2012 阿里技术沙龙

享技术·聚朋友

D2

前端性能之精确度量

aDev

iDevOps

iData TCon

About Me

花名：阿大 和我联系　　姓名：朱华军

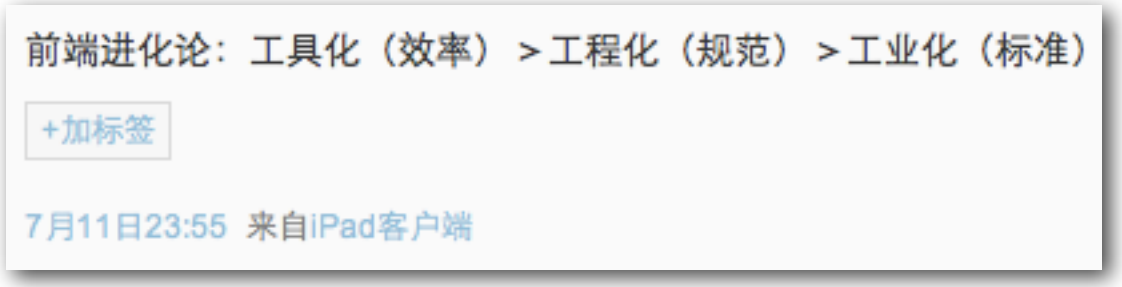
- 2009 5.4 加入淘宝，负责淘宝旅行和彩票业务
- 目前在一淘－UX团队担任一淘网站的前端并负责前端工具化的研究和推广工作

http://weibo.com/realada　http://twitter.com/zhj3618　zhj3618@gmail.com

About Me

花名：阿大 和我联系　　姓名：朱华军

- 2009 5.4 加入淘宝，负责淘宝旅行和彩票业务
- 目前在一淘－UX团队担任一淘网站的前端并负责前端工具化的研究和推广工作



http://weibo.com/realada　http://twitter.com/zhj3618　zhj3618@gmail.com

About My Team

一淘 – UX 网站组前端团队：

- 阿里集团内部最早使用Git进行生产环境代码管理的前端团队（没有之一）
  - Git仓库代码在线浏览和分享平台
  - Git仓库管理在线维护管理系统
  - 拥有基于Git Hook的前端代码自动部署和发布系统及完善的部署日志统计和查询系统

- 拥有完善的前端Demo交付环境
  - 前端demo自动部署
  - 前端demo实时截屏

About My Team

一淘首页

http://demo.etao.net/ehome

写入地址 | 只读地址 | git@git.etao.net:ehome.git

master | Tree | Commits | Branches 12 | Tags 29

Latest commit to the master branch

调整集分宝样式

commit b232924f49 [墨峰] 2012-11-15

ehome.git /

Name	Update	Message	history
api	*2012-07-16*	[墨峰] 首页推荐排行榜	
assets	*2012-09-20*	[墨峰] 添加font文件	
src	*7 days ago*	[墨峰] 调整集分宝样式	
template	*7 days ago*	[墨峰] 调整集分宝样式	
.gitmodules	*2012-08-24*	[gitServer] Auto Update "ecommon" Remote URL	
build.xml	*2011-12-27*	[yuli] build up to v3.0	

About My Team

master | Tree | Commits | Branches 12 | Tags 29

Diff selected Commits

2012.11.15

调整集分宝样式 master publish/121112 — Select
[墨峰] 2012-11-15 12:11:16 commit:b232924f49 — Tree

调整签到后按钮的宽度 — Select
[墨峰] 2012-11-15 11:11:41 commit:3bf9ca7957 — Tree

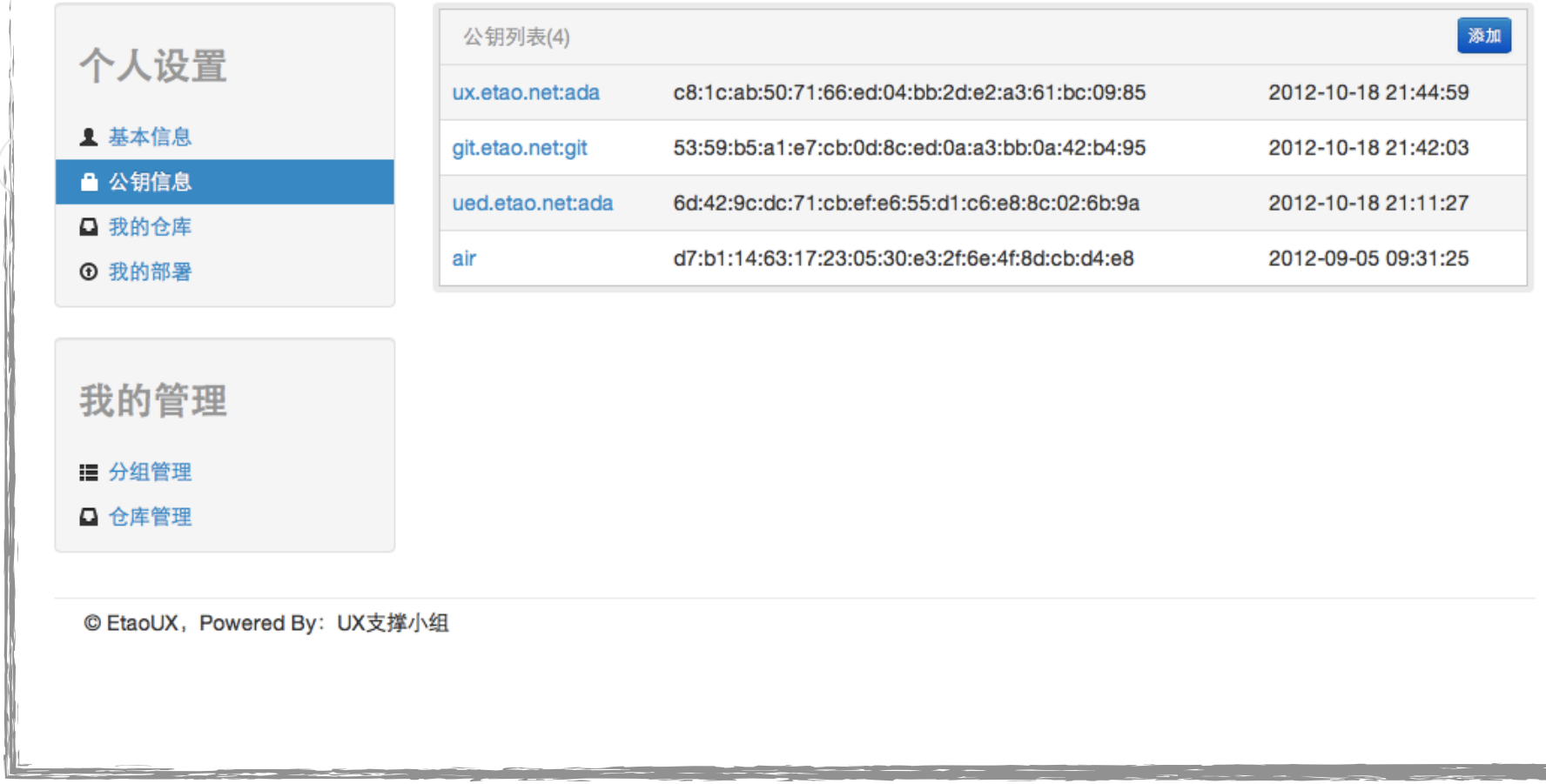
调整签到按钮的宽度 — Select
[墨峰] 2012-11-15 11:11:20 commit:60073ceb68 — Tree

