

## Slide 1

Harvard University, August 14, 2017

# Accommodating Speech Input: Challenges and Best Practices

Kim Patch  
617-325-3966  
kim@scriven.com  
[patchontech.com](http://patchontech.com)  
[redstartsystems.com](http://redstartsystems.com)

1

## Slide 2

# Questions for you

2

First, I've got some questions for you:

Who has tried speech input on the desktop?

Who uses it regularly?

Who uses it not only for dictation but also for commands-and-control?

Anybody use it hands-free?

How many people have tried it and wanted to use it but dropped it?

Why?

Anybody know anyone who uses it hands-free?

Who has been frustrated by desktop speech input?

Who's seeing someone else be frustrated by desktop speech input?

Who uses speech input on a smart phone?

Who uses speech input regularly on a smartphone?

### Slide 3

## My Background

- Journalist
- Speech input user since 1994 (Kurzweil, then Dragon Dictate starting with the first version for Windows)
- Largely hands-free for about five years, now I use a mix of speech and Wacom tablet
- Redstart Systems, speech input training
- W3C – standards
- Tools for journalists

3

Here's a little bit about my background.

My original profession was journalism. I got repetitive strain injuries from typing way too much. I got hurt just as speech recognition was available on the desktop in the mid-90s.

I used Kurzweil first, then DragonDictate starting with the first version for Windows. I was badly hurt and over the course of a couple years realized that I had to be completely hands-free on the computer in order to get my hands back for doing things like playing music.

I was happy with the way Dragon let me get words on the screen, but very unhappy with the inefficiency of speech input for command-and-control for things like navigating, moving windows around and traversing the web.

I wrote an extensive set of macros that eventually turned into an add-on for Dragon, called Utter Command, that made controlling the computer more efficient. First I had to build the scaffolding – the command-writing commands that, in turn, let me write commands hands-free.

I used Dragon largely hands-free for about five years. I started writing again. And my hands got better enough so I could start playing music again – I'm a fiddle player for a band that plays out regularly.

I still don't type. My keyboard lives on top of my computer under my desk. I'm sometimes hands-free and sometimes I use a combination of speech and the Wacom Art Pad tablet.

I've trained a lot of Dragon users – both hands-free and with the Wacom tablet combination.

I've been an invited expert with the W3C accessibility initiative for the past eight years, trying to make speech users concerns are included.

And circling back around to where I started, I'm now doing a project that has to do with tools for journalists: interactive transcripts and efficient transcription, including judicious use of automatic speech recognition for transcription.

#### Slide 4

## Speech Input Users

Keyboard/mouse use is

**Painful** and/or **Difficult** and/or **Slow**

Need to use speech for both dictation and command-and-control

4

The speech input users I train use speech because using the keyboard and mouse is painful, difficult, slow, or impossible.

As I mentioned, speech input can be useful – and is fairly easy to use out-of-the-box – when all you want to do is speed typing.

But people who have trouble using the keyboard and mouse need to use speech both for dictation and everything else. The everything else is the tricky part – full command and control means being able to navigate and make changes to text, navigate webpages including forms, work any type of control, move and size windows, and move among windows.

#### Slide 5

## 1. Desktop Speech Input

- Focus poses special problems
- Single letter shortcuts are terrible
- Cognitive load
- Quirks
- Dragon's HTML support
- Used well, speech is a great input method

5

I have six main points to make about desktop speech input, and we'll go through these one at a time in detail.

We'll talk about focus and how it poses special problems for speech input users.

We'll spend some time on single letter shortcuts – I'll show you why they're so terrible for speech users and some things you can do to mitigate that.

I'll also talk about cognitive load – paying attention to cognitive load is the key to unlocking the potential of speech input. And we'll look at what web developers can do to help lower cognitive load for speech users

I'll detail a couple of key rough edges that Dragon has – quirks like the double letter problem and issues with speech on the web.

And, even though the first five bullet points on this slide are somewhat negative, I'm also hoping to point out that speech input, used well, can be worth much more than the trouble it causes.

After I talk about these issues I'll do some demos on my laptop that touch on them.

### Slide 6

## 2. Mobile

- Speech button on the keyboard
- What's Apple waiting for?
- Agent
- Agent strategies
- The command-and-control gap on mobile

6

And after that, I'll touch on speech on mobile. There's the speech button on the keyboard, and the agent – Siri on iOS.

We'll talk a little about making the best use of speech on mobile, what's missing, and what's needed.

### Slide 7

## Equipment

- Relatively fast computer (i7 processor, 16 GB of memory, 1 TB solid-state drive)
- Dragon 15 Professional Individual (\$300)
- Good noise canceling microphone (Sennheiser ME 3) and USB soundcard pod (Andrea Pure Audio USB-SA Adapter)

7

So before we get back to desktop speech input issues I'm going to take a couple minutes to tell you about the computer I'll be doing demos on. This is a Lenovo Carbon X4 laptop – it's light – 2 ½ pounds, but it's fairly powerful, which it needs to be to handle speech well. It has an i7 processor, 16 gigabytes of memory, and a 1 gigabyte solid-state drive.

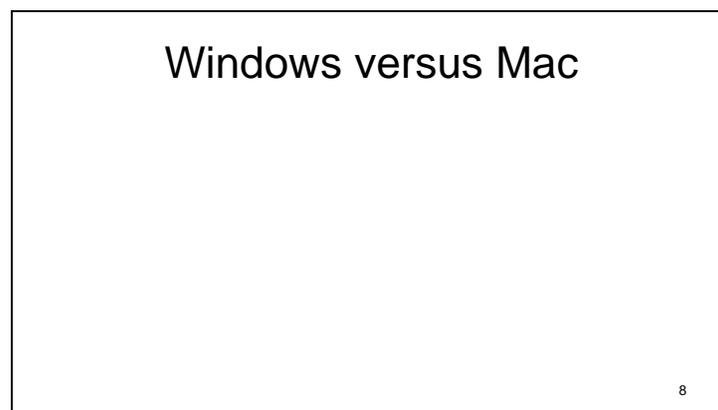
Speech input takes a lot of horsepower on top of whatever else you are doing. It starts slowing down – meaning it takes more time for every command – if you don't have a fast enough machine. This can be extremely frustrating for users.

There's a key thing to know about solid-state drives – they are fast, but only if they are less than half-full. A solid-state drive will start taking more time to access any given piece of information if it's more than half full. The way Dragon works, it frequently accesses the drive. So it's important – especially for Dragon users – to just think of the drive as half the size it is and keep it at least half empty. My recommendation is to aim for 45%. So my advice is to get a large solid-state drive and if you need more than half the space on it make use of portable drives.

A good noise-canceling microphone plugged into a USB soundcard pod is important for three reasons. It gives the speech engine a cleaner signal so the engine can more quickly decipher the words you are saying. It allows you to speak relatively softly. And it filters out background noise so you can use speech when, for instance, there's an air conditioner humming in the background, street noise drifting in from an open window, or a soft – or even loud – buzz in the room from some other equipment. It even allows you to use speech in a mildly noisy environment like a café.

The most important thing, though, is it gives the user a more consistent experience – it works pretty well everywhere rather than working well and then suddenly making many more mistakes and you don't know why because you've tuned out the noise of, say a clothes dryer running on a floor below you. This inconsistency can drive you crazy. I have personal experience with this. It took me about six months to figure out that starting a dryer on the first floor of my house meant I was going to have a horrible experience on the second floor with speech.

## Slide 8



I should also mention that I'm using Windows – it's Windows 7. Windows 10 also works well with Dragon.

The Mac version of Dragon is not as good at command-and-control by speech as the Windows version.

There's also good, accurate dictation built into the Mac operating system, including a nice command for correction, but native Mac speech input also doesn't include good command-and-control.

One reason the Mac version of Dragon isn't as good with command-and-control as the Windows version is the two versions were developed separately – the original developer licensed the Nuance engine, then Nuance later acquired that company.

The main problem with the Mac version of Dragon – this is something I regularly complain to Dragon about and encourage others to do so as well – is when you write a speech command you can't include a variable.

For instance, it's useful to be able to say a number and then "Down" to click the down arrow a certain number of times all at once. Using the Windows version of Dragon I can use the variable 1-100 to write a single macro for "1 Down" and "2 Down" and "3 Down" etc. – all 100 commands. On the Mac I've just enabled 1-10 Down, and it's 10 separate commands. Consequently I have a much smaller custom command set for the Mac.

The Mac version of Dragon also can't handle mixed input well in some programs. If you use the mouse or keyboard to change the cursor position Dragon loses track of where the cursor is, and your dictated words may be inserted at the wrong place.

Another complication is it's not easy to switch back and forth between the Mac and PC versions – because the Mac version of Dragon was developed by different company the commands are worded differently. You can fix this with custom commands, but it's not easy to switch back and forth out of the box.

