A GENERIC METHODOLOGY FOR THE

3-DIMENSIONAL VISUALIZATION OF RADIATION FIELDS

by

Joseph G. Chaput, B.Eng.

Under the supervision of Dr. Ed. Waller

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science in Nuclear Engineering

In the Faculty of Energy Systems and Nuclear Science graduate program

University of Ontario Institute of Technology

April, 2010

ABSTRACT

The radiation field visualization options available for engineers, scientists and health physicists have traditionally been based in the 2d realm, with techniques such as the generation of isodose curves. From the perspective of a health physicist the creation of 3d visuals to illustrate radiation levels within an environment is an invaluable tool both for training and As Low As Reasonably Achievable (ALARA) radiation dose planning. This thesis describes a novel technique for the creation of 3d visualizations of radiation fields. The methodology is developed and shown to be effective within the Google SketchUp Computer Aided Design (CAD) software package. The methodology takes an input file of information stored in coordinate form with a representative value at each point. It constructs elemental shapes automatically within Google SketchUp at those coordinates. All shapes are associated with an intensity value related to a pre-defined scale. The shapes are colorized and enhanced with transparency effects to complete a radiation field visualization scene.

Keywords: Visualization, Radiation, Nuclear, Field, Google SketchUp

ACKNOWLEDGEMENTS

I would like to acknowledge the University Networks of Excellence in Nuclear Engineering (UNENE) and the Natural Sciences and Engineering Research Council (NSERC) for their support provided through the University of Ontario Institute of Technology (UOIT) UNENE/NSERC Industrial Research Chairs (IRC) in Health Physics and Environmental Safety.

I would like to acknowledge and thank my colleagues at International Safety Research Inc., for their overwhelming support of my post graduate education. In particular, I would like to thank Dr. Tom Cousins whom gave me my first opportunity within the nuclear industry in 2006 and has been involved in my career in several different capacities ever since.

Above all I would like to express my deep gratitude for the support from Dr. Ed Waller, whom without this research and my post-graduate education would never have taken place.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1: INTRODUCTION

1.1 Background

The clear communication of ideas within complex sciences such as the field of health physics and nuclear engineering is inherently limited by the audience's understanding of the fundamental processes which are taking place [1]. Communication of concepts such as varying dose rates within an environment is frequently required during the planning of activities ranging from decommissioning operations, environmental assessments to emergency response exercises. When communicating with the public the adage 'a picture is worth a thousand words' holds significant value as complex ideas are often most quickly understood with the right visual imagery [2].

Safety concepts such As Low As Reasonably Achievable (ALARA) radiation dose planning, are an important aspect of any occupational health and safety program [3]. Reducing time within a field, increasing distance between a user and a source and increasing shielding are three general principals which can be used to maintain ALARA. Teaching those responsible for working in and around radiation fields about the shape, size and intensity of the radiation environment around them is an important technique to make them aware of how to take advantage of the time, distance and shielding principals effectively. Finding new and novel methods to better communicate these concepts should always be in the interest of health physicist, scientists and engineers.

There are several computer programs which have been shown capable of displaying ionizing radiation fields within three dimensional (3d) environments [4-6]. These programs are adequate for their designed purposes but beyond that they have all been designed for specific purposes and for specific goals which do not include those of this research. They are also closed in design, too simplistic for general purposes and/or not widely available for the public, limiting their usefulness and impact. Intuitively it is clear that education and communication of complex ideas in the nuclear industry could be greatly enhanced by providing engineers, scientists and health physicists a simple and easy to use process to generate 3d visualizations of their work.

1.2 Purpose

This research originated with a focused goal to study, create and develop an effective procedure for generating 3d visualizations of radiation fields. This project sought to create and assemble a process that any engineer, scientist or health physicist could emulate, adapt and put into use for any of a variety of purposes that would be limited solely by the imagination and ingenuity of the end user. This research furthermore sought to create a complete process for generating 3d visuals that would be useful for augmented reality research in future applications.

1.3 Structure

The work in this thesis is based on a 'start from first principles' approach of development. It is designed to be relatively straightforward, covering first a simple analysis of 'how' one would generate a 3d visual of a radiation field in abstract terms. This is followed by an analysis of the conceptual problems and difficulties associated with visually displaying such a field. A discussion of the types of computer programs available for this type of field construction takes place which is followed by the development of the generic methodology and its demonstration using a specific computer program. This research ends with two illustrative scenarios where this method is applied and finally with a discussion of the implications of this work.

This document is structured in the following format:

- Problem analysis
- Conceptual analysis
- Selection of a 3d modeling program
- Creating a 3d radiation model
- Constructing a 3d radiation model in SketchUp
- Results
- Impacts
- Conclusions and recommendations for future work
- References and annexes

Each chapter begins with a brief explanation of the contents therein followed by its content.

CHAPTER 2: PROBLEM ANALYSIS

This chapter discusses the theoretical problems and challenges related to the development of a process for visualizing 3d radiation fields. The problem is broken down into various sub-problems which are addressed where appropriate throughout this thesis. These concepts are kept abstract where applicable to keep with the goal of developing an overarching methodology rather than a focused methodology to maximize the future potential of this work.

2.1 Problem breakdown

Theoretically the process to generate a 3d radiation field can be divided into four distinct phases shown in Figure 1.

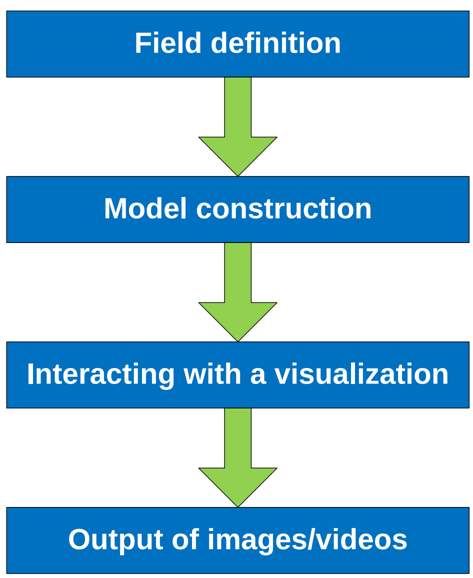


Figure 1 - Four phases to visualizing a radiation field

2.1.1 Field definition

Defining a radiation field is the first step in any visualization. A radiation field is a ubiquitous term in nuclear engineering and health physics which most commonly refers to the particle fluence and also often the energy distribution of some type of ionizing radiation within a medium, volume or other space [7]. The key feature of a field is that there is a quantifiable trait, such as the rate of particles entering each cubic centimeter of space that varies throughout an environment. For this research the types of radiation fields of primary concern are those caused by ionizing radiation, but the work herein may be applicable to the visualization of other types of radiation fields such as those from non-ionizing sources.

To describe any field requires information about that field. This information could originate through direct measurements at points within space, or calculations based on a mathematical description of a field. In general for this research it was assumed that if a generic approach to the field definition process (that is 'how' the user will define a field) was the basis for the field definition then overall flexibility of the methodology would be greater. Following this maximizes the types of fields able to be constructed as is shown in Chapters 5 and 6 and demonstrated in Chapters 7 and 8.

If this research was inherently reliant on a single definition technique, the final product could have been narrowly focused with respect to the types of fields that could be modeled. The solution was to keep the field definition process separate from the model construction process. The only exchange between the two is related to a set of outputs at the end of the field definition process. These outputs are needed for the construction method. The origin of these outputs is independent of the construction method and therefore could come from any number of generation techniques, such as an equation derived from first principles or an advanced physics code. The field definition process and expected outputs are discussed in further detail in Section 6.1.

2.1.2 Model construction

The representation of a radiation field will ultimately be via a 3d model of some kind which will be visualized as appropriate to create the 3d imagery. With respect to how the 3d model was constructed, the exact method used may limit what types of programs are able to view and interact with that field. At the onset it was desired that this research be based on freely available tools and programs as much as possible so that in the future this work can be used by any engineer or health physicist.

The construction process proposed in this thesis is based on an iterative, piece by piece (or point by point) technique. Building a model in many small parts potentially gives more avenues for errors to occur during construction, but alternatively the construction method is conceptually very easy to understand and adapt for specific projects and programs. Using fundamentally simple concepts maximizes the future compatibility of this construction methodology with other 3d modeling software. The only real requirements for this technique to be used in any 3d modeling software package is that the software:

1. Allows the automation of construction actions (such as the creation of a shape).
2. Allows transparency effects to be applied to surfaces in a construction.

This is discussed in greater detail in Section 5.2.

2.1.3 Interacting with a visualization

The final stage of the construction process takes place when the modeled field is used for its visualization purpose. In this stage, viewpoints are setup for their eventual output as static images or dynamic videos. The viewing / interaction process is entirely reliant on the construction process, as it limits which types of programs may open and view a constructed model. The assumption that the primary users will not be expert 3d modelers requires that this interaction process be as user friendly as possible.

2.1.4 Output of images / videos

Through the interaction process briefly described in Section 2.1.3, there will no doubt be a user requirement to collect and output images and/or videos of the modeled field visualization for inclusion into presentation material, reports or other media forms. The final consideration in the development of this methodology was that the end technique must make the process of generating this material very easy and straightforward.

CHAPTER 3: CONCEPTUAL ANALYSIS

Prior to the development of the generic methodology shown in this research, it was appropriate to identify the types of theoretical issues expected when visually describing a radiation field. This chapter describes some of the issues identified prior to the development of the generic methodology shown in Chapters 5 and 6. It explains how these issues were expected to encumber a 3d description of a radiation field. The terms used in this chapter were specifically developed for this research to describe the expected challenges that would need to be addressed throughout this work. This chapter concludes with the establishment of design requirements that this methodology will need to achieve.

3.1 Issues

3.1.1 Visibility

Visibility is an obvious and very important issue which needs to be addressed. For a radiation field to be completely 'visible', a computer model must allow a viewer to observe both internal and external details, similar to looking through a foggy window. This is an absolute requirement to allow all of the internal details within a field to be seen from an exterior vantage point; otherwise they would be obstructed by the outer most layer of information. These details could be something such as a change of dose rate within a localized area, or depending on the type of field being modeled, it could be a change in local particle fluence quantities or other relevant factor. This is illustrated within Figure 2.

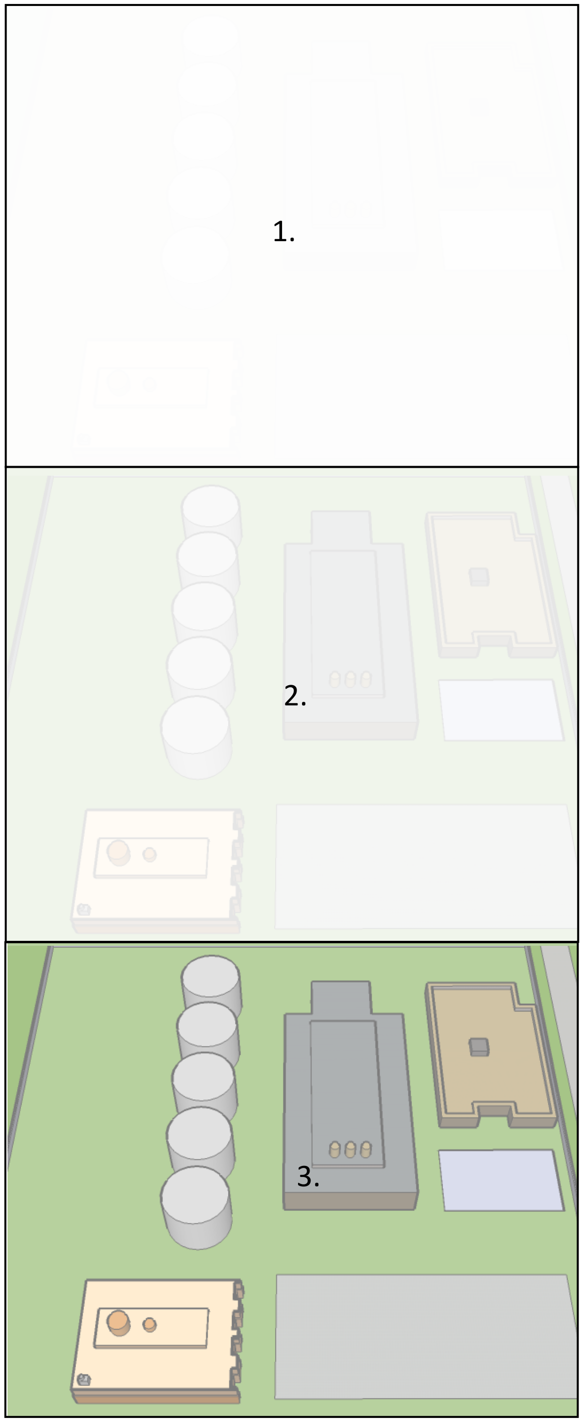


Figure 2 - Cloud visibility example

In Figure 2 there are three images in separate panels. All three images are identical except that each one has a 'fog' in front representing information that is blocking the user from seeing the background image. This obscures all of the internal details limiting the usefulness of the images. In panel 1 the fog is only 1% transparent (blocking the viewer from seeing any of the details). In panel 2 the fog is 10% transparent. The user can now make out certain structures of the obstructed images but the majority of the details are still blocked by the fog. In the final panel 3, the fog has been increased to 55% transparency. All of the internal details are visible but the user is still aware of the fog layer of information as well.

This rather simplistic illustration shows the types of visibility problems that will need to be solved by this methodology. These hidden levels of detail need to be made apparent to the user. The visibility of non-radiation models (such as background scenery) was also considered an equally important requirement in the early development of the methodology. The combining of both a radiation model and a scenery model requires some sort of transparency effect otherwise scenery details will be partially or completely obscured by the radiation model.

3.1.2 Model navigation

Navigation refers to how the end user will eventually 'move' around a modeled field. Movement within a modeled radiation field is an important feature so that the user is able to analyze a model from as many viewpoints as possible. The controls to move a viewpoint in an environment need to be as intuitive as possible, in particular for those users who may take this research and want to apply it somewhere, but are unfamiliar with using 3d software. How a user establishes multiple perspectives for a scene and manipulates a model plays a key role in the overall user friendliness of any program.

3.1.3 Colors / scales

Determining appropriate colors to be used within a field model is not as simple a problem as it initially appears. If models are made with non-intuitive colors, digesting that information can become much more difficult [8]. If a model uses too many colors within its color scale it could also be just as difficult to review and understand [9]. A set of recommendations concerning the use of color within a visualization will be required at some level to assist users in the creation of models.

3.1.4 Model portability to other software

Portability of a constructed model to alternative software is considered a desirable trait in this research. Once a field model has been created, its usefulness is directly related to the number of different analytical and visual applications available to the end user, which is dependent on the program used to view a model. This is primarily concerned with the data format of the 3d model. If a 3d model is stored in an openly documented and available 3d format such as the COLLADA format [10], it will be possible to use many different 3d viewers to view that model.

In addition to being compatible with a wide array of modeling software due to no licensing costs, the COLLADA format is based on the Extensible Markup Language (XML) [11]. This allows a COLLADA formatted file to be directly opened with a simple text editor program and properties of the file can be edited directly via the text editor, such as by adjusting the radius of a sphere, or coordinate location of a point. This type of editing is possible because of the open documentation of the COLLADA format and its XML backbone that can be read through a text editor. If a 3d model is stored digitally in a closed, undocumented or propriety format such as SLDDRW files [12], there will be limited programs available to read and display that model, limiting its usefulness.

3.1.5 Model limits

A model limit (also denoted as an 'extent') is a reference to where a radiation field model should be constructed and where it should not. There is a potential for a large amount of overhead in the development of any large or complex model. Reduction of this overhead may be possible by limiting the model to only the specific sections needed for a given scenario. This is illustrated within Figure 3. If the only area of interest was the field within the dashed square, then rendering the entire field on screen at once would increase the theoretical computer workload building the model, and also add additional overhead for the user during the field definition phase of their efforts. Constructing only the part of the model within the square that is to be visualized is easier to accomplish as well.

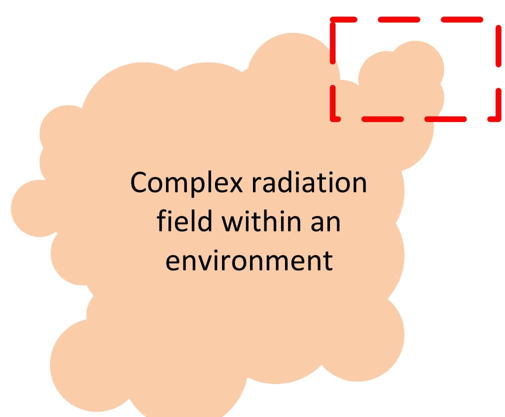


Figure 3 - Model limits example

3.2 Program requirements summary

Based on the issues discussed in Section 3.1, design requirements were established for the field visualization process that the methodology was to achieve. These requirements were used to guide the research and as a basis for an assessment of this work during the conclusions.

Table 1 - Generic methodology design requirements

Issue	Design requirement
Visibility	• Distinguish internal radiation field features without obstructing the view of other features
Navigation	• User friendly method to navigate a field model
Color/scales	• Documentation and instructions for end users to manipulate and apply logical color scales to best visualize a radiation field
Portability	• Open source file storage format to maximize the future potential uses of a modeled field
Model limits	• Capability to limit a radiation field within a region of interest
Others	• Open and easy to use such that any engineer, scientist or health physicist can construct a field model with a minimal amount of effort

The design requirements listed in Table 1 are phrased to be conceptually simply to understand. The 'Others' issue comes from the overarching goal of this research stated in the introduction and it is perhaps the most important goal. Once the methodology has been developed and successfully demonstrated it will be assessed against these goals.

CHAPTER 4: SELECTION OF A 3D MODELING PROGRAM

This chapter discusses the challenges and considerations that were involved in the selection of an appropriate modeling program to be used as the basis for this research. It includes a discussion of other similar research in the nuclear field. It concludes with an initial demonstration and verification of the selected modeling program.

4.1 Current radiation visualization solutions

Literature research into other radiation visualization work in the nuclear field was conducted; it returned several different types of software related to this work of which the most relevant will be discussed.

4.1.1 VR Dose

The most similar product that was evaluated is called VR dose, created and maintained by the ***Institutt for energiteknikk Haldon Reactor Project (Norway)***. This program can be superficially thought of as a pre-existing solution to the visualization challenge of this work. After experimentation with the demonstration version (VR Dose demo version 1.0.1) and a visual analysis of the imagery produced by the software package [13] it became clear that this does not function in the desired capacity as an open visualization tool for radiation fields for the scientific community. It is designed to function as a planning and training tool for decommissioning activities [14].

Figure 4 and Figure 5 are screen captures from the VR Dose demonstration program. The human representations are the avatars which a user controls to move around the environment, perform actions and receive a calculated dose consequence because of those actions. As a training tool for demonstrating the radiation dose consequences an action and engaging users to be aware of those consequences, VR Dose is a novel and unique piece of software [13]. For the purposes of providing a universal tool any engineer, scientist of health physicist can use for visualizations of radiation fields, VR Dose does not provide that capacity due to both the costs associated with the program and the closed nature of its design.

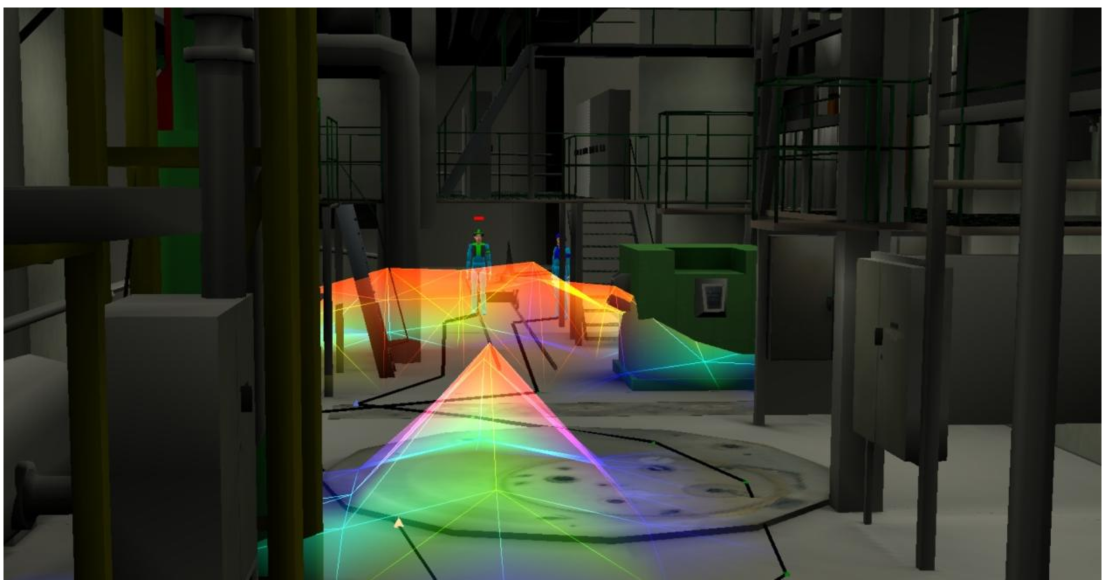


Figure 4 - Screen capture from VR Dose program showing nuclear facility environment

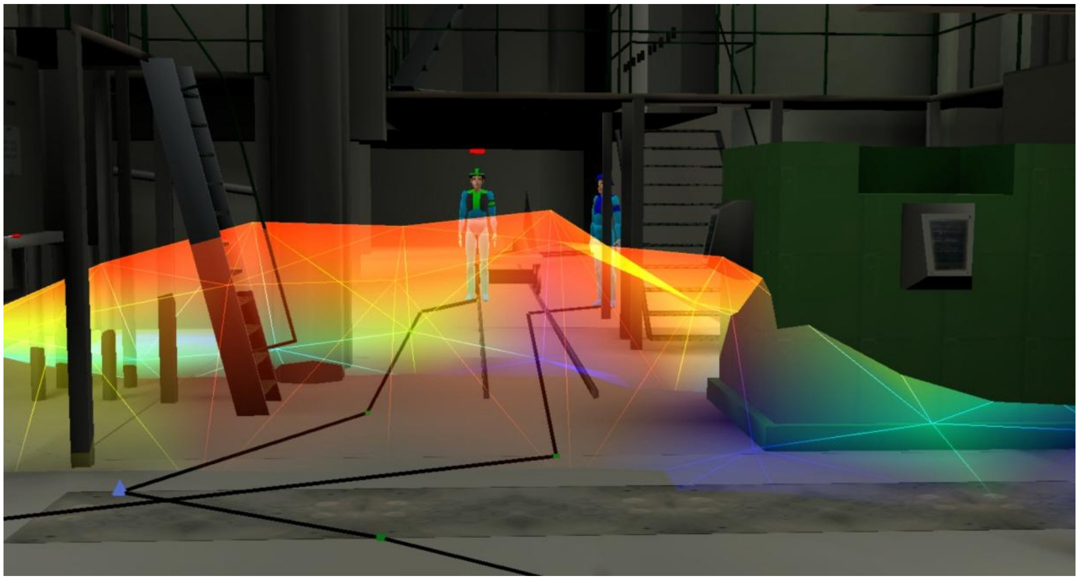


Figure 5 - Screen capture from VR Dose focusing on field shape

4.1.2 Rad Paint

A radiation visualization software package was developed at the University of Florida and is called Rad Paint [4]. Rad Paint visualizes radiation by using or 'painting', surface textures onto 3d shapes. A complex coloring of those textures is used to reflect the various radiation levels acting on that surface.

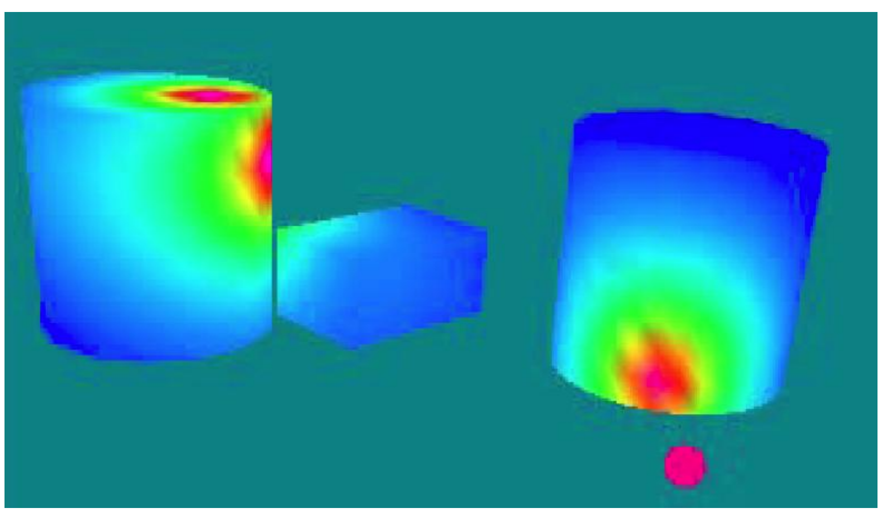


Figure 6 - Image of radiation fields from Rad Paint

Figure 6 is an image from Rad Paint program demonstrating the surface textured approach for visualizing radiation fields. This approach, although visually effective, leads to the internal details being completely hidden from the observer. This limits the potential usefulness of the program as any combination of an existing 3d model with a Rad Paint generated radiation texture will lead to significant obstruction of any internal data by the field surface textures.

4.1.3 DESIRE RadVis

The Dose Estimation by Simulation of the International Space Station (ISS) Radiation Environment (DESIRE) project was created to calculate the radiation levels on the Columbus ISS module and from those estimate the doses received by the astronauts aboard the ISS [15]. The DESIRE RadVis project was designed to take those results and visualize the dose rate and flux data in a manner easily comprehendible by scientists, medical doctors and space engineers [16].

