

针对HTML5应用的 Web Runtime 优化实践

丁俊勇
英特尔软件和服务事业部
资深软件架构师
jonathan.ding@intel.com



免责声明

- 本文件中所包含之信息仅用于本次讲演之目的，并不构成对英特尔产品（及相关服务）性能、质量的全
面、准确的表述、担保及保证
- 本文件中所包含信息的著作权归属于英特尔公司。除
非得到特别授权，任何人不得随意复制、散发、或未
经授权将该文件用于其他商业或非商业目的
- **Intel**和**Intel logo**是英特尔在美国和/或其他国家的
商标
- 本文件中所涉及的非英特尔商标为其他公司之财产
- **Copyright © 2013 Intel Corporation. All rights
reserved**

议程

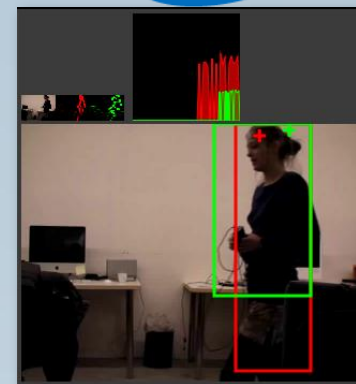
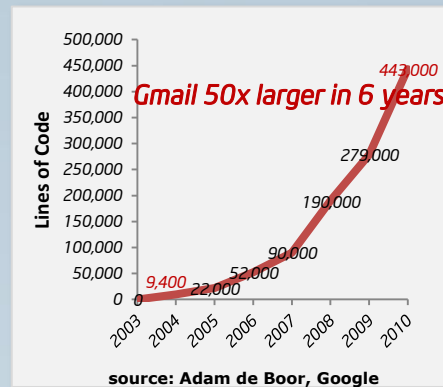


- 回顾IDF'12 《优化HTML5引擎，提升用户体验》
 - 优化的重要性
 - 优化的一般策略
- 案例探讨：什么是好的Workload
- 案例探讨：Workload的测量与分析
- 案例探讨：针对Workload的优化

回顾 - Web的性能优化至关重要



- 用户体验的全方位增强 ← 离不开性能的全面提升
 - Gmail客户端：六年50倍代码行
 - Mozilla示例：JavaScript视频分析
 - 视频音频，图片，3D，特效变换等
 - 游戏，工作，娱乐，新闻，教育等
 - 可媲美本地(native)程序的使用方式



Video analysis inside browser ([src](#))



HTML5 Angry Bird ([src](#))

Chrome Web Store

开放两星期内:

- 1.5million 下载
 - 2200+ 应用
 - 10,500 扩展和主题
 - 广泛涵盖教育、商业、游戏、社交、视频、阅读等
- Keith Dsouza on Techie Buzz ([src](#))