I want to use speech on the Mac. I support some Mac Speech users, and I think Apple is doing some key things right with speech on mobile – more about that later. But currently the most powerful way to use speech hands-free is on Windows.

## Slide 9

### More hardware

- SR Pro-1 Stenomask microphone (Sylencer)
- Wacom Art Tablet

9

While I'm talking about hardware I'll mention two more things that speech users sometimes use and I find particularly useful. I occasionally find myself in a face-to face meeting where I have to take minutes. I have a Talk Technologies SR Pro-1 Stenomask microphone for that. This was made for court reporters. It's not as comfortable as the headset microphone, but it does what it needs to – lets you speak silently – just about, and with pretty high accuracy. This works with computers and also smartphones.

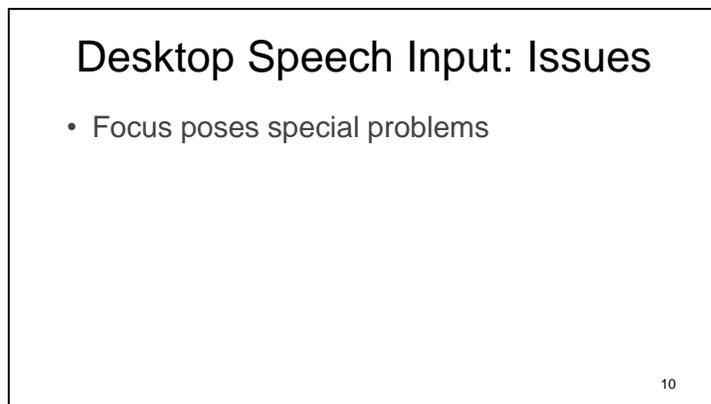
The Wacom Art Tablet is a mouse device I sometimes use in conjunction with Dragon. When I do a remote call with a user I can't always use speech on the remote machine. It's also a useful backup when I'm on the phone and want to limit muting for speech.

There are two key things about the tablet. Mouse positioning is absolute rather than relative, meaning the rectangle of the tablet maps to the rectangle of the screen. So you simply put the pen wherever you want the mouse to be – touch the center of the tablet and the mouse is at the center of the screen. I don't have to look for the mouse and then drag it from there – I just point to where I want the mouse to be. This means far fewer mouse movements.

I also don't have to click using the pen on the tablet. I can move the mouse using the tablet and click using speech. And often the click command is combined with something else. (I use the word Touch because it's easier to say than click – more about that later.) So, to click the mouse and paste I would say, "Touch Paste".

Absolute positioning is also especially useful for folks who have vision problems, because you don't have to find the mouse to move it.

## Slide 10



**Desktop Speech Input: Issues**

- Focus poses special problems

10

So, back to desktop speech input issues.

Focus poses special problems.

## Slide 11

## Focus poses special problems

### Problem

- Unlike using the mouse or touch as an input method, with speech there's no automatic focus before execution
- Speech users are the proverbial canary in a coal mine for focus issues

11

This is because speech input has been added on to an existing keyboard-and-mouse means of communicating with computers.

Mouse users automatically focus everything they do – the act of moving the mouse changes the focus. If you're using a pointer device or touch input, changing the focus and clicking is a package – you can't click on something you're not focused on.

Speech users, and to some extent keyboard-only users, are much more prone to user error when the focus isn't where the user expects it to be.

What's legitimately not working because the focus is wrong seems to the user like it's broken.

There's nothing special developers have to do for speech users here – just be really careful in following standard practice for focus and keyboard shortcuts.

### Slide 12

## Keyboard Shortcut Standards

- General Navigation
  - Predictable Tab order
- Menus and Trees
  - enable Up/Down/Left/Right, Enter
- Drop-down lists
  - enable multiple letters to navigate, Escape
- Checkboxes and Radio Buttons
  - Enable Space to toggle

12

Keyboard shortcuts are important for speech input users because we either speak them or use native speech commands that call them.

Here are the key keyboard shortcuts.

For general navigation predictable Tab order is important.

For trees and menus it's important to enable Up/Down to navigate, Left/Right to collapse/expand and Enter to activate.

For trees and menus and drop-down lists it's important to enable multiple letters to navigate and for drop-down lists it's important to enable escape to close the list.

For checkboxes and radio buttons it's important to enable Space to toggle.

### Slide 13

## Scrolling

- Keep scrolling to a minimum
- Provide
  - Large drop-down menus
  - Navigation links

13

Something else that helps speech users – and makes computing more efficient for everyone – is keeping scrolling to a minimum.

The key for developers is to provide large drop-down menus – use the available space rather than having users do extra scrolling.

And where appropriate use navigation links to give users a better mental map and make for fewer steps for back-and-forth navigation.

### Slide 14

## Focus poses special problems

**Solution**

- Developers: make sure focus is logical and apparent
- Speech users: be aware of focus, especially when something goes wrong

14

So the solution for the focus problem is to hold developers to good design practices.

## Slide 15

### Desktop Speech Input: Issues

- Focus poses special problems
- Single letter shortcuts are unworkable

15

Single letter shortcuts don't work for speech users, because our currency is strings of letters – we say everything in words.

## Slide 16

### Single Letter Shortcuts are unworkable

**Problem**

- Single character key shortcuts set speech users up for a horrible user experience

16

They're dangerous for speech users because it's easy to accidentally say whole string of keys when the focus is in the wrong field.

If Dragon interprets a stray noise as a word, or if you accidentally try to dictate when the cursor isn't in a text field, or if you say something when you think the microphone is off and it isn't, or if someone walks in and says something that's picked up by the microphone before you can turn it off -- you can trip a whole bunch of single letter shortcuts at once. Lots of things can happen at once that aren't so easy to reverse. And sometimes it's difficult to track down the string of things that have been done and undo them.

In terms of how likely Speech input users are to be affected by this issue, I like to use the metaphor of having a 10-foot-deep hole in the middle of a conference room and telling people to just avoid it rather than putting a railing around it. It's obvious that this is not good because at some point someone's going to fall in.

Even though I've been using speech input for a long time, if I'm using a program that has single letter shortcuts I know something's likely to happen to trip a string of them at some point.

For any of a half-dozen reasons, I'm going to think the focus is somewhere it's not, or I'm going to think the microphone is off when it's not, and I or someone else is going to say a phrase that the computer will then interpret and I will have to unravel.

Keeping vigilant enough to avoid this most of the time drains brainpower.

For less experienced users it's much worse. Something goes wrong and they don't know what happened, sometimes even if they're aware they are using a program or web app that uses single letter shortcuts. I've many times had to explain that it's not actually broken, it's just accidental activation of single letter shortcuts.

## Slide 17

| Gmail single letter shortcuts |                           |
|-------------------------------|---------------------------|
| <b>Compose and chat</b>       | <b>Navigation</b>         |
| p Previous message            | u Back to threadlist      |
| n Next message                | <b>k New conversation</b> |
|                               | j Older conversation      |
|                               | o Open conversation       |
| <b>Actions</b>                | <b>Application</b>        |
| x Select conversation         | c Compose                 |
| s Toggle star                 | d Compose in new tab      |
| <b>e Archive</b>              | g Search chat contacts    |
| <b>m Mute conversation</b>    | v "Moved To" menu         |
| r Reply                       | l "Label As" menu         |
| a Reply all                   |                           |
| f Forward                     |                           |
| z Undo                        |                           |

Here are the single letter shortcuts for Gmail.

I like to use the example "Hey Kim". Those six letters cause a lot of trouble. Three of them do things. E archives a message; K moves the cursor down one, and M mutes.

If you're bored and you have a Google account you don't care about try turning on Dragon and saying a long phrase into your inbox, and then try to figure out the damage. Anybody named Mike here? Any phrase that includes Mike is going to mute a conversation, go to a new conversation, then archive it.

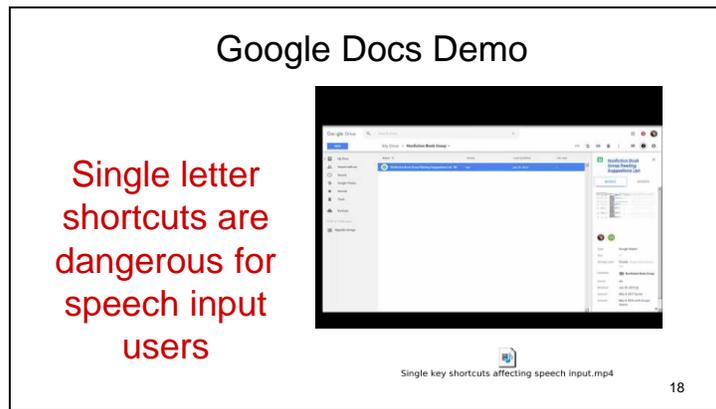
So now you have a new party trick. You can also try this out in Yahoo mail, and off-line, in Thunderbird. Twitter, Facebook Google Docs and WordPress also have single letter shortcuts.