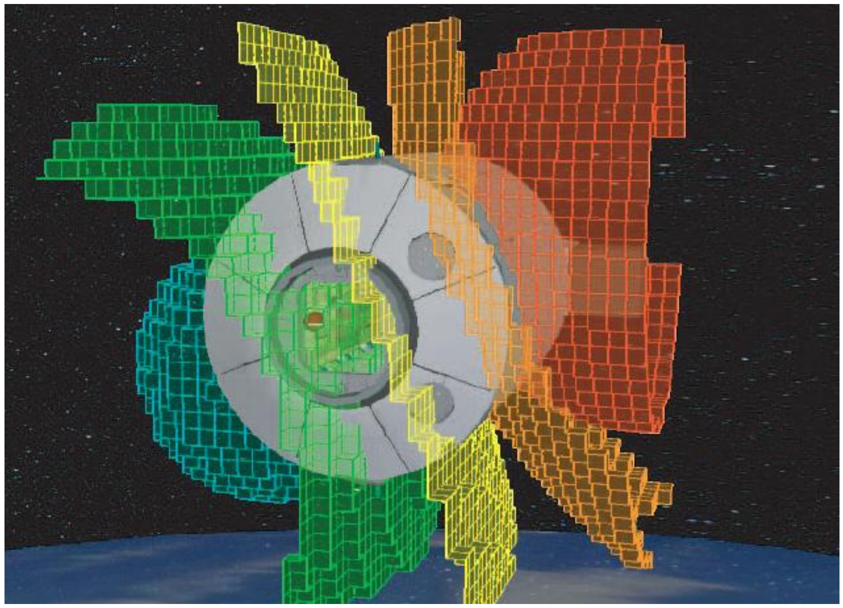


Figure 7 - RadVis image of ISS module and radiation field effect

Figure 7 is an image from a RadVis brochure, the 3d effect shown is accomplished though 3d iso-surface plots producing the very attractive results. RadVis is maintained by the same organization that maintains VR Dose. It is very attractive visually in the same way VR Dose is, but this software is designed for a single purpose (visualizing cosmic radiation) and therefore is not useful outside its intended capacity.

4.2 Off the shelf software versus a custom program

One of the primary decisions to be made was selecting between using Off The Shelf Software (OTSS) or building a custom software visualization engine. This crucial step was very important because it guided the programming challenges and all other parts of the visualization stages.

4.2.1 Off the shelf software

OTSS includes all software presently available to create, view and interact with 3d models for the purposes of creating visualizations. Selecting a 3d modeling software package from the massive variety available is not a straight forward task nor is physically experimenting with every software package achievable within reasonable time constraints. There are simply too many different 3d modeling software packages available that share many common features to differentiate themselves noticeably. To narrow down the selection process, a review of several of the most popular software packages was conducted.

Selection was focused on five mainstream candidates: 3ds Max (Autodesk Inc.) [17], Blender (Blender foundation) [18], Maya (Autodesk Inc.) [19], SketchUp Pro (Google Inc.) [20], and SolidWorks (Dassault Systèmes SolidWorks Corp) [21]. Table 2 briefly details each of the primary advantages and disadvantages associated with each piece of software.

Table 2 - Off the shelf software advantage / disadvantage analysis

Software (Author)	Advantages	Disadvantages
3ds Max	-Professional grade -Used in industry (movie, games) -Capable of animations	-Expensive -Difficult to use/learn
Blender	-Free -Capable of animations -Python scripting language	-Complicated to use/learn
Maya	-Professional grade -Animation	-Expensive

SketchUp	-Free / Professional grade -Ruby scripting language -Compatible with other free tools (such as Google Earth) -Very user friendly	-Cost associated with pro version -Limited functionality of free version
SolidWorks	-Professional grade -Real 'engineering' solution	-Expensive -Difficult to use/learn

All of the software packages are capable of presenting very impressive 3d visuals and therefore the technical capabilities are the least differentiating aspect of each package. This means that other factors such as the cost and user friendliness differences between each package became the biggest factor differentiating each package.

4.2.2 Custom program

A custom program is a label given to a program built *ab initio* for the purposes of displaying the 3d radiation models. A customized program has many advantages over OTSS and many distinct disadvantages. In addition, any consideration of a custom program requires a decision regarding the target computational platform and computer language which further complicates the decision.

Table 3 - Custom program advantages/disadvantages analysis

Advantages of a customized program	Disadvantages of a customized program
-Single purpose leads to a focused applications done very well -Relatively low cost (or free if selected language is open source) -Program can be custom tailored for radiation field display possibly leading to more effective and robust model display output	-Added high level of complexity -Need to make program available for others to reproduce efforts -Many programming languages available which require a second level of decision -Assumed much greater development time required for software due to increase in complexity

Table 3 lists the key advantages and disadvantages of a custom software program. The two most important factors of all of the listed advantages and disadvantages is the increased complexity and the greater development time required. The complexity is associated with learning a new programming language and learning how to create graphics and render visuals in 3d using that language. The greater development time is naturally associated with this type of work because not only is the generic methodology required, but a complete 3d rendering computer program is as well. This in effect doubles the effort required without providing a demonstrated justification for this level of effort. This was previously demonstrated within the Rad Paint program discussed in Section 4.1.2 and therefore building a program from scratch was not a novel approach and this research did not continue assessing that option.

4.2.3 Design decision

As discussed above it was immediately clear that the development of a custom program was beyond the scope of this work and would not meet the goals of this research therefore that option was dismissed. This meant that a decision between one of the five OTSS packages was required. As discussed previously, because each software package had such similar technical capabilities it was not possible to separate them on a technical level. Instead three points were selected: Costs, user friendliness, and open source file storage format capabilities. Google SketchUp and Blender were the top contenders because they were both free and compatible with open source file formats.

User friendliness is a difficult to assess characteristic because what one user considers friendly another may consider confusing. From a technical standpoint the

OTSS packages were considered equal (that is each software package was considered to have the potential to display radiation field models), and therefore the importance of the user friendliness of the program became the deciding factor. Because user friendliness is such an individually assessed trait a decision matrix table style of decision is simply not possible and another approach was needed.

Some experimentation took place with both programs. This experimentation was explorative in nature, designed to examine the user interface and the controls. The purpose was to establish how much difficulty a novice user would have trying to use either package. After several days of experimentation with both programs a decision was made to use Google SketchUp instead of Blender because its interface was more intuitive and overall it felt easier to use. This would ideally allow new users who try to emulate this research an easier experience if they are unfamiliar with 3d modeling software. The three key features of SketchUp can be summarized as:

1. Very intuitive interface based on experimentation
2. Free version of the software with very few restrictions
3. Scripting language with a wide assortment of documentation and training manuals

4.3 SketchUp 7

Google SketchUp 7 version 7.1.6860 (here after called SketchUp) is a computer assisted design (CAD) software tool distributed by Google Inc. [20]. It has the benefit of not only being a very commonly used tool in industry (architectural design, construction and engineering [22]) but it is also made freely available with some minor features removed [23-24]. The two most prominent features of SketchUp that were identified very early on in its consideration were the availability of a free version of the software, and the inclusion of the Ruby language inside the program to automate construction actions [25-26].

Ruby is a computer programming language designed to be open-source, conceptually easy to understand and simple to use [27]. Its inclusion within SketchUp as a scripting language allows most construction commands to be automated. This was considered appropriate for automation of the generic methodology. Other approaches to automate construction actions could have been adopted, such as using external macros run by an outside program, but Ruby's inclusion in simplifies this process [25] so that other approaches to automate construction actions are not needed and would likely reduce usability.

To verify that SketchUp would be capable of modeling the desired effects, some conceptual images and models were created. These images, although basic in nature, became the ultimate foundation for the development of the methodology. These images represent the author's initial concept for how a 'point' source of radiation would appear in a 3d model.

Figure 8 illustrates a simplistic representation of how a point source would appear. The key feature is an ever decreasing value (be it flux or dose rate) as distance from the source increases. In these initial conceptual images, the radiation fields are simply geometric primitives with a transparency filter applied. There were no calculations that took place to create these field models, rather they were experimental in nature examining the visual capabilities of the SketchUp software. The color values used in this point source model are not intended to relate to any specific values, instead they are to illustrate varying values within a field and SketchUp's ability to display that information.

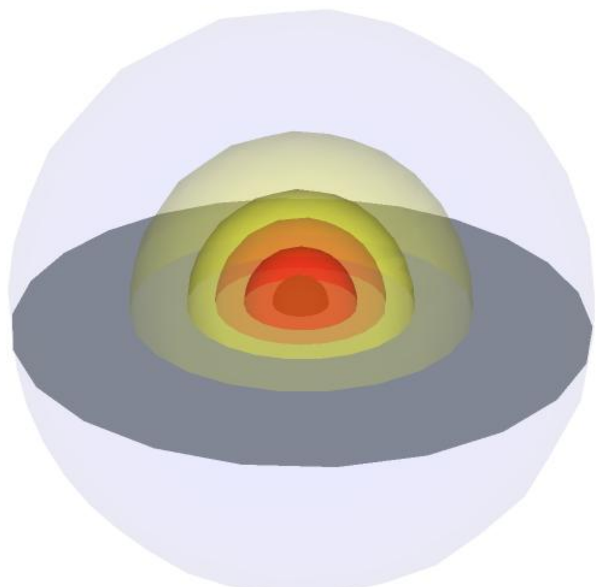


Figure 8 – Point source concept model made in SketchUp

The point source concept model was placed into two prebuilt Sketchup environments to examine how it would appear both in an indoor and an outdoor environement. Figure 9 represents the point source model from Figure 8 sized within an outdoor environment to represent increasing radiation readings coming from a source hidden in a hedge.

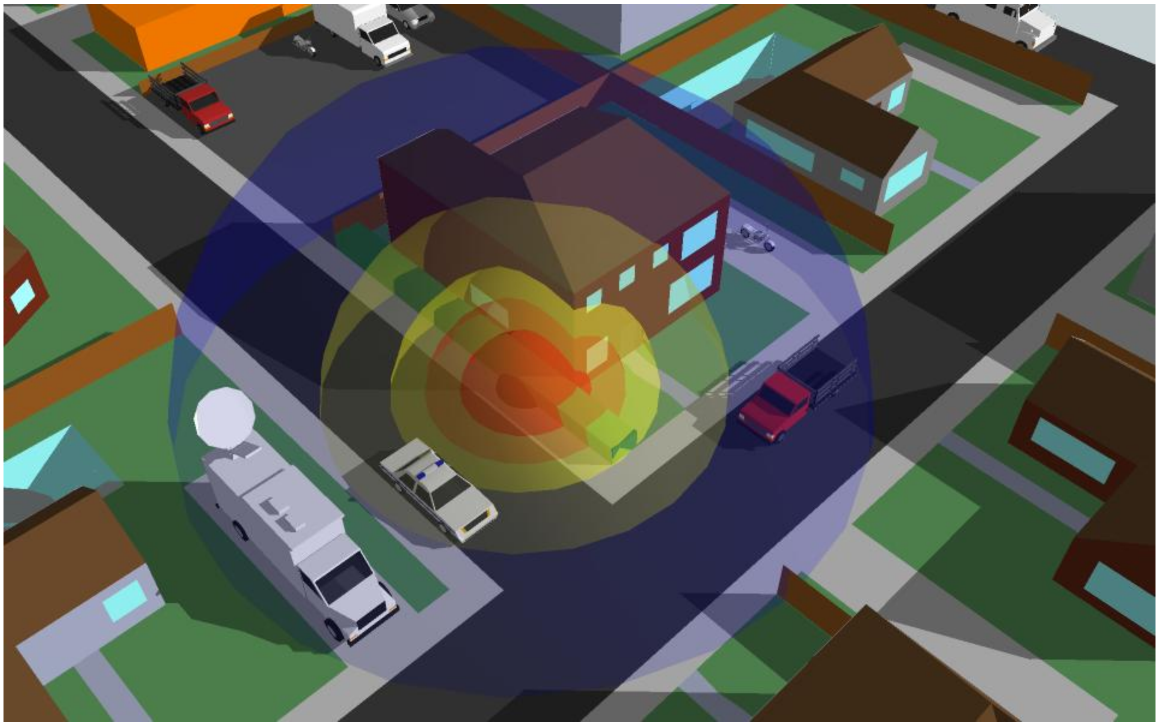


Figure 9 - Point source inserted into outdoor environment

Figure 10 represents the point source model scaled down in size and positioned on a desk within a small office. In both cases effects such as sheilding are not visually rendered or accounted for. This experiment confirmed, from a visual standpoint, that SketchUp had the potential to create an image to visualize relative changes in radiation levels in an indoor environment.

Figure 10 - Point source inserted in an indoor environment

The results were promising from a visual standpoint. This demonstration successfully showed the radiation field concept in both indoor environments and outdoor environments is possible within SketchUp. This was considered a sufficient demonstration of the visual capability of the SketchUp software. Remaining to be developed was a generic methodology to build a radiation field model that accurately represented a 'real' radiation field.

4.4 Risk

The biggest risk with the selection of SketchUp was related to the free availability of the program and the uncertainty of that being maintained in the future. One of the most important aspects for justifying the use of SketchUp was that it was made freely available on the internet with the majority of features present in the free version. To a certain extent the risk of SketchUp disappearing can be alleviated by an end user saving a current copy of the program installation executable and installing as required in the future, there are legal obstacles to overcome if there was ever a need to mass distribute that in the future.

The primary mechanism to control this risk is by documenting exactly how this methodology works, step by step in this thesis so that it can be adjusted and transferred to another visualization program if ever required. This is the reasoning for the founding principle that the entire process for generating these visualizations be as generic as possible to maximize future compatibility. Being generic in nature and open in documentation will allow others the opportunity to take the work in this research and apply it elsewhere. This was considered an appropriate mitigation strategy for this risk.

4.5 SketchUp user interface and controls

This section provides a very brief explanation of the interface and features of SketchUp. This is in no way an authoritative guide on how to use the program. This section discusses key features to be used in this work.

4.5.1 Viewpoint movement

Movement in SketchUp is relatively straight forward. The user viewpoint orbits a model at a distance. A user can zoom in and rotate their viewpoint to establish other views of a model. Unique tools such as the ‘walk’ and ‘look’ tools allow the user to give the impression that they are walking within a model and looking through the eyes of a user controlled avatar. These are very useful for presentations and demonstrations. Figure 11 identifies where these options are in the SketchUp interface.

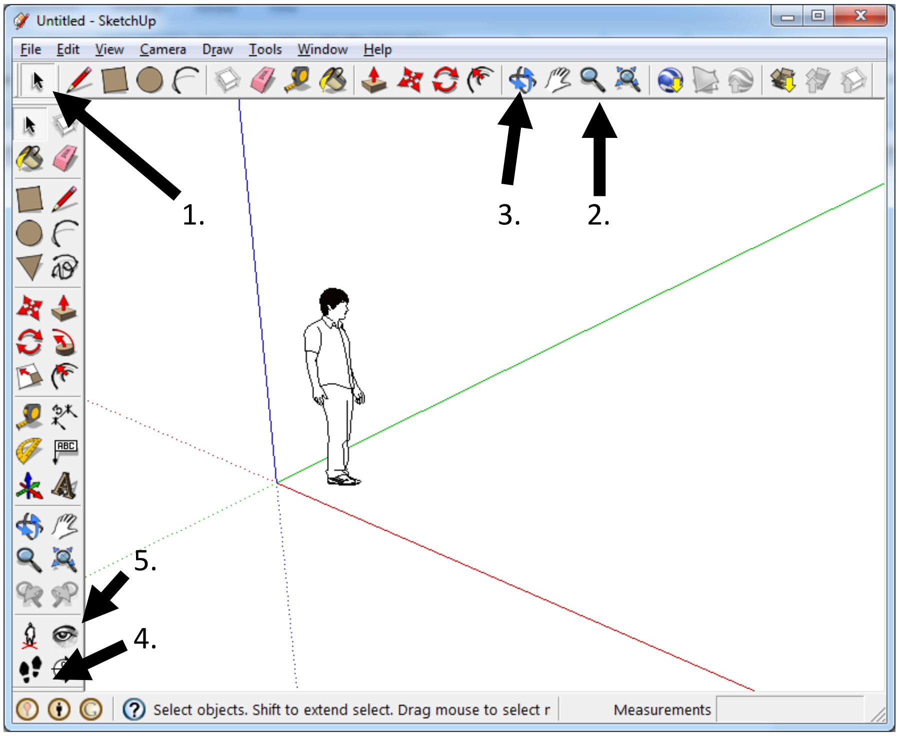


Figure 11 - SketchUp basic viewpoint movement controls

1. This allows the selection of surfaces, objects or other parts of drawings
2. Zoom in and out of current view point
3. Rotate current viewpoint
4. 'Walk' around a model, useful for demonstrating navigation on foot to a live audience
5. 'Look' around a model, used in conjunction with the 'walk' feature

4.5.2 Constructing a surface

Surfaces in SketchUp are created by drawing a shape on an axis, plane or other surface. As long as there are enough sides (minimum 3) to completely envelope a shape and as long as they fall within a common plane, a 2d surface will be created. To create a three dimensional shape, 2d surfaces can be 'pulled' outwards with SketchUp automatically adding in the sides for the other dimension. This is demonstrated as a three step process in Figure 12.

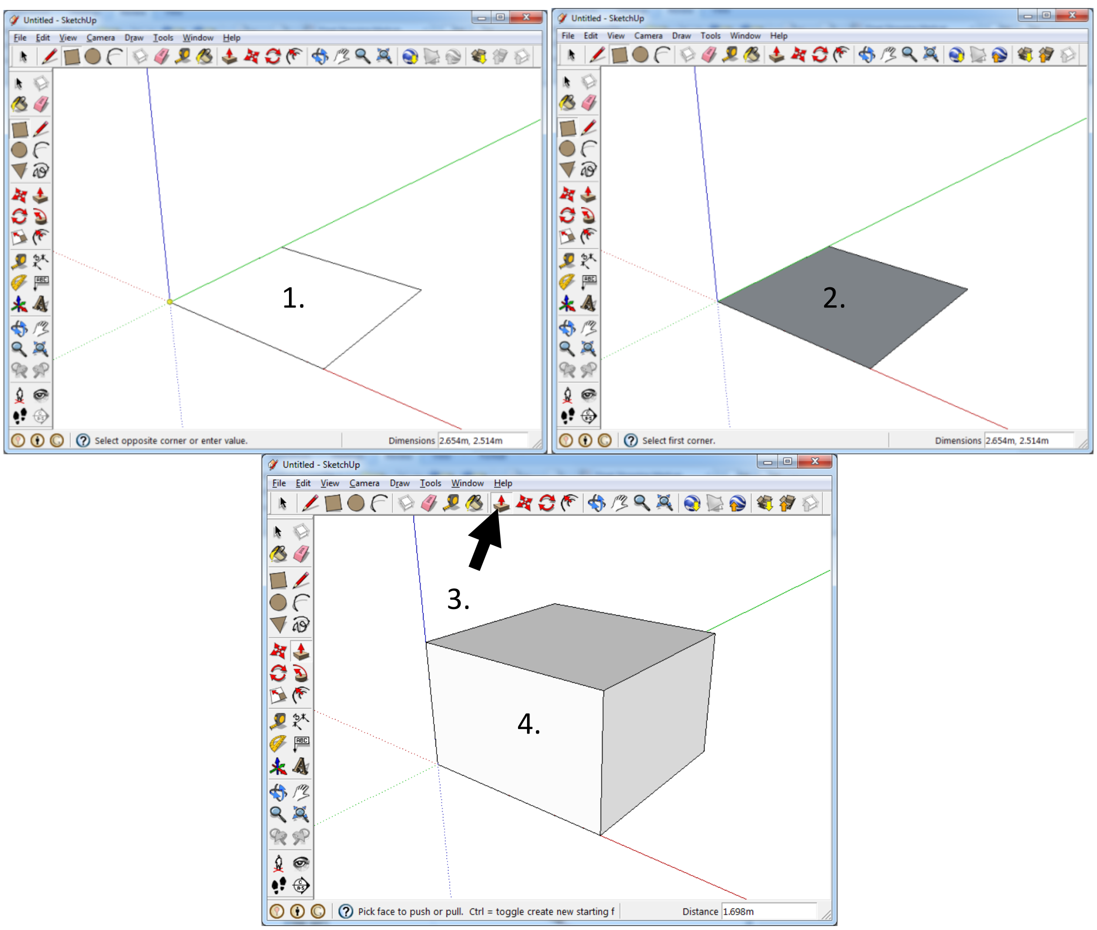


Figure 12 - Three step process to create a block in SketchUp

In step 1 in Figure 12 a square shape is being created. When all four sides have been drawn the screen will look as it does in step 2 in Figure 12. At this point the 'push/pull' action (highlighted as point 3.), is selected and the surface can be pulled upwards as in 4 in Figure 12. This in essence is how shapes and objects are drawn in SketchUp.

Other features that are required in this research will be explained as they come up. References [26] and [28-30] provide a good basis for using many of the additional features in SketchUp not mentioned here.

CHAPTER 5: CREATING A 3D RADIATION FIELD MODEL

This chapter details the development of the methodology to create a 3d radiation model. The purpose is to introduce the methodology in very general terms. For specific step by step instructions for how to do this within SketchUp refer to Chapter 6 which follows this process through from start to finish.

5.1 Block based methodology

The chosen construction methodology is conceptually straight forward to understand. A field is broken down into a set of finite elements with each element containing a series of bounds (limiting its extent in all directions), an intensity value (representing its reading) and a central coordinate location (which is in direct relation to all other elements). Each element is considered to act as a single representation of an intensity value for a field within the local confines of that element. These elements can be geometrically simple shapes such as cubes or boxes. More complex shapes are possible, but they needlessly increase complexity, whereas boxes are intuitive to work with.

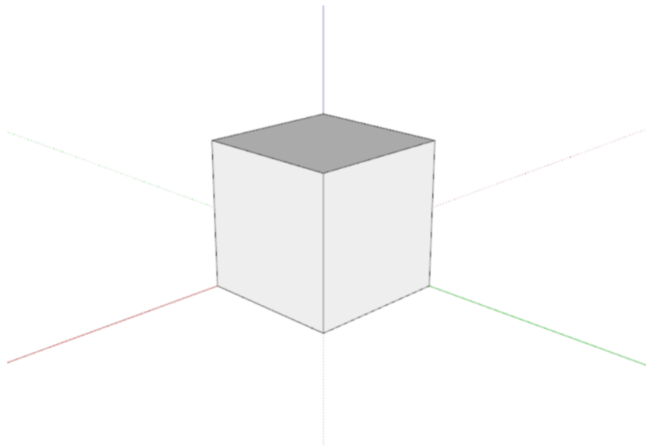


Figure 13 - Ideal single elemental cube constructed in SketchUp

Figure 13 shows an example of an element. These individual elements can be thought of as a physical representation of a volumetric pixel (voxel) [31]. Voxels can be used to represent data in a dimensional space as they contain both a physical location and a value at that location.

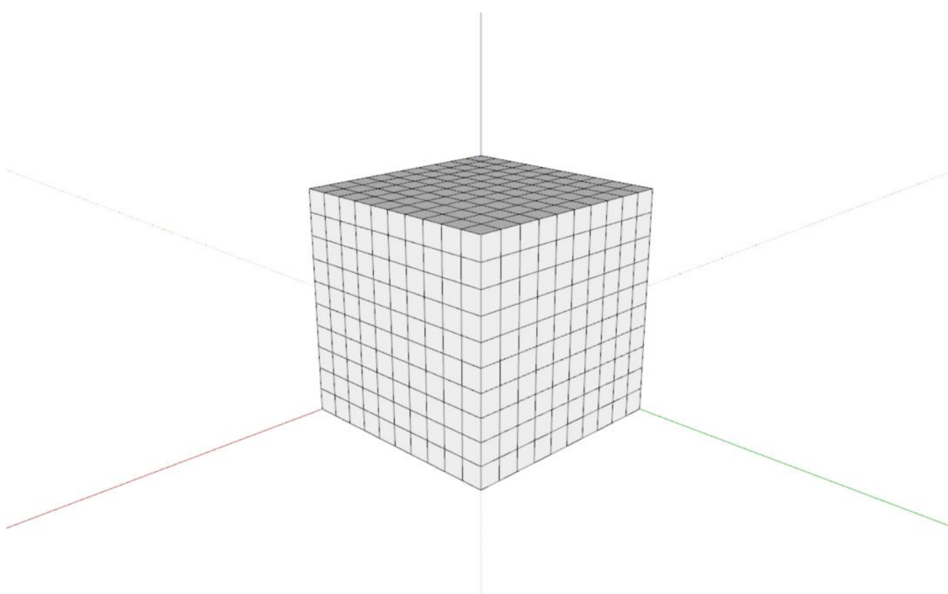


Figure 14 - Array of 10x10x10 elements to form the basis for a simple field model

Figure 14 shows these elements arranged in a 10x10x10 matrix. Using basic shapes simplifies the arrangement of these elements into a single model where all the elements can be fitted together so their boundaries do not overlap each other. Theoretically an intensity value is not limited to a single type of information (e.g., dose rate); any type of information could be visualized using this type of methodology.

5.2 The construction process

To illustrate the construction process, a simple example will be discussed. Consider a point source of radiation contained within a 7 m x 7 m x 7 m room. This source is exactly centered on all axes and can be thought of as floating in the center of the room. Consider an array of values in a 7x7x7 matrix. The value in each location in the matrix is a descriptive value related to the radiation field intensity within a 1 meter cube located in that area (this could be related to the average value within the cube, the highest value within the cube or simply the value at the midpoint within the cube). The values assigned to this matrix for this exercise are shown in Table 4 as viewed from top down slices (S), from 1 to 7.

Table 4 - 7x7x7 Matrix values

S1 =

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

S2 =

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

S3 =

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	2	1	0	0
0	0	0	1	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

S4 =

0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	0	2	0	0	0
0	1	2	3	2	1	0
0	0	0	2	0	0	0
0	0	0	1	0	0	0
0	0	0	0	0	0	0

S5 =

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	2	1	0	0
0	0	0	1	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

S6 =

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

S7 =

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

To build a model out of this information, four types of elements are needed to represent intensity levels of 1, 2, 3, and 4. They are shown in Figure 15.