远程服务

应用程序

Web引擎

操作系统

硬件平台

JavaScript引擎

Layout引擎

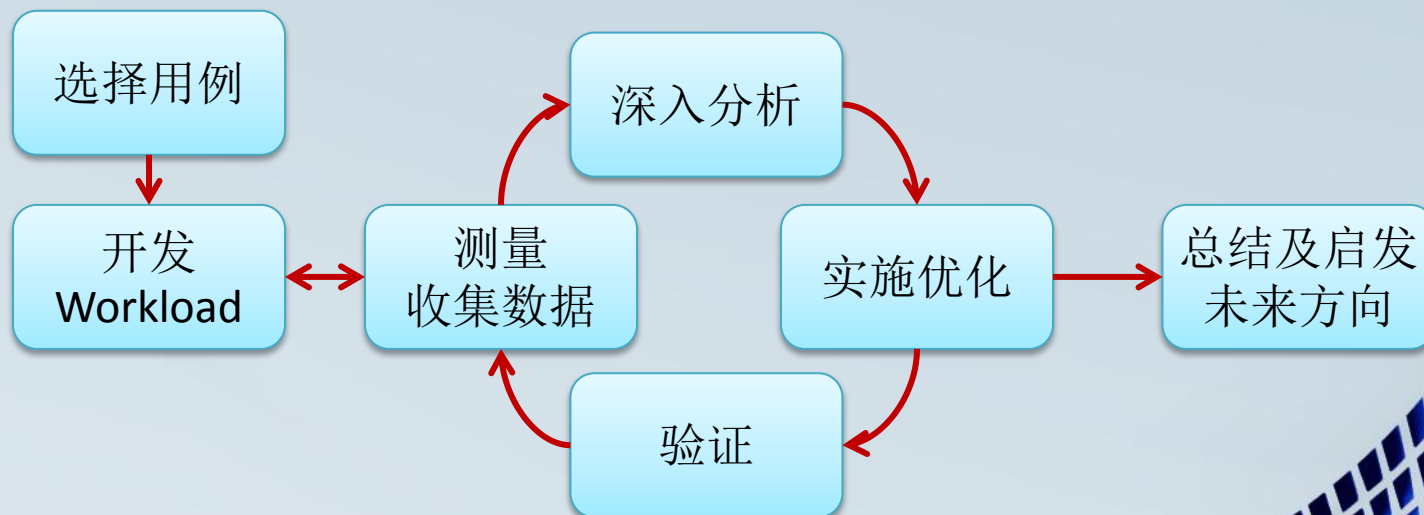
渲染引擎

- Web性能与平台整体密切相关
 - 软硬件、客户端云端协调作用
 - 本地(native)组件起重要作用
- Web引擎是最关键组件之一

回顾 - 优化策略探讨



- 优秀的workload – 源于生活，高于生活
 - 有的放矢 → 选择有代表性的用例
 - 无法测量，则无法优化 → 构建workload
- 系统的全方位的测量
- 深入的多方面的分析，发现提升的机会
- 设计合适的方案，并正确有效的实现
- 从长远和大处进一步发掘优化的潜力



