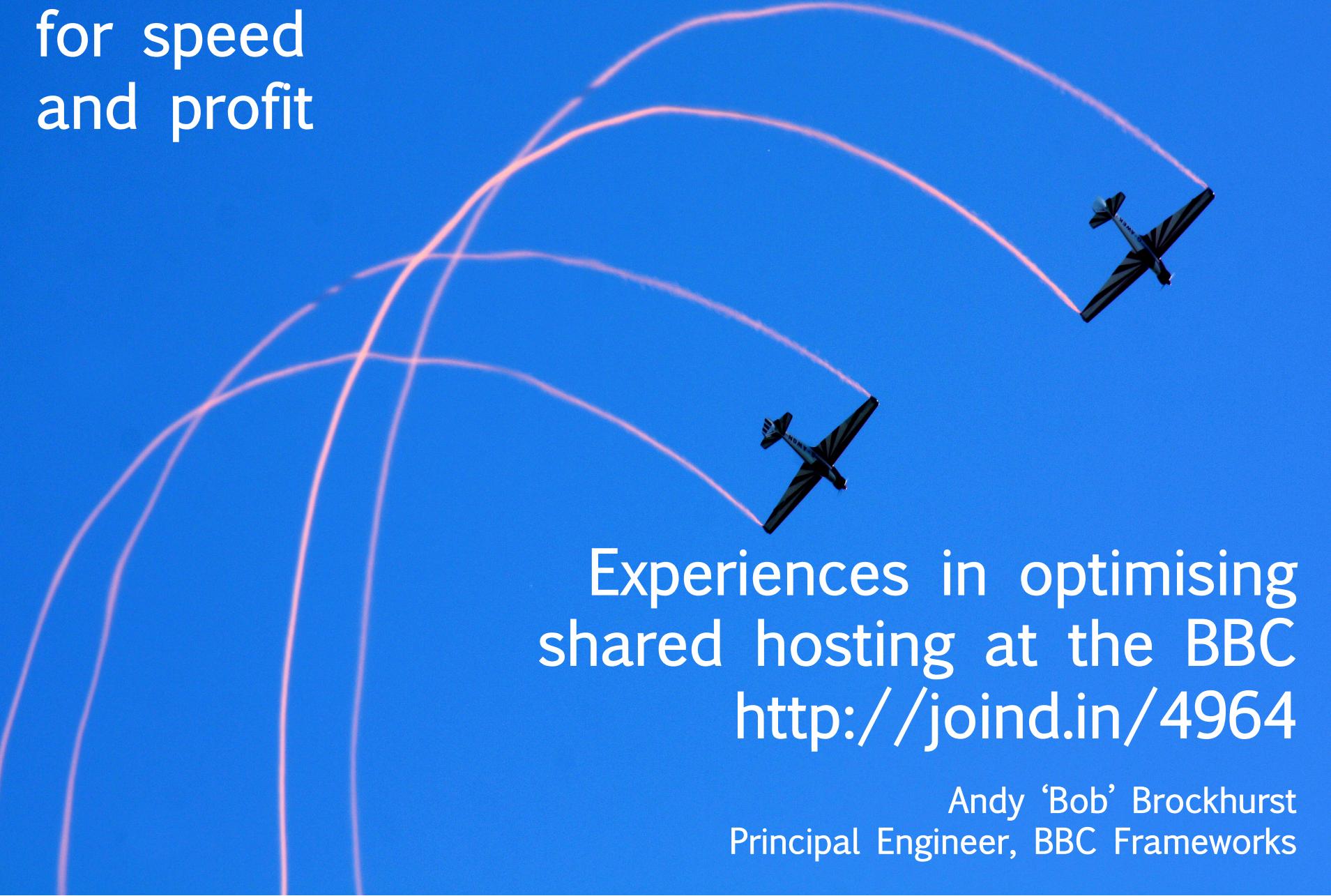
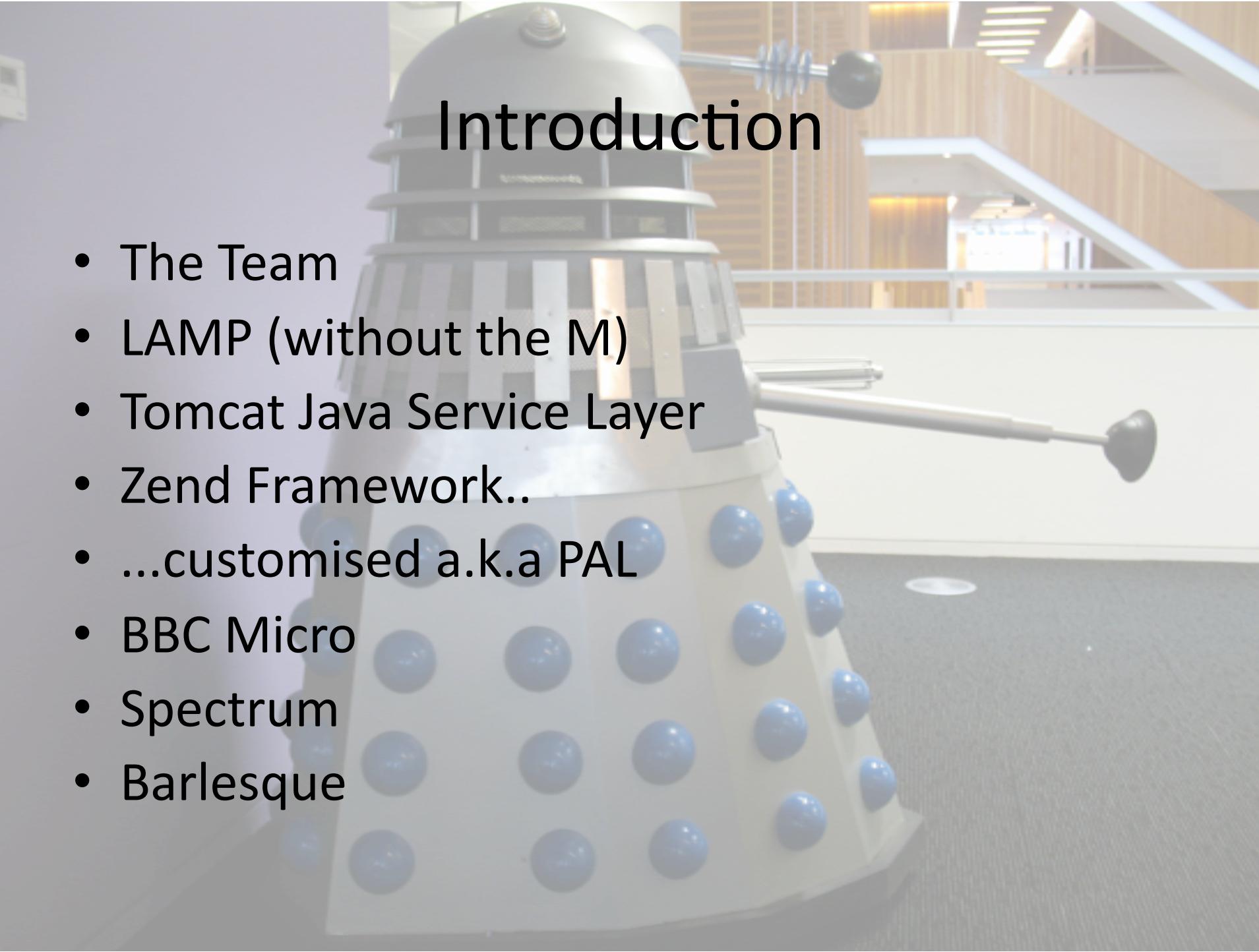


