

A PROJECT GUIDE TO

UX DESIGN

**FOR USER EXPERIENCE DESIGNERS
IN THE FIELD OR IN THE MAKING**

RUSS UNGER AND CAROLYN CHANDLER



PEACHPIT PRESS

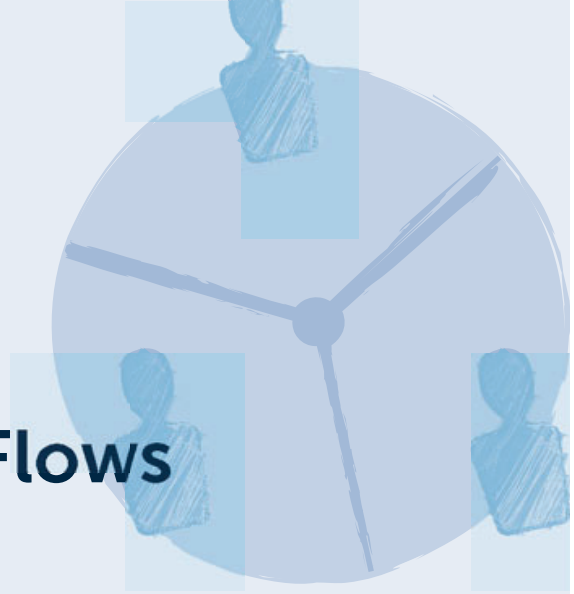
10

Site Maps and Task Flows

Structuring Your Project from Here to There and Back Again

Site maps help to identify the structure of Web sites and applications. They can show hierarchies and connections that allow your audience to gain an understanding of where users may locate content. Task flows take site maps a step further by identifying the various courses of action that a user may traverse within a section of the site. Task flows also draw the connections to error states, content, or page views based on decision points throughout the process. When used together, site maps and task flows can provide your audience with a clear picture of content structures and how users may navigate through them.

Russ Unger



What Is a Site Map?

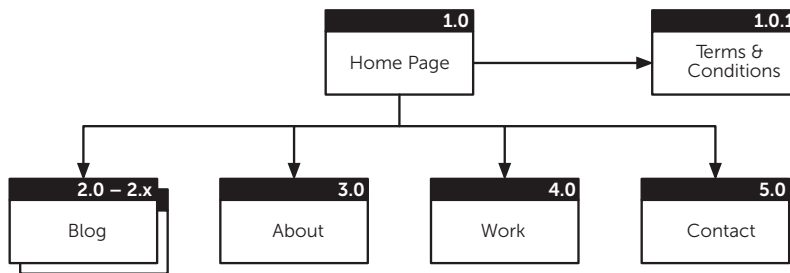


Figure 10.1 A site map for a basic Web site with blog functionality

Starting with the most basic of definitions, a *site map* is simply a visual way to display representative pages of a Web site (Figure 10.1). A simple site map generally fits on a single sheet of paper and resembles an employer's organizational chart. Site maps are not just for Web sites, however; you can use them for any type of application that would benefit from identifying pages, views, states, and instances of whatever is being displayed.

In most cases, you will use a site map to show teammates and clients how content will be organized for a Web site. It will provide an overview of the Web site navigation and, in some cases, will display all the connections each page can have.

What Is a Task Flow?

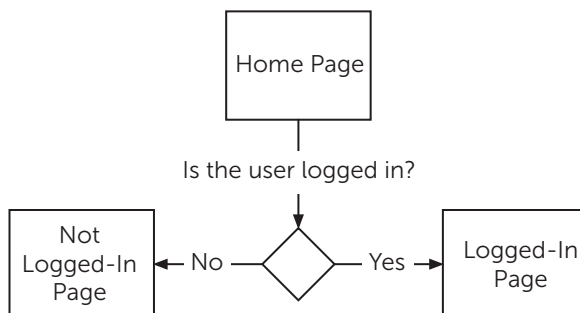


Figure 10.2 A basic task flow showing the path for a user depending on login status

Task flows identify paths or processes that users (and sometimes a system) will take as they progress through your Web site or application (Figure 10.2). Although site maps and task flows may look similar at first, the two types of diagrams serve different purposes: A site map tells you the visual hierarchy of a site's or application's layout, while a task flow gives you details of users' options and the paths they will be able to take.

Tools of the Trade

If you are just getting your start in the user experience design field and need a tool to start creating work product, you have many options:

- ▶ Microsoft Visio (<http://office.microsoft.com/visio>)
- ▶ Axure RP Pro (www.axure.com)
- ▶ OmniGraffle (www.omnigroup.com/applications/OmniGraffle)
- ▶ Adobe InDesign (www.adobe.com/products/indesign)
- ▶ Adobe Illustrator (www.adobe.com/products/illustrator)
- ▶ Microsoft PowerPoint (<http://office.microsoft.com/powerpoint>)
- ▶ OpenOffice Draw (www.openoffice.org)
- ▶ HTML
- ▶ Blueprint CSS (www.blueprintcss.org)

So how do you choose? You can ask other designers; everyone has a favorite and they're usually happy to name it. Just don't be surprised if they, like your authors, answer "good ol' pencil and paper." You can also test out free trials online or opt for a no-cost solution, such as OpenOffice Draw, which is part of the OpenOffice.org suite of tools and outputs the same formats as popular office suites.

