



开放 分享 创新

developers.google.com



Chrome, HTML5, WebGL, 和 Native Client 开发技术介绍

谷歌 开发技术推广部 大中华区主管 栾跃

Bill Luan

Greater China Regional Lead, Developer Relations

Google

基于浏览器的应用开发 是创新的未来！

(开发印在光碟上销售的客户端软件
是上一个世纪的老东西了！)

浏览器开始的一场创新的革命 今天仍旧继续着...

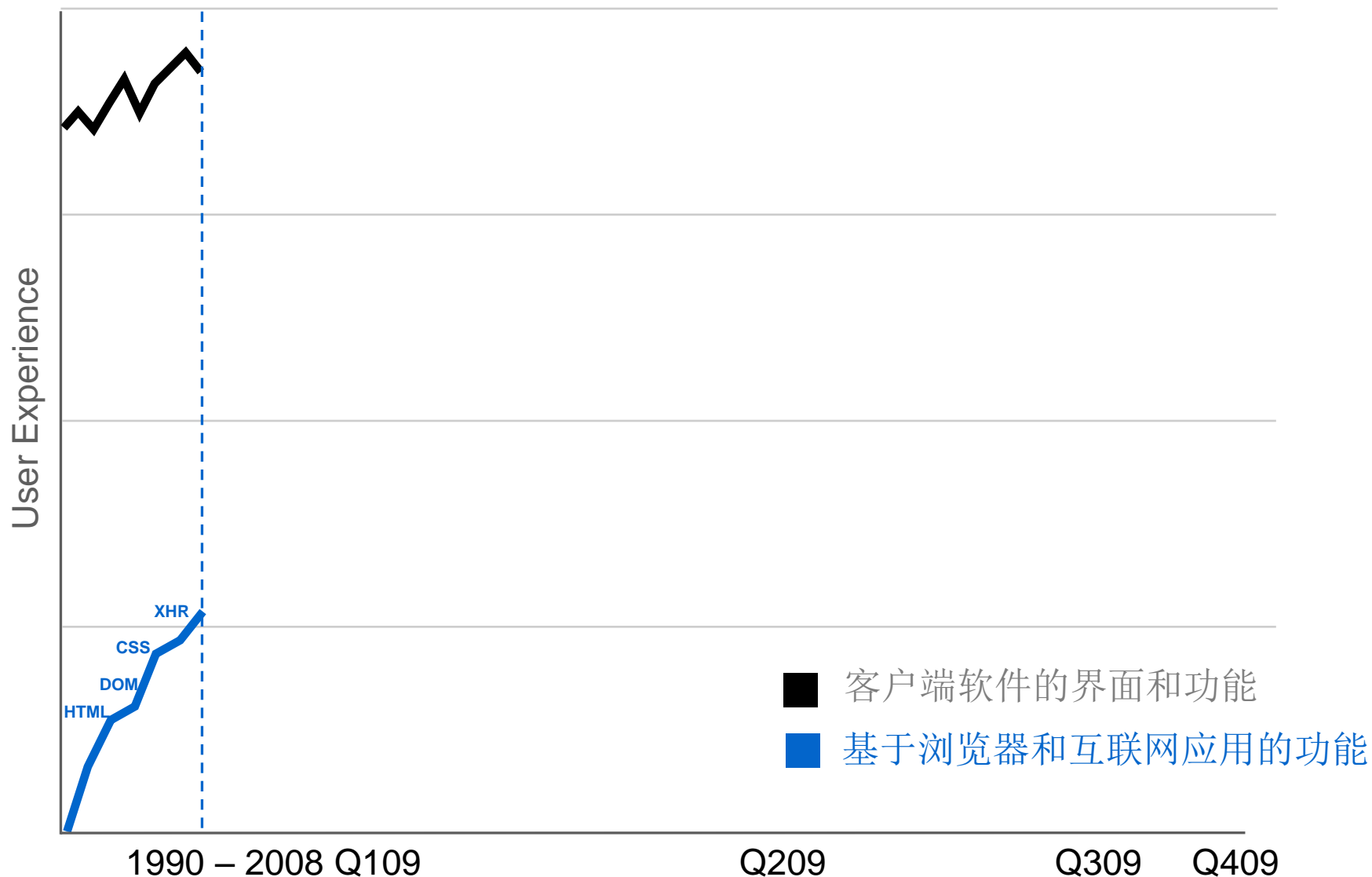
- 1995年, Netscape 推出了 JavaScript
- 1999, Microsoft 推出了 XMLHttpRequest
- In 2002, Mozilla 1.0 把支持 XMLHttpRequest 加入到该浏览器产品所提供的固有功能之一

... 从那个时刻起, 在浏览器里的网络应用开发 (web applications) 开始起飞 ...

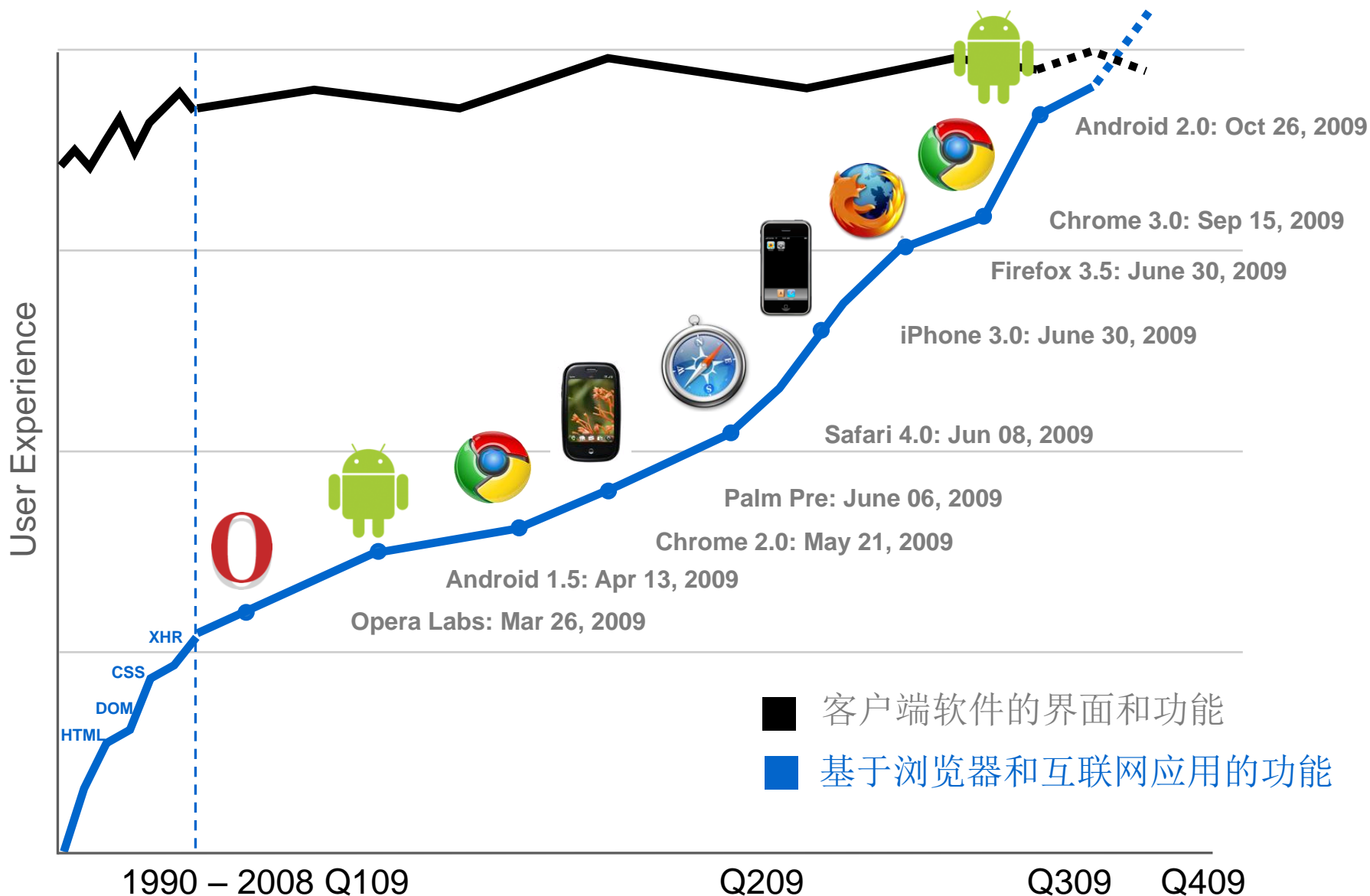
- In 2004, Gmail 开始发行
- In 2005, 带有 AJAX 功能的产品开始发行 (比如谷歌地图 Google Maps)

... 今天, 更多的网络应用也需要更加强大的功能

基于浏览器的网络应用飞速发展

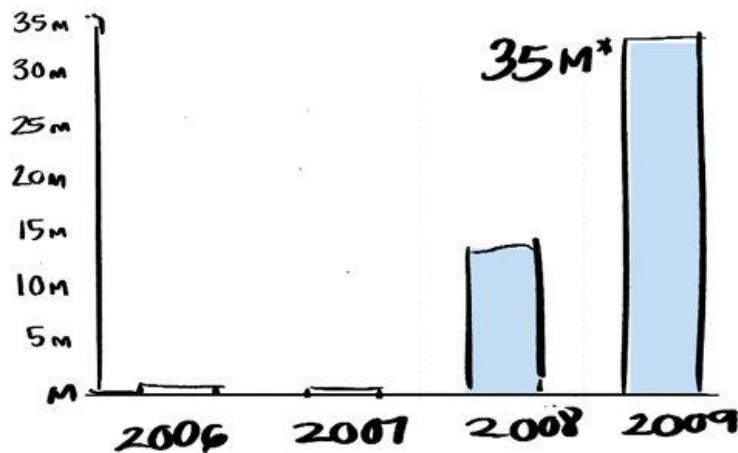


基于浏览器的网络应用飞速发展



这几年的业界的发展所显示的明显的趋势

Netbook (Shipments)



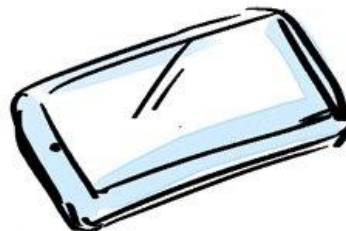
Millions of users living in the cloud



Laptops



Netbooks



Tablets



Phones



今后几年创新的商机所在

- **基于浏览器的应用 (WebApps) 将是今后几年里在信息技术应用开发上进行创新的最好领域**
 - 能够随时连网的移动设备将使百姓大众和各个行业进一步依靠互联网进行信息的获取和使用
 - 不断提高功能和性能的浏览器将是普通大众使用信息技术和服务的主要工具
 - 各种云计算服务将强化大众对浏览器的依赖
 - 具有良好安全性、管理性、和高性价比的以浏览器为主要使用界面的上网本 (Netbooks) 将在企业获得大量使用 (比如 ChromeBooks)
 - 基于浏览器的应用开发目前是一个相对较新的领域、也将是开发者们创新赢利的最好机会

- **什么是基于浏览器的应用** (WebApps)
 - 提供各种信息运算和显示功能的应用和服务软件
 - 在浏览器里运行
 - 它们可以是具备高度图像界面功能的、高速运行性能的、类似传统客户端软件功能和使用经历的应用
 - 可以用脚本语言也可以用传统语言例如 C/C++ (例如使用 Native Client 技术) 开发

基于浏览器应用的优势

- **通用性**：所有计算机上都有浏览器，用户只要使用浏览器就都能运行和看到信息
- **方便性**：不再需要安装专门的客户端软件
- **跨平台**：在桌面计算机和移动计算机（智能手机、平板电脑等等）都能看到
- **高效性**：你的应用开发一次，能在多种设备上和任何地点被多处使用
- **云计算**：这样的运用与互联网紧密联结，可以充分发挥互联网和高性能服务器的优势

实现浏览器应用愿景的条件

- **兼容性**: 浏览器需要符合相同的标准
 - 这也是为什么与公开标准兼容的重要性
- **高性能**: 浏览器需要有极高的运行速度
 - 基于开源代码的WebKit等引擎提供了这样的性能
- **支持关键技术**: 浏览器需要支持一些关键的技术、使得强大功能的设计能够实现
 - 过去客户端软件能够做到的，但是浏览器却不能提供的一些关键的功能，例如像素层面的高性能渲染、多媒体功能、多线性进程、本地数据储存等等