Single key shortcuts also stymie some keyboard users who have mobility problems, and might accidentally hit keys.

So these are a big problem for a couple of different user populations. At the same time the shortcuts are useful. They speed up some keyboard users.

So the question is how do we accommodate everyone?

## Slide 18



Google Docs also has single letter shortcuts. It's easier to demonstrate this problem in Google Docs because then I don't mess up my email.

This shows how, with the focus in the wrong place, a single word or phrase can trip several single letter shortcuts in Google Docs.

There are two examples. The second one, a single word, opens the information dialog box and opens a Google drawing in a new window.

Audio: "Dead"

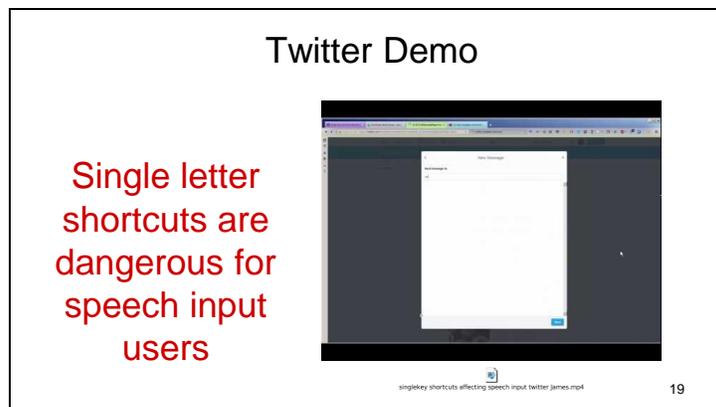
The first one, a short phrase, opens the information dialog box, opens a drop-down, and changes the document list from a list to tiles.

Audio: "I have an idea"

#### Single Key File

<https://youtu.be/xzSylA4OWYE>

#### Slide 19



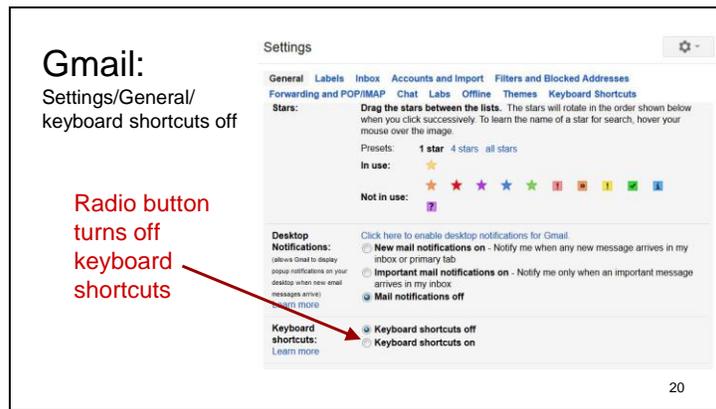
Here's an example of a single word tripping a couple of single letter shortcuts at once on Twitter:

"James"

#### Single Key Twitter File

<https://youtu.be/OPjfpDU9S08>

#### Slide 20

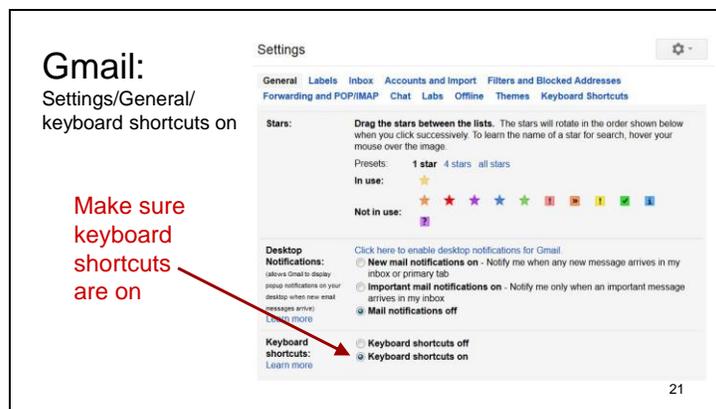


Let's go back to Gmail. In Gmail you can turn off keyboard shortcuts in in Settings – the General tab (click on the gear in the top right and choose Options).

We're switching back to Gmail because I can't show you a way to turn off single letter shortcuts in Google Docs or Twitter because you can't – in many programs and web apps, including Google Docs, Twitter, Yahoo and Thunderbird, you can't avoid the single letter shortcut trap.

Being able to turn single key shortcuts off removes the dangerous commands, but simply turning them off is not a perfect solution. It takes away the danger but leaves you without the commands.

**Slide 21**

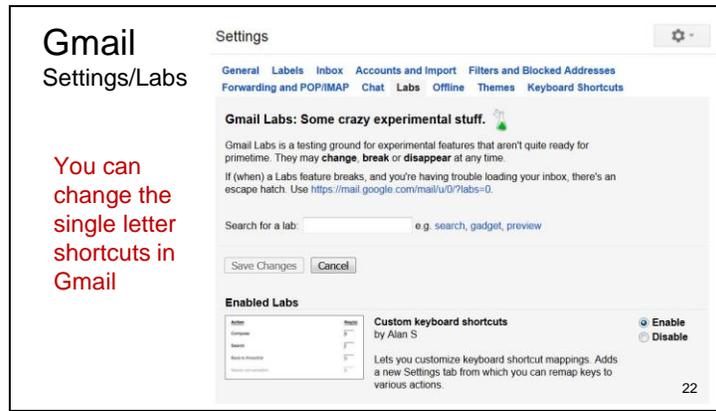


Fortunately Gmail addresses this in a better way as well.

It's a good thing, because there are a lot of single-key shortcuts.

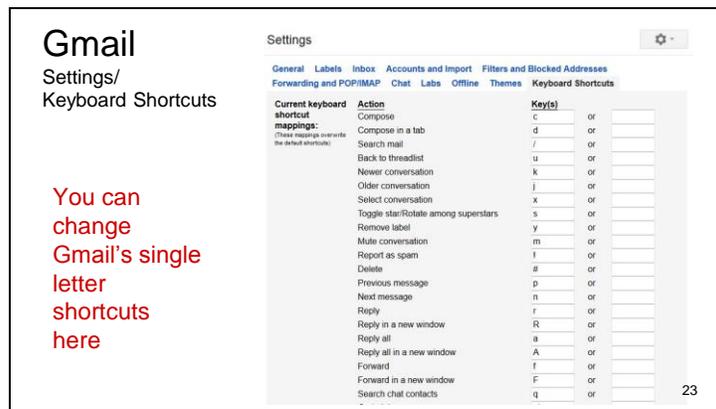
To do this you make sure keyboard shortcuts are on.

Slide 22



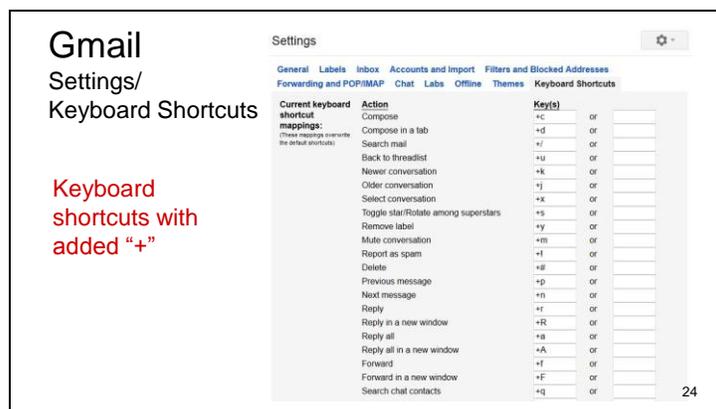
Then go to the Labs tab in settings, scroll down to find custom keyboard shortcuts, and click enable and save changes – it will then show up at the top in the Enabled Labs section like this. Once you do this the Keyboard Shortcuts tab will show up in Settings.

Slide 23



This tab allows you to change each single character shortcut in Gmail.

Slide 24



Here's how I use this ability – I put a “+” in front of every command. This takes away the danger of shortcuts being activated when I say a phrase. I can still use these by saying, for instance, “Plus Charlie” to return “+c”.

And I can still use Google's help to look up a shortcut. I just have to remember to add the plus. If I want to spend a little more time on this I can write custom macros to enable native speech commands like “This Reply”, “Reply All”, “Reply New Window”.

This takes some time, and requires me to keep up with changes in the software, but it works.

Notice the “or” column. Google lets you enable two sets of shortcuts at once. This is useful in theory – you might have two different people using the computer who might want different types of shortcuts. But it's not useful for speech users unless you can toggle between the two types, and there's no toggle button. So if a keyboard user wanted to use the standard shortcuts on my computer, it would be a tedious job to enable that.

So there are a couple of things that could be done better. One, it's great to have a couple of shortcut sets at once, but we need to be able to toggle them off and on separately. Two, the ability to import and export command sets would save users lots of time. If I could hand you a file to import instead of advising you to sit down at the computer and carry out the tedious task of putting a plus in front of every command, that would be better. There are a lot of shortcuts – this is just the first page of them.