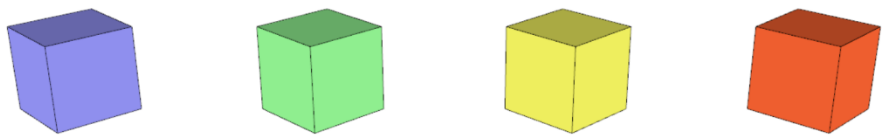


Figure 15 - Elements types 1, 2, 3 and 4 (from left to right)

Using these four elements and the locations from the matrix the following construction can take place within SketchUp as demonstrated in Figure 16.

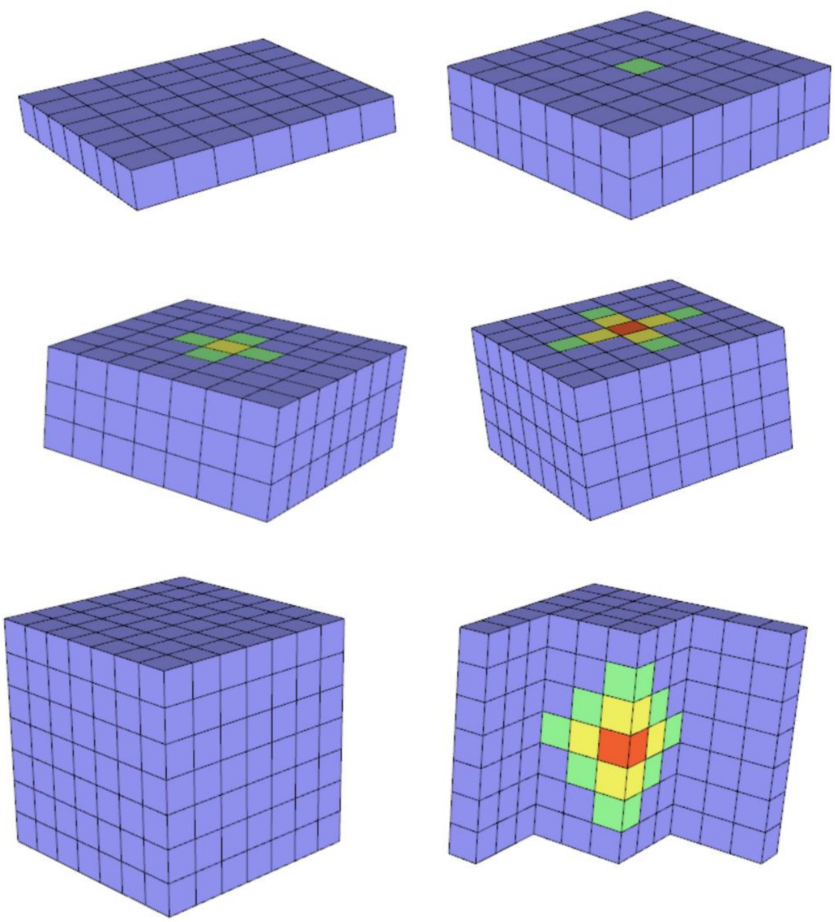


Figure 16 - Construction of the 7x7x7 field model sequence

Several layers were removed from the final image in the sequence to reveal the interior details. This process (breaking down a field into a collection of finite elements), although very time consuming to do by hand, was found to be very effective for the construction of these types of models. Consider also, that the order of construction need not start from the bottom up, rather the position of each element simply needs to be maintained via its coordinate information.

5.3 Defining a scale

The color values assigned to represent each block in the final model require a significant degree of thought when they are defined. For the model building process discussed in Section 6.3, there are two primary types of differentiations between each type of element possible: color values and transparency effects.

5.3.1 Color values

To distinguish the difference between cells of different values (within specific ranges) colors are the most direct and logical option. Intuitive color scales usually follow one of two patterns, blue to green to yellow to red variety, or the varying shades of a single color from light to a very intense dark peak [7]. An example of each is shown in Figure 17 and Figure 18.

Figure 17 - Color scale example 1

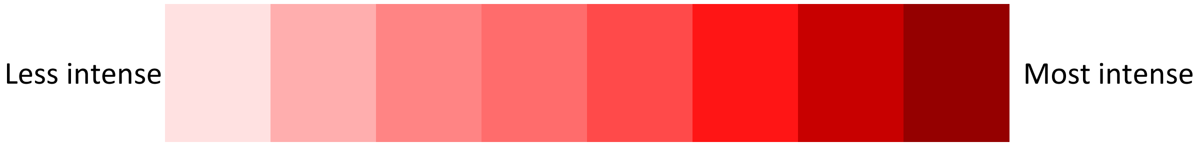


Figure 18 - Color scale example 2

Both of these scaling methods should be considered when constructing a model. However the intended application will drive the color selection in the majority of cases. For example, when designing a model for educational purposes a green to yellow to red scale may be the most intuitive for the user to understand (Green being positive and red being negative) but if misused it may draw undue concern about inconsequential aspects of a model.

During model construction each color value becomes a separate 'material' within SketchUp. Afterwards these materials can be adjusted as required, allowing colors to be changed post-construction.

5.3.2 Transparency effects

For details beyond the exterior surface of a model to be visible there is a requirement for some type of transparency effect as discussed in Section 3.1.1. Transparency effects in SketchUp involve editing a specific material and reducing the opacity property (initially set at 100). The opacity property can be anywhere between 100 and 0, where 0 is completely transparent. Section 6.4 gives step by step instructions on how to do this within SketchUp.

There is no perfect approach for determining what opacity values to assign. This is very dependent on the purpose of the model, the viewer's preference and the total amount of elements to be ‘looked’ through. Typically values assigned to the least important materials will be between 1 and 10. Values assigned to the most important materials generally approach 30. Depending on the size of the model this may change, as each ‘material’ one looks through reduces the total opacity of the pathway to the model which may or may not be the desired effect. In Figure 19 transparency has been added to the model constructed in Figure 16.

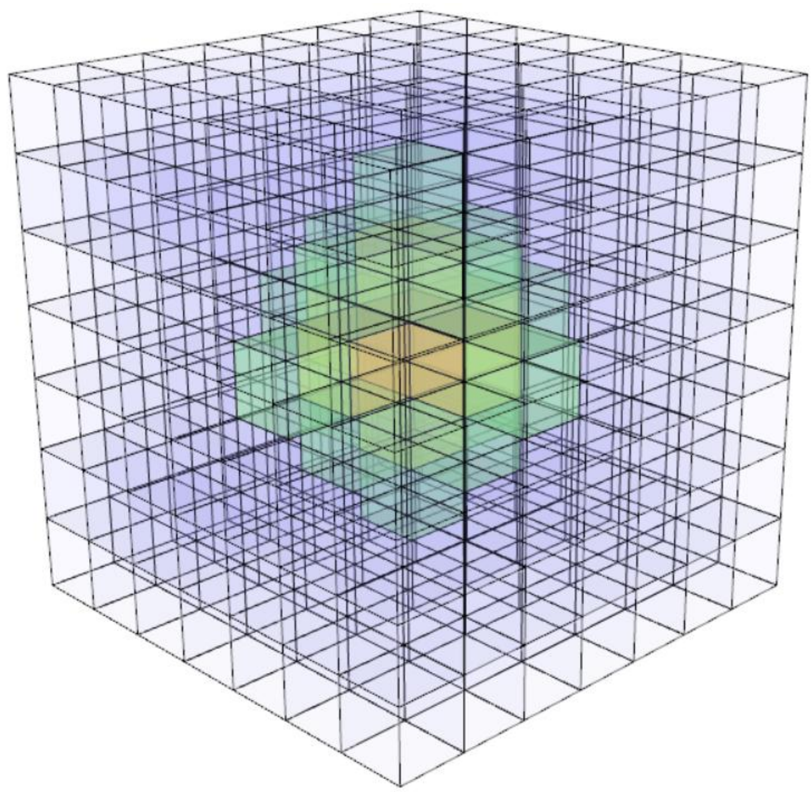


Figure 19 - Transparency applied to model from Section 5.2 (edges shown)

Note that in Figure 19 the edges for each element comprising the model are very evident and distract from the overall radiation field effect that is to be achieved. This is remedied by turning off the option in SketchUp to render edges. Figure 20 shows where the dropdown menu option is to remove edges and Figure 21 shows this applied to the model in Figure 19.

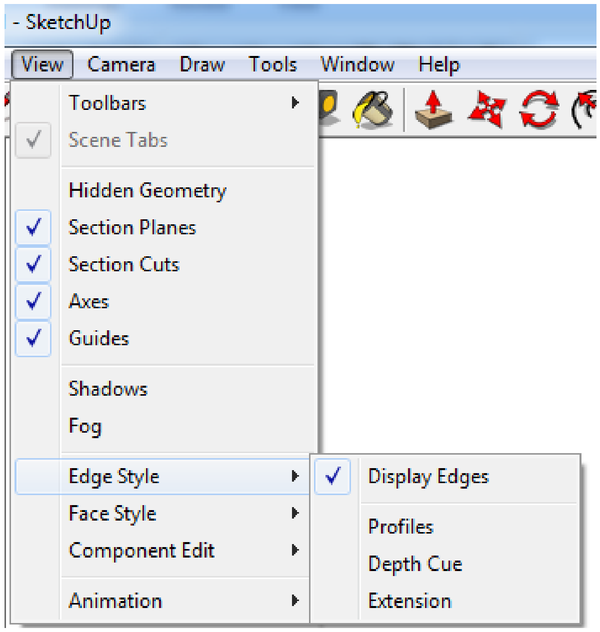


Figure 20 - Display edges menu

This produces the desired effect seen within Figure 21.

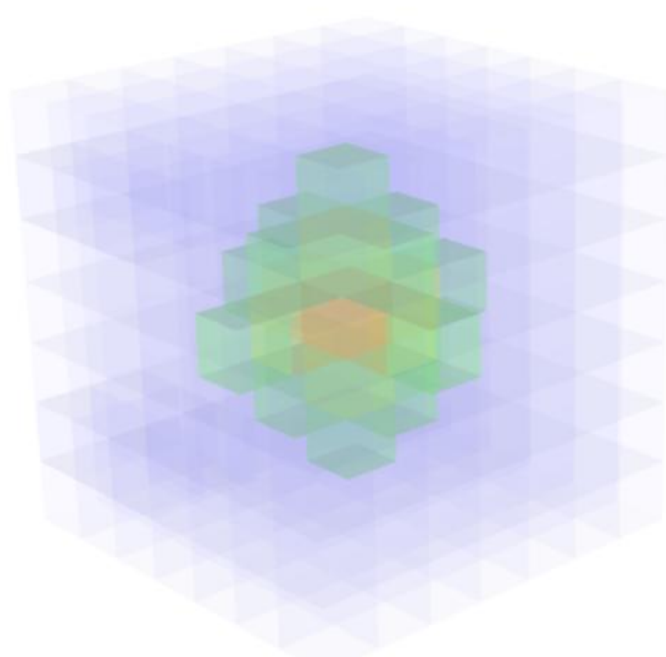


Figure 21 - Transparency applied to model from Section 5.2 (no edges shown)

As more elements with transparency are ‘looked’ through, a combined opacity factor adds up. If there are too many materials to be seen-through then the model ceases to become transparent. This is remedied by reducing the opacity of all associated materials. Alternatively, a user can simply accept this visual effect, as it is generally not very distracting.

If there are areas of interest in a particular model, such as only the highest levels of radiation beyond a threshold value, using transparency to hide the undesired elements is an easy method to highlight specific areas. In Figure 22 the outer layer has been made completely transparent to illustrate this effect.

Figure 22 - Image from Figure 21 with one material made completely transparent

The above are the basics of the generic methodology proposed in this research. Chapter 7 will demonstrate why and how this methodology can be used effectively for visualizing radiation fields.

5.4 Methodology summary

The method proposed for building and modeling a radiation field can be summarized as the following actions:

1. A data set containing (x_n, y_n, z_n, V_n) is taken (where x_n, y_n and z_n represent coordinates, and V_n represents a ‘value’ at those coordinates)
   a. The process requires that x_n, y_n, z_n, values be at a fixed distance apart to establish a fundamental element size for that model (e.g., if they are all values at a 1m, 2m, 3m, etc in all directions, this process will establish that each (x_{n+1}) is equal to (x_n+1m) and the fundamental element size is a cube of 1mx1mx1m)
2. A script is prepared for the 3d modeling program being used. It reads those values one at a time and constructs an element at each location. This script includes a scale where the V value is assessed and each element is coloured based on its value.
   a. The program is opened and the script is run to calculate each (x_n, y_n, z_n, V_n) and a shape is built (centered at the coordinate or other reference point)
   b. Based on the V_n value, that new object (volume) is given a colour, material, or whatever the term the program uses to define the appearance of an element

3. This process then repeats until a shape has been built at all of the locations specified in the data file.
    a. During the construction process different ranges of associations can be assigned to values of V. I.e.: if V is: 5> V > 3, then color = light blue which means any time a shape is built, and the V value is less than 5 but greater than 3, a color value of 'light blue' will be assigned
    b. All entities that fall within a range will share that color, or material property. They require transparency to be added to complete the visual effect.

CHAPTER 6: CONSTRUCTING A 3D RADIATION FIELD MODEL IN SKETCHUP

The general methodology for visualizing radiation fields has been developed in Chapter 5. This chapter discusses instructions to reproduce the effects within SketchUp. It is important to realize that the methodology discussed in this chapter is not applicable solely to SketchUp; rather it could be adapted to any 3d modeling program where the construction process can be automated.

6.1 3d model definition

The first step to building a model is to define the field to be used in that model. To be compatible with the Ruby code developed in this thesis (available in Annex A) a text file is required which contains values in the form shown in Table 5 and Table 6.

Table 5 - Data format for ruby script

(x-coordinate) (y-coordinate) (z-coordinate) (intensity-value)

Table 6 - Example data taken from text file

5	5	6	4.10
6	4	6	3.80
7	4	7	3.63
8	8	4	3.33
8	4	10	3.03
9	1	3	2.77

Each element to be constructed requires values as shown in Table 5 and Table 6. This is stored in a plain text file (.txt extension) which uses a separate line for each new element and tab delineation between values. The ordering of the rows does not matter as each element is built separately. To define the field and calculate the values needed for the text file, two different techniques are recommended: direct mathematics or computational software. Alternately, direct physical measurements could be used.

These values may originate from a spreadsheet software package so it is important to realize that the intensity values can be adjusted to make them easier to manipulate when they are in the spreadsheet. For example, if the values in the spread sheet are very large or very small, multiplication of all of the values by a common normalizing value to move them closer to whole numbers may aid in the steps which follow. In general, the process of developing a scale can be simplified for the user with intuitive whole number intensity values.

6.1.1 Direct mathematical approach

A mathematical definition would come from a formula such as photon intensity from a point source, line source, etc. Any mathematical formula that will give an intensity value at a coordinate x, y, z will be possible to model. Spreadsheet software can be used to save time.

One important factor to consider when using a mathematical formula is what intensity value to use inside each element. As the formula will likely only give intensity at a point x, y, z and the modeling will actually represent intensity within a volume, a decision concerning what is the most appropriate value to represent is required.

Possible options include:

- Highest value
- Value at the midpoint in an element
- Average of several values within an element

6.1.2 Computational software approach

A computational field model is one associated with a nuclear physics code such as the Monte Carlo N-Particle (MCNP) transport code [32]. If a nuclear physics code is able to keep a tally in multiple locations simultaneously, and provide output data in a format that is similar to (or could be made similar to via post processing) that in Table 6, then the output from that code is compatible with the generic construction process. Nuclear physics codes such as MCNP can provide a very flexible outlet for making graphical representations of more complicated scenes as will be shown in Section 7.2.

6.1.3 Physical measurements

Direct measurement readings offer a potential future avenue for generating the data required to build a model. They are limited in that they require a method to associate each reading with a coordinate in the simulated world. This requires either the pairing of a piece of positioning equipment (such as a Global Positioning System device, a series of laser range finders, etc) with a radiation detector during a room or field survey (either by hand or robotically). Alternatively direct measurement readings could be manually coordinated by the operator during their survey, or a room with a known field shape but uncertain strength could be calibrated to be the correct field by specific measurements at one or two key points which are used to calibrate the remaining points in the model.

6.1.4 Model resolution and extents

When model definition takes place, methodical consideration of the extent of modeling is required, as discussed in 3.1.5. This can potentially be the most important factor to be considered when constructing any visualization. Consider the modeling of a point source within a cube shaped room. The source is suspended in the center. Three different models will be constructed of 5x5x5, 11x11x11, and 22x22x22 blocks. Each model encompasses the entire room from wall to wall and the elements are appropriately sized for each model. The models are shown in Figure 23, Figure 24 and Figure 25.

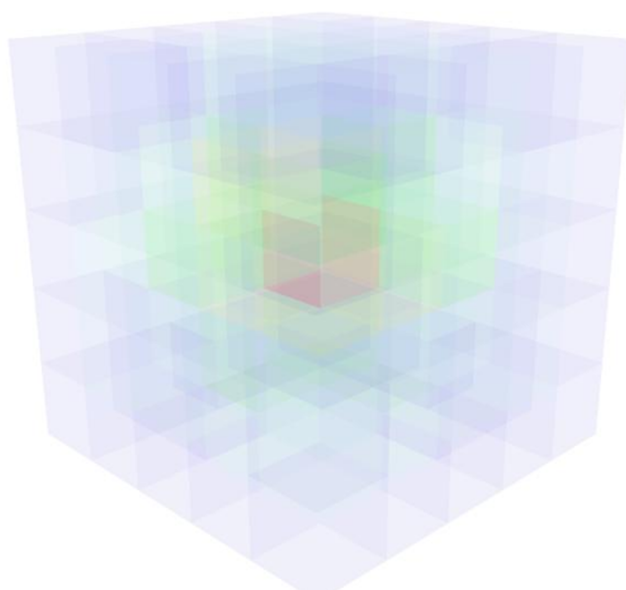


Figure 23 - 5x5x5 room model

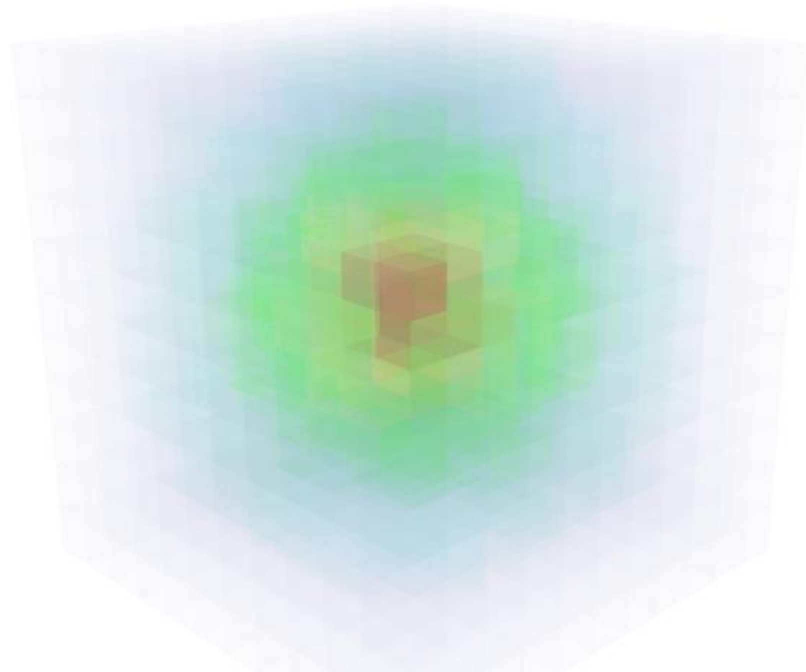


Figure 24 - 11x11x11 room model

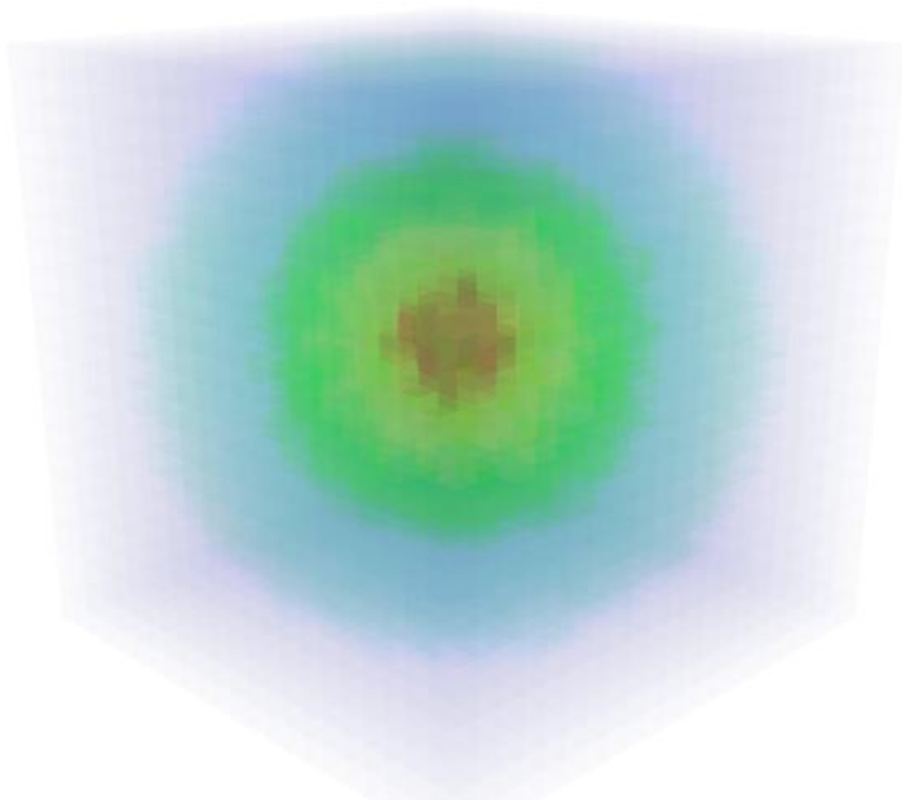


Figure 25 - 22x22x22 room model

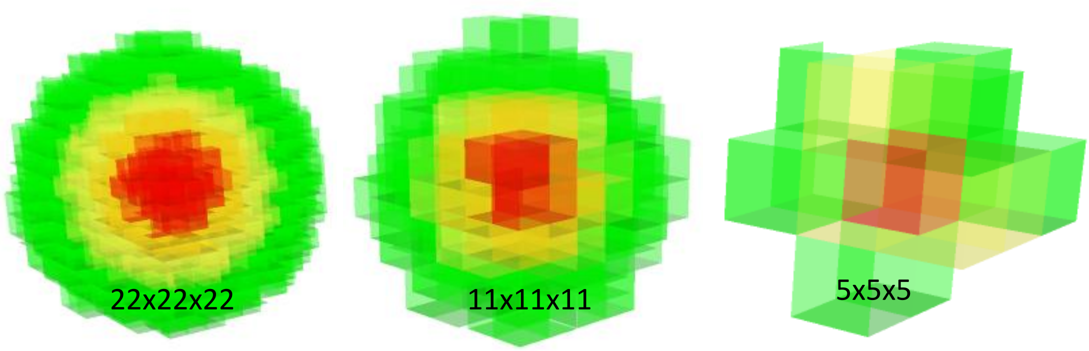


Figure 26 - Side by side comparison of models with outside layers removed and transparency reduced

In the previous figures, note the differences in the clarity of the spherical shape of the field within the room. Knowing that radiation emitted from a point source follows an inverse square relationship, one would expect this field to appear distinctly spherical but without a sufficiently large number of blocks it is difficult if not impossible to see this type of detail. A reduction in the size of the elements (followed by an increase in the total elements required) comes at the expensive of increased construction time and rending overhead. For large models with thousands of elements this overhead may not be worth the small gain in image clarity.

To decrease the number of elements while maintaining an appropriately tight resolution, a model should be limited to only the areas of interest. For example, if any value below a threshold is not needed then it could simply be taken out of the data before model construction. In cases where an entire segment of information is not required (e.g., information surrounding areas which are inaccessible) that information could be removed. This reduces the overall elemental burden on a model without removing the necessary detail. For large models there will always be a tradeoff between the need for detail and the computational power available, which can vary drastically depending on the computer used.

6.2 Data collection, post-processing and scale definition

Once the x, y, z and intensity values have been collected they should be put into a suitable spreadsheet or matrix manipulation program. In the post-processing phase the values should be examined to determine an appropriate scale. The range of values should be studied and a desired scale should be made which highlights the values (or regions) of interest without putting too many divisions within the scale (which will increase overhead as discussed in Section 6.1.4).

Other scale methodologies can be used but they are completely dependent on the intended purpose of the model. If the purpose is simply to view a range larger or smaller than a certain value then a single scale may in fact be more useful. In the development of visualizations intended for communication and education rather than engineering, the color scale will likely have increased flexibility.

If all data has been stored and processed, then the x, y, z and intensity values should be extracted and put into a tab delimited text file. This file should be located in the plug-ins directory with the Ruby script which can be launched from the plug-ins drop down menu within SketchUp.

6.3 Script preparation and model construction

Without delving into the specifics of the Ruby programming language (see references [33-36] for information concerning Ruby), a small degree of preparatory work is required to create the script that will build the model. Within Table 8 in Annex A, an illustrative example of a ruby script developed for use in SketchUp is provided. The script is commented and a novice programmer should be able to adapt and customize this script to suit their needs. This script is required to be put into a text file with an extension '.rb' which is stored in the Plugins SketchUp folder in its root directory (dependent on where it was installed). Once the input script has been created a model can be built by opening SketchUp and launching the script from the Plugins drop down menu. The script is purposefully designed to be simple and easy to manipulate by other users. There are other ways to accomplish the same construction tasks as this script, but this script specifically was selected because of its simplicity.

