Challenges with SPDY deployment

Rajeev Bector
Yahoo!
Facts!
Fact

- Very heterogeneous Serving Environment
- Fragmented Serving Architecture
- Very little SSL
Fact

- We have to live with **Inter-op** between HTTP/1.0 and SPDY for a LONG time to come.
Fact

Many sites will continue to differentiate between "base html" and CDN content

- Operational Reasons
- Administrative Reasons
- Technical Reasons
SSL

- Ads from 3rd-party ad-networks that do not support SSL.
- Sites designed without SSL in mind
- Usual co-location/virtual-hosting challenges with SSL
- Windows XP not supporting things like SNI (unfortunately, XP still makes up for a chunk of traffic).
SSL (2)

- SAN can help but number of domains is staggering
- Wildcards don't really work when you have 4-level nesting
- Pages have lots and lots of domains that they serve data over.
Goal

Can we deploy & benefit from SPDY without SSL Everywhere
Yes, we can!

- Many Pages have large amount of CDN content (Static Objects, JS, CSS, Images)
- It sounds like a good place to optimize for Header and Concurrency related inefficiencies in HTTP/1.1 (which SPDY solves)
Solution

- Serve Base Page over HTTP
- Serve CDN content over SPDY
Except
- HTML optimized for SPDY has no sharding.

- HTML optimized for SPDY has "https://" links to CDN

- It's also operationally easier to serve it using newer/separate end-points.
Solution (sorta ...)  
Sniff the UA ...  
Except that ...  

UA sniffing seems is getting more complicated every day ...
Another problem ...

Downstream Caches should not serve SPDY html to non-SPDY browsers.

Vary header ... ummmm.
Mixed-Mode Serving: Can we do better for using SPDY (and HTTP/2.0 along with HTTP/1.1)?