Beyond pencil and paper, what do we use? Many of the examples in this book were generated with Microsoft Visio 2003, using stencils created by Nick Finck, director of user experience at Blue Flavor (www.blueflavor.com) and publisher of Digital Web Magazine (www.digital-web.com). You can download Nick's outstanding stencils from www.nickfinck.com/stencils.html. Such ready-made stencils, shapes, and samples are invaluable to new and experienced practitioners alike. In addition to Nick's, check out the offerings of the Information Architecture Institute, which houses many of these tools on its Learning IA page: <http://iainstitute.org/en/learn/tools.php>.

Note Microsoft currently offers a 2007 version of Visio; however, many companies still have not upgraded to the product and for the sake of not dealing with version differences, we currently recommend Microsoft Visio 2003.

Whatever tools you decide to use, there are countless examples online from other professionals who are happy to share them and help you along in your career. These are largely free and can provide you with the framework you need to create—at the barest minimum—very professional-looking documentation. Unfortunately, many people are not taking advantage of these resources. Don't be like those people!

Basic Elements of Site Maps and Task Flows

The most basic of elements within your drawing program will be more than enough to get you started creating site maps and task flows. To ensure that your creations can be easily interpreted by a wide audience, however, it's best to use a standard set of shapes.

The Visual Vocabulary for Information Architecture is one such standard, and the one used in this book. Created by Jesse James Garrett, one of the founders of Adaptive Path (www.adaptivepath.com), it is available online at www.jjg.net/ia/visvocab. The site provides many elements to help you articulate your site maps and task flows, all of which are available with detailed descriptions and as downloadable stencils for many of the popular drawing and sketching programs (more on these in a bit).

To help you get started and become familiar with the basics, the next sections take a look at the Visual Vocabulary's core set of elements and what they represent.

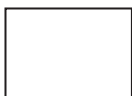


Figure 10.3 Page element from Jesse James Garrett's Visual Vocabulary



Figure 10.4 Pagestack element from Jesse James Garrett's Visual Vocabulary

Page

According to Jesse James Garrett, a *page* is "the basic unit of user experience on the Web." "Instances" or "views" of content may be more realistic today, but a page is still very meaningful. There are a number of ways to draw these pages, but the simplest, most commonly used format is a plain

rectangle (Figure 10.3). As you progress through creating site maps and task flows, you will want to find the style that best suits you for labeling and numbering your pages.

Pagestack

A *pagestack* represents multiple pages of similar content (Figure 10.4). An easy way to comprehend pagestacks is to think of dynamic content, such as a common blog page created using a publishing system. These pages are designed once and are in a design template, but you have the ability to click through many different pages of content—without actually leaving the original template design.

Decision Point

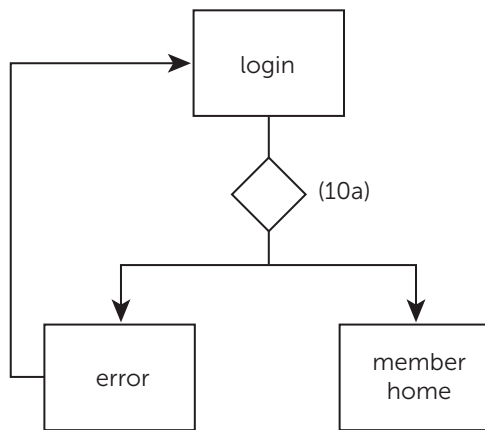


Figure 10.5 Decision point element from Jesse James Garrett's *Visual Vocabulary*

A *decision point* is used to show the path that a user can take depending on the answer to a question (Figure 10.5). The decision point 10a might be “Are the user’s login credentials correct?” The answer to that question would determine which page (or content view) would be displayed. A failed login results in an error message, while a successful one takes the user to the site members’ home page. Take the time to appropriately label decision points; you’ll be glad you did, particularly when sharing your work product with teammates or clients.

Connectors and Arrows

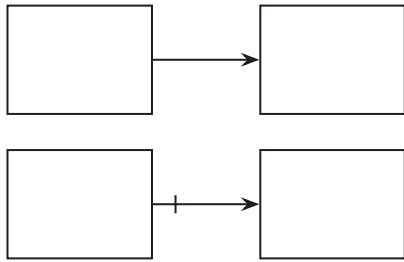


Figure 10.6 Connector and arrow elements from Jesse James Garrett's *Visual Vocabulary*

Connectors and arrows are used to show movement or progress between pages, pagestacks, decision points, and so on. Connectors generally appear where there is a call to action from one page to another. For example, a link to the About Us page from the Home page could be the connector between the two pages. Arrows (top of Figure 10.6) indicate “downstream” movement toward task completion.

Connectors with the crossbar (bottom of Figure 10.6) can be used to identify when movement back to the page you originated from (“upstream” movement) is no longer available. For example, once a user is logged into a Web site, what was the home page content may now be personalized for the user, and the generic page, or the login page, will no longer be available to the user from the path they just followed.

Conditions

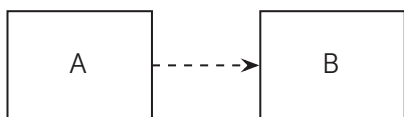


Figure 10.7 Condition element from Jesse James Garrett's *Visual Vocabulary*

A dashed line is a fairly common way to display a condition. It can appear in site maps, task flows, and other work product you may create or invent.

You can use a dashed line as a connector (as in Figure 10.7) or as a box around an area to highlight that a connection to a page—or an entire section of pages—is conditional based on some other action or event.

Common Mistakes

You wouldn't go into a presentation with a lump of peanut butter on your chin or a coffee-stained shirt. Not only would such a blunder undermine all your hard work, it could also prevent you from landing a good project. A sloppy site map or a task flow that looks unprofessional can do just as much damage.

