

PlayStation®Home – Key Design Principles for Content Development

Table of Contents

1 Introduction	3
2 Know Your Audience	4
3 Provide Goals and Objectives	5
4 Simplify Search	6
Satisficing	6
Chunking	7
Organization	7
5 Utilize Navigation	9
Basic Navigational Elements	9
Signs	12
Maps	13
Wayfinding	14
6 Highlighting Interaction Points	16
7 Minimize Load Times	19
Notification	19
Minimizing	20
Distractions	21
8 References	22
Universal Principles of Design	22
Washington Monument	22
Emergency Exit Sign	22
Bay Area Rapid Transit (BART) Map	22
Steve Souders	22
Load Times	22
Grand Theft Auto 4	22
Mile Marker	22

1 Introduction

There are a myriad of ways to approach the creative design process when developing content for any platform and PlayStation®Home is no exception. What makes PlayStation®Home stand out, however, is the ability to create content that leverages the strengths of 2D web gaming within the realm of a 3D game development platform. To help you in this process, this document describes some important key design principles you can utilize when designing your content.

Of course, many books have already been written on the topic of design principles and best practices. Two books from which this document pulls heavily are *Universal Principles of Design* (by William Lidwell, Kritina Holden, and Jill Butler) and *Don't Make Me Think* (by Steve Krug). See these books to find a number of useful and important design principles that can be applied to your design process. As a result, it should be noted that the goal of this document is not to list all useful design principles, but to highlight those relevant to developing interactive gaming content on PlayStation®Home.

2 Know Your Audience

There are two design principles that, when used together, will enable you to better understand and design content for PlayStation®Home's specific audience: User **Personas** and their **Hierarchy of Needs**.

Personas: A technique that employs fictitious users to guide decision making regarding features, interactions, and aesthetics. (Lidwell, Kritina, & Butler, 2010)

Personas enable you to identify important user types and create user profiles based on those types. For example:

Mr.face, a "power user," is a 22 year-old first-person shooter fanatic. He is a hardcore player, a college graduate with a steady job and after work he has plenty of free time to play FPSs like Modern Warfare – it probably helps that he is also single. His financial independence allows him to spend \$50+ a month on new games and he plays 20+ hours a week.

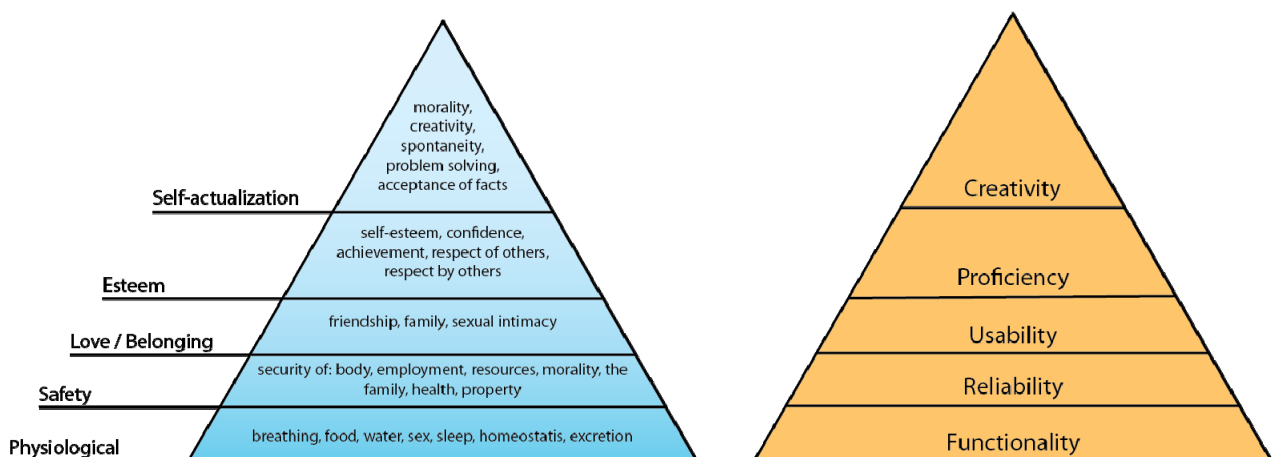
Once you identify your casual, core, and hardcore users and fill in specific details like monthly income, personal information, habits and interests, etc. you can begin to ask questions in order to qualify and quantify decisions regarding feature sets and the design of your content.

This leads us to our second design principle. Abraham Maslow's hierarchy of needs – a theory of psychology "describing the stages of growth in humans" (Maslow's Hierarchy of Needs) – provides an excellent framework to understand what elements of your design are more important or less important to complete based on the needs of your users.

Hierarchy of Needs: In order for a design to be successful, it must meet people's basic needs before it can attempt to satisfy higher-level needs. (Lidwell, Kritina, & Butler, 2010)

The authors of *Universal Principles of Design* adapted Maslow's hierarchy of needs for use when designing products. They replaced Maslow's list of needs with concepts more appropriate for product design. Use these needs to identify both where you are in the design/development process and what your users, based on the personas you create, will want from your design.

Figure 1



3 Provide Goals and Objectives

As unlikely as it may seem, decision making seems to be a common problem for most users. Ideally users would choose to participate in activities that encourage social interaction, are fun, and make you money – yet this is not always the case. The number of options available is not an issue – there are multiple locations to visit, games and activities to participate in, and people to talk with, etcetera, and the decision making process is not an issue either – users can and will find things to do. The issue is: lack of direction.