调整积分显示的位置 — Select
[墨峰] 2012-11-15 10:11:44 commit:2293643e32 — Tree

2012.11.13

替换css sprite — Select
[墨峰] 2012-11-13 14:11:39 commit:de22c05d0d — Tree

About My Team



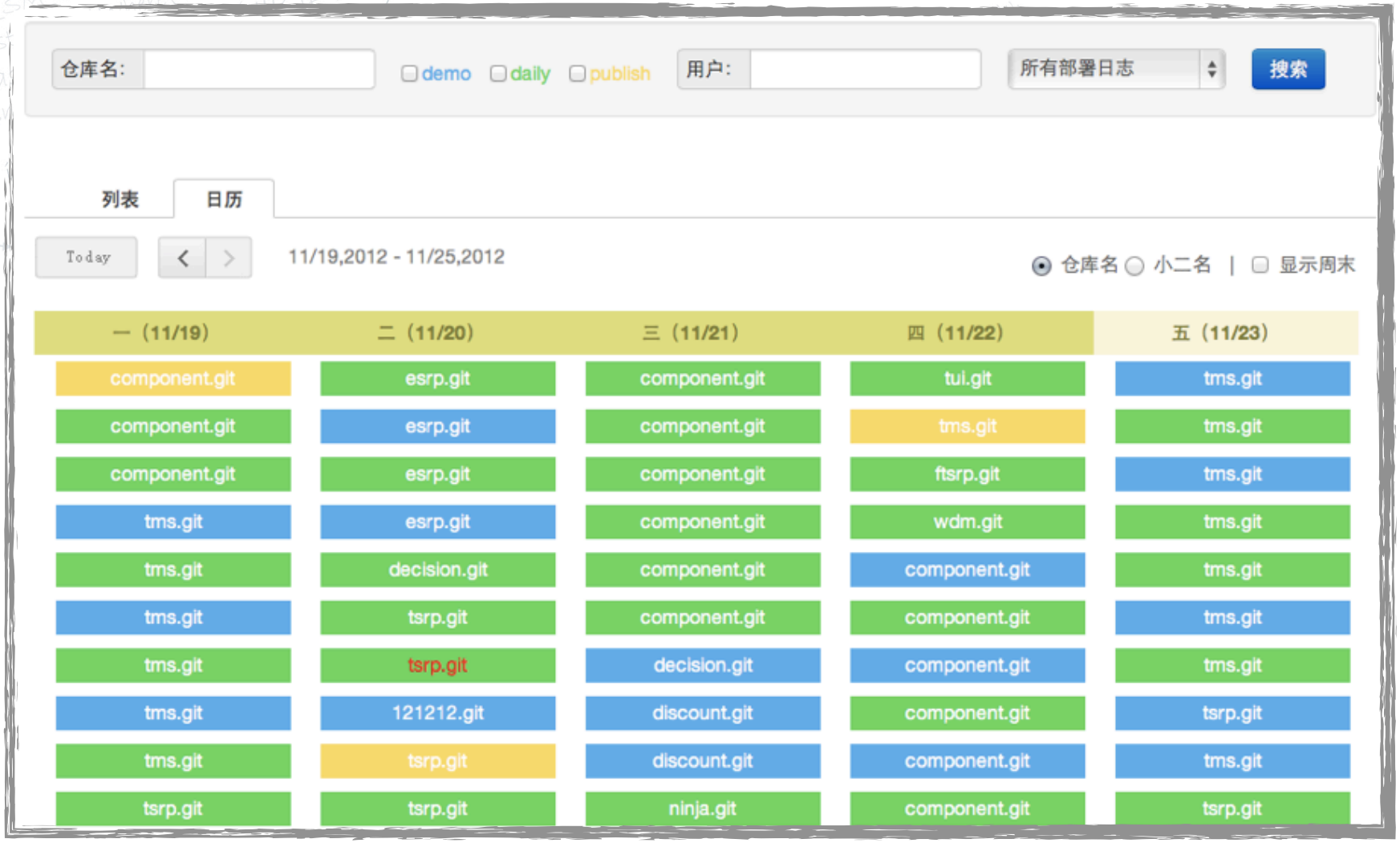
About My Team

About My Team

About My Team


```
zhj3618@air tsrp (demo) $ git push origin demo
Counting objects: 19, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 741 bytes, done.
Total 7 (delta 6), reused 2 (delta 1)
remote: 准备同步demo...
remote: 清理demo环境 [Success]
remote: 更新demo [Success]
remote: demo访问路径: http://demo.etao.net/tsrp
To git@git.etao.net:tsrp.git
 * [new branch]      demo -> demo
zhj3618@air tsrp (demo) $
```

About My Team



About My Team

ehome

一淘首页

查看Demo 代码仓库

epromotion

巨优惠外投页面

查看Demo 代码仓库

eseller

开放搜索商家入驻注册页面

查看Demo 代码仓库

esrp

etop

etuan

About My Team

WE WANT YOU

ada@taobao.com

回到主题

前端性能之精确度量

思考

如何用JS统计打开一个网页所用的时间

传统方式

```
<script type="text/javascript">
var t0 = new Date().getTime();
window.onload = function(){
    var now = new Date().getTime();
    var latency = now - t0;
    alert("page loading time: " + latency);
}
</script>
```