To help you recognize those little lapses with big consequences, the next sections take a close look at some bad designs. Learn to spot these common mistakes—then avoid them.

Sloppy Connections

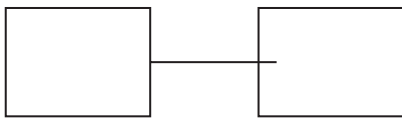


Figure 10.8 *A missed connection between two pages*

Sloppy connections are just that: Sloppy. They're badly drawn. They look very amateurish, and they give you—the author—the appearance of not paying a lot of attention to detail in your work, to say the least. Most tools have some method of assisting you with connecting your lines to your boxes. Please take advantage of it.

Do *not* get lazy, regardless of the time constraints and pressure that you might be under. In most applications, using a combination of Shift and other keys allows you to drag elements from a starting point in 45 degree angle increments. Take advantage of this built-in functionality and ensure that your connections are, well, connected. If you are showing pencil sketches, you should have an eraser on hand just in case.

Make it a rule: Always make sure any lines that touch any other object are connected with accuracy.

Misaligned and Unevenly Spaced Objects

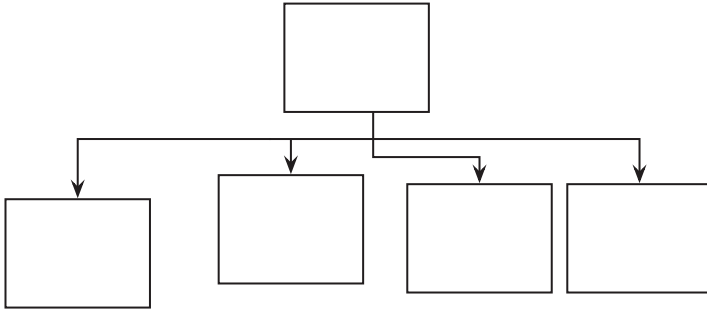


Figure 10.9 Pages that are not aligned and are unevenly spaced

Depending on the tool you are using, it can be difficult to ensure that your objects are accurately aligned or evenly spaced apart on your site map or task flow. There are some fairly simple ways to ensure that you get this basic rule down.

For starters, turn on the grid in whatever application you're using. That way, regardless of whether the tool offers options that ensure evenly spaced, appropriately aligned objects, you can always *count* the number of grids between your objects. Fortunately, when you are using pencil and paper, you can use graph paper and apply the same basic principle.

It is that easy to make your documents look professional. Unfortunately, it is also that easy to make your documents look like you really don't care about the quality of your work.

Poorly Placed Text

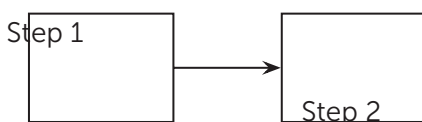


Figure 10.10 Inconsistently placed text

Careless text placement (as in Figure 10.10) seems simple to avoid, yet it is another common mistake. Find a way to make your text fit nicely in the shape you have created, and make sure any labels that are placed outside of their elements have appropriate connections (Figure 10.11).

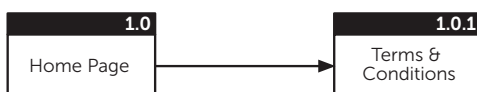


Figure 10.11 Well-placed text

It may seem basic, but proper placement of your text—along with appropriate font size and usage—will make your documents easier to read and use.

Lack of Page Numbering

It's time to establish another rule: Number every page of every site map that you create. Don't create a vague, numberless map like Figure 10.12.

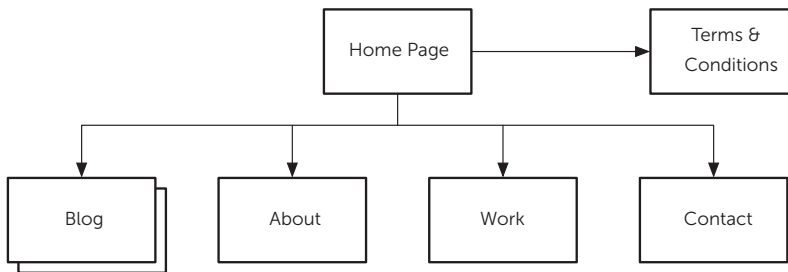


Figure 10.12 Site map without a numbering structure

Any page that you identify on your site map needs to be given a number, and your numbering system must allow for downstream changes to occur as new iterations and versions of your project are created.

You can use a variety of approaches for numbering pages; the most common is to identify your home page as either 1.0 or 0.0.0.0 (Figure 10.13). Over time, you will be able to determine which of these works best for you, but until you get comfortable and understand the advantages and disadvantages of both approaches, start by identifying your home page as 1.0. This method allows you to account for any decisions and pages that may occur prior to your home page—such as a Flash preloader, a login or register screen, or a number of other page types—as 0.X.

A numbering system on your site maps allows other documentation to sync up with it. The numbering system can proliferate to other documents, such as

- ▶ **Content matrix.** Content creators can map their copy and other content to specific pages (and to a specific element in a wireframe; more on that later).
- ▶ **Task flows.** Task flows can use the same numbering system to show how a user will proceed through the pages of a specific task.
- ▶ **Wireframes** (see Chapter 11). The pages of your wireframes should share the same numbering system as the pages on your site map to provide a clear connection between the two documents.