Monitoring your back-end for speed and profit



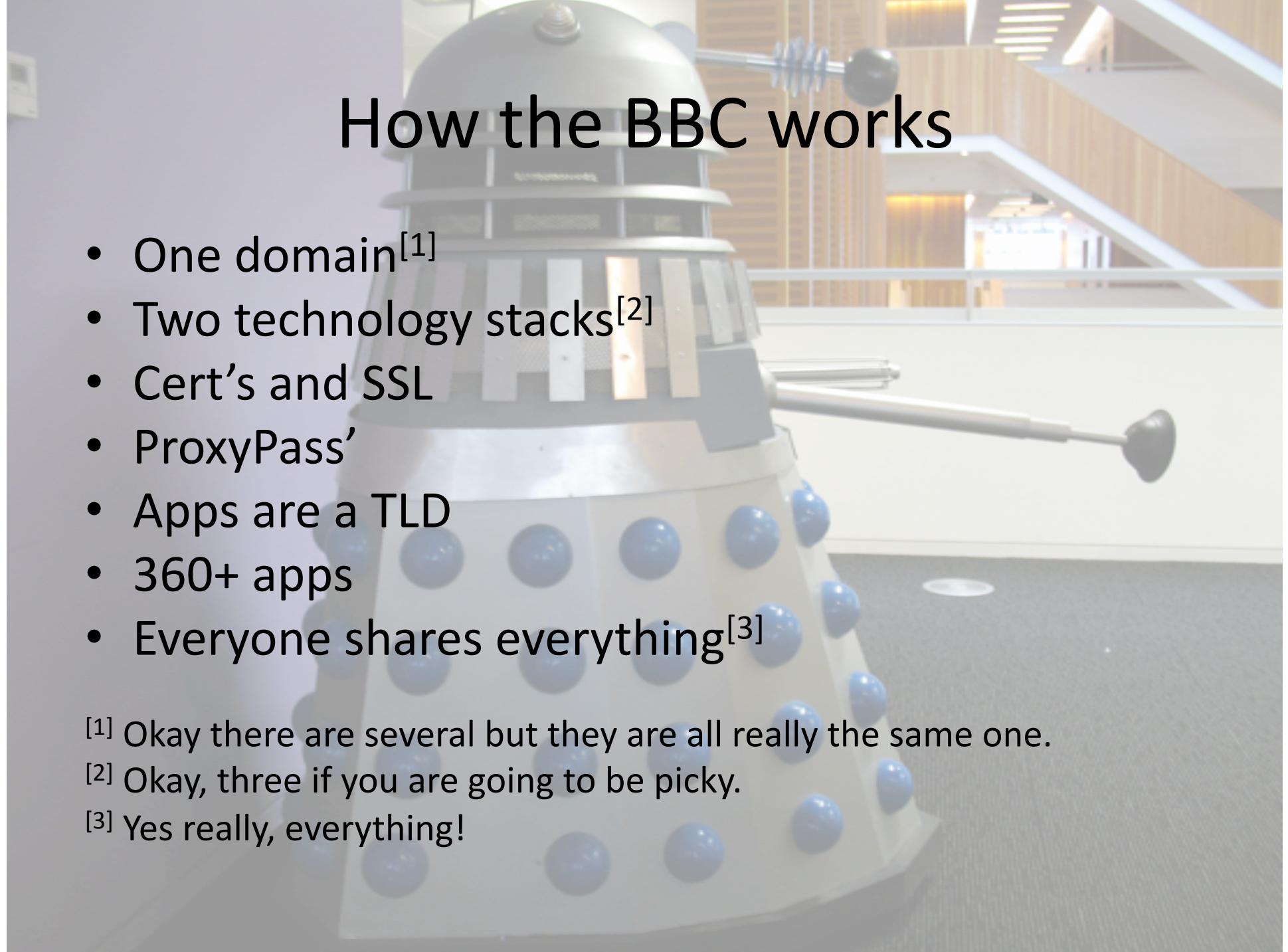
Experiences in optimising
shared hosting at the BBC
<http://joind.in/4964>

Andy 'Bob' Brockhurst
Principal Engineer, BBC Frameworks

A large, white Dalek model from Doctor Who is positioned in the foreground, facing right. It has its characteristic blue glowing eye and a long black metal probe extending from its left side. The background shows a modern office environment with wooden paneling, a staircase, and recessed lighting in the ceiling.