Imagine the following scenario:

Upon starting a game you find yourself in the middle of a dark alleyway. A crumpled dead body lies at your feet the skin is black as if burnt. On the ground lies a white card with a solid red hand.

What do you do now? With little information to go on it is hard to know exactly what to do, what goals to achieve, or questions to ask, and what steps to complete in order to achieve said goal or goals.

Unfortunately, this is the state that many home spaces place users in – when the user spawns inside a space no directions are given, or objectives are specified – the user is left to their own devices to decide what to do. While this may work for some games (e.g. simple shooters like Geometry Wars, or sandbox games like Minecraft, etc.) this does not achieve the best results within Home.

Re-imagine the same scenario, but now with additional information:

You start a new game. The credits roll, music plays, and you see a man wearing a red coat walking down an alleyway. It's dark. A fine mist hugs the ground hiding the filth littering the street. Rats can be heard scurrying underfoot through the trash.

A shadow darts out and there is the flash of a knife, an arm outraised, poised to strike. Suddenly, a flash of light! The intruder falls back, his skin turning a burnt-black as a faint scream escapes his lips. He crumples in pain and dies.

There's a noise in the alley – you've stumbled over a glass bottle – startled, the man in the red coat looks back over his shoulder, pauses, then runs away, disappearing in to the night. A small white form drops to the ground where the man in red was standing. As the camera approaches the scene you see a card: a solid red hand on a white background.

The game starts and you take control. Prompts to walk around and learn the controls. Another prompt asks the player to pick up and inspect the card on the ground.

The objective is clear: find more information about the man who left the red-handed card.

4 Simplify Search

There are basically two types of visual search: simple and complex.

Simple search occurs when there is a reduced set of objects to search, for example: finding a single target among of five versus a single target among twenty-five other objects. Simple search also occurs when a target stands out from its surroundings due to one different visual feature, for example: a distinct color change from the surrounding environment – a red apple within a black and white photo. Complex search occurs when a target shares two or more visual features and differs in another. Complex searches are slower to perform because the observer must iterate over each item multiple times in order to find the differing visual feature. Features can include color, size, orientation, shape, motion, etc. This suggests that by using a distinct color or shape we can make simple search quick and easy for users. (See [Highlighting Interaction Points](#).)

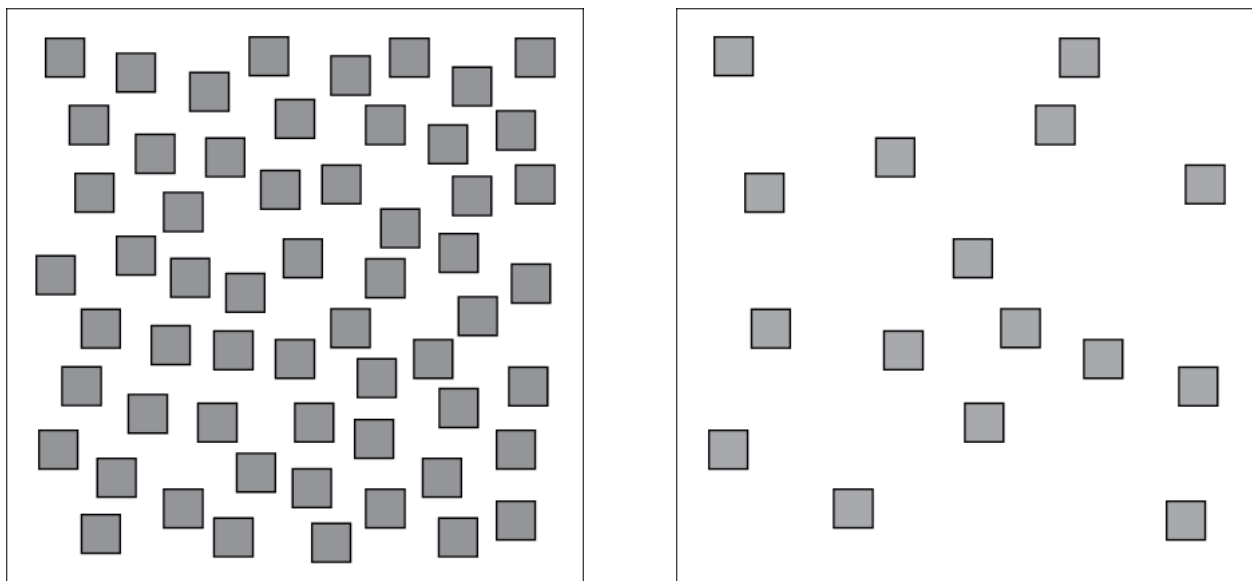
Theories support simple and complex search and suggest ways in which we can capitalize on this process. Feature Integration Theory suggests that visual features such as color and shape are registered early, automatically, and in parallel when the brain analyzes the visual field. In addition, there is evidence that performing complex searches require further, later visual processing in serial in order to integrate between targets to find the differing visual feature. Guided Search Theory suggests that attention can be directed in two ways: bottom-up processing (stimulus-driven) and top-down processing (user-driven). Therefore, by providing users with objectives we can activate user-driven search processing to direct attention toward a particular type of stimulus or set of visual features. (Eye tracking studies supports this too.)

There are other ways to simplify search, too, such as satisficing, chunking, and organization.

