

The Rich World of Mobile

Web, HTML apps, and native apps
and how we can i18nize them

The basic issues

Reading:

- Fonts and complex text rendering
- Localized string replacement

Searching, contributing:

- Text input

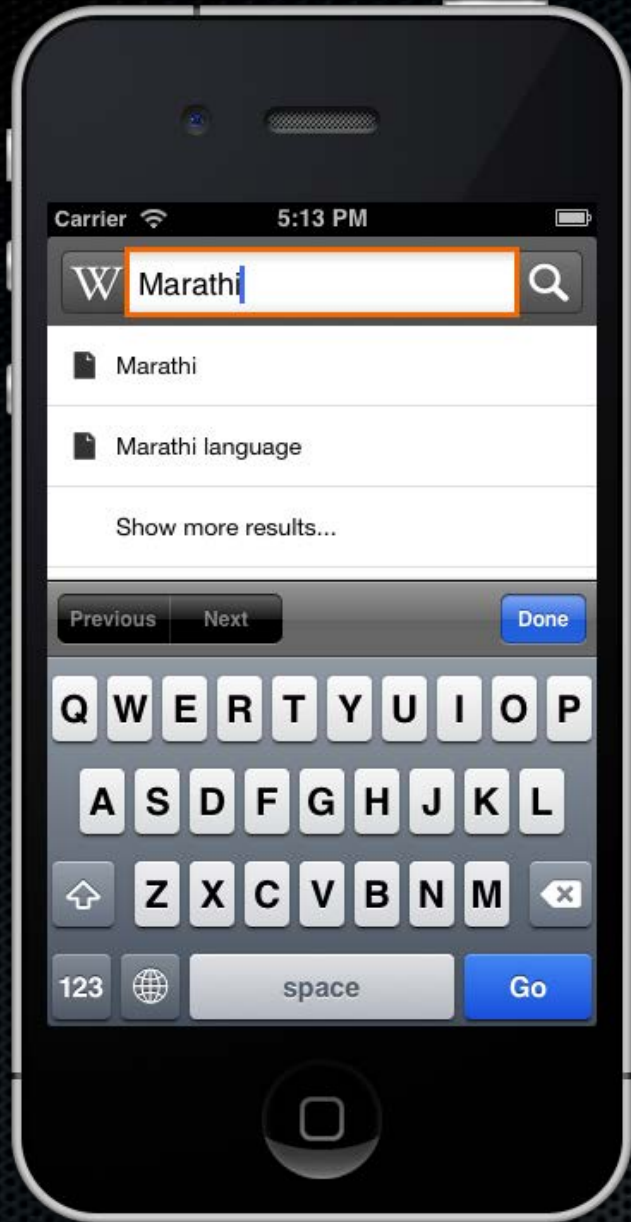
Mobile vs Desktop

IMEs and keyboards

Desktop keyboards are fixed but layout is unpredictable.

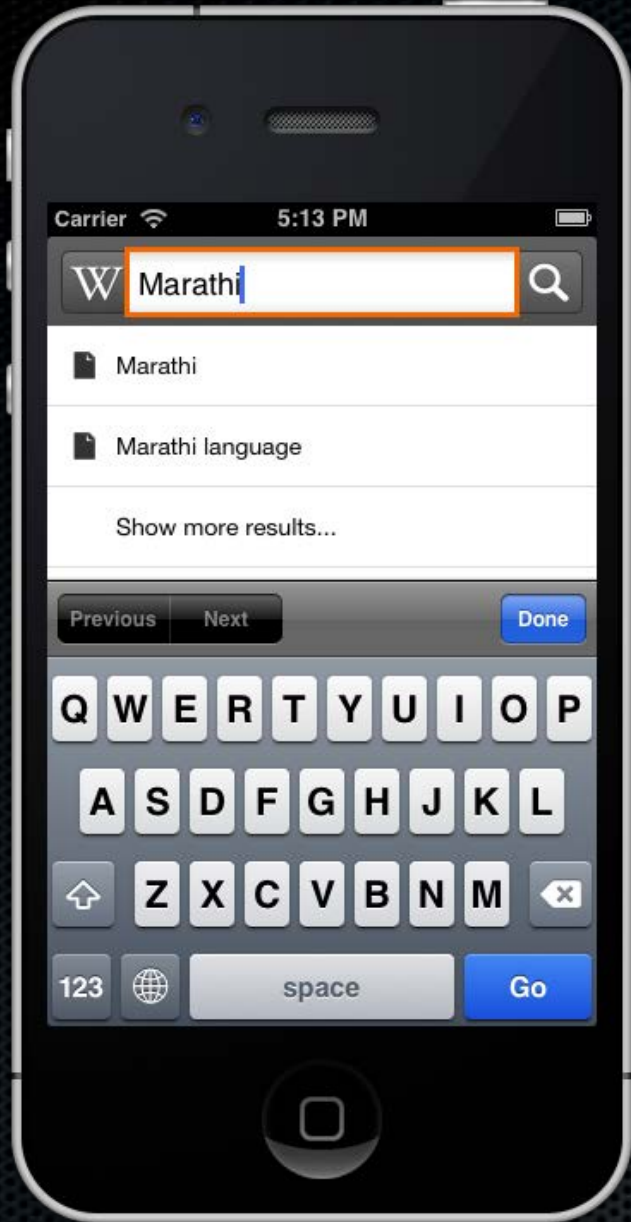
Wikimedia's work so far transliterates common Latin letters into a target language's script.





