

Information Needs Assessment

Community of Interest:
Game Modders

Before



After

Prepared by:
Bruce Norman

Table of Contents

SECTION I: RESEARCH

Table of Contents	Pg. 2
<i>Occupational Definitions</i>	Pg. 3
Multimedia Artists and Animators	Pg. 3
Video Game Designers	Pg. 4
Synopsis of Community	Pg. 5
Titles, Descriptions, and Functions	Pg. 5
<i>Information Needs</i>	
Assumptions	Pg. 7
Knowledge	Pg. 7
Decisions	Pg. 8
<i>Research Methodology</i>	Pg. 8
Insights	Pg. 9

SECTION II: INTERVIEWS

Interview Approach	Pg. 10
Interview Questions	Pg. 10
Questions and Responses Chart	Pg. 10
Subject Matter Experts	Pg. 14
**** ****	Pg. 15
SME Analysis and Response	Pg. 15
Interview Findings	Pg. 19

SECTION III: ANALYSIS AND SUMMARY

Analysis of Work Activities and Information Needs	Pg. 21
Issues and Opportunities	Pg. 22
Reflections	Pg. 23
Concluding Remarks	Pg. 24
Recommendations	Pg. 24

SECTION IV: APPENDICES

Glossary of Terms	Pg. 27
Bibliography	Pg. 29
Revision History	Pg. 30
Community Responses	Pg. 30

SECTION I: RESEARCH

Occupational Definitions

Although there is no actual formal definition for game modders, I have chosen to use the definitions of a Multi-Media Artist and Animator as well as Video Game Designers because many modders use the techniques and skills they learn while modding to pursue these career fields.

27-1014.00 - Multi-Media Artists and Animators

“Create special effects, animation, or other visual images using film, video, computers, or other electronic tools and media for use in products or creations, such as computer games, movies, music videos, and commercials.

Sample of reported job titles:

- *Animator*
- *Creative Director*
- *Art Director,*
- *Artist, Graphic Artist*
- *Graphic Designer*
- *Illustrator*
- *3D Artist*
- *3D Animator*
- *Animation Director*

Tasks:

- *Create two-dimensional and three-dimensional images depicting objects in motion or illustrating a process, using computer animation or modeling programs.*
- *Design complex graphics and animation, using independent judgment, creativity, and computer equipment.*
- *Make objects or characters appear lifelike by manipulating light, color, texture, shadow, and transparency, or manipulating static images to give the illusion of motion.*
- *Apply story development, directing, cinematography, and editing to animation to create storyboards that show the flow of the animation and map out key scenes and characters.*
- *Participate in design and production of multimedia campaigns, handling budgeting and scheduling, and assisting with such responsibilities as production coordination, background design and progress tracking.*
- *Create basic designs, drawings, and illustrations for product labels, cartons, direct mail, or television.*

- *Develop briefings, brochures, multimedia presentations, web pages, promotional products, technical illustrations, and computer artwork for use in products, technical manuals, literature, newsletters and slide shows.*
- *Script, plan, and create animated narrative sequences under tight deadlines, using computer software and hand drawing techniques.*
- *Implement and maintain configuration control systems.*
- *Assemble, typeset, scan and produce digital camera-ready art or film negatives and printer's proofs.”¹*

15-1099.13 - Video Game Designers

“Design core features of video games. Specify innovative game and role-play mechanics, story lines, and character biographies. Create and maintain design documentation. Guide and collaborate with production staff to produce games as designed.

This title represents an occupation for which data collection is currently underway.

Tasks:

- *Create core game features including storylines, role-play mechanics, and character biographies for a new video game or game franchise.*
- *Document all aspects of formal game design, using mock-up screenshots, sample menu layouts, game play flowcharts, and other graphical devices.*
- *Collaborate with artists to achieve appropriate visual style.*
- *Conduct regular design reviews throughout the game development process.*
- *Consult with multiple stakeholders to define requirements and implement online features.*
- *Create and manage documentation, production schedules, prototyping goals, and communication plans in collaboration with production staff.*
- *Create game play prototypes for presentation to creative and technical staff and management.*
- *Determine supplementary virtual features, such as currency, item catalog, menu design, and audio direction.*
- *Develop and maintain design level documentation, including mechanics, guidelines, and mission outlines.*
- *Devise missions, challenges, or puzzles to be encountered in game play.”²*

¹ As defined by O*Net Online created for the U.S. Department of Labor, Employment & Training Administration
<http://online.onetcenter.org/link/summary/27-1014.00>

² As defined by O*Net Online created for the U.S. Department of Labor, Employment & Training Administration
<http://online.onetcenter.org/link/summary/15-1099.13>

Synopsis of Community

Modders are individuals or teams of people that create modifications (also known as mods) to software that can include the introduction of new functions, objects, or information not originally intended by the designer or developer. In the case of game modders, this can include new sounds, models, textures, animations, scripts, and behaviors that allow the modder to alter the game world to suit their own vision or desires. Such modifications are also referred to as customer created content and are often freely shared with others on the web. The most common form of mods are those that introduce new models and/or textures to the game world, thus the skills and knowledge needed to create them will be the focus of my research.

Modders will often create content that the developers had never considered or were too limited by scheduling or funding to include in the game. The content created by modders is often superior in quality than what the designers themselves created because modders are not restrained by those limitations. For example, in the game community that I frequent, horses were never introduced into the game world by the developer. A modder created a mod that not only allowed you to buy and ride a horse, it also allowed you to own a ranch where you could breed your own horses and sell them. This was content that was never envisioned or intended for the game, but by creating new models, textures, animations, and scripts, the modder has given a new depth to the game by making his work freely available to the community he is part of. Similar mods that create new races, creatures, weapons, clothes, homes, cities, and quests are regularly added to upload sites on the internet that game players can download to enhance their own gaming experience.

Titles, Descriptions, and Functions

The goal of many modders is to enhance the re-playability of a game and how immersed a player becomes in the role he assumes inside the game world. Modders generally fall into a few different categories:

- Modelers are responsible for the creation of three dimensional objects, buildings, characters, and miscellaneous items in the game world. They create the physical representation of everything inside the game. Modelers also UVW map and unwrap the models to create two dimensional images that are passed on to a texture artist. Modelers are usually required to limit the amount of detail in an object so that it doesn't tax a computer's processor when rendering and animating it in the game. It is their goal to create as accurate a representation of the object as possible with a minimal number of polygons or faces.
- Texture artists add color and detail to the texture map using a painting-capable program such as Photoshop. These details represent the features of an object that are not generally created in the model itself, such as material textures, facial features, skin, etc. It would be impossible for a computer to render billions of individual blades of grass modeled and placed in a game, but by painting the grass in a texture map that has been derived from a simple surface model, the illusion of a grassy field can be created.
- Texture artists often have to follow a certain artistic style in the detail of their artwork to blend in with the game environment. The amount of detail they can create is often limited by how large a texture map file can be. The larger the file,

the more of a computer's resources it uses when rendering the model and texture in the game.

- Animators are responsible for creating the motion sequences of a particular object, creature, or character. They do this by creating axial pivot and rotation points for the model and use distance and time to define how each part of the model moves based on those points. Animators will usually create a set of animations for each type of object, creature, or character. These may include mundane tasks such as walking, sitting, or eating to more complex motions such as riding a horse or choreographed attack sequences. With modern games, the goal is to create movement that is as life-like as possible.
- Scripters are programmers who can add scripts that affect a sequence of behaviors or to add new dialog and quests. Behaviors often define how an object interacts with the world around it based upon the programming defined in the game engine. For example, many games use additional characters that will follow and assist the main character (the player) during the course of the game. The Artificial Intelligence of these characters, how they move, and how they follow and assist the player, is often defined by scripts.
- Editors are people who are familiar with using a game's native editing tool that allows sounds, scripts, objects, animations, and textures to be imported into the game world. Editing is often difficult because it may be hard to visualize how the placement of objects in the editor will actually look inside the game world. Editing tools often use an overhead view of cells within the game world where items are placed, but the in-game play is often from a first or third person perspective. Achieving a balance of space and the number of items in an area is often difficult to accomplish. A greater concentration of items affects how fast the computer can render these items, which in turn can affect the speed of the game play.

While these are all considered individual skill sets, it does not mean that an individual modder is not capable of performing multiple, or even all, of these tasks. Many modders are professional game developers whose love for a particular game inspires them to continue adding content long after the expected lifecycle of the game has ended.

Information Needs

Assumptions

Modders need to be familiar with:

- Instant messaging or chat software for collaboration and file sharing
- The native game editor to import and place items in the game world
- Modeling software to create new models and animations for the game
- Painting or image manipulation software such as Photoshop to create or alter textures
- UVW mapping and unwrapping of models
- Third party programs developed for specific game-related editing or enhancement
- Archive compression software to package the mod for uploading to host sites
- Programming software to create scripts and dialogue
- Game design terminology and processes
- File formats that are acceptable or convertible for use in-game
- Community specific etiquette and rules
- File organization and proper placement of mod data files in appropriate folders
- Basic concepts of anatomy, architecture, sketching, and design
- Where to find free-use resources and tutorials
- Their own limitations and skill sets
- The skill sets of their teammates
- How to coordinate and work with groups of people
- Game lore and culture as originally envisioned or created by the developers
- How to properly plan and execute a project
- Game mechanics and the limitations imposed by the game engine
- Game engine bugs or issues that affect modding
- Plot development for quests and dialogue
- How to assign textures to models
- How to make 3D models using reference drawings or pictures
- Backing up their work to more than one location
- Iterative development and testing
- Debugging and addressing errors within the mod
- Uploading files to a host site
- How to prioritize work in the Construction Set
- How to properly scale models for in-game placement
- Time management

Knowledge

Most modders have no formal education in creating game content. A large number of them are self-taught, younger than 18 years old, and use the skills they gain while modding to focus their college studies towards game design and development careers. Other modders range in age, experience, and education. Some are game development professionals, often part of the design teams that created the games that they mod for.

There is no single demographic to define those who mod for games. They include housewives, retirees, doctors, bartenders, etc., from many different countries and cultures.

Although it is preferable for a game designer to possess some form of formal education, it is not necessary if they possess experience and demonstrable skill. Many colleges now offer Bachelor's and Associate's Degrees as well as certificates for a number of game development fields and skills.

A basic knowledge of a number of graphics, modeling, and programming software is required to create content to be placed into the game world. Additional software knowledge regarding text editing and archive compression are also considered by many modders as a necessity.

Decisions

There are numerous decisions to be made when planning and executing a mod or creating mod content. These can be relatively simple decisions such as color or item placement in the game to something significantly more complex, such as how to write the code for a script to change an object's behavior. While these decisions may be relatively easy for an individual who is creating a mod since they are aware of their own skills and limitations, this becomes more difficult when teams create mods.

Most teams will identify a leader or leaders to organize the members and assign tasks, maintain a schedule to determine progress, prevent scope creep, and to serve as the hub of communications between the members. The teams will need to make decisions based on individual member's skills and experience and how the modding responsibilities will be divided amongst each team member.

Research Methodology

A Google search for mods, modders, and modding was all I really needed to get started. I chose game modders as my Community of Interest because it is something that I have just begun to do myself. A lot of the research that I have done prior to this report has helped me to understand what modding involves, but I have always focused on one particular game that I enjoy. Doing the research for this project forced me to go out and look at other modding communities that have grown up around specific games to see how they approach game modding. There are a number of internet sites that deal specifically with modding and modding communities. Each of these communities often develop their own tutorials that are game specific and detail how to add content to the game world using a variety of editing tools and third party programs. A lot of the information about modding a particular game is often referred to as "tribal" knowledge and is based upon what previous modders have learned and passed on to their community.

Learning the basics of modding usually has to begin with knowledge of the game's editing tool and what file formats it accepts. Without the editor it is usually impossible to add content to the game. I have noticed that some of the other communities I have

visited seem to make it unnecessarily difficult to place content into the game world by lacking developer supported or created tools for the modders to use. Many times these tools are created by the modders themselves and shared with others. Tutorials for using these game editing tools can often be found on community forum sites, on the developer's website, or included with the game.

While I used a number of online resources for my information on this Community of Interest, I also have used the knowledge and insight of a number of skilled modders from the community forum that I frequent. There are some communities that do not share their knowledge of modding skills and techniques, but the forum that I joined has a number of skilled modelers, animator, texturers, and scripters who are eager to share their experience and tricks that they have learned. This has made my research considerably easier, as I regularly communicate with a number of them online and they are very supportive of my first attempts in modding.

The biggest problem I have had so far is in finding "official" sources of information on modding, as this is usually viewed as a hobby amongst a loosely knit community of individuals with varying degrees of design and development skills. There is no "official" definition of game modding beyond what the internet community has decided it should mean. However, many modders possess skills that have real-world counterparts in the game design industry, so I have chosen those job titles as they generally define the same skill sets many modders possess. Although I did consult with a librarian at the library around the corner from my house, all she could do was point me to a few books on game design that really had nothing to do with game modding.

