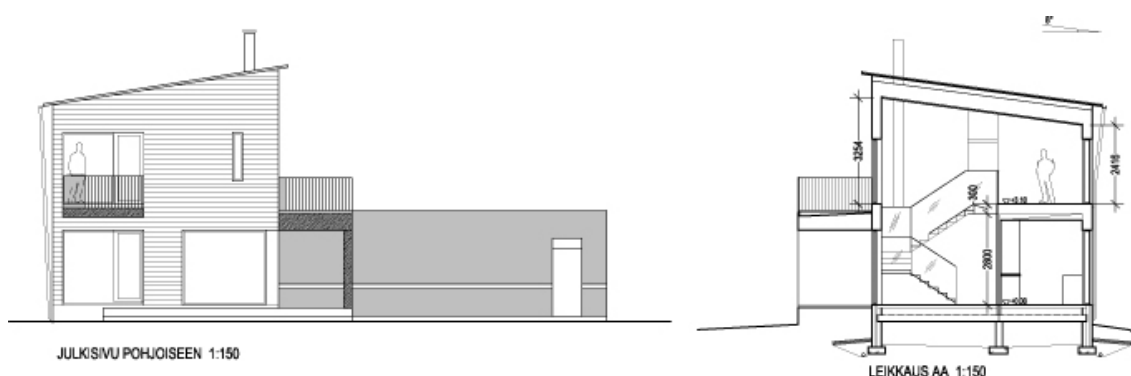


Figure 13: Example materials delivered by client. (Evocativi, 2009.)

If the drawings are delivered in three-dimensional format, the file can be imported straight into SketchUp by using following formats: SKP, 3DS, DWG, DXF, DEM or DDF. Other types can be converted into such formats by using a software called Accutrans, which was introduced in chapter 4.1.4 as a free alternative.

The model is then shared between all involved parties and developed further in close collaboration. The purpose is to streamline the process and share work between professionals of their own fields, delivering high quality results. These professionals can represent element manufacturers, like company producing window frames or other fundamental parts. Regardless of this, Evocativi or alternatively the client usually creates the basic SketchUp model. Sharing the workload this way will speed up the process and deliver ideal results.



Figure 14: SketchUp model from actual Evocativi project. (Evocativi 2009.)

After the structure is complete, the building is furnished by using free models from Google Warehouse or from Evocativi's own library. The idea is to quickly create prototype of the interior and exterior. The objects themselves can be downloaded straight into SketchUp by using the software's own search feature, and simply placed to the scene as they are. This way the building can gain more authentic look within minutes and give better explanation for interior designers what is wanted. Google Warehouse can offer a good leverage to speed and overall efficiency, even though the models are likely to be replaced in later production. The initial furnishing can also be done by the client, which is one strongpoint of the software and the overall workflow used. Example result of this can be seen in figure below.



Figure 15: Interior of the previous building. (Evocativi, 2009.)

After the initial furnishing is completed and the materials are selected, the interior designers can continue from there. This way the textures can be defined more closely and not just to determine that object is made from wood. Closer definition is required for the production phase, since the artists need adjust materials according to these specifications. The same process is gone through for all items, elements and other objects that are essential. Client can outsource this designing process to any company or hire the designer to work beside the artists at Evocativi. This way he/she can coordinate working more efficiently and add required details instantaneously. With this kind of workflow SketchUp can offer a flexible working environment and the materials needed for final 3D production.

In business terms, SketchUp is able to offer value, efficiency and a new way to produce high quality products. The final results may not be dependent of the model created in SketchUp, but the program itself can streamline the production more efficiently. Also, by distributing the model to the client and other partners will reduce the amount of errors in the final production run. This will benefit all involved companies from a financial point of view, as well as give them possibilities to affect to the production development.

The workflow phase is also demonstrated in the figure below, indicating more clearly that the production is thought as a separate part. SketchUp is not only tool for streamlining, but it also enables it better than before. Since the tool can be distributed free, anyone can create or modify existing models and there for give valuable input for the project.