Introduction

- The Team
- LAMP (without the M)
- Tomcat Java Service Layer
- Zend Framework..
- ...customised a.k.a PAL
- BBC Micro
- Spectrum
- Barlesque



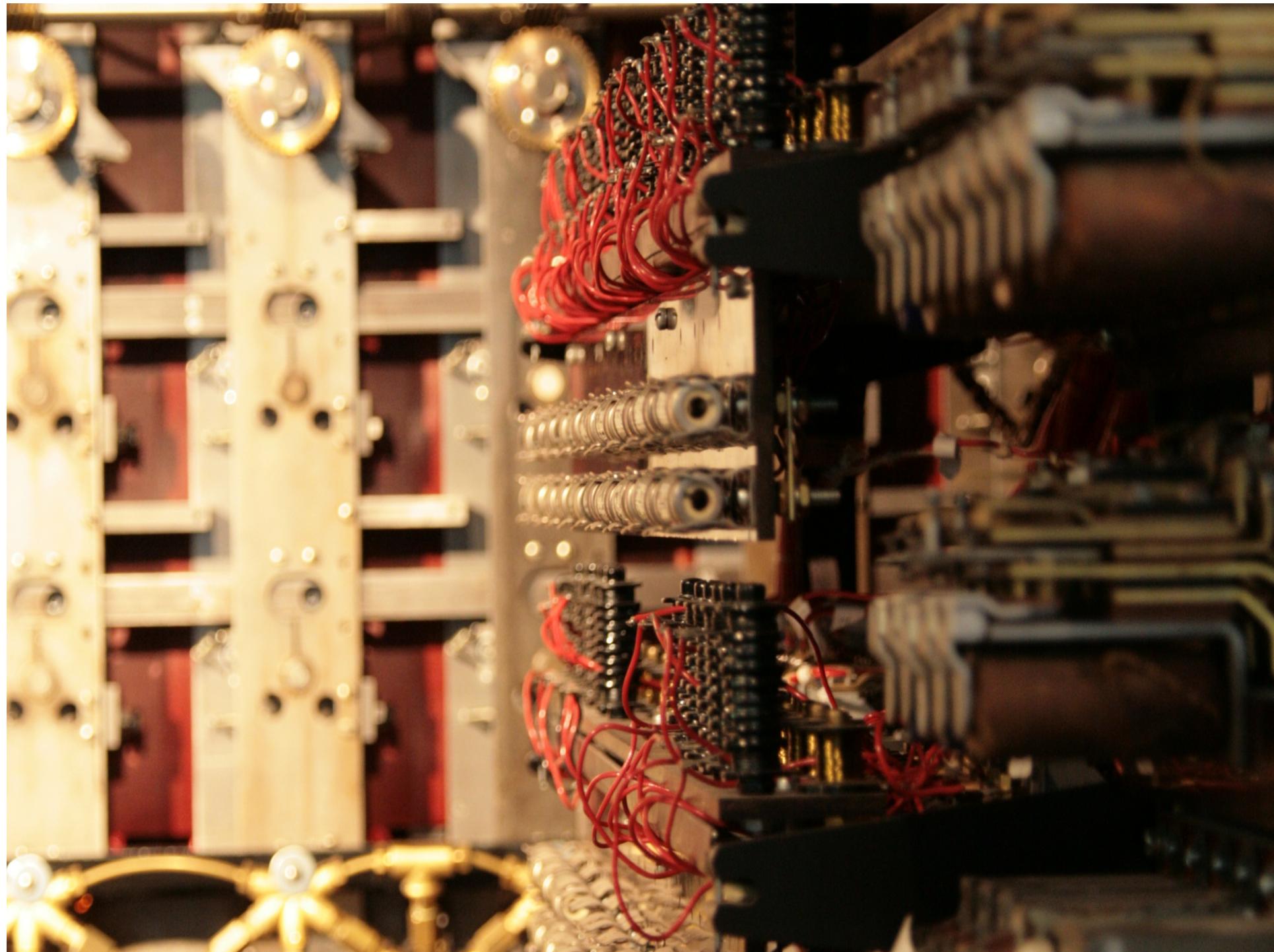
How the BBC works

- One domain^[1]
- Two technology stacks^[2]
- Cert's and SSL
- ProxyPass'
- Apps are a TLD
- 360+ apps
- Everyone shares everything^[3]

^[1] Okay there are several but they are all really the same one.

^[2] Okay, three if you are going to be picky.

^[3] Yes really, everything!