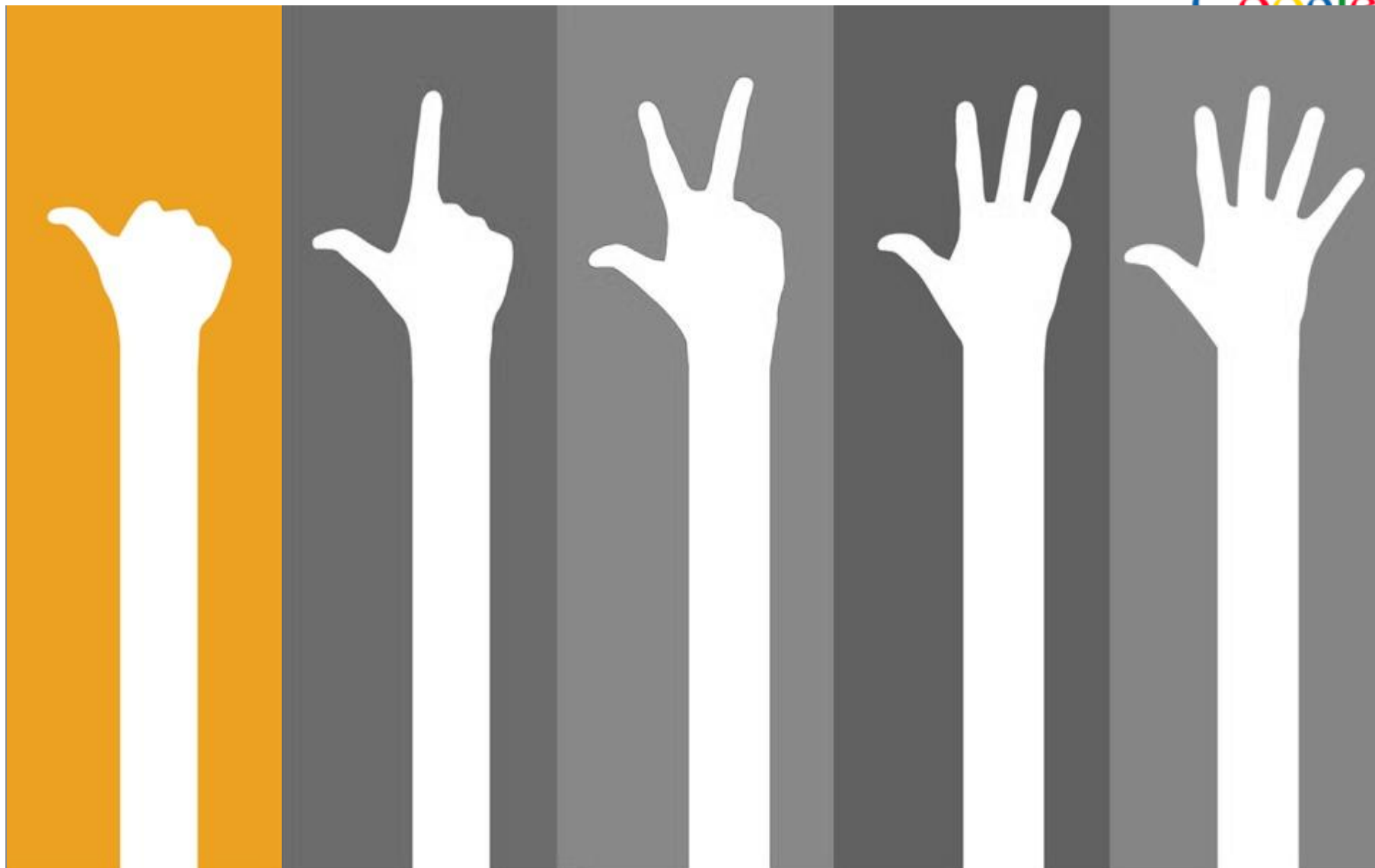
HTML5打破了这些关键技术的技术瓶颈



HTML5提供了这些关键开发技术

- 高性能图像渲染、像素层次的细节、直接利用GPU图像渲染硬件功能: Canvas and SVG
- 三维图象支持: CSS3、WebGL
- 多媒体支持: Audio and Video
- 地理位置信息整合: Geolocation
- 多线程同步运算: Web Workers
- 离线存储: Web Storage
- 数据传输: Communication: Cross Document Messaging & XMLHttpRequest Level 2
- 实时双向通讯: WebSocket





canvas

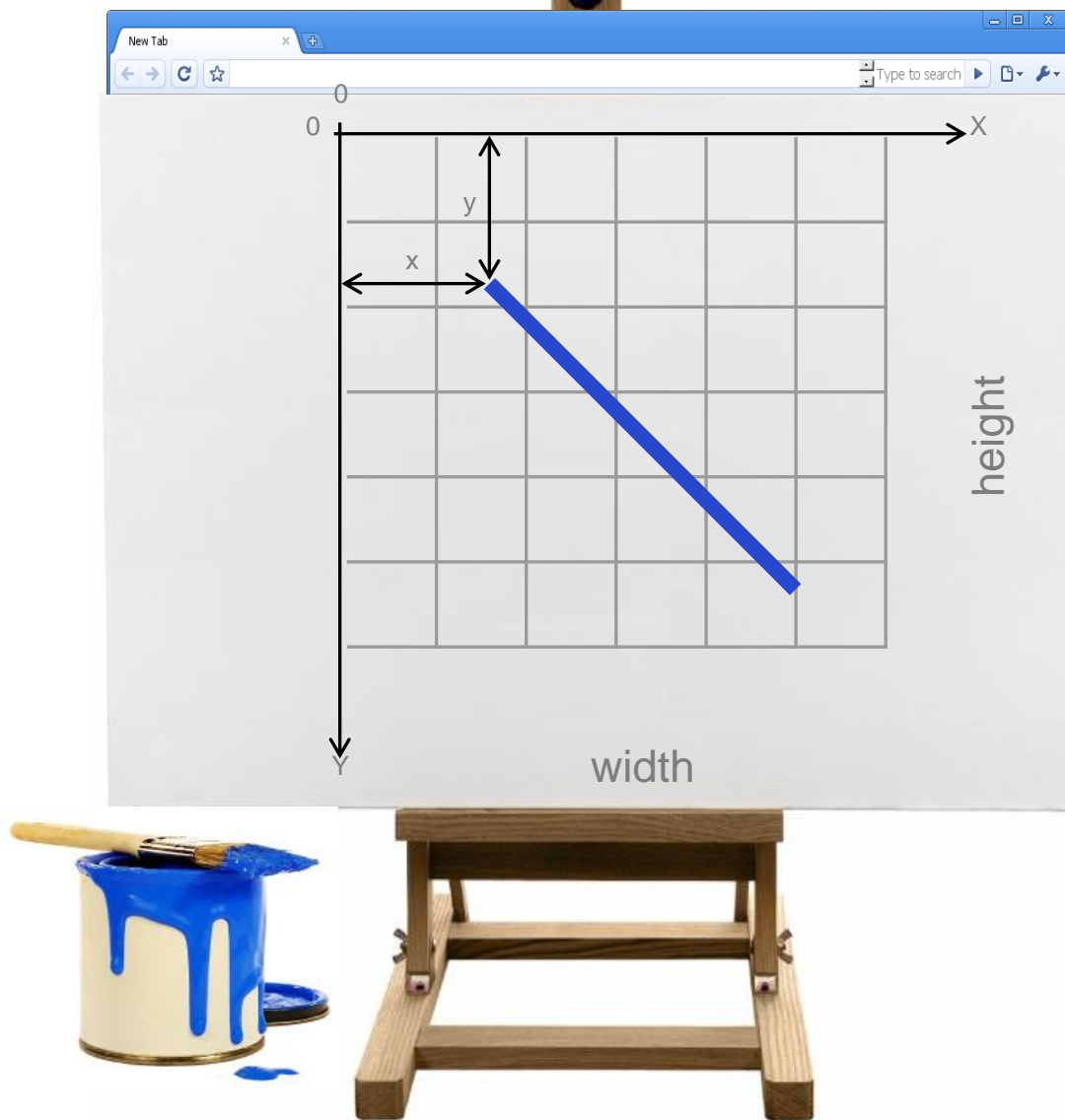
video

geolocation

app cache &
database

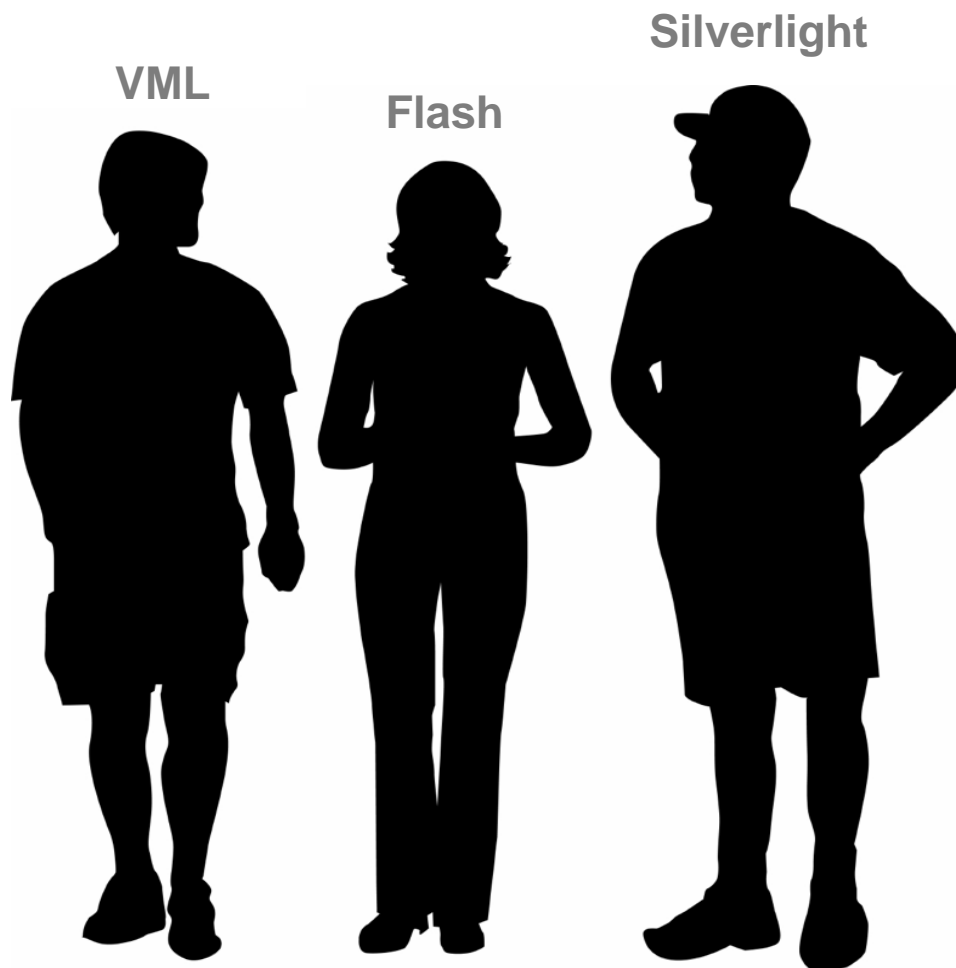
web workers

直到最近，你是无法在浏览器里画画的



过去你要是需要进行图像渲染...

- ... 逃不掉几个通常的嫌疑犯：必须要使用受某个供应商控制的插件

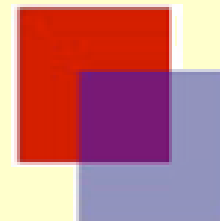


HTML5 画布技术<Canvas>把你解放出来

- 它是HTML5所最早标准化的技术之一，也是最早被现代化浏览器所支持和实现的 – Chrome, Safari, Firefox, and Opera
- 它为开发者们提供了一个可以进行二维图像和三维图像渲染的“画布”

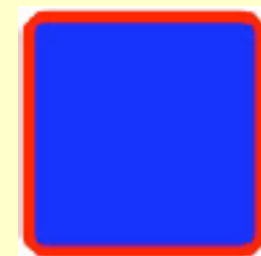
```
<canvas id="myCanvas" width="150" height="150">  
</canvas>
```

```
var canvas = document.getElementById('myCanvas');  
var ctx = canvas.getContext('2d');  
ctx.fillStyle = "rgb(200,0,0)";  
ctx.fillRect (10, 10, 55, 50);  
ctx.fillStyle = "rgba(0, 0, 200, 0.5)";  
ctx.fillRect (30, 30, 55, 50);
```



- **SVG** - Scalable Vector Graphics提供矢量图功能
- 它也使用HTML那样的简单代码格式

```
<!DOCTYPE html>
<html>
  <body>
    <svg width="200" height="200">
      <rect
        x="0" y="0"
        width="100" height="100"
        fill="blue" stroke="red"
        stroke-width="5px"
        rx="8" ry="8"
        id="myRect" class="chart" />
    </svg>
  </body>
</html>
```



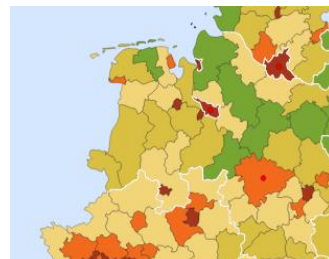
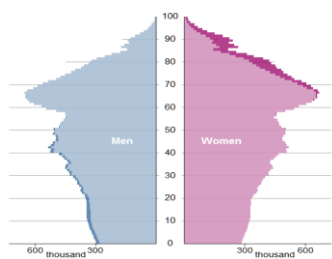
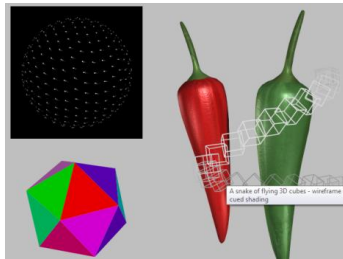
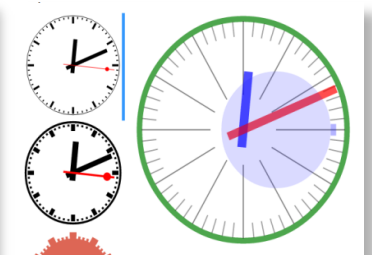
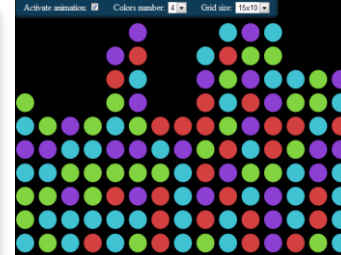
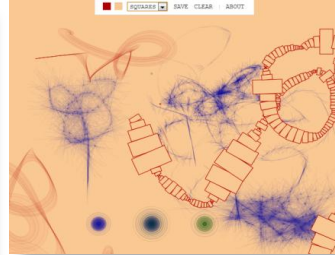
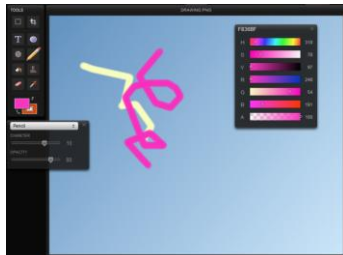
HTML5 画布技术<SVG>解决矢量图需要

