Grouping

1

Dragging two pages together forms a group.

When one is positioned over the other, an indication is given that they will form a group.

Many visualizations are possible for this indication. A tractor beam might be an attractive option.

2

Each tab shrinks in size and a bounding box of fades in.

You can resize a group, which scales everything inside of it. Feedback is given in real time.

Resizing isn't aspect-constrained, so that you can have thin groups or square groups.

3

Adding more tabs causes each tab to shrink in size and for the tabs to arrange them selves.

An optimum space-efficient layout should be used.

Pages that get dragged over a group also get a visual indication that it will be added to the group.

4

You can drag a page out of a group, which causes both the page to return to its normal size, as well as reduce the group dimensions (and increase the internal page size).

The page animates to the original size when it gets dragged out of the group.

The after state.

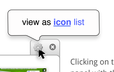
5

You can have a group of size one. When you remove the

When you drag the lone straggler out of the group, the group fades away and is gone. Also, if the page was scaled down it animates back to full size.

6

When the mouse is over a group for more than 250ms, a title bar appears with a couple options: title, configuration, and close.



Clicking on the configuration icon shows a panel with the choices for how that group should act. At the beginning this might not exist if there are no choices.

Choices may be:
- thumbnails
- text (list view)
- tree (for searches)
- manual (arrange as you wish)

This list should be extensible by extensions.

By clicking in the name area, or by just started to typing a tab after forming a group, you can then begin naming a group. The name remains visible even when you are not hovering over the group:

Groups Within Groups

1

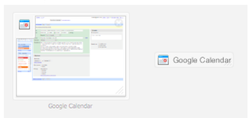
To take advantage of the semantic power of physical grouping, you should be able to put groups within groups. (Obligatory yo-dawg meme goes here).

Simply drag one group into another.

Just like with pages, groups also get an indication that dropping will create a sub group.

Dragging a sub-group out of a group, causes the group to animate back to the original size.

An Example of Icon and List Views



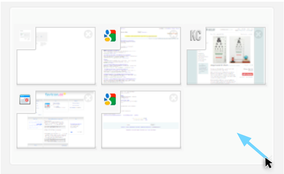
The image on the right shows what a page might look like in list view. Inside a group switched to list view, it would look the same but without the border around each page name.

Resizing Groups

In here we describe the optimal algorithm for resizing groups on-the-fly, and for allowing non-confusing shuffle algorithms.

My original work got killed by HG so will have to recreate it. Coming soon.

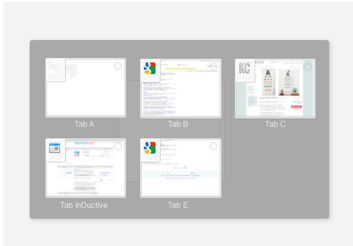
Scaling Groups Down



When you scale a group down far enough, it turns into a stack. The stacks are still groups and so still namable and all that jazz.

Clicking causes the stack to fan out in place (that is the group doesn't grow in size but the the pages are arranged in an overlay over other the rest of TabCandy)

Some interaction notes:
* The top-left page is always the page which shows on the top of the stack.
* Adding a new page into a stack puts it on top (and hence in the top-left position).
* Each stack should have a different random rotation orderings, and that ordering should be preserved across unstacking/restacking. This gives each stack an iconic look.



Because we are using an overlay, we are no longer constrained by keeping the size of the group from overlapping with others pages and groups. Therefore we can size the tabs at a large visually-useful minimum size.

You can still drag pages around to re-order them inside the overlay.

You can drag a page out of an overlay to put it into another group. When it crosses the boundary of the overlay, the overlay shrinks away and you can place the page anywhere you like.

Option/Control Tab

Switching between tabs two or more tabs quickly is an important use case. In TabCandy, we borrow some of the paradigms of the operating system's alt-tab.

1 Hit option-tab

Timeline

Last week of April

Release Revision-A version of TabCandy. This release is fully dog-foodable, but is not feature complete or interaction complete.

This release starts gathering momentum behind the idea and gathers important feedback.

[Instrument this release?]

