‘I can’t find what I am looking for’ is one of the most common complains staff make about intranet content. Contributing to this issue is poor search, and poorly named or simply missing material. However, most often, the issue comes down to poor site structure and a lack of good information signposting.

Developing navigation to ensure the pathways and language are optimised to steer staff to the information they seek is fundamental to a successful solution. A number of familiar and well established user-centred design (UCD) techniques can be called upon to develop this navigation.

Essential UCD techniques include:
- content inventory and evaluation
- card sorting
- information architecture development
- wireframe development
- usability testing

However if the architecture itself is not tested, significant time invested in wireframe development and site build may be wasted. Having a simple and effective method for assessing the proposed site architecture, before wireframing, allows the designer to refine and retest the site map before developing site visuals which tend to be more difficult and expensive to rework.

For those less familiar with these UCD techniques in an intranet environment, they are covered in detail in our book *Designing intranets: creating sites that work*:


**Introducing tree testing**

Formerly known as ‘card-based classification evaluation’, tree testing is a quick, simple, and cost-effective technique to test how well your site structure guides site-users to the content they are looking for.

The core of tree testing is a set of mock menus that provide a simple representation of how the navigation will work on the new site. This provides a hands-on way for staff to see if they can find what they need.

It works as follows:
- Create a set of menus (on paper, or online) based on your draft structure.
- Develop a set of scenarios or tasks that represent how staff will use the site.
- Recruit a range of representative staff.
- Evaluate staff members’ actions as they look for the information listed in each scenario.
- Use the successes and failures of the site structure (the staff themselves never fail) to refine what you have, and retest.
- Repeat as necessary until the site structure is optimised.

The mock menus can be presented to staff in paper form or via an online ‘tree testing’ facility.

**Tree testing on paper**

The starting point for tree testing is the draft information architecture or site structure to be tested which will have been developed through earlier UCD steps.

Starting with the top level menu items, these are numbered 1, 2, 3, etc. Underneath that, the menu items are numbered 1.1, 1.2, 1.3 … 2.1, 2.2, etc. Then 1.1.1, 1.1.2, etc.

*Stephen Byrne* is an organisational psychologist and senior consultant at Step Two Designs, an intranet and content management consultancy based in Sydney, Australia. Stephen specialises in intranet strategy and design, knowledge practices and organisational change.
Before long, the entire draft structure of the intranet is mapped out as a list, down two or three levels deep.

Once the navigation structure is converted into a numbered list, it can then be written out onto filing cards, as shown below. Think of this as a paper mock-up of an on-screen menu structure.

The first card shows what appears on the homepage as main menu items. Below that is the next level of navigation, written out onto multiple filing cards, and onwards down the menu structure.

Where needed, long menus can be split across several cards. The net result is a small stack of menu cards, ready for use in the testing.

Create scenarios

Many of the steps prior to tree testing will have helped determine the key purposes of the intranet site and the principal tasks that staff are trying to complete.

These form the basis of the scenarios that will test the effectiveness of the site structure. The scenarios should:

- represent key tasks that people need to achieve using the intranet
- broadly cover the extent of the architecture being assessed

Write each scenario on a filing card in clear and readable handwriting. Alternatively, print mailing labels that can be stuck onto the filing cards. This is particularly useful if you need to create multiple sets of cards as it will save writing time.

Identify each card with a letter. This helps to identify individual scenarios when analysing the results. Keep the letters small so they do not distract participants from the actual scenario titles.

Run the testing sessions

Select a broad cross-section of staff from across the organisation, ensuring that these are end users rather than stakeholders.

With each participant, give them a brief overview of the project, and an explanation of the activity that they will be doing. This doesn’t need to be extensive, just enough to get them working productively through the scenarios and menu cards.

Step through the tree-testing process with each participant:

- Ask the participant to read the scenario.
- Show them the top menu card, and ask them to choose where they would look for the information.
- Show the next level down based on their selection (if they chose menu 2, show the card that has 2.1, 2.2, etc.).
- Continue until the lowest level of the menu structure is shown.
- If the participant looks at the card and doesn’t think that they are in the right place, ask them if they would like to make another choice.

