

hello world



```
{  
  name: '姜维',  
  nick : '玄寂',  
  company : '一淘'  
}
```

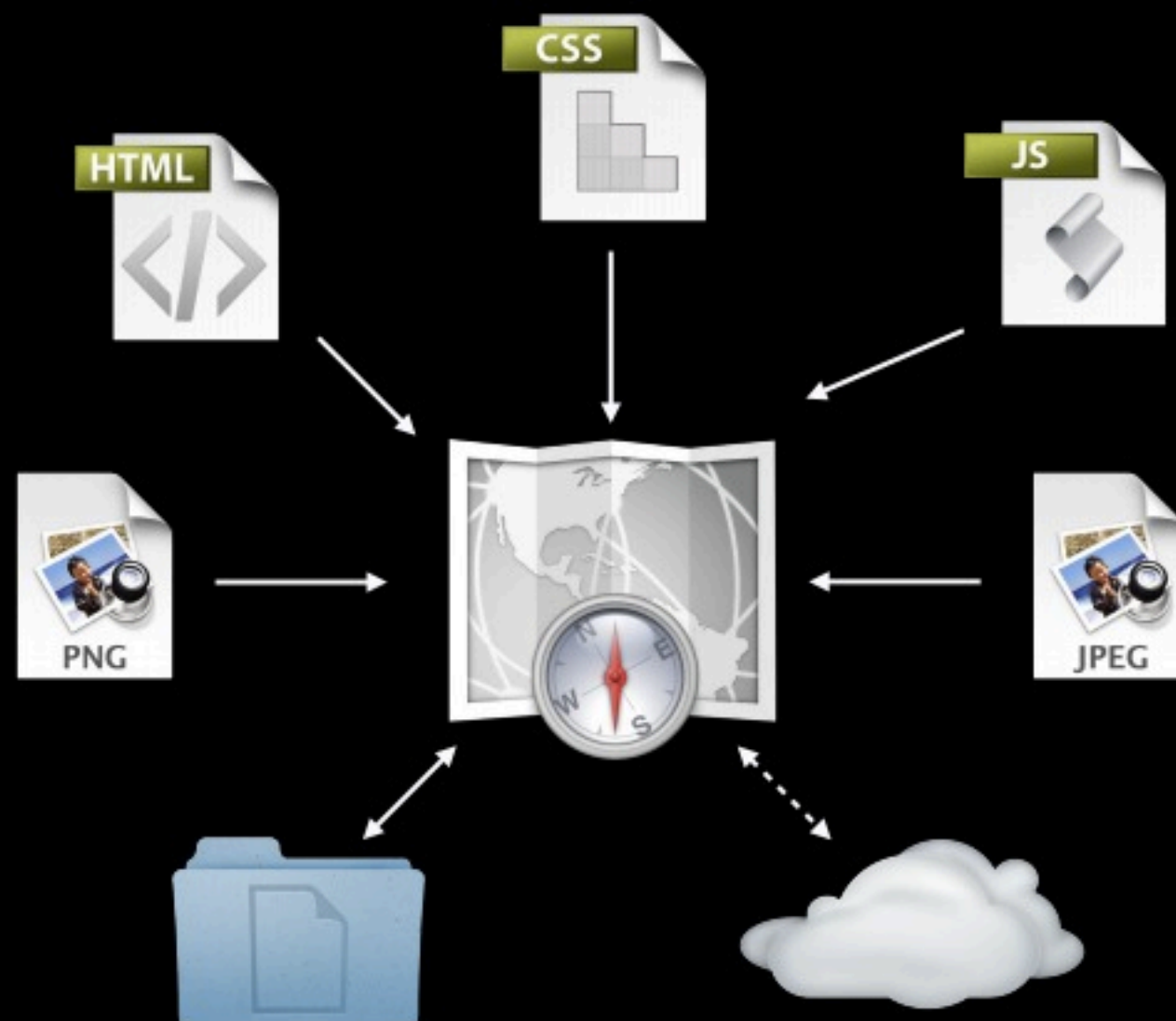


applicationCache

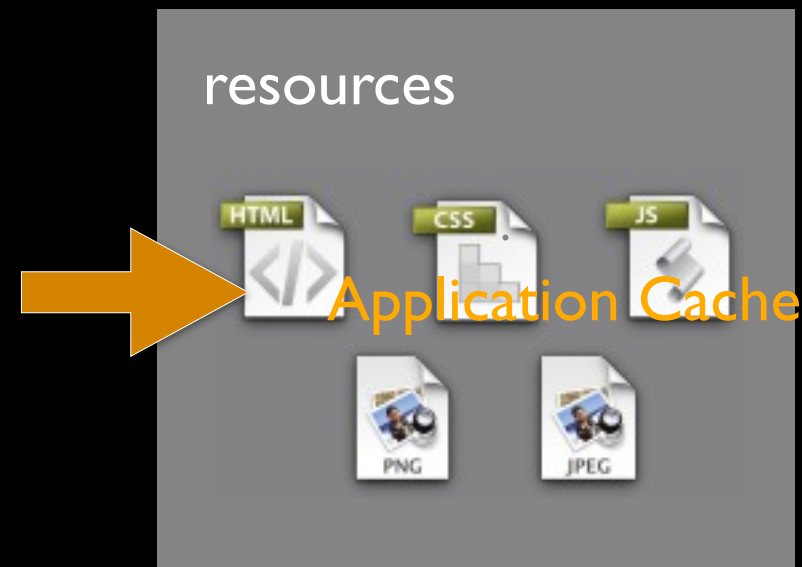
1. applicationCache

2. H5项目实践

一个网络应用需要什么呢



Application 资源文件和数据



离线存储把HTML、JavaScript、CSS 以及一些媒体文件存储到本地。

offline data

- Local Storage
- SQL Database



the cloud

- XMLHttpRequest



实现目标

- 加快站点加载速度
- 解决网络的不给力
- 快速实现部署

当前应用的情形

- 基于canvas的游戏，网络阅读器（杂志等），js应用（计算器等）



Application Cache

- index.html

```
<!DOCTYPE html>
```

```
<html manifest="application.manifest">
```

```
<head>
```

```
..
```

```
</html manifest="application.manifest">
```

Manifest到底是神马玩意儿

CACHE MANIFEST

This is a comment.

Cache manifest version 0.0.1

If you change the version number in this comment,

the cache manifest is no longer byte-for-byte

identical.

demoimages/clownfish.jpg

demoimages/clownfishsmall.jpg

CACHE:

Additional items to cache.

demoimages/stonessmall.jpg

NETWORK:

All URLs that start with the following lines

are whitelisted.

<http://example.com/examplepath/>

<http://www.example.org/otherexamplepath/>

FALLBACK:

demoimages/currentImg.jpg images/stockImage.jpg