6.4 Transparency filters and color adjustment

There are no transparency filters applied in the model at the time of construction. Adding transparency to the model is as simple as carrying out the following steps:

1. Open the SketchUp material menu control panel (Figure 27)
2. Select all active materials 'In Model' (Figure 27)
3. Adjust each material opacity value individually (Recommend to move from lowest value to highest value) (Figure 28)
4. Continue adjusting opacity values until the desired visual effect is achieved (Figure 29)

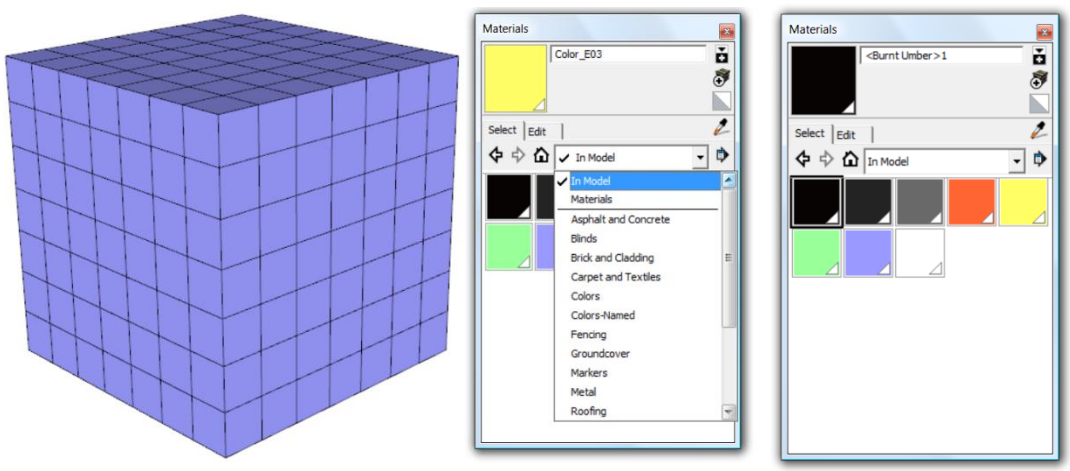


Figure 27 - SketchUp material menu

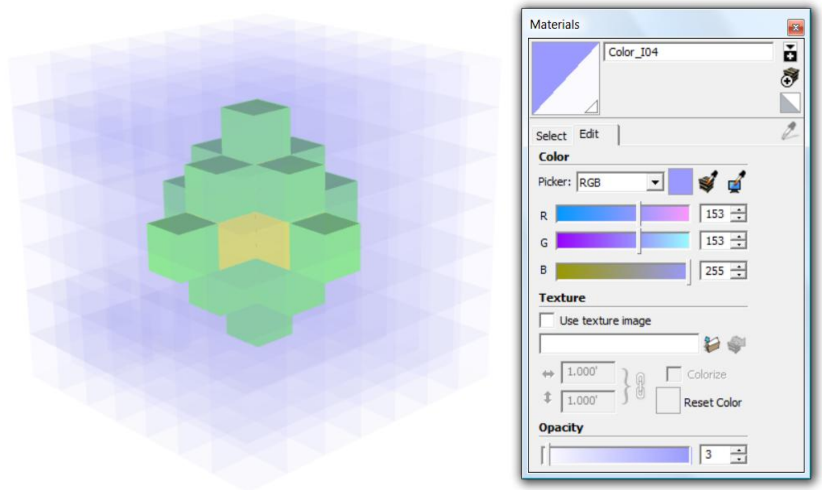


Figure 28 - SketchUp adjusting Opacity

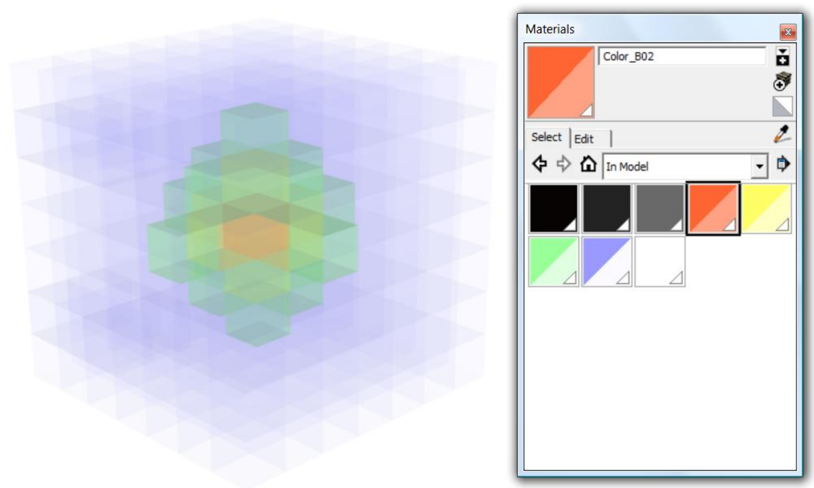


Figure 29 - SketchUp final material menu

The model is complete after these steps have been followed. Materials can be adjusted post construction at any point. Opacity is adjusted under each material in the materials section. Colors can be adjusted in the same menu where the transparency effect is adjusted and adjustments can happen at anytime once a model has been built.

6.5 SketchUp model manipulation

As there are numerous guides on the internet, in print, and within SketchUp itself on manipulating a model this topic is considered outside of the scope of this thesis [26][37]. After it had been constructed, a model can be manipulated to be:

- Combined with other resolution models to increase resolution in areas of interest
- Combined with other models such as physical CAD models of the features in the scene
- Added to digital photography to put the image in context with the real world via the match photo feature [38] (see Chapter 8)
- Exported into other formats
- Manipulated into making video panoramas

6.6 Model geospatial orientation

Geospatial orientation refers to the process of locating a field model at its proper location within another model. To complete a field scene a model may need to be mixed with another model of background scenery (as in Figure 9 and Figure 10). This can be accomplished in one of two ways described in the following two sections.

6.6.1 Manual geospatial orientation

A model can be constructed and 'imported' into another model. To do this requires that there are two separate files (one of the scenery and one of the radiation field model). The following steps can be used to merge two models:

1. The scenery model is opened first
2. The ‘import’ function within the ‘file’ dropdown menu in SketchUp is selected
3. The field model is selected and imported
4. Manually the field model needs to be positioned within its correct ‘space’ in the scenery model (this is most easily accomplished by finding a reference point such as a corner of the field model and placing that at the correct location thereby positioning the rest of the field model in the correct location)

Two key factors need to be taken into account during this process. Firstly, if a model uses a different scale then one of the two models will need to be rescaled within SketchUp until both are the same scale. Secondly, if the field model is angled along the x, y, z axes or some combination of the three differently than the scenery model, one of the two models will need to be aligned at the correct orientation with respect to the other.

6.6.2 Automatic geospatial orientation

An automatic geospatial oriented field model is one that is constructed in its proper place with respect to the scenery when it is being built. This is accomplished by:

1. A scene is constructed with a coordinate system that is compatible with the one used to define the field model (i.e., both models share the same axis location, scales and origin)
2. The spacial coordinates for each element should therefore relate to their coordinates in the scenery model. The code used in Annex A will construct

elements with the data point defining the outer most point on the bottom of each element. If the coordinate represents the midpoint within the element then either the code will need to be changed to reflect that (by adjusting where the element is constructed), or the data entered in the program will need to be adjusted appropriately.

3. When the field model is constructed it will be appropriately placed within the scene.

6.7 Missing information

Thus far there has been no discussion of what would happen if information required for visualization were missing. This effect is best illustrated through an example. The model used in Figure 25 will be used as the basis for this experiment. This model will be randomly reduced to:

1. 75% of the original
2. 50% of the original
3. 33.3% of the original
4. 25% of the original
5. 15% of the original
6. 10% of the original
7. 5% of the original

The reduction will be completely random. As the purpose of this is to examine how random missing data affects the visuals each time a reduction occurs, efforts will be taken to manage the transparency in such a way as to maintain consistency between the models.

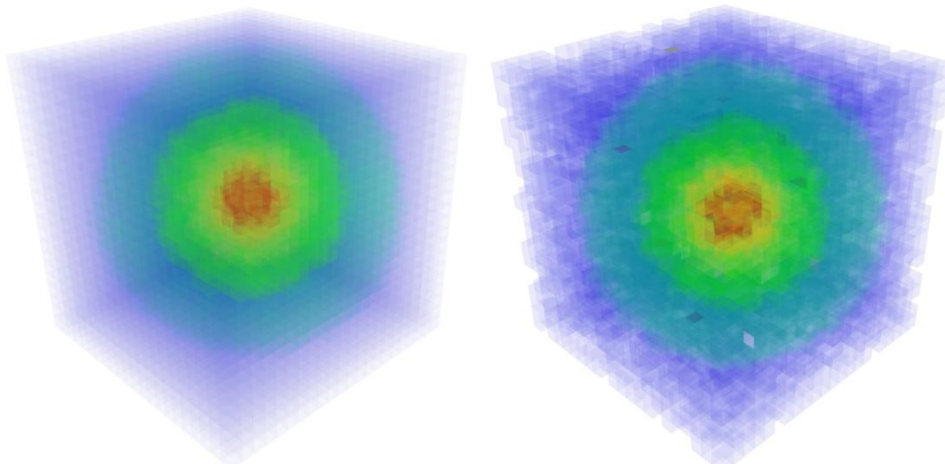


Figure 30 - Complete left and 75% right of original 22x22x22 model

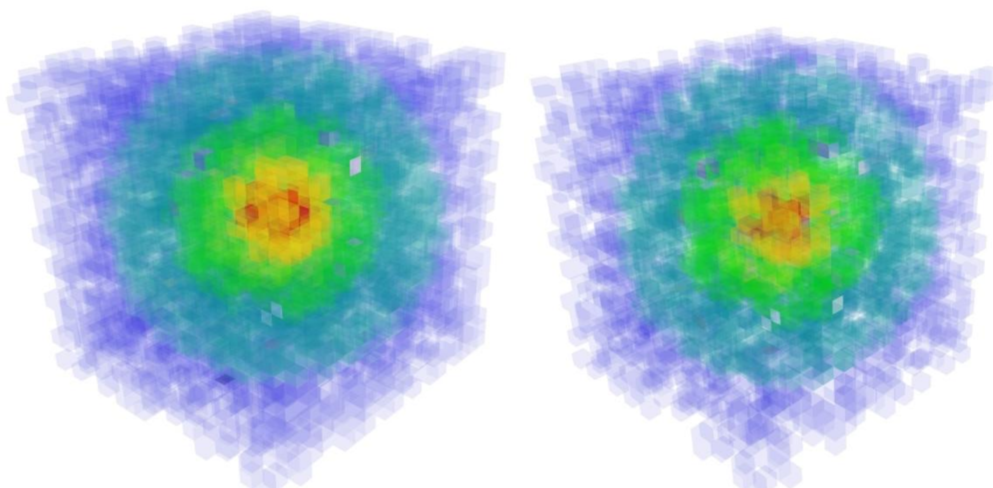


Figure 31 - 50% left and 33.3% right of original 22x22x22 model

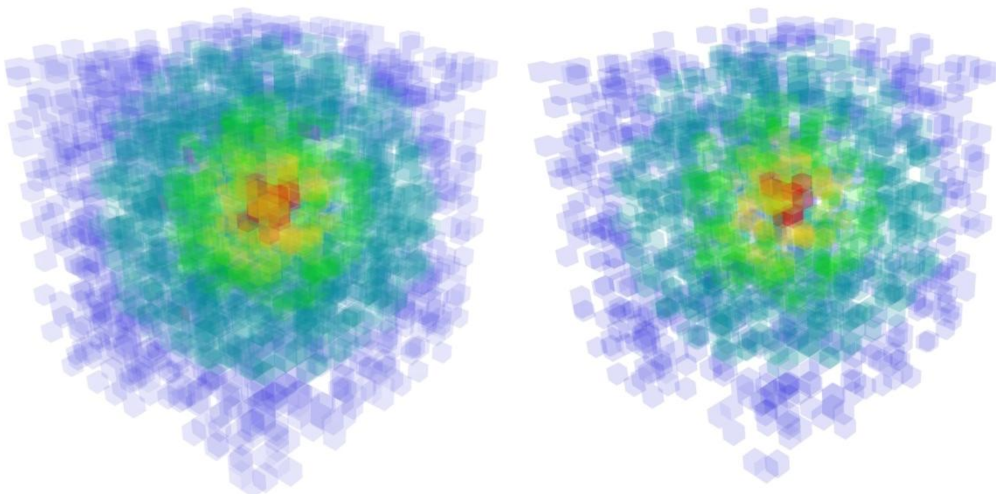


Figure 32 - 25% left and 15% right of original 22x22x22 model

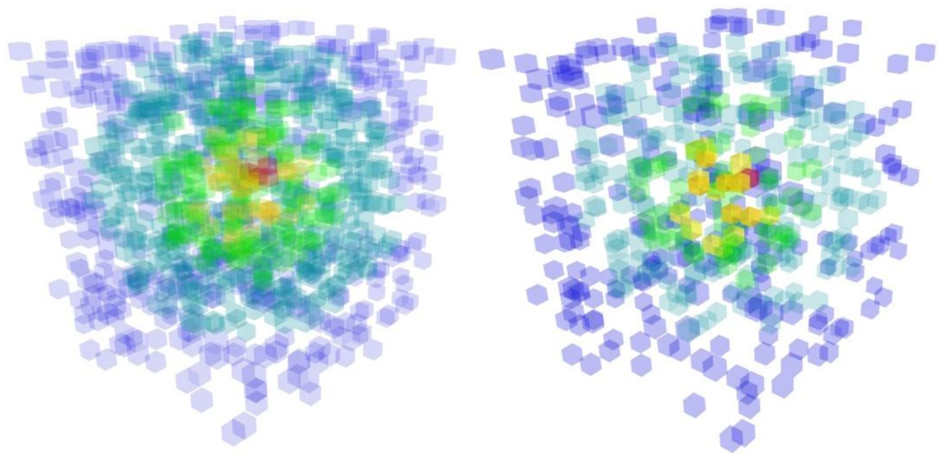


Figure 33 - 10% left and 5% right of original 22x22x22 model

There is a clear progression of the quality from Figure 30 to Figure 33. The circular shells are easily seen until only 15% of the original data remains as seen in Figure 32. At this point the circular shape becomes more difficult to identify because of the high volume of void (empty) space in the model. In the final two models shown in Figure 33 the circular shapes becomes very difficult if not impossible to decipher.

This is one simple example of how error in the form of missing information may manifest using this methodology. Future work using this methodology may be developed which has the capability to dynamically alter the size of the elements to ‘fill’ in the voided space where there is information missing. Similarly, future work could be developed which instead of filling the voided space simply be expanding the size of the known values, it could instead interpolate the voided values by using the known values.

CHAPTER 7: RESULTS

This chapter explores the results of the developed methodology via a demonstration of two models for two unique scenarios. These scenarios were specifically designed to show two different applications of this generic methodology.

7.1 Photon intensity scenario

7.1.1 Scenario

A nuclear engineer is planning an exercise for first responders (police, fire departments, and ambulance workers primarily). In this scenario, the first responders are tasked with searching a house for a stolen radioactive source (15 mCi of Cesium-137). The engineer is responsible for developing instructional material for the radiation safety officer (RadSO) who will be inside the house observing the source and controlling the first responders so they are not unduly exposed during the course of their investigation. Figure 34 shows the layout of the room.

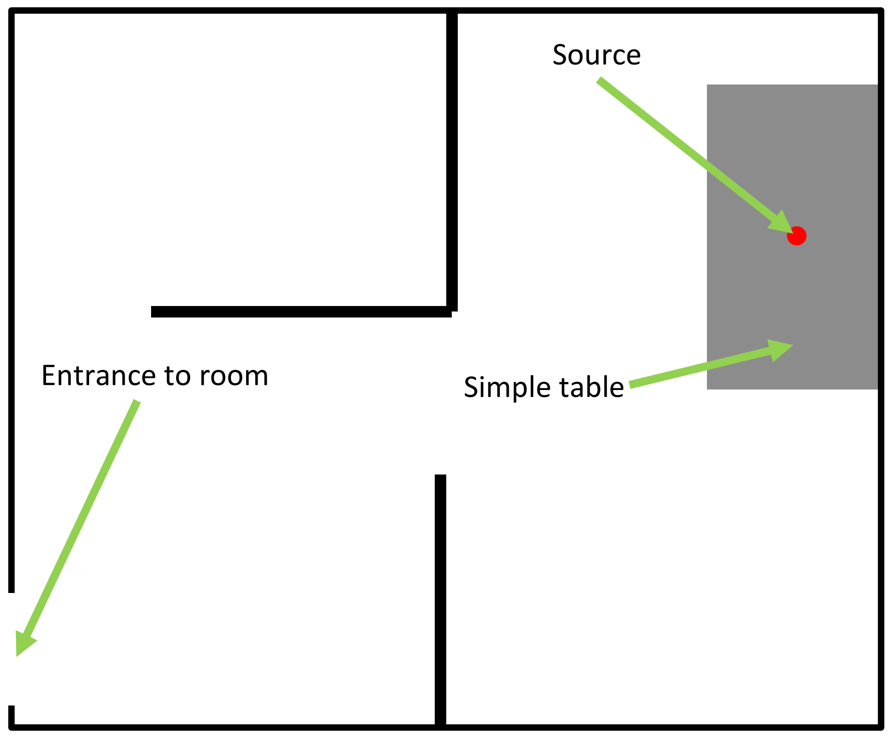


Figure 34 - Simple source scenario layout

The nuclear engineer has decided to construct a model of the scenario in SketchUp and combine that with a model of the radiation field surrounding the source on the table with appropriate dose rate information. He believes this is the most effective way to teach the RadSO the makeup of the radiation field in the room from that source.

7.1.2 Initial scene construction

An approximate model of the scene was constructed within Google SketchUp, as shown in Figure 35 and Figure 36.

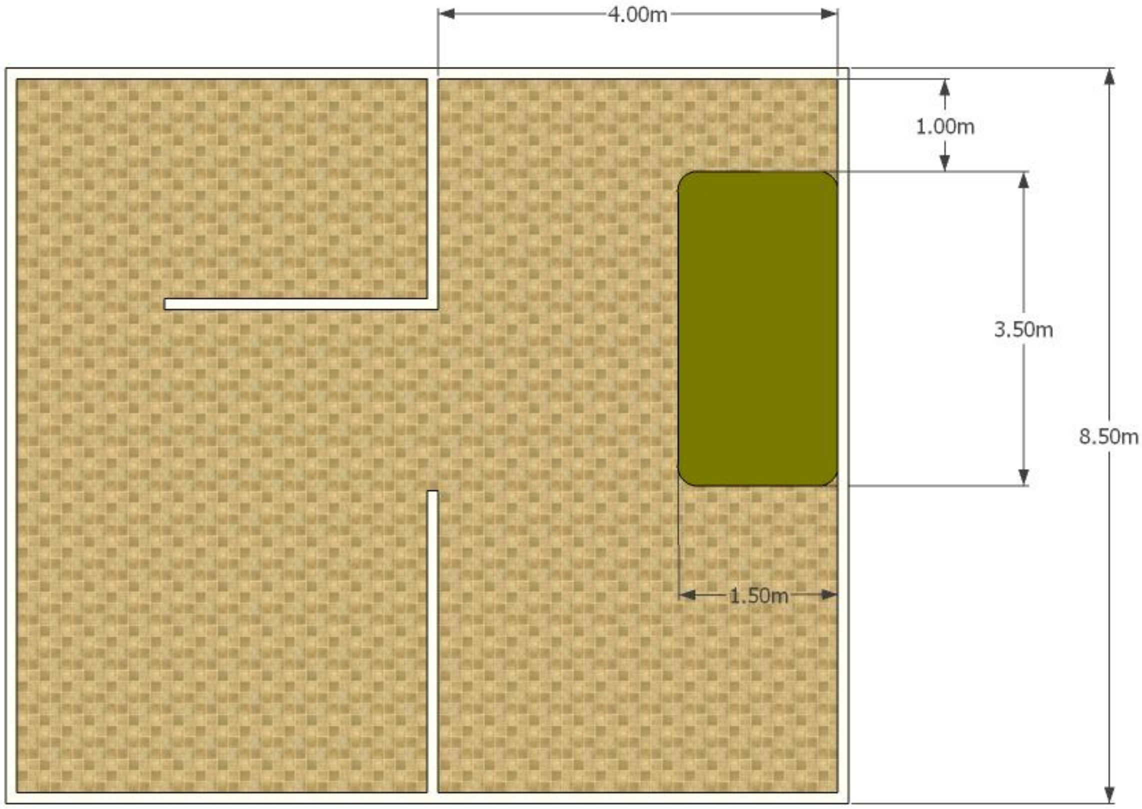


Figure 35 - SketchUp model of scenario (with dimensions)

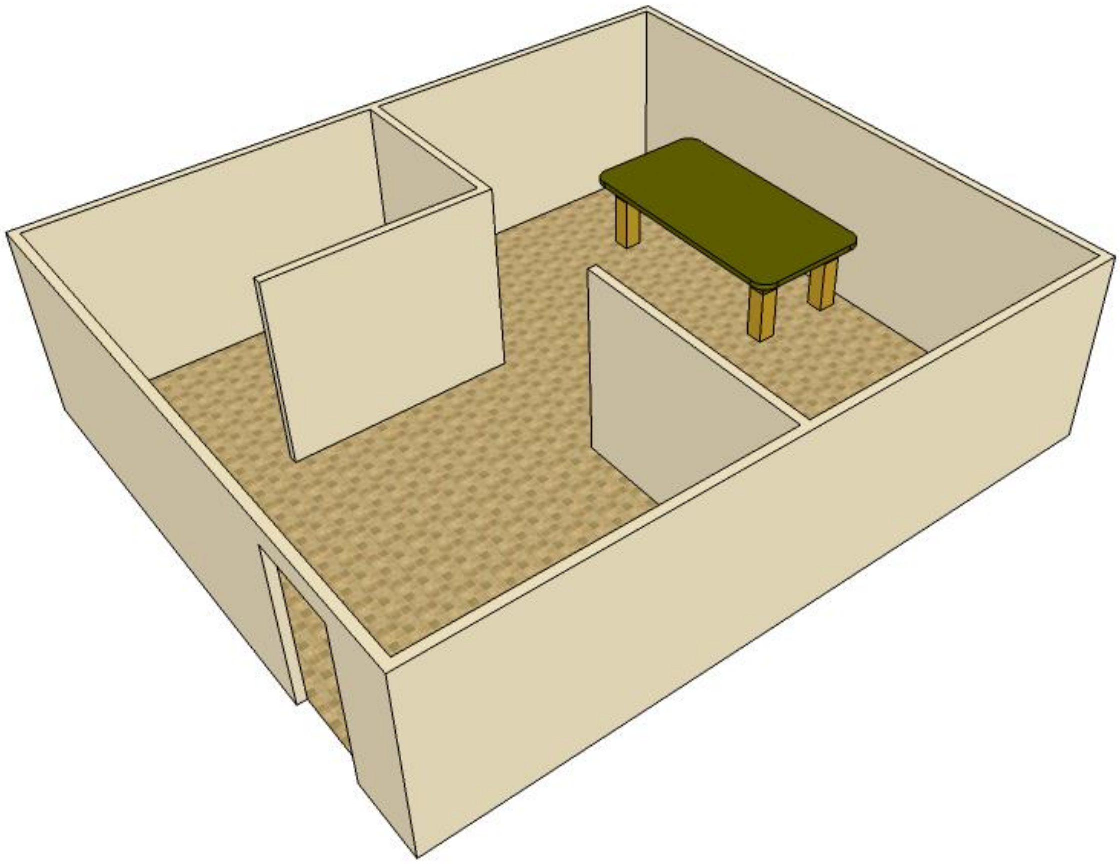


Figure 36 - Isometric view of SketchUp model

7.1.3 Point source calculation

The nuclear engineer was aware that a point source can be modeled based on an inverse square relationship between the flux and the distance between the source and the target. Therefore the flux at a point (particles/cm^2 per second) can be derived from the following equation based on the surface area of a sphere:

$$\phi_i = \frac{P_i(E_i)A}{4\pi \cdot d^2} \qquad (1)$$

Where, P_i is the probability of emission of a photon of energy E_i

d is the distance from point source to receptor point (cm)

A is the activity of the source in decays per second (becquerel, Bq)

And ϕ_i is the flux at that distance d ($cm^{-2}\ s^{-1}$)

For the Cs-137 source only the gamma emissions will be considered. The probability of emission of a gamma ray is 85.1% per decay with an energy level of 661KeV (based upon the Ba-137m state). Using the conversion factors from ICRP 74 for particle fluence to operational dose quantities (H*10) [39], this yields a dose conversion factor (DCF) of 2.924E-12 Sv per fluence in each cm^2. Therefore the dose rate at any distance from the source can be calculated by multiplying the DCF by the flux at that distance. Attenuation in air is not required in this scenario because of the small size of the room and low interaction probability of the photons with air.

Using equation 1, the DCF and an Excel spreadsheet, a setup of coordinates from -500 cm to 500 cm in all directions was created. The source was centered at (0,0,0). A total of 22x22x22 (10648) points were used in this radiation field model which requires a step size of 45 cm in all directions. Using the distance between the coordinates and the point source as d, the dose rate was calculated with respect to the center of each element. To produce models automatically within SketchUp a Ruby script is required to import the information from a text file into SketchUp following the procedure outlined in Chapter 6.

7.1.4 SketchUp construction

The construction script was run on a blank SketchUp model template, and the resultant models are shown in Figure 37 and Figure 38.

