Web performance Brownbag.

So web performance is a massive subject I am going to run through some tools all of us can use, how our stuff get rendered on our screens and to look at some cool things that are coming our way to help with performance.

1. **Performance has 4 constrains**, I am going to delve into all these areas throughout this brown bag, but I have not defined these into section as there is lots of overlap.
   1. Network
   2. GPU
   3. CPU
   4. Screen
2. We know that - **Fast site + Great UX = Happy users**
3. **How do we make fast sites?** – Just send over HTML and enhance the site with JavaScript if the entire user interface is built with JavaScript users won’t see anything until everything is downloaded. If we just send HTML and enhance with JavaScript users get something they can start to use straight away.
4. **Things we can all use or do to help us understand how our sites perform.**
5. **Test on a slow connection –** You can use fiddler or load conditioners to simulate different real-world connection scenarios.
6. **PSI –** We have warnings for mobile and desktop we should fix these if we can.
7. **WPT** – Top right gives us feedback and you can click through for more info. It gives us waterfalls, videos and lots of data.
8. Akamai ran a survey some time ago how long do you think 40% of people would wait before leaving a site? 3 seconds!
9. **Desktop video –** Users start to see content at around 2.2 seconds so we are doing well.
10. **Let’s take a look at mobile 3g connection -** Users start to see content at around 8.2 seconds not so good.
11. **Speed Index –** Link off to google and read first paragraph.
12. **Different ways of measuring web performance –** First view, repeat view and flow view.
13. **Summary – Use tools to validate your work.**
14. **Critical rendering path - how does our stuff get to a browser? To start to optimise this we need to understand what happens. This is all taken from Google web Fundamentals which I recommend to everyone.**
    1. **Stuff browsers can animate quickly –** We want to avoid painting and layout as these are expensive – Styles, layout and Paint are on the CPU.
    2. **Layer creation criteria –** examples dev tools, Always promote any animating element to its own layer.
15. **Emulation Chrome dev tools**
    1. **Emulate retina display**
    2. **Enable layer creations**
    3. **Pinch to zoom**
    4. Always test on the correct devices.
16. **Some things coming up that are exciting**
    1. **Picture element**
    2. **Will-change property**