But the best ability for speech input users would be a shortcuts column that allowed up to 25 characters for each command. Then, instead of changing the shortcut, then writing a custom macro in the speech input program attaching a couple of words to the shortcut “+ c” I could instead put the native speech commands here. And keeping up and sharing speech commands would be easier. And this would enable any speech engine you're using – not just Dragon.

## Slide 25

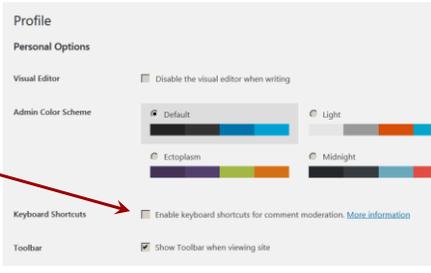
| WordPress: Comments Shortcuts |              |         |                                 |
|-------------------------------|--------------|---------|---------------------------------|
| j                             | move down    | Shift-a | approves checked comments       |
| k                             | move up      | Shift-s | spam                            |
|                               |              | Shift-d | deletes checked comments        |
| a                             | approve      | Shift-u | unapproves checked comments     |
| s                             | spam         | Shift-t | moves checked comments to trash |
| d                             | delete       | Shift-z | restores checks comments        |
| z                             | undo         |         |                                 |
| u                             | unapprove    |         |                                 |
| r                             | inline reply |         |                                 |
| q                             | quick Edit   |         |                                 |
| e                             | edit screen  |         |                                 |

25

WordPress also provides single letter shortcuts with a thoughtful solution for everyone. The only single letter shortcuts in WordPress are in comments, which really save keyboard users time. There are other ways to carry out all these actions.

## Slide 26

### WordPress: Comments Shortcuts Toggle



Checkbox  
Toggles off  
letter key  
shortcuts for  
comments

26

The image shows a screenshot of the WordPress 'Profile' settings page. The 'Keyboard Shortcuts' section is highlighted with a red arrow pointing to the 'Enable keyboard shortcuts for comment moderation' checkbox, which is currently unchecked. A red text box on the left contains the text 'Checkbox Toggles off letter key shortcuts for comments' with an arrow pointing to the checkbox. The page number '26' is in the bottom right corner.

And you can turn them off.

## Slide 27

### Single Letter Shortcuts are unworkable

**Solutions:**

- Better-than-nothing: way to turn off single letter shortcuts
- Better: way to remap shortcuts to include a symbol
- Great: way to remap shortcuts to include up to 25 character keys, including a sharing mechanism

27

The image contains a list of three solutions for single letter shortcuts. The first is 'Better-than-nothing: way to turn off single letter shortcuts'. The second is 'Better: way to remap shortcuts to include a symbol'. The third is 'Great: way to remap shortcuts to include up to 25 character keys, including a sharing mechanism'. The page number '27' is in the bottom right corner.

So the solutions for single letter shortcuts are straightforward.

## Slide 28

### What you can do

**Developers:** make sure single letter shortcuts can be toggled off, or better yet, adjusted, saved and shared.

**Others:** complain as loudly as possible.

28

The image contains two paragraphs of text. The first is 'Developers: make sure single letter shortcuts can be toggled off, or better yet, adjusted, saved and shared.' The second is 'Others: complain as loudly as possible.' The page number '28' is in the bottom right corner.

But many programs that use single letter shortcuts don't let the user adjust the shortcuts.

So if you're developer, you now know what to do, and if you are not a developer, you've got the information you need to complain about this as loudly as possible.

Single letter shortcuts are dangerous for folks with mobility issues, and a disaster for speech users.

#### Slide 29

### Desktop Speech Input: Issues

- Focus poses special problems
- Single letter shortcuts = unworkable
- Cognitive load

29

It's important to think about cognitive load no matter what you are doing. But with speech, cognitive load is inherently high.

#### Slide 30

### Cognitive load

**Problem**

- Remembering commands

30

Speech input has a lot of promise because there is no real estate problem – the keyboard has a limited number of keys. Mouse and touch input have just a few actions each – a few kinds of clicks and drag, or one or more types of presses and some gestures. So to get enough commands you have to go through several steps – often levels of menus.

Speech input has no physical real estate limit like the keyboard, mouse and touch. But there's another limit.

Instead of being limited by the number of actions you can carry out, you're limited by the number of actions you can easily remember.

This is why speech users sometimes complain that their brains hurt at the end of the day. Thinking of what to say to command the computer is work.

### Slide 31

## Command Wording

- It's important that commands be consistent
- So they become habit
- So you can lower cognitive load

**Consistency = habit = easier**

31

It's important to be consistent with commands  
So they become habit  
So you can lower cognitive load

Dragon's commands tend to be inconsistent – you get lots of ways to do something – for instance 24 ways to go to the end of the line. The best thing to do is pick one way and stick with it.

### Slide 32

## Intag

- Third-party add-on
- Say what you see

32

Another tool that mitigates the cognitive load problem is Intag. It's a third-party tool that numbers elements and links. All you have to do is say the number. Saying what you see is cognitively easier than remembering a command. I'll come back to this later.

**Slide 33**

**Cognitive Load**

**Solutions**

- Use efficiently worded commands
- Use Intag to limit memorized commands

33

So the keys to limiting cognitive load are using efficiently and consistently worded commands and also using numbered links and elements to give yourself a break from thinking of commands.

**Slide 34**

**Desktop Speech Input: Issues**

- Focus poses special problems
- Single letter shortcuts are unworkable
- Cognitive load
- Quirks

34

Despite being around for more than 20 years

**Slide 35**

**Quirks**

**Problem**

- Despite being a 20-year-old program, Dragon still has some quirks

35

Dragon still has some major quirks.

## Slide 36

### Quirk Demos

“Notepad Open”  
“Size 50”  
“Cap Twenty-five”

  
Double Character Demo.mp4

36

It often picks the wrong homophone, even when it seems obvious that it shouldn't. Say “Cap Twenty-five” and Dragon will return numeral “25”.

Let me show you.

**Notepad Open**  
**Size 40**  
**Cap Twenty-five**  
**Window Close Right Enter**

And it sometimes returns double characters.

Most of the time I use two factor authentication in Google, the first number comes out double. So I have to notice this, and then go back and delete the first number. Here's what happens – and how I correct it.

**Double Character File**  
<https://youtu.be/OPjfpDU9S08>

## Slide 37

### Quirks

**Solution**

- Know how Dragon works and warn users
- Complain to Dragon

37

The solution to Dragon quirks is just to know them, and keep complaining to Nuance.

## Slide 38

### Desktop Speech Input: Issues

- Focus poses special problems
- Single letter shortcuts are unworkable
- Cognitive load is key
- Quirks
- Dragon's HTML support

38

Dragon's HTML support has some quirks as well.

## Slide 39

### Dragon's HTML support

#### Problems

- Bad choices: accidental clicks versus having to say "click" all the time
- Hidden commands
- The fallback solution, using the mouse by speech, requires several commands for each mouseclick

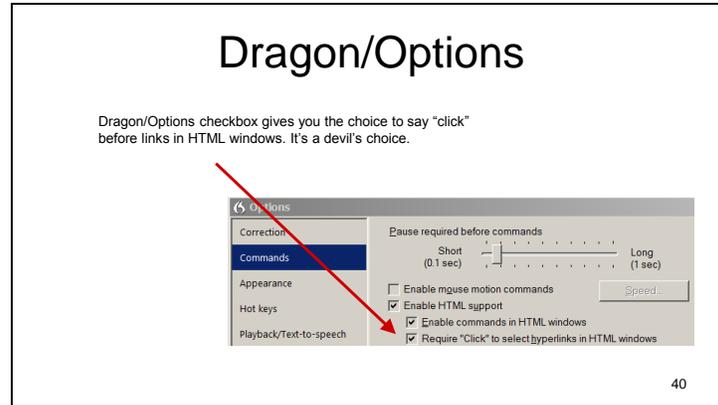
39

There is an inherently bad choice in the way Dragon enables speech users to click links. Users are either prone to accidental clicks or have to say the word "click" an inordinate number of times throughout the day. Click is a particularly bad choice for a frequently used speech command because it's hard to say.

Dragon also enables hidden commands, which are prone to being accidentally invoked because it's hard to avoid what you don't know is there.

And if you want to use Dragon's mouse solution as an alternative it's very slow going. It takes several utterances to do a single mouse click.

## Slide 40



Dragon has always allowed users to click links by saying any words or combination of words contained in the link.