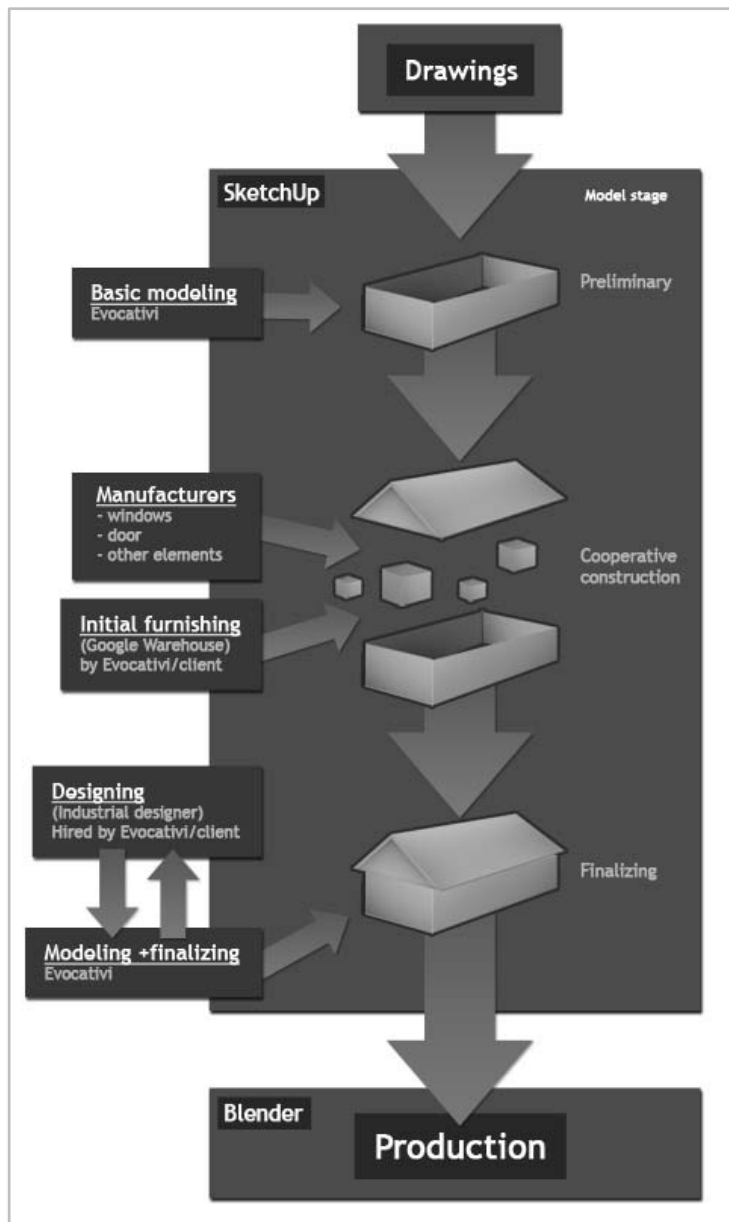


Figure 16: Workflow phases with SketchUp

4.3.3 Production

Tool / Software:	Blender
Process:	<ol style="list-style-type: none"> 1. SketchUp models are imported to Blender 2. Models are textured, polished and finalized 3. Proper lighting is created 4. Camera positioning and animations
Purpose:	Conclusive modeling and -animation is created for the rendering process. Purpose of this is to maximize the product quality.
Strength:	Blender can offer various tools for high quality production and it can be freely distributed

After the SketchUp model and its interior has been completed, the transformation process can begin. Turning the rough models into photorealistic representations require much work, which is mainly because the high requirements that Evocativi enforces. When watching the company's previous work, it can be easily seen that the rendered animations look life-like and not computer generated.

At this point the decision is required to be made, are the models going to be used for game environment or in a pre-rendered animation (see figure 12. crossroad). If game models are decided, the working methods differ dramatically from the ones used in cinematic production and this is explained in chapter 4.4.

All of the 3D models are inspected and depending on the artist, detailed into the final form. This is due to the fact that the detailing methods vary from artist to artist, but the final goal is to have photorealistic representation of each and every object. To succeed in this, Evocativi has scattered the workforce around the world and there for allows production to run 24 hours a day. Detailing is performed locally in each studio and then assembled at Evocativi's headquarters at Oulu, Finland. This also shortens the time required for each project, and therefore cuts the overall costs to produce cinematic visualizations.

Evocativi enforces strict quality standards on the final renders, aiming for the best possible results. The lights, materials and everything else in the scenes are created for maximal resemblance to real-life, including the smallest details. The level of quality is monitored throughout the process.

4.3.4 Project finalizing and delivery

Tool / Software:	Blender +VRay
Process:	<ol style="list-style-type: none"> 1. Final video- and screen captures are demonstrated to client 2. Final adjustments are done 3. Scene is sent to render farm 4. Composition and post production 5. Product is delivered to client
Purpose:	Final stage determines the camera animations, what is to be rendered and render farm usage

In the final stage, the client will be approached with a showcase that demonstrates camera animations and still image shots. The presentation itself can be done with a simple animation, and after the approval the actual heavy rendered can be executed. Altering the camera positions- and animations should be done no later than in this phase - as the rendering can take a very long time to re-do. For the rendering itself Evocativi uses third party engine called VRay, that enables as high quality results for both 3Ds Max and Blender.

Evocativi uses their own machines as the render farm, which means that the animations and images are generated using other computers than the ones employees use for daily work. This automatically means that the time required for rendering is greatly shortened by using such method and the workstations are relieved from the heavy burden. Companies benefit from using render farms the most, since the time to render is directly connected to expenses. Although, the cost is higher if the farm is hosted within the company and to avoid these extra expenses the free Renderfarm.fi -service is worth trying. The final adjustments can still be done afterwards and passes can be rendered individually, as professionals like to have them.



Figure 17: Final results (Evocativi, 2009.)

After the project is ready, all materials and models are delivered to the client. Final renders are often added into Evocativi homepage for sales purposes and to advertise the company capabilities. Please visit www.evocativi.com for more information.

4.4 realXtend (additional workflow)

The previous chapters introduced an actual workflows for rendering cinematic animations, but Evocativi also uses virtual environments to present their visualizations. This chapter explains the differences with game models, how they should be created and focuses on the requirements that they hold. For this reason the subject is approached in different angle than before and like mentioned earlier, the emphasis in the actual models. This is merely due to the fact that architects are commonly unaware how these models are created and what limits they hold.