Well chosen scenarios reflect how staff use the site

The menu structure for the new intranet written out onto filing cards, three levels down.
• Go back a level, or to the beginning, and follow their second choice. Don’t ask them to look in more than two places as the intent is to understand where they would look first.

• When a final menu item has been chosen, note this down against the scenario (for scenario C, they chose 3.2.2). If they have two attempts, note down both destinations.

The session should only take 10–15 minutes, and can quickly cover a large number of scenarios in that time.

Analyse the results
The key to tree testing is how the results are written up. Create a spreadsheet as shown below, listing the menu numbers down the left side (1, 1.1, 1.2, 1.2.1, 1.2.2, etc.). Across the top go the tasks (A, B, C, etc.).

Where a participant chooses a menu, mark an ‘X’. A big x (‘X’) for their first choice, and a small x (‘x’) if they had to make a second choice.

The patterns should quickly become apparent. Task A is a problem, as there is considerable scatter. Staff weren’t sure where to look ending up in several locations.

Task B has strong grouping. In this case, staff looked in the correct location, so this is a successful result. In the case of Task C, most people looked in one location, but this was the wrong spot: a poor result.

There will rarely be a task where everyone looks in a single location, but this doesn’t cause a problem in practice. As long as the majority of staff can find the majority of items, the navigation scheme is successful.

Use the results to tune the IA, and retest

Refine and repeat

Based on the outcomes of the testing sessions, the intranet team can identify the areas of the navigation that need improving. This may involve minor tweaks in wording, or major restructures.

Either way, making changes to the draft structure is easy at this stage. It shouldn’t take more than an hour to rewrite the menu cards, and then they are ready for use again in further tree testing sessions.

The key to this phase of the design process is that it is ‘iterative’. A little bit of testing is done, and refinements are made based on the results. Some further testing, and refinement.

With a round of testing and refinement taking no more than a day, quite a few rounds of testing can be done within a week (or part thereof). Experience has shown that major fixes are made early on, so it should not be necessary to run more than two or three rounds of testing.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>XXX</td>
<td></td>
</tr>
<tr>
<td>1.2</td>
<td>X</td>
<td>XXXXX</td>
</tr>
<tr>
<td>1.2.1</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>1.2.2</td>
<td></td>
<td>XXXXX</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.1</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>2.1.1</td>
<td>XX</td>
<td></td>
</tr>
<tr>
<td>2.1.2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sample results of a tree testing session
Tree testing online

Simple electronic prototypes can also be created to test the structure, using a variety of tools. Some teams have even mocked up the structure on a file server, using Windows Explorer as a super-simple way of allowing staff to navigate the draft structure.

Perhaps one of the more elegant ways to conduct tree testing is using the dedicated online service called TreeJack: www.optimalworkshop.com/treejack.htm

TreeJack also begins with your draft information architecture, which can be imported directly. Scenarios are also developed and put into TreeJack. The testing environment now looks like the figure below. The scenario is presented, and the test participant clicks their preferred choice. The next level down is presented automatically, and again they make their choice.

Online testing sessions are much simpler for both participant and facilitator, and generate a range of comprehensive statistics. It also allows testing with a larger group of participants, unsupervised.

Regardless of whether the testing is done by hand or electronically, the same insights will be gained. Use an approach that you are most comfortable with, and don’t be afraid to experiment over time.

Verbal feedback from users is as valuable as the hard data

Consider soft data too

It is important not to limit the face-to-face facilitated sessions as you risk missing valuable user feedback particularly in the earlier sessions.

If the candidate experiences difficulty it can be useful to understand what caused the difficulty. For example, any of the following could be at fault:

- the scenario was not well worded
- words in the site map were not well understood
- different pathways were experienced as leading similar information
- the expected pathway did not exist

The nature of the difficulty provides critical input into why the proposed navigation is not working, and points to where effort is needed to find optimal solutions.