传统方式

```
<script type="text/javascript">
var t0 = new Date().getTime();
window.onload = function(){
    var now = new Date().getTime();
    var latency = now - t0;
    alert("page loading time: " + latency);
}
</script>
```

缺点：？

没有包含数据从后端返回到浏览器所消耗的时间

传统方式二

```
<script type="text/javascript">
var t0 = 1351323103268 ; // 后端输出的服务器时间戳
window.onload = function(){
    var now = new Date().getTime();
    var latency = now - t0;
    alert("page loading time: " + latency);
}
</script>
```

传统方式二

```
<script type="text/javascript">
var t0 = 1351323103268 ; // 后端输出的服务器时间戳
window.onload = function(){
    var now = new Date().getTime();
    var latency = now - t0;
    alert("page loading time: " + latency);
}
</script>
```

缺点：？

前后端需要进行时间同步修正，处理复杂

仍然无法统计到后端返回内容前所消耗的时间

传统方式三


```
<!-- page 1 -->
<script type="text/javascript">
window.onbeforeunload = function(){
    var t0 = new Date().getTime();
    // 将离开page1页面的时间写入cookie
    KISSY.Cookie.set("t0", t0);
}
</script>
```

```
<!-- page 2 -->
<script type="text/javascript">
// 从cookie获取page1页面离开时纪录的时间
var t0 = KISSY.Cookie.get("t0");
window.onload = function(){
    var now = new Date().getTime();
    var latency = now - t0;
    alert("page loading time: " + latency);
}
</script>
```

传统方式三

```
<!-- page 1 -->
<script type="text/javascript">
window.onbeforeunload = function(){
    var t0 = new Date().getTime();
    // 将离开page1页面的时间写入cookie
    KISSY.Cookie.set("t0", t0);
}
</script>
```



```
<!-- page 2 -->
<script type="text/javascript">
// 从cookie获取page1页面离开时纪录的时间
var t0 = KISSY.Cookie.get("t0");
window.onload = function(){
    var now = new Date().getTime();
    var latency = now - t0;
    alert("page loading time: " + latency);
}
</script>
```

传统方式三

```
<!-- page 1 -->
<script type="text/javascript">
window.onbeforeunload = function(){
    var t0 = new Date().getTime();
    // 将离开page1页面的时间写入cookie
    KISSY.Cookie.set("t0", t0);
}
</script>
```

```
<!-- page 2 -->
<script type="text/javascript">
// 从cookie获取page1页面离开时纪录的时间
var t0 = KISSY.Cookie.get("t0");
window.onload = function(){
    var now = new Date().getTime();
    var latency = now - t0;
    alert("page loading time: " + latency);
}
</script>
```

缺点：？

页面间关联依赖，部署和实现复杂

无法统计直接进入的页面

只有一个全局的时间，没有各个阶段的精确统计

统计在页面加在过程中进行，脚本的执行本身影响统计结果

W3C Navigation Timing API

http://www.w3.org/TR/navigation-timing/

https://developer.mozilla.org/en-US/docs/API/navigationTiming

http://msdn.microsoft.com/en-us/library/ff974680(v=vs.85).aspx

W3C Navigation Timing API

定义了在浏览器中一个与页面时间信息相关的JavaScript接口：

PerformanceNavigation

window.performance.navigation

PerformanceTiming

window.performance.timing

W3C Navigation Timing API

window.performance.navigation

window.performance.timing

W3C Navigation Timing API

window.performance.navigation window.performance.timing

```
>>> console.dir(window.performance)
▼ navigation                    PerformanceNavigation { constructor=F
    TYPE_BACK_FORWARD           2
    TYPE_NAVIGATE               0
    TYPE_RELOAD                 1
    TYPE_RESERVED               255
    redirectCount               0
    type                        1
  ▶ __proto__                   [xpconnect wrapped native prototype] {
▶ timing                        PerformanceTiming { constructor=Perf
```

W3C Navigation Timing API

window.performance.navigation window.performance.timing

```
>>> console.dir(window.performance)
▶ navigation                    PerformanceNavigation { constructor=Perfo
▼ timing                        PerformanceTiming { constructor=Performa
    connectEnd                  1351410372171
    connectStart                1351410372171
    domComplete                 1351410376848
    domContentLoadedEventEnd    1351410373805
    domContentLoadedEventStart  1351410373766
    domInteractive              1351410373765
    domLoading                  1351410372299
    domainLookupEnd             1351410372170
    domainLookupStart           1351410372170
    fetchStart                  1351410372171
    loadEventEnd                1351410376849
    loadEventStart              1351410376848
```