4.4.1 Designing a game model

The workflow continues from where the SketchUp model was imported into Blender (chapter 4.3.3). The modeling and texturing techniques methods vary between artists, but disregarding the target 3D game environment, the basic structure of the models still remains the same.

There are three rules that should be remembered when creating game models. The first one is to have low amount of polygons as possible. Secondly, use textures efficiently, and finally reuse the same materials multiple times without letting user realize it. All interactive virtual environments exhibit these three factors and only by following them the best results can be achieved.

Model

The models are initially created by using a low amount of polygons and depending of the element, might be transformed into very high - up to several million polygons and then projected back to the low poly version. The idea is to present the shapes in the textures and not in the actual object geometry. This is due the fact that games run real-time and need to keep the refresh rate high enough (typically over 24 frames per second). Modern computers can run millions of polygons with decent or good results, but the low spec computers have to be always taken into notice.

Textures and UV's

Before any textures can be applied on the models, they needs to have UV -data. This means that the 3D surface is flattened into a plain 2D format and the images (textures) can be

placed over it, creating the visual look. By combining the different textures, applying material options and adjusting them correctly will finally create materials, which work in similar than in cinematic animations.

Commonly used textures (maps) in game engines:

- **Color map** is the only texture that is mandatory, and which determine the physical material of the object. The image can hold any combination of various materials and in game environments should contain lighting (AO) baked on it.
- **Specular map** determines the shininess of the surface. The effect is achieved by using a black and white image - where black represents the areas that do not reflect any light and white is the opposite.
- **Normal map** reacts to light and creates the illusion of complex surface. This way the model can look very convincing and realistic, especially when combined with specular map that strengthens the effect.
- **Parallax map** creates illusion of deeper depth by altering texture coordinates, which depends the camera angle. This map is most commonly used with static objects as walls, like brick- or stonewalls.
- **Reflection map** is the reflection that the surface shows. Combined with another texture that determines the areas affected, the result can be very convincing and realistic.

Baking ambient occlusion and normal maps

To give the object finalized look, ambient occlusion is applied into the color map. This means that global lighting is rendered into the object texture. By faking the shadows, object can look more authentic and have a sense of gravity affecting them. This is a simple operation that can be made within most 3D software. In Blender the feature is accessible from the render menu, under "bake" and the feature work well without any custom adjustments, as long as the Ambient Occlusion is switched on.

Normal maps can be created in a similar way to AO mentioned before, but the difference is just that there is now target from which the data is from baked. This means that there is the high poly model with millions of polygons and the game model with only few hundred. The process is usually fast and requires little adjustment to work, as long as the two models are approximately the same size and shape. After a successful bake a normal map should have been created to represent the complex shapes of the high poly model.

Reuse of models and textures

After the models are ready, they can be imported into the realXtend environment and used as they are. The procedure differs from the photorealistic workflow greatly, since there is no

need to render and final results is displayed real-time. Also, the changes can be applied easily on the fly, even if the models are already in place. This also helps when same model is used in several places, the materials can be easily switched to others and new variations can be created fast, without wasting too much memory.

5 Adaptation to a new workflow

The case work explained in the following chapter is an actual project, provided by Evocativi. The decision was made that Oliver Walter and Jani Lintunen would execute the work, where the primary goal was to demonstrate Renderfarm.fi and Evocativi's workflow in commercial use. This thesis will document the workflow and the final results that will be automatically uploaded into Renderfarm.fi service, under gallery.

5.1 Delivering the SketchUp model

Process:	<ol style="list-style-type: none"> 1. Discussion with the client (Evocativi) 2. Workload distribution 3. Schedule planning
Purpose:	To ensure fluent workflow during the project: all tasks, schedule and meetings were decided



Figure 18: Delivered SketchUp model screenshots

The SketchUp model above was created by Eero Tervo (the CEO of Evocativi) and it was used as the base model during the whole project. In a few short meetings the materials and overall look were decided, since the house was still under development and did not have them specified yet.

By knowing the size of the house, it was possible to determine how much work it would require. The reserved time was approximately two- to three weeks and the plan was to create fully furnished interior and simple outdoor scenery.

5.2 Adding furniture and details into the SketchUp model

Process:	<ol style="list-style-type: none"> 1. The SketchUp model is improved according to the clients wishes 2. The results are accepted or declined 3. More detailed schedule is made
Purpose:	The interior was furnished by using Google Warehouse free models, and this way the upcoming tasks and complete overview was seen instantly

The pilot project did not have any specifications or requirements for the interior and there for it was furnished within few hours, just by using Google Warehouse. This way the scale of the project was seen better and the workload could be estimated more closely. Like explained in chapter 4.3.2, some projects might have professional designers for decorating purposes, but in this case there was no such need. The decoration was simply there to give a good sense of space. At the same time a list was made to determine which objects are mandatory and which are not.



Figure 19: Updated SketchUp model

After finishing the SketchUp model it was approved by Mr. Tervo, after which the work continued normally. Most of the furniture was decided to be used in the final scene and were actually based on real design brands. The unnecessary items were just deleted or replaced with better suiting options.

The tasks were divided so that Oliver Walter was responsible for the structural- and elemental features, as well as overall lighting of the house. Creating the detailed furniture, items and composition was assigned to Jani Lintunen. It was estimated that the workload would be evenly divided, and in the end it was.