- 矢量图功能 - 对图像的放大



Canvas 和 SVG 使用的区别

- SVG适合于高层面 (High level)的应用
 - 图像的直接导入和导出 (Import/Export)
 - 应用使用简单的界面
 - 应用需要较多的交互性的功能
 - 使用中等量动画
 - 编程设计使用互相关联的类 (Tree of objects)
- Canvas 适合于低层面 (Low level)的应用
 - 不需要对图像进行鼠标的控制
 - 主要以JS为编程语言
 - 像素层面的图像显示
 - 使用大量的动画
 - 较多的管理需求 (bookkeeping)



Canvas & SVG demos

新的网页设计和渲染技术

- **显示和渲染技术: HTML5**

- 画布(Canvas)
- 矢量图 (SVG)
- 像素渲染技术 (WebGL)

- **显示和渲染技术: CSS**

- **CSS过滤器** (CSS Filter)
- **CSS区域** (CSS Regions)
- **自定义着色器** (CSS Custom Filters – Shaders)
- **CSS 形状** (CSS Exclusions – Shapes)

HTML



CSS



显示和渲染技术: CSS

CSS过滤器的一些基本功能函数

- 合并 (Merge)
- 模糊 (Blur)
- 噪音 (Noise)
- 卷积 (Convolution)
- 点光源 (Point Light)
- 射灯 (Spot Light)
- 移位 (Displacement Map)
- 等等



使用CSS模糊过滤器加入朦胧效果



```
.blurme { filter: blur(4px); }
```


过滤器还可以合并造成新的效果



```
.combined { filter: hue-rotate(180deg) sepia(80%); }
```


过滤器合并顺序可以造成不同的效果



```
.ordered { filter: sepia(80%) hue-rotate(180deg); }
```

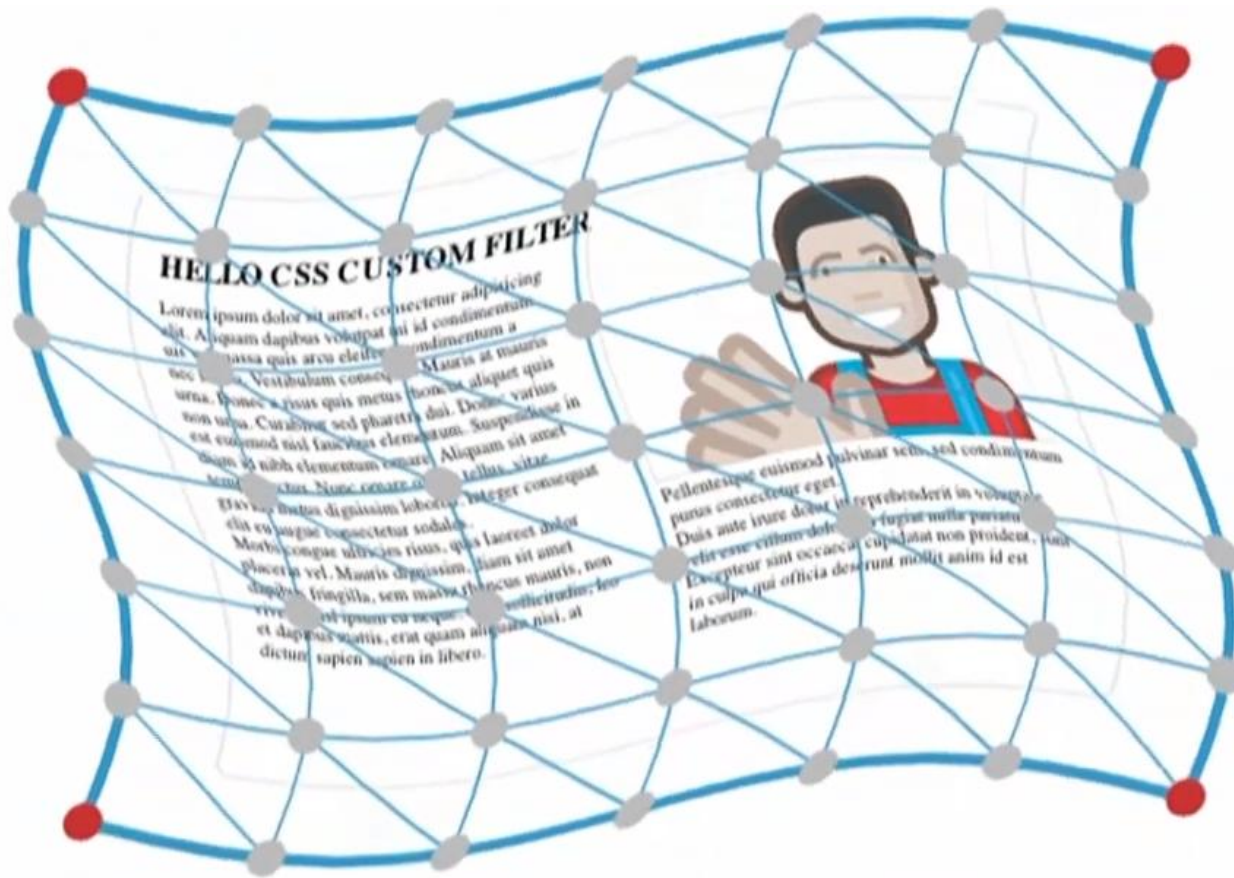

将常用的CSS过滤器进行包装组合性使用

- 对不同的CSS过滤器进行选择
- 采用事先预定义的组合
- 让浏览器去进行最佳化的渲染
- 将运算工作量的负载转移到GPU上去

```
.filterme {  
  -webkit-filter: opacity(100%) grayscale(0%) sepia(0%)  
    saturate(100%) hue-rotate(197deg) invert(0%) brightness(100%) contrast(100%) blur(0px)  
    drop-shadow(gray 0px 0px 10px);  
  -moz-filter: opacity(100%) grayscale(0%) sepia(0%)  
    saturate(100%) hue-rotate(197deg) invert(0%) brightness(100%) contrast(100%) blur(0px)  
    drop-shadow(gray 0px 0px 10px);  
  filter: opacity(100%) grayscale(0%) sepia(0%)  
    saturate(100%) hue-rotate(197deg) invert(0%) brightness(100%) contrast(100%) blur(0px)  
    drop-shadow(gray 0px 0px 10px);  
}
```


自定义过滤器 (也叫CSS 着色器)

- 把任何网页的内容当作一个平面对待
- 把这个平面当作一个三维图象处理的纹理来对待(texture)
- 为这个平面提供顶点(vertex)和片段的着色(fragment shading)
- 对这个网页的内容进行像后处理那样的任意变化操纵





canvas

video

geolocation

app cache &
database

web workers

在浏览器里进行视频显示过去一直求救于第三方的插件

- 你的客户是否能够看到视频还往往像赌博一样无法事先确认和靠运气...



// HTML 5 makes
<video> as easy
as

HTML5提供了视频<video>和音频<audio>的功能

- 让浏览器可以在网页上轻松地放映视频和音频
 - 不需要再依赖任何第三方的插件
 - 编程的代码就像嵌入一个图像文档那样方便
`<audio src="song.mp3">`
- 提供了各种常用的多媒体视频音频的播放功能
 - 播放 (Play)
 - 暂停 (Pause)
 - 停止 (Stop)
- 可以用代码进行动态的控制 (Scriptable)
- 已经成为开源代码浏览核心WebKit 的固有功能

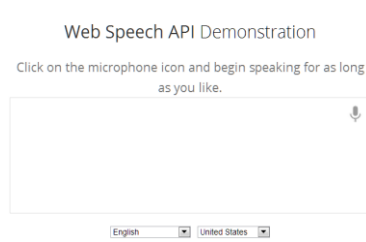
新的Web Speech API功能

- 让开发者可以在网页里方便地嵌入语音输入的功能

```
var recognizer = new SpeechRecognition();
recognizer.continuous = true;
recognizer.interimResults = true;
recognizer.onresult = function(e) {
  if (e.results.length) {
    var lastResultIdx = e.results.resultIndex;
    console.log(e.results[lastResultIdx][0].transcript);
  }
};
recognizer.start();
```

JS

- 可以做到连续的音频输入，甚至由声音驱动的应用 (Voice Driven App)