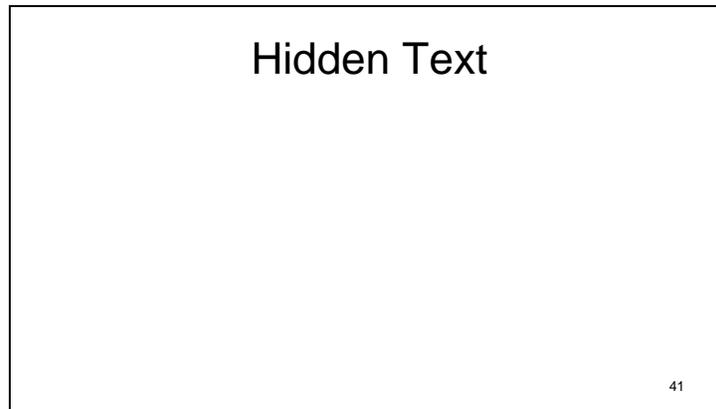
The problem with this mechanism is, especially on pages with lots of links – you can easily accidentally click links if you happen to say any word or combination of words in any of the links (with a pause before and after it). Dragon/Options/Commands gives you the option to proceed any of these commands with “click”. Click around a website for 10 minutes and you’ll grow very tired of saying “click”. It’s a word that’s especially hard on your voice. Try saying it out loud a few times (you’re using the back of your throat and it gets tiring. Dragon also uses click to click the mouse. You’ll hear me saying “Touch” instead of “click” when I’m controlling the mouse by speech – you say that word using the front of your mouth and so it’s easier to say a lot).

I wish I could say that Dragon has, over the past 20 years listened to user complaints about this enough to give us an option to use another word rather than “click”, but I can’t. They haven’t.

My solution, once again, is to use the third-party tool Intag to click links. Saying numbers is cognitively easier and you’ll save your voice.

(As a fallback, Utter Command contains a mouse tool that lets you click anywhere on the screen using a single command, as if the screen is a 100 x 100 grid. I use both).

## Slide 41



Starting in version 13, Dragon added ARIA support, but there's an inherent problem with enabling spoken labels that might be different from what's visible on the screen.

This means some commands that speech input users can say can be hidden. The concept of a hidden command isn't so much a problem for other means of input because you don't usually go pressing key combinations randomly, and if you do, you're aware of what keys you're pressing.

However, if you pause before and after a group of words that makes up a command, the speech input engine returns the command, not the words. This is the way speech engines generally work, and it generally works really well, especially with a well-scoped command set. But if you have a whole bunch of hidden, extra commands, and worse yet, commands made up of words that you are likely to use to input text on that page, it becomes easy to unexpectedly click a hidden command.

And when this happens it can be difficult for users to tell what happened – maybe it's a link and the user is whisked off the page. And if the user does get back to the page sometimes text they've entered is gone. So hidden speech commands are minefields.

The current choice for the speech input user is to enable web commands that include a bunch of hidden commands, enable the commands with a preceding word like "click", which is hard on your voice, or turn off the ability to click links by speech. Given these choices many users turn off these commands. I have them off now.

If I turn them on, and happen to say any isolated word or phrase contained in one of what might be as many as 100 aria-label tags on any given webpage, something unexpected would happen. Whatever might happen is usually not a disaster, and sometimes it is.

## Slide 42

## Dragon's HTML support

### Solutions

- Use a command set that's well scoped, and well understood, which means no live hidden text
- Use Intag instead

42

If the user is doing fairly simple things infrequently— just clicking a couple of things on a webpage, the simple choice is to make sure the preceding click option is turned on and use that.

But for most users, I recommend turning off Dragon's HTML support and using Intag.

### Slide 43

## Desktop Speech Input: Issues

- Focus poses special problems
- Single letter shortcuts are unworkable
- Cognitive load
- Quirks
- Dragon's HTML support
- Used well, speech is a great input method

43

Now that I've outlined the issues, I'll show you how it's possible to tap the strengths of speech input and avoid the pitfalls.

### Slide 44

## My Setup

- Dragon Pro Individual
- **Dragon add-on Intag** (#'s elements and links)
- **Dragon add-on Utter Command** (adds command phrases)

44

I'll show you some demos. Again, I'm using Dragon NaturallySpeaking Pro on a fairly fast computer – an i7 processor, 16 gigabytes of memory, and a solid-state hard drive that I keep at least half empty to keep it running fast. And I use a particularly good noise cancelling microphone, the Sennheiser ME 3, plugged into a soundcard pod.

I'm also using two third-party tools, Intag, which numbers elements and links, and Utter Command, which is consistently worded and allows for command phrases.

I should also mention that some of the Intag commands you'll hear are reworded to be consistent with the Utter Command style wording. This wording style is consistent, concise and follows the order of events.

### Slide 45

## Speech Input Toolkit

### Mouse

- **Intag** (click on anything, low cognitive load)
- **50 By 50** (change focus)
- **Name a mouse click** (saves steps)

### Keyboard

- **Keyboard phrases** (saves steps, low cognitive load)

### Native speech

- **Site, Folder, File, Tab, Enter lists** (saves steps)
- Search the web for...

45

Here's more detail on the speech input tools I use. The demos will include all these methods.

Intag numbers active elements on the screen, including links on webpages.

Utter Command allows you to speak mouse coordinates based on a 100 x 100 grid – a single command places and clicks the mouse – this is much faster than Dragon's MouseGrid, which requires multiple steps.

I use a second Utter Command mouse tool – the ability to name a mouseclick. This takes set up, but speeds actions that you might use frequently that require clicking, then clicking again – like choosing from a drop-down. You can use this to quickly change colors in Google documents and Word.

Utter Command also allows you to speak keyboard shortcuts in phrases. This takes advantage of existing knowledge. It saves steps and lowers cognitive load.

Utter Command also allows you to maintain lists of sites, folders and files you go to frequently. These save steps.

You can also maintain lists of words you want to mix with Tab and Enter keys. You can also have different spoken and written forms – these also save you steps.

One thing Dragon does really well is search. I'll show you that too.

#### Slide 46



#### 1 Escape

#### Screen Clear

#### Show Numbers

Say a number to single click on that element

18 (Program)

16 (Folder)

Notice the nice feedback – you can see the click happening. I'll do the same thing again.

18

16

Say a number followed by a mouse command to do more than a single click. You can say a number plus Touch Twice, Touch Right, Control Touch or Shift Touch.

#### 30 Touch Right

#### 30 Touch Twice

On programs that are enabled, you can get a lot of link numbers. This is useful if you want to click around a program to see what's there or how things work.

9

## 1 Escape

6

Also notice that numbers don't appear on elements that are grayed out

## 1 Escape

### Window Close

### 40 By 40

You can also specify that they appear just in certain areas.

### Taskbar Numbers

6

### Window Close

Intag also lets you drag – you can put email in folders this way. I'll show you the drag function from the desktop.

### 30 Drag To 14

### Undo That

Want to see that again?

### [Refresh numbers]

### 30 Drag To 14

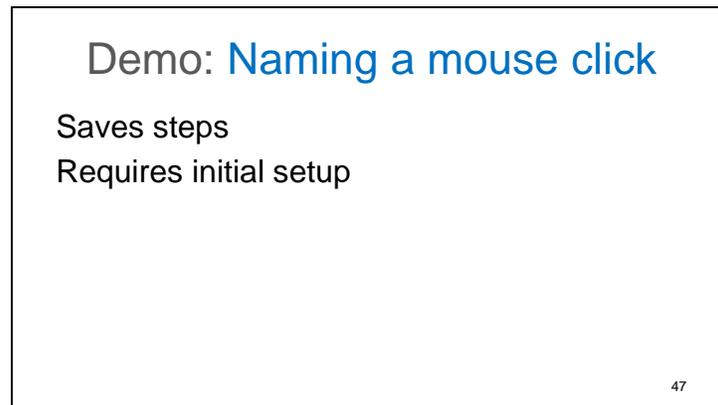
### Undo That

Even though you can easily turn them off, all the numbers can get busy. The other thing that Intag allows you to do is make your own overlays – you can only tag what you know you need so there are fewer numbers.

## Hide numbers

### Powerpoint Open

### Slide 47



Picture a Web app that uses drop-down lists for color. You should be able to access these via keyboard shortcut and then use the up/down and enter keys to select a color, or via numbered links. If you can't, you have to use the mouse by speech. This is tedious. We do have a workaround, but this workaround requires set up, so it's only worth it to enable something you use a lot. This is the naming a mouse click workaround – to set it up I put my cursor on the places I need to click and name them beforehand. Once it's set up it's a pretty good workaround.

I'm going to show you this in WordPad. This method can be used anywhere, including in a word processing app like Google Docs. I'll change colors by speech a couple of times, then I'll add a bullet.