On mobile and tablet devices the on-screen keyboard can change to fit the language you're typing in.





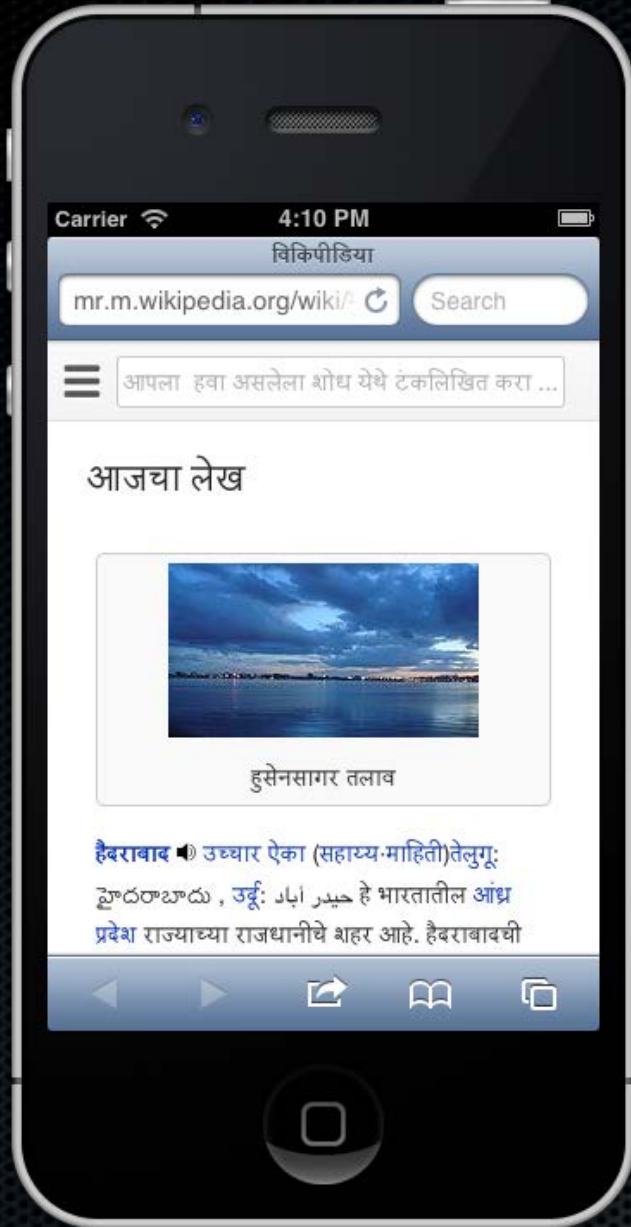
<- JavaScript can easily transliterate on that...

...but can HTML + JS make an on-screen kbd like that? ->



The mobile landscape

Mobile Web



- Always up to date
- Only online operation
- Limited system integration
- Has to work across large variety of OSes, browsers

Today:

- Supports whatever fonts, IMEs the host OS provides

Future:

- WebFonts can supplement default fonts
- JavaScript transliterating IMEs can supplement Latin input

Cordova + HTML5



- Mobile web tech wrapped in a local app...