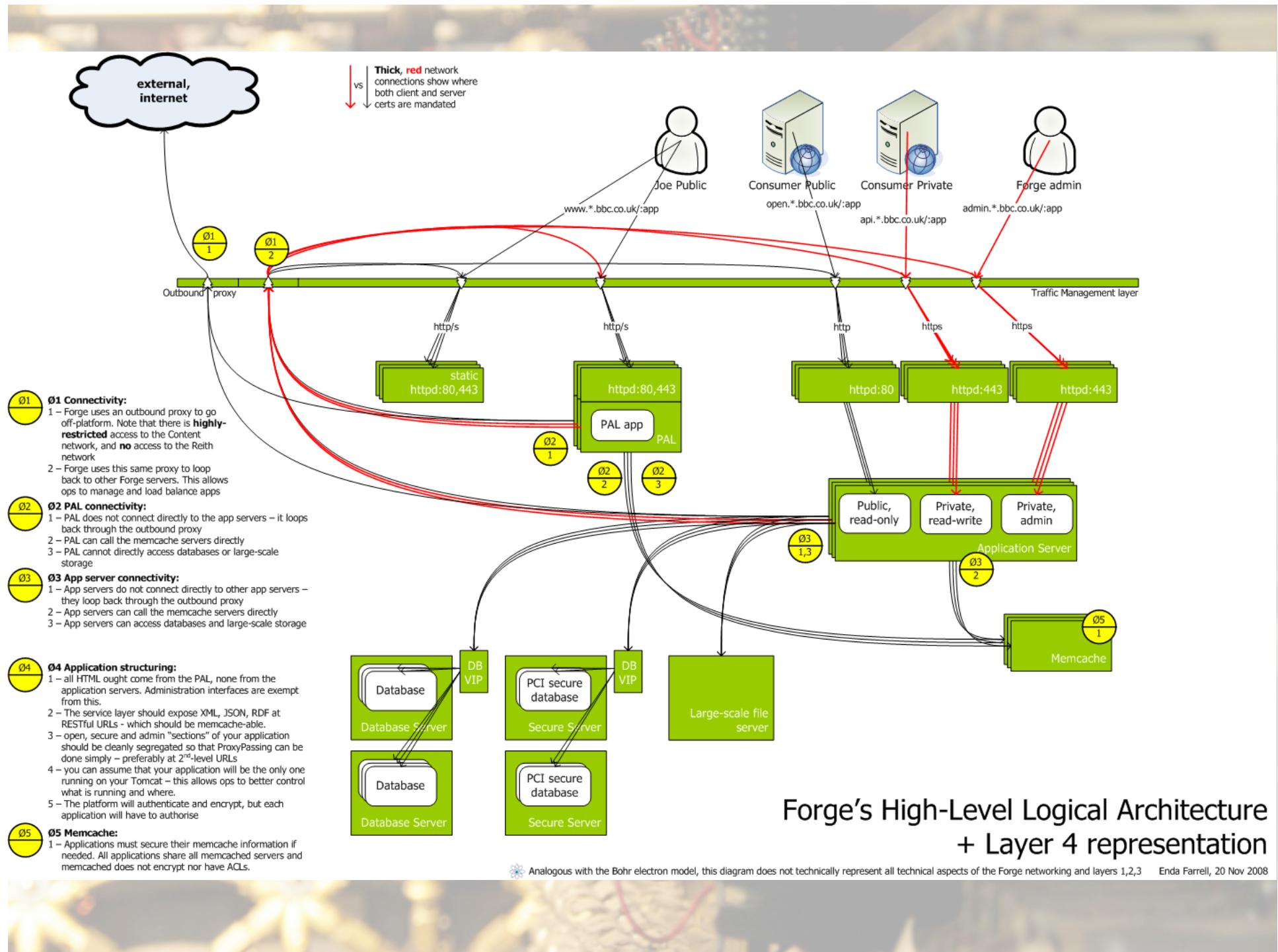
Network Topology

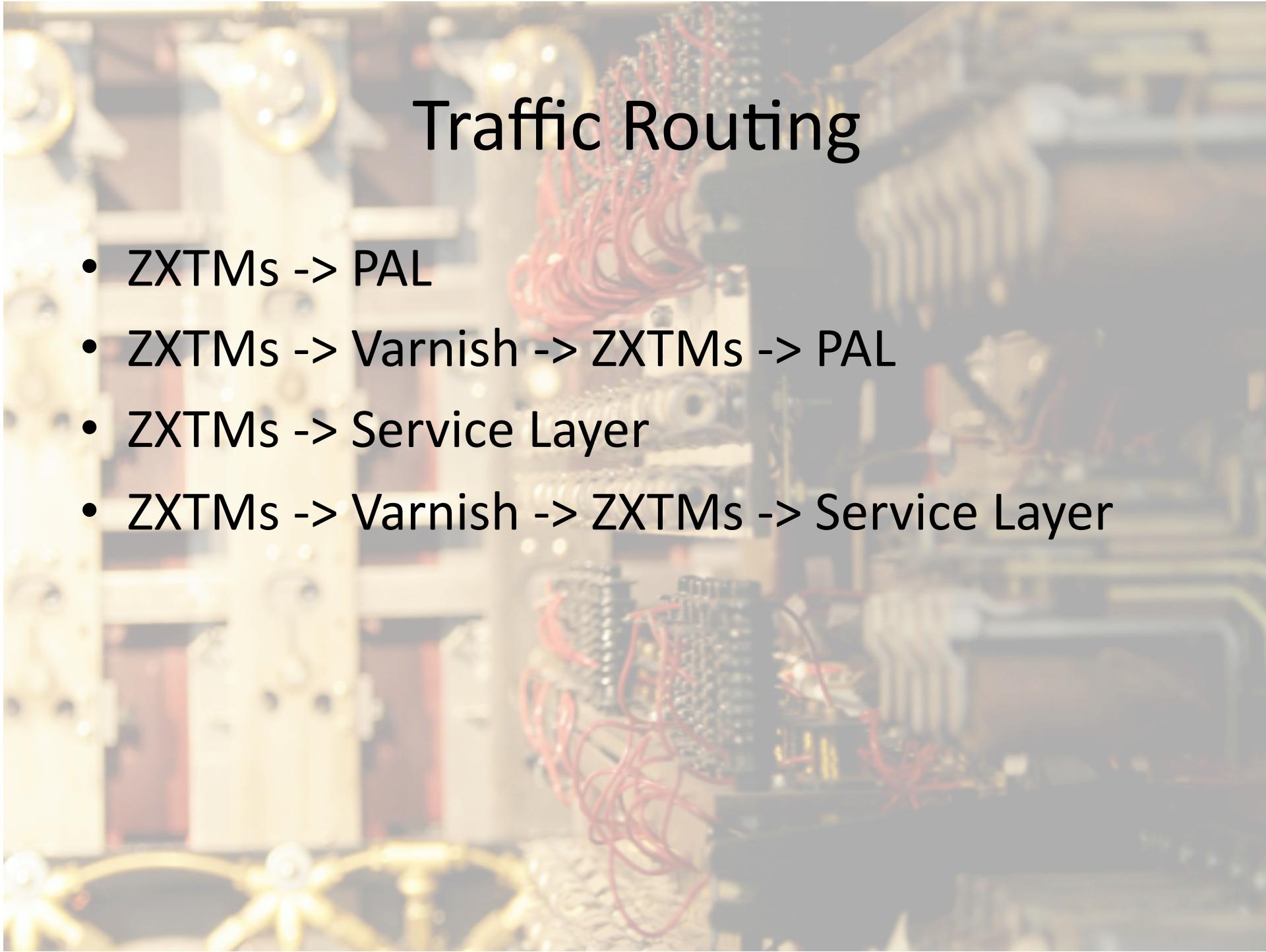
- Dual DC^[1]
- No DC affinity^[2]