分析

打开一个网页要经过哪些步骤

点击连接

DNS查询与解析

数据库查询

建立连接

输入网址

返回数据

渲染页面

执行脚本

加载图片

业务处理

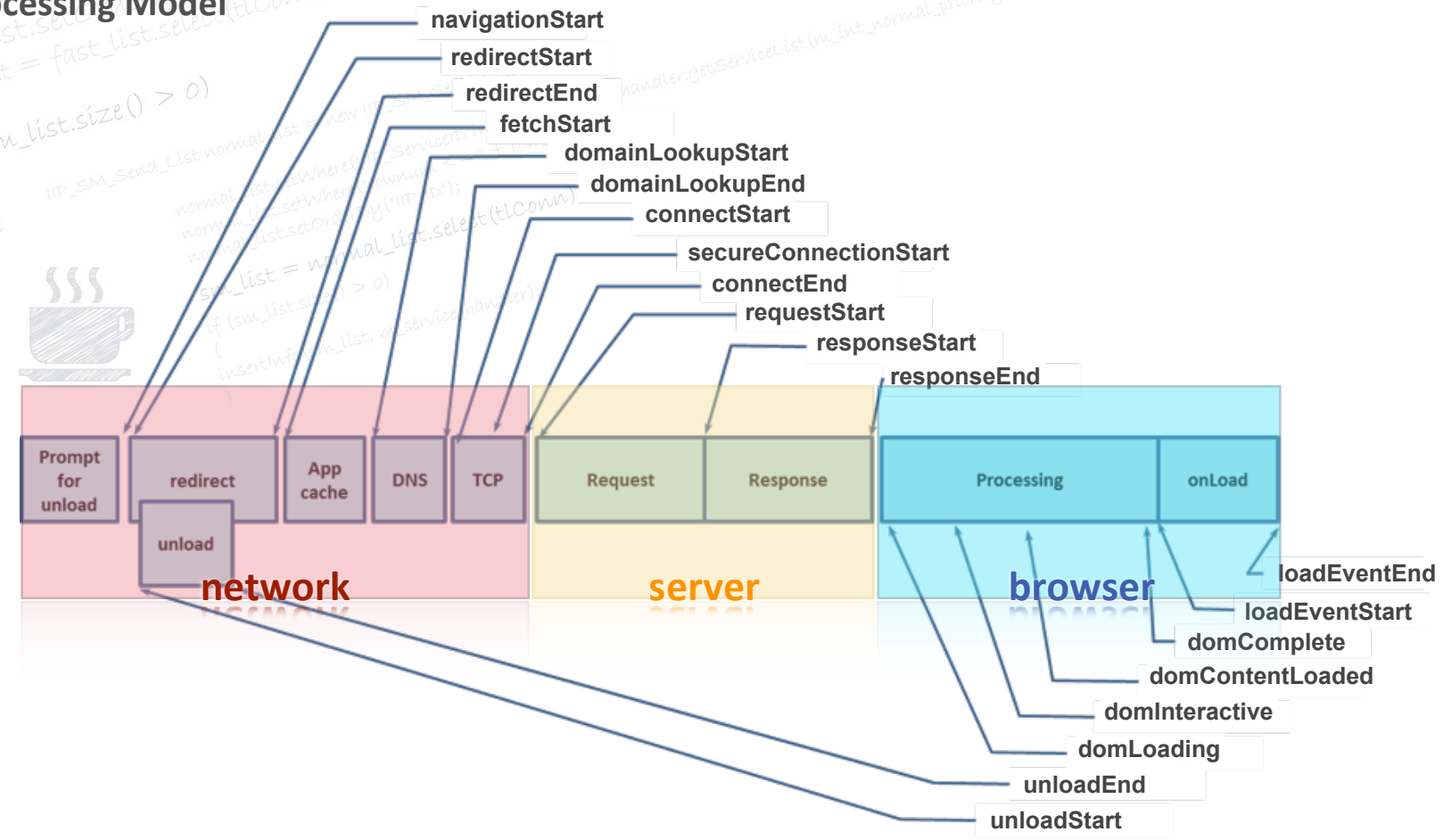
分析

打开一个网页要经过哪些步骤

network	server	browser
点击连接	数据库查询	渲染页面
输入网址	业务处理	执行脚本
DNS查询与解析	返回数据	加载图片
建立连接		

W3C Navigation Timing API

Processing Model



W3C Navigation Timing API

PerformanceTiming interface Attribute

- navigationStart
- redirectStart
- redirectEnd
- domainLookupStart
- domainLookupEnd
- fetchStart
- connectStart
- secureConnectionStart
- connectEnd
- requestStart
- responseStart
- responseEnd
- unloadEventStart
- unloadEventEnd
- domLoading
- domInteractive
- domContentLoadedEventStart
- domContentLoadedEventEnd
- domComplete
- loadEventStart
- loadEventEnd

W3C Navigation Timing API

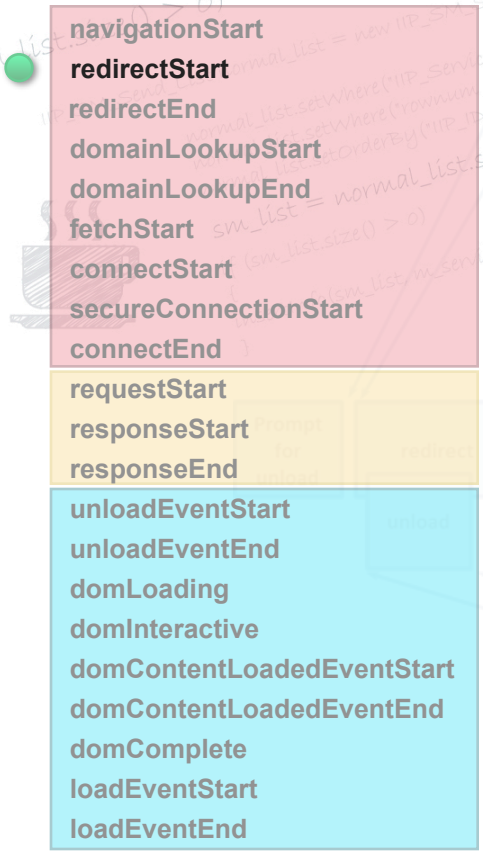
PerformanceTiming interface Attribute

- navigationStart
- redirectStart
- redirectEnd
- domainLookupStart
- domainLookupEnd
- fetchStart
- connectStart
- secureConnectionStart
- connectEnd
- requestStart
- responseStart
- responseEnd
- unloadEventStart
- unloadEventEnd
- domLoading
- domInteractive
- domContentLoadedEventStart
- domContentLoadedEventEnd
- domComplete
- loadEventStart
- loadEventEnd

This attribute must return the time immediately after the user agent finishes prompting to unload the previous document. If there is no previous document, this attribute must return the same value as fetchStart

W3C Navigation Timing API

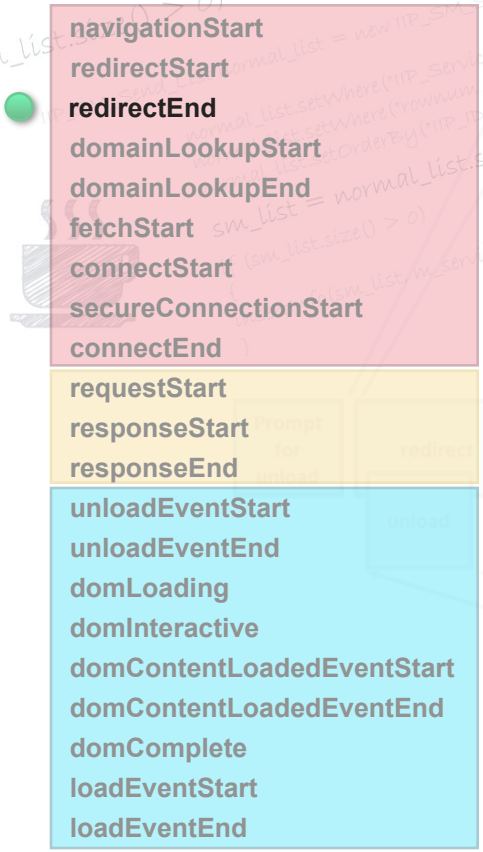
PerformanceTiming interface Attribute



If there are HTTP redirects or equivalent when navigating and if all the redirects or equivalent are from the same origin, this attribute must return the starting time of the fetch that initiates the redirect. Otherwise, this attribute must return zero