<html manifest="path/filename">

- 必须指明manifest字串以及路径，manifest作为html的一个属性
- 站点离线存储容量限制为 **5MB**
- 如果manifest文件或者其内部所列的某一个文件不能正常下载，那么整个应用的更新将被视为失败，而浏览器将继续使用老的缓存。
- 缓存的更新必须建立在修改manifest文件的基础上。

创建一个manifest文件

- 服务端须指明manifest的媒体类型（MIME） => `text/cache-manifest`
- 在HTML中声明的manifest文件，必须与其 同源 (domain and scheme)。
- 首行文本必须为： `CACHE MANIFEST`。
- 随后可以包含如下section：
 - ◆ `headings`，需cache的资源文件地址（Resource URLs），需网络访问的资源地址（Network URLs），`comments`。
 - ◆ `headings`的合法值有：CACHE、NETWORK、FALLBACK；
 - ◆ 资源文件地址（Resource URLs）可以是绝对路径或者也可以是相对于manifest文件的相对路径；
 - ◆ Network URLs 可以是一个“*”号通配符
 - ◆ 注释可以出现在任何一个section中，但其必须单独一行并以“#打头”

服务端配置示例

Mac OS

```
cd /  
cd etc/apache2  
sudo vi mime.types
```

```
pigcan-2:apache2 pigcan$ cd /  
pigcan-2:/ pigcan$ cd etc/apache2  
pigcan-2:apache2 pigcan$ sudo vi mime.types
```

```
text/cache-manifest manifest
```

```
sudo apachectl restart
```

```
defaults write com.apple.finder AppleShowAllFiles -bool true  
KillAll Finder
```

headings

CACHE:

默认情况下CACHE都会在起始处被隐式声明

但是在存在别的header且位于他们之后则必须显示声明！

NETWORK:

NETWORK上所列文件都为网络白名单。

可以针对某一个文件，某一个文件夹，某一个域，或者是通配符“*”

FALLBACK:

所有 FALLBACK URLs 必须与 manifest 文件同源。

Note: 引用manifest的HTML不管你怎么操作，其都将被缓存。

如何更新缓存?