[1] One more soon(ish)

[2] Well a couple of apps do^[3]

[3] We don't talk about them

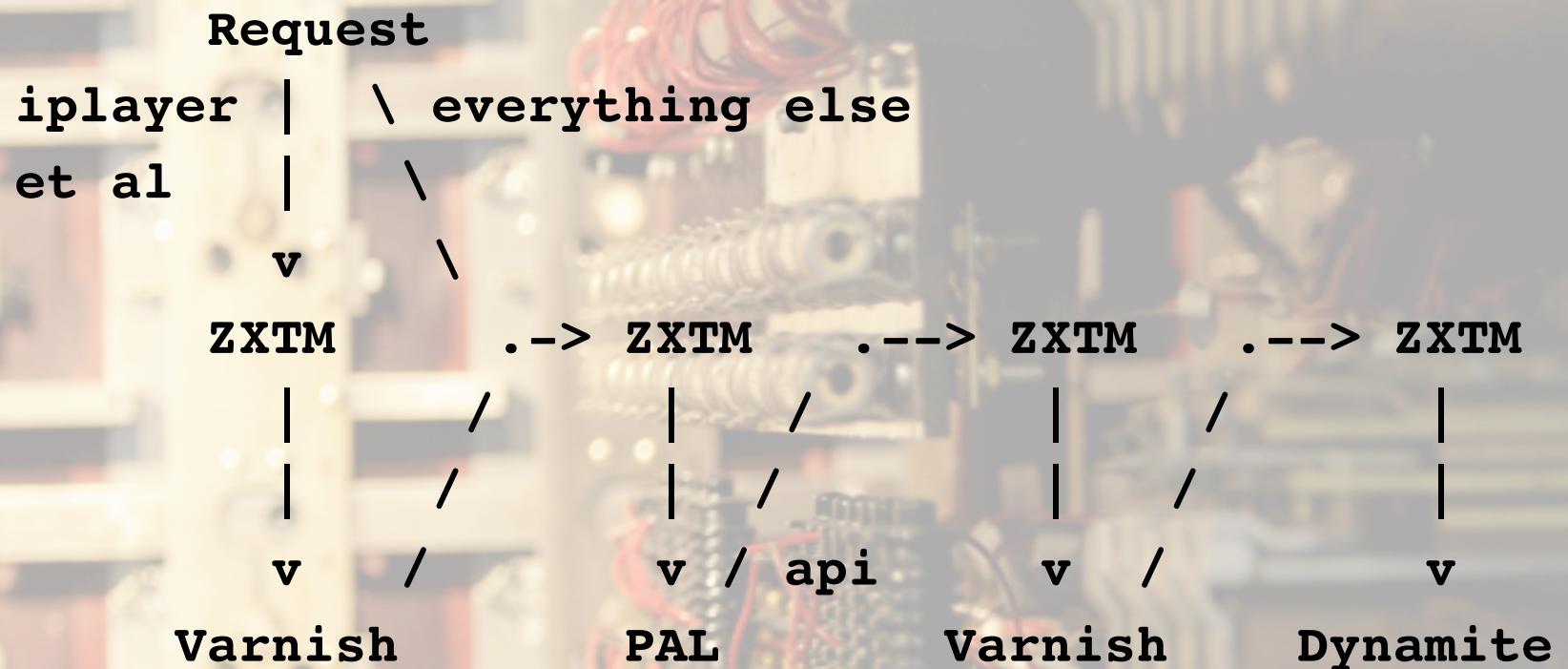




Traffic Routing

- ZXTMs -> PAL
- ZXTMs -> Varnish -> ZXTMs -> PAL
- ZXTMs -> Service Layer
- ZXTMs -> Varnish -> ZXTMs -> Service Layer