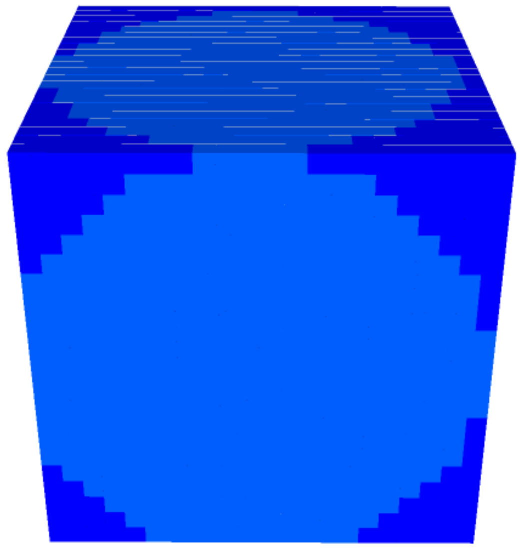


Figure 37 - Point source SketchUp model (before transparency is applied)

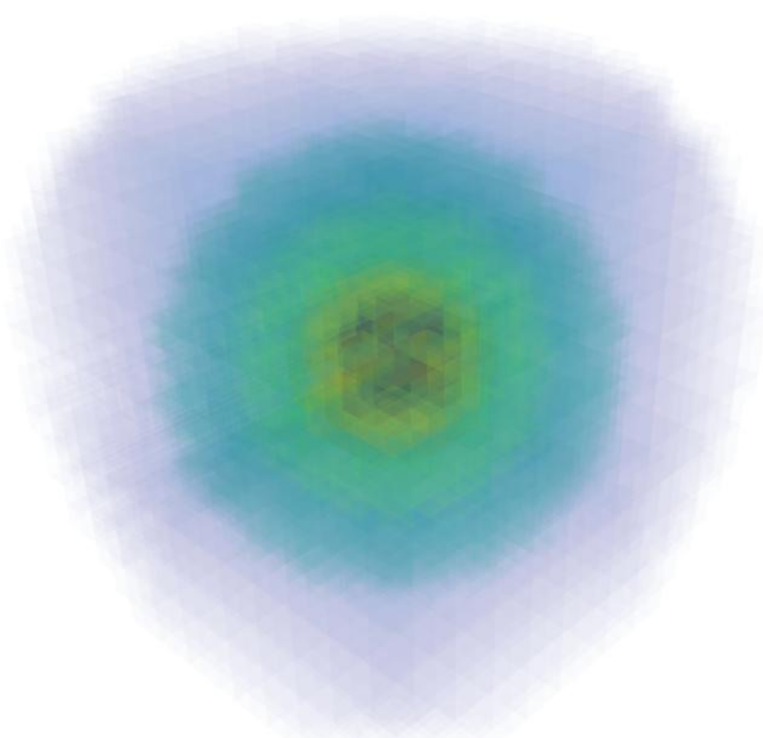


Figure 38 - Point source SketchUp model (after transparency is applied)

Once transparency effects have been applied to the model, the shape of the radiation field is abundantly clear and very easy to understand.

7.1.5 Model amalgamation

This field model was combined manually with the 3d model of the environment. The field model was moved onto the table such that the center point in the model was directly on the surface of the table where the source was positioned.

7.1.6 Complete model

The complete model is shown in Figure 39 and Figure 40. The outer most layer (representing values less than 2.5 μSv/h) was removed to avoid overwhelming the RadSO with unimportant information. It is worthy to note that this example did not consider the shielding effects of the table, nor any scattering within the room. To generate data corresponding to detailed shielding effects, a more complex simulation (such as MCNP) would be required.

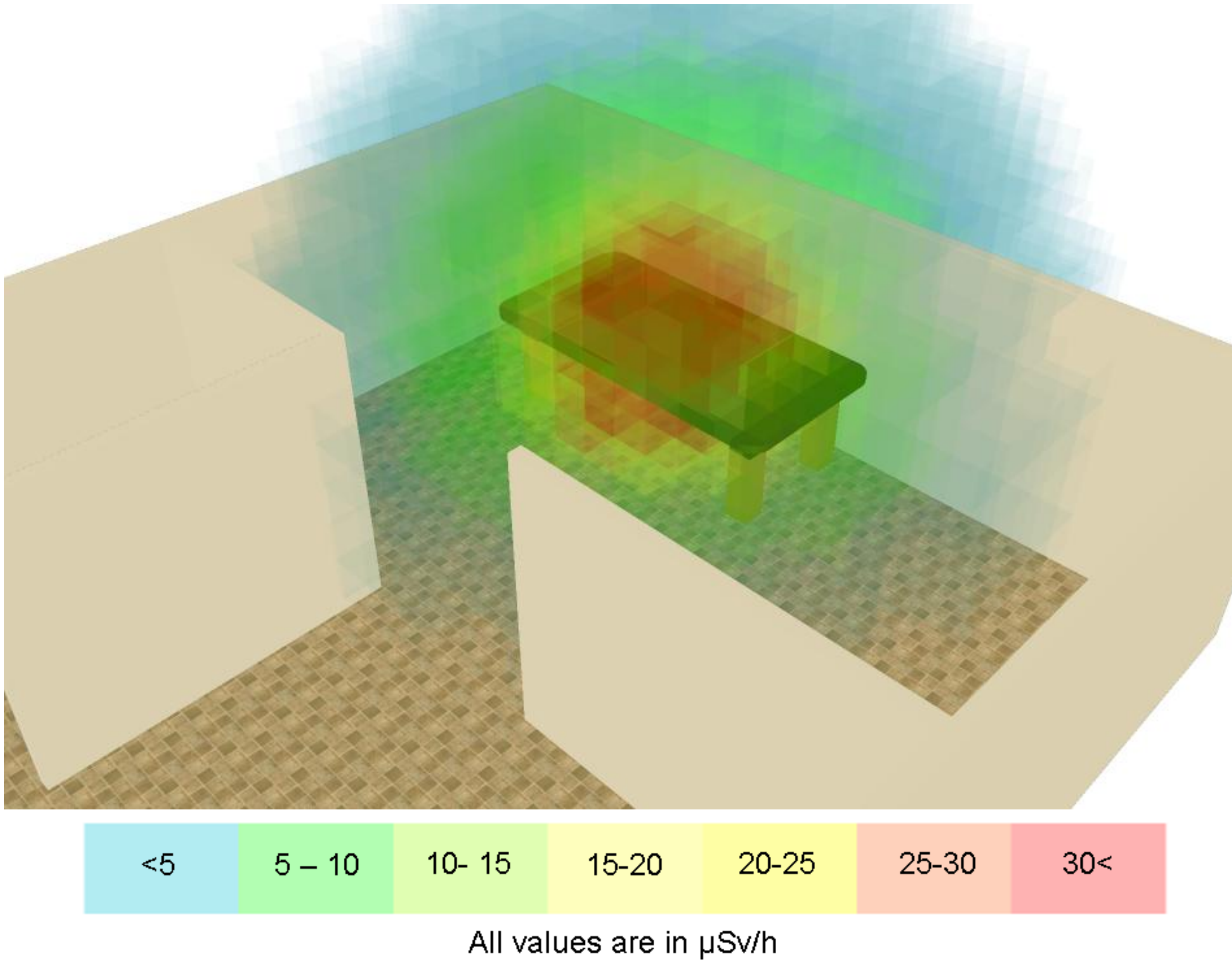


Figure 39 - Point source on table with scale (isometric perspective)

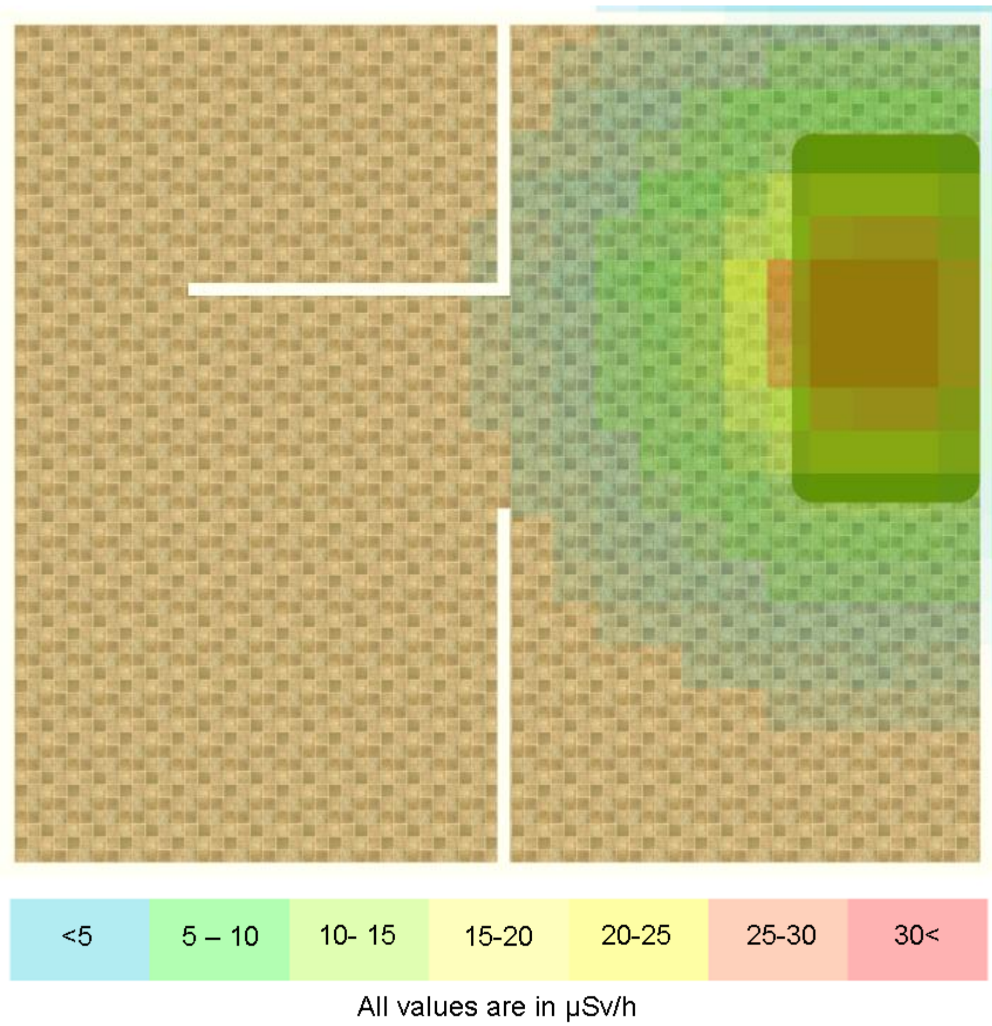


Figure 40 - Point source on table (top perspective)

7.1.7 Conclusions from point source scenario

With the model complete the RadSO can be given very specific information (such as depicted in Figure 39 and Figure 40) outlining where they can position themselves within that room to maintain ALARA and still oversee the scenario. When the RadSO arrives onsite there may be obstructions within the room that cannot be moved. If the radiation safety plan was based around only a few measurements taken around the source, the RadSO would have to create a new plan and possibly revise their positioning to a new location using only those measurements (or direct measurements). With the material produced from this method, obstructions can be marked off on the maps and a new location can be selected based on the very easy to decipher color imagery. This will help the RadSO maintain ALARA.

This is a simple, but novel illustration of the potential this methodology has to improve ALARA planning and training. Using this technique, simple images can be made which can contain far more useful information than a small selection of measurements. The shape of the field is made abundantly clear to the RadSO in all three dimensions and based on this information intelligent decisions can be made rather than best estimates.

7.2 MCNP visualization

7.2.1 Scenario

In this hypothetical scenario a member of a radiation protection group is tasked with teaching a group of technicians about the hazards they will encounter while working in a room where there is an irradiation device currently ‘stuck’ open. These technicians are skilled laborers in their various mechanical fields but they have never worked around nuclear equipment or inside environments with active radiation fields before.

Calculations and historical measurements have shown that the only area in the room where the dose rate from the irradiator exceeds a safety margin set by the facility (arbitrary in this scenario) is directly within the beam path and in close proximity to direct contact (less than 5cm) with the shielding surrounding the source. The workers need to access two different panels which are outside of the high radiation field areas. The workers are uncertain about the risks of radiation and have requested clear and concise instructions from the radiation protection staff which show them where they can and cannot move within the room. Figure 41 shows a 2d map of the scenario.

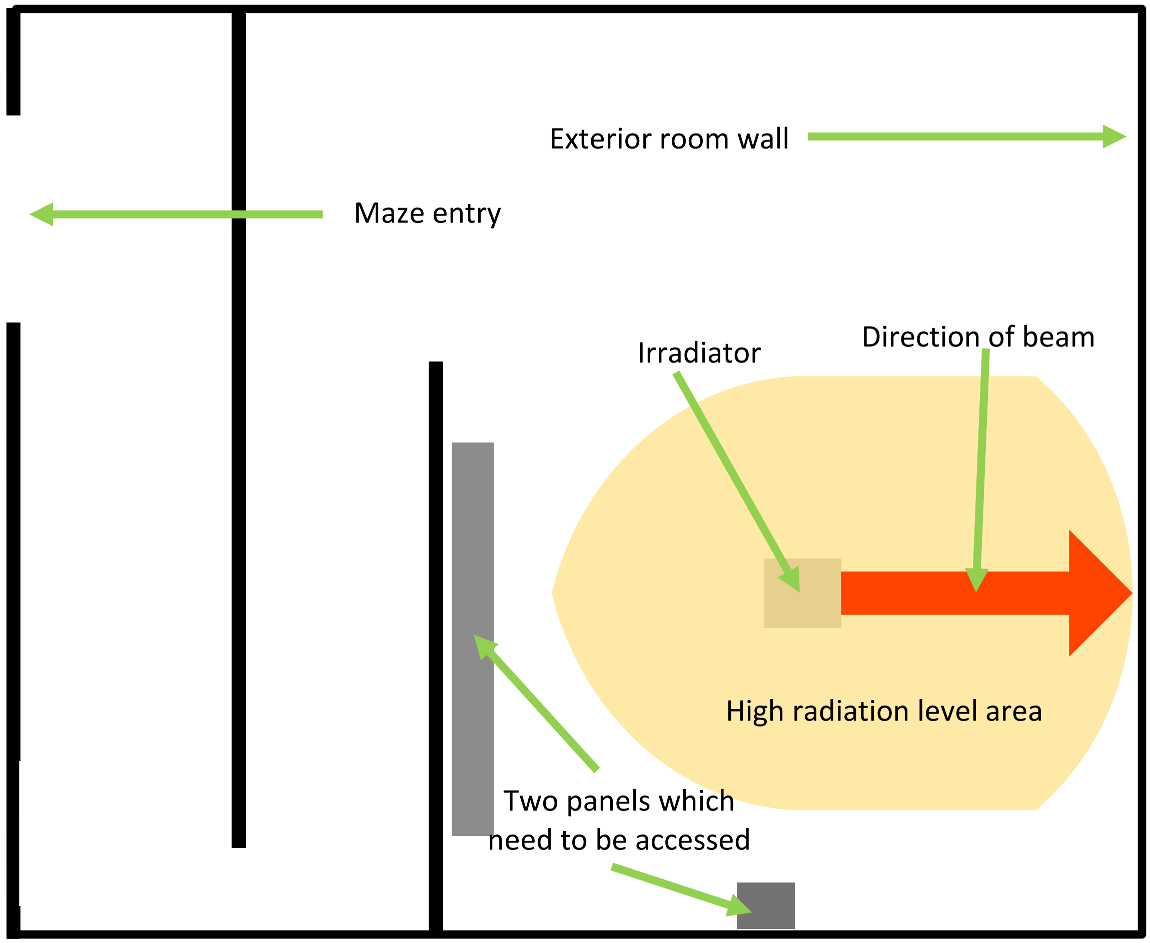


Figure 41 - Irradiator scenario layout

The radiation protection staff has chosen to construct a 3d visualization of the irradiation room so that they can do a complete walk through of the area with the technicians and explain every element of the repair in detail.

7.2.2 Initial scene definition within MCNP

7.2.2.1 MCNP

A model was constructed in MCNP (software version MCNP5 – 1.51). MCNP is a general purpose Monte Carlo particle simulation code [40]. It allows the construction of geometry with specific properties (elemental composition, density etc) for a simulation.

Particles (photons, neutrons, electrons etc) are then simulated and 'ran' through the geometry interacting and depositing energy through these interactions. These Monte Carlo simulations are used to provide very accurate dosimetry calculations in complex geometries.

In Figure 42 and Figure 43 the MCNP model of the irradiator is seen, as viewed through the MCNP visual editor application (MCNPX Visual Editor Version 26e) in both 2d and 3d visualizations modes.

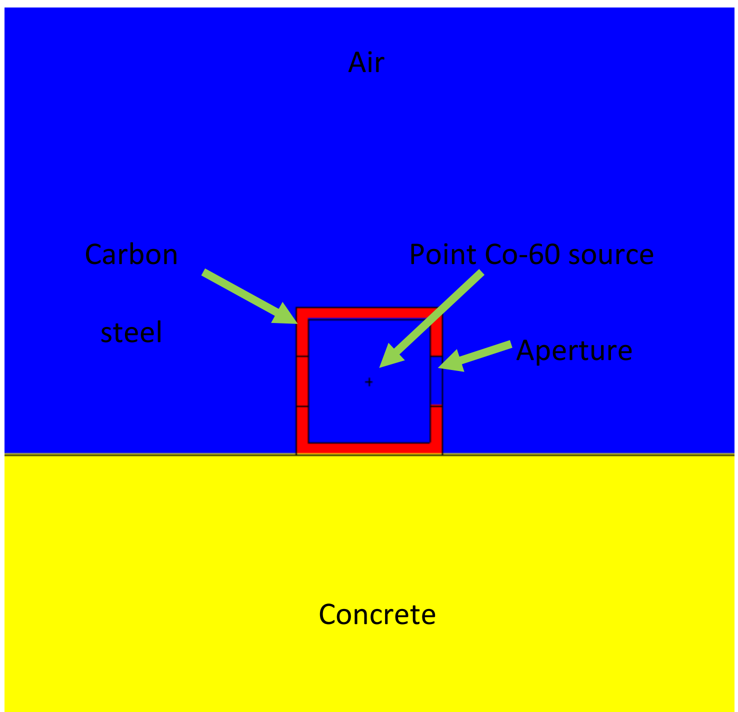


Figure 42 - 2d MCNP diagram of the scenario

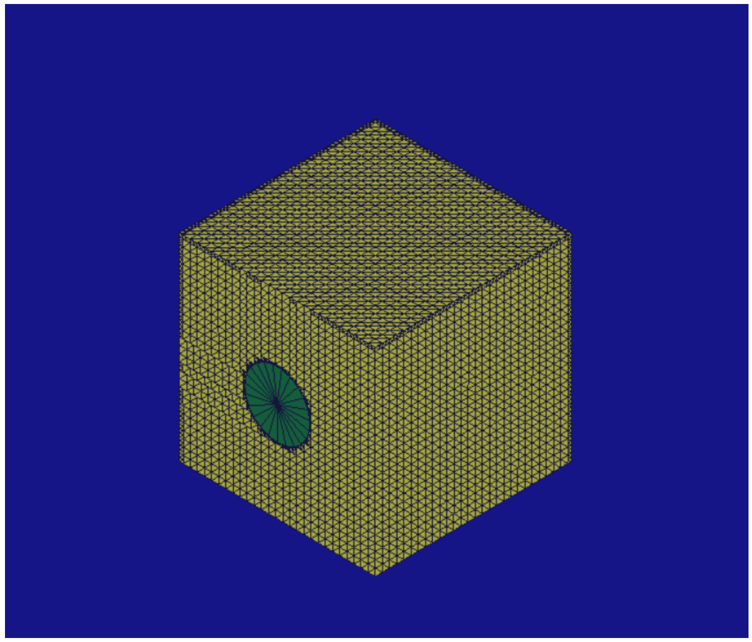


Figure 43 - 3d MCNP model in ‘dynamic’ view mode

The model consists of a concrete floor, a box with the same dimensions as the one described in the scenario (including the aperture), and a Cobalt-60 source centered in the box. As this is for education purposes, this simple model is sufficiently detailed to produce the data needed for field visualization without burdening the user with the modeling of details such as the hallway, walls or surface abnormalities on the shielding.

7.2.2.2 SketchUp

A second model with the same dimensions was constructed in SketchUp, as depicted in Figure 44, Figure 45 and Figure 46. This model will serve as the SketchUp scenery for the field visualization. Additional details that are not required within the MCNP model (such as surface textures and colors), can be added in SketchUp at this stage creating a more realistic visual model of the scene than is possible within the MCNP visual editor.

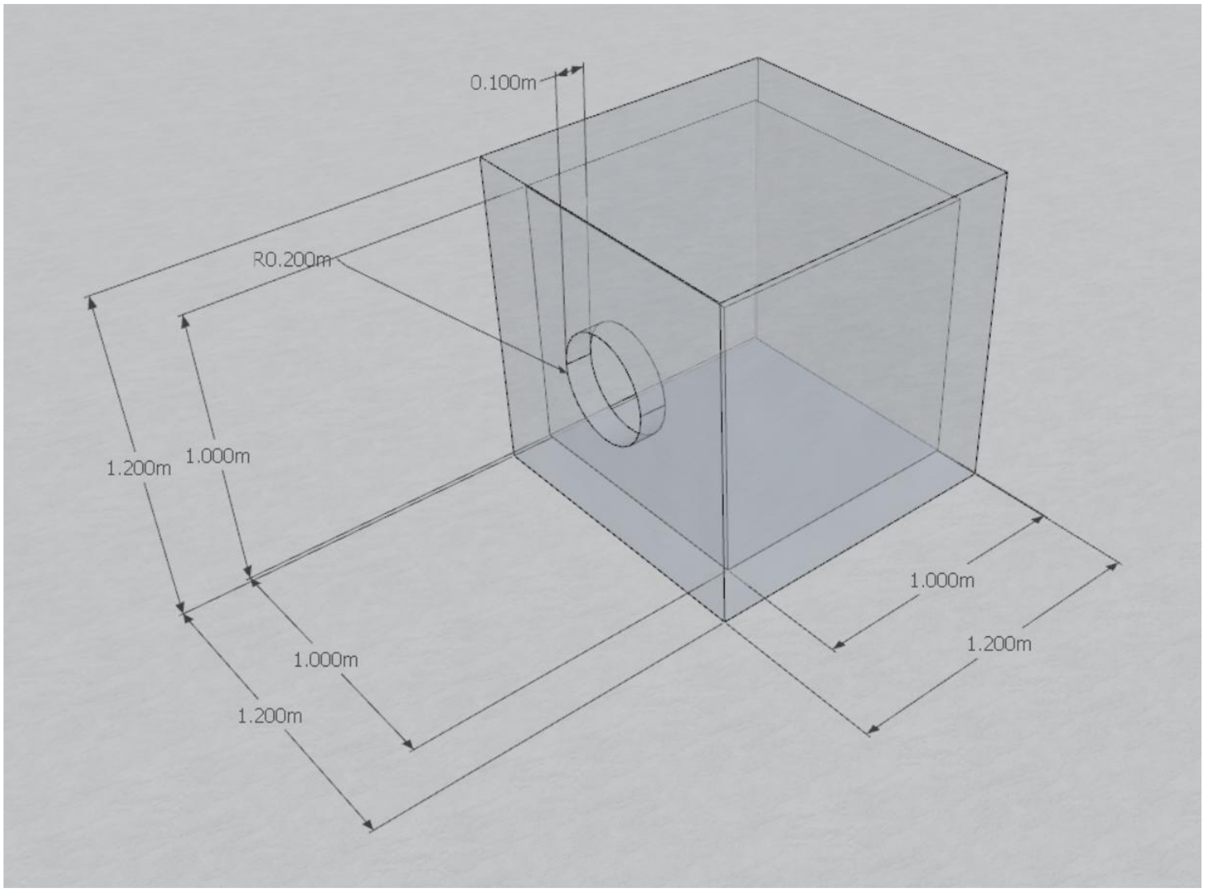


Figure 44 - SketchUp model of the irradiator with dimensions

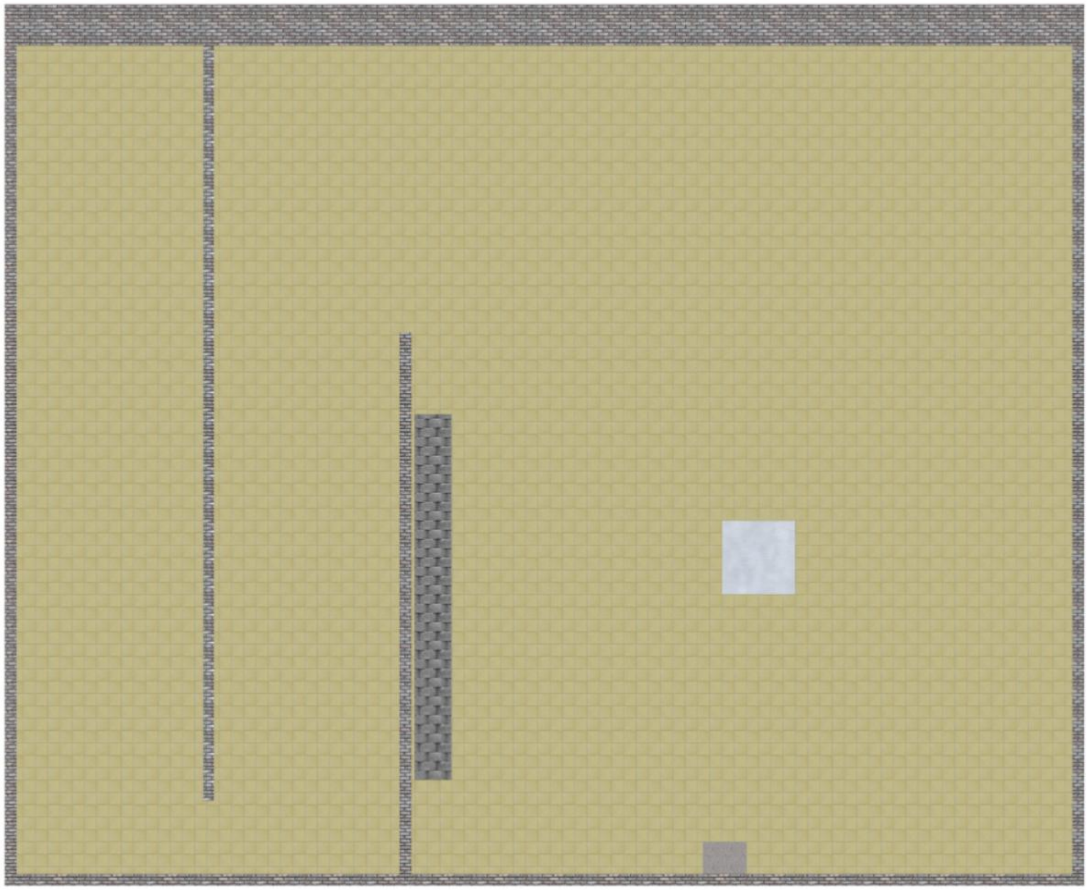
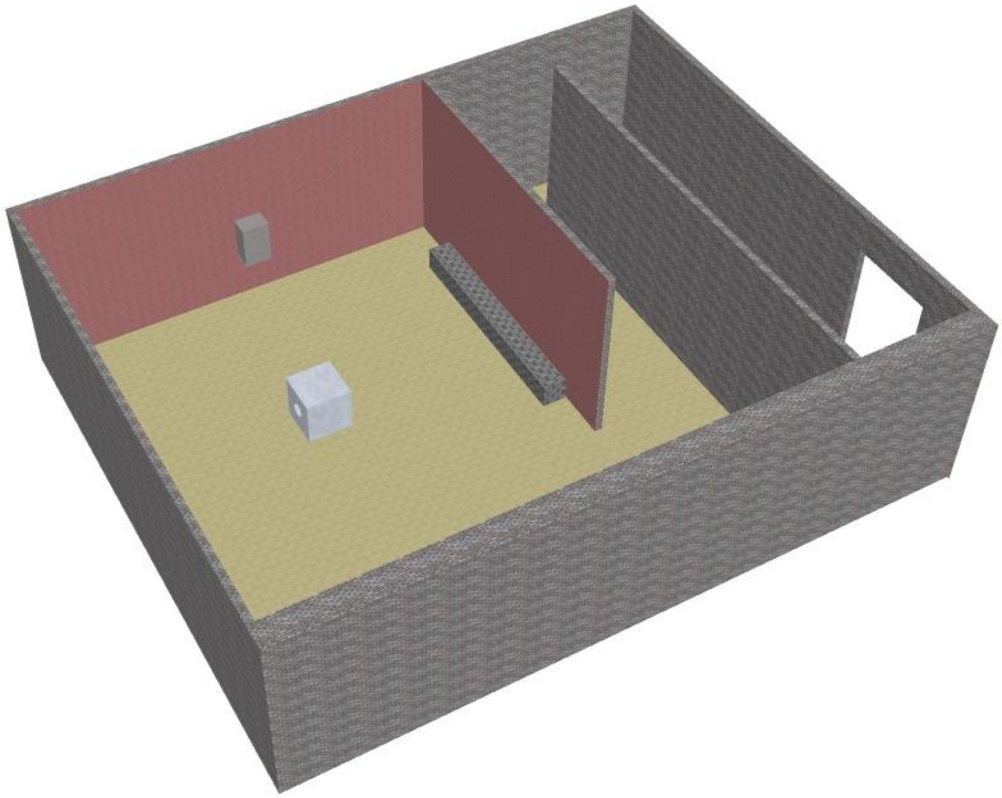


Figure 45 - Overhead view of maze using SketchUp



7.2.3 Figure 46 - Isometric perspective of the SketchUp modelMCNP calculation and processing of the output

To generate the data required for the model input file the MCNP mesh tally function will be used. Mesh tallies in MCNP are a method of calculating values (such as fluxes or doses) within an MCNP file which is independent of the geometry of the problem. They can be in several different shapes but for this scenario the rectangular shape is appropriate.