W3C Navigation Timing API

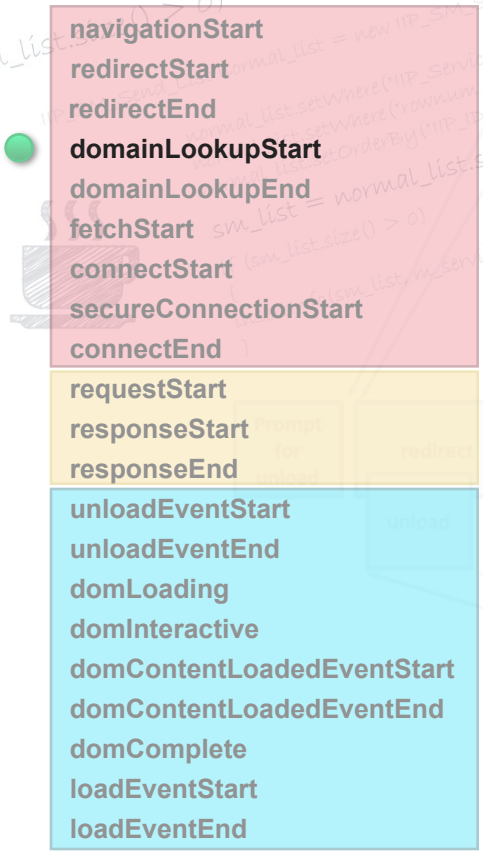
PerformanceTiming interface Attribute



If there are HTTP redirects or equivalent when navigating and all redirects and equivalents are from the same origin, this attribute must return the time immediately after receiving the last byte of the response of the last redirect. Otherwise, this attribute must return zero

W3C Navigation Timing API

PerformanceTiming interface Attribute



This attribute must return the time immediately before the user agent starts the domain name lookup for the current document. If a persistent connection is used or the current document is retrieved from relevant application caches or local resources, this attribute must return the same value as fetchStart

W3C Navigation Timing API

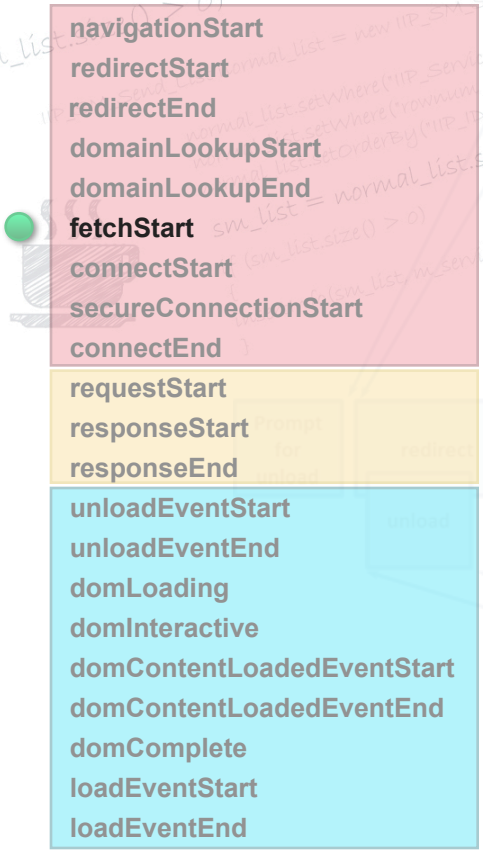
PerformanceTiming interface Attribute

- navigationStart
- redirectStart
- redirectEnd
- domainLookupStart
- domainLookupEnd
- fetchStart
- connectStart
- secureConnectionStart
- connectEnd
- requestStart
- responseStart
- responseEnd
- unloadEventStart
- unloadEventEnd
- domLoading
- domInteractive
- domContentLoadedEventStart
- domContentLoadedEventEnd
- domComplete
- loadEventStart
- loadEventEnd

This attribute must return the time immediately after the user agent finishes the domain name lookup for the current document. If a persistent connection is used or the current document is retrieved from relevant application caches or local resources, this attribute must return the same value as fetchStart

W3C Navigation Timing API

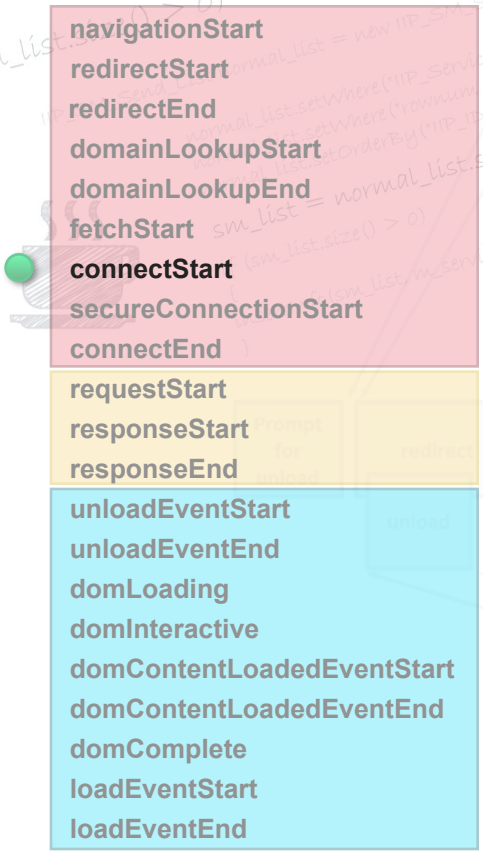
PerformanceTiming interface Attribute



If the new resource is to be fetched using HTTP GET or equivalent, fetchStart must return the time immediately before the user agent starts checking any relevant application caches. Otherwise, it must return the time when the user agent starts fetching the resource