Insights

While there is an incredible amount of information about modding for numerous games online, what I have noticed is that there is a lack of "legitimacy" concerning the practice of modding. Many professional-level research resources have no information about the subject and its related skills and practices at all, although game and software modding has existed for well over a decade. Modding is a widespread phenomenon, spanning multiple countries, disciplines, and demographics. I found it rather frustrating that I had to rely on information gathered by the modding community itself in order to complete my research. While I do not feel that this makes the information any less trustworthy, I do feel that it indicates that it may be a long time before game modding is taken seriously as either a hobby or as an alternative avenue to gaining skill and experience as a segue into a game development career.

SECTION II: INTERVIEWS

Interview Approach

My objective is to use an existing modding community to answer questions that I pose regarding common practices, processes, and information requirements. To this end, I have started my interview process by creating a thread on the modding forum that I frequent to get the community member's response to the questions I have listed here. As the project progresses, I will add more questions to explore their responses and delete those that do not contribute to the information I am seeking. I will also seek confirmation of these common practices by interviewing a specific modder with experience in producing mods. Because this modder lives in a different country, I will interview him using Windows Messenger Live. I plan to question him regarding the areas of modding that he is most proficient with and to explore his information sources and modding practices.

Interview Questions

Based on the interview approach, I have identified a number of questions that will help me to gather common information and practices used by an existing modding community. The following chart graphs the responses (see *Appendices, Community Responses*) that I received to the questions I posted on the community forum. I will use this information to seek confirmation from my subject matter expert in a subsequent interview.

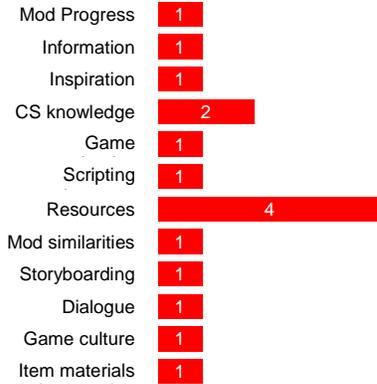
Questions and Responses Chart

1) How do you decide what kind of mod you will do either individually or as a member of a modding team?

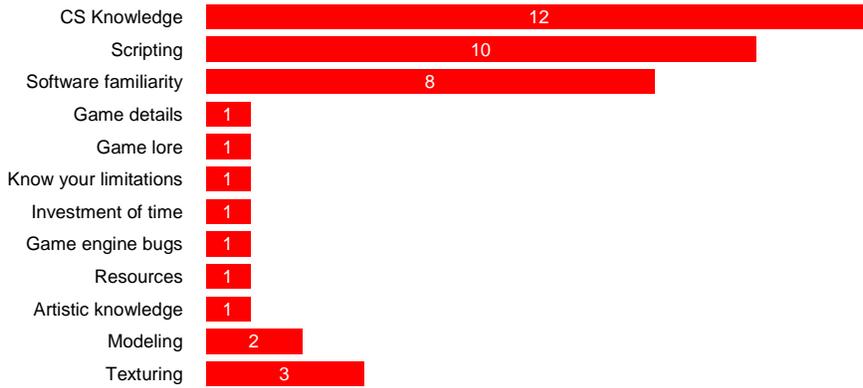


2) What kind of information do you need when doing a mod?

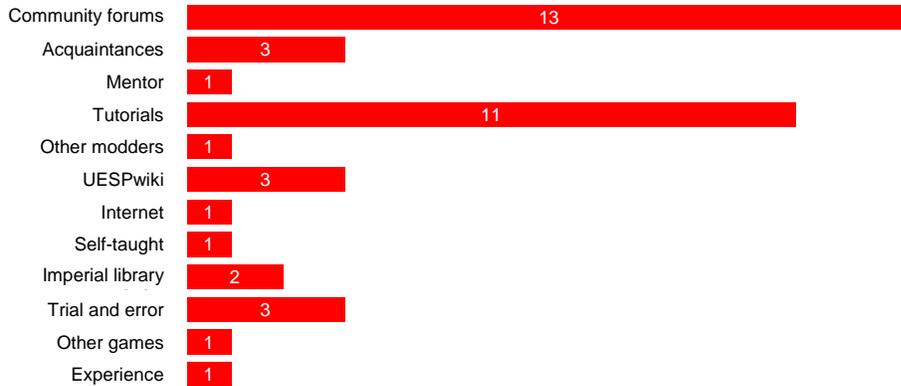




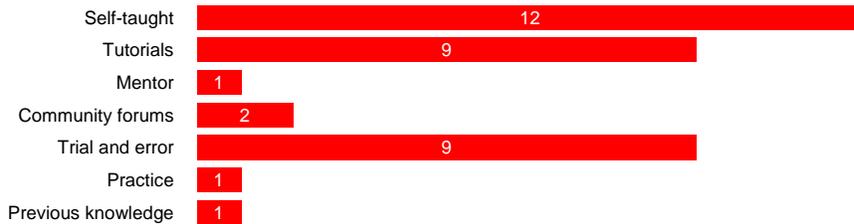
3) What kind of knowledge do you need when doing a mod?



4) Where do you find this information?

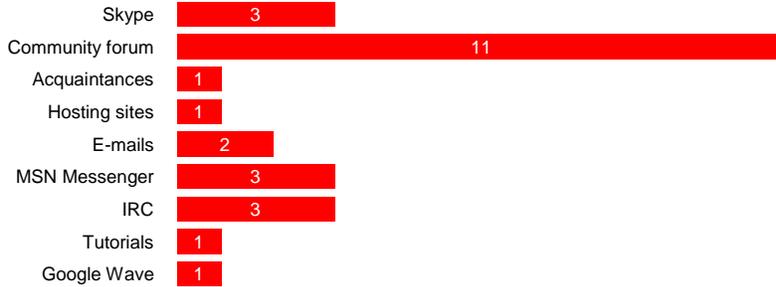


5) How did you learn how to mod?

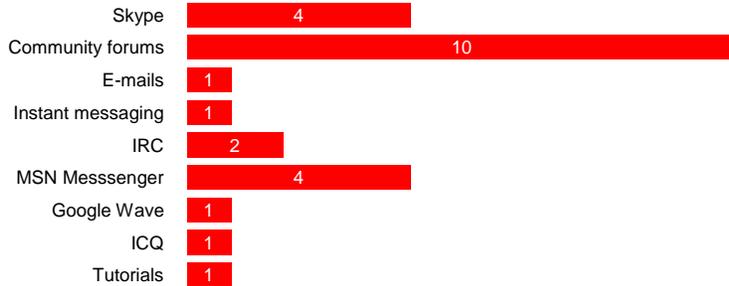


6) How do you share information and ideas with other mod team members?





7) If someone is 'mentoring' you, or vice versa, how do you communicate with them to share information?



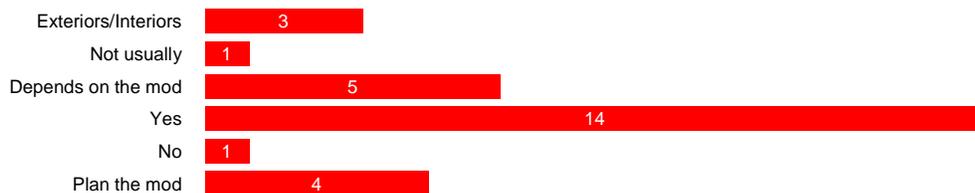
8) Do you find it helpful to know more than one aspect of modding, such as modeling?

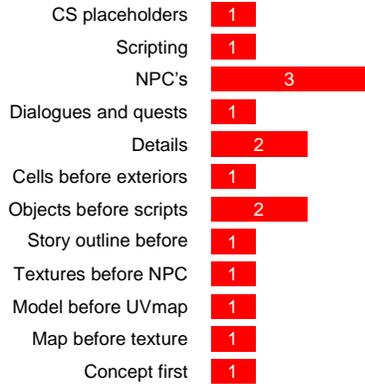


9) How do you approach the work involved in making a mod, either alone or as a member of a team?



10) Do certain parts of the mod have to be done before others?





11) Can you describe the normal processes you go through to create a mod?

Refer to Appendices (Community responses) for details. A typical process follows these steps:

- Define the mod scope
- Break it into manageable parts
- Gather resources/create concepts
- Define scripting/dialogue/quests
- Create new resources, etc
- Place items in CS
- Test the mod
- Bug fix
- Write the read me
- Release the mod
- Address errors

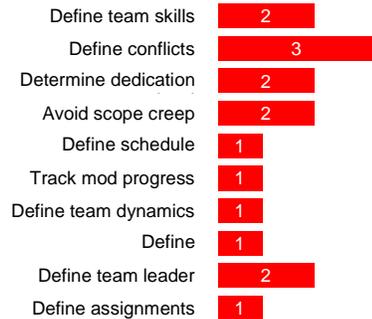
12) What kind of software knowledge do you need to create a mod?



13) What are some of the things you should know before you begin a mod?

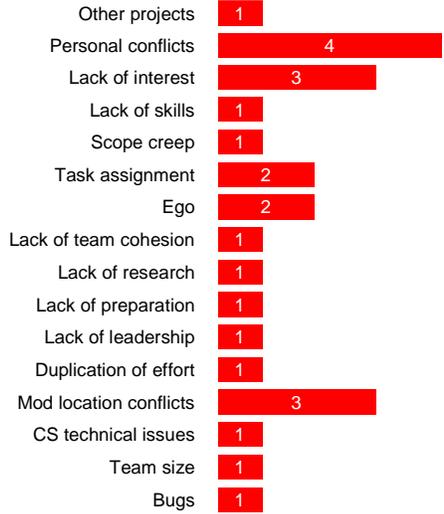


14) What are some things you need to consider when working with a team in a group project?

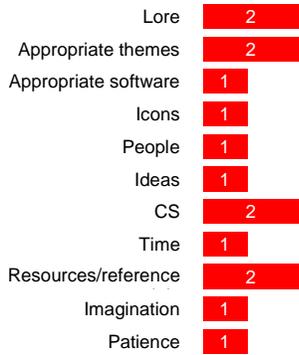


Resolving disputes **1**

15) What kinds of conflicts or problems are common when creating a mod, either as an individual or in a team?



16) What kind of resources do you need to create a mod?



17) When creating mods, what programs do you most commonly use?

This is a list of all programs mentioned. See Appendices for individual responses.

3DSMax 7-zip Audacity Bink Blender BSA browser Cdex Cinema 4D Corel Painter 10 Crimson Editor CS
 DDS converter Deep Paint 3D DirectX Redistro Enchanted editor Excel Fiddler 2 Firefox Fraps GMST vaccine Google HxD
 IDA Pro Irfanview MCP Mesh Generator MGE MLOX Morrowind AnimKit MWEdit Nif Stripper Nifskope Notepad OlllyDBG
 Paint.net Photoshop Pidgin Programmer's Notepad 2 PSPad Python RADTools Reorder mods Sculptris Skype Soundbooth
 Strawberry Perl Tes3cmd Tesame TESDTK TESPCD TESTool The Gimp Trillian UltraEdit Visual Studio 2008 Windows
 Grep Windows Movie Maker WinMerge WinRAR WTV wxPython Wyre Mash ZBrush

Subject Matter Experts (SME's)

To get a broader interpretation of common modding practices, I have posted a thread in the game modding forum that I frequent to get a community-wide response to my questions. These SME's will allow me to identify common practices, information, and knowledge used by game modders in a particular community. However, I will also rely on information gathered from directly interviewing at least one specific modder who has assisted me in my own learning process.

The subject matter expert will review the responses provided by the community and add his own input to those answers. He will also add his own responses to the questions so that I can analyze how well the community responses represent the practices and information requirements of the community as a whole. I will also use my own knowledge of this community of interest when doing my analysis and recommendations.

**** *****

My subject matter expert is **** ***** , a bartender who spends a great deal of his spare time modding. I have chosen him for his ability to create mod content and place it in game using the native Construction Set editor. He is very knowledgeable with making and taking modder-created content and placing it in a game world environment. He is also capable of texturing, physiqueing and rigging models, and animating. **** is a self-taught modder who likes making mods for the enjoyment of others.

SME Analysis and Response

1. How do you decide what kind of mod you will do either individually or as a member of a modding team?

I think there are more people out there that do requests than the chart shows, but I can agree that most modders create what they want for the game.

I usually collaborate with a lot of people, although on separate projects, and do a lot of requests as well. But I do make my own stuff for my game when I want it. Actually, right now I'm doing a total conversion, so I will be focusing on that the most as it is a huge amount of work.

2. What kind of information do you need when doing a mod?

Those answers are all over the place, but I guess that makes sense because it all depends on the mod you are making. Sometimes all you need to know is the name of the model and texture you want to change in the game. That means you need to know your way around the CS. Sometimes you need visual references for creating models and textures. If you are a lore purist, you need to be familiar with the game lore that Bethesda (the game developer) created.

I don't see the need to strictly adhere to the lore. It puts limits on what you can create. So, again, it all depends on the mod. I usually use visual resources I find online and every once in a while I need to watch a tutorial or two to figure out how to do something. It's nice to plan beforehand what you are going to put into the mod, so concept art is a good thing to have. And if you are collaborating with someone, you need to know what their skills and abilities are as well.