5.3 Importing objects into Blender

Process:	<ol style="list-style-type: none"> 1. SketchUp models are taken into Blender as OBJ format 2. Models are then fixed into appropriate geometry 3. Details and textures are created 4. Steps 1 -3 are repeated to all objects
Purpose:	The SketchUp furniture will act as a guidance for final look and some of them might even be used as they are. Access to free models will also grant fast and flexible workflow.

Since Evocativi has no need for game engine implementation, the scene was created with higher cinematic standards. This meant that several time consuming limits were neutralized. Before the final decision was made, the environment was also thought to have second purpose in realXtend, but in the end it was not necessary. The same models cannot be used in both purposes, unless quality standards are greatly lowered.

The figure below presents the model that was imported straight from SketchUp into Blender. At first glimpse the geometry might look good for modeling purposes, but in without proper preparation the model geometry needs to be rebuild completely. This can be avoided by following the instructions in attachment 3, SketchUp to Blender.



Figure 20: Imported SketchUp model

In Blender all geometry was inspected throughout every object and corrected if needed. Clear polygon structure means easier modeling, UV mapping and finally, decreases the amount of surface errors. The exported model can also function as a reference, which then can be re-created completely to suite better the requirements.

The next step was to get reference images of all objects. This is to ensure that the end results would appear life-like, to have new ideas and see what materials are they made from. For example the sofa is basically just fabric, but it is necessary to know what kind of fabric and how it looks like. All the little details, like wrinkles need to be taken into notice and how they form around the surface.

It is recommendable to do background research and when the model is taken into sculpting phase, the larger details can be crafted successfully by hand. This way the results will not look like something that was modeled as perfect, but to have the illusion of not being done perfectly in all aspects - same as in real-life. In sculpting the shapes are created by painting shapes into model surface, creating distinctive forms. The process reminding traditional clay sculpting, but no actual mass is added. The model surface is rather edited to have the feeling of sculpting. With more simple models the sculpting is not necessary, since the process can be replaced with easier and faster techniques. It is up to the artist and how he/she wants to proceed.

The image below represent one of the results that were produced, demonstrating the sculpting outcomes and how non-static the model can be made by using the technique.



Figure 21: Simple render of final model version

5.4 Creating the scene

Process:	<ol style="list-style-type: none"> 1. All models are divided into separate <code>.blend</code> files 2. Single scene is created and all objects are linked there 3. Environment is created
Purpose:	By linking everything into one scene, the work can be done simultaneously. This way a group of artists can avoid overlapping saves, share new models automatically and update them.

In order to have fluent workflow between multiple artists, the file structure and linking mechanism needs to be adjusted correctly. Even if only two people work simultaneously, this is still very important. To avoid any further issues, all of the models were done in separate `.blend` files and linked into a single scene by using groups. In case something is updated, the changes occur instantly in the final scene. This way overlapping saves can be avoided and the files kept relatively small, which was important for the backup system used.

The other thing that was done in this stage was the environment. It was built in the final scene itself and this way differed from the other objects used. The reason for doing this was that the results had to be seen with the house and adjusted accordingly. Having the environment in the final scene would also make the later stages easier. When cameras are created, it is important to see results instantly and edit the environment at the same time.

5.5 Lighting setups

Process:	<ol style="list-style-type: none"> 1. Direction of the sun and overall lighting is decided 2. Lighting setup is created separately for in- and outdoors 3. Test renders are made to ensure right material setups for different objects
Purpose:	Artist worked together to determine optimal lighting setups and to have best result as possible

The lighting setup was created in two separate sets, where exterior and interior were divided into their own layers. This way the affected areas could be controlled easily and the lighting switched between the different setups.

The lighting itself had to be built with Blender's internal render engine limitations in mind. It was not possible to use any external renderer and thus most of the lighting effects had to be faked. To clarify this term briefly: An advanced raytrace engine can calculate the lighting bounces, along with the colors that bleed from the object surface extremely well and realistically. Since these effects are still poorly adopted into the Blender's own render engine, the scene requires several additional lights to create an authentic result. This method is often called faking, since the light sources are un-natural and do not resemble real-life arrangement. The floor is a good example. Since it was rectangular in shape and one area light was enough to cover the whole surface. The light was placed under the floor - pointing upwards, so that the walls could be blended better with the floor by faking a color bleed. This also lowers the rendering time.

Test renders were made frequently after this phase, mainly to ensure that the materials correspond correctly and look real. This was to ensure correct values in materials and get them fixed if needed. Even the slightest change could mean a great deal in the final render.

5.6 Camera animations

Process:	<ol style="list-style-type: none"> 1. Camera cycles are created 2. Animations are recorded by using screen capture software. This saves time and is approved/declined by the client
Purpose:	Cameras are animated with architectural aspect in mind. The cycles go through important aspects that should give good coverage of the building. This animation is then recorder with simple screen capture, so no heavy rendering is required.

Camera animations were created mainly by using paths, constrains and simple animation features. The camera placement and -movement was something that is commonly seen in architectural visualizations. The house was covered from outside to inside and the purpose was to give a wide view.