- ▶ **Visual design.** Visual designers can sync design pages and elements to specific pages on your site map. This allows them to segment their inventory when it is time to hand off their designs to developers.
- ▶ **Quality assurance documents.** Quality assurance teams may author testing scripts that are dedicated to a specific page or pages on the site map.

Your attention to detail and structure at this point helps keep everyone else who touches the project on track and provides them with structure for their tasks.

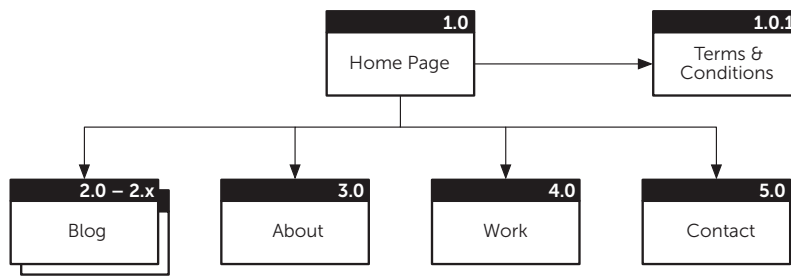


Figure 10.13 Site map that correctly connects pages, with elements that are aligned, evenly spaced, and appropriately numbered

In short, numbering the pages on your site map will make everyone else’s life easier—and that means your life will be easier, too.

The Simple Site Map

In addition to containing page numbers, Figure 10.13 is a good model for creating the map of a basic Web site that has limited dynamic functionality and a mostly static nature. The pages identified for this Web site were

- ▶ Home
- ▶ Blog
- ▶ About
- ▶ Samples of work
- ▶ Contact

As you can see, this simple site map incorporates the core elements from the visual vocabulary and maintains a professional style and appearance. Most importantly, it provides a very clear picture of the navigation, pages, and conditions available to users of the Web site.

Advanced Site Maps

A simple site map can generally fit on a single sheet of paper and most likely it looks something like an employer's organization chart. More advanced site maps, however, can expand to multiple pages.

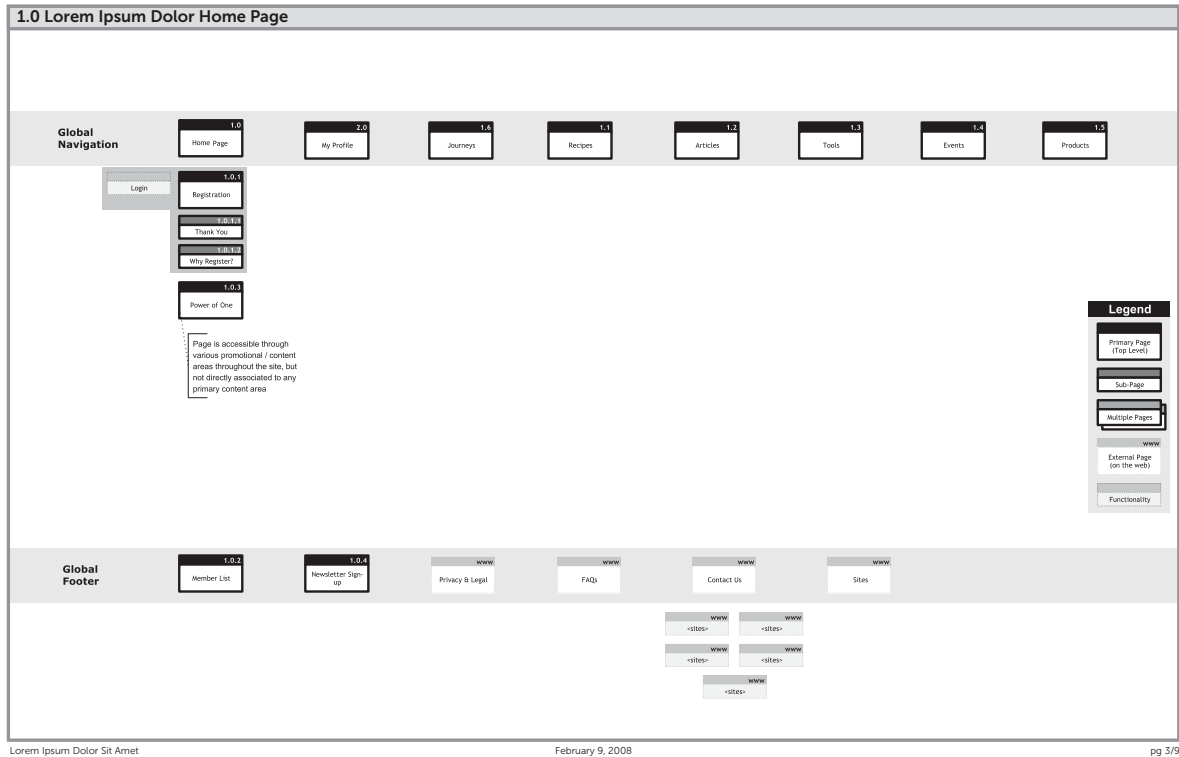
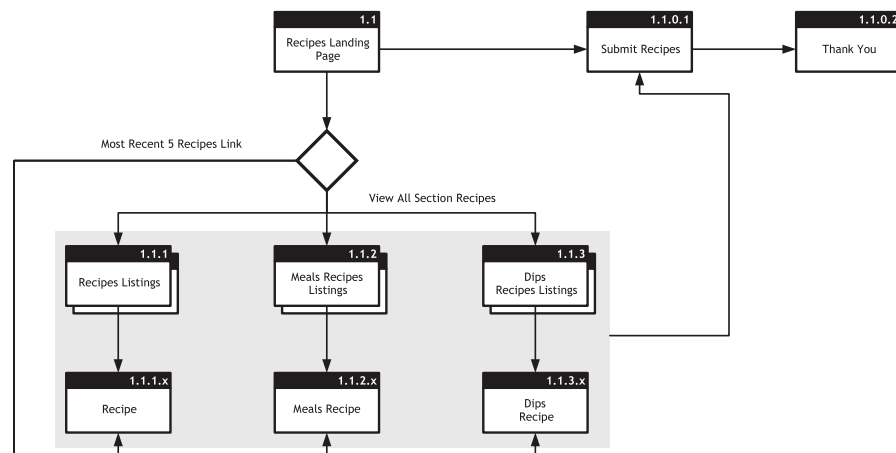