- ▶ 用户手动清除了浏览器缓存。

- ▶ manifest文件发生更改。

注: 更新了manifest中所列的文件的更新并不意味着浏览器会重新缓存该资源文件, 要触发浏览器的更新必须是manifest文本本身发生了改变。

- ▶ 程序的方式更新应用程序的缓存。

► 程序的方式更新应用程序的缓存

- 自定义UI DOM事件

checking	客户端尝试更新，或者首次访问时试图下载manifest文件
noupdate	manifest文件未发生任何更改
downloading	客户端查询到更新并成功获取到，或者首次访问时资源文件下载ing
progress	客户端下载资源文件ing
cached	manifest所列文件全部被下载好，并被成功缓存
updateready	manifest所列文件已经全部被重新下载好，同时脚本中可以使用方法swapCache()来更新缓存
obsolete	manifest被发现已经更改为404或者410页面
error	manifest是一个404或者410的页面，缓存过程被终止 manifest没有发生改变，但是引用它的页面并没有被正常下载 在获取manifest所列文件时发生致命错误 在更新过程中manifest文件又再次发生更改

<http://dev.w3.org/html5/spec/offline.html>

Cache states (cache状态值)

- 自定义UI DOM事件

checking	状态为2，更新开始，逐字节比对manifest文件
noupdate	状态为1，更新处理完成
downloading	状态为3，表示开始下载最新的资源
progress	
cached	状态为4，表示缓存更新已处理完成并已存储
updateready	状态为4，表示缓存更新已处理完成并已可用
obsolete	状态为5， application cache group obsolete
error	状态为0，表示有错误发生， manifest文件找不到或者资源找不到

iOS - > <http://is.gd/oQIePC>

重要

*Note: These events are delayed until after the **load** event has fired.*

浏览器检测到<html>元素含有
manifest属性

调用checking事件

从来没有访问过

调用downloading事件(之后便开始下载资源文件)

调用progress事件(告知多少资源文件被下载好同时告知还有多少资源文件有待下载)

调用cached事件

之前访问过

距离最近一次访问时缓存的manifest文件与当前访问的manifest文件是否有发生更改?

no

调用nouupdate事件

notice : HTTP CACHE

yes

调用downloading事件(重新下载每个资源文件)

调用progress事件(告知多少资源文件被下载好同时告知还有多少资源文件有待下载)

调用updateready事件(if (appCache.status == appCache.UPDATEREADY)
{window.applicationCache.swapCache()可进行缓存更新})

可能会有一些疑问

web应用超过一张页面，那manifest属性只需要加在home page上呢，还是全部需要加上？

需要在manifest文件里面罗列所有web应用页面么？

移动浏览器支援情况



android 2.1



ios 2.2



opera 11

tips - offline Applications for iOS

```
<meta name="apple-mobile-web-app-capable" content="yes">
```

//全屏

```
<meta name="apple-mobile-web-app-status-bar-style" content="black">
```