At this point the camera cycles need to be tested. The best solution for this is to use a screen capture software and record viewport in real time. This saves time and removes the need for a rendering process. This way the complete picture is seen faster, no computing power is required and the resolution can be as large as the screen. Software like CamStudio can be used for this purpose.

5.7 Final adjustments

Process:	<ol style="list-style-type: none">1. Test renders are made and composition is created with <i>node editor</i>2. Node values are then animated to fit camera movement and surrounding conditions
Purpose:	

The final adjustments were done with Blender's node editor. Nodes are basically mathematical functions that are added and connected to other nodes, altering the for example colors. Tool like node editor is commonly used to enhance final output quality and the same functions can be found from most video editing softwares, like Adobe After Effects. By combining these nodes the result can greatly differ from the original render. It also saves time from post-production phase - as the effects are usually done only after the rendering phase is completed and the whole animation is ready. Figure below is a test render from later stages and it demonstrates the power behind simple lighting setup, which uses previously mentioned light faking.



Figure 22: Pilot project test render result

Final adjustments are done by using node editor and going through all materials. The figure 23. is a screenshot from the scene node editor and on the left side are the rendered layers (so called passes). The layers are individual images rendered separately, like diffuse colors, ambient occlusion (AO) and shadows. They are then combined and used to enhance default render, which could need stronger shadows and brighter colors for example. The biggest difference is usually made with AO pass, which was blended to the image as 11% opacity and with multiply filter.

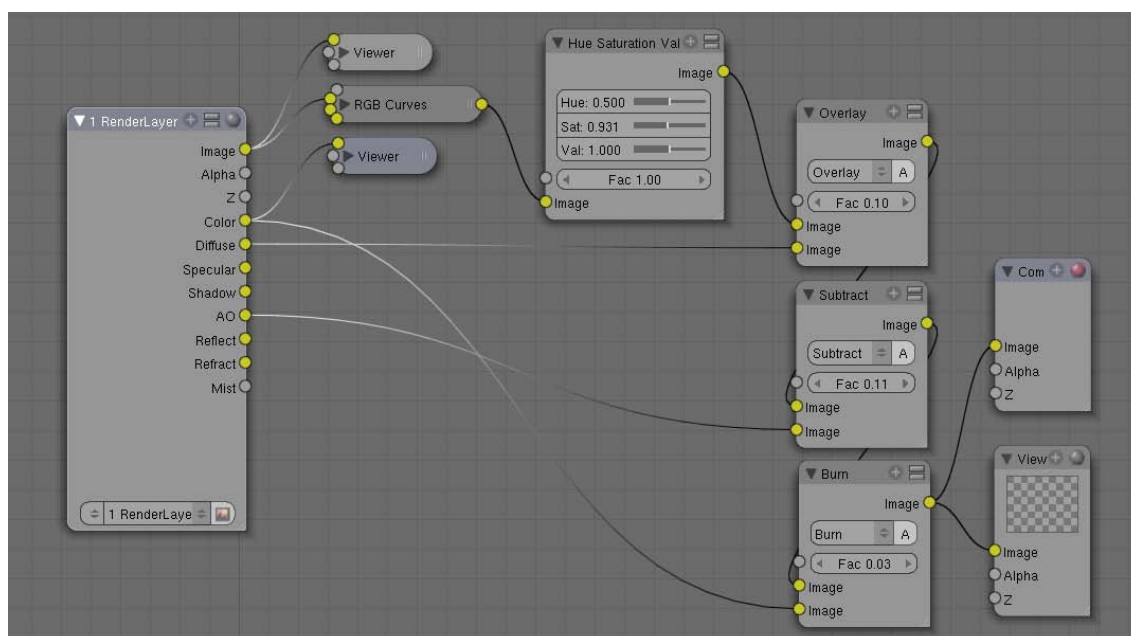


Figure 23: Nodes used to produce finalizing for previous image

5.8 Rendering

Process:	<ol style="list-style-type: none"> 1. Packing all into a single .blend file 2. Render setups and limitations of render farm are checked 3. Render
Purpose:	The scene and all of its data is packed into a single file, so that the render farm can more easily distribute it to clients. Since Renderfarm.fi was used the procedure is necessary.

The final animation did not require any special effects to be used, like for example fluid- or smoke simulation. Even though there was nothing like that to bake inside the Blender file, the

scene needed to be packed into a single file that contained everything (models, materials, textures, animations, e.g.). To decrease the final file size, most of the textures were resized. Some of the images were changed from PNG to JPEG and the resolution made smaller. This way the scene took well under the maximum 250Mb of the render farm limit and the scene still had the original details wanted - without suffering any visual loss.

It is worth noting that the compression tends to ruin little details of the image and it is therefore recommended that format like JPEG are used carefully. For this reason it is wise to keep good resolution copy as a backup. In this project, most of the textures had their own Photoshop files that were exported in smaller size.

When using distributed rendering the file size is something to keep in mind, reason for this is that the whole file needs to be sent for hundreds- or even thousands of users across the world. Renderfarm.fi has a maximum limit of 256Mb, and it can be bypassed if needed, but this would not benefit the service or the users.