Traffic Routing

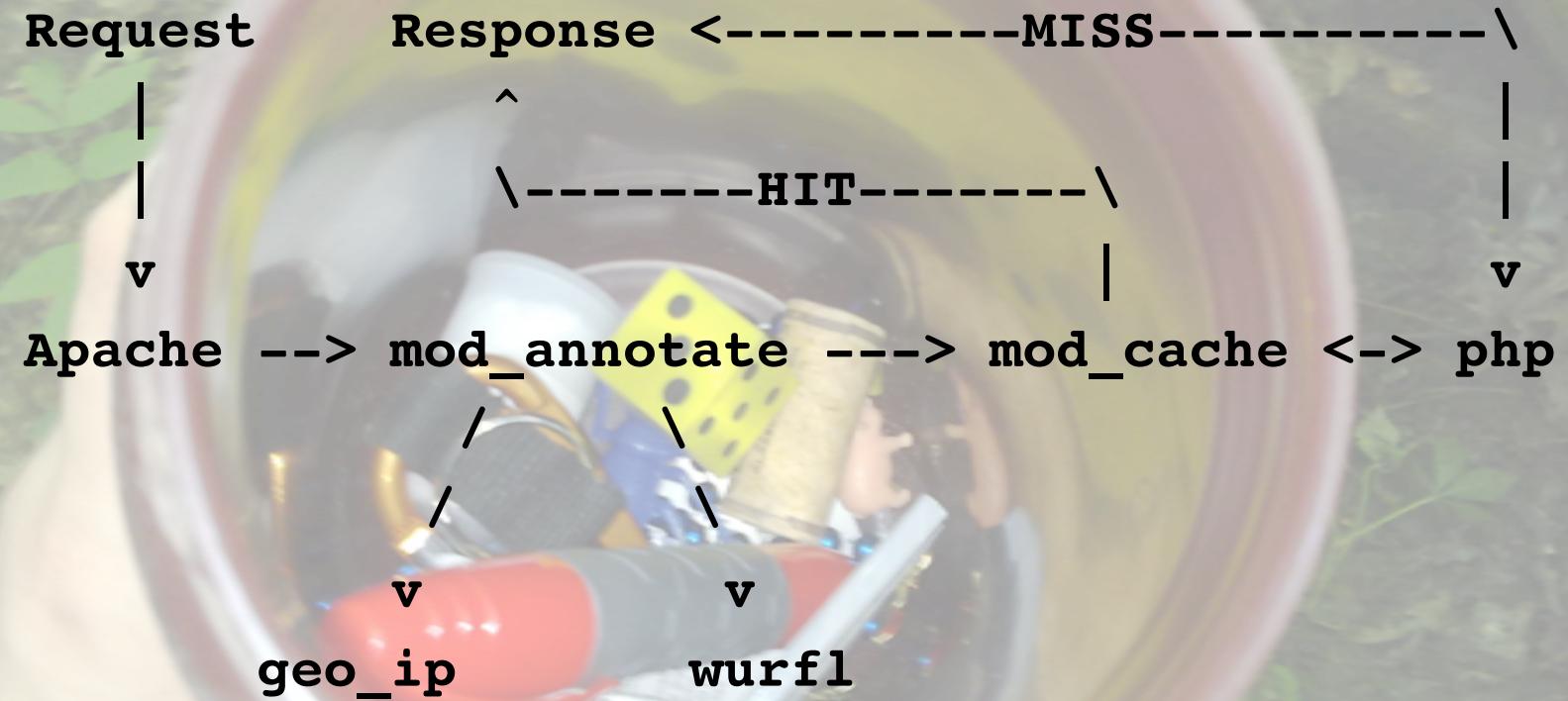




Caching, caching and more caching

- mod_cache
- Barlesque
- Advertising
- mod_annotate
- Varnish
- Cookies

mod_cache & mod_annotate



A photograph of a vibrant, multi-colored wheelbarrow (red, yellow, green) filled with various items like a yellow bucket and a red watering can, set against a backdrop of lush green foliage.

Varnish

- Custom VCL
- Extended with C to provide annotations
- Normalise headers
- Strip Cookies
- Set Cookies
- Cache busters
- ESI

Annotations to Improve hit/miss ratios

- Geo IP data
- Device Data - WURFL

Annotations – Normal Client/Server

```
GET /iplayer/ HTTP/1.1
Host: www.bbc.co.uk
User-Agent: Blah blah blah chuffing fish turnip
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-gb,en;q=0.5
```

```
HTTP/1.1 200 OK
Content-Length: 8392
Content-Type: text/html
Cache-Control: max-age=30
Date: Sat, 27 Feb 2012 12:25:00 GMT
Expires: Sat, 27 Feb 2012 12:25:30 GMT
```

Annotations – Varnish/Server

```
GET /iplayer/ HTTP/1.1
Host: www.bbc.co.uk
User-Agent: Blah blah blah chuffing fish turnip
X-DEMI-Device: desktop
X-Country: uk
X-Ip_is_uk_combined: yes
X-Ip_is_advertise_combined: no
...

```

```
HTTP/1.1 200 OK
Cache-Control: max-age=30
Date: Sat, 27 Feb 2012 12:25:00 GMT
Expires: Sat, 27 Feb 2012 12:25:30 GMT
Vary: X-DEMI-Device, X-Ip_is_uk_combined
...

```

WHAT ARE
YOU
LOOKING AT?



Pre-existing Monitoring

- "SNMP" KPIs to Zenoss
- Apache error logs

WHAT ARE
YOU
LOOKING AT?

Monitoring Shortfalls

- Granularity
 - Per app data
 - In app data
 - Per service data
- Frequency
 - /me shrugs

Monitoring Solutions

- Apache logs
- Prototypes

WHAT ARE
YOU
LOOKING AT?





Per Applications Stats for Varnish

- Add header response from WebApp
- Inline C in VCL
- Statsd implementation
 - Hit/Miss/Pass ratios
 - Inter-quartile ranges
 - Updated every 10s

Varnish VCL – Setup

```
C{
#include <sys/socket.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <sys/un.h>
#include <unistd.h>

#define STATSD_SOCKET "/var/run/statsd/statsd.sock"
#define STATSD_UNCONNECTED -1
#define STATSD_CONNECTING -2
static int statsd_fd_permanent = -1;
static int statsd_fd = STATSD_UNCONNECTED;

static int statsd_get_fd() {
    # see next slides
}

static int statsd_increase_cache_counter(...) {
    # see next slides
}

static void statsd_close_fd(int fd) {
    (void) __sync_val_compare_and_swap(&statsd_fd, fd, STATSD_UNCONNECTED);
}
}C
```

Varnish VCL – Open Socket

```
static int statsd_get_fd() {
    int fd, i;
    struct sockaddr_un addr;
    fd = statsd_fd;
    if (fd >= 0) { return fd; }
    i = __sync_val_compare_and_swap(&statsd_fd, STATSD_UNCONNECTED, STATSD_CONNECTING);
    if (i != STATSD_UNCONNECTED) { return i; }
    addr.sun_family = AF_UNIX;
    strcpy(addr.sun_path, STATSD_SOCKET);
    fd = socket(AF_UNIX, SOCK_DGRAM, 0);
    i = connect(fd, (struct sockaddr *) &addr, sizeof(addr));
    if (i == -1) {
        close(fd);
        statsd_fd = STATSD_UNCONNECTED;
        return -1;
    }
    if (statsd_fd_permanent == -1) {
        statsd_fd_permanent = fd;
    } else {
        dup2(fd, statsd_fd_permanent);
        close(fd);
        fd = statsd_fd_permanent;
    }
    statsd_fd = fd;
    return fd;
}
```