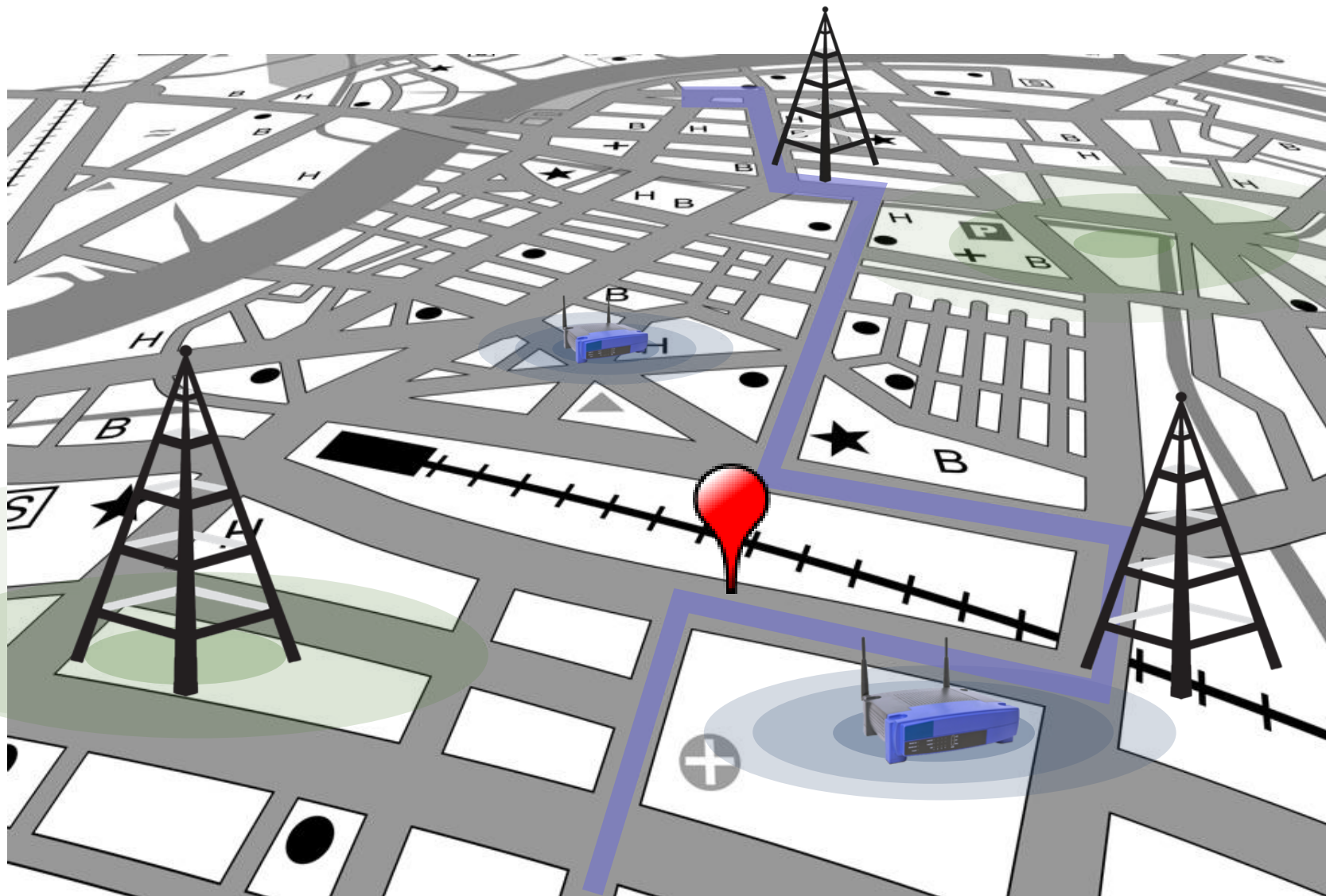




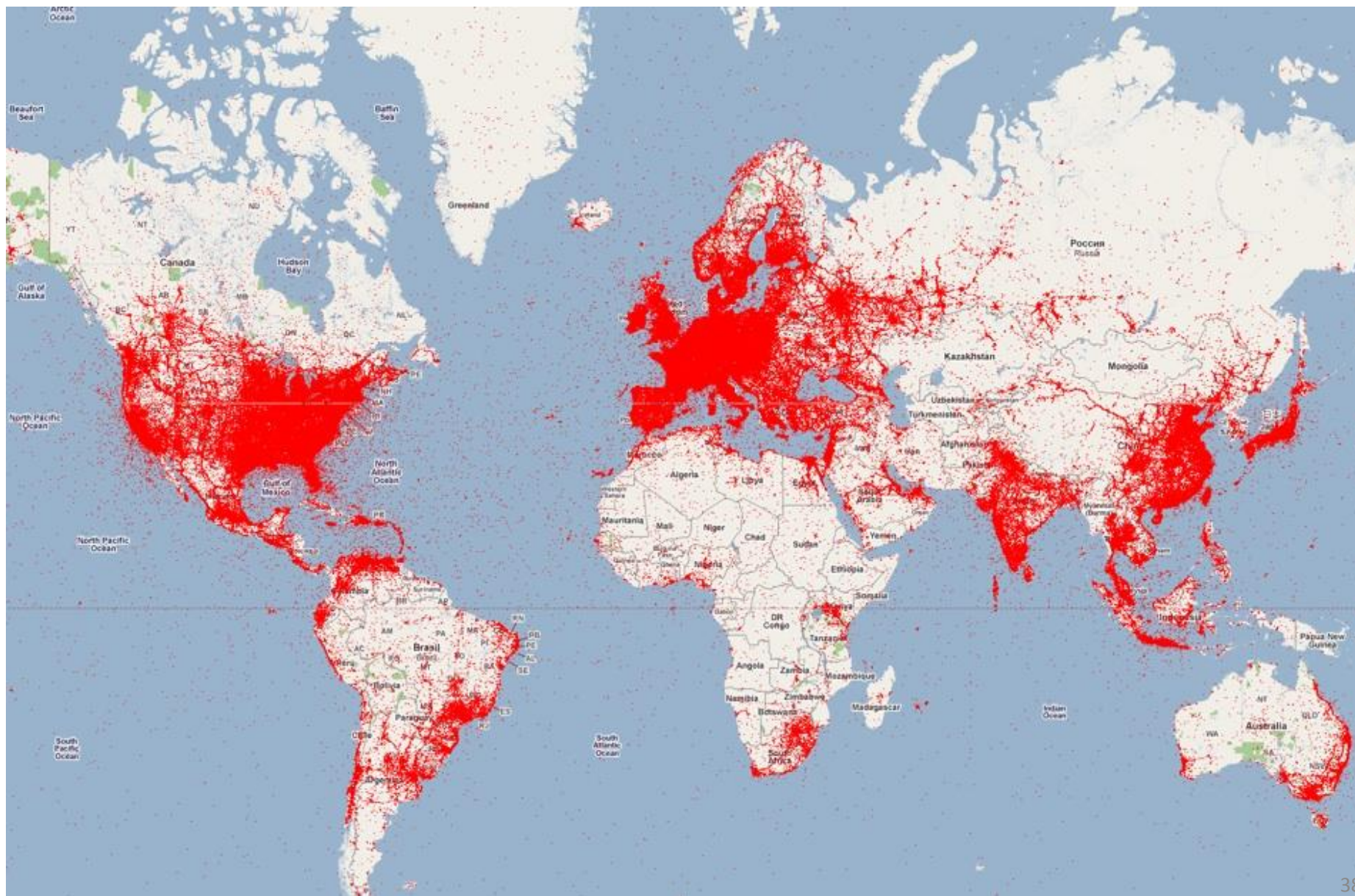
video demos



计算机和手机的IP地址以及移动或无线信号塔解决了任何设备地点计算的问题



而且全球都有很好的覆盖



// the
geolocation api
brings browser-
based location to
your apps

HTML5的Geolocation技术帮助定位

- 在客户设备上使用JavaScript API轻松找到客户的位置
- 支持各种定位信号和技术：GPS, IP地址, 蓝牙、无线
- 作为服务的选项, 可以将你的位置与你信任的朋友分享
- 位置的信息可以用程序进行动态化的实时跟踪

```
// Single position request.
```

```
navigator.geolocation.getCurrentPosition(successCallback);
```

```
// Request position updates.
```

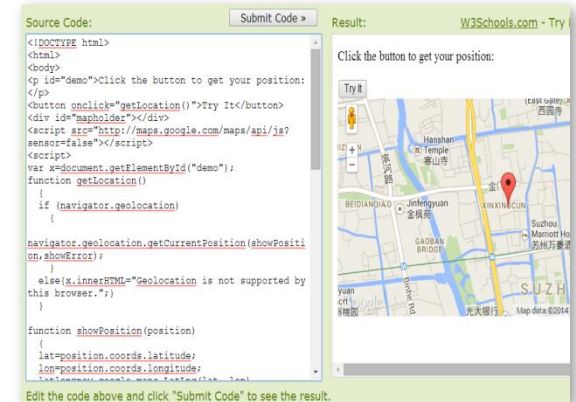
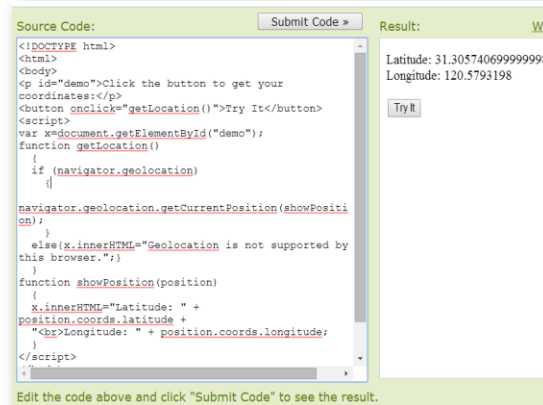
```
navigator.geolocation.watchPosition(successCallback);
```


HTML5 geolocation 代码举例



你只需要下面这样的简单代码就可以将位置画在谷歌地图上：

```
function showMap(position) {  
    // Show a map centered at  
    // (position.coords.latitude, position.coords.longitude).  
}  
// One-shot position request.  
navigator.geolocation.getCurrentPosition(showMap);
```

geolocation demos



canvas

video

geolocation

app cache &
database

web workers

// web workers
defines an API
for running
background
scripts

HTML5提供了多进程功能Workers



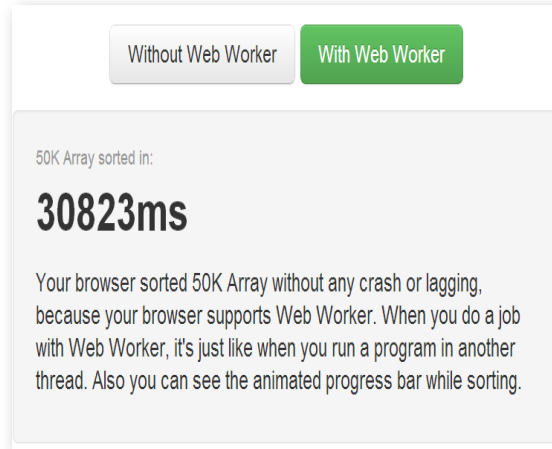
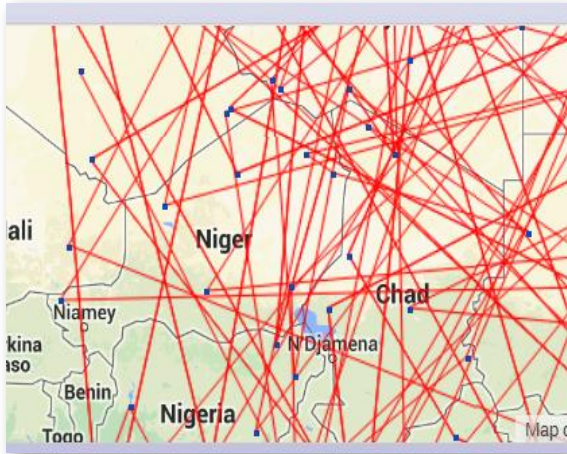
- Workers 提供了让应用程序将运算任务分到不同的线程的手段
- 可以将运算负载重的任务卸载到一个单独的线程，使应用程序不会受到影响
- 可以支持3种多线程的模式：
 - **Dedicated** (绑定到一个制表 - bound to a single tab)
 - **Shared** (在多个窗口之间共享)
 - **Persistent** (浏览器关闭后仍旧可以运行)

main.js:

```
var worker = new Worker('extra_work.js');  
worker.onmessage = function (event) { alert(event.data); };
```

extra_work.js:

```
// do some work; when done post message.  
postMessage(some_data);
```

web workers demos

现代化浏览器对HTML 5的支持

	Chrome	Firefox	Safari	Opera
canvas	✓	✓	✓	✓
video	✓	✓	✓	✓
geolocation	✓	✓	✓ (iPhone)	✓
app cache	✓	✓	✓	✓ (mobile)
database	✓	✓	✓	✓ (mobile)
workers	✓	✓	✓	✓ (mobile)

HTML5 & CSS3 READINESS

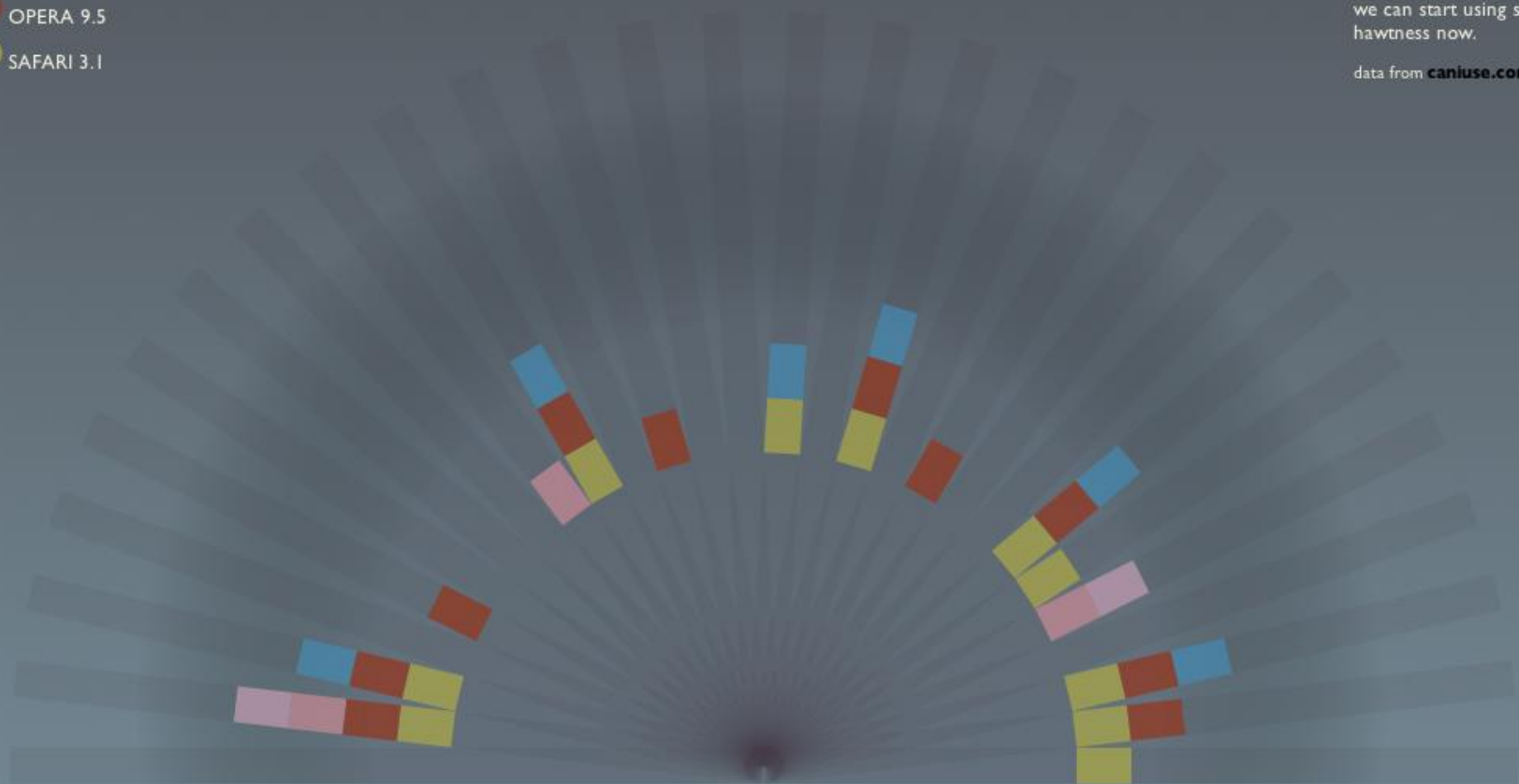
2008 / 2009 / 2010 / 2011

- IE 6
- IE 7
- FIREFOX 2.0
- OPERA 9.5
- SAFARI 3.1

Many of these features are ready to implement *today*.

We don't have to wait for entire specs to be completed, we can start using some hawtness now.

data from caniuse.com



FROM THE ASYLUM RUN BY **PAUL IRISH** AND **DIVYA MANIAN** | **ABOUT THIS SITE**



HTML5 & CSS3 READINESS

2008 / 2009 / 2010 / **2011**



Many of these features are ready to implement *today*.