Other limitations that the pilot had, due to the requirement of frames being rendered by multiple BOINC machines (16 per image):

1. SSS (*sub surface scattering*) means material that allows object to cast light through its geometry. The idea is to calculate how the object scatters light into the surface and display, for example organic behavior. Because the light bounces need to be calculated and smoothed, the process has some requirements that distributed rendering will not support. SSS will work normally only if the frame is not shared with other computers. The result will otherwise show seams in the final animation and the same goes for most of the other visual limitations.
2. Unlike SSS, the following effects are created in Blender with node editor. The difference is that nodes work as post-production tools and can be done afterwards with many other softwares, where SSS need to be rendered into the picture. None of these are usually mandatory, but highly recommended, so the real limitation is the lack of SSS feature when dividing frames.
3. The one of the most common effect that can be found from 3d renders is combination of blur and depth maps. The result is usually called *depth of field* (DOF) and it means that the focus point is determined by sharpening "important" areas, where back- and/or foreground is blurred. In rendering this is generated afterwards and it requires the whole picture to be handled at once. Otherwise the image will have clear seams, meaning that the task is done piece by piece.

4. Glare and glow effects are also similar to the DOF mentioned above. The idea is to separate bright spots from the final render, blur them in the right manner and mix them back in the original image. This way the renders will be a lot more realistic. As with the DOF, glare and glow effect the whole picture needs to be processed at once.

6 Benefits of using open source 3D tools in companies

Blender is widely known for the advanced tools in for example modeling, which is why some professionals want to use it as part of their workflow. The software can be easily implemented into any workflow completely, but companies and their clients are often afraid of its affect on the final product quality. It is true that some of the features do not match the qualities in competing software, but the same goes for all 3D programs. It is recommended that companies would adopt Blender gradually and for specific parts of the existing workflow. This way the results can be observed more easily and the software value seen in the management side, which usually is the biggest obstacle with this kind of implementations. For companies the overall goal might not always be to save money when using open source programs, but to have tools that enable better production speed and quality. Saving money can be just an additional long-term benefit. Also, by using free software the organizations can expand easier and faster, as licenses are no longer the problem and option for piracy is eliminated completely. A good example of this is Evocativi Ltd, which introduced the new working method for 24h production without having to invest large amounts of money into required licenses. It is also beneficial for the client side, as they can easily preview the work using the same software it was made with, freely available for them as well.

From a company perspective, the openness of 3D software could become very important factor in time. When using software like Blender, the actual source code is possible to be modified completely, but with other commercial programs this could require considerable investments. The edited version of Blender can even be sold forward as their own product, without paying any commissions to Blender Foundation or other parties. By having open license the modifications can be made without investing as much money as it would require otherwise.

As an added bonus, the organizations can benefit from the publicity offered by certain open source programs, such as Blender. Projects done completely or even partially with the software can get a great deal of publicity, and the results are often displayed on many famous websites. Keeping the business aspect in mind, this is an intriguing way to get advertisements.

The previous chapters have explained how Blender and SketchUp can be used for architectural visualizations, but they also introduced some of the key downsides. No program can be the best at everything, not even Blender or SketchUp, but as far as money is considered they are able to deliver professional tools without a price tag.

7 Conclusions

Streamlining work in companies is essential for having successful workflow and above all, fast production. When using open source software available today, organizations can be more efficient - and save costs without losing product value. To understand this concept, this thesis has demonstrated workflows and results achievable with them by using just open source software. The tools are after all free and the only thing required is to have talented people working with them.

Proprietary programs used in professional studios can cost thousands of euros, but high cost does not guarantee high quality results. Many freelancers today acknowledge the potential that Blender and other similar no-cost or low cost alternatives offer. The fact that Blender is open source, something that still makes surprisingly many decision makers within companies vary, unfortunately leads many to bypass this excellent product. To understand the value of open source, companies often need someone from the inside to take the initiative and demonstrate that the same or better results can be achieved by alternative methods.

Nowadays social networks and the Internet provide a vast amount of information, scripts and free materials that cover Blender features well. Well-known artists like Andy Goralczyk have even earned a lot of reputation just for being active users of Blender, which has also a positive effect on the reputation of Blender. It is common to see creative professionals start using Blender for their own work and adopt the new tool with positive reactions. Goralczyk's work can be seen at www.artificial3d.com.

This thesis was created in order to research and demonstrate how Blender and SketchUp can be used as professional tools. By acknowledging the capabilities that these programs - the other open source and the other relatively low-cost - demonstrate, it is possible to maximize the output. For example the render result can be, according to Evocative preventive: "almost exact between 3Ds Max and Blender". This was tested in practice by their behalf, by using external renderer called V-Ray. It supports both softwares equally, offering substantial proof that Blender can today be considered a serious competitor to proprietary software.

References

About GPL license. Blender.org site documentation. Accessed 12.10.2009.
<http://download.blender.org/release/GPL-license.txt>

About Renderfarm.fi -service. Renderfarm.fi site documentation. Accessed 3.11.2009.
<http://www.renderfarm.fi/page/about-renderfarmfi>

CGenie 2009 survey, Blender developer interview. 2009. Accessed 4.3.2010
<http://cgenie.com/articles/130.html>

Community survey made by CGenie. Used 20.11.2009.
<http://cgenie.com/component/content/article/47-articles/104-cg-community-survey-upgrades-09.html>

Hess, R. 2008. The Essential Blender: Guide to 3D Creation with the Open Source suite Blender. San Francisco, CA, USA: No Starch Press.