**Delta File**

**Window 0 By 0**

**Size 24**

**I want to bullet this line and make it red**

**Left Backspace Right**

That last command was speaking a keyboard phrase to make a correction

Watch the cursor – this next command is a named mouse click that clicks the Bullet icon

**Bullet Touch**

**This Line**

This next command clicks twice – I'll do a few of these

**Color Red Touch**

**Words I want**

**Color Blue Touch**

**Words and make**

**Color Gray Touch**

**Go End**

**Window Close · Right Enter**

Slide 48

**Demo: Command Phrases**

Speaking keyboard in phrases is more comfortable and efficient than single keys

Efficient wording makes phrases practical and lowers cognitive load

Keyboard phrases tap existing knowledge

48

After the first couple of commands you'll hear me speaking keyboard phrases – I'm just saying the keys in the order I'd press them. I'm using the alpha-Zulu alphabet for letters because they're better recognized that way.

**Notepad Open**

**Size 24**

**cap Fact**

**Left Backspace Sierra**

**Backspace · Right Echo**

**Backspace Left · Backspace India**

**End Exclamation**

**Exclamation Times 5**  
**Line Delete**  
**Window Close**

You'll see more of this in the Forms demo, next

**Slide 49**

| Command Phrases                    | Commands |
|------------------------------------|----------|
| "Notepad Open "                    |          |
| "Size 24"                          |          |
| "Cap Fact"                         |          |
| "Left Backspace Sierra"            |          |
| "Backspace · Right Echo"           |          |
| "Backspace Left · Backspace India" |          |
| "End Exclamation"                  |          |
| "Exclamation Times 5"              |          |
| "Line Delete"                      |          |
| "Window Close"                     |          |

49

Here are the commands I just said.

**Slide 50**

**Demo: Forms**

Speech input enables users to carry out several steps at once.

This works well when tab order is done properly.

50

One of the things speech input enables users to do is carry out several steps at once. This is one of the things that makes speech input very powerful.

I use a lot of combined commands to save steps – here's an example.

This works well when the tab order on pages where tab order is properly done.

**1 Escape**  
**Screen Clear**

**Demo 2 File**

## **Zoom In 2**

**2 Tab Kim · 1 Tab Patch**

**3 Tab Address · 2 Tab Boston**

**1 Tab Massachusetts · 2 Tab Zip**

**1 Tab Email · 1 Tab Phone**

## **Zoom Out 2**

Now I want to move the cursor back up to the top to show you something else. There are several ways I can move the cursor. Right now I'll do it using Intag.

## **Show Numbers**

**47**

I can leave the numbers up or I can hide them

## **Hide Numbers**

Now, watch carefully – this is an example of speaking keyboard – I'm going to say 2 keys in a row

## **Tab Delete**

I'll do the same thing again – watch the cursor

## **Tab Delete**

Now I'll speed things up

## **Tab Delete Repeat 10**

I used 13 speech commands to zoom in twice, fill out 10 fields, zoom out twice and delete 10 fields (it would have taken 53 steps by keyboard and mouse to do this not counting letters). This is where speech input can really get powerful -- but only if Tab order is predictable

[I mentioned before that I can also move the cursor back to the top different ways – here's by speaking keystrokes.

## **Shift Tab Repeat 10**

## **Shift Tab Repeat 2**

And here's another way to do this – using Utter Command mouse commands, which let you place and click the mouse using a single command. There's a low-tech helper for this – I have numbers you can print out and laminate and tape on two edges of your screen. They're always there when you want them and they don't take up screen space.

## **Tab Repeat 10**

**18 By 18**

## **Window Close**

**PowerPoint Open**

**Slides Touch**

**Slide 51**

## Filling out Forms Commands

"Demo 2 File"  
"Zoom In 2"  
"2 Tab Kimberly · 1 Tab Patch"  
"3 Tab Address · 2 Tab Boston"  
"1 Tab Massachusetts · 2 Tab Zip"  
"Zoom Out 2"  
"Show Numbers"  
"47"  
"Hide Numbers"  
"Tab Delete"  
"Tab Delete"  
"Tab Delete Repeat 10"

"Shift Tab Repeat 10"  
"Shift Tab Repeat 2"  
Tab Repeat 10  
18 By 18  
"Window Close"

51

Here are the commands I just said.

### Slide 52

## Web Demo

- Navigating links
- Navigating among tabs
- Search

52

Now I'll show you how an experienced speech user might do basic Web navigation. Here I'm using a mix of Intag, Utter Command and straight Dragon.

**Weather Site**

**Amazon Site**

**New York Times Site**

**Google Advanced Site**

**Kentucky Derby**

**30 By 30**

**Screen Down**

**Show Numbers**

**13**

**2 Down Enter**

**22**

**Hide Numbers**

**Go Back**

**2 Tab Back**

(Here are the Dragon search commands)

**Search Web for green apples**

**Search Web for accessibility**

## Document Close Times 5 Window Close

### Slide 53

### Basic Web Commands

|                        |                                |
|------------------------|--------------------------------|
| "Weather Site"         | "22"                           |
| "New York Times Site"  | "Hide Numbers"                 |
| "Amazon Site"          | "Go Back"                      |
| "Google Advanced Site" | "2 Tab Back"                   |
| Kentucky Derby         |                                |
| "30 By 30"             | "Search Web for green apples"  |
| "Screen Down"          | "Search Web for accessibility" |
| "Show Numbers"         | "Document Close Times 5"       |
| "43"                   | "Window Close"                 |
| "2 Down Enter"         |                                |

53

Here are the commands I just said.

The site and keystroke commands were Utter Command. The numbers were Intag. The 3 search commands at the end were straight Dragon.

### Slide 54

## The Power of Speech Undo

54

### Word Open Size 20

I'm doing this demonstration in Word, but it's applicable in any text field. Implement undo well, and you give speech users a superpower. I'm going to have to put down a little more text to show you the superpower period

### 7 Befores

**text to properly demonstrate the superpower period**

**New Line Okay, I'm almost ready period**

**New Line Just about there now period**

**Line Delete**

**New Line Almost there period**

**New Line Very very close now.**  
**New Paragraph Okay, I'm ready period**  
**7 Undo**  
**5 Undo**  
**20 Redo**

## Slide 55

### What Developers & Designers Can Do

- Be predictable with Tab order
- Follow keyboard shortcuts standards
- Provide good undo support
- Limit scrolling
- Let users adjust, organize and share keyboard shortcuts

55

To summarize – here are the key things that designers and developers can do to make speech users' lives easier.

Be predictable with tab order – the predictability enables us to use more sophisticated commands.

It's also important to provide good undo support. This makes it so a mistake isn't the end of the world.

In general follow keyboard shortcuts standards -- the less the user has to think about what web site or program she's using the more brainpower is freed up to do actual work.

If you're an app developer or designer, and especially if you use single key shortcuts, make sure to allow users to adjust the shortcuts. And if you include a facility to organize and share keyboard shortcuts, you'll save a lot of user time. If I could share my gmail shortcuts, and you wanted to try them out, for instance, it would be much easier to just load them instead of spending 20 minutes putting a plus sign in front of every single command.

Limit scrolling – this is good for everyone.

## Slide 56

## Questions about speech input on the desktop?

56

Questions about speech input on the desktop?

### Slide 57

## Some Good News

WCAG 2.1 There are several current proposals that would improve things for speech users.

**CAVEAT:** WCAG 2.1 is a moving target, the exact wording may change.

57

Switching hats for a minute, I can mention that there's some good news on the horizon in terms of standards and speech. There are several current proposals that look like they will be in the 2.1 update of WCAG that would improve things for speech users.

I have to add a warning here – the update is still a moving target so the exact wording of these may change. So there's a big caveat for all of these.

### Slide 58

## Character Key Shortcuts

If a keyboard shortcut consisting entirely of one or more character keys is implemented by the content, then a mechanism is available to turn it off or to remap it to a shortcut that uses at least one non-character key.

58

Character Key Shortcuts tackles the keyboard shortcut problem.

The guidance is, if web developers provide single-key shortcuts, which are good for keyboard users, they make sure users have a way to turn off the single character shortcuts or – even better, remap them to a shortcut that’s more appropriate for that user.

### Slide 59

## Concurrent Input Mechanisms

Web content does not restrict use of input modalities available on a platform except where the restriction is essential, required to ensure the security of the content, or respect user settings.

59

Concurrent Input Mechanisms gives some guidance about making sure not to restrict the user’s choice of input modalities.

This helps speech input users because sometimes web developers don’t realize users may be using speech to, for instance, pick an item off a drop-down menu. If you can type a string of keys to do this, you should also be able to speak a word to do this.

### Slide 60

## Accessible Name

Where an active control has a visible label, the accessible name for the control includes the text string used for its visible label.

60

And Accessible Name gives some guidance about aligning the words users see on the screen that are associated with a control like a button with the accessible name of the control that can be accessed programmatically.

Aligning the visible label with the accessible name lays the groundwork for speech programs to leverage the accessible name for control in a way that's discoverable by users.

### Slide 61

## 2 - Mobile

- Speech button on the keyboard
- What's Apple waiting for?
- Agent
- Agent strategies
- The command-and-control gap on mobile

61

Now, I'll talk a little bit about speech on mobile. There's the speech button on the keyboard, and the agent – Siri on iOS.