W3C Navigation Timing API

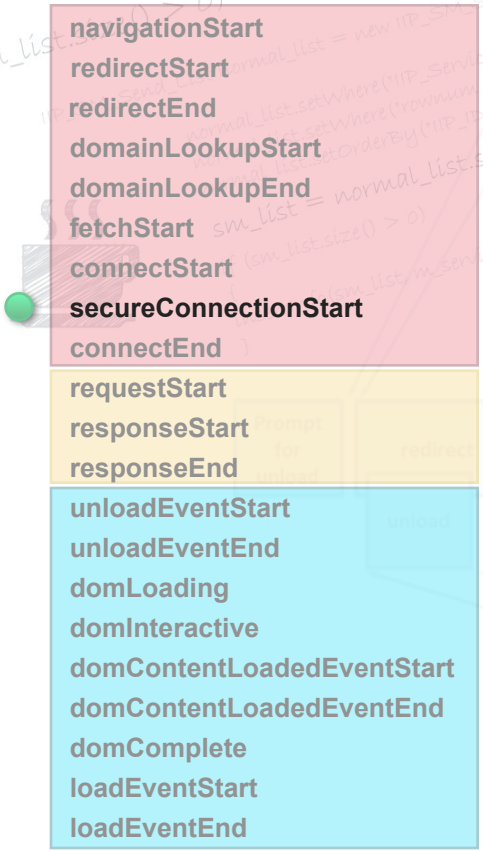
PerformanceTiming interface Attribute



This attribute must return the time immediately before the user agent start establishing the connection to the server to retrieve the document. If a persistent connection is used or the current document is retrieved from relevant application caches or local resources, this attribute must return value of domainLookupEnd

W3C Navigation Timing API

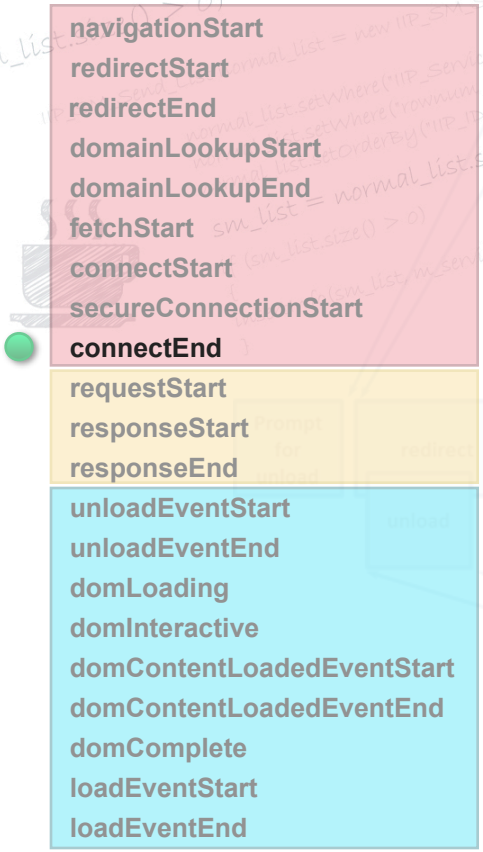
PerformanceTiming interface Attribute



This attribute is optional. User agents that don't have this attribute available must set it as undefined. When this attribute is available, if the scheme of the current page is HTTPS, this attribute must return the time immediately before the user agent starts the handshake process to secure the current connection. If this attribute is available but HTTPS is not used, this attribute must return zero

W3C Navigation Timing API

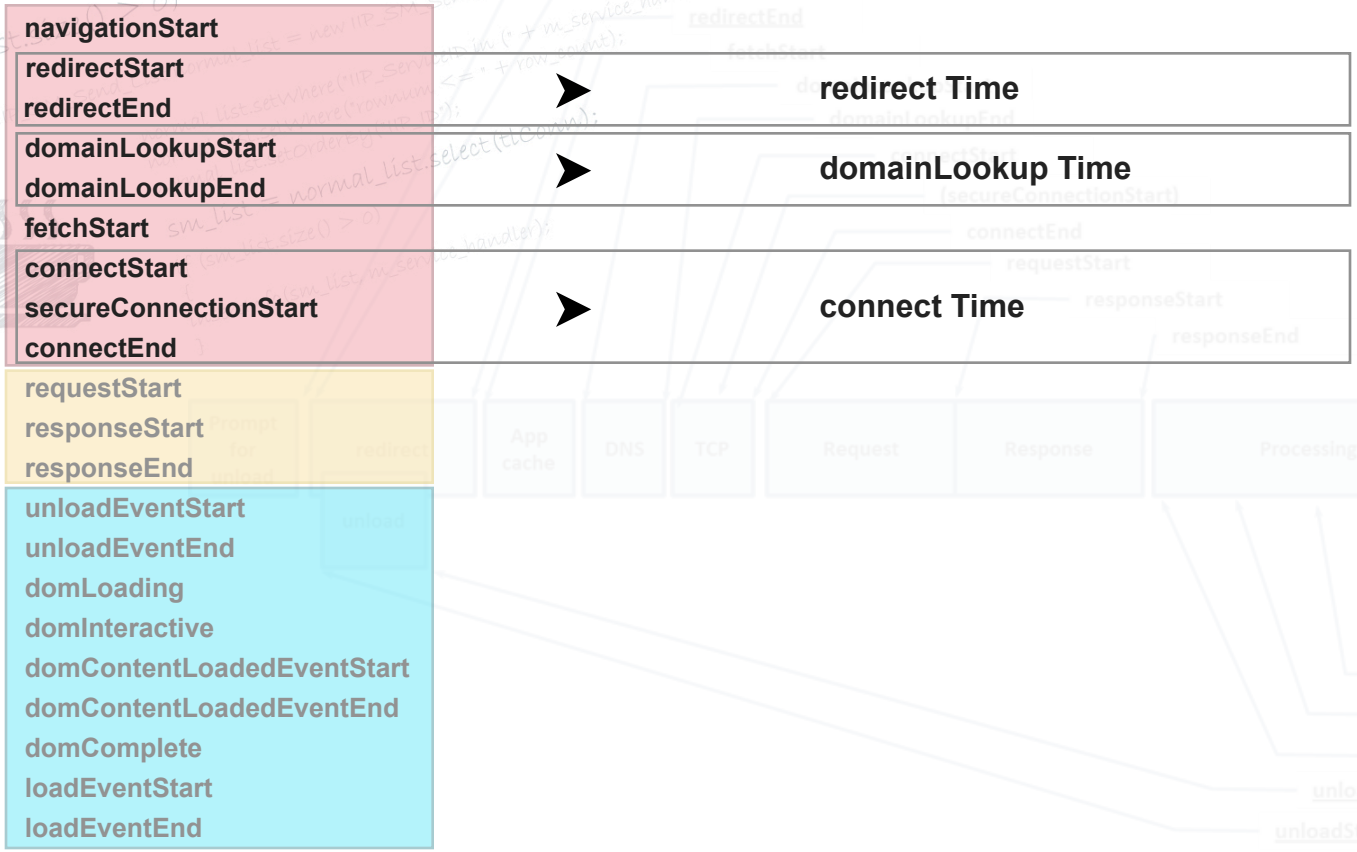
PerformanceTiming interface Attribute



This attribute must return the time immediately after the user agent finishes establishing the connection to the server to retrieve the current document. If a persistent connection is used or the current document is retrieved from relevant application caches or local resources, this attribute must return the value of domainLookupEnd