Varnish VCL – Increment Counter

```
static int
statsd_increase_cache_counter(const char *name, const char *action, const char *reason)
{
    int fd, length, i;
    char buffer[128];

    if (name == NULL) { return 0; }

    length = snprintf(
        buffer + 1, sizeof(buffer) - 1,
        "varnishstats:%s_%s[%s]", name, action, reason
    );
    if (length < 0 || length >= sizeof(buffer) - 1) { return 0; }

    buffer[0] = 1;

    fd = statsd_get_fd();
    if (fd < 0) { return 0; }

    i = send(fd, buffer, 1 + length, 0);
    if (i == -1) {
        statsd_close_fd(fd);
        return 0;
    } else {
        return 1;
    }
}
```

Varnish VCL – Calling statsd

```
sub vcl_deliver {
    if (resp.http.X-Varnish-Cache-Pass) {
        C{
            const char *webapp = VRT_GetHdr(sp, HDR_RESP, "\011X-Webapp:");
            const char *reason = VRT_GetHdr(sp, HDR_RESP, "\025X-Varnish-Cache-Pass:");
            statsd_increase_cache_counter("varnish", "cache", "pass");
            statsd_increase_cache_counter("varnish", "pass", reason);
            statsd_increase_cache_counter(webapp, "cache", "pass");
            statsd_increase_cache_counter(webapp, "pass", reason);
        }C
    } else {
        if (obj.hits > 0) {
            /*
             * This hit is monitored in vcl_hit because the X-Webapp header is
             * lost when a 304 is generated.
             */
        } else {
            C{
                const char *webapp = VRT_GetHdr(sp, HDR_RESP, "\011X-Webapp:");
                statsd_increase_cache_counter("varnish", "cache", "miss");
                statsd_increase_cache_counter(webapp, "cache", "miss");
            }C
        }
    }
}
```

Varnish VCL – Calling statsd

```
sub vcl_hit {
    if (obj.cacheable) {
        C{
            const char *webapp = VRT_GetHdr(sp, HDR_OBJ, "\011X-Webapp:");
            // e.g. varnish_cache_hit
            statsd_increase_cache_counter("varnish", "cache", "hit");
            // e.g. iplayer_cache_hit
            statsd_increase_cache_counter(webapp, "cache", "hit");
        }C
    }
}
```



Application Performance

- Apache logs
- XHProf
- Sample 1 in x requests
- Modified XHProf console
- teleportd transport
- Data framing

Data transport

- Generate profile_id header
- teleportd
- gzipped data
- Add data framing
 - Maximum socket length
 - Multiplexed streams

Data framing

- Send profile_id plus length of profile data
- Use length as framing key
- Insert key on newlines
- Write as much as we can
- Keep writing
- Throw away on error

Data framing - PHP

```
try {
    $fp = @fopen($pipe_path, "r+");
} catch (Exception $e) {
    error_log("Exception thrown opening pipe $pipe_path:
        $e->getMessage()");
}

if ($fp !== false) {
    stream_set_blocking($fp, 0);

    if(flock($fp, LOCK_EX | LOCK_NB)){
        $output = gzencode(json_encode($data));
        $size = strlen($output);
        $data_to_write = "s" . $size . "\n" . "d" . $size . "|" .
            str_replace("\n", "\nd" . $size . "|", $output) . "\n";
        if(fwrite_stream($fp, $data_to_write) === false) {
            error_log("Error occurred while writing to $pipe_path.
                fwrite returned false");
        }
        flock($fp, LOCK_UN);
    }
    fclose($fp);
} else {
    error_log("Unable to open pipe $pipe_path for xhprof data");
}
```



HAR

- HTTP Archive format
- teleportd transport
- Post data into KV
- Possibly CouchDB views

HAR – Http ARchive



The background of the slide features a scenic view of Niagara Falls, with water cascading over the falls and mist rising from the base.

Home Preview HAR About 2.0.13 Schema

Show Page Timeline | Show Statistics | Clear | 

/sport/shared/football/accordion/partial/competiti...clientbbcfmtj&cachebuster=cb132378286887553968

- + MEMCACHED:INCR spc 200 OK incr 53 B 0ms
- + GET matches?startdate=2013-08-01 Memcached api.int.bbc.co.uk 504.6 KB 6ms
- + MEMCACHED:LOAD spc 200 OK load 505.2 KB 5ms

3 Requests 1009.9 KB (504.6 KB From Cache) 68ms (onload: 279ms)

HAR Example

Home | Preview | **HAR** About 2.0.13 | Schema

Show Page Timeline | Show Statistics | Clear |

▀ /sport/shared/football/accordion/partial/competiti...clientbbcfmtj&cachebuster=cb132378286887553968

▀ MEMCACHED:INCR sport__pal_BBC_Monitoring_Counte 200 OK incr 53 B

Headers Response

Response Headers

Key sport__pal_BBC_Monitoring_Counter_ApplicationRequest

Request Headers

Key sport__pal_BBC_Monitoring_Counter_ApplicationRequest

▀ https://api.int.bbc.co.uk:443/sportsdata/statsapi/football/matches?startdate=20110913&enddate=20120313&competition=118996116 1 KB

Headers Response Cache

Response Headers

Cache-control no-transform,max-age=60,stale-if-error=3600,stale-while-revalidate=60
Connection close
Content-type application/json
Date Tue, 13 Dec 2011 13:27:09 GMT
Etag "ccedbde796e1b954b371b2e4c39948e66f2830e"
Keep-alive timeout=45
Server Apache
Vary Accept
X-cache-detail "cache miss: attempting entity save" from api.back.int.local

▀ MEMCACHED:LOAD sport__pal_BBC_Http_Client_Reque 200 OK load 505.2 KB

Headers Response

Response Headers

Key sport__pal_BBC_Http_Client_RequestCached_a2ecc2c25ae7bf74589b27d9768b9ff571f7f10b

Request Headers

Key sport__pal_BBC_Http_Client_RequestCached_a2ecc2c25ae7bf74589b27d9768b9ff571f7f10b