Figure 10.14 Advanced site map home page view

When creating site maps that are more advanced or for larger scale Web sites and applications, one approach is to use your first page to identify any of the steps required to reach the site's home page. (That's correct, we're suggesting you use a task flow as part of your advanced site map.) In addition, you should identify all your top-level pages, global navigation elements, and footer elements. This allows you to show a very high-level overview of the site up front and provides your team and clients with a clear picture of the project.

The first page is also an appropriate place to include a legend or key to help in reading your site map (see Figure 10.14). Your team and your clients will need one. Don't skip this step!



Lorem Ipsum Dolor Sit Amet

February 19, 2008

pg 4/9

Figure 10.15 Advanced site map section view

Pages that you create after your first page should essentially map back to it. For every top-level page, you should have at least one page following that identifies all the pages, pagestacks, and external content that will be required for the Web site or application (Figure 10.15). If necessary, do not be afraid to connect subpages together. Site maps can grow to become more expansive than any single sheet of paper of standard size will allow. This is nothing to worry about, as long as your site map is well organized and the connections are clearly documented for your audience.

These examples are more than enough to get you started in the world of creating site maps. As you begin to make your way through a variety of projects and you find that your skills—and often your team or client needs—are growing, you will find that there are vastly different approaches and methods you can take toward delivering site maps.

Breaking the Site Map Mold

You have now seen solid examples of site maps that should fit most of your needs in getting your primary tasks accomplished. Don't let those models prevent you from exploring ways that work better for you—and please share them with us! Different approaches can highlight information beyond basic site architecture. For example, consider the site map in Figure 10.16, which was kindly provided by Andrew Hinton, senior information architect at Vanguard.

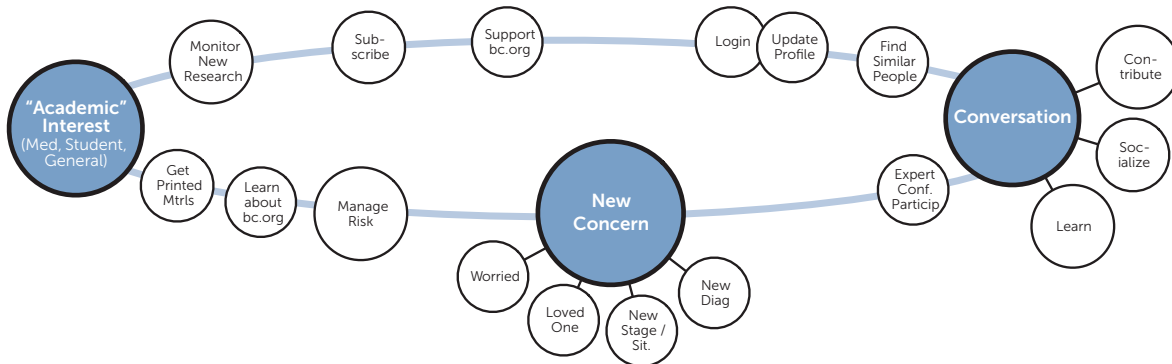


Figure 10.16 Advanced site map. Courtesy of Andrew Hinton.

This site map not only shows the various pages of the Web site, it also serves to provide insight into user paths and priorities. Andrew (www.inkblurt.com) says he created the site map after seeing an example from Wolf Noeding that sparked his creative flame. Andrew uses this site map to show various user scenarios and mental models related to the Web site. The larger circles on the map perform an additional function: They highlight top-level areas of the site that receive the most traffic.

Like all good user experience practitioners, Andrew borrowed—but also gave credit. There are limitless ways that you can expand your site maps as you begin to get more comfortable using the tools and identifying your work product—and client—needs. Let inspiration strike you where you find it! Don't be afraid to try something new, but take your time to make sure the time you spend is useful and valuable.

Task Flows

Using many of the same basic elements as site maps, task flows are diagrams that identify a path or a process that users (and sometimes a system) will take as they progress through your Web site or application.

You can use task flows in a number of different ways. When used in conjunction with a site map they can show how a user arrives at a page with a specific set of information displayed. Sometimes they are used to show how a specific user type (a persona) would expect to traverse a Web site and what that persona would expect to see based on their personal mental model. You also can use task flows to identify complex processes that need to be clearly understood before the project is sent to the development team.

You might not use task flows on every project that you work on, and they may not always end up as work product that you present to your clients, but it is always a good idea to use them—even if just in a pencil-and-paper format for your own benefit.

A little clarity can go a long way.

In order to create a task flow, you need to have an understanding of the user's objective. In some cases you will receive a requirements document, and in other cases you may receive (or author) a use case. Although a use case consists of just a few sentences summarizing tasks and goals, it will allow you to synthesize the user's view into the experience.