Complement face-to-face testing with remote testing

Online testing sessions are much simpler for both participant and facilitator, and generate a range of comprehensive statistics. It also allows testing with a larger group of participants, unsupervised.

Verbal feedback from users is as valuable as the hard data

Regardless of whether the testing is done by hand or electronically, the same insights will be gained. Use an approach that you are most comfortable with, and don’t be afraid to experiment over time.

Consider soft data too

It is important not to limit the face-to-face facilitated sessions as you risk missing valuable user feedback particularly in the earlier sessions.

If the candidate experiences difficulty it can be useful to understand what caused the difficulty. For example, any of the following could be at fault:

- the scenario was not well worded
- words in the site map were not well understood
- different pathways were experienced as leading similar information
- the expected pathway did not exist

The nature of the difficulty provides critical input into why the proposed navigation is not working, and points to where effort is needed to find optimal solutions.

Complement face-to-face testing with remote testing

Online testing sessions are much simpler for both participant and facilitator, and generate a range of comprehensive statistics. It also allows testing with a larger group of participants, unsupervised.

Verbal feedback from users is as valuable as the hard data

Regardless of whether the testing is done by hand or electronically, the same insights will be gained. Use an approach that you are most comfortable with, and don’t be afraid to experiment over time.

Consider soft data too

It is important not to limit the face-to-face facilitated sessions as you risk missing valuable user feedback particularly in the earlier sessions.

If the candidate experiences difficulty it can be useful to understand what caused the difficulty. For example, any of the following could be at fault:

- the scenario was not well worded
- words in the site map were not well understood
- different pathways were experienced as leading similar information
- the expected pathway did not exist

The nature of the difficulty provides critical input into why the proposed navigation is not working, and points to where effort is needed to find optimal solutions.
Benefits of tree testing

Using tree testing as part of the intranet design process provides a number of benefits:

- It provides input into the draft intranet early in the design process, when changes are quick and easy to make.
- It takes little time for the participants, who only need to give you 10–15 minutes to provide valuable input. This means that participants are simple to recruit and are happy to be involved.
- It focuses on real tasks that people would undertake. In contrast, activities such as card sorting focus only on the content, and the outcomes may not allow users to complete tasks easily.
- It is simple and easily understood. It can be explained to participants in a matter of minutes.
- It is quick and cheap to run. The evaluation takes few resources and a short amount of time so can be done as needed.
- It is low-tech and flexible. The classification can be modified during the evaluation if necessary to test alternative approaches.
- It needs no special training for organisers.
- It involves users in the design process, and demonstrates that the intranet will be created with their needs in mind.
- It allows a large number of staff to be involved in the design process.

Quantify improvements

In addition to refining the structure of the site, tree testing can also be used to quantify the improvements. Testing can be done on the existing site as a baseline which can be compared against the outcomes of each stage of testing.

There are a number of different ways of putting figures to the improvements. The simplest approach is to calculate the percentage of scenarios that are successfully completed. This might look like:

Staff could successfully complete only 52 per cent of tasks on the old intranet. After one round of improvements this immediately rose to 75 per cent, and the final structure has a 87 per cent success rate.

If the same tasks are used for the usability testing, this can allow the results to be correlated to some degree. Tree testing can also be done on the final structure, just before or after go-live.

Recognise the limits of tree testing

Tree testing is not a silver bullet that produces a perfect site structure for the new intranet. While the testing is reasonably realistic (staff start at the homepage and work their way down the structure), it is still testing the navigation in isolation.

On a real intranet, navigation is intertwined with page layout, with links grouped together and presented in ways that dramatically change how staff make use of the site.

Tree testing overlooks the role of cross-linking, or more complex navigation schemes such as filtered searching and faceted navigation.

Tree testing should therefore only be seen as producing an ‘80 per cent solution’ for the navigation, with further refinement using techniques such as usability testing when the page layouts have been developed.

Nonetheless, by identifying problems early, it allows them to resolved when it is cheap and easy to do so.