History of Blender. Blender.org site documentation. Accessed 15.2.2010.
<http://www.blender.org/blenderorg/blender-foundation/history/>

Import DXF to Blender. Blender.org site documentation. Accessed 28.10.2009.
<http://wiki.blender.org/index.php/Extensions:Py/Scripts/Manual/Import/DXF-3D>

Mullen, T. 2008. Bounce, Trumble and Spash! Simulating the Physical World with Blender 3D. Indianapolis, Indiana. USA: Wiley Publishing, Inc.

Ojasalo K., Moilanen T. & Ritalahti J. 2009. Kehittämistyön menetelmät: uudenlaista osaamista liiketoimintaan. WSOY.

Osipa, J. Stop Staring: Facial Modeling and Animation Done Right (second edition). Indianapolis, Indiana. USA: Wiley Publishing, Inc.

Seppä, A. 2006. Open Source in Finnish Software Companies (Discussion papers). ETLA, The Research Institute of the Finnish Economy.

SketchUp to Blender. Famous tutorial site, KatsBits. Accessed 20.1.2010
http://www.katsbits.com/htm/tutorials/sketchup_converting_import_kmz_blender.htm

SketchUp original developer. SketchUp's original creator homepage. Accessed 7.12.2009.
http://www.imvu.com/catalog/web_images/how_to/sketchup_index.htm

Yang, J. 2007. Construction Innovation: Smart and Sustainable Built Environment. Emerald Group Publishing Limited.

Table of Figures

Figure 1: Community survey made by CGenie. (CGenie survey 2009.)	12
Figure 2: Blender 2.5 alpha development roadmap. (Blender Foundation 2010.)	12
Figure 3: Blender feature review. (TDT 3D 2008.)	13
Figure 4: SketchUp screenshot of Google Warehouse -model	15
Figure 5: realXtend game engine footage 1. (realXtend homepage 2010.)	16
Figure 6: realXtend game engine footage 2. (realXtend homepage 2010.)	16
Figure 7: Screenshot of Renderfarm.fi -website	18
Figure 8: Hof-r-haus, north (Oliver Walter, 2009.)	24
Figure 9: Stockmann lighting project. (Oliver Walter and Julle Oksanen, 2005.)	25
Figure 10: Blender light pollution render. (Oliver Walter, 2009.)	26
Figure 11: Concerthall lighting and value calculations. (Oliver Walter, 2006.) ...	27
Figure 12: Evocativi's complete workflow diagram.....	28
Figure 13: Example materials delivered by client. (Evocativi, 2009.)	29
Figure 14: SketchUp model from actual Evocativi project. (Evocativi 2009.)	30
Figure 15: Interior of the previous building. (Evocativi, 2009.)	31
Figure 16: Workflow phases with SketchUp	32
Figure 17: Final results (Evocativi, 2009.).....	34
Figure 18: Delivered SketchUp model screenshots.....	37
Figure 19: Updated SketchUp model	38
Figure 20: Imported SketchUp model	39
Figure 21: Simple render of final model version	40
Figure 22: Pilot project test render result	43
Figure 23: Nodes used to produce finalizing for previous image	44
Figure 24: Blender version 2.49 standard import formats. 2010.....	50
Figure 25: Blender version 2.49 standard export formats. 2010.....	50

Attachment 1 File formats

.blend

Blender features its own file format that is able to pack multiple scenes into a single *.blend file. It can contain all objects, materials, textures, sounds, images, post-production effects, animations and everything else needed. Because the file is organized into separate data blocks, users can link or append basically anything from scene to any other Blender file.



3D Studio (.3ds)...	MilkShape3D ASCII (.txt)...
AC3D (.ac)...	Motion Capture (.c3d)...
Autodesk DXF (.dxf .dwg)	Motion Capture (.bvh)...
COLLADA 1.3.1 (.dae) ...	OpenFlight (.flt)...
COLLADA 1.4(.dae) ...	Paths (.svg, .ps, .eps, .ai, Gimp)
DEC Object File Format (.off)...	Pro Engineer (.slp)...
DirectX(.x)...	Raw Faces (.raw)...
LightWave (.lwo)...	Stanford PLY (*.ply)...
Lightwave Motion (.mot)...	Video Sequence (.edl)...
Load MDD to Mesh RVKs	Wavefront (.obj)...
MD2 (.md2)	X3D & VRML97 (.x3d / .wrl)...
MilkShape3D (.ms3d)...	

Figure 24: Blender version 2.49 standard import formats. 2010.

Import and export functions in Blender are done with python programming language (scripts) and they support large scale of different file types, which allows Blender to work seamlessly through various workflows. For example these formats are supported straight from the package and without any external scripts:

Figure X: Blender version 2.49 standard **import** formats. 2010

3D Studio (.3ds)...	OpenFlight (.flt)...
AC3D (.ac)...	OpenInventor (.iv)...
Autodesk DXF (.dxf/dwg)	Quake 3 (.map)
Autodesk FBX (.fbx)...	Raw Faces (.raw)...
COLLADA 1.3.1 (.dae) ...	Save Current Theme...
COLLADA 1.4(.dae) ...	SoftImage XSI (.xsi)...
DEC Object File Format (.off)...	Stanford PLY (*.ply)...
DirectX (.x)...	VRML97 (.wrl)...
LightWave (.lwo)...	Vertex Keyframe Animation (.mdd)...
Lightwave Motion (.mot)...	Wavefront (.obj)...
M3G (.m3g, .java)...	X3D Extensible 3D (.x3d)...