3. What kind of knowledge do you need when doing a mod?

I definitely agree with needing to know the CS. You can't put anything into the game without using the game editor. A few modders only make new models and textures as resources for other modders to put into the game, so I guess they don't need to know it. As for scripting and software familiarity, again, I can agree with that if that is

what they are doing for the mod and if they need specific software for certain things, such as making models and textures.

I use the CS a lot and I do a lot of modeling and texturing. Occasionally I'll rig a model for animation, but you can't create new animations for this game without replacing an existing one. It's hard-coded into the game. So you better make sure it works for every character, creature, etc. that uses it.

4. Where do you find this information?

Some of those answers are redundant. Getting your information from the community forums indicates you are getting it from other modders, so you could probably combine them. But I agree that most people use the forums and tutorials for their information. There really aren't a lot of other places to go if you are looking for information specific to the game outside of the developer forums and a handful of fan sites. But I guess it depends on the type of information you need.

I usually get my information from the forums or from other modders that I know. I mentor a few people on how to model and texture, and I'm being mentored on how to write scripts by someone I am teaching how to model. It doesn't have to be a one-sided relationship. And if I don't know or someone I know doesn't know, I'll find some tutorials online for whatever I need to know.

5. How did you learn how to mod?

I definitely agree with the top three: self-taught, tutorials, and trial and error. We have a fair amount of information about modding on the forums and through various websites, but learning it yourself really is the most common way to learn it. If you can find someone who knows what they are doing already and they don't mind being harassed, you can build a pretty good relationship with them that evolves into a "mentor/student" type relationship.

I'm pretty much self-taught, but I did it watching tutorials, asking lots of questions, and experimenting. A lot of the knowledge we have now about certain techniques, especially in texturing, were not possible when people first started modding. There were a few trailblazers back at the beginning that spent the time to figure out how everything worked and then experimented to make new things work, like gloss maps.

6. How do you share information and ideas with other mod team members?

I don't think people read that question close enough, because most teams will not use the forums to share ideas and information. That's not really what the forums are for. Teams will usually use Skype or Messenger or some other form of instant messaging.

You know what I use since you harass me every day. I usually use MSN Messenger. I used to use Skype, and I guess I still do on occasion. It depends on who I am chatting with and what they prefer.

7. If someone is 'mentoring' you, or vice versa, how do you communicate with them to share information?

Again, I find the community forum answer odd. It takes a lot of time to post to the forum, wait for a response, and repeat ad infinitum. If it's a general question, then, yeah, I agree with that. If someone is asking for someone else to "walk-thru" a process with them, then I don't think that would work well. It's not really what the forum is for.

I use MSN Messenger all the time. It makes it a lot easier to have a conversation with someone when you are trying to show them something. Plus you can drag and drop files and pictures to share with them.

8. Do you find it helpful to know more than one aspect of modding, such as modeling?

The community got that one right. But I guess someone is always going to blow the curve with a "no". Who said no?

It's definitely better to know more than one aspect of modding. I think it's pretty hard to know them all. I don't know of anyone who can do everything. I can model, texture, rig, animate, use the CS, and I'm learning how to script. But there is still a lot I don't know. Hopefully, by the time I'm done with my TC (total conversion) I'll have most of this stuff figured out. But I still won't be able to create my own music like some people can.

9. How do you approach the work involved in making a mod, either alone or as a member of a team?

Those answers are all over the place too. It all depends on the mod, since some are so simple they can be done in five minutes. But if you plan on doing a bigger mod, especially as a team, you definitely need a plan. You also need a committed team, or at least a committed core team. And if you are working with a team, you need to know what they can do and how the work for the mod can be split up between the team members.

I really should plan better. I have a tendency to just jump right in until I start running into problems. Then I have to stop and regroup. I don't usually work with teams. I'll make stuff for other people, like new models and textures, and I'll mentor people so they can do things for themselves. I should find some others who might be interested in helping me with my TC, but it's always better to have something to show before going to ask others to get involved or to ask for help. I don't think half the people out there on the forums realize how much work goes into making something when they put in a request.

10. Do certain parts of the mod have to be done before others?

They got that one right, too. Yes. It's kind of hard to place an item like a building into the game before you have created the landscape itself. You don't need all the plants and stuff, but you do need the ground. And you need the exterior of a building placed before you can place the interiors and door markers, etc. Again, it all depends on the mod. A lot of mods just place a single item, like a weapon, into the game. There's really not a lot to that. All you really need to know is which cell you want to put it in.

With the TC that I am doing, the first thing I had to do was create the land. Now that I have that done, I can place the trees, villages, NPC's (non-player characters), etc. It takes a lot of time and can be very monotonous. You need all of these items in place before you can attach scripts to them.

11. Can you describe the normal processes you go through to create a mod?

I'm assuming that this list is something you pulled from the responses. That's a pretty good idea of the basic processes. You would be there all day if you tried to list every process, and every mod is different.

What you have there is pretty much how I do things, but then you learned from me so you would know that already. Sometimes there are more steps, sometimes less.

12. What kind of software knowledge do you need to create a mod?

There were only five responses for this question? At least they got it right. Unless you are only creating resources for other people to use, you definitely need to know the CS.

If you are creating new content, like I do, you need some sort of modeling software knowledge. It doesn't matter which one, as long as it has a format that can be used by the Nif Exporter to create a .nif model to import into the game. You also need some sort of image editing software like Photoshop.

13. What are some of the things you should know before you begin a mod?

I can agree with needing to define the mod as the most important. You need to know what you want to do with it and what will and won't be included in it. I've seen a lot of team mods fail because they keep adding to it and people lose interest.

I usually like to look at what has already been modded in-game before I try to add something. There are so many mods out there that it's hard to find an area to place something that won't conflict with another mod. There are a lot of mods that people always play in their games, so it's best to avoid placing content in the areas they affect. At least if you want people to use your mod.

14. What are some things you need to consider when working with a team in a group project?

A lot of those are good answers, and most of them should be addressed. You need to avoid conflict between modders, know their skills and dedication, and you should assign someone to coordinate the team and divide the tasks. A lot of times people will volunteer to do certain things and the person who starts the mod is usually the one calling the shots.

I'm amazed more people didn't say communication, because it's very important to keep in touch to determine the progress of the mod. I know of one mod where a person who volunteered to do the interiors wasn't heard from in over 6 months, so the team assumed he dropped out and they got someone else to do the work. But then he showed up with a huge amount of stuff done. At the very least, proper communication prevents duplication of the work the team is doing.

15. What kinds of conflicts or problems are common when creating a mod, either as an individual or in a team?

I can agree with most of the answers given on this question. I'm surprised personal conflicts had the most answers, since most people seem to get along fairly well in this community. Lack of interest and mod location conflicts are definitely good ones too.

I'm surprised more people didn't say scope creep. That's my problem, but then you know that. I started out building a village and it progressed to a total conversion. And we all know the success rates for those. How many have actually been completed in 8 years? One?

16. What kind of resources do you need to create a mod?

Some of those answers are questionable. You don't really need lore. I've never made a mod that was lore "correct". I've never even read any of it outside the game. It doesn't need to be an "appropriate" theme, either. You just can't post inappropriate content or links to such on the company owned forums. But it doesn't mean you can't make them, and other sites will host them. But the CS, time, and patience are definitely something you need if you plan on making a mod.

I spend a lot of my free time on the computer making resources for mods, either my own or for others. I usually use visual reference materials when I make models, so something someone else has drawn or created concept art for. I'm terrible at it myself. I guess you also need access to the programs used to create whatever it is you are doing, whether that is modeling, image editing, or programming/text editing software.

17. When creating mods, what programs do you most commonly use?

I use Blender for modeling and animation, Gimp for texturing, and Nifskope to place the textures on the models before placing them into the game. I use Scenelmmse to view the textures on the models. And obviously, I use the CS to create the mod. Out of that list I also use 7-Zip, Messenger, Google, MGE, Notepad, and Skype. Everything I use is free, open-source programs for creating mod content. I'd like to learn 3DSMax and Photoshop, but can't afford them.

Interview Findings

I had a number of responses to the questions I posted online from the Morrowind modding community. I feel that those who responded create an accurate representation of the community itself, with varying degrees of experience, skills, and knowledge. They also strongly represent the diversity of common modding practices, interests, preferences, and demographics. These subject matter experts, although not directly interviewed, allowed me to identify common information needs that the community has, including resources, software, management processes, "production" processes, and group communications.

While my subject matter expert agreed with many of the commonalities that I found in reviewing the responses from the community forum thread, there were a number of

points that he did not necessarily agree with. Due to my own exposure to the modding community, I will attest that this is to be expected as there is no defined process or preferred methods for creating a mod. The scope of the information available and the variety of resources and software that can be used allows each modder to use materials and methods that they are most familiar or comfortable with.

While each modder has their own preferred methods, order of operations, and even choices of software, there are some common practices that each modder can agree upon as being both necessary and, in some cases, absolute. One of the things most modders could agree upon was that knowledge of the game's native Construction Set editor is necessary. Both the game engine and the Construction Set editor have built-in limitations that all modders must conform to. I questioned my subject matter expert about these limits in some of our casual chats.

Hard coded limitations in the game engine, such as the ability to add new animations without replacing existing ones, create obstacles for many modding projects. There are also a number of bugs in the engine that cause problems with duplication of objects in game and the so-called "72hour bug" that causes an arbitrary loss of in-game inventory items if the game is not saved on a regular basis.

One group of modders are currently attempting to address these problems by writing a new game engine that opens new venues and opportunities for modding this particular game. This is a significant deviation from normal modding, which until now has been limited to changing in-game content. If successful, this modding team will have changed the actual platform upon which the game content is played. I have not found mention, at this time, of any other modding community attempting to rewrite a game engine, as the amount of time, effort, and programming knowledge required is usually only available to game development companies.

The Construction Set editor also sets a number of limitations that the modder must understand. The only types of models that are capable of being imported into the game are referred to as nifs due to their format extension: i.e. .nif. One modding group created an exporter/importer that allows modders to convert from a number of 3D model formats to a .nif file, thus greatly expanding the number of modeling programs that can be used. A similar limitation exists with textures, which can use .dds and .bmp image formats. Another limitation involves how many items can be placed into any single cell. Rather than allowing themselves to be hindered by these limitations, the modding community has risen to the challenge to find new ways around them to pursue their endeavors.

A majority of the initial assumptions that I made were confirmed by the modding community responses and my subject matter expert. Having been exposed to this community for several years prior to becoming an active participant, I had a good idea of what information needs would be required when I began the project.

SECTION III: ANALYSIS AND SUMMARY

Analysis of Work Activities and Information Needs

Activities	Requirements	Resources	Decisions/Deliverables	Targets
Define mod/details	-Determine schedule -Task assignment -Define storyline -Appoint team leader -Determine resources needed and available -Identify theme	-Brainstorming -Collaboration -Group meetings/chats -Online requests	-To define the scope of the mod -To create ideas -To plan deadlines -To assign work to appropriate team members -To plan quests, dialogue, and storyline	-Modders -Team members -Mod community
Gather/define resources	-Concept art -Game lore -Reference drawings -Textures -Music	-Custom made by mod team -Online resources -From books -From other modders	-To create mod content for the game -To get ideas for placement in the CS	-Modders -Team members -Artists -Writers -Musicians -Mod community
Manage and organize mod progression	-Manage timeline -Track Progress/status -Maintain team communications -Maintain material backups	-Individual responsibility or team leader/assigned team member -Schedules	-To plan the mod stages of development -To track progress of the mod and completion status -To avoid scope creep -To	-Modders -Team leaders -Appointed team members
Create resources	-Model Creation -UV Map/Unwrapping -Texturing -Rigging/Animation -Create scripts -Create quests -Create dialogue -Create NPC's	-Free online resources -Modder created resources -Other games -Tutorials	-To place into the game using the game editor CS -To use in other mods -To upload to resource sites -To experiment with new techniques	-Modders -Team members -Mod community -Modelers -Texture artists -Programmers -Animators
CS Editor Placement	-Construction Set editor -Models -Textures -Animations -Scripts -Quests -Dialogue -Modder permission to use their resources	-The Construction Set game editor is provided by the game manufacturer -Most similar game editors are provided by other developers or created by the modding community	-To create mods -To create alternate layouts to determine best placement -To experiment with game and modder created content	-Modders -Team members -Mod editors
Test	-Functioning mod -Beta testers -A documentation system -Game to test it in	-Community forum members -Team members -Volunteers	-To find errors or bugs in the mod -To determine if scripts and quests are performing properly	-Modders -Team members -Beta testers
Debug	-Construction Set game editor -Text editing software -Time	-Team members -Game developer -Input from beta testers and team members	-To remove errors and bugs from the mod -To prepare for release of the mod	-Modders -Team members

	-Patience -List of documented bugs or problems in-game -Dedicated and knowledgeable team members		-To ensure scripts and quests perform properly	
Write read me	-Knowledge of the mods storyline or what it adds to the game -Permission from modders whose content is used	-Individual modder or team member assigned to create the read me -Read me templates	-To inform modders of what the mod does -To explain how the mod is installed -To advise other modders what conflicts with other mods may exist	-Modders -Team members
Upload to host site	-Host site -Archive compression software -Proper packaging of files	-Numerous host sites are available to modders to upload mods to -There are a number of free archive compression programs available	-To circulate the mod for download -To make modder created content available to others	-Modders -Mod community

Based on the interviews performed, the community responses, and my own experiences with this Community of Interest, I have been able to identify the following information related issues and opportunities within this community.