We don't have to wait for entire specs to be completed, we can start using some hawtness now.

data from **caniuse.com**



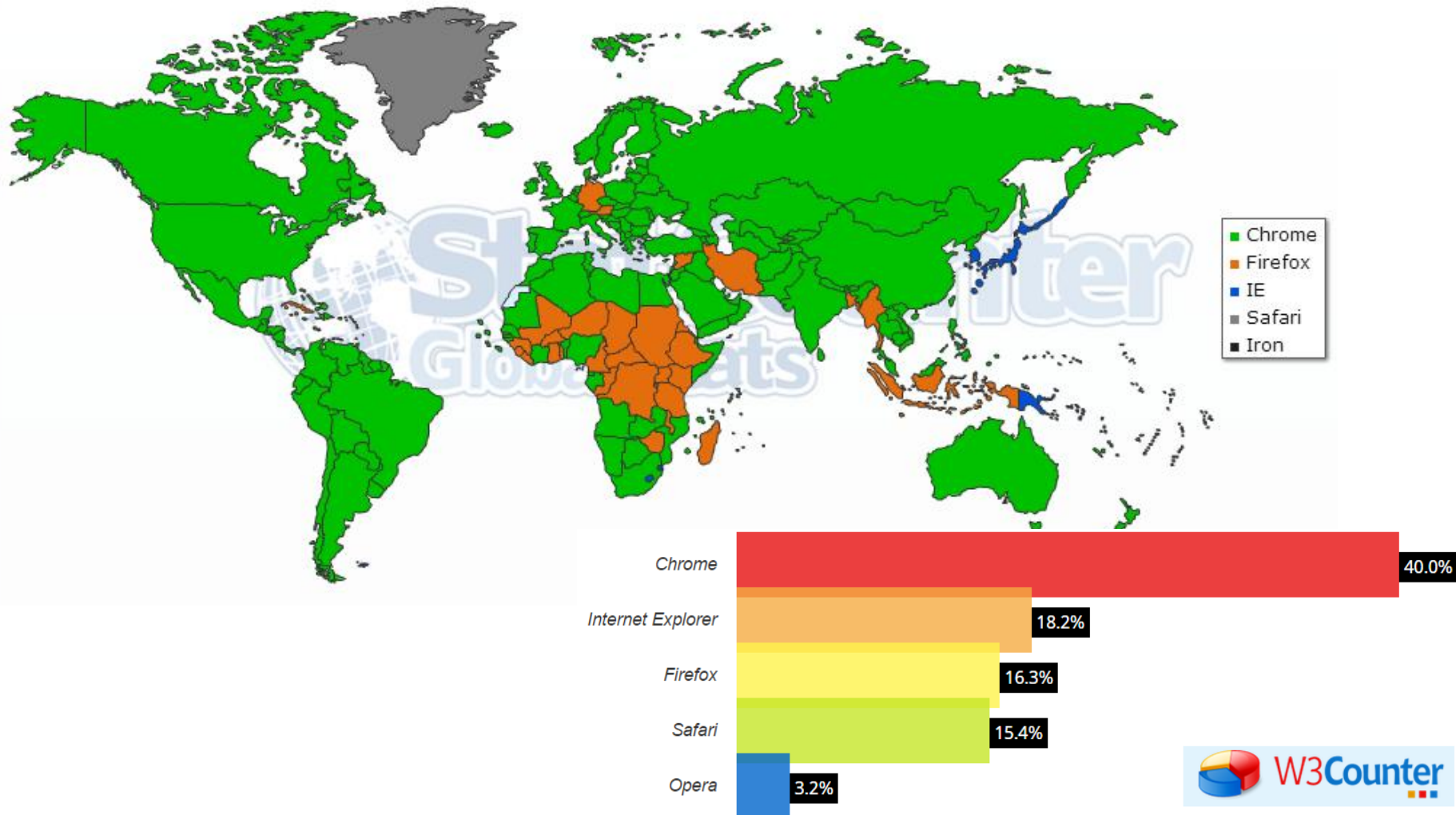
FROM THE ASYLUM RUN BY **PAUL IRISH** AND **DIVYA MANIAN** | **ABOUT THIS SITE**



现代浏览器用户的增长势头

StatCounter Global Stats

Top Desktop, Tablet & Console Browsers Per Country from Aug to Sept 2014



全球浏览器市场的最新数据



750M

+

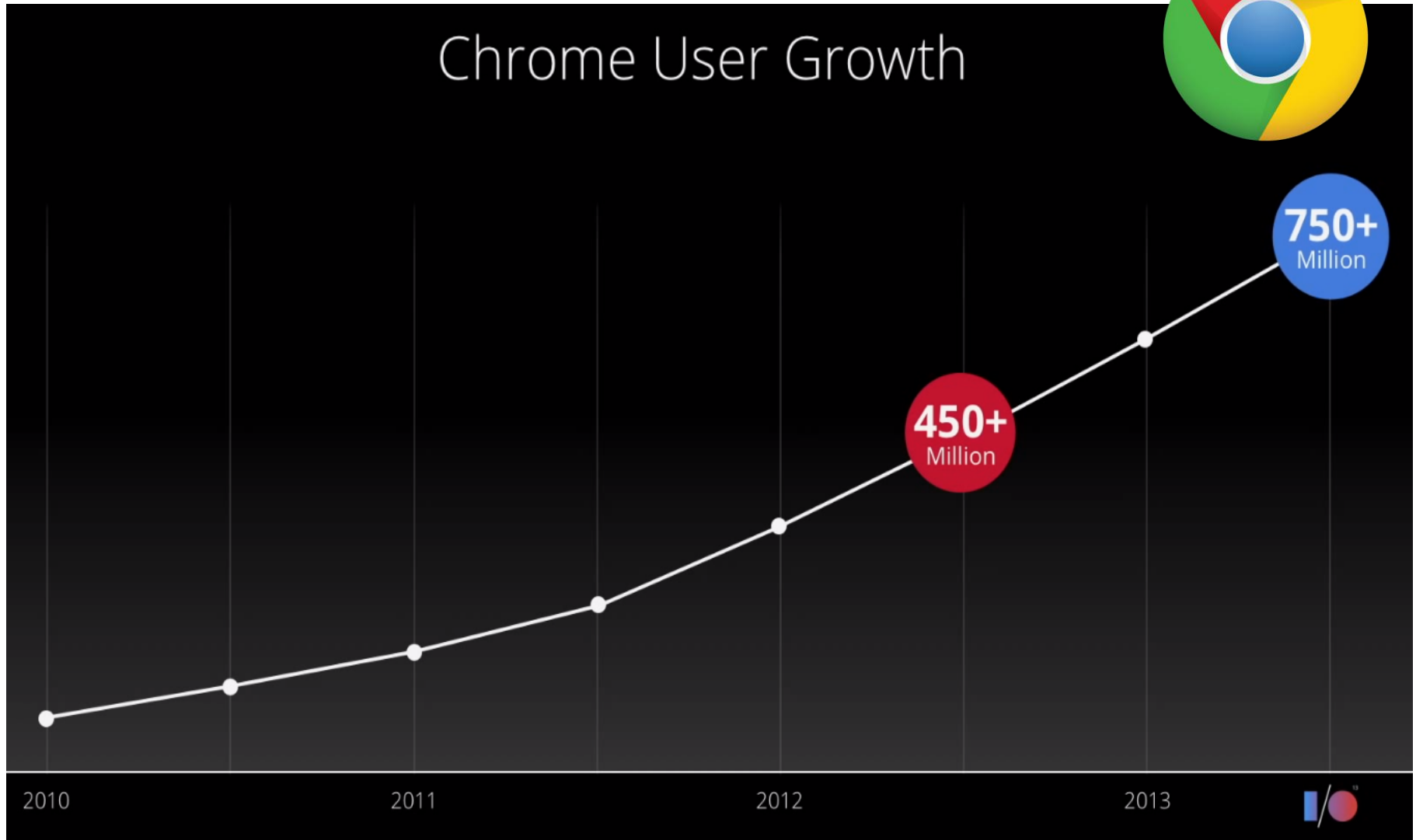
超过7.5亿的使用者！



Chrome的用户增长历史



Chrome User Growth



Chrome 浏览器的意义



- **完全支持国际公开标准的浏览器**
 - 对HTML语言的支持，完全符合W3C组织的标准
- **内核采用开源的WebKit引擎**
- **产品技术代码开源**
- **对新型HTML5的技术具备领先业界的支持**
- **支持Native Client技术, 充分发挥移植本地代码(Native Code)向浏览器转换的开发优势**
- **不断增进的运行速度和性能**
- **不断增进的开发者工具**

更多的网络应用开发技术

- 新的图像压缩技术
- 新的视频压缩技术
- 新的音频技术
- 新的多媒体数据实时通讯技术



新型图像技术WebP来帮助解决问题

- 更加优化的的图像数据压缩
- 提供有损和无损的压缩模式
- 提供Alpha通道的功能
- 提供动画的功能
- 大大降低你的应用和游戏的发行负载、让他们下载和使用速度更快、显示质量更好