Workload原则

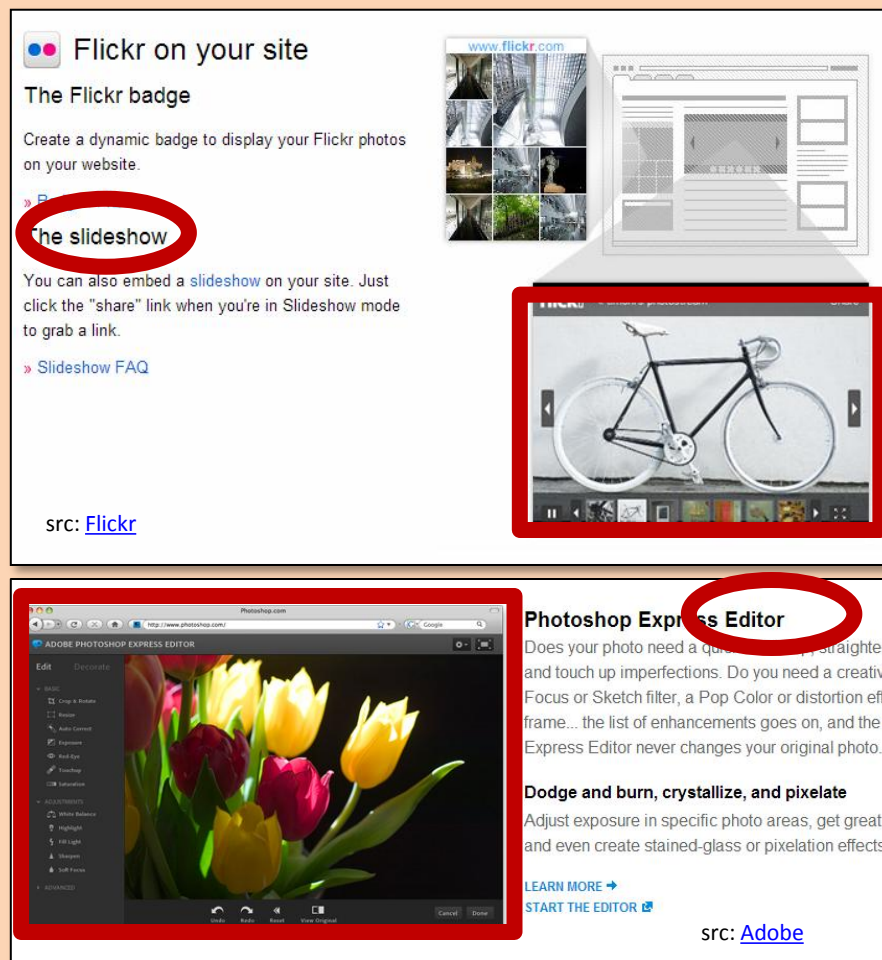


• 源于生活

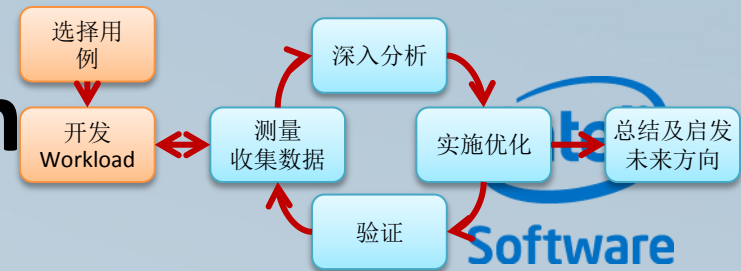
- 典型
- 重要

• 高于生活

- 可测量: 定义并实现指标
- 可再现
 - 减少差异: 多次测量, 不同测量者
 - 标准化测量规则: 电源管理选项, 设备测试方向, 屏幕亮度, 是否外接电源...
- 易用性
 - 易于安装, 配置
 - 自动化
 - 是否开源
 - 是否可扩展



Workload案例 - MyAlbum



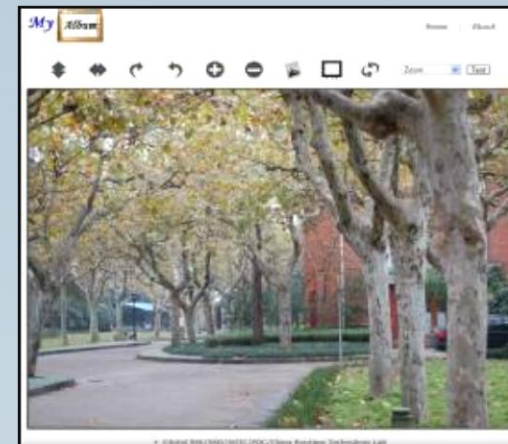
MyAlbum – SlideShow



MyAlbum – FancyShow



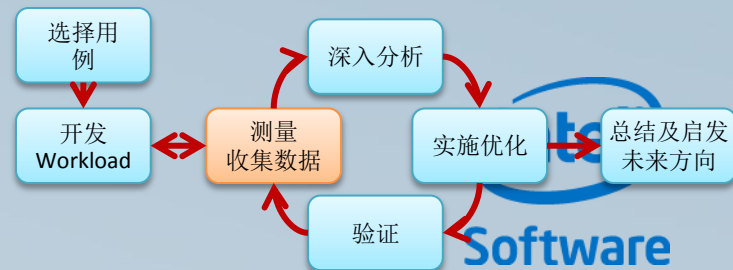
MyAlbum – Editor src: Intel



- 9个相册，即 (JPG, PNG, GIF) × (大, 中, 小)
- 每个相册含150张图片
- **自动**轮流加载并显示每张图片
- 每次执行两轮，分别按原大小显示和全屏显示
- 最终运行结果定义为 **M**
- 建议**执行3次以上**，最终结果取中位值

- $M = \text{GeoMean}(N(\text{原大小显示}), N(\text{全屏显示}))$
- $N(x) = \text{GeoMean}(T(\text{jpg}), T(\text{png}), T(\text{gif}))$
- $T(x) = \text{GeoMean}(E(\text{大}), E(\text{中}), E(\text{小}))$
- $E(x) = \text{Average}(t(1), t(2), t(3))$
- $t = \text{加载并显示完对应相册的时间} / 150$