Issues

- While the developer’s forums serve as a community hub, a great deal of the information that modders use is spread out over numerous sites, thus it is sometimes difficult to locate. This includes modder’s resources, models, textures, tutorials, threads, and the mods themselves.
- The response time to forum based questions are restrictive to learning in a real-time capacity. Sometimes it can take days for a specific member to respond to a posted question or e-mail.
- A lot of the information in this community is “tribal” knowledge, in that it applies specifically to this particular Community of Interest. This knowledge must be passed on, often in a mentor/student relationship or in the forums, but there is no dedicated system to preserve or categorize the knowledge that has already been shared. As people leave the Community of Interest, their knowledge can be lost.
- There is an overwhelming amount of information and resources available to modders. The pool of knowledge created by the modding community has no discernable management system to organize, store, maintain, update, and secure it.
- Group and mentor/student communications rely primarily on instant messaging. However, sometimes this is not enough to properly convey ideas or instructions, especially when dealing with new concepts or unfamiliar software.

Opportunities

- There is an incredible amount of information available to beginning and experienced modders through the community forums and links to other sites. It is

entirely feasible for a person with no experience or prior knowledge to learn how to create their own content and add it to the game. All it takes is time and patience.

- A number of fan sites and forums have built up around this community. While this makes it difficult sometimes to find information, it also ensures that all the accumulated modding knowledge to date will not be lost if one site closes. Many sites duplicate threads involving works in progress as well as serving as mirror host sites for completed mods.
- There are a lot of resources (models, textures, animations, etc) available to the community, either packaged individually or within a mod. These resources can be used (with the creator's permission) to create other mods. This allows a beginner to practice with the game editor Construction Set without having to create any content of their own. Trying to learn everything at once can be overwhelming.
- The community forums have a number of rules and regulations in place to maintain a relaxed atmosphere that is conducive to both learning and modding. Many community members volunteer their time to moderate the forums and to pass on their own knowledge.
- There are a lot of tutorials online beyond the forums to assist with learning the individual programs that are need to model, texture, and create content for mods.

Reflections

I can honestly say I will never look at the Morrowind modding community in the same way again. I have been a member of this community since February 27, 2005, but I have only just recently ventured into creating mod content myself. Trying to approach this subject objectively was difficult because I like the Morrowind modding community as it is. I am comfortable with it as a social medium. However, the entire process of dissecting the information needs of the modding community through interviews and analysis has made me realize that there are opportunities to make contributions to the community outside of modding. Identifying specific practices or ideas that can be used to improve, organize, and preserve the knowledge base of this community is a goal worth pursuing.

My initial thoughts were to pursue creating a legacy website that can be used as a forum, wiki, and online data warehouse. As I am pursuing both Web Development and Database IT degrees, this is something that I could conceivably incorporate into my studies. However, as indicated by my research, the sheer magnitude and scope of the information and resources available to this community would be an almost insurmountable amount of work to collect, organize, and maintain. For those reasons I have decided that such an endeavor might be something better pursued after I graduate.

Getting other modders to contribute their views was the easiest part of this project to complete, although I did have to interview my subject matter expert during several chat sessions to get the answers I needed. Trying to condense my research into a logical analysis of their information needs was much more difficult. In retrospect, there were a number of questions that I could have asked that were either more specific to their informational needs or that might have helped me narrow the list of responses to the questions posed. A series of polls would have been easier for them to answer and may

have elicited more responses. It would have also saved me a considerable amount of effort in documenting their individual responses.

Considering that this community has evolved around creating content for a video game, I was surprised to see a number of practices that were very similar to those I have been exposed to in Project Management classes. Management of teams, defining plans and project scopes, avoiding scope creep, task assignment, progress tracking, iterative development, and conflict resolution are all subject matter that I have covered in previous classes, yet I am seeing those same practices being applied by game modders who have probably not been exposed to them in an academic environment. It is highly likely that there were/are a number of modders who have served in some form of management position during their careers, and other modders have adopted practices and techniques they introduced to the community.

As unusual as this may sound, I have personally never completed playing the game that I mod for, and was surprised by how many other modders have not as well. Many members have adopted a mod-as-you-go approach to playing the game, creating mods to repair what they see as problems or oversights with the game or to add content that they would like to see in it. Others only play the game to check the performance of mods they create or to test it for bugs. Modding for the game is almost as popular as the game itself and the Construction Set editor has been used as a teaching resource for college classes.

The problem with any game is that people will eventually lose interest or move on to another game. The same can be said of game modders. When they move on, a part of the community's knowledge base is lost if they have not passed on techniques or practices that they know. There is also no "centralized" data warehouse, at least for this community, to store the accumulated knowledge, tutorials, resources, and mods that have been created or conceived.

Concluding Remarks

The information needs of modders are as diverse as the mods they create and the modders themselves. There is no standard development process that all modders can agree upon, although many follow similar development plans. A few of the things that most modders believe is that it is better to start a mod by developing a plan and sticking to it, knowledge of the native Construction Set editor is a necessity, and that it is helpful to know more than one aspect of mod development. Most modders are self-taught, often using tutorials and trial and error to learn. While a number of modders use instant messaging to communicate, the most common method for maintaining contact with team members and acquaintances is through community forums.

Creating mods and mod content is an incredibly time consuming endeavor, requiring patience, knowledge, and the ability to recognize when it is time to step back and take a break. Knowing your own limitations based upon your skills, knowledge, and dedication will be most helpful in identifying mod projects that are beyond the scope of your abilities.

Recommendations

Many modding communities and forums have evolved over time to serve as a social network that allows modders to associate with others with whom they share common interests. However, a lot of these communities and forums fail to provide specific information needs that modders may need to pursue their craft. Modders need access to a number of resources, including tutorials, game lore, 3D models, textures, music, etc., and that information is often difficult to find if you are not familiar with the community websites.

The Morrowind modding community has a developer sponsored official forum and a number of associated fan sites that have been created by modders and fans of the game. Having to search through numerous websites for information can be time consuming, and some of this information is lost when websites are closed. Although most community sites are loosely associated, they would all benefit from a centralized data repository similar to an enterprise portal that could serve as an online warehouse and hub for forum threads, mod resources, game lore, and the mods themselves.

Using a modified enterprise portal would allow each individual forum and website to save and access their information in a single location. By including each contributing website in the framework, these websites would become a secure access point to the portal using a web-based user interface. Forum and site members could then access the information in the portal using their website specific username and password. Individual backups could still be maintained locally to preserve the integrity of each site's information if desired, and each website can still be personalized to the owner's own preferences.

By using the portal as a repository for the modding community's accumulated knowledge, searching for information on specific mod-related information and tutorials would be easier and quicker to perform. Individual websites could share responsibility for its content management, and storage costs could be shared between them. Identifying duplicated data, or finding outdated or incorrect data that has been spread over multiple websites could also be managed more efficiently if all of that data was in one location.

With all the information being updated and stored in a single location, it would also be easier to categorize the data by type, theme, author, and content. The last time I checked several years ago, there were over 7,000 mods of various types for the Morrowind community alone. There are also at least a dozen websites with active forums, and numerous sites with resources, tutorials, and game lore. Trying to find specific information is often difficult at best.

One of the strengths of the Morrowind modding community is that the information is spread out between numerous websites, thus, if individual sites close or disappear, the majority of the community's knowledge will not disappear with them. When a website closes without passing the information they had stored on to another modding site, a considerable amount of knowledge and mods can be lost. Over 3,000 of the earliest mods were lost to the community when a European fan site disappeared without any warning. An enterprise portal keeps all of the information from its contributing websites

up to date, thus it would back this knowledge up before sites close to ensure that the information they hold is preserved for future modders.

Many modders are resistant to change and some use their websites as a source of revenue through advertising. By using a modified enterprise portal, they can still maintain their individual sites without losing their revenue sources. However, convincing an entire community as old and as developed as this one to develop an enterprise portal would still require a significant amount of time and effort. Most participants would expect to see a prototype to showcase its benefits before they would commit to such an endeavor. This is an established practice in the modding process as well. Many modders expect to see some amount of effort in starting a mod before they will "join" a group or team.

Another recommendation for the modding community involves software for communications between modders, especially those working in groups or that are in a mentor/student relationship. Using the community forums requires waiting for a person to respond, which can sometimes take days. Many forums allow the members to send a personal message or e-mail to another member, but this is not a good way to hold a conversation. To this end, many modders have adopted instant messaging programs like Skype and MSN Messenger to hold person to person chats. While this may work for team brainstorming, casual conversations, and sharing files or pictures, it is still inadequate at times, especially for those in a mentor/student relationship.

Because many people who are just beginning to mod are not familiar with the numerous programs that are used to create mods, it is often difficult to explain it verbally or while typing. I was looking for an alternative software solution that could be used by the entire modding community. Because many modders are underage or not financially capable of spending large sums of money for software, it was important that this alternative be cheap, or better yet, free. I mentioned this to my subject matter expert, who is also my mentor, and he pointed me to a program he had just been introduced to that allowed him to remotely access and control another person's computer.

This program, called Teamviewer, is designed for collaboration and presentations. It allowed him to remotely access my computer (with my permission), open software that I was still relatively unfamiliar with, and give me a real-time demonstration of how to use the program. It was almost like a video tutorial created to answer my specific questions as I had them. The amount of time and typing that this demonstration saved was significant. And best of all, the software is free and easy to use. This program, or one similar to it, is something that I would highly recommend for groups and mentors/students to add to their resources. It simplifies the learning process and can save hours of frustration and/or confusion.

SECTION IV: APPENDICES

Glossary of Terms

Animation – the process of giving motion to a three dimensional objects, often by rigging the object to a skeleton, and moving parts of that skeleton based on measurements of distance and time.

Cell – a cell is a section of a game world, often denoted by coordinates that specify a certain part of a grid on a two dimensional plane.

Construction Set (CS) – the native game editor for the game The Elder Scrolls: Morrowind, which is used to add content to existing or modder created cells and environments. Once this content has been added, the editor program can save the changes as a mod.

Editing tool – a proprietary program usually provided by a game's developer used to place content created by designers or modders into the game world.

Enterprise portal - also known as an *enterprise information portal* (EIP) or *corporate portal*, is a framework for integrating information, people and processes across organizational boundaries. It provides a secure unified access point, often in the form of a web-based user interface, and is designed to aggregate and personalize information through application-specific portlets. One hallmark of enterprise portals is the decentralized content contribution and content management, which keeps the information always updated.

Features

- Single Sign-On - enterprise portals can provide single sign-on capabilities between their users and various other systems. This requires a user to authenticate only once.
- Integration - the connection of functions and data from multiple systems into new components/portlets/web parts with an integrated navigation between these components.
- Federation - the integration of content provided by other portals, typically through the use of WSRP or similar technologies.
- Customization - Users can customize the look and feel of their environment. Customers who are using EIPs can edit and design their own web sites which are full of their own personality and own style; they can also choose the specific content and services they prefer. Also refers to the ability to prioritize most appropriate content based on attributes of the user and metadata of the available content.
- Personalization - Personalization is more about matching content with the user. Based on a user profile, personalization uses rules to match the "services", or content, to the specific user. To some degree, you can think of the two like this: customization is in hands of the end user, personalization is not. Of course actual

personalization is often based on your role or job function within the portal context.

- Access Control - the ability for portal to limit specific types of content and services users have access to. For example, a company's proprietary information can be entitled for only company employee access. This access rights may be provided by a portal administrator or by a provisioning process. Access control lists manage the mapping between portal content and services over the portal user base.
- Enterprise Search - search enterprise content using enterprise search

Face – a planar triangle, created by three points located in three dimensional space.

Game engine – a program that renders creator content into a game world using specific rules and programming to control how that content behaves, moves, and reacts inside that world.

Game world – the content of a game as rendered by the game engine using models, textures, animations, programming, and scripts. The game world can be something as simple as a game board or as expansive as an entire universe.

Gloss map – a map that creates a shiny effect on an object. On a gloss map, any part that is not intended to be shiny is blacked out.

Mod – a plug-in for a game with content created to alter the game beyond the original designer's intent. That content can be sounds, models, textures, animations, or programmed scripts.

Modder – a person who creates new content for a game that was not intended or included by the game's developer.

Model – a three dimensional representation of an object created using software such as Maya, 3D Max, or Blender on a computer.