Satisficing

Of these two examples, which do you think is easier to search?

Figure 2



If you picked the image on the right you are on the correct path. However, there are still too many objects, and they are too uniform in size, shape, and color to easily, at a glance, identify any single one. This is

important to note because, when presented with multiple options of similar importance people will tend to **satisfice**.

Satisficing: The preference to settle for a satisfactory solution, rather than pursue an optimal solution. (Lidwell, Kritina, & Butler, 2010)

In environments with multiple targets (i.e. objects, locations, items, etc.) of similar importance this is likely to result in the selection of a target within close range rather than an intended, expected, or optimal target.

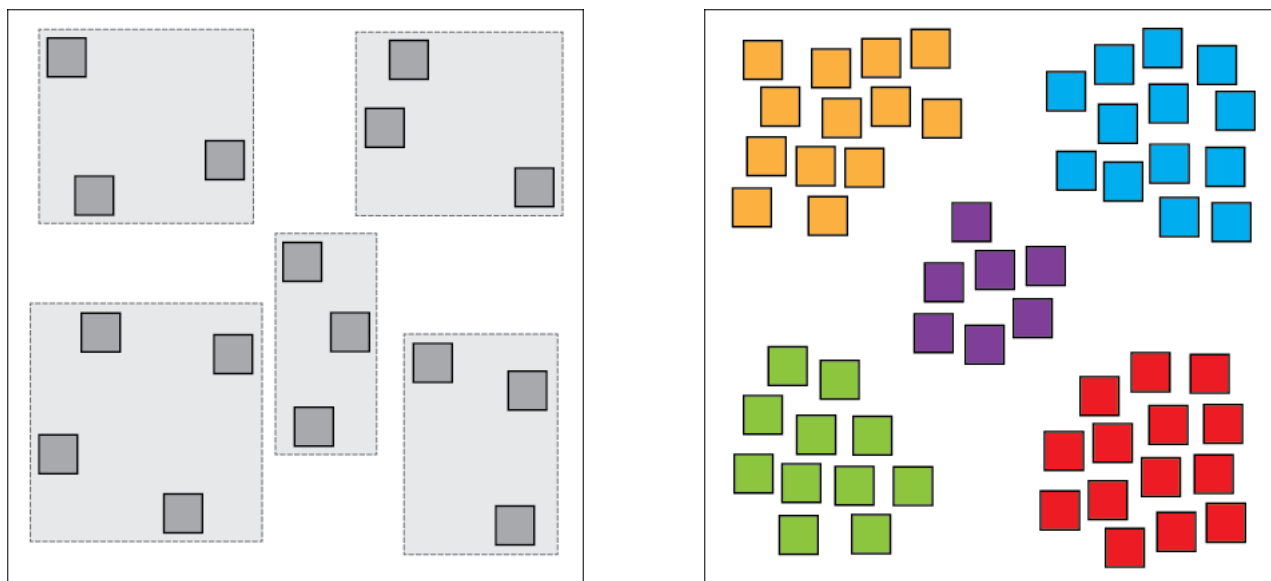
Chunking

To simplify the search process, reduce the set of potential choices dramatically. Grouping objects is one way to accomplishing this. Eliminating objects will also have the same effect. Use **chunking** to reduce the number of options into an 'ideal' range.

Chunking: A technique of combining many units of information into a limited number of units or chunks, so that the information is easier to process and remember. (Lidwell, Kritina, & Butler, 2010)

Chunking is based on research that shows people can only hold around 4-7 things in short-term memory. If the list of items grows your ability to hold additional items in memory degrades and you begin to lose or forget items in the list. Use this to your advantage and simplify environments, interfaces, or any other items a player may search in order to maximize users' capabilities to "see" all available options and effectively choose among them.

Figure 3



Organization

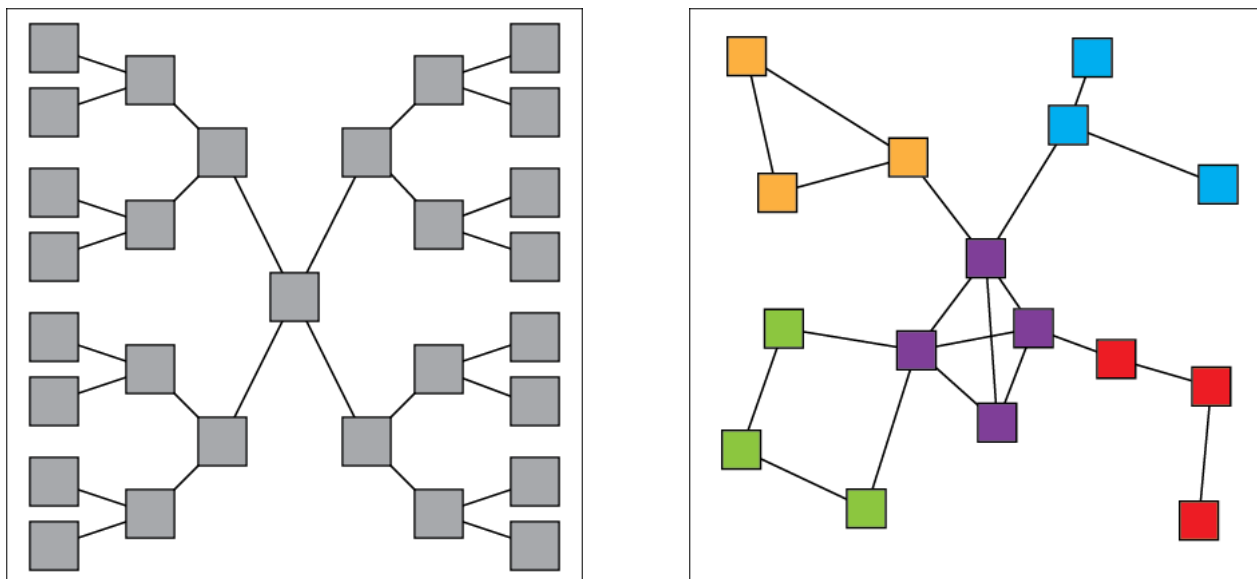
One easy way to simplify search is to organize content within a **hierarchy**.