WebP和JPG图像的比较

有损压缩

JPEG vs WebP

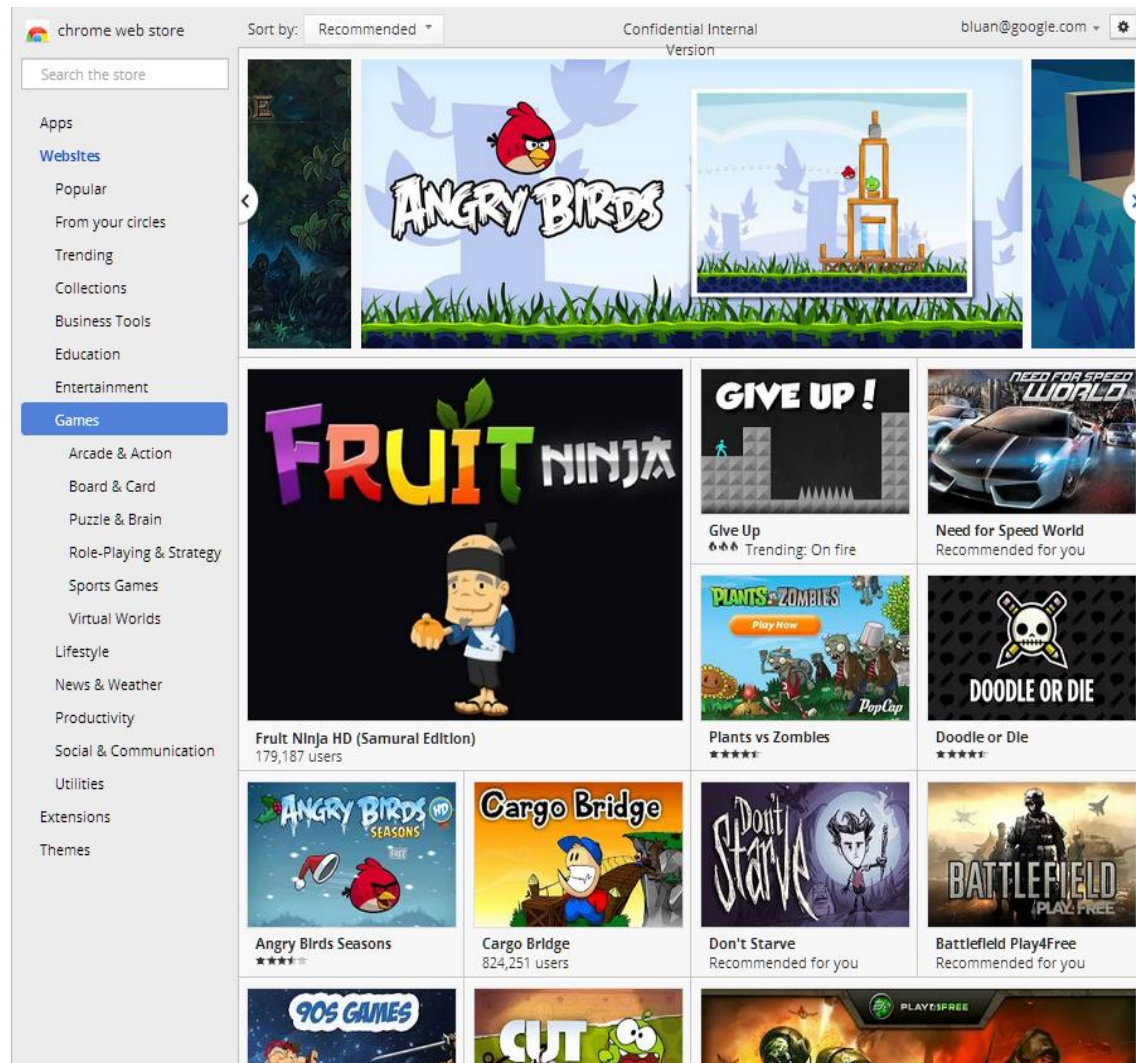


JPEG
676K

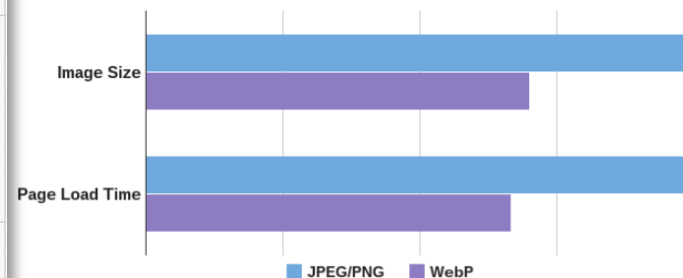


WebP
468K

WebP使用案例: Chrome Web Store



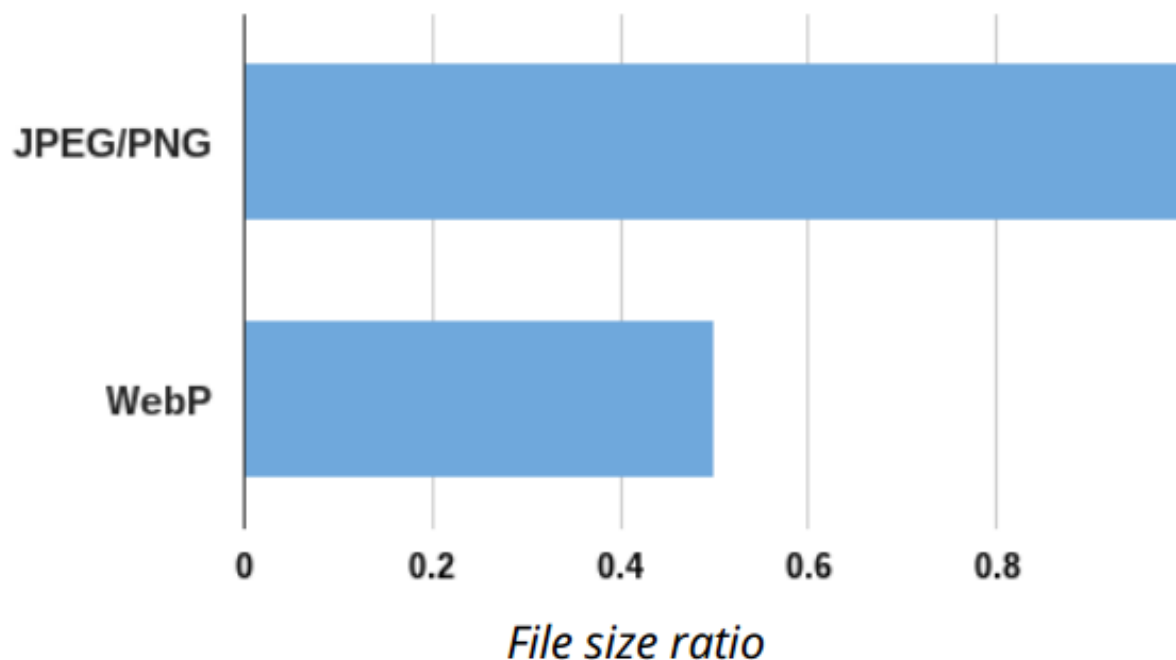
采用WebP 使得网站的
图像储存量降低
30%，网页显示速度
降低近三分之一



在移动平台上WebP的优势更加明显



- 移动应用和游戏使用更大的图像使用量
- 采用WebP可以节省近50%的流量和带宽
- 帮助你的用户省钱并获得更流畅的使用经历



谷歌的大量产品和服务在开始使用了

Google maps

GmailTM
by Google

chrome

Google docs



Google play



You TubeTM

Google Display Network

GoogleTM
Image Search

Google
shopping

PicasaTM

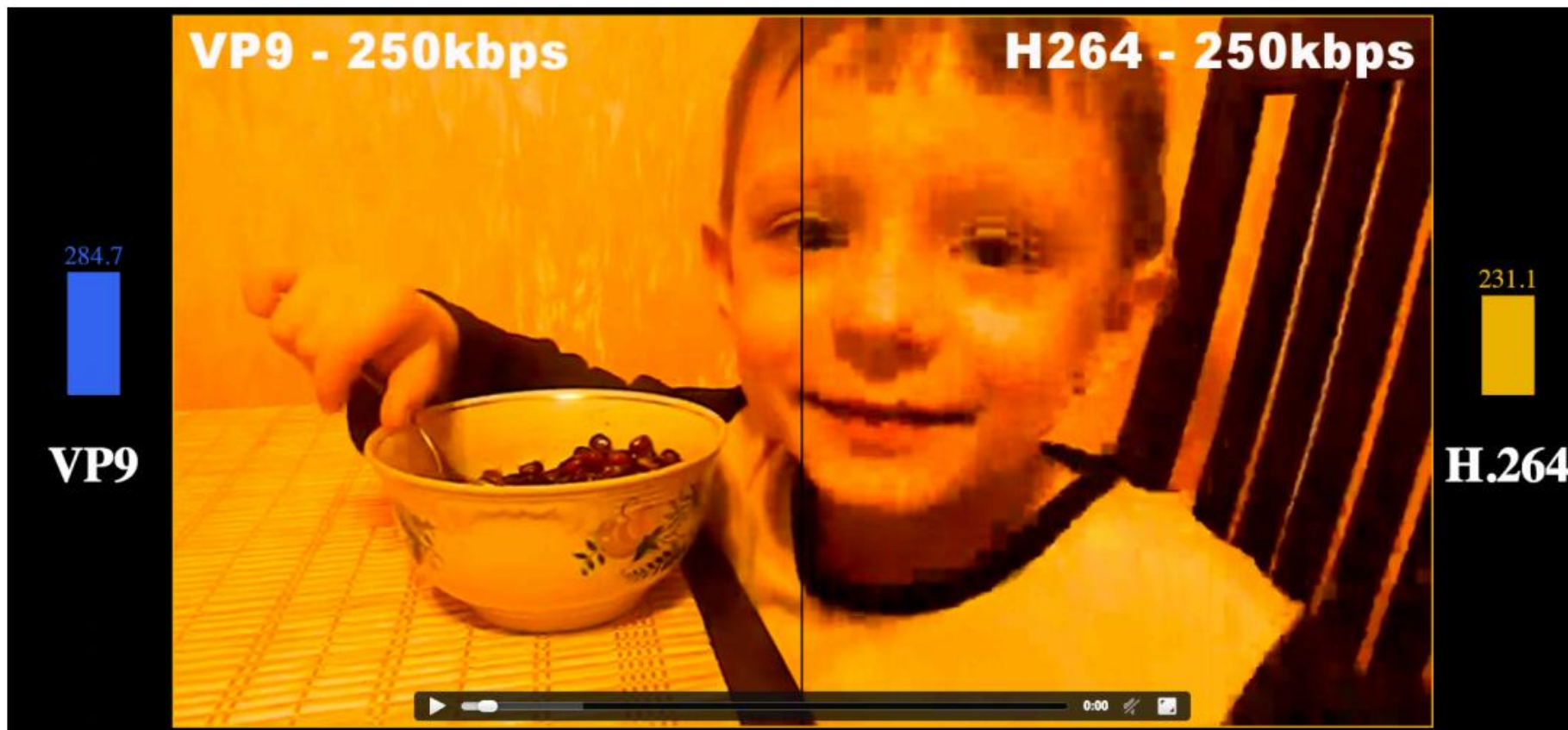


WebM技术是什么

- 包含VP8和VP9的一个开源的、高质量的视频编码、解码、显示技术
- 采用开源的 Vorbis或Opus音频技术
- 使用基于开源的Matroska的容器技术



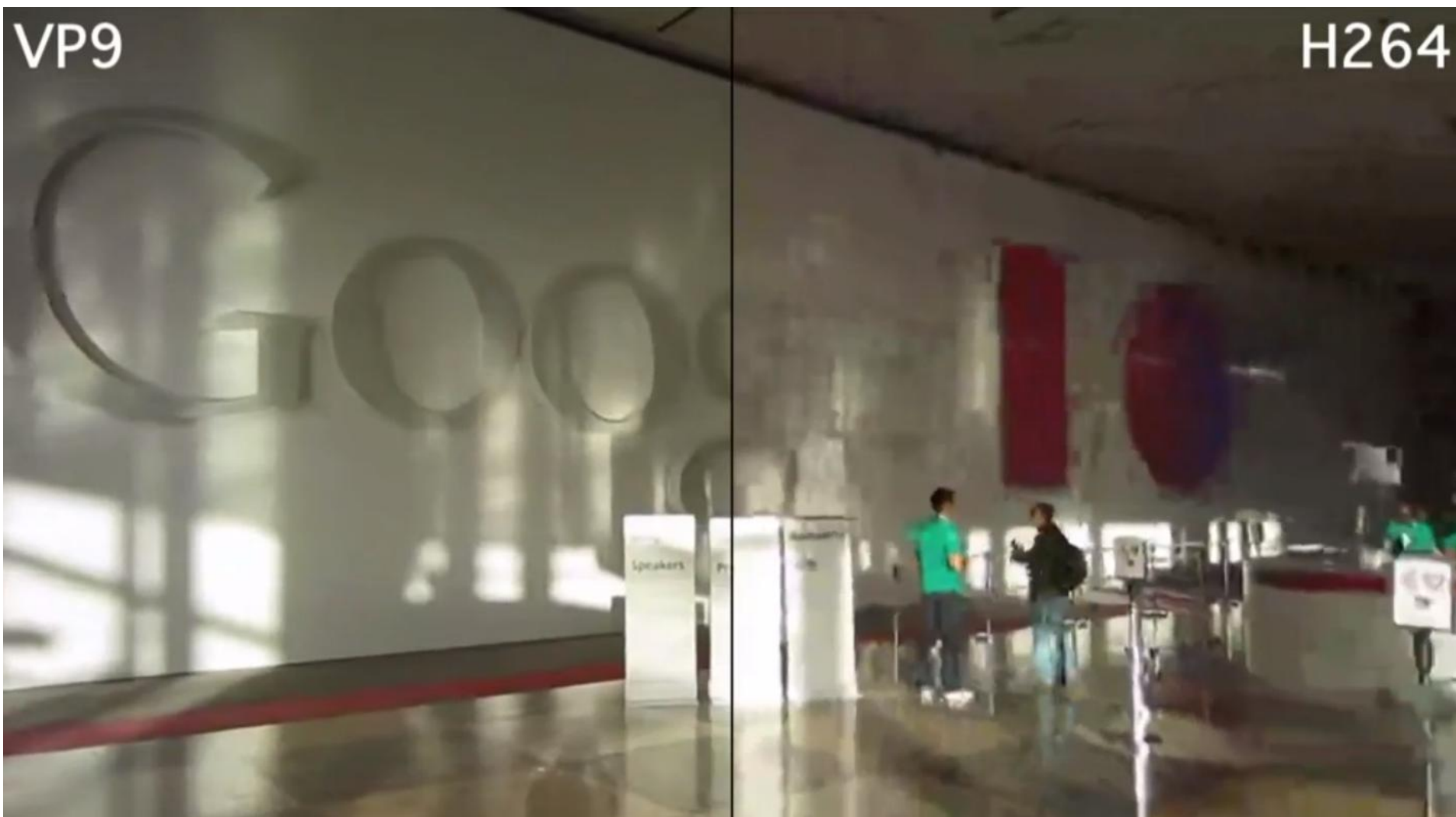
VP9视频的质量和Bitrate与H.264的比较



VP9视频的质量和Bitrate与H.264的比较



VP9视频的质量和Bitrate与H.264的比较



VP9

把你的视频用VP9进行编码和发布

- 使用高质量的视频
- 节省50%的频宽

新的音频技术: Web Audio API

- 新的网络应用开发API让你方便地使用各种音频功能， 从麦克风输入到制成各种声音
- 能够进行动态的声源定位、对声音进行空间化处理、并混合在三维空间中的多个声源
- 支持实时的声音分析、可视化、JavaScript处理
- 提供强大的模块化路由系统和卷积引擎(convolution engine)支持各种音响效果、模拟各种房间回音效果、 发送混音等等

Live microphone input

Web Audio API

```
navigator.getUserMedia({audio: true}, function(stream) {  
  var audioCtx = new window.AudioContext();
```

```
    // Create an AudioNode from the stream.
```

```
    var source = audioCtx.createMediaStreamSource(stream);
```

```
    // TODO: create and connect up filter/analyser/javascript node(s) for processing.
```

```
    // Connect it to the destination to hear yourself.
```

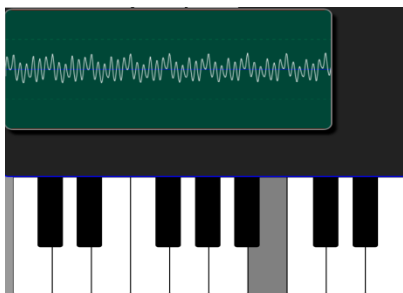
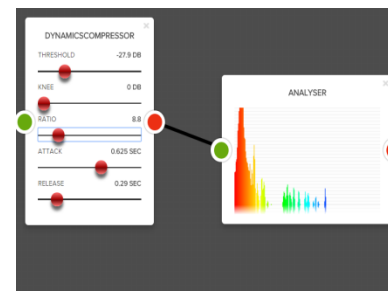
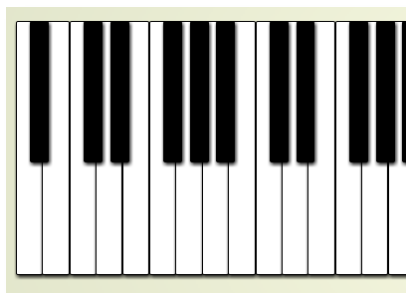
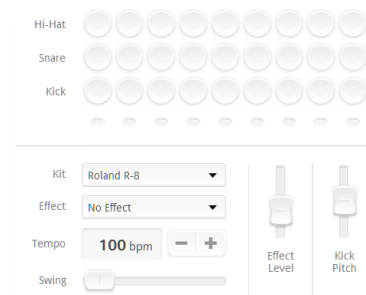
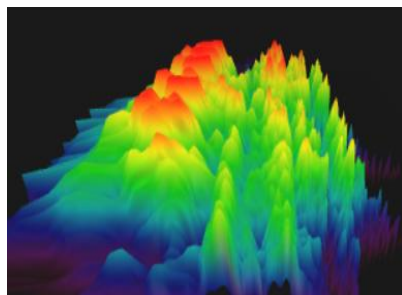
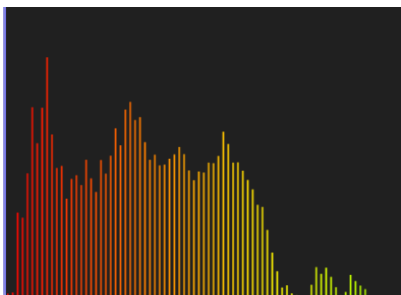
```
    source.connect(audioCtx.destination);
```

```
  }, function(e) {  
    console.log(e);  
  });
```