Normal map – a texture map that creates a sense of depth by using light and dark values to represent how far or close a part of an object is to a person.

Plug-in – created by the mod editing tool, a plug-in contains the model, texture, animation, and programming data that is placed into the game world by the modder.

Rigging – rigging defines the animation of a simple framework, often called a skeleton, of an object with its various pivot points to create motion. The movement of the rigged skeleton moves the object it defines in a like manner.

Script – programming that defines a sequence of expected behaviors. A script can control how an object reacts to forces around it or can define how an object performs a specific function or movement based on a schedule.

Texture map – a map represents the flat, 2D image of a three dimensional object that has been UVW unwrapped. A texture map is an image that defines the materials on that object, such as the leather wrapped around the handle of a sword or the brick on building's wall.

Tile set – describes an entire set of texture maps that define a certain type of architecture common to a game.

Total conversion - describes a mod that removes and replaces all of the content, models, textures, quests, scripts, dialogues, and occasionally aspects of game play, of the original game.

Unofficial patch - a mod for an existing game that fixes errors that were not corrected by the developers of a game in an official release, unofficial patches are often produced by dedicated members of the modding community.

UVW mapping – the process of assigning three dimensional coordinates to each point on a model to define its surface. This surface is unwrapped into a 2D image so that a texture artist can assign materials and details to it.

UVW unwrapping – unwrapping is the process of taking a mapped three dimensional surface and flattening it into a 2D representation.

Bibliography

"Modding." *Wikipedia, the Free Encyclopedia*. Web. 03 Oct. 2010. <<http://en.wikipedia.org/wiki/Modding>>.

"Mod (computer Gaming)." *Wikipedia, the Free Encyclopedia*. Web. 03 Oct. 2010. <[http://en.wikipedia.org/wiki/Mod_\(computer_gaming\)](http://en.wikipedia.org/wiki/Mod_(computer_gaming))>.

"Morrowind Mods." *Bethesda Softworks Forums*. Web. 03 Oct. 2010. <<http://forums.bethsoft.com/index.php?forum/12-mods/>>.

"Tutorials - Mod DB." *Games and Mods Development for PC XBOX Playstation Nintendo - Mod DB*. Web. 03 Oct. 2010. <<http://www.moddb.com/tutorials>>.

White, Walker, Christopher Koch, Johanne Gehrke, and Alan Demers. "Better Scripts, Better Games." *Association for Computing Machinery. Communications of the ACM* Mar. 2009: 42. *Under Construction*. Association for Computing Machinery. Communications of the ACM, Mar. 2009. Web. 03 Oct. 2010. <<http://proquest.umi.com.librarydb.edcc.edu:2048/pqdweb?did=1666268921&sid=1&Fmt=2&clientId=1848&RQT=309&VName=PQD>>.

Wallis, Alistair. "Is Modding Useful?" *Game Developer 2007: 25. Under Construction*. CMP Media LLC. Web. 03 Oct. 2010. <<http://proquest.umi.com.librarydb.edcc.edu:2048/pqdweb?index=42&did=1299742941&>

SrchMode=1&sid=4&Fmt=3&VInst=PROD&VType=PQD&RQT=309&VName=PQD&TS=1286078817&clientId=1848>.

Midgetalien. "[LIST] Morrowind Video Tutorials." *Bethesda Softworks Forums*. 29 Apr. 2009. Web. 03 Oct. 2010. <<http://forums.bethsoft.com/index.php?/topic/984278-list-morrowind-video-tutorials/>>.

Angiolillo, P. 2005. Gaming making the grade. *Technology Review* (Sept 27, 2005)

SilverMagics. "Morrowind Construction Set - Tutorial 1." *YouTube - Broadcast Yourself*. 10 Mar. 2009. Web. 15 Oct. 2010. <http://www.youtube.com/watch?v=h_2O2cXC66U&feature=related>.

SilverMagics. "Morrowind Construction Set - Tutorial 2." *YouTube - Broadcast Yourself*. 10 Mar. 2009. Web. 15 Oct. 2010. http://www.youtube.com/watch?v=qzLQk6db0gM&feature=channel_page

"Enterprise Portal." *Wikipedia, the Free Encyclopedia*. Web. 02 Dec. 2010. <http://en.wikipedia.org/wiki/Enterprise_portal>..

Revision History

Date	Revision Notes
10/03/2010	Created Draft 1
10/29/2010	Created Draft 2, Added interview questions and responses, added to Glossary of Terms, Bibliography, updated Assumptions, added Table of Contents and Revision History
11/24/2010	Final Version, revised existing content, added numerous sections such as analysis, recommendations, issues, opportunities, concluding remarks, interview findings, and occupational definitions.

Community Responses

The community forum thread can be viewed at this web address:

<http://forums.bethsoft.com/index.php?/topic/1120938-who-wants-to-help-me-get-an-a/>

Responses: The ElderScrolls: Morrowind Game Modding Community		
Question	Responses	Commonalities
How do you decide what kind of mod you will do either individually or as a member of a modding team?	<p>I play the game, see what I want in it now, or see what other people want</p> <p>If an idea comes to mind and I think I can handle it, I try to at least start it. I like to try to help friends with their mods or ideas, if my 2 cents is worth it.</p> <p>The motivating factor here for me is frustration. When I play, I notice things that cause me problems and when I find enough</p>	<p>What I want in the game What others request</p> <p>What I want in the game</p> <p>Problems with game</p>

	<p>related to a single element of play, I make a mod to fix it . . . or I fix an existing mod to suit me.</p> <p>solo decision is just my desire to do what I feel I can. I would never lead a team. but do give input and resources to team efforts on occasion. I think most team mods start with an idea by one the changes as a group works on it.</p> <p>Individually: If no one else is creating what I want to play with then I'll do it myself, I only make mods of things or quests I want in my own game Collaboratively: I have to trust them (based on forum comments), feel that I have the time, ability and desire to help and that it has to be fun</p> <p>As a team-member, usually sitting down and doing some quick brainstorming to come up with concepts, then picking the most popular of those. Individually, mostly the same thing.</p> <p>How excited I get about it, basically. When coming up with my own ideas, I usually let them simmer for weeks or months, and if I'm still interested after that time I may go ahead and take the first steps (planning, or messing around in the CS). When looking at a team, I also look at how well the team is organized and how excited its members are about the project.</p> <p>I see something I think would be great to have in the game or I see a request from someone else that I think might be interesting to do.</p> <p>I dream up some impossible scenario and then break it down into manageable small parts and start working, then I put it aside for newer and better things.</p> <p>Brainstorming! I try to imagine how I would like things to be if I were the creator of the game.</p> <p>My ideas come from either playing the game and wishing for a new feature or looking at the mods that have been done and wanting something a little different.</p> <p>If I think it sounds neat and plausible, I'll take a whack at it. I do stuff that I am interested in and have a blast creating.</p> <p>I usually encounter some small thing I want to change or add to my personal game...</p>	<p>What I want in the game</p> <p>What I want in the game</p> <p>Brainstormed ideas with a group</p> <p>What I want in the game</p> <p>What I want in the game</p> <p>What I want in the game What others request</p> <p>What I want in the game</p> <p>What I want in the game</p> <p>What I want in the game What others request</p>
--	---	--

	<p>Then when I've made that, I usually think "That's good, but wouldn't it be cool if..." and then anything might happen. If I had been God, I would have probably created Adam and Eve and the garden, then saw that it was good, and thought "Wow! What a great place for a tyrannosaurus Rex!"</p> <p>Personal interest: there's something in the game that's not there that I want and no one else has done (or done the way I want it).</p> <p>Don't really decide, I mostly landscape. The only aspect that really interested me.</p> <p>I have never really worked in a team. When I want to mod, I mod what I want at that moment, so it could be anything.</p> <p>I don't technically make mods, I usually make textures and or models. I decide by inspiration mostly, if I find some artwork inspiring or if I think of something interesting I usually go for it.</p> <p>I have to like the idea. If I don't like it or won't use it I don't make it.</p> <p>I work by myself most of the time so I make my decisions based on what I would like to make, or have the ability to make. I do requests from time to time, so sometimes I let other people decide for me.</p>	<p>What I want in the game</p> <p>Resources of items I find inspiring or interesting</p> <p>What I want in the game</p> <p>What I want in the game What others request</p>
<p>What kind of information do you need when doing a mod?</p>	<p>where to find tutorials, I practically read all of them 1st to see what I can find out, or ask other people in the "community"</p> <p>Everything possible. First is the Do's and Don't's, so I can define the limits of my imagination.</p> <p>Any and all documentation I can find, for existing mods, tutorials, and original game details like NPC IDs questlines, just about everything on UESPwiki is absolutely essential.</p> <p>lore. a plan of what you are going to do for a mod. and a list of who's stuff you used to give credit for.</p> <p>I normally want game lore information</p> <p>Plot summary, any implementation specifics that may be required, any constraints (no MWSE, no MGE). The</p>	<p>Tutorials Community input</p> <p>Do's and don't's</p> <p>Tutorials Original game details UESPwiki</p> <p>Game lore Planning Credits</p> <p>Game lore</p> <p>Plot summary Game dependencies Implementation specifics</p>

	<p>more knowledge, the better. If someone were to ask me to make a mod and give me an 80-page tech specs doc, well... no thinking on my part.</p> <p>What the mod is about, who is working on each area, how its structured, what is already done and what still needs doing.</p> <p>2 & 3 go together: the information you need for a mod depends on what type of mod you want and the knowledge stems from the information. For example: if you want to make an uber weapon, you need to know what types of weapons are already there and how to make new ones.</p> <p>Information comes usually in the form of inspiration. Sometimes from other games. Sometimes from MW itself.</p> <p>It differs. It would be wise to know the lore of a game before you start modding. Knowing the CS and knowing the game-mechanics is also vital.</p> <p>Primarily I'll need technical details such as the scripting functions found in MWSFD and how-to videos on blender. Catalogs of the objects available in the construction set are helpful.</p> <p>A lot of the stuff I make I try to make fit into lore. I read the lore A LOT, I would say that I've become fairly familiar with it. When I read it I pay close attention to what things I think could be modded into the game.</p> <p>Since I can't do original meshes, I need to know where I can find them as modder's resources.</p> <p>I need to know if anything similar has been done, and whether or not an alternative is needed. Sometimes a similar mod will have been done, and perhaps wasn't very successful, so it helps to know why it was or was not generally liked.</p> <p>Depends on the mod. If it's a quest or companion mod, you need to know how</p>	<p>Mod details Team structure Mod progress</p> <p>Mod details</p> <p>Information A source of inspiration</p> <p>Game lore CS knowledge Game mechanics</p> <p>Mod details Scripting requirements CS knowledge Tutorials</p> <p>Game lore</p> <p>Modder's resources</p> <p>Mod similarity and success</p> <p>Storyboarding Dialogue</p>
--	--	---

	<p>to write a story, how to write dialogue in particular, how real people act, how the culture the character(s) are from will affect them, etc. Since that's primarily what I do, that's mostly the information I need.</p> <p>Well, I need concepts and ideas for places I'm going to build.</p> <p>Since I always do mods myself, I mostly require no additional info. Sometimes, when I want to be correct about some facts, I may go through USPWiki.</p> <p>The most important information, for me, when modeling and texturing, is knowing and understanding what the materials of any given asset are or are made of. Knowing the materials and how to best recreate them can really improve the final product.</p> <p>Depends. If I'm making something for myself all I need is a basic idea to run with. If I'm doing something for someone else, the more specific they are the better.</p> <p>Depends on the mod. For models and textures I prefer a good visual reference, but a lot of times I just use my imagination.</p>	<p>Game culture</p> <p>Concept ideas and resources</p> <p>Factual reasources</p> <p>Item materials for texturing</p> <p>Mod details</p> <p>Visual references</p>
<p>What kind of knowledge do you need when doing a mod?</p>	<p>basic Knowledge of the CS and how it works.</p> <p>Depends on the mod. In my case, hopefully not much. I stick to simple tasks in the CS.</p> <p>Depends on the nature of the mod. A modder needs to know their way around the CS even for the simplest of mods. Basic scripting is required for just about any mod whether it is in dialog or individual scripts. For texturing and modeling, some experience with the programs required are not optional.</p> <p>knowledge of at least the cs. and other 3d and art programs as your skill increases.</p> <p>If doing a Morrowind Mod - need to</p>	<p>CS knowledge</p> <p>CS knowledge</p> <p>CS knowledge Scripting Software familiarity</p> <p>CS knowledge Software familiarity</p> <p>Game details</p>