### Slide 62

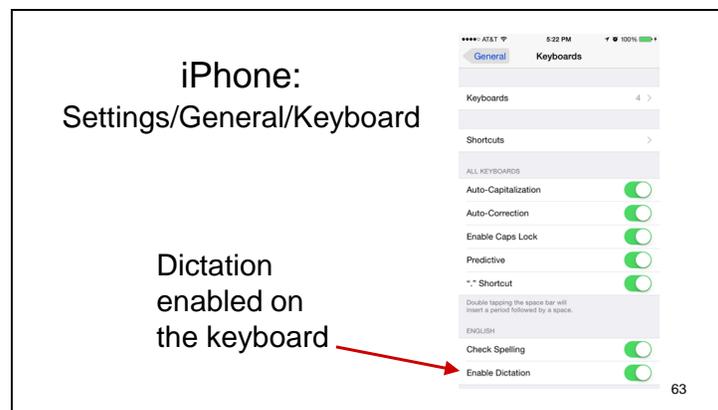
# Mobile

- Speech button on the keyboard

62

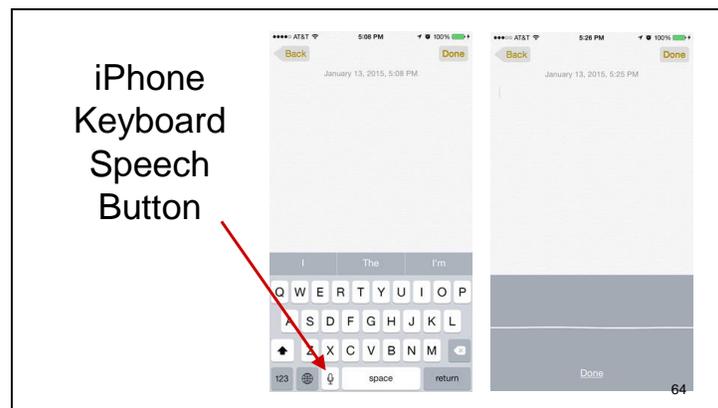
First, the speech button on the keyboard.

## Slide 63



You'll see the speech button show up on your keyboard if you have it enabled in settings/general/keyboard.

## Slide 64



The speech button shows up to the left of the spacebar on the system keyboard (some alternative keyboards don't retain the speech button).

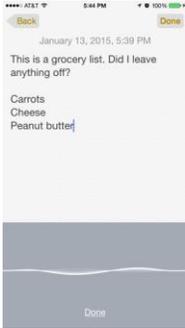
When you hit the button a line appears that moves when you talk. Words appear on the screen. You can hit done when you're done or wait until it turns off automatically because there's a silence or enough time has gone by.

## Slide 65

### Dictating Text

Speak punctuation, line and paragraph commands without pausing.

Example:  
This is a grocery list **period** did I leave anything off **question mark**  
**new paragraph** carrots **new line**  
cheese **new line** peanut butter



The screenshot shows an iPhone text entry interface. At the top, it says 'Back' and 'Done'. Below that, the date and time are 'January 13, 2015, 5:39 PM'. The main text area contains the dictation example: 'This is a grocery list. Did I leave anything off? Carrots Cheese Peanut butter'. A blue line is visible at the bottom of the text area, and a 'Done' button is at the very bottom.

65

Dictating is basic on the iPhone. You can speak punctuation, line and paragraph commands, but you can't make changes or corrections by speech.

Speak punctuation, line and paragraph commands without pausing to speed things up. If you say the words in the example you get what appears on the screen in the image.

[Example: This is a grocery list **period** did I leave anything off **question mark**  
**new paragraph** carrots **new line** cheese **new line** peanut butter]

Recognition has gotten really good, but Apple still hasn't given us a way to correct text by speech – that still has to be done via touch. Starting with the iPhone 6s – two rev's back – you could dictate without having to be connected to the Internet (you can test this by putting your phone in airplane mode and trying to dictate). This was good news – now that the speech engine software resides locally on the iPhone, it is in theory easier to do correction commands.

## Slide 66

### Mobile

- Speech button on the keyboard
- What's Apple waiting for?

66

So my question to Apple is “when are we going to be able to correct using speech input?”

**Slide 67**

## Dear Apple

Now that we can use speech input on iPhones and iPads without having to be connected to the Internet, we need to correct text by speech. How about adding my favorite Mac dictation correction command “Replace X with Y”?

67

Since Yosemite, Mac users have been able to correct by speech using the excellent correction command “Replace X with Y”. We need that on mobile. If you agree, this is something to pester Apple about – more about the pestering part later.

**Slide 68**

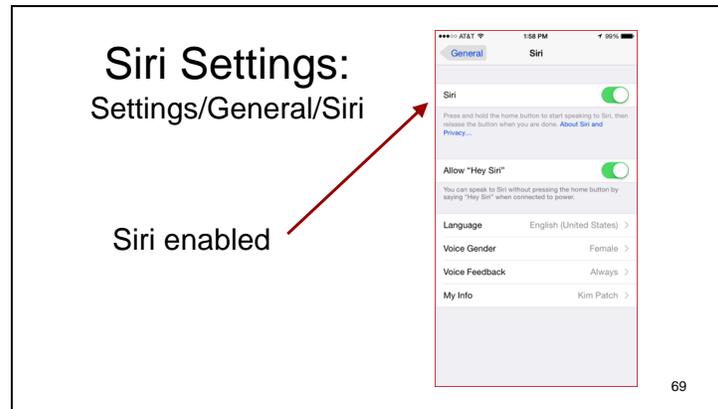
## Mobile

- Speech button on the keyboard
- What’s Apple waiting for?
- Agent

68

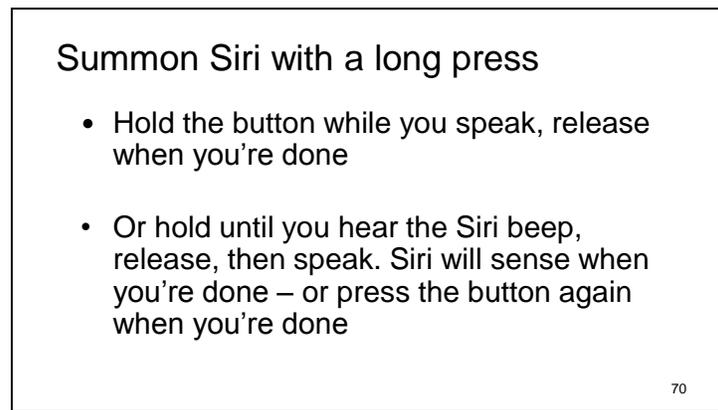
You can also use speech on mobile through an intelligent agent – on the iPhone, Siri

## Slide 69



This works if Siri is enabled in Settings/General/Theory

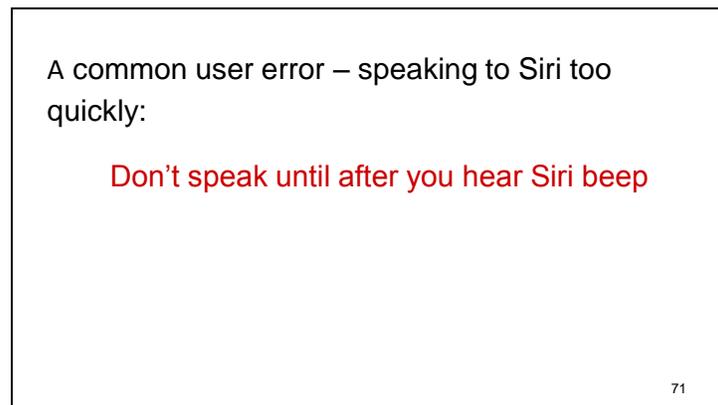
## Slide 70



There are two types of Siri long presses. The first one is quicker, the second one is probably easier on your hands.

[Demonstrate: Hi Siri are you there]

## Slide 71



[You might've noticed there was space between the Siri beep and when I started talking. ]

It's important to speak only after you hear Siri beep, and not anticipate things and speak too soon. Otherwise your message may be clipped.

If Siri doesn't hear the first word or the first piece of the first word your command is likely to be interpreted incorrectly. This is probably the most common cause of errors.

### Slide 72

## Mobile

- Speech button on the keyboard
- What's Apple waiting for?
- Agent
- Agent strategies

72

So does it matter how you talk to smartphone agents like Siri? It does.

### Slide 73

## Two Types of Things You Can Say

1. Question: Based on your question, Siri decides whether to do a web search, a Wolfram Alpha search or use an app to give you an answer
1. Command: Ask Siri to **open** an app, **show** you a deeper look into an app, or use an app to carry out an **action**

73

There are two types of things you can say to Siri.

You can ask a question, and based on your question, Siri will decide what to do – a web search, Wolfram Alpha search or do something with an app.