W3C Navigation Timing API

PerformanceTiming interface Attribute



W3C Navigation Timing API

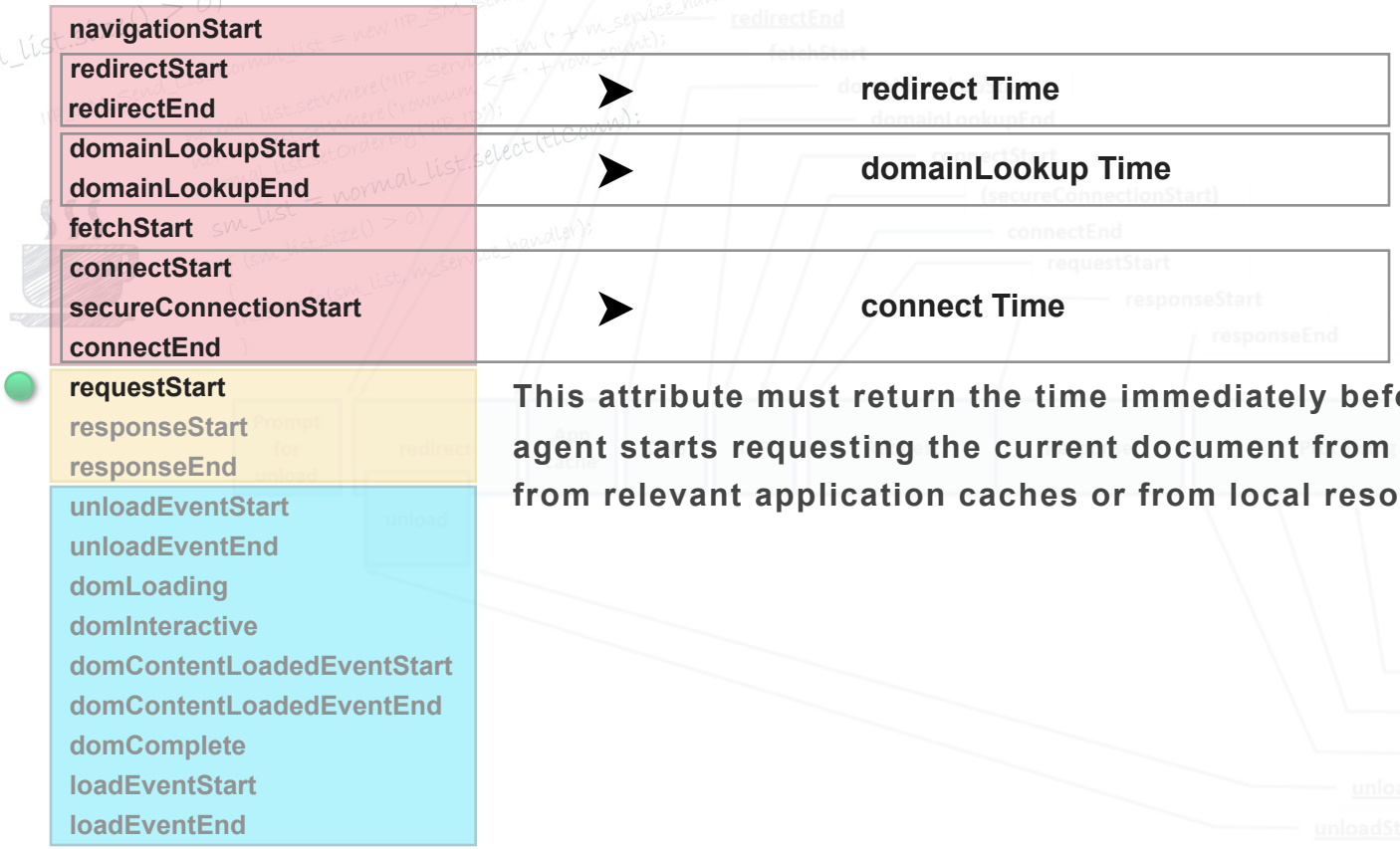
PerformanceTiming interface Attribute

优化原则：

- 减少重定向和坏连接
- 精简页面静态资源的数量和域名数量
- 使用缓存
- 谨慎使用Cookie和Https

W3C Navigation Timing API

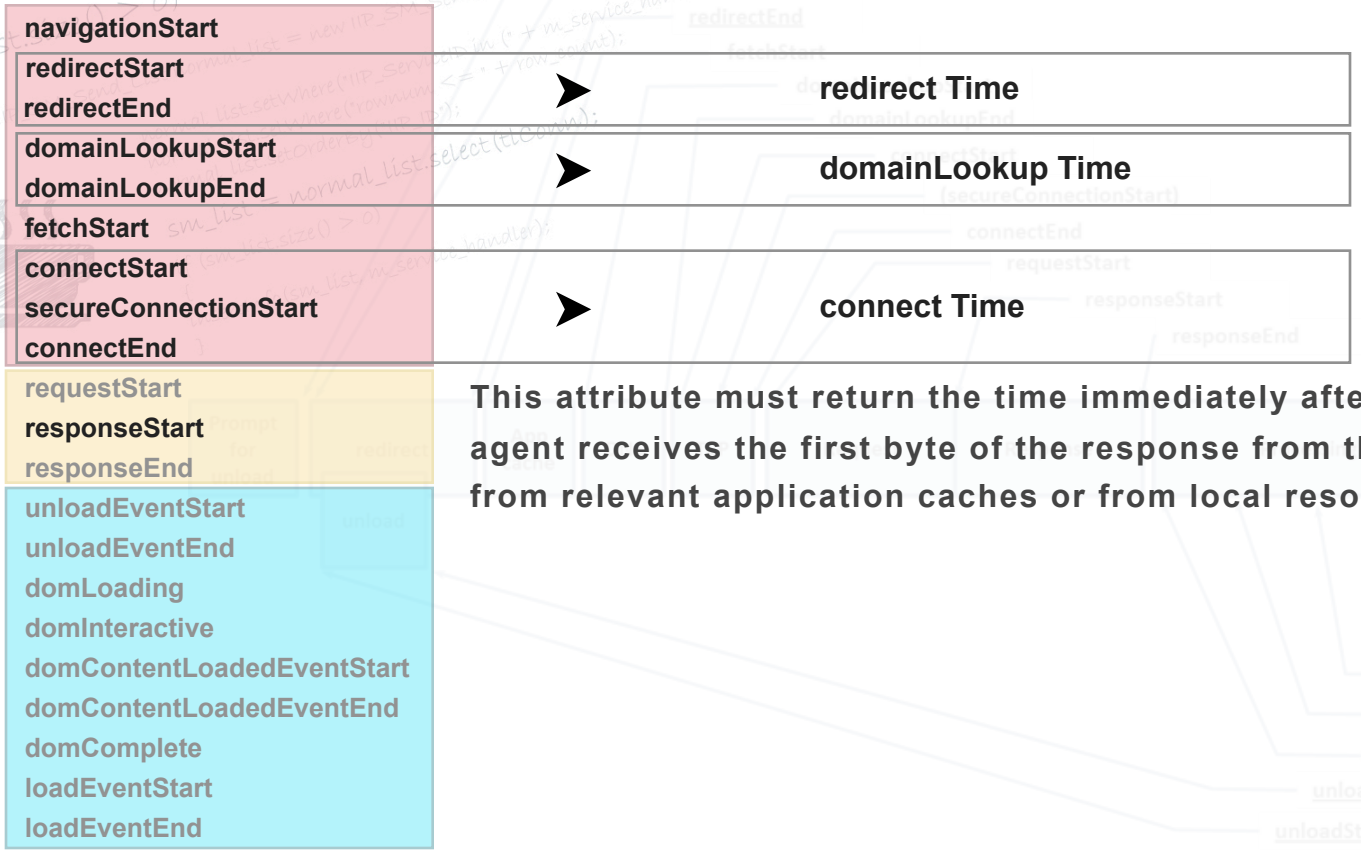
PerformanceTiming interface Attribute



This attribute must return the time immediately before the user agent starts