	<p>know the game, understanding the story, characters, places, races and lore. Need to know how to ask for help and where to go - the CS forum or some of the unofficial forums can be really helpful Need to know your limits - modding is time consuming - creating quests with less generic NPC's can take months even years - need to know if you have time and energy to finish what you start.</p> <p>Before I started modding I had never used Nifskope, the CS or GIMP - never written a script before - these can all be learned as you go they are not pre-requisites to modding</p> <p>Need to know that regular and multiple back-ups are your friend!!!</p> <p>Knowledge of the CS and the game system. Knowledge of the scripting language, bugs in the engine, resource locations and people to ask are a plus.</p> <p>Depends on the mod.</p> <p>Artistic knowledge?</p> <p>Depends on what type of mod you're doing.</p> <p>A basic understanding of programming logic is helpful in scripting. Knowing the 3d software of your choice is essential for modeling.</p> <p>I use tutorials when I'm working on particularly tricky stuff like scripting or modelling. Otherwise I have the head knowledge of the CS and Nifskope to make what I want.</p> <p>Of course knowledge of the CS is a good placeto start, then scripting, texturing, or how to adapt things into what is needed. A fair bit of experience with various modding tools and utilities is helpful as well.</p> <p>Understanding how dialogue and the journal work is usually important as well.</p>	<p>Game lore CS knowledge</p> <p>Know your limitations Investment of time</p> <p>CS knowledge Software familiarity Scripting</p> <p>Back up your work</p> <p>CS knowledge Scripting Game engine bugs Resources</p> <p>Artistic knowledge</p> <p>Scripting Software familiarity</p> <p>Scripting Modeling CS knowledge Software familiarity</p> <p>CS knowledge Scripting Texturing Software familiarity</p> <p>Scripting</p>
--	--	--

	<p>How to use the tools, primarily (be it Photoshop, NifSkope, or the Construction Set).</p> <p>Only landscaping, and my mind, nothing else</p> <p>Mostly I just build my mods with vanilla stuff, except for dialogues and quests, of course, so I need to know how CS works and that's it.</p> <p>Modeling and texturing knowledge is nice. Other than that knowledge of the programs used to create the fore mentioned is also very helpful.</p> <p>Basic idea of how to use the CS and the limitations of what can be done in Morrowind. Anything more specific depends on the mod itself. If I'm doing something that requires scripting, then I need to know how to script. If I'm making textures I need to know how to use an image editing program.</p> <p>Not sure exactly what you mean, so I'll say knowledge of my tools and the end result.</p>	<p>CS knowledge Software familiarity</p> <p>Scripting CS knowledge</p> <p>Modeling Texturing</p> <p>CS knowledge Scripting Texturing</p> <p>Software familiarity</p>
<p>Where do you find this information?</p>	<p>PES, friends (mainly), Tesnexus and other websites made by a community member</p> <p>From my initial CS experiments, someone was kind enough to teach me the basics.</p> <p>I ask. I start with my husband, who is also a modder, then I google it, and if I can't find the answer or when I find it, if it doesn't make sense, I ask on the forums.</p> <p>online sites and forums</p> <p>For lore and game info I go the Imperial Library and UESP (often) also the lore forum For scripting and how to do different things I ask in the CS forum I searched online and downloaded</p> <p>1. MWSFD 2. Morrowind Mod Maker Manual</p>	<p>Community forums Acquaintances</p> <p>Mentor</p> <p>Tutorials Community forums Other modders</p> <p>Community forums</p> <p>UESPwiki Imperial Library</p> <p>Community forums Tutorials</p> <p>Tutorials Community forums</p>

	<p>3. The complete guide lite 4. MW-advice for new modders V2 5. Housebuilding 6. Srikandi-MW-Item_index 7. MW-Clothing-Catalog V2 8. Landscaping tutorial V1.2 9. NPC Scripting 10. Quest tutorial 11. Dialogue tutorial 12. Wrye Notes 13. Korana's Glow Mapping Tutorial</p> <p>Mah brains. Or Google, or here, or GHF.</p> <p>For construction set stuff, the CS wiki, the CS forums, and the TESAlance. If I can't find something, I ask my network of friends/acquaintances/coworkers.</p> <p>There are plenty of resources: the UESP, fellow forum members, various guides, etc.</p> <p>On these forums if I don't know it myself.</p> <p>I prefer to learn things myself. I tend to remember them longer.</p> <p>I use Morrowind Scripting For Dummies this forum and Great House Fliggerty for scripting and dialog reference. For modeling I look to the tutorials at nifTools.org, blender.org and YouTube.</p> <p>The Imperial Library, for lore. And for other things like tutorials and resources I refer to links at this forum a lot.</p> <p>Often by trial and error, but more often in tutorials or existing help aids like "Scripting for Dummies". When it's really a tough problem, or outside of experience entirely, the forums are an invaluable source.</p> <p>Tutorials abound, but these forums have been the primary source.</p> <p>In my mind, and sometimes few locations from real life or some game inspire me with some ideas.</p> <p>I discover it myself, or read some tutorials.</p>	<p>Community forums Internet</p> <p>UESPwiki Community forums Acquaintances</p> <p>UESPwiki Tutorials Community forums</p> <p>Community forums</p> <p>Self-taught</p> <p>Tutorials Community forums</p> <p>Imperial Library Tutorials Community forums</p> <p>Tutorials Community forums Trial and error</p> <p>Tutorials Community forums</p> <p>Other games Experience</p> <p>Tutorials</p>
--	--	---

MGMT 215, Fall 2010

	<p>I started out with some help from friends, tutorials, and simple trial and error. Most recently I learn a lot watching video tutorials and again, by trial and error (tinkering around).</p> <p>I either poke at it until it works or I look up a tutorial online.</p>	<p>Acquaintances Tutorials Trial and error</p> <p>Trial and error Tutorials</p>
<p>How did you learn how to mod?</p>	<p>I wanted to make 3D worlds and MW was the only thing available, so I taught myself the keys and then hunted down VERY basic tutorials</p> <p>Someone taught me the basics, and I experimented from there. If I had a question, I'd ask in the forum and hope for an answer that I can understand.</p> <p>By customizing and doing bugfixes on other people's mods.</p> <p>self taught, still learning.(there ain't no end to that)</p> <p>Just doing it - trial and error</p> <p>Trial and error and error and error.</p> <p>Muddling about in the CS, mostly. I skimmed the My First Room tutorial and the official ones by Bethesda, but I either did not understand or did not have the patience to follow them. After those basics I basically just started doing stuff, failing horribly at first but eventually learning from my mistakes.</p> <p>I started out by opening up the CS and seeing how it was laid out. Then I opened up various mods to see how they were done. I read some guides and browsed through the forums. Finally I started making small mods for myself to change things I didn't like about the game and decided that I wanted to release something.</p> <p>I wanted to make a game.</p> <p>It started with the Neverwinter Nights toolkit a long time ago. When I saw that my favourite game had a similar tool, I couldn't resist. During these years</p>	<p>Self-taught Tutorials</p> <p>Mentor Self-taught Community forums</p> <p>Self-taught</p> <p>Self-taught</p> <p>Trial and error Self-taught Trial and error</p> <p>Trial and error Tutorials Self-taught</p> <p>Self-taught Tutorials</p> <p>Self-taught</p> <p>Trial and error Self-taught</p>

	<p>there's been a lot of "trail & error" until I finally learnt how to handle the CS.,</p> <p>I started with a couple of basic tutorials and then added in liberal doses of trial and error. When I found something that I couldn't figure out a post to the forums and short wait resulted in answers.</p> <p>Taught myself the basics a few years back, and have picked up everything else along the way.</p> <p>Trial and error mostly at first, then I used tutorials and practice, practice, practice. I made literally dozens of mods of which none were published and most have been just deleted from my computer when I finished them.</p> <p>Reading a few tutorials, porting my knowledge from my novel-writing and 3D artwork, and mostly trial-and-error.</p> <p>I'm self taught.</p> <p>Once I wondered, how easy is it to make something in MW, so I just opened CS and started doing stuff, that's how I basically learn, apart from tuts.</p> <p>Read answer 4.</p> <p>Some combination of trial and error and tutorials.</p> <p>Reading lots of tutorials and lots of trial and error.</p>	<p>Tutorials Trial and error Community forums</p> <p>Self-taught</p> <p>Trial and error Tutorials Practice</p> <p>Tutorials Trial and error Previous knowledge</p> <p>Self-taught</p> <p>Self-taught Tutorials</p> <p>Tutorials Trial and error</p> <p>Tutorials Trial and error</p>
<p>How do you share information and ideas with other mod team members?</p>	<p>I prefer to suggest it as an idea rather than say "we're doing this now I like it blah blah blah"</p> <p>Skype, because it's so convenient.</p> <p>Forum threads, PMs and by turning my chair around and asking my husband.</p> <p>forums, pm's on forums. emails. and free hosting sites to move modding materials.</p> <p>Carefully (so no feelings are hurt)</p> <p>If I'm on a team, forums most often,</p>	<p>Brainstorming</p> <p>Skype</p> <p>Community forum Acquaintances</p> <p>Community forums Hosting sites E-mails</p> <p>Community forums</p>

	<p>although MSN messenger and/or IRC are valuable tools.</p> <p>Answering other people's questions when they come up. I've also written a few guides and tutorials.</p> <p>I don't work in a team.</p> <p>IRC/MSN or Google Wave.</p> <p>Forums, websites etc</p> <p>Through the use of forum threads.</p> <p>Mainly over the forums and PMs, but I may be moving to Skype or IRC channels if I get a reliable internet source.</p> <p>I've never worked as a team member. I tend to avoid that... I think because I would feel restricted by doing an assigned task, and it would then be work rather than play.</p> <p>I don't much, but when I do, it has been via these or Emma's forums.</p> <p>I used to but they were ripping me off haha. Now I only share with closest members from here. How? See question below</p> <p>Currently Skype.</p> <p>The forums here, email, or MSN messenger usually.</p> <p>I've never worked on a mod with other people, but if I were to do so, I'd prefer using the PM system of the forums here. I'm hardly ever on an instant messenger and next to never check my e-mail.</p> <p>PM's and email</p>	<p>MSN Messenger IRC</p> <p>Community forums Tutorials</p> <p>IRC MSN Messenger Google Wave</p> <p>Community forums</p> <p>Community forums Skype IRC</p> <p>Community forums</p> <p>Community forums</p> <p>Community forums</p> <p>Skype</p> <p>Community forums MSN Messenger</p> <p>Community forums</p> <p>Community forums E-mails</p>
<p>If someone is 'mentoring' you, or vice versa, how do you communicate with them to share information?</p>	<p>skype,</p> <p>Skype.</p> <p>I give him a kiss and say 'Please" in my sweetest pleading voice.</p>	<p>Skype</p> <p>Skype</p>

	<p>no, sometimes here I give help. or ask for help.</p> <p>Via forum threads With trusted mentors via email</p> <p>Forums or IM or chat, again.</p> <p>Primarily instant messaging, like IRC, msn, or the chatbox built into some forum software. I try and answer forum posts, as well.</p> <p>No one actually mentored me, but I did ask a lot of questions when I first started modding.</p> <p>IRC/MSN or Google Wave.</p> <p>I would say that the best way is through the forums.</p> <p>Personal messages on the forums.</p> <p>I've only helped on the small scale and so have mainly communicated directly in the forum or through private messages.</p> <p>I've used MSN Messenger, Skype, and a long time ago ICQ.</p> <p>Via skype or msn</p> <p>PMs, Skype and tuts.</p> <p>MSN messenger</p> <p>Same as above.</p> <p>Mostly self-taught, but I have learned a lot from forum posts communicating with others</p>	<p>Community forums</p> <p>Community forums E-mails</p> <p>Community forums Instant messaging IRC MSN Messenger Community forums</p> <p>Community forums</p> <p>IRC MSN Messenger Google Wave Community forums</p> <p>Community forums</p> <p>Community forums</p> <p>MSN Messenger Skype ICQ Skype MSN Messenger Skype Tutorials Community forums MSN Messenger</p> <p>Community forums</p>
<p>Do you find it helpful to know more than one aspect of modding, such as modeling?</p>	<p>yeah, the more you know the easier it is.</p> <p>Of course, it would be <i>very</i> helpful. But I don't.</p> <p>Oh yes. You can't get away with a "just" mod anymore. As in this is "just" a house mod. Or this is "just" a dialog mod. It doesn't matter if you can script or model your butt off, if you can't put an</p>	<p>Yes</p> <p>Yes</p> <p>Yes</p>

	<p>intelligible sentence together or apply textures to models, people are going to notice . . . and tell you about it.</p> <p>yes, very much so.</p> <p>Yes - I started modding to teach myself new skills - it would be more productive if I farmed out the scripting to those who know what they are doing but I really want to learn how to do it</p> <p>Yes. As a single modder, or leader of a group, it becomes vital. As a member of a group, less so, but still valuable.</p> <p>Yes, immensely. I was making an interior/exterior expansion the other day and decided that it would be neat to have a custom candlestick, so I popped into Max and made one. Such a request would probably not have been fulfilled by the community, and certainly not in 30 minutes. Its awesome to not be restricted.</p> <p>The more you know, the easier it is to make a mod. If you don't know how to do something in particular, like making a model, you'll have to find it, do without, or ask for help and hope someone makes it for you.</p> <p>Lol. Well Modeling is my main branch of modding. So yes, being able to integrate what I make with MW is important.</p> <p>Yes it would be awesome if you know all the "ways" of creating a mod. There's always limitations when you don't know things like scripting, modeling and so on.</p> <p>Absolutely. Creating a housing mode with a few new items such as a deck of cards or a new desk model really make the work stand apart from the rest. This is one of the reasons that people keep mentionin Koran and Princess Stompers houses when someone wants to know people's favorite. The ability to create dialog, books and scripts change a mod for a static piece of eye candy to an active part of the experience.</p>	<p>Yes</p> <p>Yes</p> <p>Yes</p> <p>Yes</p> <p>Yes</p> <p>Yes</p> <p>Yes</p> <p>Yes</p> <p>Yes</p>
--	---	--