A mesh tally is defined on 3 axes. A user inputs where a mesh tally will be calculated (i.e. from 0 to 300 cm on one axis), then the user defines how many subdivisions within that direction are required (i.e. 10). As this mesh is extended into the other axis, a 3d mesh is created. When the MCNP simulation is run, the program will keep track of the value (flux, dose, etc) specified within each part of the mesh. Finally, the program will output an *mctal* file. An mctal file is a collection of coordinates representing each element of a mesh, the reading inside that element and the error associated with that reading. The mesh tally can be used to limit the extents of the problem to only producing measurements within specified areas of interest.

For this example two MCNP mesh tally calculations were required (See Annex B for MCNP code). The first calculation extends the mesh tally from the source at the center of the irradiator, to 200 cm in all directions with step sizes of 16 cm (thereby giving an elemental size of 16x16x16 cm). This will be delineated as the 'inner' field. This tight mesh is used to establish the shape of the field around the outer surfaces of the irradiator in greater detail than a coarse mesh.

The second mesh tally extends 500 cm in all directions. The step size in this mesh is 40 cm. This will be delineated as the 'outer' field. This tally is designed to establish the shape of the beam protruding from the aperture.

It should be noted that it was not required to use cubic shaped elements; other shapes could have been used. Cubes were selected for simplicity: 15675 elements are in each model which will eventually lead to a combined model of 31350 individual elements in the field visualization.

Both models were built using the exact same ranges in their color scales which ensures they are completely compatible with one another. The only change between the Ruby scripts for the models is the size of each elemental shape (from 16x16x16 to 40x40x40). The data from the mesh tally runs are converted into a format compatible with the code in Annex A (as this code cannot read an *mctal* file directly). This required importing the *mctal* file into a spreadsheet program and extracting the coordinates and readings (in this case average absorbed energy in each mesh component).

7.2.4 SketchUp construction

The scripts were run separately on blank SketchUp model templates producing the results depicted in Figure 47 and Figure 48.

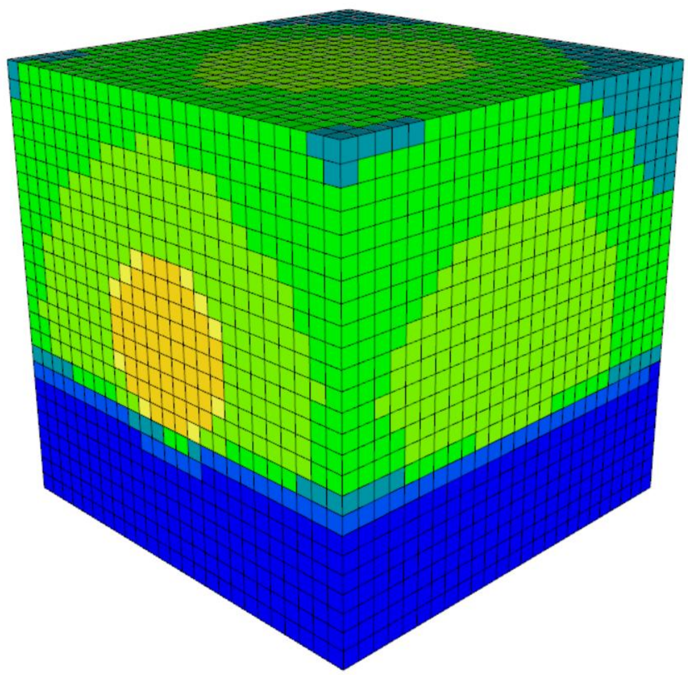


Figure 47 - Inner mesh tally SketchUp model (before transparency is applied)

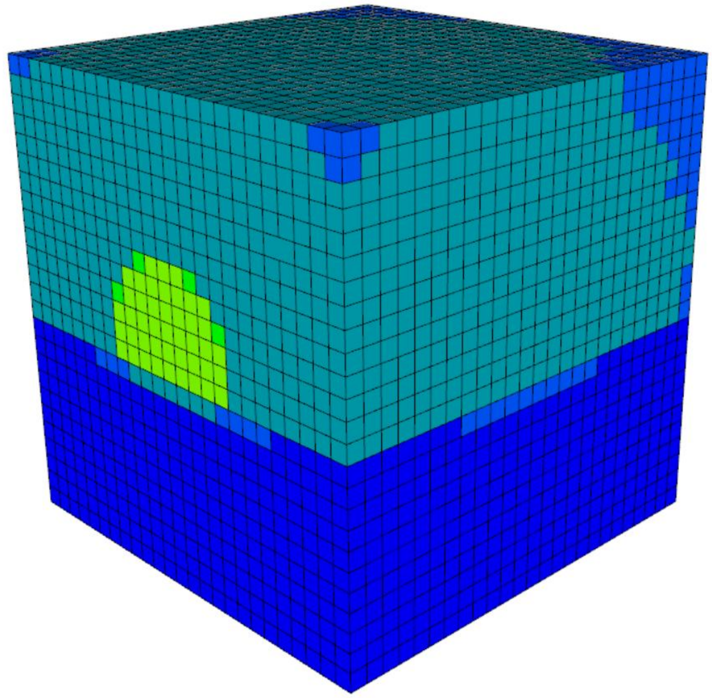


Figure 48 - Outer mesh tally SketchUp model (before transparency is applied)

These models are not complete until they have had their transparency properties adjusted. This takes place by matching the opacity values in both models to appropriate levels. There is an esthetic aspect of model manipulation required at this point. It is always up to the user to determine what is most appropriate to suit their particular needs. The results are depicted in Figure 49 and Figure 50.

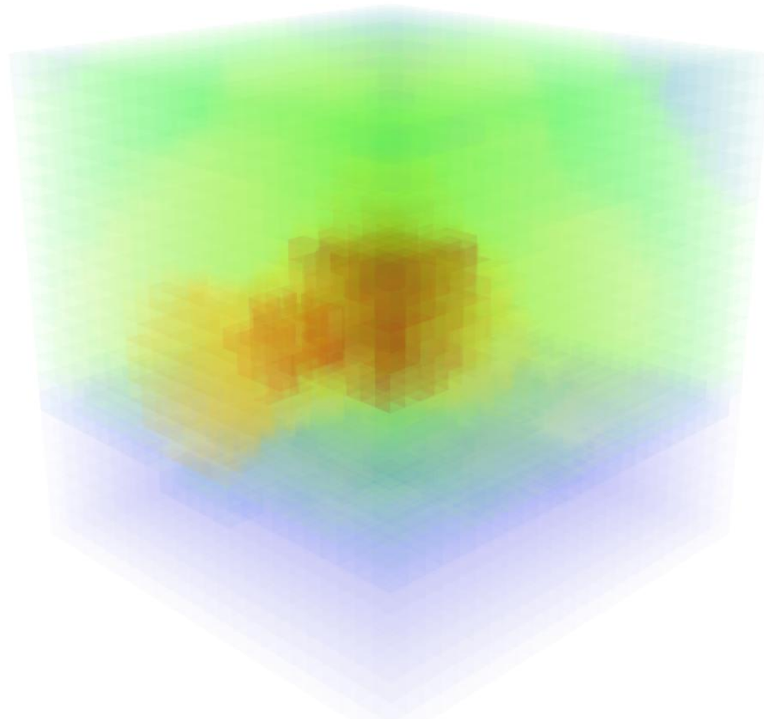


Figure 49 - Inner mesh tally SketchUp model (after transparency is applied)

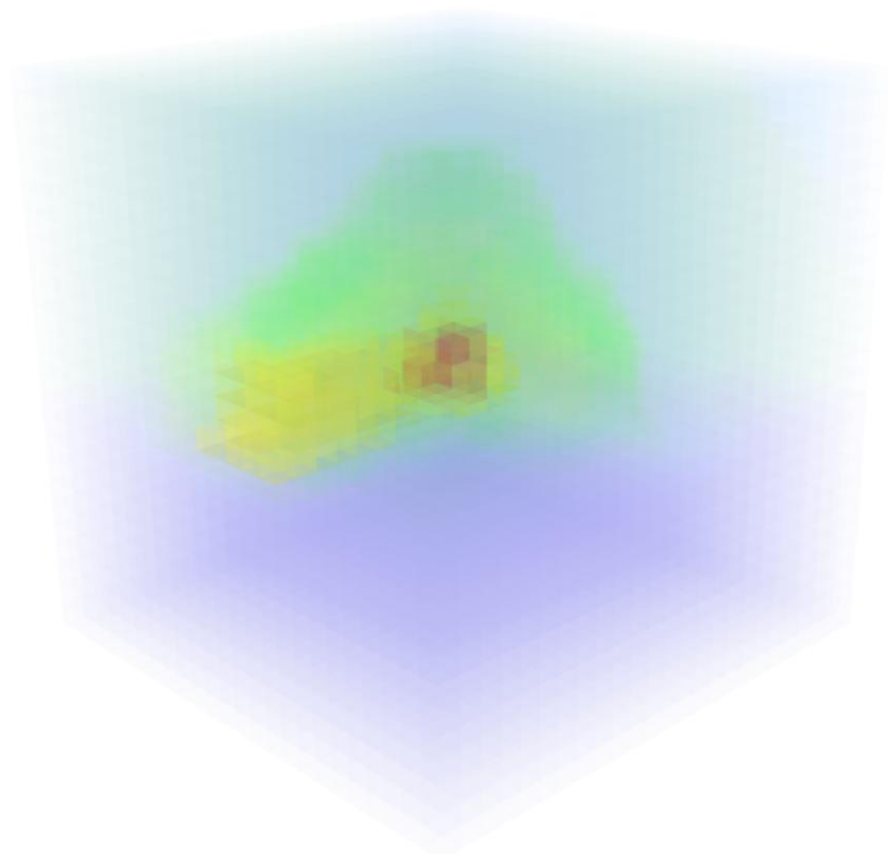


Figure 50 - Outer mesh tally SketchUp model (after transparency is applied)

7.2.5 Model amalgamation

The two models were constructed and imported into the model of the irradiator. All three models shared the same geo-referencing system (i.e., both share the same coordinate system where point (0,0,0) is the same in all three models). This allowed the models to be automatically positioned within the final construction.

The complete model is shown in Figure 51, Figure 52 and Figure 53. Note that tools such as SketchUp's 'walking' and 'looking' tools (discussed in Section 4.5.1) can be used to make powerful live demonstrations to audiences where required.

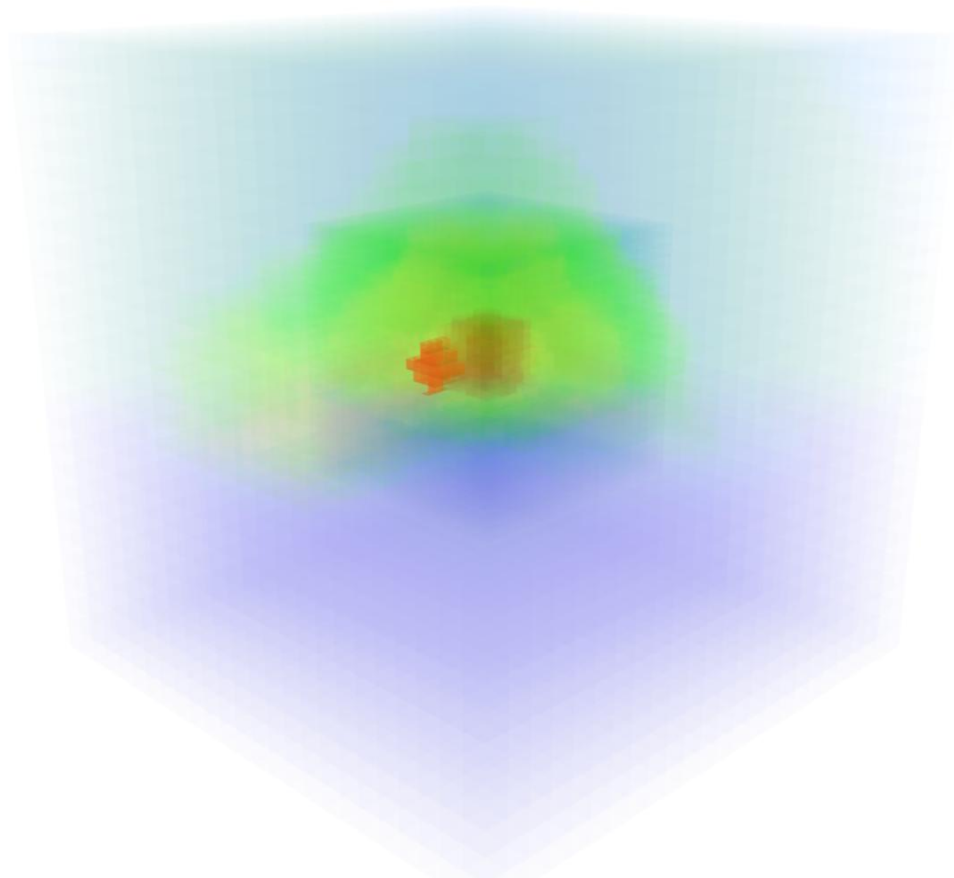


Figure 51 - Inner and outer mesh tally models aligned with each other

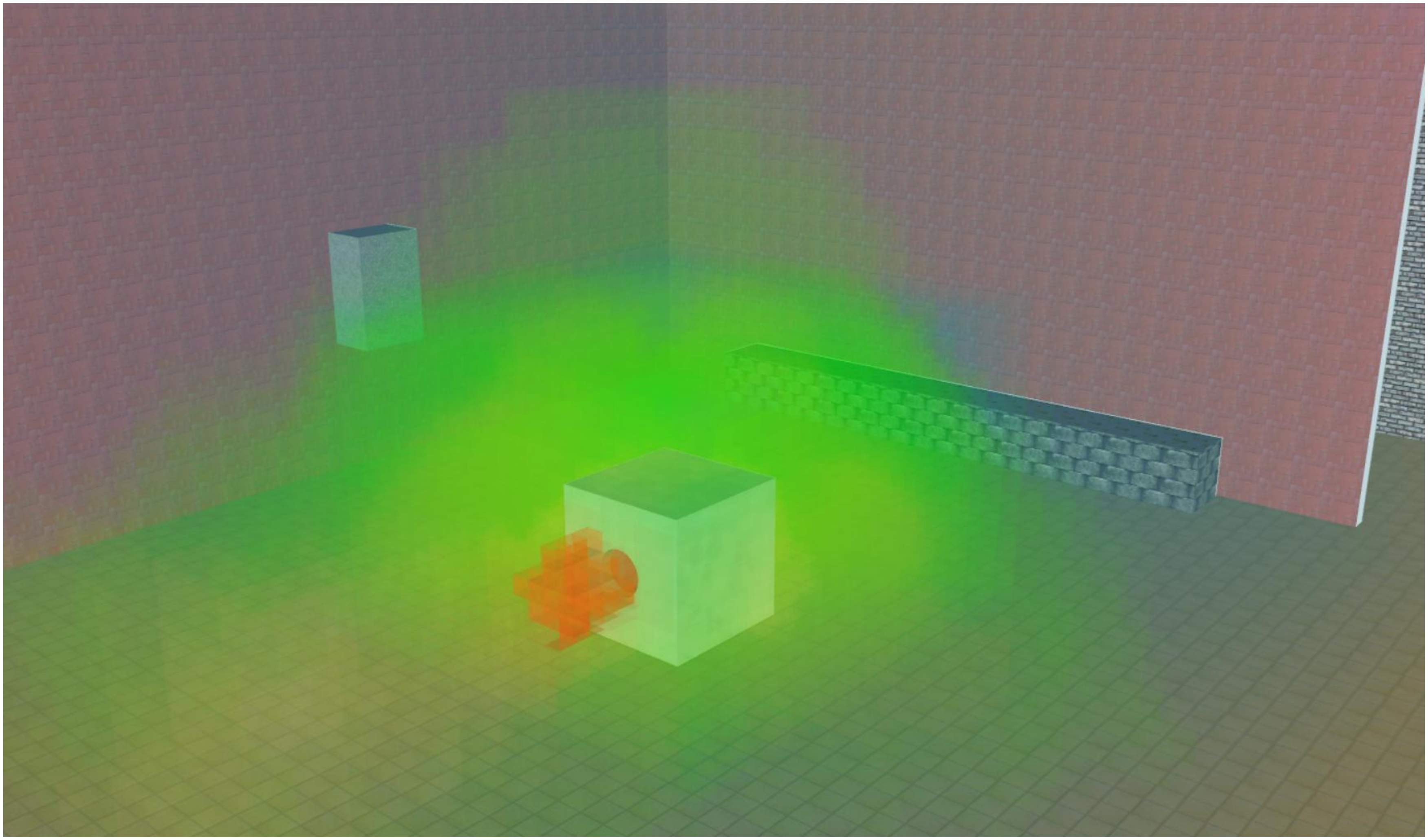


Figure 52 - Complete model of the irradiator with both inner and outer visualizations

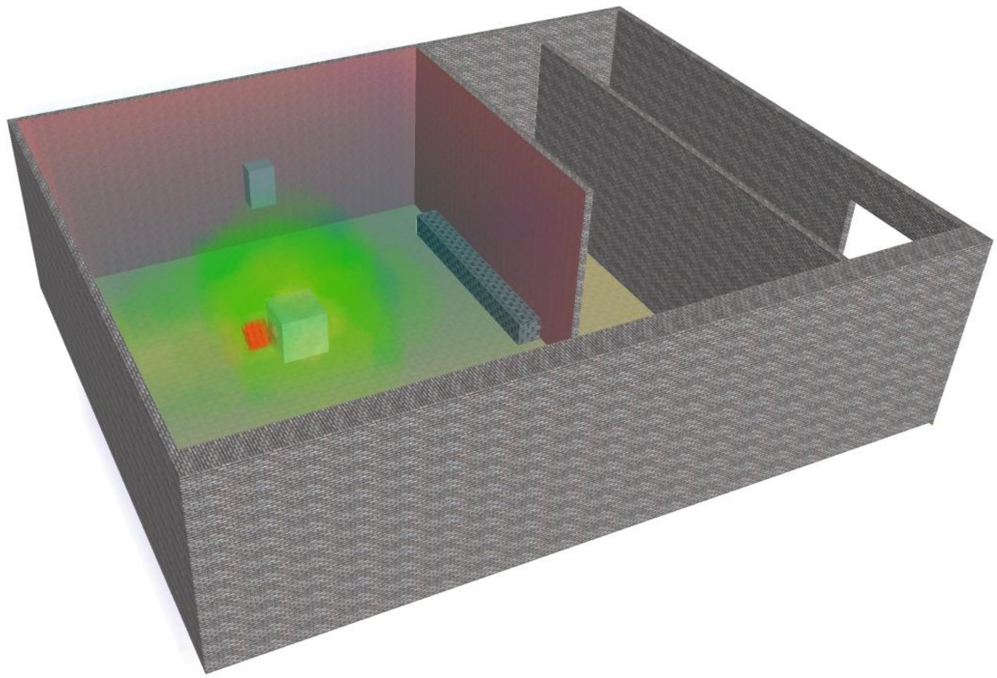


Figure 53 - Complete model isometric perspective

With the model constructed the technicians can be 'walked' through the repair using the tools discussed in Section 4.5.1. Furthermore the technicians can also be provided with maps of the irradiation room with the areas to be avoided clearly marked. In Figure 54 the technicians have been provided with a topographical map of the room that shows the area to avoid. This was generated by removing all layers of the field models (by making them completely transparent) except for the 'green' layer. All transparency was removed from the green layer. This highlights the many different uses of a model once it has been constructed.

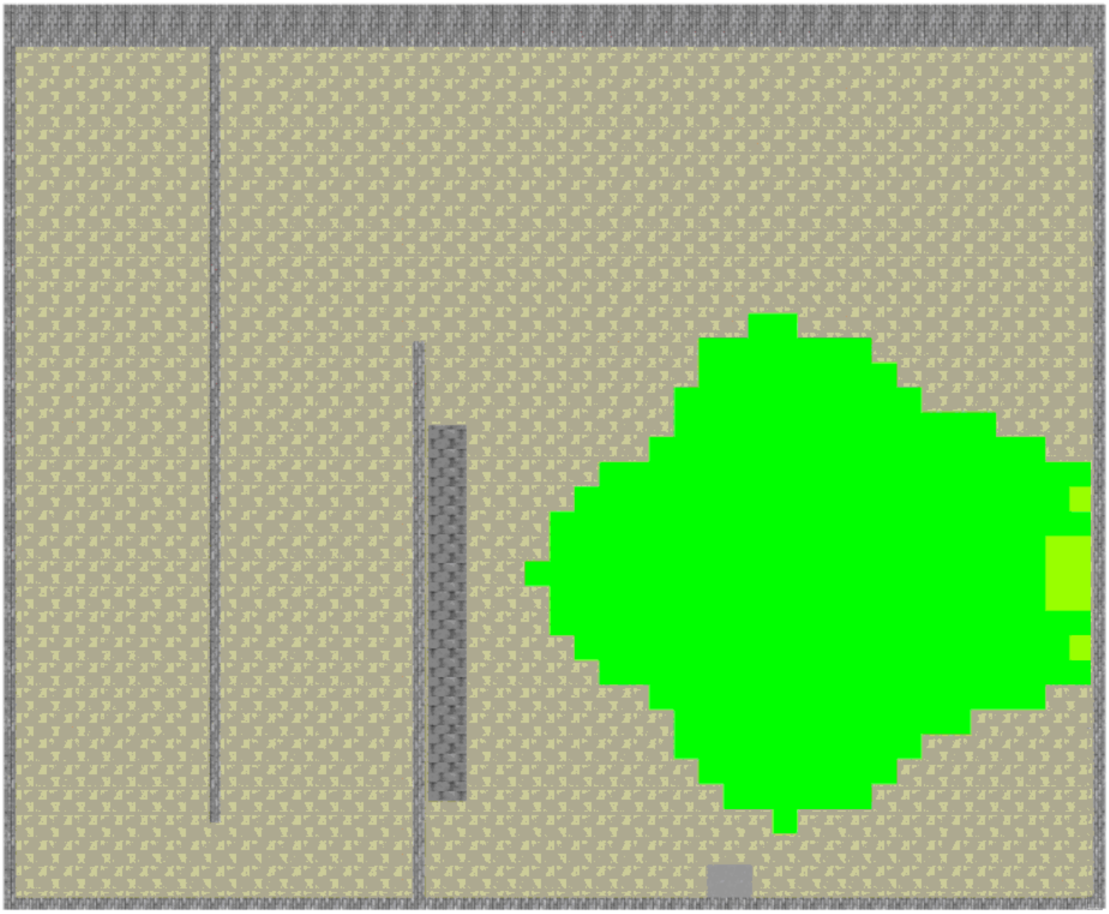


Figure 54 - Top view of irradiation room with area to avoid marked in green

In Figure 55 the same technique was applied except this time the highest 3 levels (yellow, light red and dark red) were made to be mostly non-transparent (opacity set at 80%). Again, this allows the technicians to be provided with 'reminder' material explaining where the strongest radiation fields are. This material can be a useful refresher prior to entry, or even used during their repairs to remind them about the shape of the radiation field in the highest dose rate areas.