The use case for the scenario in Figure 10.17 might look like this:

- ▶ System displays project list.
- ▶ User selects a project.
- ▶ System displays basic project information, in Write mode.
- ▶ User changes status of project to Closed.
- ▶ System checks for pending tasks.
 - ▶ If there are pending tasks, system displays error notice.
 - ▶ If there are no pending tasks...

- ▶ System checks for pending payments.
 - ▶ If there are pending payments, system displays error notice.
 - ▶ If there are no pending payments...
- ▶ System displays summary page.

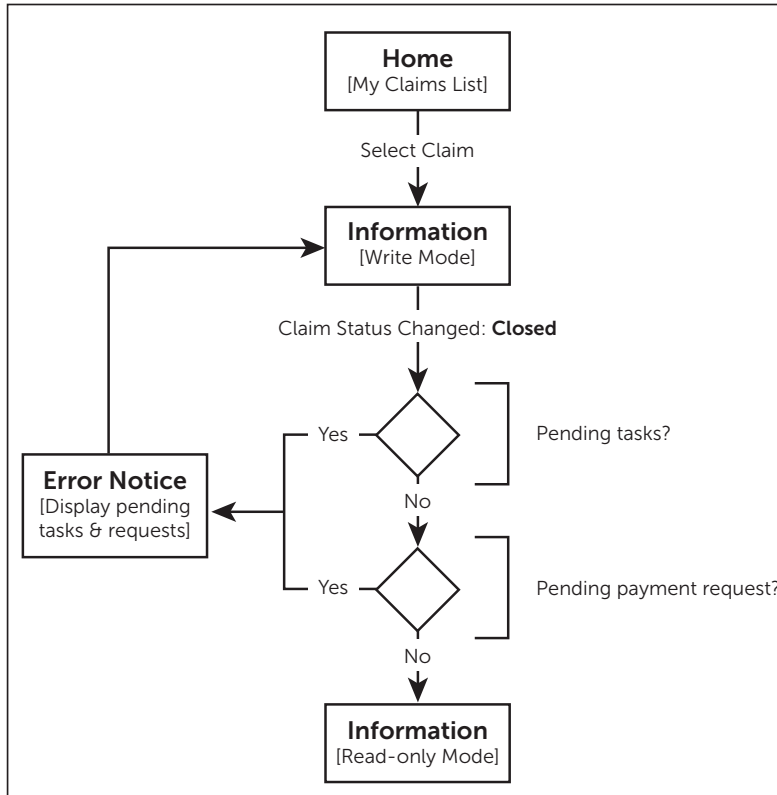


Figure 10.17 This task flow identifies how a system displays information to a user based on the responses to multiple conditions.

The task flow in the figure depicts the sequence of information displayed to a user based on whether a variety of conditions from the use case are met. If either question in the center (“Pending tasks?” or “Pending payment request?”) is answered with a yes, the system displays an error notice, potentially delivering additional information to help the user complete the required tasks prior to making forward progress. If both conditions are answered no, the system provides the user with a display that identifies success.

The task flow in Figure 10.18 shows the paths that a user could take from a calendar application through a travel shopping site. The task flow is very high level in that it identifies three very different paths that require testing by users to ensure that the detailed flow for each path meets user needs.