3 Requests 1009.9 KB (504.6 KB From Cache)

Extending Zend_Http_Client

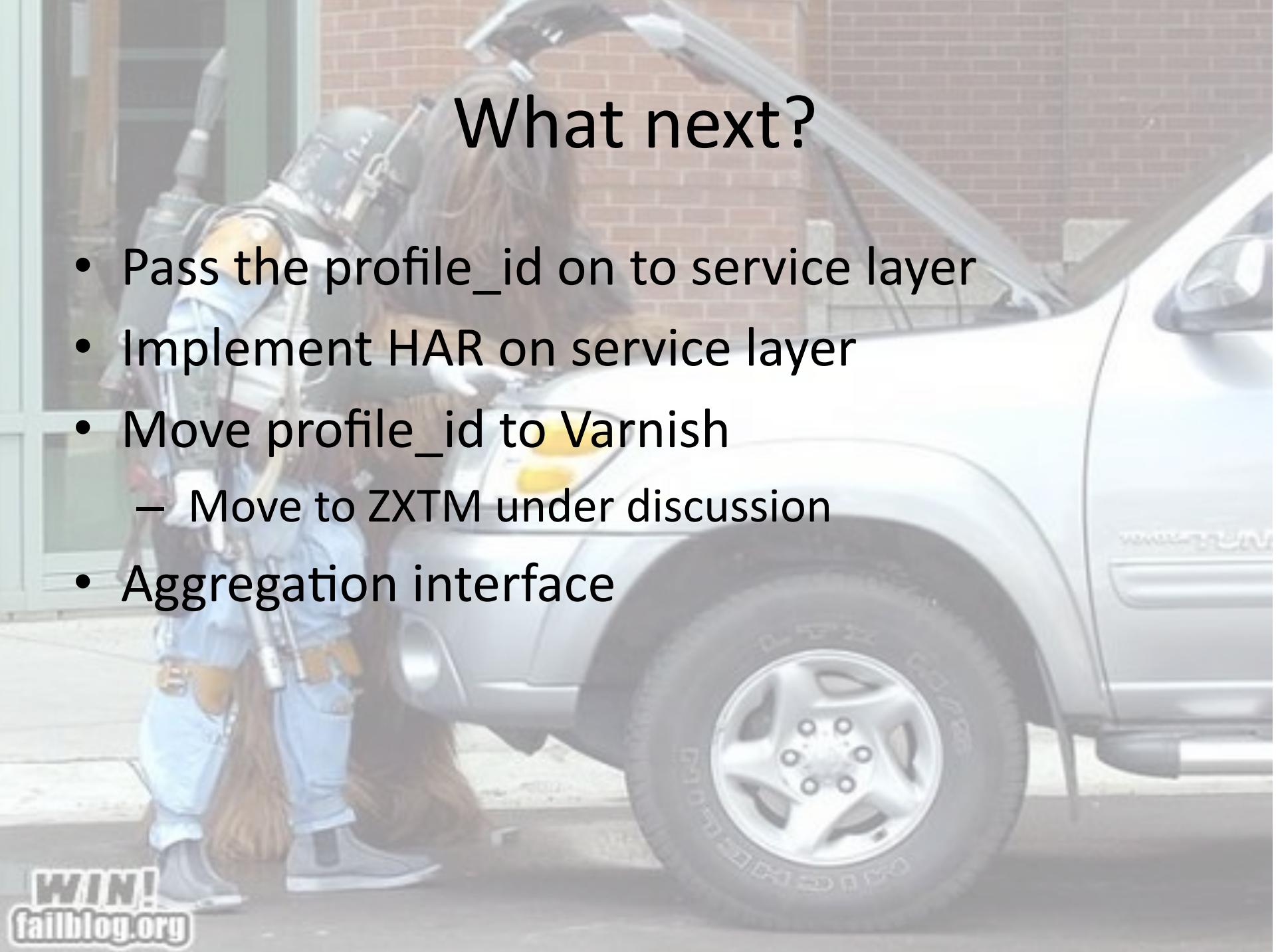
- Already extended as BBC_Http_Client
 - Handles Client Cert protection
 - Uses memcache
- Added performance information
 - Use curl handle info where possible

Extending Zend_Cache_Core

- Already extended as BBC_Cache_Core
 - Handles consistent hashing algorithm
 - Prevents key conflicts
- Added performance information
 - Use microtime
 - Record record sizes etc.
- Can also record APC calls
 - Not recommended – too many calls



WIN!
failblog.org

A photograph of a person wearing a cowboy hat, blue jeans, and a patterned shirt, standing next to the front left side of a silver SUV. The background shows a brick building.

What next?

- Pass the profile_id on to service layer
- Implement HAR on service layer
- Move profile_id to Varnish
 - Move to ZXTM under discussion
- Aggregation interface



TO MAKE A LONG STORY
SHORT

Thanks for listening

- Thanks to flickr users:
 - victimus
 - Sportavia Fournier RF4-Redhawks Display Team-Duxford Oct 2010
 - <http://www.flickr.com/photos/victius/5184316825>
 - dgjones
 - Office Dalek, London, 14-10-06
 - <http://www.flickr.com/photos/dgjones/284592369>
 - b3cft
 - Bombe rebuild detail
 - <http://www.flickr.com/photos/b3cft/3797123899>
 - eszter
 - Cache contents
 - <http://www.flickr.com/photos/eszter/3851576235>
 - Enjoy Surveillance
 - What are you looking at?
 - <http://www.flickr.com/photos/enjoy-surveillance/34795807/>
 - TaranRampersad
 - Varnish (1)
 - <http://www.flickr.com/photos/knowprose/4862298148/>
 - Solo
 - 45th Annual Watsonville Fly-in and Air Show
 - <http://www.flickr.com/photos/donsolo/4959045491/in/photostream/>
 - SF Brit
 - Sunset over Iguaçu
 - <http://www.flickr.com/photos/cnbattson/4333692253/>

TO MAKE A LONG STORY
SHORT

Thanks for listening

- This presentation:
 - github.com/b3cft/PHP-London-2012-Talk
- Me:
 - Andy “Bob” Brockhurst
 - Twitter: [@b3cft](https://twitter.com/b3cft) (and pretty much anywhere online)
 - www.kingkludge.net
 - <http://joind.in/4964>