	<p>Absolutely, if you want to be able to do things independently and not rely on asking for everything.</p> <p>I wish I could do modeling... but I can't afford 3ds and I have a problem with Blender... That's because I'm legally blind, and when I try to use Blender large chunks of the GUI are invisible to me, but...</p> <p>The more you know the more you can do, it's as simple as that.</p> <p>The mod I am working on is primarily a quest/companion mod, but I'm finding that knowing how to texture and landscape are useful, also.</p> <p>Extremely, but unfortunately I don't</p> <p>I find it perfect, to be able to make everything you want, just as you want it to be. Currently learning to model.</p> <p>Not really. I stick to what I know and like and do that. I am rather stubborn sometimes.</p> <p>Oh yes. There's only so much one can do with the CS alone.</p> <p>Yes, while I could focus on one aspect of modding, I like being able to multi-task.</p>	<p>Yes</p> <p>Yes</p> <p>Yes</p> <p>Yes</p> <p>Yes</p> <p>Yes</p> <p>No</p> <p>Yes</p> <p>Yes</p>
<p>How do you approach the work involved in making a mod, either alone or as a member of a team?</p>	<p>1st a tell me self it'll take no time, then I think of what I want to make, get some images, read some lore, make or find some models and other resources, and start making it take shape</p> <p>By myself, I try to imagine parts of the mod in my mind, and see what I can do in the CS. As far as helping, someone asks for help, I do what I can.</p> <p>I try to limit my scope before I start, so I don't end up with a mega-all-encompassing mod that is never finished. I decide on what the last thing I'm going to do is and when I get there, I stop.</p> <p>decide what to do. decide how best to</p>	<p>Define mod Get resources Game lore</p> <p>Define mod Determine CS placement</p> <p>Define mod Limit scope of mod</p> <p>Define mod</p>

	<p>do it. rough it out. fine it up. fix what I missed.</p> <p>I grit my teeth and get on and do it - modding is 2 parts inspiration, 5 parts perspiration, 2 parts frustration and 1 part damnation</p> <p>Iterative development, working on bits and pieces, finishing important chunks first, with plenty of background music.</p> <p>Not sure what you mean by this question. Modding used to be my primary leisure time activity, but now I only do it a couple hours a week (due to college). I try to prioritize and do the important stuff first, but I will admit that I get distracted and work on goodies when fancy strikes me. I try to do non-CS stuff, like planing and writing, when I don't have access to a modding computer so that I can dedicate available modding time to actually modding.</p> <p>I think about a mod idea, then promptly forget about it. Actually, this question and #11 are basically two different aspects of one question.</p> <p>Alone until and if anyone wants to help.</p> <p>I prefer working alone. How you approach the work ahead of you differs on what type of mod you have in mind.</p> <p>Chaotically. I'll have a vague idea of what I want and then throw random pieces into it in no preplanned or organized manner.</p> <p>I mainly work alone (unless I make a particular item for someone), so I work on my mod when I have the time. And if it's a big project I just throw effort at it willy-nilly for the most part. If it's a smaller project, then the question doesn't apply as much. I just git-r-done!</p> <p>For me, it's a hobby, something I do to relax and fulfill my creative side's needs. I suppose I approach it as someone else might approach building model airplanes</p>	<p>Define procedures Iterative development</p> <p>Just do it</p> <p>Iterative development</p> <p>Prioritize by importance Plan details Write scripting/dialogue</p> <p>Random placement No defined plan</p> <p>Just do it</p> <p>Work on it until its no longer enjoyable.</p>
--	---	--

	<p>or playing golf. I try to set aside time for it, and I work on it until it's no longer enjoyable, or I run out of time.</p> <p>Sometimes months go by with me doing nothing productive, but this oddly gives me a fresh outlook when I go back to it.</p> <p>Relaxed. I always work alone anyway, so I don't make myself a time limit or anything.</p> <p>I work alone, so I do the mods at my own pace, depending on how much stuff I have to do outside CS and my mood.</p> <p>99% of what I do is usually by myself. I usually just come to the conclusion that I will do whatever I feel like doing.</p> <p>With a lot of time and caffeine.</p> <p>In my usual scatter-brained method, I start in the middle, work towards the beginning, and hope to finish at the end. Honestly, my 'mod work' habits are completely abstract and counter-intuitive.</p>	<p>Take a break when needed</p> <p>Don't set any time constraints</p> <p>Do at own pace</p> <p>No defined plan</p>
<p>Do certain parts of the mod have to be done before others?</p>	<p>exteriors</p> <p>Not usually. Most of my cases are some interiors that can just stand alone. But sometimes if you have a massive interconnecting interior with various exits and stairs, you need to see how they all match up. So sometimes you have to complete stacked interior levels in pieces until you figure it out.</p> <p>Absolutely, but just like a trip, once you've established a destination, then you have to plan out the steps to get there. Different people are going to plan different routes based on their own strengths and weaknesses.</p> <p>first thing would depend on the mod. most of mine start with dropping new meshes and textures into the directory for use.</p> <p>For my quest mod I need to plan out the story line and understand clearly why the player is being asked to do things</p>	<p>Exteriors</p> <p>Not usually Depends on the mod</p> <p>Yes Depends on the mod</p> <p>Depends on the mod</p> <p>Yes Plan the mod</p>

	<p>(what's my motivation for this scene) - I needed to create each NPC's faces before writing their backgrounds and dialogue so I could visualise them more as a character</p> <p>Reserving IDs and creating placeholder items, usually. Some scripts may need to be created early on, as well as actors (NPCs and creatures).</p> <p>Definitely. In my experience, I do exteriors first, then interiors, then NPCs, then dialog and quests. Once basic groundwork is laid parts overlap, for instance, I might be working on the interiors of a town as I detail the exterior.</p> <p>Depends on the mod. You can't add dialog or scripts for an NPC that doesn't exist in the mod already.</p> <p>Yes. first there must be a design phase (Includes brainstorming. Then you test your idea and if it seems like it will work you have a green light to proceed with your normal procedure.</p> <p>Both yes and no. Depends on what type of mod you're doing.</p> <p>In some cases yes. Objects must be created before they can be scripted. Cells must be made before dialog can be filtered for that cell. It generally works best to create cells, then objects (including npc's) and do the scripting and dialog last. This is only nescissary when one part has a depency on another.</p> <p>If it is a big project, it is important to get basic planning down first, background, scale (size), map (for landmass), etc., basic planning. After that is laid out, completion of interiors/exteriors/dungeons should be completed as much as possible before moving on to quests, dialogue. Then during these quests, additional fleshing out of the landscape, dungeons will probably be needed.</p>	<p>Yes CS placeholders Scripting</p> <p>Yes Exteriors before interiors NPC's Dialog and quests Details</p> <p>Yes NPC's before dialogue and scripts</p> <p>Yes Plan the mod</p> <p>Depends on the mod</p> <p>Yes Cells before objects Cells before dialogue Objects before scripts</p> <p>Yes Plan the mod Exteriors/interiors Details</p>
--	---	--

	<p>If you haave a quest, you generally need some kind of story outline that includes a way to start the quest, what has to be done to complete the quest, and what the player should get out of the experience when it's done.</p> <p>If you are using scripted items, you usually have to create the items first so you know what object IDs will be used in the script.</p> <p>Absolutely. A lot of planning has to go into it before anything is done; my texture work has to be done before I can start the companion; the companion needs to be set up before I can start the quests; etc.</p> <p>Depends on what kind of mod are you working on. If you're working on a landmass mod you need to make the exterior 1st, obviously.</p> <p>Not really.</p> <p>Modeling> mapping> sometimes secondary apps for baking normal, ambient occlusion, and sometimes other maps> texturing, and then it is a matter of implementing it into the game properly.</p> <p>Depends on the mod? I tend to leave whatever would be the most annoying for last.</p> <p>The concept is always first.</p>	<p>Yes Story outline before quest</p> <p>Yes Objects before scripts</p> <p>Yes Plan the mod Textures before NPC NPC before quest</p> <p>Yes Cells before exteriors</p> <p>No</p> <p>Yes Modeling before mapping Mapping before texturing</p> <p>Depends on the mod</p> <p>Yes Concept first</p>
<p>Can you describe the normal processes you go through to create a mod?</p>	<p>fun fun fun! (if it's not it lame... I dont make mods when there "lame" to me...)</p> <p>1.get an idea. 2. Say <i>hey, what a cool idea!</i> 3. Try to imagine it in pieces, with various CS objects. 4. Try to create it, or something close to it, in the CS. 5. Procrastinate.</p> <p>A) I decide what I want to accomplish. B) I propose it to the community or to my husband to get a feeling for how difficult it will be and if it's outside my ability to accomplish. C) I get all the dialog into the mod. I find</p>	<p>It has to be fun</p> <p>Define the mod Break into manageable parts Create in CS</p> <p>Define mod Gauge interest in it Determine limitations Define scripting/dialogue Create in CS Test the mod</p>

	<p>front of a screen punching keys repunching keys. and screaming, till you hold your tongue right and it all falls into place)</p> <p>G. release of the mod to your avidly waiting fanbase (crickets and more crickets)</p> <p>H. going back and fixing what you missed.(most people have been very polite about this for which i am eternally grateful)</p> <p>I. and finally, but not least. forgetting how much blood sweat and tears you put into your last project, so that the next EUREKA moment grabs you fast. and you start all over again.(i am in the middle of all this now. its a vicious circle, but addictive. i now mod more than i play)</p> <ol style="list-style-type: none"> 1. Get idea 2. Check PES to see if has been done 3. Check forum threads to make sure no one is doing it 4. If small mod - just do it while idea is hot 5. If large mod spend hours planning and writing notes - agonising over where to place it - check the Morrowind Grid Use Map for a spare place to put a mod - realise there is no where left so pick place which seems right (only a few will download it anyway) <p>For dialogue I use a spreadsheet template i created which replicates the CS dialogue entry screen - this enables me to sort dialogue do spell checking and generally keep track of the story</p> <p>Generally, I'll get an idea (often from dreams), write it down on the ream of paper I keep on my desk, and open up the CS or Visual Studio (it's a modding tool, no lie!). Depending on the mod, I'll start with the basic cells (classes for a program) and work my way up from there. I do the basics of everything, then go back through and test, fix bugs, add details, etc. That repeats indefinitely until the project is done, when it would enter alpha testing. If the alpha succeeds, then I make sure everything is in, and beta test it. If that succeeds and no more</p>	<p>Define the mod Check for similarities to other mods If small, just do it If large, plan mod Check location conflicts</p> <p>Define scripting/dialogue</p> <p>Define the mod Create in CS Define scripting/dialogue Define the basics Test the mod Bugfix Add details Repeat Test/Bugfix Release</p>
--	--	--

	<p>bugs are found, it goes to release.</p> <p>The idea comes first, I think about it for a good long while and refine it. Then I start playing in the CS - if I still like the idea, and if its feasible, I move it to a mental active project status. I usually do a variable amount of consecutive days of work on a mod before taking a break and tackling something else (be it another mod, some art, writing, website maintenance, etc.)</p> <p>I think of an idea or find one someone suggested; then I plan what it would take to make the mod; I gather up any resources needed (like meshes, textures, icons, scripts, etc.); then I start writing the mod. As I go along, I test it to make sure it works on a rudimentary level. After I'm mostly done, I'll ask my wife to test it and then fix anything she mentions. If it passes beta testing, I write a read me and release it, fixing any bugs or adding features later on.</p> <p>Design, Lore & concept, In-game art, Testing, Gameplay & story implementation, Testing. Something like that.</p> <p>I usually like to write a huge list of what's missing and what I'd like to see in the world around me. Then during the process I add / remove things from this "list" until everything's "perfect" in my point of view.</p> <p>build something. test what I've build. repair until it tests correctly then and something else and repeat.</p> <p>Come up with an idea Gather necessary resources Create basic set-up, dungeons, landscape, house Implement dialogue, quests PLAYTEST Clean using Enchanted Editor Release</p> <p>Imaginative, followed by discovery of feasibility or worth, information gathering, item creation, scripting,</p>	<p>Define the mod Create in CS</p> <p>Define the mod Plan the mod Gather resources Create in CS Test the mod Bugfix Write the Read me Release Address errors</p> <p>Define the mod Create In CS Test</p> <p>Define the mod Create in CS</p> <p>Create in CS Test the mod Bugfix</p> <p>Define the mod Gather resources Create in CS Define scripting/dialogue Test the mod Bugfix Release</p> <p>Define the mod Determine feasibility Gather resources</p>
--	--	--