FishIETank 案例 - 测量



Microsoft FishIETank Benchmark ([src](#))



```
function init() {  
    // other code omitted here ...  
    // start animation  
    setInterval(function () { draw(); }, 16.7);  
    // other code omitted here ...  
}  
function draw() {  
    // other code omitted here ...  
    // count one frame, and show FPS  
    fpsMeter.Draw(fish.length);  
    // other code omitted here ...  
}
```

	JS FPS	Real FPS
A	39	25

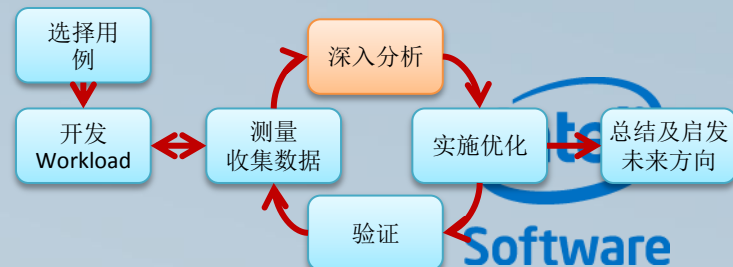
	JS FPS	Real FPS
B	53	12

用Chrome运行并查看报告FPS

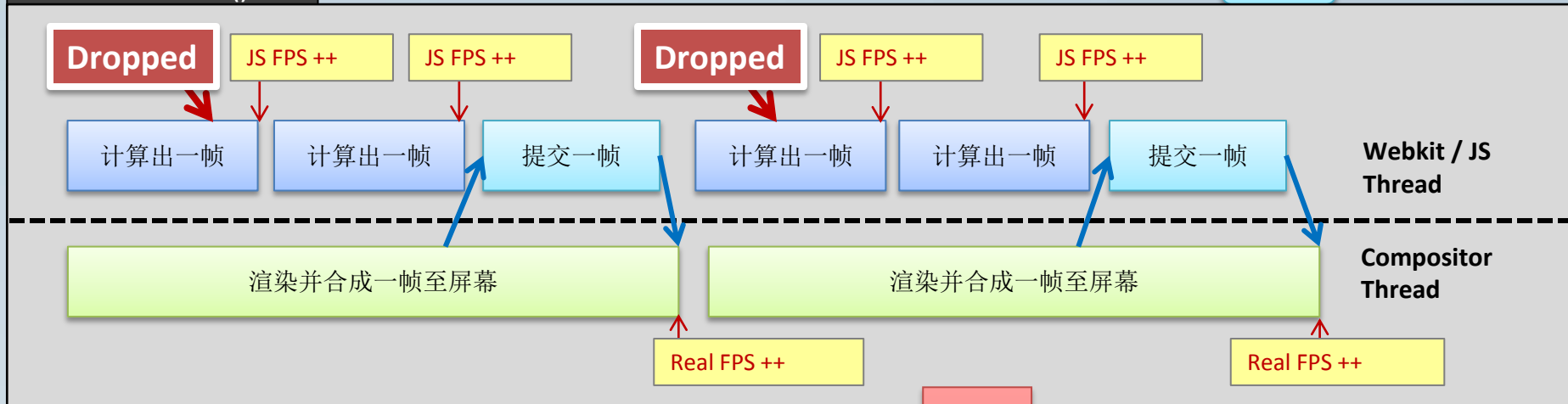
A和B哪个设备的性能更好?

使用Chrome里的FPS Counter查看
现在哪个设备更好?