Chunking: A technique of combining many units of information into a limited number of units or chunks, so that the information is easier to process and remember. (Lidwell, Kritina, & Butler, 2010)

Hierarchical structures can also be used to limit the number of choices available enabling users to prioritize their search. For example, organizing related content into categories and letting the user choose

between the categories before presenting additional info. Organization can also apply to the layout of your environment and based on the layout the nodes of your hierarchy will have different inherent priorities. For example, the nodes in linear, circular, hub-and-spokes, and tree-shaped hierarchies all have different inherent priorities.

Figure 4



5 Utilize Navigation

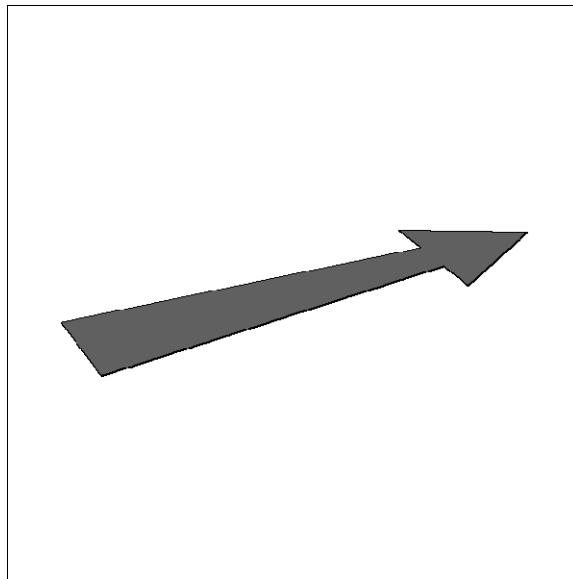
In many ways navigational difficulties are linked with the problems of decision *making* and *searching*. When a user enters a new environment if she is not immediately presented with objectives to complete she will start searching for something to do. You have probably done the same thing each time you've entered a store without a concrete objective in mind. However, if you do have an objective, for example: finding a critical ingredient for tonight's dinner, if there are few navigational elements enabling you to find your item it is likely you will become increasingly frustrated, eventually leaving the store. Improving the way in which users navigate an environment will minimize the frequency with which they get lost or frustrated, improving the user experience.

Basic Navigational Elements

To better understand navigation it is good to identify key elements we utilize when moving through an environment. Kevin Lynch outlines five basic navigational elements in his book *The Image of the City* where he identifies fundamental elements of city design. These key elements are paths, edges, districts, nodes, and landmarks.

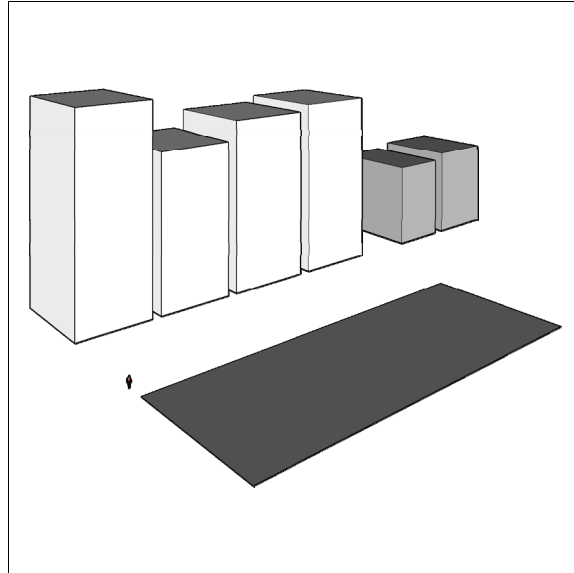
Paths are generally any horizontal surface we use to traverse from point to point using any mode of transportation. For example: store aisles, sidewalks, trails, and roads are all paths.

Figure 5 Paths



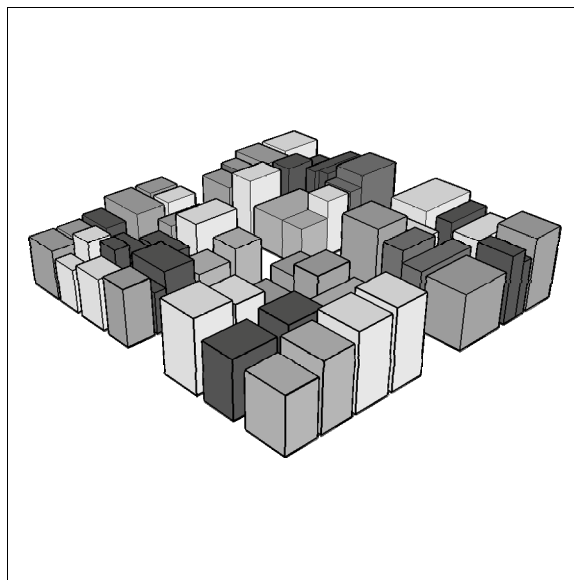
Edges are the vertical, otherwise non-traversable surfaces within an environment that separate or provide boundaries for paths. This includes: fences, walls, buildings, shorelines, etc.

