

[Pushing Pixels](#)

Leaving no pixel behind

- [In Motion](#)
- [Terms](#)

Search for:

July 25th, 2012

The usability of radial menus

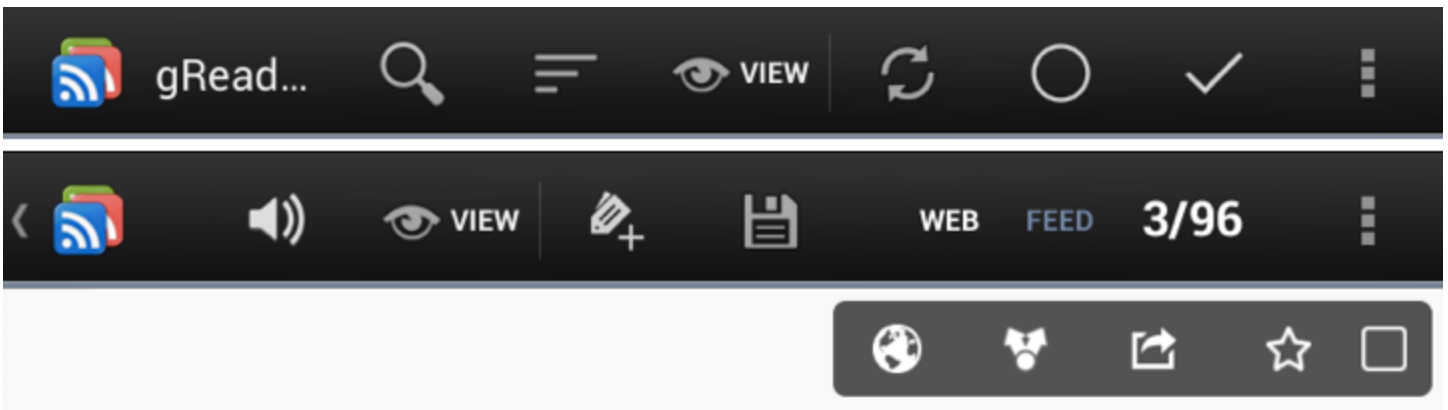
In his “[Touch means a renaissance for radial menus](#)” Josh Clark argues that the time has come for a widespread adoption of radial menus in touch-based interfaces:

The radial menu is seeing a renaissance in touch interfaces, and that’s a good thing. Microsoft yesterday previewed the Office 15 productivity suite [\[video\]](#), including OneNote, its first Metro-style app for Office. (Metro is the touch-based design language introduced in Windows Phone and now set to storm Windows 8 this fall.) OneNote features a radial menu as a kind of right-click contextual menu. Tap the ever-present menu circle, and out pops a wheel of icons to work on your current selection. [Here’s a clip from yesterday’s demo](#) that illustrates the action.



I can think of a few reasons why radial menus have not seen mainstream adoption beyond games. But the main one for me is how it looks to the first time user.

A few days ago I’ve decided to give [gReader](#) a try on my Nexus 7. Now while I’m its first time user, it’s certainly not the first time I opened an RSS reader on a smaller screen. I know what features to expect, at least on the basic level. But gReader’s entire interface is icons. A lot of them. Some of them are for basic features, some are for more advanced. And all are exposed via monochrome icons.



And that's a problem. I want to become quickly proficient with it, but I don't want to spend time guessing what each icon does. And not only do I not know what will happen when I press some cryptic icon, I also don't know what this operation will affect (single entry, all feed entries, all entries under this tag). Most importantly, there's no way to undo the last operation, or at least none that I could see immediately. After a particularly "unfortunate" tap I ended up marking about 50 entries as 'read' and had to spend some time figuring out how to find that exact chunk.

Now back to radial menus. As Josh points out there's a lot of research that indicates that people are much more efficient once they know the layout of each such menu. In a traditional drop down menu you have a relatively small margin of error when you want to select a specific element; the direction is always the same, and the selection is controlled by how much you move the mouse and how is that translated to onscreen pixels. In a radial menu, however, the distance is the same, but the direction matters. So if you know that 'bold' is always to the right, while 'italic' is to the top-right, your muscle memory will kick in to remember that:

Even better: you get faster with radial menus over time, because they take advantage of muscle memory in a way that list-based menus cannot. Radial menus are essentially gesture-based: touch-swipe-release. That's why some call radial menus "marker menus": it's like making a mark on the screen. Swiping to 2 o'clock has one meaning, and swiping to 6 o'clock another. Like all physical actions—playing an instrument, typing a keyboard, serving a tennis ball—**gestures get embedded in muscle memory, which is faster to access than visual memory.** Tap-swipe is faster than scanning for an item in a linear list, just like touch-typing is faster than hunt-and-peck.

But the problem is how well you can present these options to the beginner. A circle of eight icons is nice if each one is immediately and clearly understandable. And some operations are just too complex for a simple pictographic representation. So then you can display text descriptions for each one, but it's a mess. How do you display them in a way that is quickly scannable, uncluttered and does not take over the entire screen? And doesn't that take away the very goal of training your muscle memory? And adaptive UIs didn't work out that well (where the system adapts to your perceived level of expertise and usage patterns).

I think this is what has been standing in the way of translating the relative popularity of radial menus in games (and the shown increase in efficiency for long time usage). Gamers are willing to invest a lot of time if that improves the way they progress in a game. But throwing such an interface at a user of a 'boring' software package is going to be quite alienating. It's going to be interesting to see the user reaction to the proposed radial menu in OneNote.

Thanks for stopping by. My name is **Kirill Grouchnikov** and I am a user interface engineer. I currently work at Google on the [Android](#) team.

[more...](#)



Recent Posts

- [The elements of software style](#)
- [The usability of radial menus](#)
- [Reading on the web](#)
- [On cable TV and the leverage games behind the big money](#)
- [No control is an island](#)

From ["In Motion"](#)



- [Beth Mickle, production designer of "Drive"](#)



- [Jayse Hansen, screen graphics of "The Avengers"](#)



- [Romain Lacourbas, cinematographer of "Colombiana"](#)



- [Laurence Bennett, production designer of "The Artist"](#)



- [Katie Spencer, set decorator of "Hanna"](#)



- [Roberto Schaefer ASC, cinematographer of "Quantum of Solace"](#)

Connect

- [Google+](#)
- [Twitter](#)

© Copyright 2012 | [Pushing Pixels](#) | Original theme supported by [sell files](#) | All Rights Reserved