Last week of May

Release Revision-B version of TabCandy. This release is feature complete, meaning that features are frozen after this time.

This release should take into account the feedback we've received. We'll have blogged extensible about the process of moving from Revision-A to Revision-B.

Last week of June

Release Revision-C version of TabCandy. This release is feature complete, and interaction complete. We are now happy with all of the features and how they work.

By this point, we hope to have tens of thousand (if not more) users. I'd like to figure out how to integrate this with the beta versions of Firefox.

Last week of July

Release Revision-D. This release focuses on polish—both performance and fiddly edge-cases. By this release, we hope to have something we are feeling confident about trying to get onto trunk.

Metrics of Success

1. People spend less time hunting for a tab that they want to switch between. [quantitative]
2. Users can better manage the sites that they want to read later but don't want to bookmark. [qualitative]
3. People can still move quickly between a small set of tabs (i.e., the random access that the traditional tabs interface gives). [both]
4. People keep more tabs open at once while not increasing tab-switch times. [quantitative]
5. People garbage collect more aggressively. [quantitative]
6. Continues to work well for small sets of tabs. [qualitative]

Metrics of Success (from Beltzner)

*** Time to find a particular tab**
*** Average time spent hunting**
*** Raw switch time (back and forth)**
*** Psychometric time?**
*** User acceptance button**

Common task flows that we think people are doing with tab

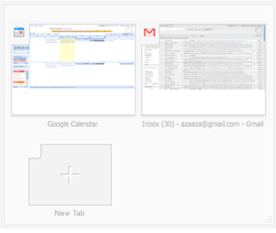
Open new/switch back and forth/actions required for these things

Finish the specification for minimum thing for shipping.!

Adding New Pages

When TabCandy is open

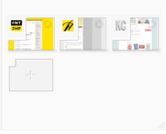
Inside a group



Each group includes a new page button.

Pages that are opened inside of groups using the new page feature have a clear spot that they will go — where the new page button is.

The new page button cannot be dragged around, and under resize either: (a) goes on the far left or far bottom at the same size as the other tabs in the tall/thin cases or where there is extra space, (b) below the other tabs scaled to a size that takes up the rest of the space, with some minimum height.



On the page in general

There should be a global new-page button. It appears in the "New Tab" group (i.e., where a new tab would appear if opened as an orphan).

Escape!

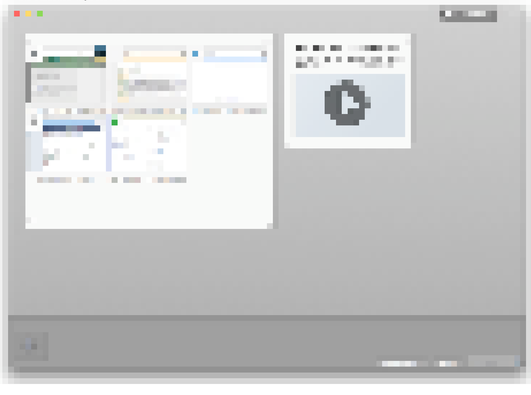
When a new user first enters Tabcandy we want to make sure that she knows how to get out. We've heard reports that people will hit the close button for the window (doesn't work) as well as click on the tab that just did the zoom animation (works) or use the escape key (works).

In this exploration we are looking at providing both a visual indication of the state of Firefox (see right) as well as a method for escaping.

Open questions:
1. Doesn't this break monotony? :(
2. Should it go on the left or right. We started with it being on the left because that is where the back button goes. However, since we have abandoned the back button metaphor, we instead favor it on the right.

This is the good one. In particular, the **right side** version.

First-Run Experience



All tabs opened prior to first invocation of tabcandy automatically get put into a group. There is also a little help-group that automatically gets shown on first-invocation (it doesn't go away until explicitly closed).

The movie does a half-way zoom (lightbox style) and is streamed from a Mozilla server. It explains in less than a minute all of the basic actions.

Open questions:
* How do I get the help back?
* Do we need explicit help for getting out of TabCandy? (probably answered by user testing)
* Where do closed groups go in general? (A history might be useful)