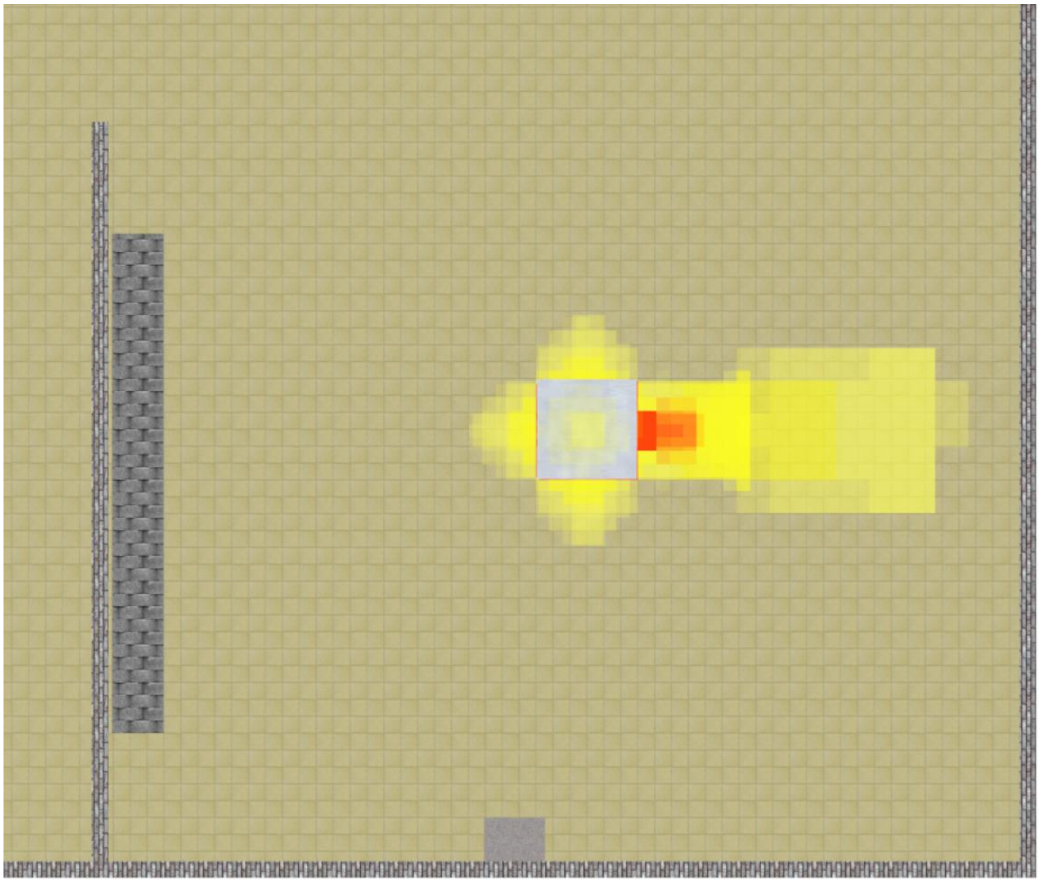


Figure 55 - Top view of irradiation room with the most hazardous levels shown clearly

7.2.6 Conclusions from irradiator scenario

Using the complete model the technicians can be given a wide array of material to enhance their understanding the hazards within the irradiation room and educating them to be aware of what they need to do to maintain ALARA. A walk through covering, the repair process, in a step by step manner is a very powerful learning tool that can be provided with this methodology. Using the model to produce additional refresher material such as the topographical maps in Figure 54 and Figure 55 demonstrate just how many additional uses are able to be accomplished with this novel technique. This further confirms the usefulness of this procedure for ALARA planning and training.

CHAPTER 8: IMPACTS

This chapter discusses the impacts of this work. It discusses the requirements established in Section 3.2, and some novel uses of this research. It is intended to highlight the many different possibilities for this methodology to be used to solve future problems.

8.1 Design requirement analysis

In Section 3.2 Table 1, design requirements were established for this research. Each requirement will be listed separately and a discussion of how it was met will take place. Table 7, shows that every design requirement set out to be achieved by this methodology has been met.

Table 7 – Discussion of design requirements and achievements

Issue	Design requirements
Visibility	• Distinguish internal radiation field features without obstructing the view of other features
Discussion: As has been shown in Sections 7.1.6 and 7.2.5 this methodology and this process produces visuals which allow internal details to be seen without being obstructed by external views using transparency effects. This design requirement was achieved.	
Navigation	• User friendly method to navigate a field model

Issue	Design requirements
Discussion: Using the SketchUp program provided a very user friendly method to navigate the field models. One of the biggest factors in selecting SketchUp was its very intuitive interface and high degree of user friendliness. Although this particular design requirement is subjective, it is considered to have been achieved.	
Color/scales	• Documentation and instructions for end users to manipulate and apply logical color scales to best visualize a radiation field
Discussion: In Section 5.3.1 a discussion of this issue took place and a recommended approach for defining and using color scales was given. This design requirement has been achieved.	
Portability	• Open source field storage format to maximize the future potential uses of a modeled field
Discussion: Google SketchUp can save files in the open source COLLADA format. In the future this COLLADA format may become very useful for additional uses of this methodology (such as exporting models to other formats). This design requirement has been achieved.	
Model limits	• Capability to limit a radiation field within a region of interest
Discussion: In this methodology this is possible during the organization of the data when it is fed into the construction process. This design requirement has been achieved.	

Issue	Design requirements
Others	• Open and easy to use such that any engineer, scientist or health physicist can construct a field model with a minimal amount of effort
Discussion: SketchUp provides a freely available program any engineer, scientist or health physicist can download and install. The methodology is conceptually very easy to understand and therefore should be easily adapted by others. The field definition process is very open ended maximizing the potential definition techniques. This final (and perhaps most important) design requirement has been achieved	

8.2 General radiation field display

As shown in the previous chapters, this methodology can be used for general applications where a 3d visualization of a field would be beneficial. These applications will likely fall into one of two very general categories: Education and Engineering

- Education field visualizations are those constructed where the purpose is simple information communication. These fields are expected to be simplified not in their appearance, but rather in their meaning. For example a simplified radiation field may simply use a 3 color gradient scale of green, yellow red to indicate general field intensity. These levels while appropriate for general education uses will likely hold no bearing with respect to an engineering based field model where viewing a much wider range of intensity values would be more important to maximize the detail available for visual analysis.

- Engineering field visualizations will require careful planning in the selection of the methodology used to derive the field. This class is specific to those that will be used to design and communicate complex radiation field information in a manner that provides a high degree of scientific context. Engineering fields would likely be comprised of tighter meshed constructs and finer resolution builds. It is expected that most fields designed for engineering purposes will require several simplified iterations to fine-tune a model before a final more complex model is eventually built.

8.3 2d data

This methodology can be applied to very quickly construct 2d topographical models for import into 3d worlds. These 2d models can be built and colored in the same way as a 3d model. In addition, since their plane of interest is located in only the 2 dimensional areas, a third dimensional perspective can be added by also using the value of each reading and relating that to a height.

In Figure 56 an excel spreadsheet was used to establish 10 point sources randomly distributed on a 2d plane (of 100x100, 1x1 m cells). The sources randomly vary in activity between 20% and 100% of the largest activity source. Using the conditional formatting feature in Microsoft Excel™ the cells in that plane were automatically colorized between green (lowest), yellow and red (highest) dose rates.

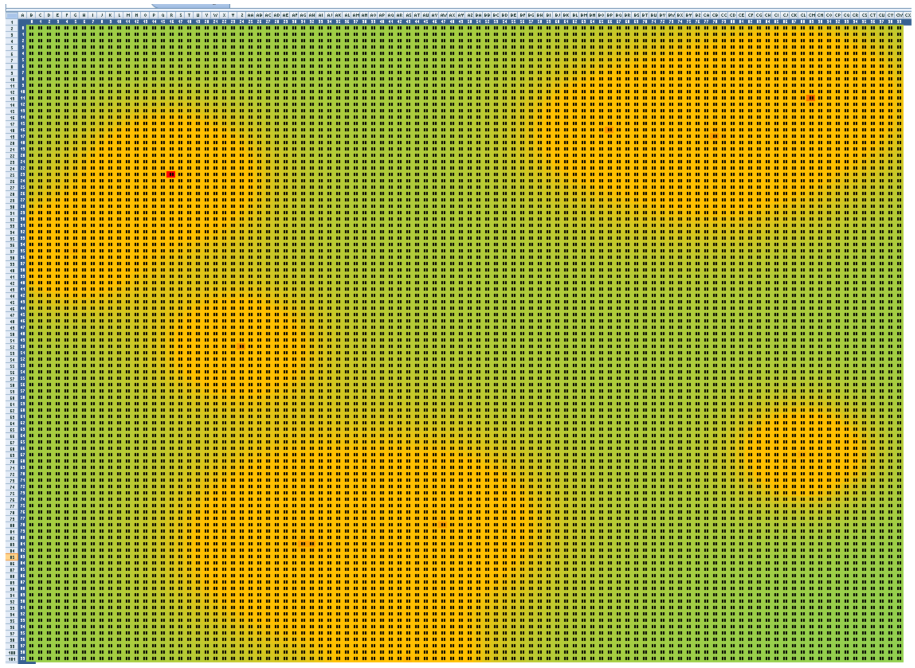


Figure 56 - 2d data from several points sources plotted in Microsoft Excel™

That data was organized and put into a format compatible with the SketchUp code shown in Annex A (I.e., X_n, Y_n, Z_n, V_n). Since there is no 'z' value for each coordinate (rather they only have x and y values), z was simply set to 0 throughout the build process. The dose rate in each cell was used to control the height of the block at that cell (in the 'z' direction), meaning the higher the reading within a cell the higher the cell was constructed (replacing 'pushpull 1' in the code in Annex A with 'pushpull color' causes each box to be pushed to a height equal to the 'color' which is the reading value). Figure 57 illustrates the results of this.

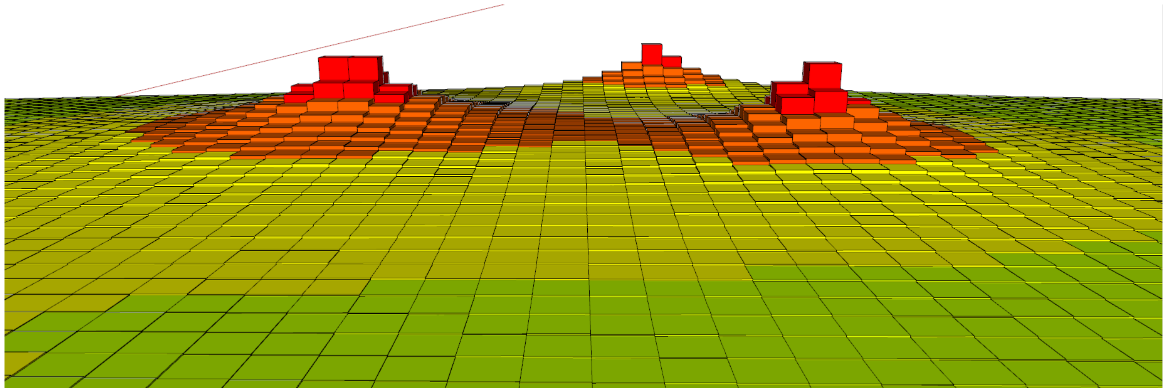


Figure 57 - 2d data plotted in SketchUp (color and height representing intensity)

While graphing 2d data was not a specific goal of this research, this experiment has shown that this methodology is adaptable and applicable for other uses beyond 3d representations increasing the novel uses possible. Building on the personal experience of the author and the 2d method developed above, this technique can be used to assist and organize decommissioning information from building surveys.

In building decommissioning operations, many hundreds or thousands of pieces of information are gathered and require organization for analysis. This information is visually a distraction on a fixed 2d plane, but using the 3d methodology discussed above this technique can create powerful 3d records of buildings where measurements were taken.

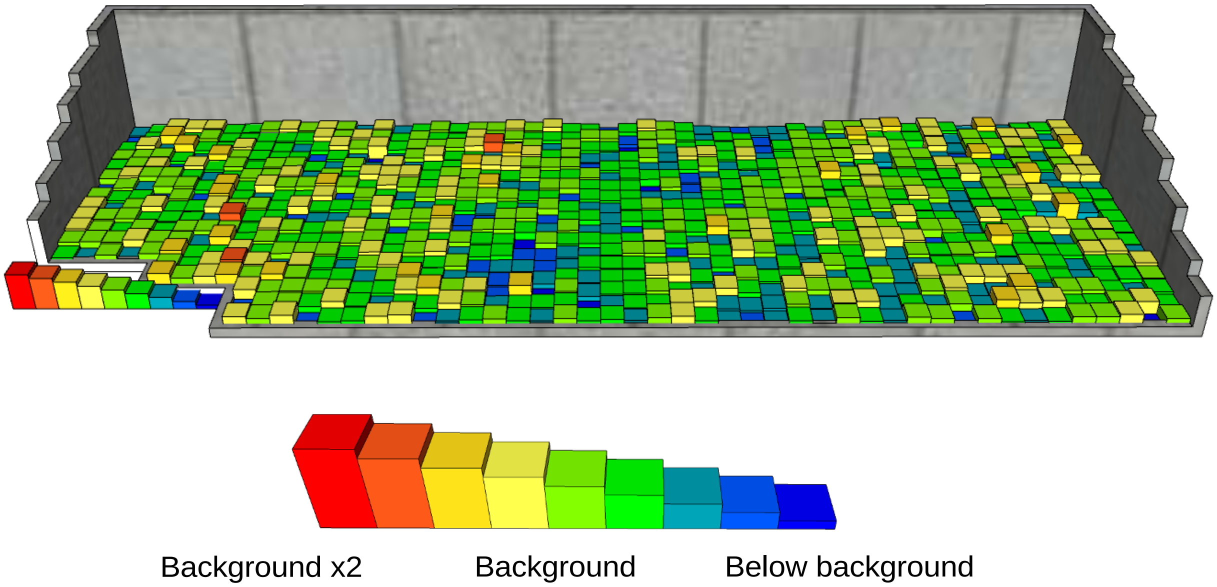


Figure 58 - Illustrative decommissioning data visualization

In Figure 58, illustrative decommissioning data from a field survey within a large building was taken and graphed using the methodology discussed. The height of each element was broken down into nine possible heights representing the full range of results from below the measured background to twice that background measurement. This image can be quickly read and understood. Any areas with unusual readings would be made visually easy to distinguish and therefore easy to identify for future investigations.

8.4 Models into still pictures

Placing a 3d model into a 2d image is a common technique used for product placement, computer graphics and computer generated scenery in movies and other forms of entertainment. Modeling an existing structure and overlaying (with proper scaling) a radiation field model can greatly add to the realism and enhance the understanding others may or may not have developed. In the images that follow, the point source model from Section 7.1 has been inserted into photographs of a common living room to illustrate this effect (Figure 59, Figure 60 and Figure 61). This has been accomplished using the match photo feature in SketchUp [38].

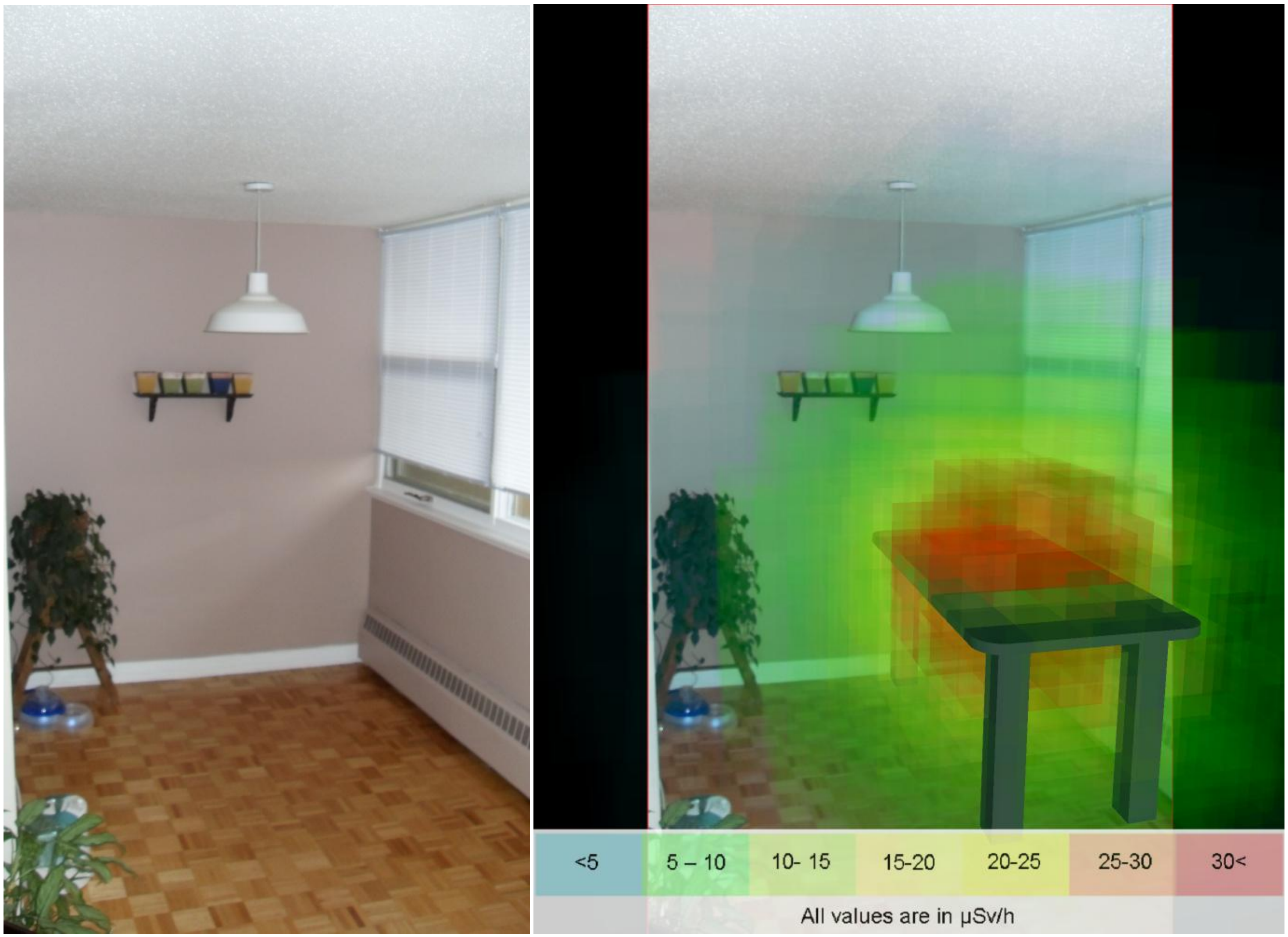


Figure 59 - Original image and image with digitally inserted radiation field (angle 1)

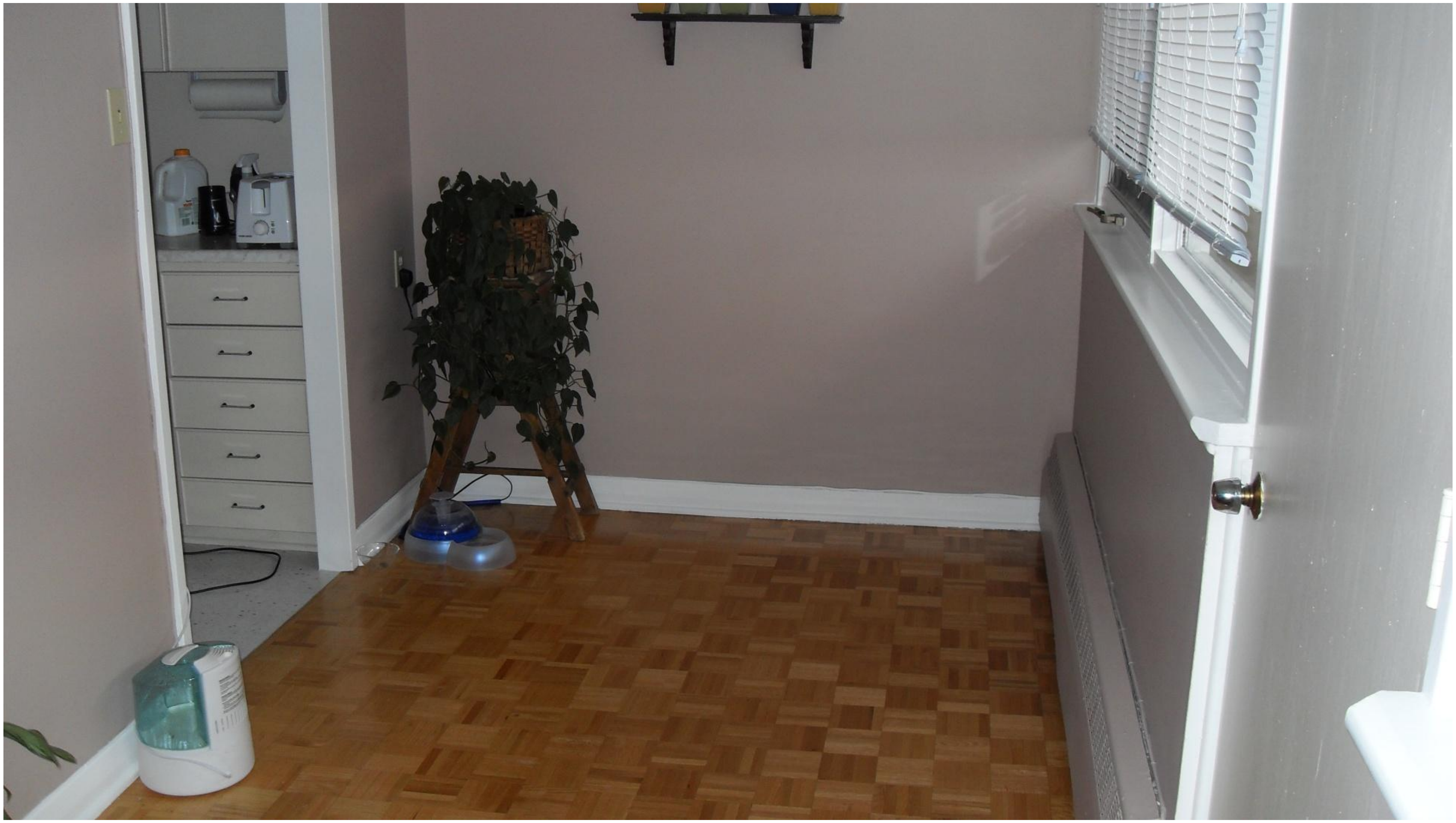


Figure 60 - Original image (angle 2) for photo match demonstration

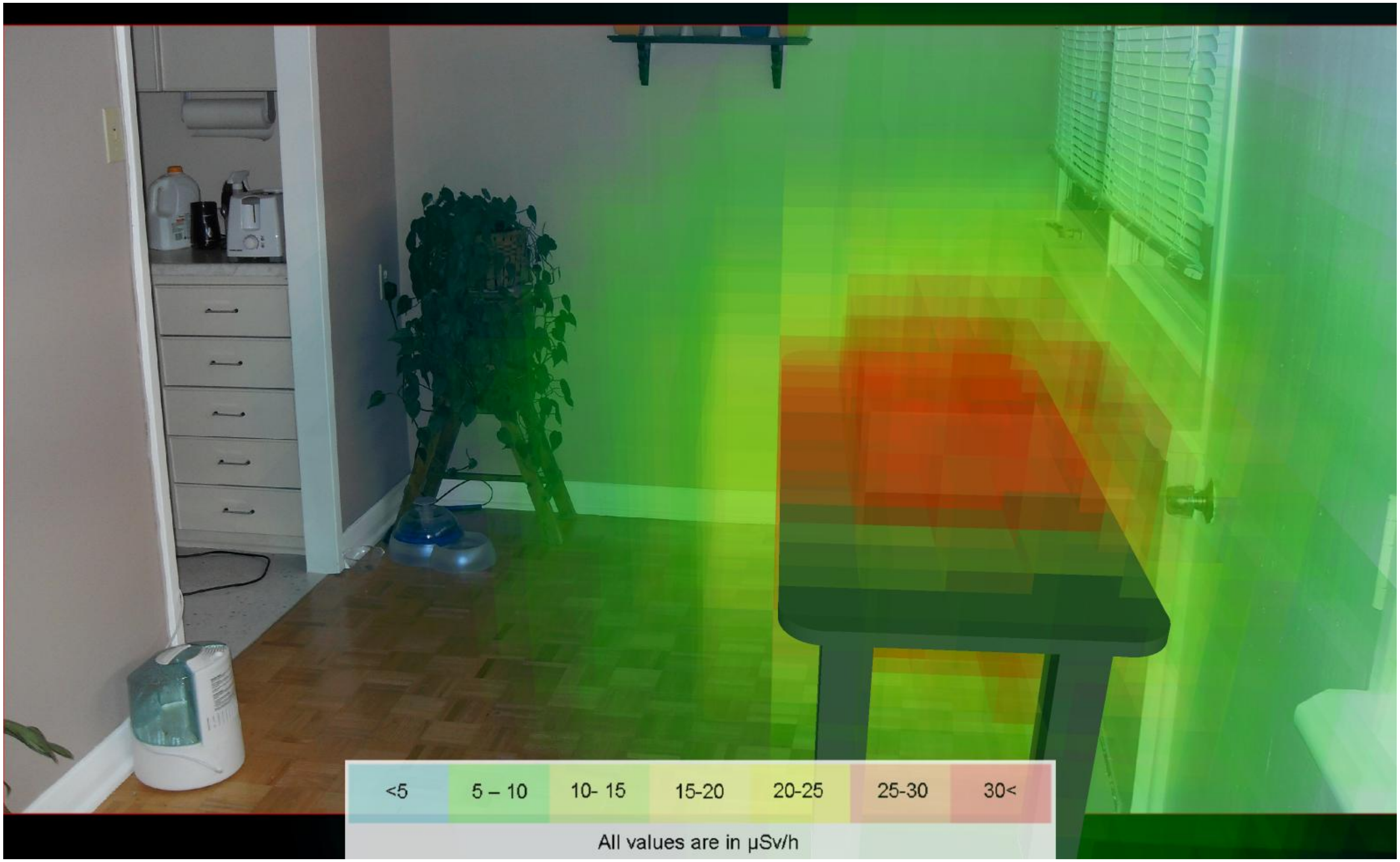


Figure 61 - Original image with digitally inserted radiation field (angle 2)

Inserting radiation models into still images novel and powerful use of this research as it combines intuitive radiation field imagery with real-life pictures to enhance audience understanding.