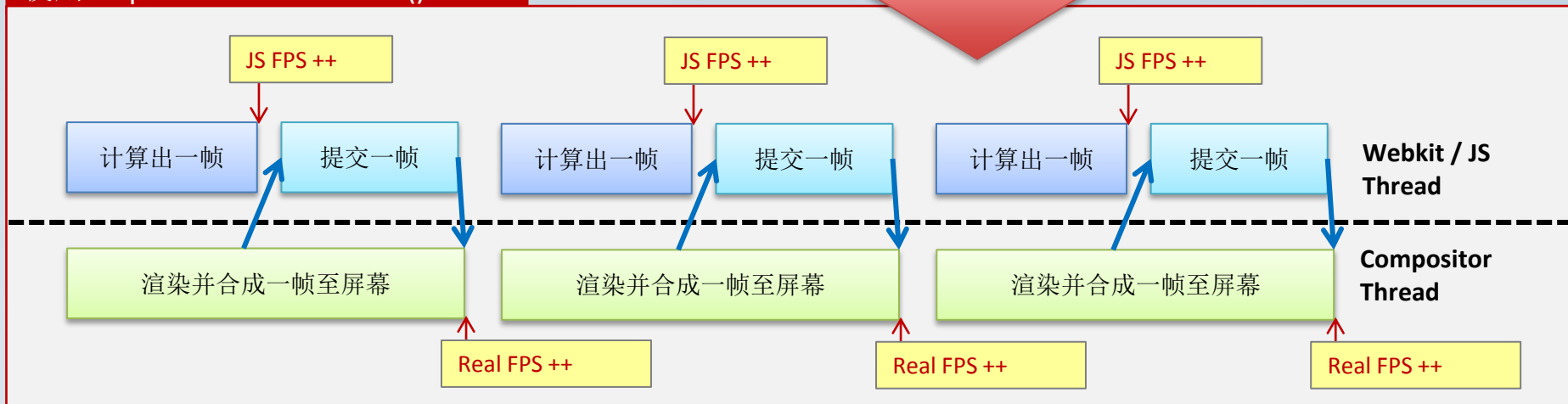
FishIETank 案例 - 分析



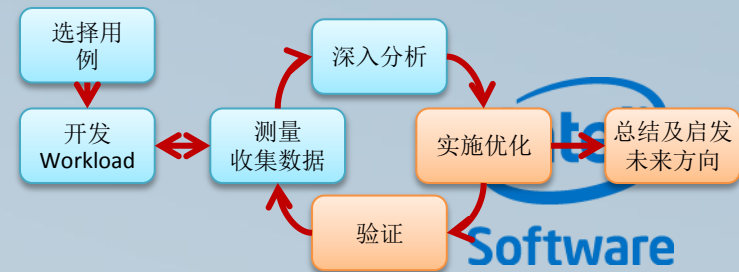
使用 setInterval()



使用 requestAnimationFrame()



SunSpider优化案例分享



Workload: 业界广泛认可的JavaScript性能基准测试

测量分析: 通过Profiling寻找热点，并分析JIT生成的汇编指令

- 某些子用例中IDIV(耗时操作)占用较多CPU
- IA上V8引擎对整数除法均生成IDIV指令

优化思路

- (右)移位指令在除数为2的整数幂时更高效

优化实现

- Optimize JIT的LIR层加入相关分析
- 生成优化的代码，并处理特殊情况

优化结果验证

- Intel Atom® 上个别子用例性能提高30%
- 贡献到upstream ([link](#))

未来优化思路

- 相关优化可以应用到乘法, 求模等运算
- 针对非常量的乘数/除数的移位操作

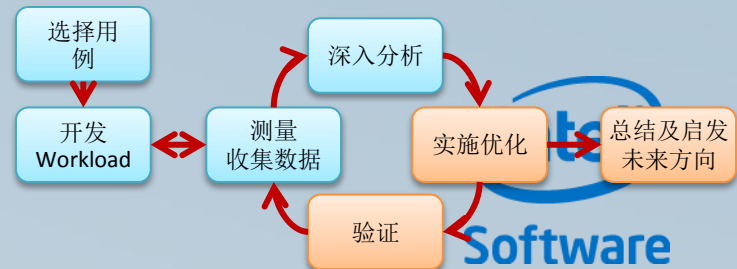
src:[SunSpider](#)

```
function A(i,j) {  
    return 1/((i+j)*(i+j+1)/2+i+1);  
}
```

```
cmp eax, 0x80000000  
jne 0x178  
cmp ebx, 0xFFFFFFFF  
je 0x17ab3538    ;;; check for MinInt / -1  
cdq  
idiv ebx  
test edx, edx    ;;; check for the remainder  
jne 0x17ab3538
```

```
test dl, 0x1    ;;; check for the remainder  
jne 0x1c8  
sar edx, 1
```