Figure 6 Edges



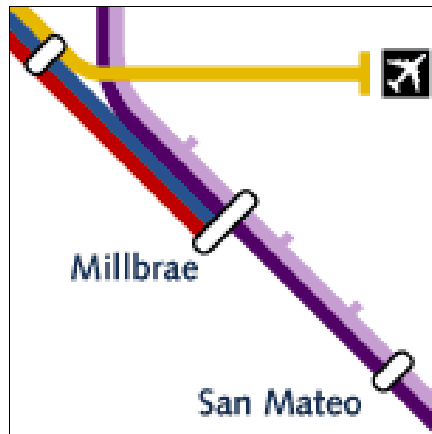
Districts are essentially neighborhoods; collections of paths and edges that relate to one another. In Home neighborhoods could be different regions within a single space, or a whole space with a collection of related spaces that 'surround' it.

Figure 7 Districts



Nodes are basically intersections between the same or different types of paths. The following image displays the intersection of transit nodes between BART and CalTrain at the Millbrae station.

Figure 8 Nodes



Landmarks tend to be objects whose shape, size, and/or color merge to form distinct points of reference. We tend to think of landmarks as large buildings like the Washington monument, but they can also be small things like a distinct red mailbox on the corner of your street or a memorable grocery store. Half-Life 2's use of the Citadel is probably one of the most easily memorable uses of a landmark within a game to date.

Figure 9 Landmarks



Figure 10 Half-Life 2's Use of The Citadel

Signs

Adding signs within your environment are a simple and easy way to enable users to easily navigate your space. They are unique in that they synthesize shape, size, color, and location to represent complex ideas, generally without words – essentially a specialized mini-landmark. When well-designed they are memorable and easy to understand at a glance; a good example of the design principle: *picture superiority effect*.

Figure 11

Picture Superiority Effect states: “pictures are remembered better than words.” (Lidwell, Kritina, & Butler, 2010)

By combining the PSN bag icon with geometry within your screen you can pair the concepts of shopping and the PSN commerce system with that specific location as the place to go in order to accomplish the purchasing of items. This is much more memorable than if you were to read a list of important locations on a notification board. However, what happens when that the locations on the notification board are placed in proximity to one another as is reflected in the environment? The sign becomes a map.

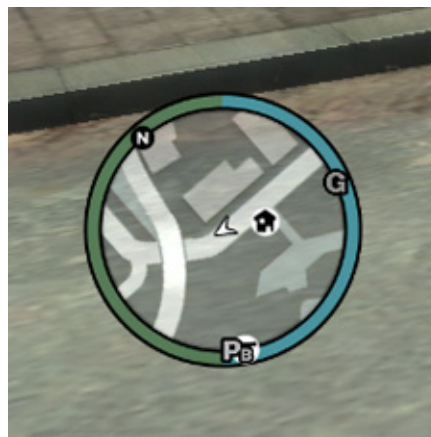
Figure 12



Maps

Maps can be another good addition to your space. Many exploration games include a mini-map as an element of the heads-up display enabling players to quickly and easily reference their current location based on a map of the environment. Most large-world or exploration focused games include mini-maps including the Grand Theft Auto series, RPGs like Neverwinter Nights, and MMOs like World of Warcraft and Guild Wars.