//状态栏

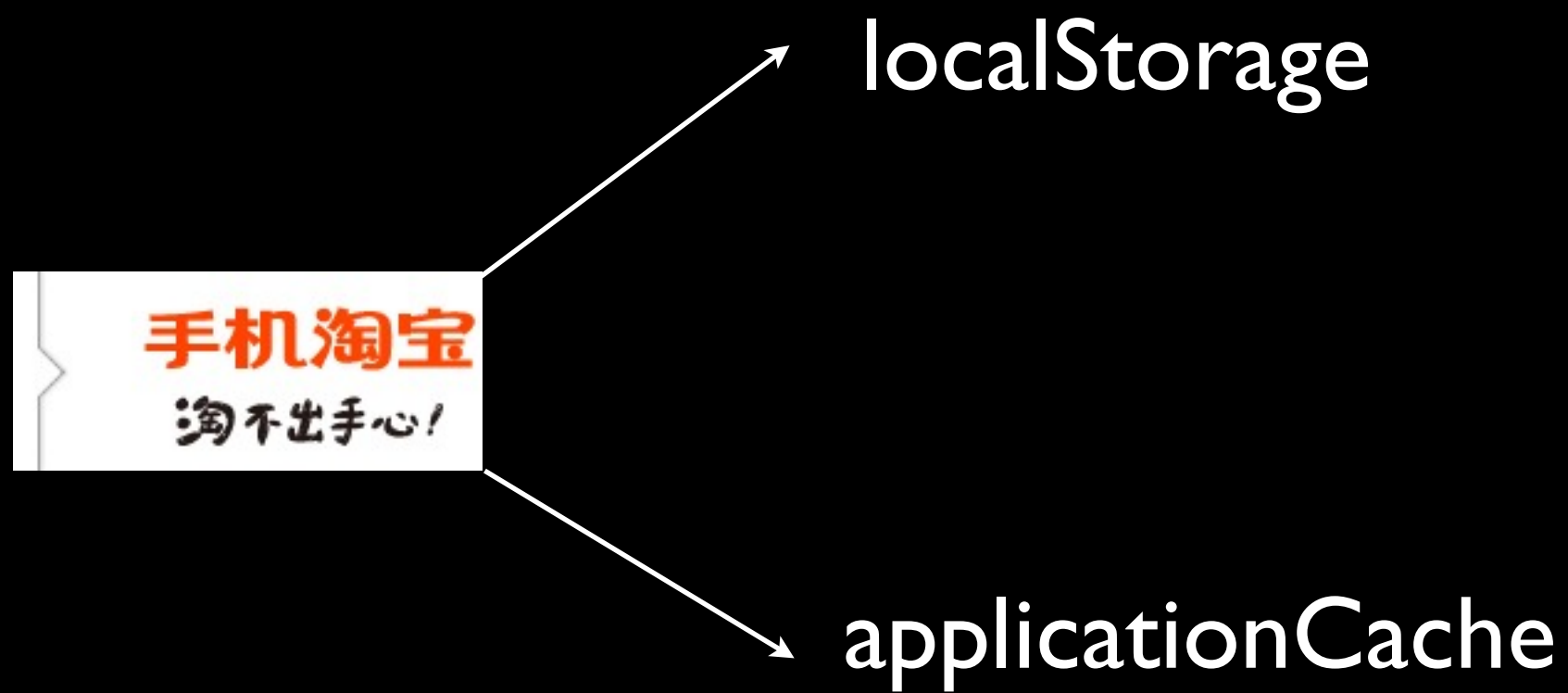
```
<link rel="apple-touch-icon" href="/custom_icon.png">
```

//桌面图标

```
<link rel="apple-touch-startup-image" href="/startup.png">
```

//启动画面





localStorage

网页上只加载一小段js，该js用来监测真实环境js状态（指明是否被成功存储）

如果没有被存储，那么js将被异步请求，并存储

如果成功存储，则直接从localStorage中调用。（localStorage只能存储字符串，所以这边还需要做特殊处理）

其实意在解决applicationCache不能单点更新问题

另外关于localStorage的测试结果为：主域与子域之间存储信息完全各自独立，不能相互访问！另外主域和各个子域均享有独立的存储空间，理论值为各自拥有5MB存储空间。

缺点： 请求

localStorage + cookie

情况一：没有与之相关的cookie记录，判断为用户首次访问：则接下来资源文件形式则以变量 - 字符串 形式输出到页面。

情况二：存在含有版本信息的cookie记录，提交服务器与之对比之后发现无改变，则输出页面，前端自动调用存储在localStorage的资源文件。

情况三：存在含有版本信息的cookie记录，提交服务器与之对比之后发现无改变，但是用户清除了缓存，也就是本地已经不存在相关资源文件存储信息，但此时页面根据cookie记录的信息也已经下来了没有相关脚本的html页面。此时就需要异步有序加载资源文件。为什么是有序呢，因为js完全可能存在依赖关系。

情况四：存在含有版本信息的cookie记录，提交服务器与之对比之后发现有改变，后端在页面输出的时候以变量字符串形式输出到页面，随后前端进行存储。

情况五：在一切都顺利但是比较极端的情况是，本地存储空间耗尽，此时需要单独发送请求，请求相关资源文件，予以外链支持。

崩溃.....

applicationCache + localStorage



html、js、css、ttf、jpg、
png、media

ajax - data

高枕无忧了？

集群、兼容性

兼容性

android：用seajs 模块化 applicationCache 组件，不能运行(部分安卓机型)。

android：manifest更新缓存依旧不更新 - 时间戳

当然还有更加恐怖的UC、QQ等

集群

举例：

我们无线服务器，监测到 静态资源服务器 某一台服务器 资源内容已经发生改变，然后马上生成新的manifest文件开始推送，此时用户访问，用户拿到新的manifest文件，然后发生本地更新，但是静态服务器，并没有完全发布完，此时他到了一台并没有发布好的静态服务器去获取静态资源文件，那么这个用户就悲剧了，他的文件永远也更新不了了！！！！

策略服务器

祝： 屌丝并快乐着。

谢谢。