	<p>dialogue, testing, testing, testing... followed by scrap it and start over or get someone to beta test it. Then scrap it or consider publishing it.</p> <p>I touched on this above, but it starts out with a lot of pre-production planning. Outlines, descriptions, notes. Then comes texture work (if necessary). Then comes actual production. Then beta testing. Then release. I haven't actually gotten to those last three steps, incidentally.</p> <p>The 1st couple of processes are most irritating. 1st I imagine what kind of place I want to build. Then I need to scout for resources (since I don't model or texture really, can eventually tweak them a bit). Then organizing them, naming them, and importing them in CS. This is the part where I mostly give up from some mod. Then the best part comes, which is actually working on the mod in cs.</p> <p>1: Always the idea. 2: starting to create concepts for landscape, interiors. Taking notes. 3: Thinking of how the mod would look inside MW 4: Searching for modding resources 5: Landscape, playtest 6: Interiors, playtest 7: Deep detail, playtest 8: Dialogues+Quests, playtest 9: Fixing 10: Showing it to others 11: Releasing</p> <p>I get an idea, ponder whether or not it's possible to actually make, ponder if it's within my skill set, write a basic concept down and get to creating. I really don't put a whole lot of planning into anything.</p> <p>Create the concept, start working, re-work, procrastinate, re-work again and hopefully release.</p>	<p>Create new resources Define scripting/dialogue Test the mod Release</p> <p>Define the mod Plan the mod Create new textures Create in CS Test the mod Release</p> <p>Define the mod Gather resources Organize and name resources Create in CS</p> <p>Define the mod Create concepts Visualize placement Gather resources Create in CS Test the mod Add interior detail Test the mod Complete details Test the mod Define scripting/dialogue Test the mod Bugfix Release</p> <p>Define the mod Determine limitations Create in CS Release</p> <p>Define the mod Create the mod Release</p>
<p>What kind of software knowledge do you need to create a mod?</p>	<p>i'm computer illiterate.. all i can do is mod related stuff. i dont play games cause i dont even get how to install them. stick the CD click next. hope it</p>	<p>CS knowledge</p>

	<p>works. so for me all i need to know software wise was how to use the internet. which taught me the CS, which i guess is part of this question. U NEED TO KNOW THE FLIPPING CS!</p> <p>Intermediate Construction Set knowledge is essential, everything else is optional.</p> <p>Only how your OS and the CS works.</p> <p>Depends on what you're trying to do. For something simple you only need to know the CS basics.</p> <p>At minimum CS knowledge</p>	<p>CS knowledge</p> <p>CS knowledge</p> <p>CS knowledge</p> <p>CS knowledge</p>
<p>What are some of the things you should know before you begin a mod?</p>	<p>that when u aim to high it's never gonna happen. and where to get the information you'll need or the models or whatever.</p> <p>Knowledge of the CS, your preferred file manager. Basic knowledge of an archive program for packaging and distribution.</p> <p>Whether its possible. How hard it is to do. Whether someone has already done it.</p> <p>What your limitations are: time, experience, and most importantly, your desire to see a mod through. Don't try to make a TC if you're not going to finish it.</p> <p>What you're trying to do, exactly. If you stray too much, it's easy to make the mod much larger than it was originally going to be.</p> <p>What you are trying to make, and how it will fit into MW (as in avoiding conflicts)</p>	<p>Don't overreach Needed resources</p> <p>CS knowledge Compression program</p> <p>Define feasibility Determine uniqueness</p> <p>Define limitations</p> <p>Define the mod Prevent scope creep</p> <p>Define the mod Avoid in-game conflicts</p>
<p>What are some things you need to consider when working with a team in a group project?</p>	<p>their skillzorz. also people who are like "we SHOULD do this NOW" are bad, when people offer advice it's great. when people "force" advice onto you it's quite annoying. and dedicated people... u need them. oh and ppl u have to teach everything to first isn't helpful (i can hardly talk).. i guess for this section i wouldn't know much though...</p> <p>You'll rarely make what you want, you need to keep scope down or you'll never</p>	<p>Define the skills of team Define personality conflicts Determine the dedication to finish the mod</p> <p>Avoid scope creep</p>

	<p>finish. Don't be the next big idea modder.</p> <p>How much time you have to devote to the mod. What has already been accomplished, what needs to be accomplished, and whether your skillset is necessary. What the atmosphere of the team is (serious and goal-based, friendly and casual, etc.). How well the team aspect of the mod has been thought out, and how it can be improved (e.g. how communication happens, who is in charge of what, etc.).</p> <p>How the division of labor will be assigned, who will be the team leader, and how disputes between members are handled. Also, can you work with your fellow teammates?</p> <p>You have to know what it is the team project needs. And stay away from what it doesn't need.</p> <p>Can you all work together, and who's organising the project.</p>	<p>Define schedule conflicts Track mod progress Define the skills of team Determine the dedication to finish the mod Define team dynamics Define communications Define team leader</p> <p>Define task assignments Define team leader Define dispute resolution Define personality conflicts</p> <p>Avoid scope creep</p> <p>Define personality conflicts</p>
<p>What kinds of conflicts or problems are common when creating a mod, either as an individual or in a team?</p>	<p>other mods. people disagreeing on doing something. modding mana goes dry etc etc</p> <p>Group cooperation, team member's areas of skill and interest, time and dedication per member, mod scope and how best to split it among the team, how to keep the team running smoothly.</p> <p>Allow me to comment on the team, since thats what I have the most experience in. People who join and don't contribute anything (with an optional "omg i so awesome i can model and texture and make quests" before disappearing). People who can't abandon their ego and personal ideas in favor of the team. People who don't do the research and ask dumb questions, or suggest things that are not within the scope of the team. In terms of conflicts, its mostly about the scope/nature of the mod and how it should be organized and lead. Deciding who should work on what can also be an issue, as can merging everything</p>	<p>Working on other mods Personal conflicts Lack of interest</p> <p>Personal conflicts Lack of appropriate skills Lack of commitment Scope creep Task assignment</p> <p>Lack of commitment Ego Lack of team cohesion Lack of research Lack of preparation Lack of leadership Task assignment</p>

	<p>together.</p> <p>I don't work on teams, so I won't answer that outside of duplication of effort and personality conflicts. As for individual made mods: there are so many mods out there already, that you always run the risk of changing something another mod has changed and causing a conflict between yours and the other.</p> <p>Individual? The problems are endless. Technical issues within the CS, that decide to pop up on their own, and apparently for no reason. Problems with the mod itself, like having a ghost object in your game, but it's not there in the CS cell. All kinds of problems, and since you're alone, they can be big problems. Problems as a team? There usually isn't much to complain about. As a team you can overcome all technical issues together, and gather up some great ideas. Unless the team gets too large, then you have to deal with too many egos. Then the idea gathering turns into idea rejection. It's very counterproductive.</p> <p>Using the same locations/ file names as other modders.</p> <p>Script bugs, ID or landscape conflicts or lack of resources, or personality or vision/design conflicts in a team.</p>	<p>Duplication of effort Personal conflicts Mod location conflicts</p> <p>CS technical issues Team size Ego</p> <p>Mod location conflicts</p> <p>Bugs Mod location conflicts Personal conflicts</p>
<p>What kind of resources do you need to create a mod?</p>	<p>the interesting ones that fit your ideas. like if you were making a telvanni mod imperial resources would be useless. and also, lore... lore is so important, so the imperial is a resource i guess?</p> <p>It depends on the mod, but everything from external C++ libraries to little 32x32 TGA icons.</p> <p>Depends on what you mean by resources. I'd say people and ideas are the most important ones.</p> <p>Only the CS unless you're making models or textures, then you'll need a modeling program like 3DMax or Blender for models and Gimp or Photoshop for textures. The biggest</p>	<p>Lore Appropriate themes</p> <p>Programming software Icons</p> <p>People Ideas</p> <p>CS Appropriate software Time</p>

	<p>resource you need is time.</p> <p>If it's simple CS objects, nothing at all. Depends on what you're trying to do. Lots of people have made resources to use, so you can literally look over those for days on end. If you're working with some competent modders, they can just make objects as they're needed.</p> <p>Game lore, reference material, tools, imagination, and time and patience.</p>	<p>CS Resources</p> <p>Lore Reference materials Imagination Time Patience</p>
<p>When creating mods, what programs do you most commonly use?</p>	<p>Irfanview, NifTexture, DDS converter, nif_tx_Stripper (new to me but I like it very much), photoshop 7.0 (old and clunky but predictable), The Gimp, nifskope, BSA Browser, Reorder Mods, and Audacity for sound.</p> <p>nifskope, blender, CS, skype (for asking people questions) and mozilla firefox (to find stuff easy) and usually i have the datafiles on the CD open to look through. and before i forget photoshop!</p> <p>Irfanview, TESCS, Nifskope, TESAME, BSA Browser, GOOGLE, Enchanted Editor, Wrye Mash, TESPCD, MLOX.</p> <p>The Morrowind CS, Photoshop CS4, Gimp, TESAME, TESDTK, Trillian, BSA Browser, 7-Zip</p> <p>TESCS, 3DSMax, Wrye Mash, Nifskope, TESAME, WinRAR, MGE, Fraps, Yacoby's Mesh Generator.</p> <p>7-Zip, DDS Converter 2, FRAPS, Photoshop, RADTools, UltraEdit, Windows Grep, WTV (for quick viewing of DDS files), NifSkope, Morrowind AnimKit, BSA Browser, Wrye Mash, Morrowind Enchanted Editor, TESPCD, tes3cmd, TESTool, TESCS, MWEdit, MGE</p> <p>Not sure if you also want listed the support programs that allow some of these apps to work - like Python, wxPython, Strawberry Perl, VC 2005 & 2008 Redistros, DirectX Redistro, etc.</p>	<p>3DSMax 7-zip Audacity Bink Blender BSA browser Cdex Cinema 4D Corel Painter 10 Crimson Editor CS DDS converter Deep Paint 3D DirectX Redistro Enchanted editor Excel Fiddler 2 Firefox Fraps GMST vaccine Google HxD IDA Pro Irfanview MCP Mesh Generator MGE MLOX Morrowind AnimKit MWEdit Nif Stripper Nifskope Notepad OllyDBG Paint.net Photoshop Pidgin Programmer's Notepad 2 PSPad</p>

	<p>TES Construction Set MW Edit - MWSE scripting Enchanted Editor - error correction and clean up TESFILES - packages everything up in to a folder so nothing gets left out 7z - to package the file blender the Gimp</p> <p>TES Construction Set MWEEdit Enchanted Editor some home-rolled tools to do little things</p> <p>3DS Max 5.1 and 2010 x64 Cinema 4D (r10.1, occasionally) Deep Paint 3D ZBrush/Sculptris</p> <p>Photoshop CS4 (extended) Corel Painter 10 Wacom tablet</p> <p>Soundbooth</p> <p>Visual Studio 2008 Pro (all the time, and sometimes 2010 Ultimate) OllyDBG and IDA Pro (5.5) (for working directly with the engine code/memory and occasionally with my apps) Fiddler 2 (I think?) for HTTP debugging (used with CnE only, so far).</p> <p>Pidgin and Skype for chat (Pidgin for IRC and MSN, mostly) WinRAR Notepad/PSPad/Programmer's Notepad 2 HxD occasionally for hex-editing the data files WinMerge for diffing data and resource files</p> <p>I've used notepad but I prefer Crimson Editor. I also neglected to add Enchanted Editor</p> <p>Graphics:</p> <p>The Gimp Irfan Viewer Paint.net (for icons) DDS Converter (rarely)</p>	<p>Python RADTools Reorder mods Sculptris Skype Soundbooth Strawberry Perl Tes3cmd Tesame TESDTK TESPCD TESTool The Gimp Trillian UltraEdit Visual Studio 2008 Windows Grep Windows Movie Maker WinMerge WinRAR WTV wxPython Wyre Mash ZBrush</p>
--	--	--

	<p>Wacom Tablet Software</p> <p>Nifskope - for modelling the texture</p> <p>Sound:</p> <p>cdex - for ripping to Wav Audacity - mixer</p> <p>Movies:</p> <p>Fraps The Gimp (processing stills) Windows Movie Maker - both XP and Win 7 versions Bink - Rad Tools free version</p> <p>Modding:</p> <p>GMST vaccine (not really a program but hey I always use it) Notepad - for drafting scripts Excel - for managing dialogue and quests (also for spell checking) TESAME TESDTK (don't ask) TesTool (when desperate) TESPCD Wrye Mash MCP MGE (for movie making) Windows Media Player (I mod to music) Internet - for research, questions, image search, uploading</p> <p>Wacom Tablet 3ds max Gmax Photoshop nDo Sculptris Xnormal Meshlabs Nifskope Sceneimmerse TES Construction set MGE MCP</p> <p>Photoshop 7 (Haaaate the newer versions. Hate them.) Blender NifSkope</p>	
--	---	--

	Notepad Blender Gimp NIFSkope CS	
--	--	--