Figure 13

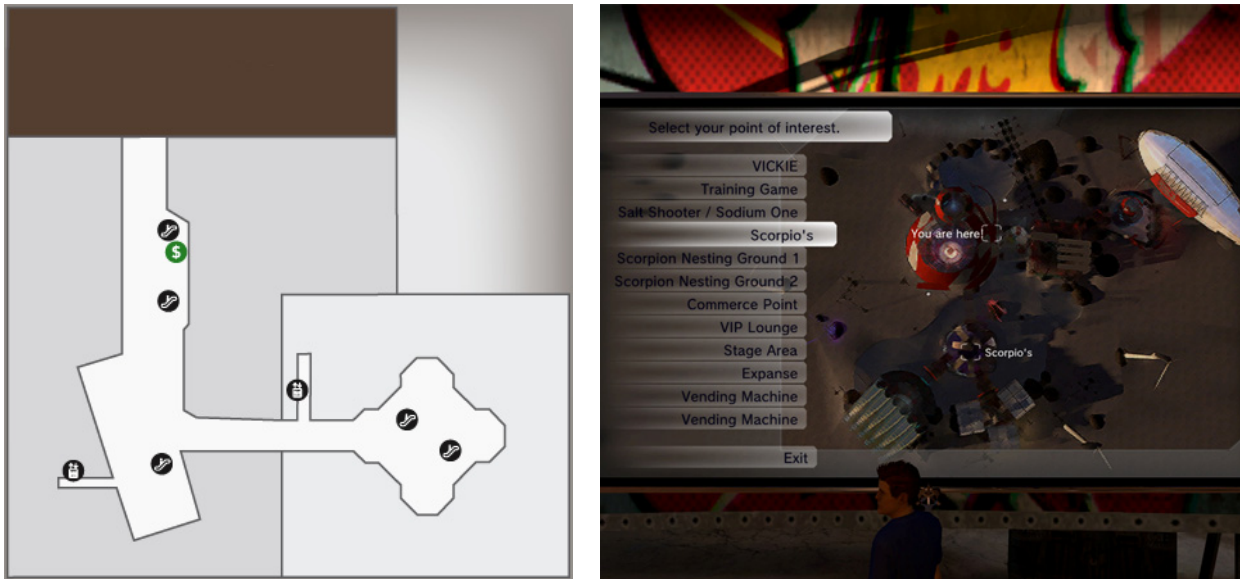


Maps are especially useful for enabling users to reference different varieties of information based on location-specific awareness. Collecting different information types in this way is an effective use of another design principle: *layering*.

Layering is “the process of organizing information into related groupings in order to manage complexity and reinforce relationships in the information.” (Lidwell, Kritina, & Butler, 2010)

Through layering you can relate useful information like the player's position with other information that may be directly or indirectly useful. For example, most in-game mini-maps display the player's position and the position of the current game objective, key locations like safe houses, and sometimes the positions of enemies, opponents, and/or friends. However, taking a page from mall directories, you could also display genre-specific content, commerce points, entry and exit points, special points of interest or even highlight locations that require special privilege. The Sodium Hub includes an interactive map that might be improved if, when passive, it included icons for most of the key interactive items a player might be interested in, including one for the commerce point as well as an obvious character marker. It might also be useful to see the locations of all other players as well as to be able to toggle their names so you knew who they were.

Figure 14



Wayfinding

When traveling we are always searching and evaluating our environment. *Wayfinding* is the process we use to guide how we search and evaluate our environment in order to arrive at a selected destination. As a design principle this is something we can use to our advantage.

Wayfinding is “the process of using spatial and environmental information to navigate to a destination.” (Lidwell, Kritina, & Butler, 2010)

Both *Universal Principles of Design* and *Don't Make Me Think* basically agree on four stages of navigation: 1) orientation, 2) route decision, 3) route monitoring, and 4) destination recognition. These are the steps a person makes as they evaluate their progress when traveling between two points.

Orientation – recognizing navigational elements to determine your location and direction. Capitalize on this process by *chunking* the environment, further subdividing it into smaller sections, then use landmarks and signage to easily identify important areas and to generate uniqueness.

Route Decision – evaluating orientation information in order to decide on a destination. Simplify by minimizing navigational choices, this will also maximize the user's tendency to *satisfice*. Adding signs and prompts at natural decision points like intersections helps too. Overall shorter is better, as route complexity is not a large factor.

Route Monitoring – evaluating route decisions to monitor progress towards a destination. Enhance route monitoring by using paths to connect destinations. Even just a visual distinction will help guide users. Also, if all paths have a distinct beginning, middle, and end it is easier to gauge your distance along that path. Along those lines, it is also useful to create distance gauges. Distance gauges enable users to easily monitor progress, as they progress from point A to point B. Though the mile marker comes easily to mind, foreshadowing future locations by providing distant views of those locations works equally as well – the foreshadowing within Uncharted’s levels demonstrate this well. Providing visually enticing distant gauges also provides breadcrumbs which lure people along the path.

Figure 15



Destination Recognition – identifying when you’ve reached a destination. Ideally locate destinations at dead-ends as this rewards users for navigating the environment. Dead-ends without worthwhile content waste users’ time, are anti-climatic, and are a disincentive for searching the nooks and crannies. Remote areas of a space with only a set of seating areas to show for the player’s travels should be left on the cutting room floor. Destinations can also be located mid-path or at intersections between paths. If this is the case use vertical and visual elements as barriers to disrupt the traffic flow. Using a landmark will also provide identity information making the location easier to find.

6 Highlighting Interaction Points

There are a number of essential cues used to evaluate and identify the world around us. “The visual sensations of color, shape, motion, or polarization of light, as well as other senses such as smell, sound, touch,” (Lynch) etcetera are all used consciously and unconsciously in many aspects of our lives including searching, navigating, and decision making. We also use them when we evaluate what we want to interact with.

Games rely most on color, shape, motion, and sound. Use one or more of these cues in order to focus users’ attention on a particular point or item of interest. The design principle that reflects this best is called *highlighting*.

Highlighting is “a technique for bringing attention to an area of text or image.” (Lidwell, Kritina, & Butler, 2010)

Overall, highlighting is best used sparingly. *Universal Principles of Design* specifies “no more than 10 percent of the visible design.” With text: **bold** is the best way to call attention without adding a lot of noise. CAPS are another way to differentiate text and should generally be used as an alternative to using a different typeface. Color can be useful, but only in combination with other techniques. Inversing is mentioned with regards to text and icons, but is likely harder to apply to 3d elements within a scene. Blinking is an extremely “powerful technique for attracting attention” but can also be very distracting – adding a lot of noise to a scene.

Blinking can be used successfully, however. A good example was used on the words “PLAY” directly above the COGs single-player attract object when it was displayed within the SCEA Hub. Here PLAY pulsed slowly between a bright white and a dim grey; it drew the players’ eye, but not to the expense of the surrounding environment.