Today:

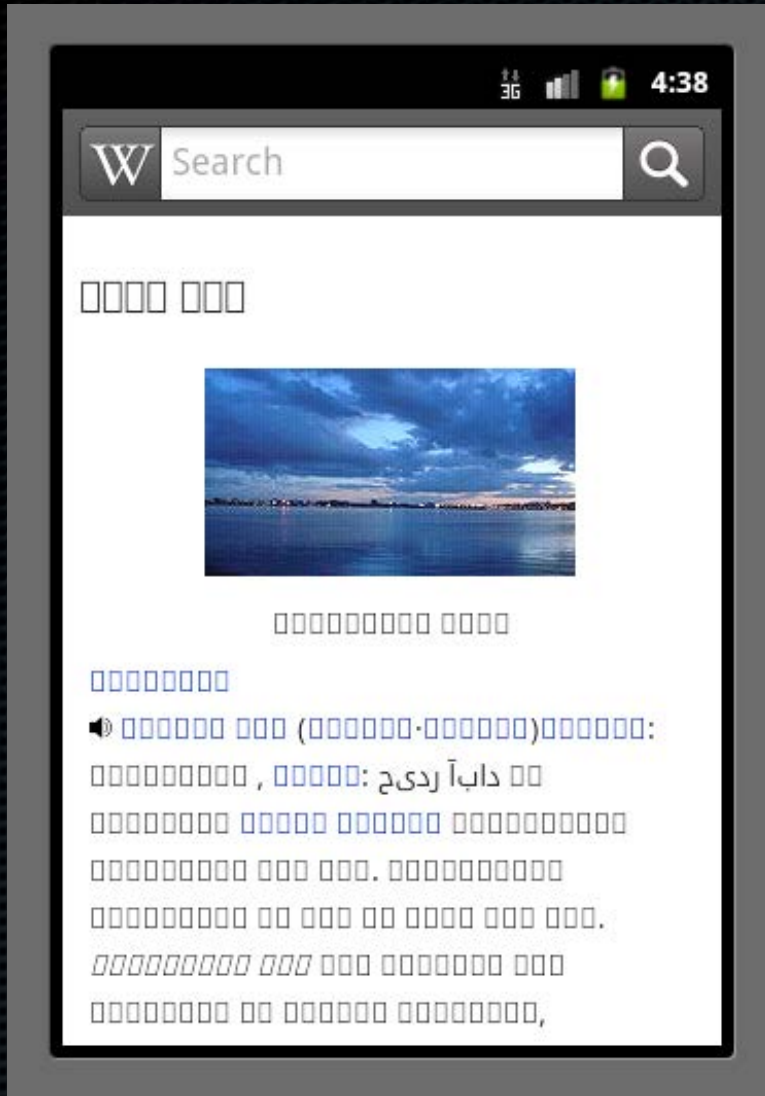
- support any fonts, IMEs available to the host OS

Future:

- WebFonts can supplement default fonts, limited by rendering engine
- JavaScript transliterating IMEs can supplement Latin input
- Could make use of native library plugins?

Cordova + HTML5

- Android 2.x is broken!
- WebFonts don't help
- Have to wait for it to go away for some languages :(





Firefox OS

- HTML5 app works without Cordova bridge

Today:

- appears to support any fonts, IMEs available to the host OS

Tomorrow:

- WebFonts should work
- JavaScript transliterating IMEs can supplement Latin input
- IMEs could be made more powerful with native system integration... and the system is HTML+JS. Possibilities?

Android Native

Today:

- Versioning disparities (fragmentation) in font support
- user can install custom native IMEs

Tomorrow:

- System fonts can be supplemented by the app
- Custom in-app IMEs would need to be ported to native Android from JavaScript
- Native Android IMEs would be a plus
- In-app libraries may be useful too



iOS Native



Today:

- limited Indic input support (Hindi only?)
- user can't install custom native IMEs

Tomorrow:

- System fonts can be supplemented by the app
- Custom in-app IMEs would need to be ported to native iOS from JavaScript
- In-app native libraries could be useful for other apps

Where we're going

First steps

- Use WebFonts in mobile web and HTML 5 apps to ensure font availability for compatible text rendering engines
- JavaScript IMEs transliterating Latin chars as "lingua franca" default: should work anywhere

Native apps evolution

- Truly native IMEs are possible as custom apps on Android. Yuvi's working on a port of jquery.ime rulesets on Latin keyboard.
- Custom IME keyboards for native iOS apps as a library might be useful. Can we help create an extensible one?
- What's the landscape with native on other OSs? BlackBerry, Windows Phone, Tizen, etc

HTML revolution?

- Are custom on-screen touch keyboards feasible in HTML web apps to replace/supplement default on-screen keyboards for web apps?
- Any synergy with web and Firefox OS?

end

<https://www.mediawiki.org/wiki/Mobile>