8.5 Quick scene analysis (MCNP)

As most beginner level users learn very quickly, MCNP is a very powerful tool but also one that requires a dedicated understanding of how to properly use the MCNP executable. Using the mesh tally technique discussed in Section 7.2, and a source term, a very general view of a scene can be created revealing details which the MCNP visual editor may not be able to display or several individual single area tallies may not be able to establish. For more complex scenes this can be further enhanced to reveal areas of interest, increase the users understanding of the movement of particles within a model and visualizing where areas of high degrees of uncertainty are within a simulation.

8.6 Other novel uses

Exporting the models into other programs is a potential very novel and useful use of this research. Consider the following formula:

$$C(x,y,z) = \frac{Q_o}{\pi U \sigma_y \sigma_z} * \exp\left(-\frac{y^2}{2{\sigma_y}^2}\right)\left[\exp\left(-\frac{(z-H)^2}{2{\sigma_z}^2}\right)+\exp\left(-\frac{(z+H)^2}{2{\sigma_z}^2}\right)\right] \quad (2)$$

Where C(x,y,z) is the concentration at an point downwind (from 0,0,0)

Q_0 is the emission rate

σ values represent diffusion along the appropriate axes (x, y, and z) y is the horizontal distance off the plume axis

x is the distance downwind on the plume axis

z is the height above the axis

H height where the emission is taking place

This is a Gaussian Plume Dispersion formula for determining concentration downwind of a point source emission [41]. Using this equation a set of concentrations values in air downwind can be calculated for an environmental dispersion. Using a dose conversion factor for immersion a dose rate value can be calculated from the concentration in air values for specific isotopes. This can be used to create a data file and modeled in SketchUp.

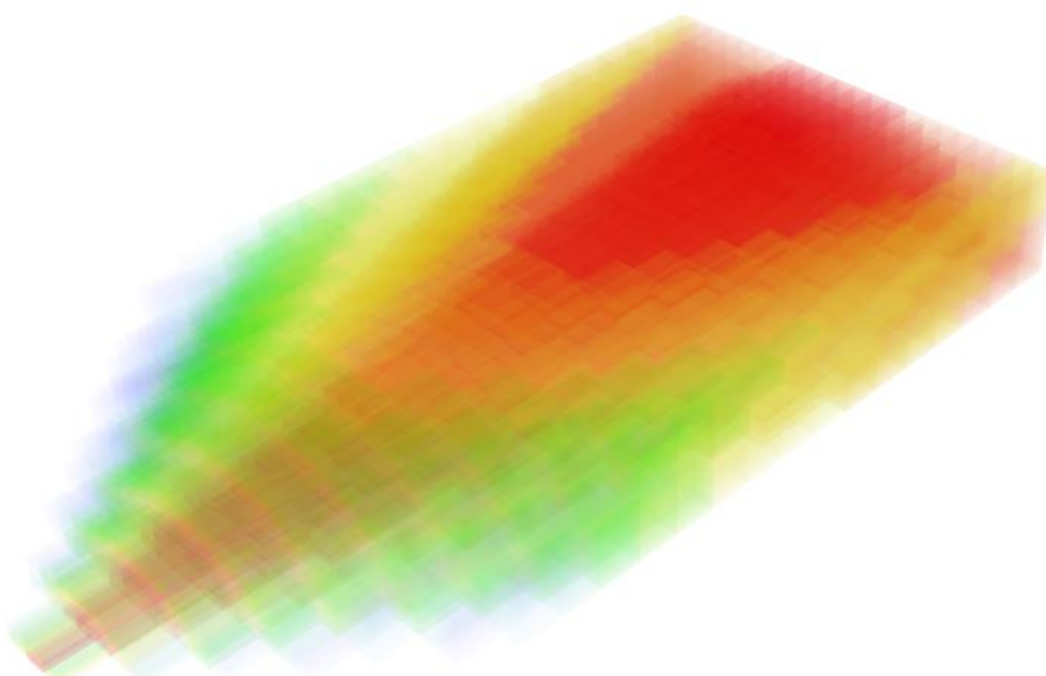


Figure 62 - Simple Gaussian plume dispersion model

Figure 62 shows a resultant plume model built in SketchUp. This model has been limited in extent downwind which is responsible for the clipped edge effect. In this model, rather than having perfectly cubic elements, elements were created that were 25m in height, 200m in length and 100m in width. They were stacked in an array 22x22x22 from ground level (z = 0).

To illustrate a novel use of this model, another piece of software will be introduced. Google Earth is a freely available piece of software provided by Google Inc. It can be used for referencing geographical information, visual searching (by geo-location), and many other uses [42-43]. The key feature that will be shown next is the ability of SketchUp to export a file into a format that Google Earth can read and put into its visualization engine. This is novel because it allows radiation models to be placed into Google Earth detailed scenery which includes 3d buildings, terrain heights and high resolution satellite imagery.

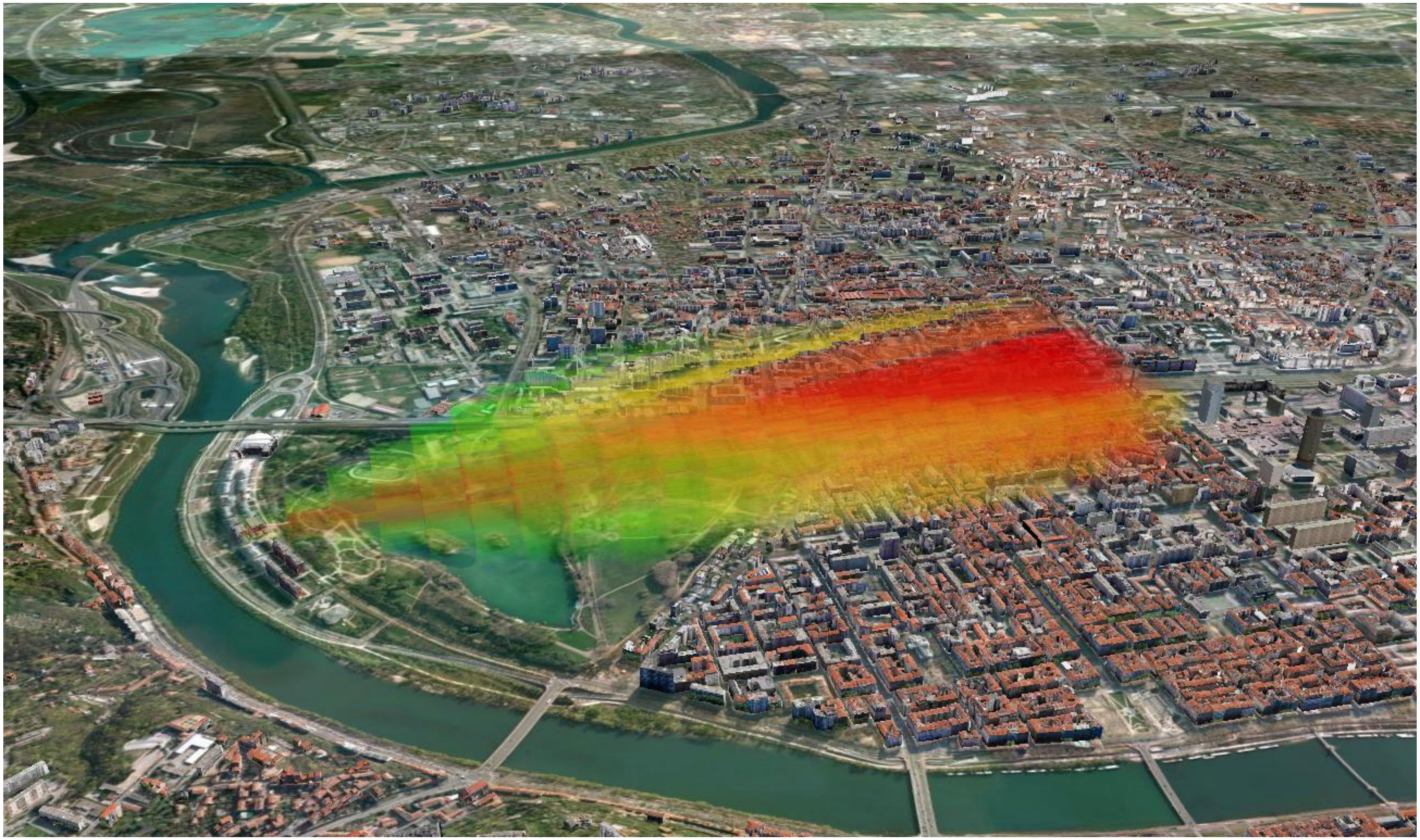


Figure 63 - Gaussian model imported into Google Earth (location Lyon, France)

Figure 63 is an example of such an import into Google Earth. The model which illustrates the differences in dose rates found within a Gaussian Plume, has been imported into Google Earth at a location (Lyon, France) to provide an advanced visual effect. This could be used to teach first responders the radiological consequences following a radioactive dispersal device, (typically a radioactive source mixed with high explosives). Novel uses such as this, can extend usefulness of any radiation models built in SketchUp.

CHAPTER 9: CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

9.1 Conclusions

In this thesis, a generic methodology for the display of 3d radiation fields was developed, discussed and demonstrated. This research began by breaking down the method to develop a radiation field into four distinct phases: Field definition, field construction, interaction and output of video/images. These were analyzed and an approach was formulated which focused on keeping the field definition process separate from the modeling process to maximize potential definition techniques.

The types of issues expected to be encountered with 3d radiation field visualizations were discussed and analyzed. Visibility, user navigation, color scaling, portability of the model format, and limiting the model to reduce overhead were all parts of this discussion. Overall design requirements for the methodology and research were established and eventually shown to have been achieved.

Several 3d modeling programs were reviewed and compared against the development of a standalone program. Through a comparison of the advantages of each approach, a final selection was made to use the Google SketchUp CAD package. This decision was verified and demonstrated through conceptual image creation afterwards to confirm the decision. The primary risk associated with SketchUp (the continuation of the free version being made available), and mitigation measures were developed which relied upon the development of a generic method that will be applicable to most any software package and therefore portable away from SketchUp if the free version ceased to be distributed.

An elemental approach to radiation field modeling was engineered, discussed and developed. Modeling radiation fields via a piece by piece (or point by point) technique was developed. The final construction of these pieces would be made visually into a radiation field through the application of transparency filters to the piece surfaces. This methodology was then applied within SketchUp and a discussion of model extents, resolution and missing data points took place.

Two scenarios were developed and analyzed, and this methodology was shown effective modeling both a mathematically defined field (from a point source) and an MCNP (or Monte Carlo based) field. Several other more novel uses of this technique to display 2d radiation data and to export 3d models into Google Earth were shown.

Based on the ease of use and the freely available nature of the SketchUp program, the methodology provided within this research will become a powerful tool for engineers, scientists and health physicists to create and display radiation fields in 3d with relative ease. These models can be used for training, ALARA planning, engineering or countless other uses. It is the conclusion of the author that eventually, models such as the ones developed within this research will become as common place as a typically scatter plot or bar graph within the nuclear field.

9.2 Future work and applications

The generic methodology and its incorporation into SketchUp show a great deal of future potential. There are many pathways this research could take in the future depending on the applications it is put towards.

- Developed in this work was a method to import fields into SketchUp. Using the Ruby language within SketchUp, there is a potential to create a wide variety of additional functions inside SketchUp that could remove the need for outside programs (such as Microsoft Excel, MCNP or others). Development could include creating methods to assign radioactive material within an environment in SketchUp and have radiation fields automatically construct themselves around those areas by running MCNP directly through the SketchUp interface.
- Future work could create plug-ins for other 3d modeling programs to extend the availability of this technique. Future work could build the radiation fields directly into the open source COLLADA file format using XML via a text editor (as discussed in 3.1.4), removing the need for the construction of plug-ins unique to each program as most will be able to read this file format.
- With a complete methodology now developed to create and display radiation fields unique applications can be developed to take advantage of those fields. Augmented reality applications which use this modeling technique to allow users to interact with these fields is a logical next step. As these fields have been developed in many pieces, the dose to a user as they navigate a radiation field (moving from element to element) could be tracked during ALARA training using augmented reality techniques.
- Alternative methodologies which differ or improve upon the methodology

developed here will always be a logical extension of this work. Alternative methods can be used to contrast against this methodology and develop a recommend set of visualization options which vary depending the scene, scenario, etc.

REFERENCES

1 - Paradis J. G. and Zimmerman M. L. The MIT guide to science and engineering communication. The MIT Press; 2nd edition, Massachusetts. 2002.

2 - Smith, K. L., et al. Handbook of visual communication: theory, methods, and media. Lawrence Erlbaum Associates Inc., New Jersey. 2005.

3 - Cember H. Introduction to health physics. McGraw-Hill, New York. 1996.

4 - Yang M., Radpaint – “A Web-Based Interactive 3D Virtual Radiation Field Application”, University of Florida, Florida, 2002.

5 - VR Dose (*Computer software*), Halden *Virtual Reality Centre*. Institute for Energy Technology, Halden, Norway. 2006.

6 - European Space Agency’s DESIRE (Dose Estimation by Simulation of the International space station Radiation Environment)(*Computer software*), Halden *Virtual Reality Centre*. Institute for Energy Technology, Halden, Norway. 2005.

7 - Spohn, H. Dynamics of charged particles and their radiation field. Cambridge University Press, Cambridge. 2004

8 - Rheingans, P. *Task-Based Color Scale Design.* In Proceedings Applied Image and Pattern Recognition (1999), SPIE pp 35-43.

9 - Tory M., Mouller, T. *Human Factors in Visualization Research*. IEEE Transactions on visualization and computer, VOL. 10, NO. 1, Jan/Feb 2004.

10 - COLLADA Organization official website. The Khronos Group. Sony Computer Entertainment Inc. Website last accessed on 8 Mach 2010: https://collada.org/

11 - Extensible Markup Language official website. World Wide Wed Consortium. Website last accessed 8 March 2010: http://www.w3.org/XML/

12 - SolidWorks 3D CAD official website assembly fileformat. Dassault Systèmes. Website last accessed on 8 March 2010:

http://help.solidworks.com/2010/English/SolidWorks/circuitworks/AllContent/SolidWorks/NonCore/CircuitWorks/c_component_library.html?id=8d6d4280821e441da1ad2f6a976f0c1b#Pg0

13 - Ding A., Zhang D., Xu G. *Training Software Using Virtual-Reality Technology and Pre-Calculated Effective Dose Data (2009)*, Health Physics Vol 96 – Issue 5 pp 594 – 601

14 - VR Dose official website. Institute for Energy Technology (IFE), Norway. Website last accessed on 8 March 2010:

http://www.ife.no/departments/visual_interface_technologies/products/vrdose/index_html/view?set_language=en&cl=en

15 - The DESIRE project official website. Department of Physics at the KTH Royal Institute of Technology. Website last accessed on 8 Marc 2010:

http://gluon.particle.kth.se/desire/

16 - DESIRE RadVis official website. Institute for Energy Technology (IFE), Norway. Website last accessed on 8 March 2010:

http://www2.hrp.no/vr/products/radvis/info/index.html

17 - 3ds Max 2010 (*Computer software*). Autodesk Inc. San Rafael, California, U.S. 2010.

18 - Blender official website. Blender Foundation. Website last accessed on 8 March 2010: http://www.blender.org/

19 - Maya 3D 2011 (*Computer software*). Autodesk Inc. San Rafael, California, USA. 2010.

20 - Google SketchUp 7 (*Computer software*). Google Inc. Menlo Park, California, USA 2010

21 - SolidWorks 3D CAD 2010 (*Computer software)*. Dassault Systèmes. Concord, Massachusetts, USA. 2010.

22 - Google SketchUp industries. Google Inc. Website last accessed on 8 March 2010: http://sketchup.google.com/industries/index.html

23 - Google SketchUp 7 Professional. Google Inc. Website last accessed on 8 March 2010: http://sketchup.google.com/product/gsup.html

24 - Google SketchUp 7 free download website. Google Inc. Website last accessed on 8 February 2010: http://sketchup.google.com/download/index2.html

25 - Google SketchUp Ruby API documentation. Google Inc. Website last accessed on 8 February 2010: http://code.google.com/apis/sketchup/

26 - Murdock K. Google SketchUp & SketchUp Pro Bible. Wiley Publishing Inc. Indianapolis, Indiana. 2009.

27 - Ruby website main page. Ruby Visual Identification Team. Website last accessed on 8 March 2010: http://www.ruby-lang.org/en/

28 - Chopra A. Google SketchUp for Dummies. Wiley Publishing, Inc., Indianapolis, Indiana. 2007.

29 - Roskes B. and deWitt B. The SketchUp Book (5). Conceptual Product Development, Inc. West Caldwell, New Jersey. 2005.

30 - Google SketchUp Training Videos. Google Inc. Website last accessed on 8 February 2010: http://sketchup.google.com/training/videos.html

31 - Möller T. and Haines E. Real-time rendering 2nd Edition. A.K. Peters Ltd. Wellesley, Massachusetts. 2002.

32 - X-5 Monte Carlo Team. Monte Carlo N-Particle Transport Code, Version 5 Volume 1: Overview and Theory. Los Alamos National Laboratory and the University of California. 2007.

33 - Cooper P. Beginning Ruby from Novice to Professional. Springer-Verlag New York. 2007.

34 - Berube D. Practical Ruby Gems. Springer-Verlag, New York. 2007.

35 - Williams J. Rails Solutions: Ruby on Rails Made Easy. Springer-Verlag New York.

2007.

36 - Marick B. Everyday Scripting with Ruby: For Teams, Testers and You. The Pragmatic Bookshelf. Raleigh, North Carolina. 2006.

37 - Google SketchUp Training Videos. Google Inc. Website last accessed on 8 February 2010: http://sketchup.google.com/training/videos.html

38 - Google SketchUp Match Photo instructions. Google Inc. Website last accessed 4 March 2010: http://sketchup.google.com/support/bin/answer.py?hl=en&answer=94921

39 - International Commission on Radiation Protection (1997). Conversion Coefficients for use in Radiological Protection against External Radiation. ICRP Publication 74. 1997.

40 Los Alamos National Laboratory: MCNP Home Page. Diagnostic Applications Grou, Los Alamos National Laboratory. Website last accessed 8 March 2010: http://mcnp-green.lanl.gov/

41 - Cooper J. et al. Radioactive releases in the environment: impact and assessment. John Wiley and Sons Ltd. Sussex, England. 2003.

42 - Google Earth official website. Google Inc. Website last accessed on 8 March 2010: http://earth.google.com/

43 - Google Earth download location. Google Inc. Website last accessed on 8 March 2010: http://earth.google.com/intl/en/download-earth.html

ANNEX A. ILLUSTRATIVE RUBY CODE

Table 8 is example code that can be used to construct a field using the methodology developed in chapter 5. As with any programming effort this is by no means the only method to accomplish this. The code shown below is specifically designed to be easy to understand so that any novice programmer can take it and put it towards their own tasks.

Table 8 - Illustrative Ruby Code

Ruby Code

```
require 'sketchup'


# Show the Ruby Console at startup

Sketchup.send_action "showRubyPanel:"


# Add a menu item to launch the plugin.

UI.menu("PlugIns").add_item("Field construction") {

  UI.messagebox("Field construction is about to take place")


  # Call the field construction method

  field_build

}
```

Ruby Code

def field_build
 	
# Get handles to model and the entities collection it contains 	
model = Sketchup.active_model
entities = model.entities
 	
File.open('Location of text file').each_line {|line|
values = line.split(' ')
x1 = values[0].to_f
y1 = values[1].to_f
z1 = values[2].to_f
color = values[3].to_f
 	
x2 = (x1 + xdimension)
y2 = (y1 + ydimension)
 	
# Create a series of "points", each a 3-item array containing x, y, and z. 	
pt1 = [x1, y1, z1]
pt2 = [x2, y1, z1]

Ruby Code
pt3 = [x2, y2, z1] pt4 = [x1, y2, z1]   #Evaluate the color value to determine the color of element to build #Colors are determined for each material by Red/Green/Blue/ values case color when 8..1000 then face1 = entities.add_face pt1, pt2, pt3, pt4 face1.material = [255,0,0] face1.back_material = [255,0,0] face1.pushpull 1, true when 7..7.99 then face2 = entities.add_face pt1, pt2, pt3, pt4 face2.material = [255,86,25] face2.back_material = [255,86,25] face2.pushpull 1, true when 6..6.99 then face3 = entities.add_face pt1, pt2, pt3, pt4 face3.material = [255,220,25] face3.back_material = [255,220,25] face3.pushpull 1, true

Ruby Code

```
when 5..5.99 then
  face4 = entities.add_face pt1, pt2, pt3, pt4
  face4.material = [254,254,76]
  face4.back_material = [254,254,76]
  face4.pushpull 1, true
when 4..4.99 then
  face5 = entities.add_face pt1, pt2, pt3, pt4
  face5.material = [128,255,0]
  face5.back_material = [128,255,0]
  face5.pushpull 1, true
when 3..3.99 then
  face6 = entities.add_face pt1, pt2, pt3, pt4
  face6.material = [0,255,0]
  face6.back_material = [0,255,0]
  face6.pushpull 1, true
when 2.5..2.99 then
  face7 = entities.add_face pt1, pt2, pt3, pt4
  face7.material = [0,160,177]
  face7.back_material = [0,160,177]
  face7.pushpull 1, true
when 2..2.49 then
```

Ruby Code

<pre>   face8 = entities.add_face pt1, pt2, pt3, pt4
 	
    face8.material = [0,86,255]
 	
    face8.back_material = [0,86,255]
 	
    face8.pushpull 1, true
 	
  else
 	
    face9 = entities.add_face pt1, pt2, pt3, pt4
 	
    face9.material = [0,0,255]
 	
    face9.back_material = [0,0,255]
 	
    face9.pushpull 1, true
 	
end
 	
}
 	
end</pre>

ANNEX B. MCNP MODEL REFERENCE MATERIAL

This annex provides the MCNP code that was used to generate the visualization depicted in 7.2.

Table 9 - MCNP code for 7.2

MCNP Code
Simple Irr w a hole (mesh calculation) 200*25 close and 500x25 far mesh
c
c Source (centered in safe)
c
c Air in safe
1 2 -1.29E-3 +3 -4 +7 -8 +11 -12 imp:p=1.0
c Safe (carbon steel)
2 1 -7.8212 +15 +2 -5 +6 -9 +10 -13
(-3 : +4 : -7 : +8 : -11 : +12) imp:p=1.0
c Air in room
3 2 -1.29E-3 -14 (-2 : +5 : -6 : +9 : -10 : +13)
(-17: +10 : -18 : +19 : -20 : +21)
imp:p=1.0
c External world
4 0 +14 imp:p=0.0
c Hole

MCNP Code
5 2 -1.29E-3 +8 -15 -9 imp:p=1.0
c Block for other hole
6 1 -7.8212 +6 -15 -7 imp:p=1.0
c Ground
7 3 -2.350 +17 -10 +18 -19 +20 -21 imp:p=1.0
c blank line
c
c
c Surface cards
c
1 so 2.0 $source in center of safe
c
2 py -60 $outside safe wall -y
3 py -50 $inside safe wall -y
4 py 50 $inside safe wall +y
5 py 60 $outside safe wall +y
c
6 px -60 $outside safe wall -x
7 px -50 $inside safe wall -x
8 px 50 $inside safe wall +x

MCNP Code

```
 9  px  60      $outside safe wall +x
c
 10  pz  -60     $outside safe wall -z
 11  pz  -50     $inside safe wall  -z
 12  pz  50      $inside safe wall  +z
 13  pz  60      $outside safe wall +z
c
 14  so  5000.0  $outer world cell
 15  cx  20
 17  pz  -500    $Bottom of ground
 18  px  -500
 19  px   500
 20  py  -500
 21  py   500
c   blank line

c     Mode cards: photon transport
c
mode   p
c
c
```

MCNP Code
c c volume card C vol 1.0000E+06 7.0287E+05 5.2349E+11 0 1.2566E+04 1.2566E+04 1.1000E+08 c c Material cards c c carbon steel, dens=7.8212 g/cm3 c air, dens=1.29E-3 g/cm3 c concrete, dens=2.350 g/cm3 c c c c Carbon steel m1 26000.04p 0.99 6000.04p 0.01 c c air m2 7014.04p 4.190E-05 8016.04p 1.130E-05

MCNP Code

```
     18000.04p     2.510E-07
c        concrete (generic mixture)
m3  1000.04p    -1.000E-02
      8000.04p    -5.290E-01
      6000.04p    -1.000E-03
     11000.04p    -1.600E-02
     12000.04p    -2.000E-03
     13000.04p    -3.400E-02
     14000.04p    -3.370E-01
     19000.04p    -1.300E-02
     20000.04p    -4.400E-02
     26000.04p    -1.400E-02
c
c
c     Source cards
c
sc1   Source
c     source is centered at x=0 and goes from y=-0.5 to x=+0.5
c     with a radius between 0 and 1 cm.
sdef   erg=d1 cel=1 pos=d2 axs=0.0 1.0 0.0 rad=d3 ext=d4
c       Photon energies for Co-60
```

MCNP Code
si1 l 1.1732 1.3325 sp1 1.0000 1.0000 c position si2 l 0.0 0.0 0.0 sp2 1 c source radial distribution si3 0. 1. c source axial distribution si4 0.5 c c Tally cards c c Using an F4 tally with no dose function c c Tally cards fmesh4:p origin=-500 -500 -500 imesh= 500 iints= 25 jmesh= 500 jints= 25 kmesh= 500

MCNP Code
kints= 25 fmesh14:p origin=-200 -200 -200 imesh= 200 iints= 25 jmesh= 200 jints= 25 kmesh= 200 kints= 25 c Peripheral Cards c nps 50000000 $ number of particles ctme 60 $check prdmp and nps parameters print 90 175 178 $print tables