Figure 16



Color as a design element can also be particularly effective for highlighting elements within a scene when used sparingly and when combined with other techniques. One example of this is the red-orange 76 ball that highlights a specific location as having a 76 gas station.

Color “is used in design to attract attention, group elements, indicate meaning, and enhance aesthetics.” (Lidwell, Kritina, & Butler, 2010)

Figure 17



When vibrant and in contrast to the surroundings, color is an effective attention getter. The World of Warcraft yellow exclamation point is a good example of this and is also distinctive because of its shape. The RC Rally icons are easily recognized as belonging in the same group because they are both red. SCEA Home's controller icon has a distinctive color that is shared with SCEA's other core Home spaces. It can also be grouped with the RC Rally icons because of its shape. Aurora's information point is highlighted both through color and motion – it rotates around its vertical axis.

Figure 18

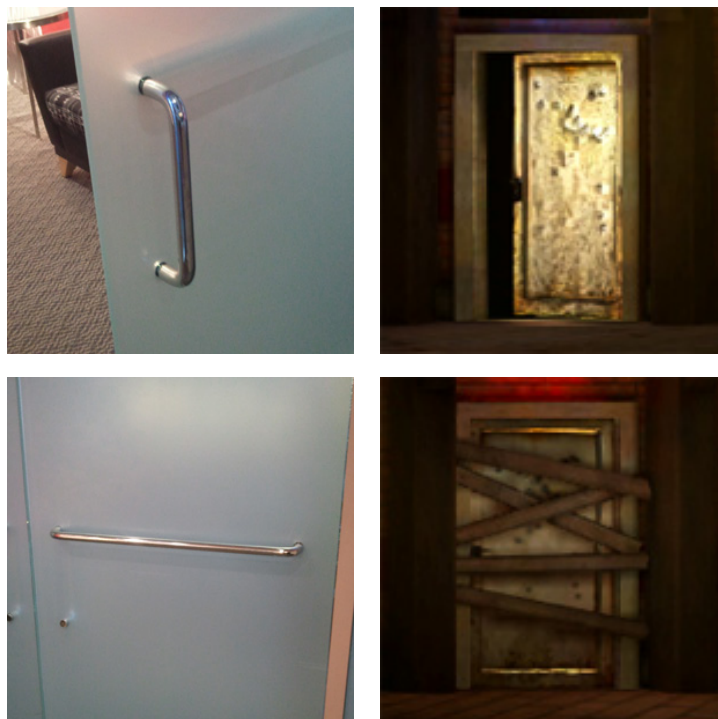


In addition to highlighting and the use of color, *affordance* and *mapping* are other useful design principles that can be applied to interaction points.

Affordance is “a property in which the physical characteristics of an object or environment influence its function.” (Lidwell, Kritina, & Butler, 2010)

Donald Norman in *The Design of Everyday Things* uses door handles as an example of affordance. Doors with horizontal bars afford the ability to push therefore signifying that the door opens outward. Doors with vertical handles afford the ability to pull therefore signifying that you must pull in your direction to open the door. This also works for elements within virtual environments. Doors as elements within an environment signify a change in location, whether internal to the environment, for example: a door from a bedroom to a hallway, or external to the environment, for example: from outside on the front porch to the inside entranceway of a house. In addition, the open door inside the Bootleggers lobby affords movement between locations whereas the barricaded door does not.

Figure 19



Mapping is “a relationship between controls and their movements or effects. Good mapping between controls and their effects results in greater ease of use.” (Lidwell, Kritina, & Butler, 2010)

Norman also described how mapping is used within interfaces and how good mapping is best seen through similar behavior, layout, or meaning. For example pulling up on the button for an electronic car window causes the window to raise, while depressing it causes it to lower. While the relationship between controls like buttons, levers, etc. mapping to PlayStation®Home’s controls are 100% accurate, it is important that any virtual controls mimic how those controls would normally work within the real-world. This will maximize the familiarity and ease with which users interpret the meaning of your controls when interacting with them in the virtual environment.

7 Minimize Load Times

Google and other big tech companies have been concerned with the speed and performance for a long time now. So, it is no small wonder that web performance optimization has been an important issue. Steve Soulders¹, a web performance expert working at Google, has discussed at conferences and published on his blog a number of examples that demonstrate the importance of reducing load times.

- **Amazon:** 100 ms delay caused a 1% drop in revenue.
- **Google:** 400 ms delay caused a 0.59% decrease in search requests per user (earlier tests list larger numbers).
- **Shopzilla** succeeded in reducing the loading time from 7 down to 2 seconds, whereby the conversion rate increased by 7-12%, they observed a 25% increase in page requests, they were able to retire 50% of their servers, thus saving energy costs.

Home is not a webpage, however. So even though we know load times are important – as the information shows – we must find other ways to improve the experience that do not include optimization. The best ways to do this are to **notify** users when loading is still occurring, to minimize the time spent loading by utilizing available functionality, or **distract** them completely.

Notification

Telling or showing users that a load is occurring, through text or via an icon, is the easiest way to communicate the current state of your content. Loading icons became standard once graphical user interfaces were added to computer operating systems in the 80s and 90s – the hourglass in early versions of Microsoft Windows or the pinwheel in Mac OS, for example – and computer games do not tend to deviate. Be aware, however that displaying a loading icon as a 3d object may force users to learn what your ‘icon’ means. You can minimize the learning curve, though, with some consistency and a little text. Being consistent with other loading icons makes it easy for users to recognize and understand the meaning behind yours and if there is any doubt you can always clarify by adding text within yours.