Or you can issue a command that directs Siri to do something more specific.

Both types are useful.

It's a good strategy to direct Siri more closely when you can because you can be more methodical in your wording, which makes good things happen for both Siri and you. Methodical commands are well recognized. And they become habit for you, which lowers cognitive load.

#### Slide 74

### Command Wording

- It's important to be consistent with commands
- So they become habit
- So you can lower cognitive load

Consistency = habit = easier

74

This slide should look familiar. It doesn't matter whether you are on a desktop or mobile phone. Your brain works the same way.

It's important to be consistent with commands  
So they become habit  
So you can lower cognitive load

Although Siri is advertised as an input method that you can say anything to, strategic use of words makes Siri more likely to do what you want rather than spit back something that makes for a good joke.

It also makes you more likely to use Siri because you'll repeat what actually works.

#### Slide 75

### Siri Commands That Work Well For Me

- **Open** Commands  
Open any app (but not home screen icons that you add)
- **Show** Commands  
Deeper view (notes, email, calendar, reminders, alarms)
- **Action** Commands  
Carry out action (write a note, reminder, calendar item, alarm, or email; call someone, get directions, search web)

75

You can think of Siri commands in three categories – open, show and action.

Open commands simply launch an app.

[Unfortunately, this still doesn't work for home screen icons that you add – for instance a direct link to a website that you've added by clicking on the share button and choosing “add to home screen”.]

Show commands get you a deeper view of some key apps.

And action commands let you do something in an app.

### Slide 76

### Open and Show commands syntax

Open <app name>

Show notes/reminders/calendar/alarms/email

Show notes/reminders/calendar/email <month> <#>

Show email from <name>

76

Your mileage may vary, but this is a good set of open and show Siri commands to start with. A note about the syntax – slashes show choices, angle brackets are descriptions of what you can say.

Practice using these commands with this wording until it becomes habit, and you save steps throughout the day.

Here are some examples that follow these patterns:

### Slide 77

### Open and Show commands Examples

Open Safari

Open Settings

Show Calendar

Show Alarms

Show Email January 5

Show Email from Eric

77

## Slide 78

### Action Commands Syntax

|   |  |
|---|--|
| <p>Reminder &lt;anything&gt;<br/>Note &lt;anything&gt;<br/>Add to Note · &lt;anything&gt;<br/>Alarm &lt;time&gt; AM/PM<br/>Timer &lt;time&gt;<br/>Nearby &lt;anything&gt;<br/>Directions Home/&lt;anywhere&gt;<br/>Search Web/Wikipedia · &lt;anything&gt;</p> <p>Reminder/Calendar &lt;time&gt; AM/PM · &lt;anything&gt;<br/>Reminder/Calendar &lt;time&gt; AM/PM tomorrow · &lt;anything&gt;<br/>Reminder/Calendar &lt;time&gt; AM/PM next &lt;day&gt;/&lt;month&gt; &lt;#&gt; · &lt;anything&gt;</p> | <p>Call &lt;Name&gt; home/work/mobile<br/>Message/Email &lt;Name&gt;<br/>Message/Email &lt;Name&gt; · &lt;anything&gt;</p> |
|---|--|

78

And here's a good set of action commands to think about.

The dots mean you can stop in the middle so you can do this as one command or as two. This is useful for learning.

## Slide 79

### Action Command Examples

|   |  |
|---|--|
| <p>Note don't forget to talk about notes<br/>Add to Note talk about reminders too<br/>Reminder make sure to mention alarms<br/>Alarm 7 AM<br/>Timer 15 minutes<br/>Nearby Restaurants<br/>Directions Cambridge Brewing Company<br/>How do I get home<br/>Search Web tiger pictures<br/>Call Eric cell</p> | <p>Message Eric hi, how are you doing?<br/>Email Eric Gmail hi, how are you doing?<br/>Reminder 3 PM tomorrow feed cat<br/>Calendar 12:00 Monday lunch with Adam</p> |
|---|--|

79

Here are example commands written out follow that syntax. The simpler ones are on the left – they're the ones to start with. The more complicated ones are on the right – they are a little more fragile both in terms of how easy they are to say and how the computer might interpret them.

I tell people to practice using these commands with this wording until it becomes habit, starting with the ones on the right.

There are a couple of tricky things about messaging and email.

If you have existing message threads and you say a first name from one of them and there's no ambiguity, it will assume you're adding to an existing thread and it will say the last name of that person.

So since I already have a thread going with my husband Eric and I don't have threads going with other Eric's I can message Eric using just his first name:

For instance, **Message Eric hi, how are you doing?**

About email – if you have multiple addresses for a name Siri will prompt you to choose the right name. For someone you email a lot you can save a step by adding that address as a separate contact entry. I have Eric in as Eric Gmail.

**Email Eric Gmail hi, how are you doing?**

**Slide 80**

## Mobile

- Speech button on the keyboard
- What's Apple waiting for?
- Agent
- Agent strategies
- The command-and-control gap on mobile

80

We're up to the mobile gap – the big chunk that still missing on mobile.

**Slide 81**

## The Mobile Gap

The speech button on the keyboard and Siri enable you to do a lot of things with speech on mobile, but there's a big chunk missing. There's no command-and-control by speech.

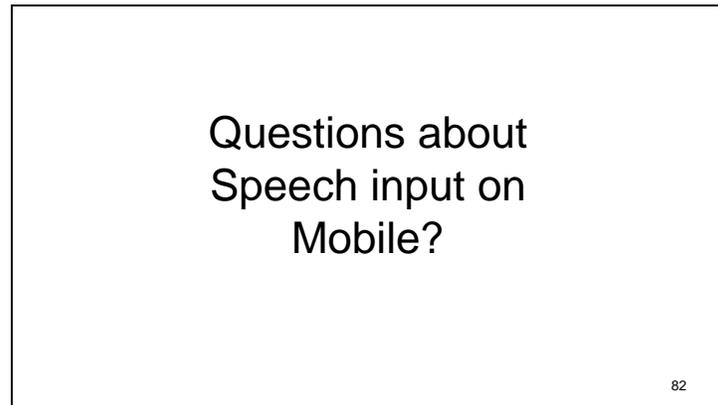
81

The speech button on the keyboard and Siri enable you to do a lot of things with speech on mobile, but there's a big chunk missing. There's no command-and-control by speech. You can't turn a page or hit a button once you've launched an app.

Command-and-control has historically been the tricky part on the desktop. I'm hoping command-and-control by speech input will eventually be implemented on smart phones at the

platform level, and that the developers will provide efficient defaults but also allow users to adjust, save and share commands so speech input can evolve. It's how language works.

## Slide 82



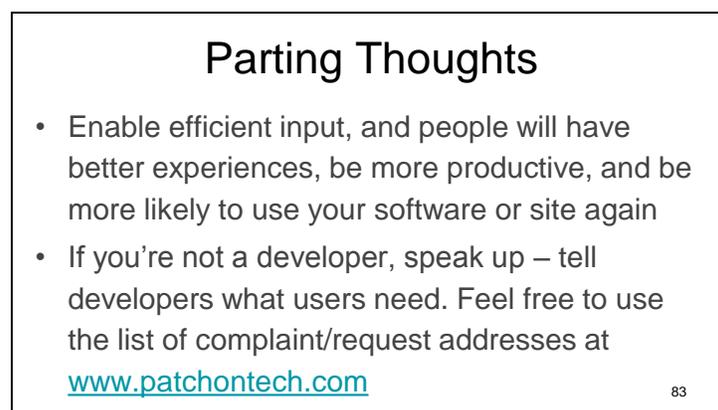
Questions?

I was expecting someone to ask about Dragon Anywhere. It's a recent Nuance product that allows you to use command-and-control but just in a Dragon writing app on the iPhone.

I don't use it for a couple of reasons. It only works in one app. It's like having a keyboard that only works in one program. I use Editorial and Scrivener for writing on my iPhone. I want to use speech in these programs. Second, it's an expensive iPhone app – \$15 a month, which works out to \$180 a year. That's a showstopper for many people.

Again, full speech input for smart phones needs to be implemented at the platform level – as part of the operating system – so there's a consistent interface across apps.

## Slide 83



Some parting thoughts:

Make input efficient and people will have a better experience.

They'll be more productive, and they'll be more likely to use the software again or come back to a website.

If you're not a developer, speak up – tell developers what users need.

Feel free to use the list of complaint/request addresses at [www.patchontech.com](http://www.patchontech.com)

This talk will also be posted there.

#### Slide 84

A rectangular slide with a black border containing text. The text is centered and includes a date, a title, and contact information.

Harvard University, August 14, 2017

## Accommodating Speech Input: Challenges and Best Practices

Kim Patch  
617-325-3966  
kim@scriven.com  
[patchontech.com](http://patchontech.com)  
[redstartsystems.com](http://redstartsystems.com)

84