Figure 25: Blender version 2.49 standard export formats. 2010.

More information about file formats can be found at:
http://wiki.blender.org/index.php/Dev:Ref/File_Formats

.dae

COLLADA is a XML based format that uses data blocks to define content, which can hold nearly all 3d assets known. Support for COLLADA has grown rapidly to game engines and other popular software's that benefit from this open format greatly.

.dwg

Autodesk DWG ("drawing") format is basically an exclusive file type for all CAD programs that can contain two- and three dimensional design's, plus the needed meta data. The format is secured by watermark, so the original creator and the authenticity can be identified by anyone, which of course narrows compatible programs. This is also one of it downside's, even thou it is one of the most used formats among CAD's.

**.dxf**

AutoCAD DXF (Drawing eXchange Format) is a platform independent format type that does not require original application software, hardware or operating system to run. The file can contain any number of 3D -models, images, or any other graphical elements needed. The format can also have intelligent metadata that is able capture the design intent of the data being represented.

**.lwo**

Lightwave format was introduced in Oliver Walters workflow and this was because of its ability preserve geometry data, without errors. Unlike other formats that CAD softwares are able to export, by using IWO files the results can be maintained good.

.dwf

Autodesk's DWF (Design Web Format)



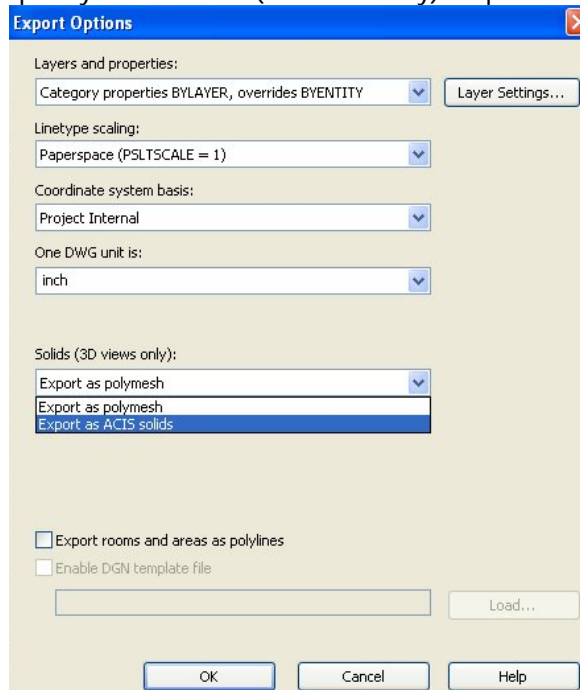
Attachment 2 Autodesk Revit to SketchUp

During the thesis research it came obvious that architects need to be able to import models into different programs. The problem was presented that Blender and Autodesk's Revit do not work well together, since between the import/export process data loss occurred and the geometry had to be rebuild.

One of the possible solutions is that the models can be alternatively taken into SketchUp and not Blender. This way the workflow introduced by Evocativi is still valid and can be used successfully in any architecture project.

The simple step by step procedure to import Revit model into SketchUp:

1. Execute **Autodesk Revit**
2. Open file
3. Export model as CAD Format
4. Click the "Options..." -button from the export window
5. Specify under Solids (3d Views Only): Export as ACIS Solids



6. Name the file and press "Save"
7. Execute **SketchUp**
8. File-> Import -> Then specify under Files of Type ACAD Files .dwg
9. Press "Open"
10. Make it active by clicking it
11. Press right mouse button over the model -> Hit "Explode"

Attachment 3 SketchUp to Blender

Whenever objects are imported from program to another, there is a possibility that the geometry needs to be rebuilt partially or completely. This can be caused by a compatibility error, unwanted tessellation or any similar event. Luckily SketchUp and Blender work well together and the models can be used directly, but some preparation is still recommended. The geometry should be modified so, that it can be easily used after the export and there are basically two common procedures required: Turn all faces into four-sided polygons or alternatively triangles. This is executed by default in SketchUp, during the export (automatic tessellation), but the results are often horrid. The second one is to delete all unnecessary geometry, since SketchUp will save the whole scene as one. (SketchUp to Blender. 2010.)

In addition to previous, when moving model from SketchUp to Blender the scale of the objects change dramatically. This is due to Google SketchUp using a larger scale system than Blender. In order to see the mesh fully in the 3D view-port, the view settings need to be changed. By increasing the "Clip End" value from view properties makes the view distance longer and makes the mesh fully visible. Another way to achieve similar results is to scale the objects down, but it needs to be done every time something is imported and might increase the amount of work in the long run.

The basic functions that should be performed after importing models to Blender are that unnecessary vertex points should be deleted: select all vertex points and pressing remove double vertices. One should also convert all triangles back to four-sided polygons and then recalculate the face normal's - making the geometry more usable.

For exporting and importing, OBJ format works the best and is able to maintain the same quality, without any losses.