Consistency states: “the usability of a system is improved when similar parts are expressed in similar ways.” (Lidwell, Kritina, & Butler, 2010)

Designs can be consistent in four different ways: aesthetic, functional, internal, and external. The following icons are all aesthetically consistent because they all look and behave similarly. The icons used by the Home client and Sodium 1 & 2 all contain transparent elements on the outside of the circle with one highlighted element moving in a clock-wise direction². They are functionally consistent too, which enables users to easily recognize and understand this functionality when it is repeated within your game. The Sodium 1 & 2 icons actually have “Loading...” as text within the icon, and in multiple languages, for exactly this purpose.

¹ A number of the examples are listed in his JSConf.edu talk: <http://blip.tv/file/3060565>

² This clock-wise element is actually very important – it mimics the movement of the hands on a clock which you subconsciously relate to the passage of time.

Figure 20



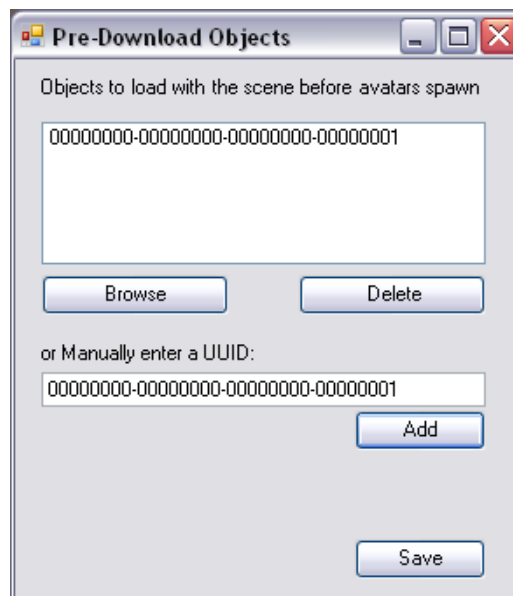
Internally they are consistent with one another – the Sodium 1 & 2 icons look the same but their color schemes are consistent with the color for each game. Externally they are all consistent with loading icons elsewhere.

Minimizing

Within Home there are different sets of functions that help you to minimize load times: pre-downloading objects, scripting scene downloads, and scripting scene transitions.

Pre-downloaded objects are downloaded with a scene when a scene is requested by the user or through script. To specify which objects will be downloaded, add them to the “Pre-Downloaded Objects” list within the scene editor. These objects though downloaded with a scene are still loaded after a scene launches.

Figure 21



In the Scene Editor go to: **Edit > Edit Pre-Download Objects List.**

The Lua APIs for *SceneDownload* enable you to quickly and easily determine if a scene is cached and up-to-date, get its download size, and to start downloading it. You can also check to see what scene is current downloading, check its progress, and cancel if need be. Use these APIs to download scenes that you know the player may want to visit next or may be required to visit. This is especially useful for community hubs that link to one or more game scenes. If you are providing players with objectives to visit

those scenes then by downloading the scenes in the background you will minimize or possibly eliminate wait times that might have occurred otherwise.

Figure 22



Distractions

Distractions can prove to be useful when used effectively. The Lua APIs for *SceneTransition* enable you to replace the default loading screen with a custom scene transition designed by you. This could be as complex as scripting the player as they walk through a door or as simplistic as displaying a looping movie or a series of scripted loading screens.

These distractions hide the fact that a scene or object may loading or downloading. The Aurora space from nDreams uses this concept by giving first-time users a movie-like introduction to the environment. The Sodium spaces from Lockwood Publishing provide the experience of transitioning between spaces by displaying a short movie prior to relocating the user, however they released this functionality before the *SceneTransition* API was available.

Figure 23



8 References

1. http://en.wikipedia.org/wiki/Maslow's_hierarchy_of_needs

Universal Principles of Design

Authors: Lidwell, Willian; Kritina, Holden; Butler, Jill
Published: 2010; Beverly, MA by Rockport Publishers

Washington Monument

http://en.wikipedia.org/wiki/File:Washington_Monument_Dusk_Jan_2006.jpg

Emergency Exit Sign

http://en.wikipedia.org/wiki/File:PublicInformationSymbol_EmergencyExit.svg

Bay Area Rapid Transit (BART) Map

http://commons.wikimedia.org/wiki/File:Revised_BART_map.svg

Steve Souders

<http://stevesouders.com/>

<http://blip.tv/jsconfeu/steve-souders-fast-by-default-3080199>

Load Times

<http://blog.yottaa.com/2010/11/secret-sauce-for-successful-web-site-web-performance-optimization-wpo>

<http://glinden.blogspot.com/2006/11/marissa-mayer-at-web-20.html>

<http://techcrunch.com/2009/03/06/marissa-mayer-on-charlie-rose-the-future-of-google/>

Grand Theft Auto 4

<http://www.wired.com/gamelifeline/2008/05/grand-theft-aut/>

<http://www.vgchartz.com/game/7103/grand-theft-auto-iv/> (9.41m 360, 2012-02-01)

<http://www.vgchartz.com/game/7102/grand-theft-auto-iv/> (8.92m PlayStation®3, 2012-02-01)

Mile Marker

http://commons.wikimedia.org/wiki/File:Mile_marker.svg

Figure 24