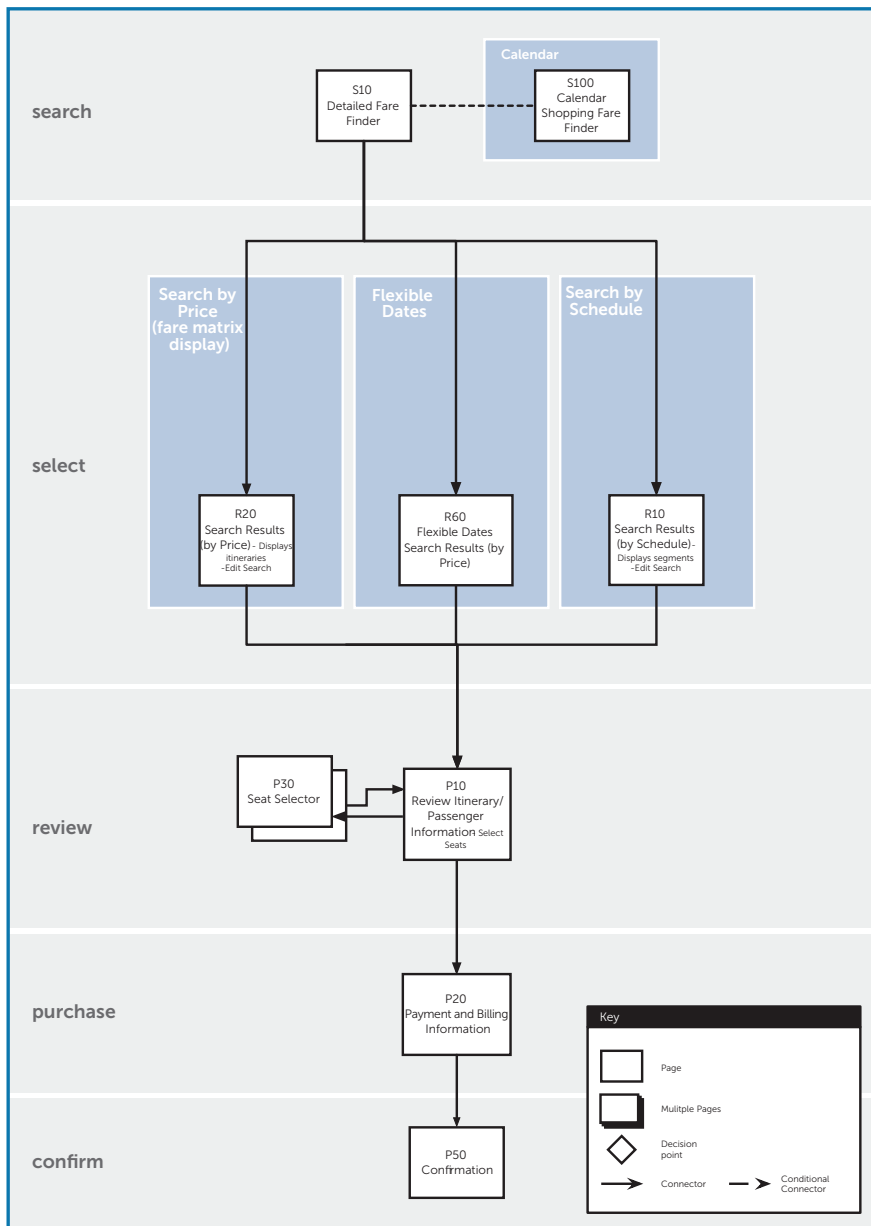


Figure 10.18 Task flow used to demonstrate the path of a user through the phases of a purchase process

Users of this application are able to enter a set of dates for their travel and then make purchases based on their own priorities. After users set their dates to search for travel, they can prioritize their results according to what is most important to them: price, flexibility of travel dates, or travel times (schedule).

The task flow identifies the high-level paths that a user could take in order to provide direction to the persons facilitating the testing. Detailed task flows could be created for each of the paths in the groupings and then provided to the development team to create the pages necessary for testing.

Taking Task Flows to the Next Level

As with all the topics in this book, don't feel as if what you are seeing here is the beginning and end of the universe of task flows. Explore new uses and expand your use of the basics outlined here as much as possible—as long as there is a good purpose for it.

As your skills with creating task flows continue to grow, you may find yourself creating a work product that is a bit more colorful, has more options, includes modified or improved language rules, and so on.

Process Flow

Figure 10.19 shows a task flow Will Evans (www.semanticfoundry.com) took to the next level and turned into a process flow diagram.

His process flow is very high level and flexible and is used here to show that in the many steps of a project process, the first phase of the project only appears to be a single step, however, in this particular case, it is important to understand that phase does not consist of a single event. Instead, the first phase of the project, in this case, is actually composed of many different activities:

- ▶ User research
- ▶ Market research
- ▶ Ethnography and contextual inquiry
- ▶ Usability testing
- ▶ Competitive analysis
- ▶ Market analysis
- ▶ Culture analysis
- ▶ Log file analysis

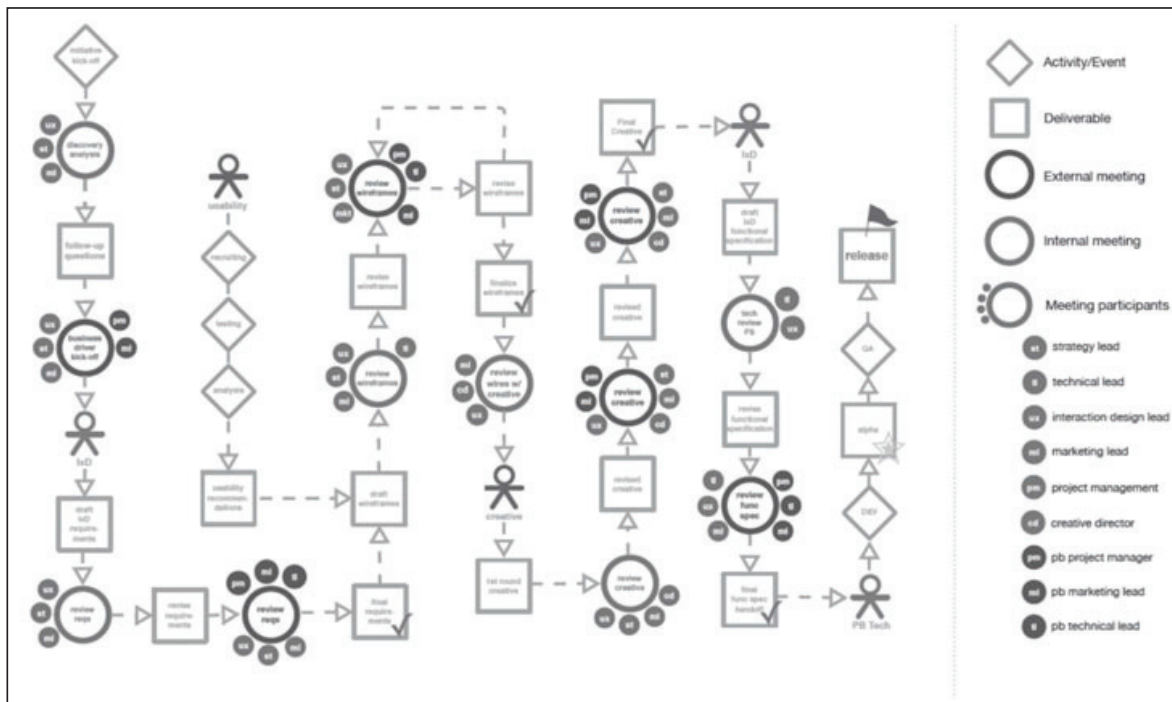


Figure 10.19 This process flow diagram takes task flows to the next level to articulate complex scenarios. Courtesy of Will Evans.