FishIETank优化案例分享



Workload: 业界广泛认可的HTML5 Canvas 2D 性能基准测试

测量分析: 通过tracing发现时序(pipeline)中的性能瓶颈

- GPU的延时导致整个时序延迟一个VSYNC间隔
- 大量时间用于GL Context之间的切换

优化思路

- 避免昂贵的GL Context切换

优化实现

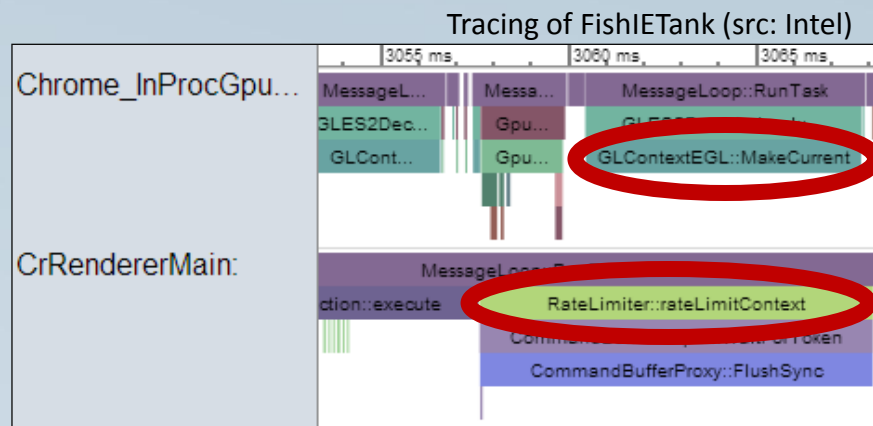
- 创建虚拟的Context供程序逻辑使用
- 多个虚拟Context共享一个真实的Context
- 虚拟Context记录并处理自身状态变化
- 虚拟Context的切换不产生真实的切换

优化结果验证

- Intel Atom® 上性能提高超过20%
- upstream 完成实现([link](#))

未来优化思路

- Context切换时间因GPU而异。进一步做有针对性处理

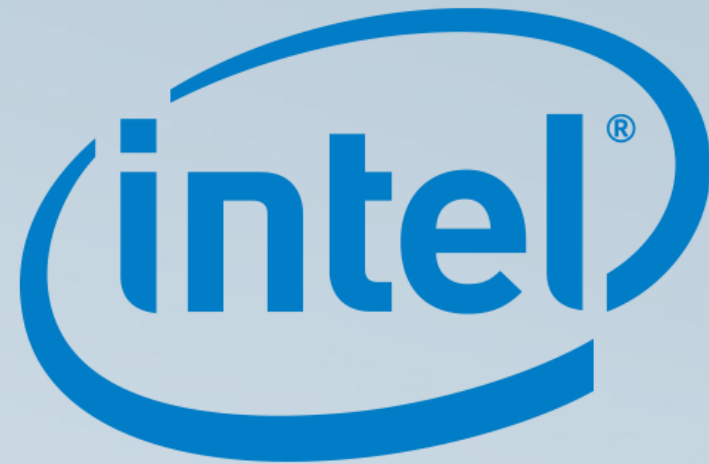


总结



- 性能优化对**HTML**应用至关重要
- **HTML5**的优化通常从建立可靠的**Workload**开始
- 测量和分析帮助深入理解应用场景和浏览器内部流程
- 成功的优化是对前期所有工作的升华和进一步拓展

Intel将与业界一道，致力于不断提升**HTML5**的性能和体验



Software

software.intel.com/html5