JS

Web Speech API Demonstration

利用Web Audio API的开发案例



<http://webaudiodemos.appspot.com/>

<http://chromium.googlecode.com/svn/trunk/samples/audio/index.html>

<http://chromium.googlecode.com/svn/trunk/samples/audio/samples.ht>

ml

Web RTC

WebRTC的主要功能特点

- 多媒体信息流 (MediaStreams): 采集和使用客户终端上的多媒体设备信息 – 摄像头、麦克风等
- 点对点设备的多媒体数据连接 (PeerConnection): 方便地进行音频及视频通话
- 数据通道(DataChannels): 应用程序的数据传输

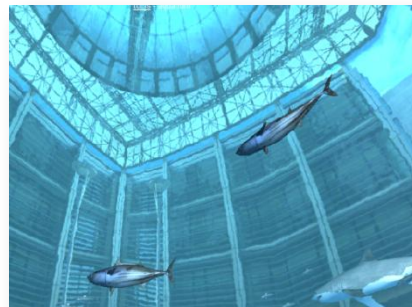
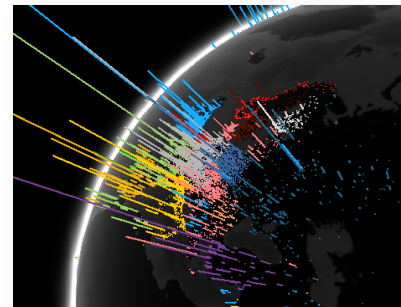
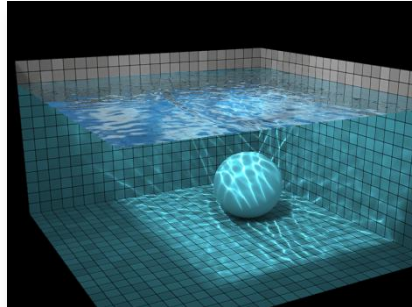
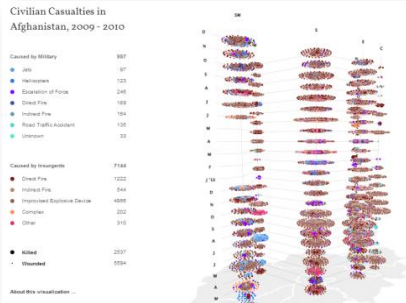
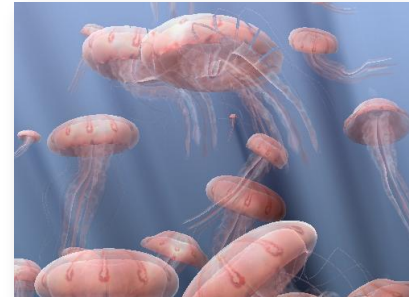
使用getUserMedia API的示范:

```
<script type="text/javascript">
navigator.webkitGetUserMedia({video:true}, onGotStream, onFailedStream);

onGotStream = function(stream) {
  var url = webkitURL.createObjectURL(stream);
  video.src = url;
}
</script>
<video id="video" autoplay="autoplay" />
```

A rectangular button with a light gray background and a thin black border. The word "Start" is centered in a black, sans-serif font.The word "FACEKIT" in a bold, black, sans-serif font. The letters are slightly shadowed, giving it a 3D appearance.

更多HTML5应用开发的案例



更多的HTML5参考资料和学习资源



谷歌开发者网站：

developers.google.com

HTML5ROCKS

[Presentation](#) [Playground](#) [Tutorials](#) [Resources](#)



INTERACTIVE PRESENTATION

Watch the slideshow
that lets you play
with HTML5 features.



CODE PLAYGROUND

Jump headfirst into
HTML5 by
experimenting with
each API.



STEP BY STEP TUTORIALS

Take a guided tour
through code that
uses HTML5
features.

www.html5rocks.com

- www.whatwg.org/html5
- www.chromium.org/developers/web-platform-status
- blog.chromium.org
- diveintohtml5.org
- quirksmode.org

谷歌的开发技术：开放、开源、创新！



- 不断在平台产品上投资、创新、发展
- 支持开源代码、推动创新
- 社区参与和推动技术的发展方向
- 开发产品让开发者们免费使用
- 为大大小小的开发者们创造创业的机会

创新的自由属于我们大家！

The freedom to innovate



Google
Developers



谢谢!

Thank You!

谷歌 开发技术推广部 中国市场主管 栾跃

Bill Luan, China Country Lead, Developer Relations

Google bluan@google.com



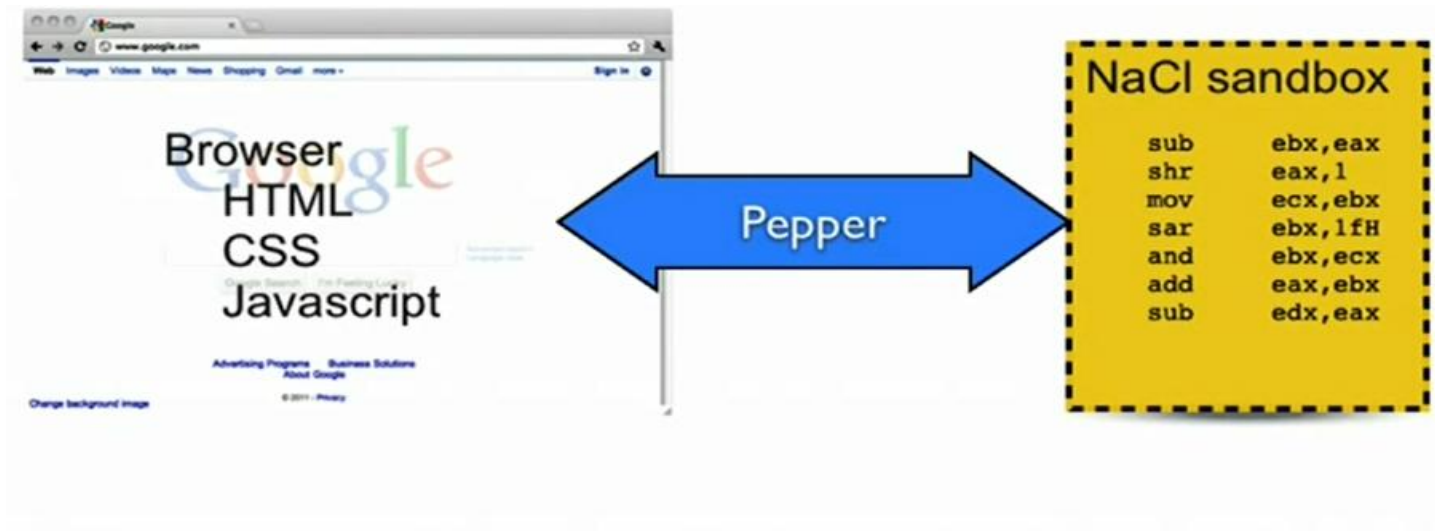
开放 分享 创新

developers.google.com

什么是本地客户端Native Client?

NaCl

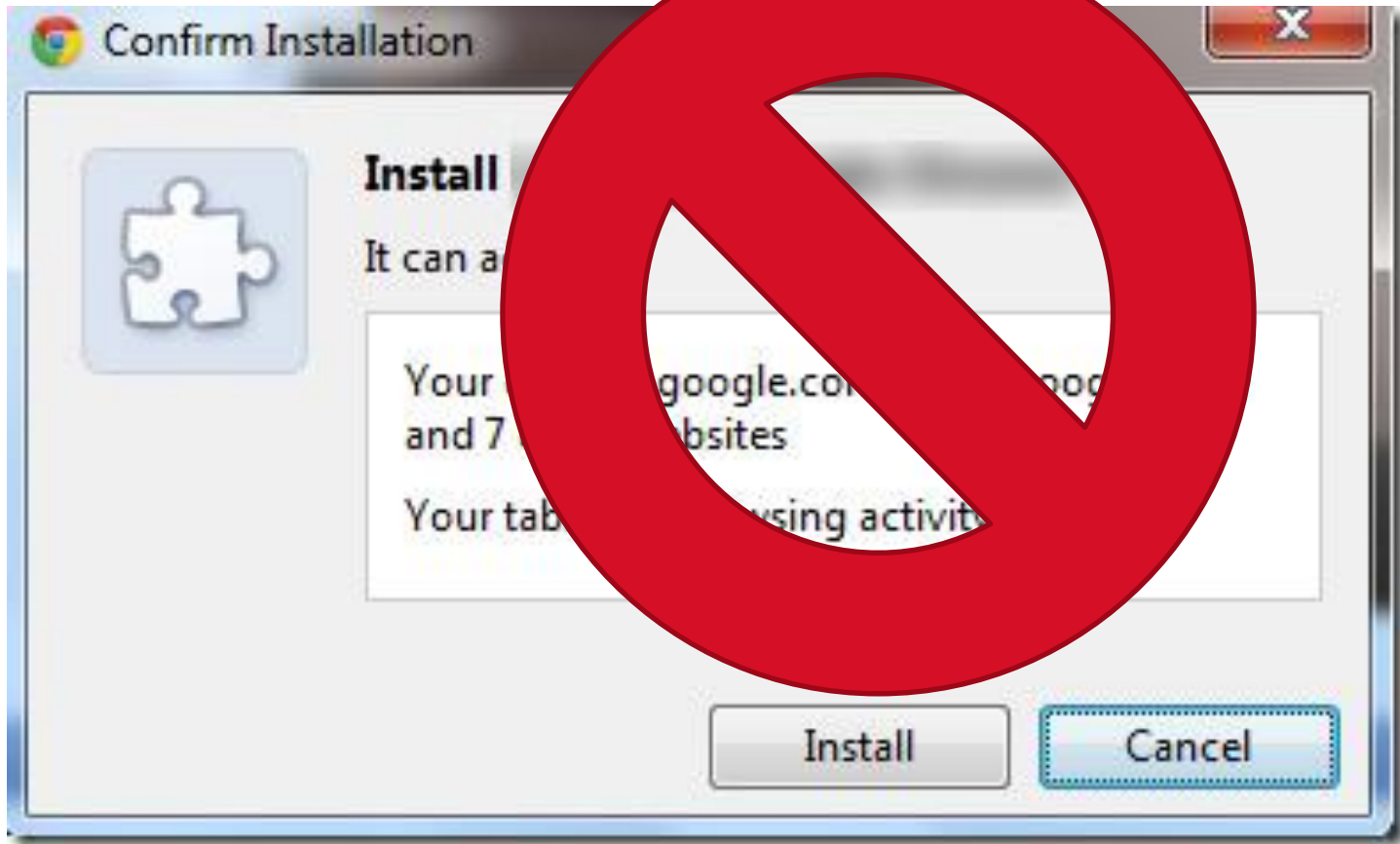
- 它让客户端软件代码在浏览器里运行!
- 它让C/C++代码在Chrome浏览器里运行
 - 采用“沙盘”分割模式(Sandbox)来运行本地客户端软件的代码、像JavaScript那样安全
 - 把网络应用(基于JS)代码的安全优势和本地代码(Native code)的性能优势综合起来



什么是本地客户端Native Client?



Native Client: No Plugins!



为什么应该考虑使用Native Client?