For each of those activities, reports are generated that feed into other various documents prior to the project kickoff, where the necessary stakeholders would gather to determine the scope, priority, and dates. All of this is shown in the process flow diagram.

As you can see from this example, the sky is the limit when it comes to creating task flows. Look at the example above and consider ways to take your deliverables to the next level. It may take a bit of practice, but with a little finesse you can create task flows that will blow away your clients!

Swimlanes

James Melzer (www.jamesmelzer.com), principal information architect at SRA International Inc. (www.sra.com), has created a number of diagrams that extend far beyond the basic task flows. The diagram in Figure 10.20 shows a task flow that was extended to show “swimlanes” of actions, notifications, and so on in a process that had a lot of events happening at the same time—with this project a traditional approach to task flows could have been a nightmare!

Instead, James explored extending the basic task flow to encompass all the various steps and actions taking place in a format that was much easier to understand.

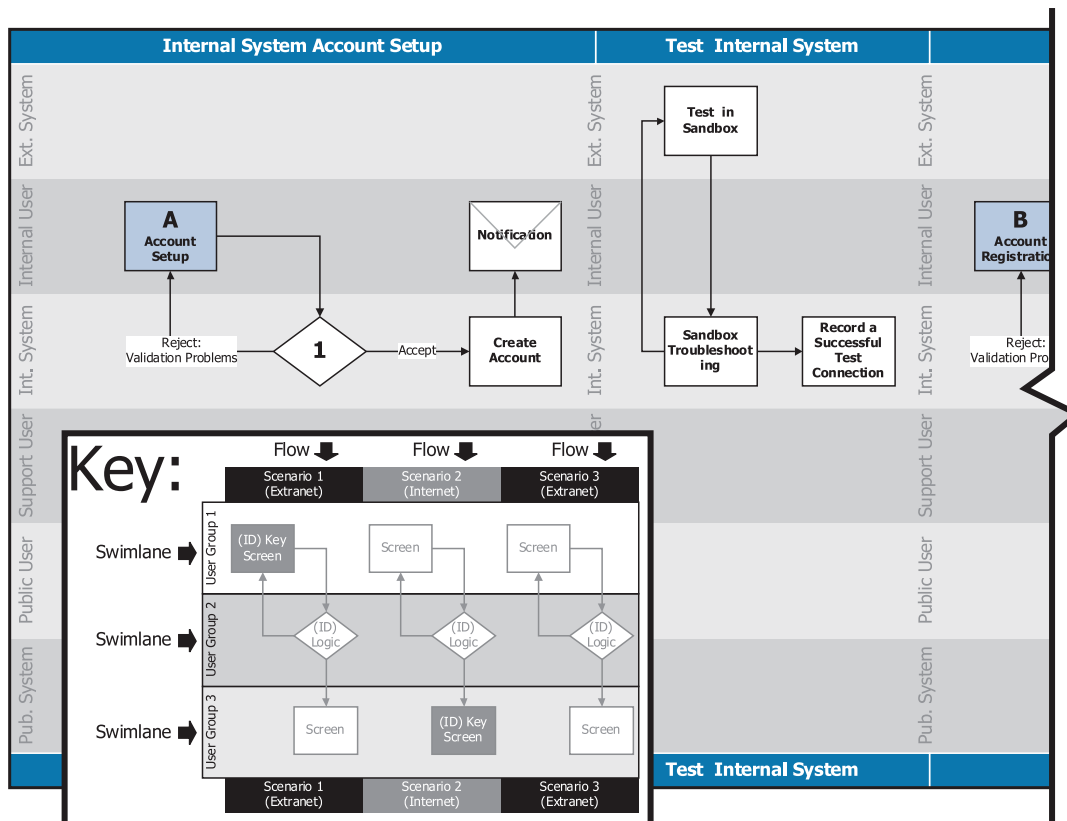


Figure 10.20 This swimlanes diagram is an example of expanding task flows to illustrate complex scenarios with multiple actions in many places. Courtesy of James Melzer.

James described the project and swimlanes as follows:

The system lets people manage information about buildings they own. This extension would allow building services partners to provide data system-to-system on behalf of their customers, reducing the data entry needed from the owners. The project had three parts: partners configuring the presentation and operation of their data services, customers signing up and using the partner data services, and ongoing partner data management and troubleshooting on the back end.

We were planning out a major extension to an existing system. We knew early on that nearly all the service scenarios involved multiple users and multiple systems. There were a number of notifications, and a lot of the processes were asynchronous. This diagram helped us identify, design, and explain the service scenarios needed for the project. In the full version of this work product, we actually had detailed wireframes arranged underneath the flows in this diagram. The whole thing covered a wall. Once we were fairly confident in the design concept, we chopped it up into a more traditional multipage specification.

The important thing to remember here is to not limit yourself in your uses of task flows or site maps. Stretch the boundaries of the basics that you've been shown in this chapter. In the event you really need something to test your mettle, spend some time creating a task flow for how to tie your shoes.

Good luck!

11 Wireframes and Annotations

Design and Direction—Before the Visual Design Begins

Wireframes and annotations are ways to identify the proposed content and structure, as well as functional behaviors, of a view of a Web page or an application. When combined with site maps and task flows, these documents are also extremely useful for identifying prototyping scenarios and proof of concepts. Wireframes are typically presented in grayscale, bereft of graphical elements or finalized content; instead they use placeholder content to highlight representative locations that can be used as guidance in the visual design.

Russ Unger