- 开发语言的选择和过去开发代码的投资
 - 从C/C++语言转换到JavaScript的难度
- 运行速度的需要
 - 直接使用CPU和GPU的需要
 - SIMD instruction 和支持Multicore的需要
- 基于浏览器的运行的方便
 - 非常方便的产品发布
 - 完全免去令人头痛的复杂安装过程
 - 面向全球的Chrome Web Store (CWS)所带来的产品发布优势和市场营销优势

Native Client 的优势

- 开发高性能运行的游戏 - 运行速度在传统客户端软件速度的10%相差范围之内
- 方便地移植过去的客户端游戏程序，把它们变为能够在浏览器里运行的游戏
- 不需要任何插件
- 把3D世界带入浏览器：让你开发出在浏览器运行的三维游戏
- 充分利用OpenGL ES 2.0、高质量音响和视频功能
- 不断增加的对游戏开发中间件软件的支持，比如Unity, Havok, FMOD, WWISE, 和Bullet等



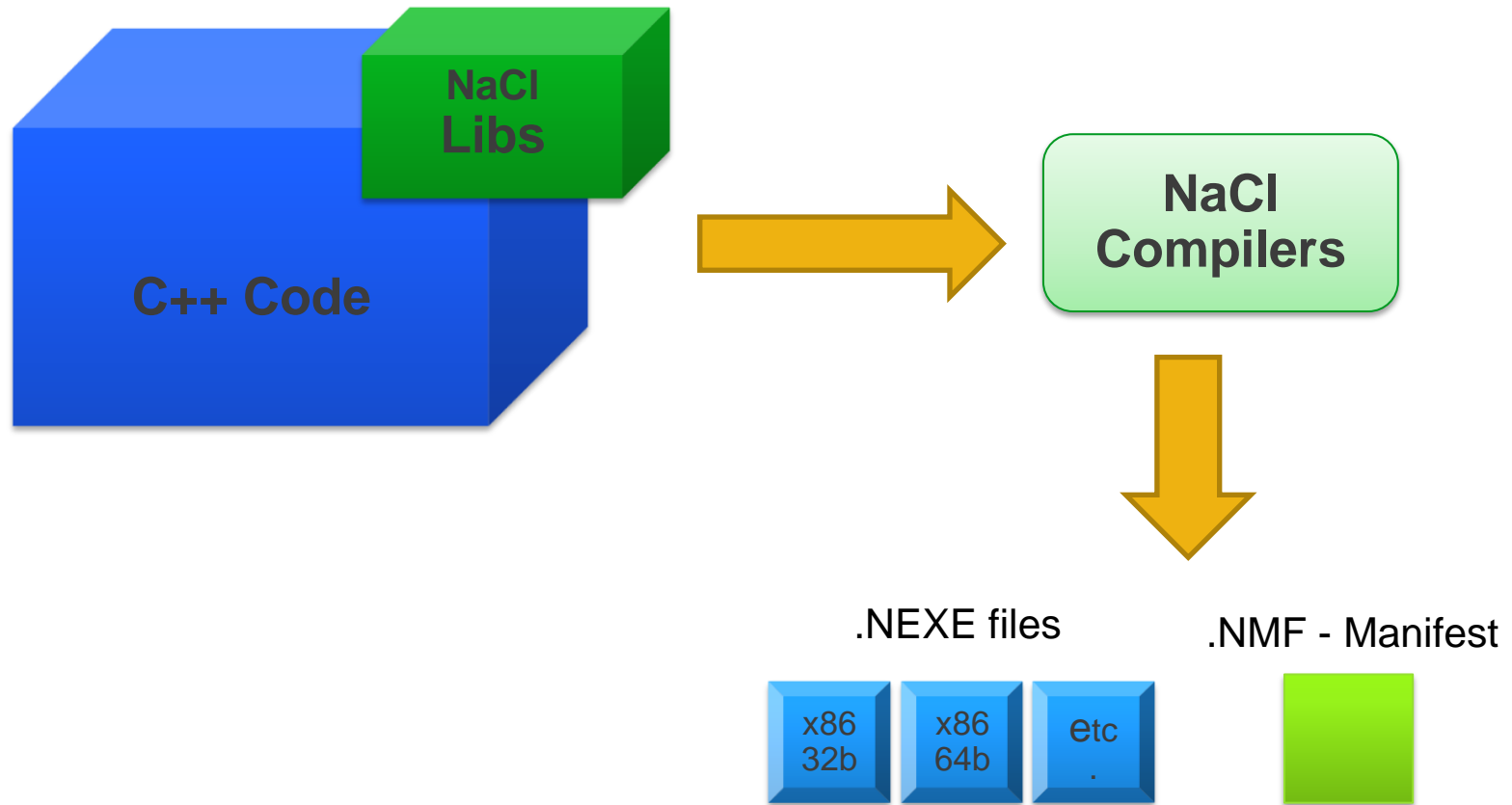
支持NaCl的游戏开发中间层软件



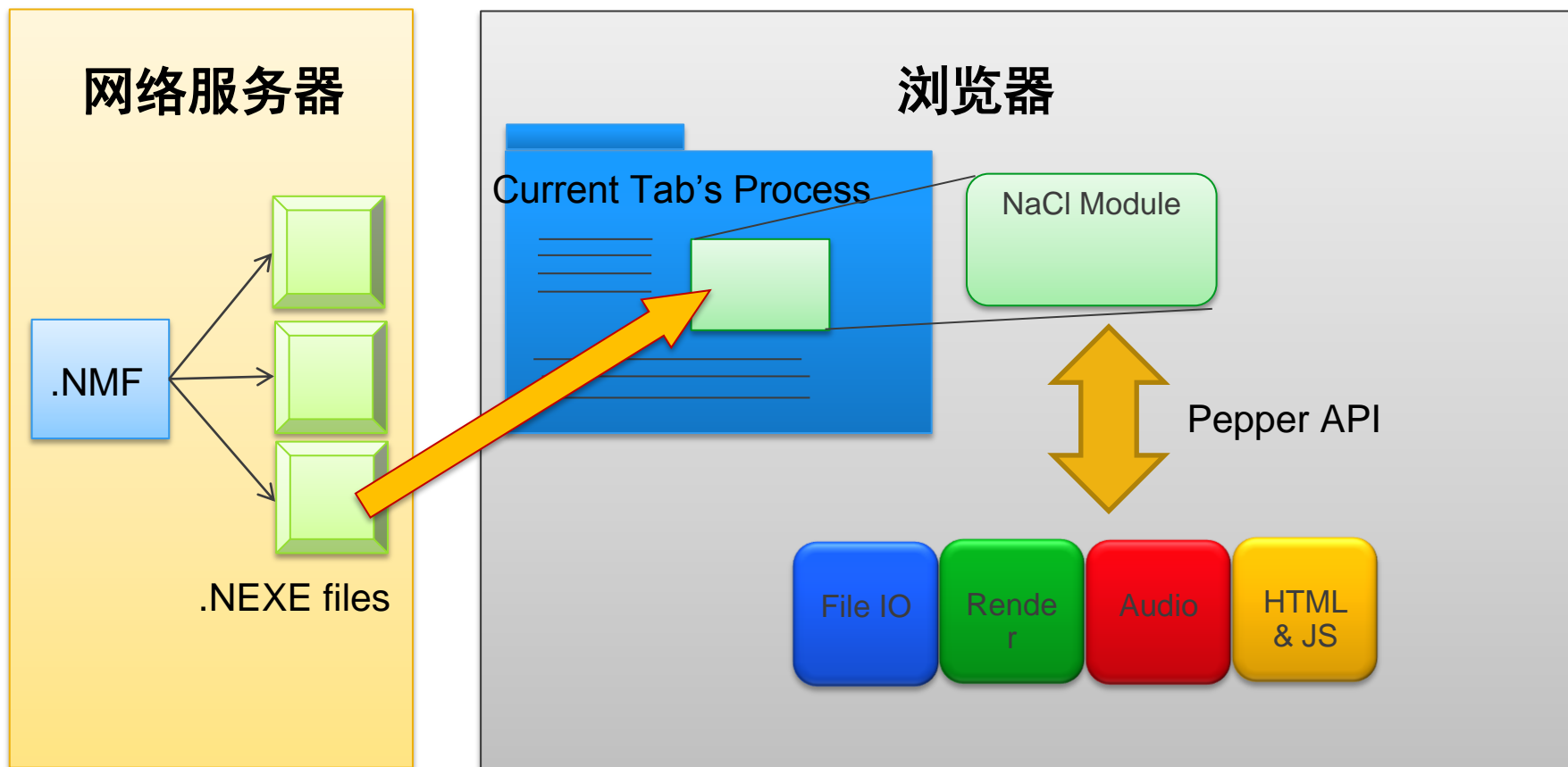
mono™



如何移植C/C++代码到Native Client



Native Client 代码的运行



基于NaCl技术开发的游戏案例和演示



SebTracer simple raytra...



Bullet Physics NaCl Test



Flocking Geese



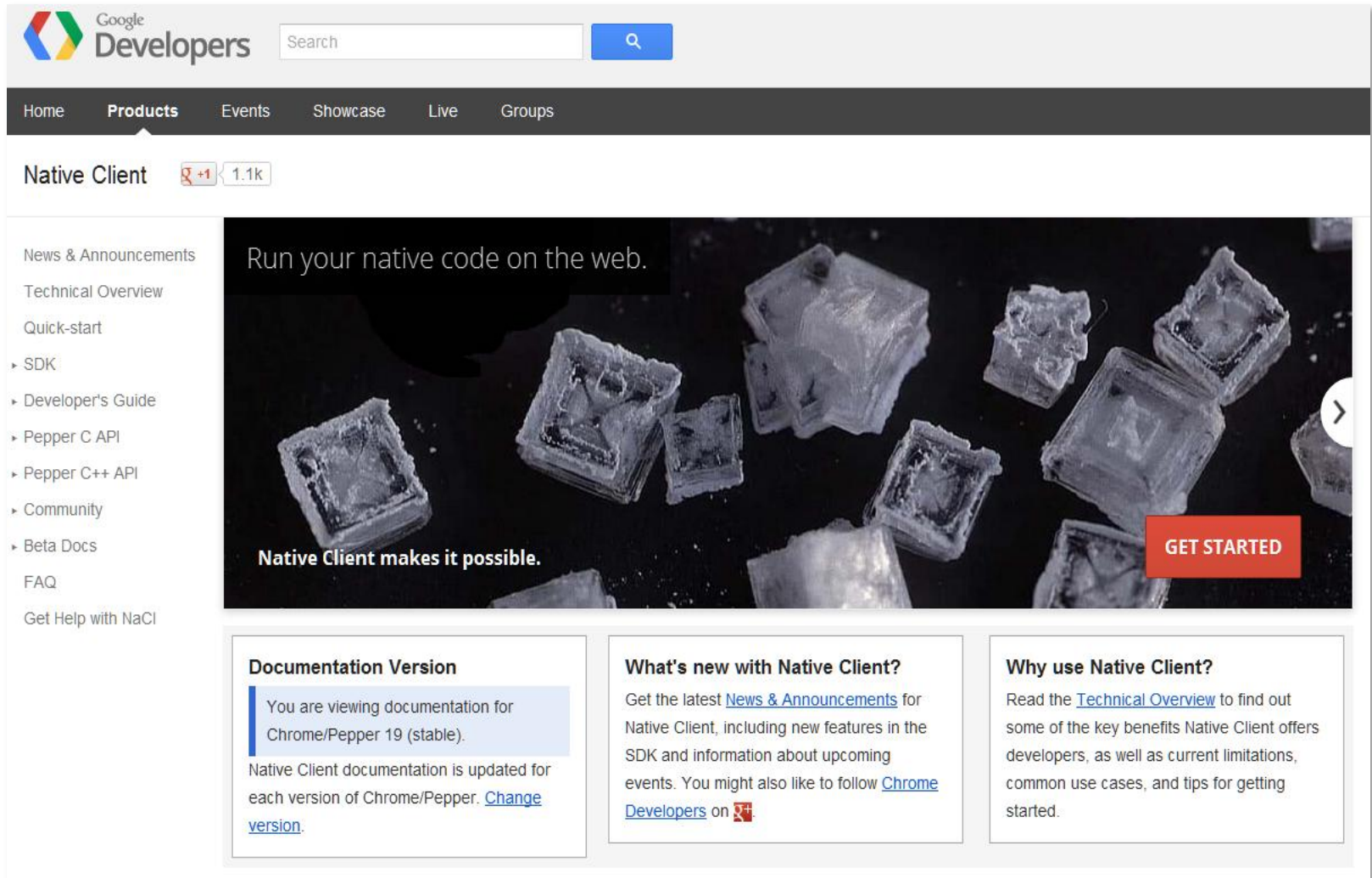
SodaSynth



sixty second shooter

更多的资源：

- developers.google.com/native-client/



The screenshot shows the Google Developers website for Native Client. The header includes the Google Developers logo, a search bar, and navigation links for Home, Products, Events, Showcase, Live, and Groups. The main content area features a large hero image with the text "Run your native code on the web." and "Native Client makes it possible." A red "GET STARTED" button is visible in the bottom right of the hero image. On the left, there is a sidebar with links to News & Announcements, Technical Overview, Quick-start, SDK, Developer's Guide, Pepper C API, Pepper C++ API, Community, Beta Docs, FAQ, and Get Help with NaCl. Below the hero image, there are three columns of content: "Documentation Version" (stating the current version is Chrome/Pepper 19), "What's new with Native Client?" (linking to news and developers), and "Why use Native Client?" (linking to the technical overview).

Google Developers

Search

Home Products Events Showcase Live Groups

Native Client +1 1.1k

News & Announcements
Technical Overview
Quick-start
▶ SDK
▶ Developer's Guide
▶ Pepper C API
▶ Pepper C++ API
▶ Community
▶ Beta Docs
FAQ
Get Help with NaCl

Run your native code on the web.

Native Client makes it possible.


GET STARTED

Documentation Version

You are viewing documentation for Chrome/Pepper 19 (stable).

Native Client documentation is updated for each version of Chrome/Pepper. [Change version.](#)

What's new with Native Client?

Get the latest [News & Announcements](#) for Native Client, including new features in the SDK and information about upcoming events. You might also like to follow [Chrome Developers](#) on .

Why use Native Client?

Read the [Technical Overview](#) to find out some of the key benefits Native Client offers developers, as well as current limitations, common use cases, and tips for getting started.

推广基于浏览器应用的好处



- 让千千万万的用户在任何设备(电脑和手机)上及任何地点享用你的开发成果和服务
 - 例如你的网页可以同时被桌面计算机和手机的用户所使用
- 充分利用先进浏览器的性能的提高
 - 例如Chrome数倍的JavaScript运行速度的提高、WebGL技术直接使用硬件图像加速的功能等
- 充分利用先进浏览器中的各种开发工具
 - 例如利用Chrome里面debugging的功能