requesting the current document from the server, or from relevant application caches or from local resources

W3C Navigation Timing API

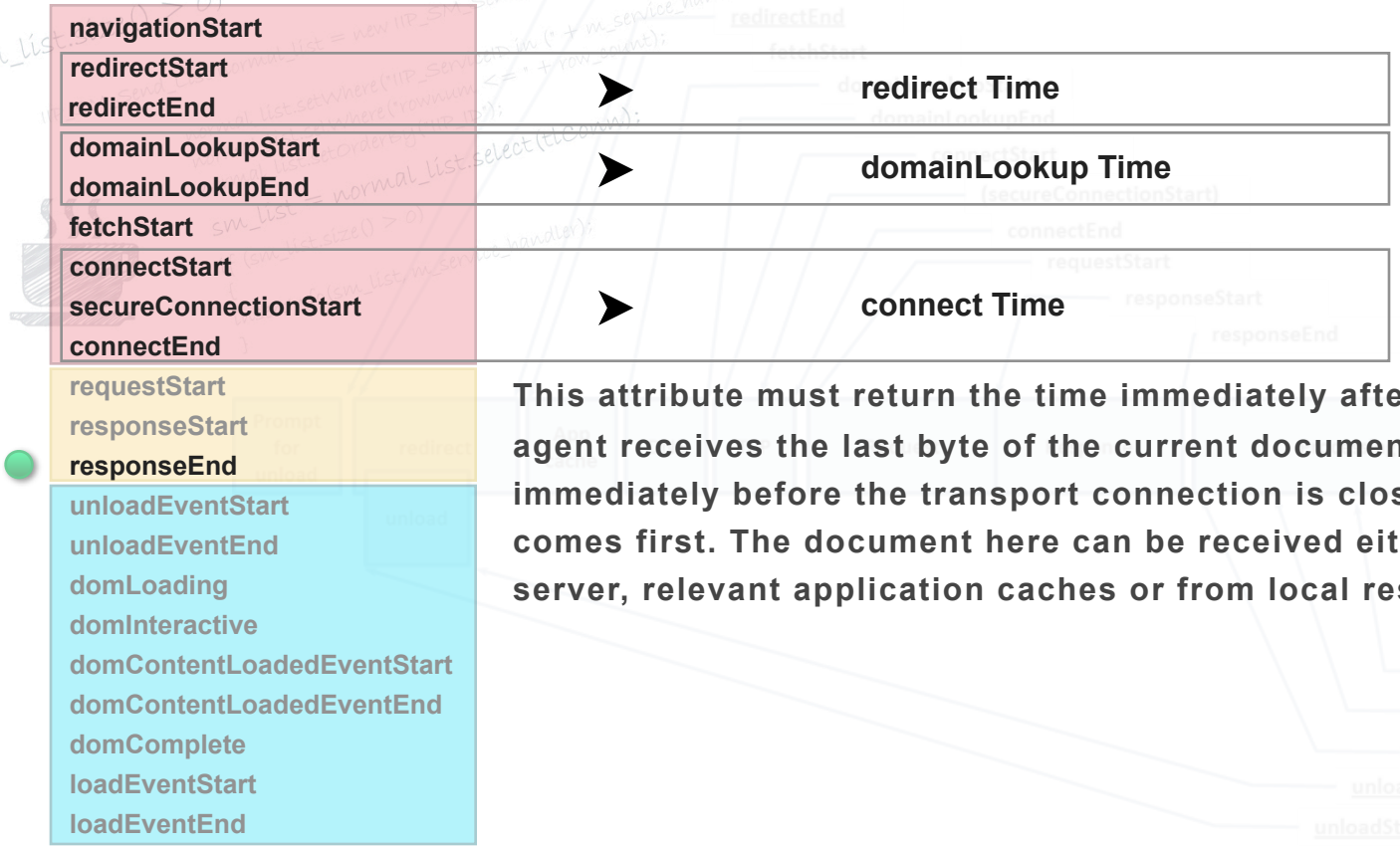
PerformanceTiming interface Attribute



This attribute must return the time immediately after the user agent receives the first byte of the response from the server, or from relevant application caches or from local resources

W3C Navigation Timing API

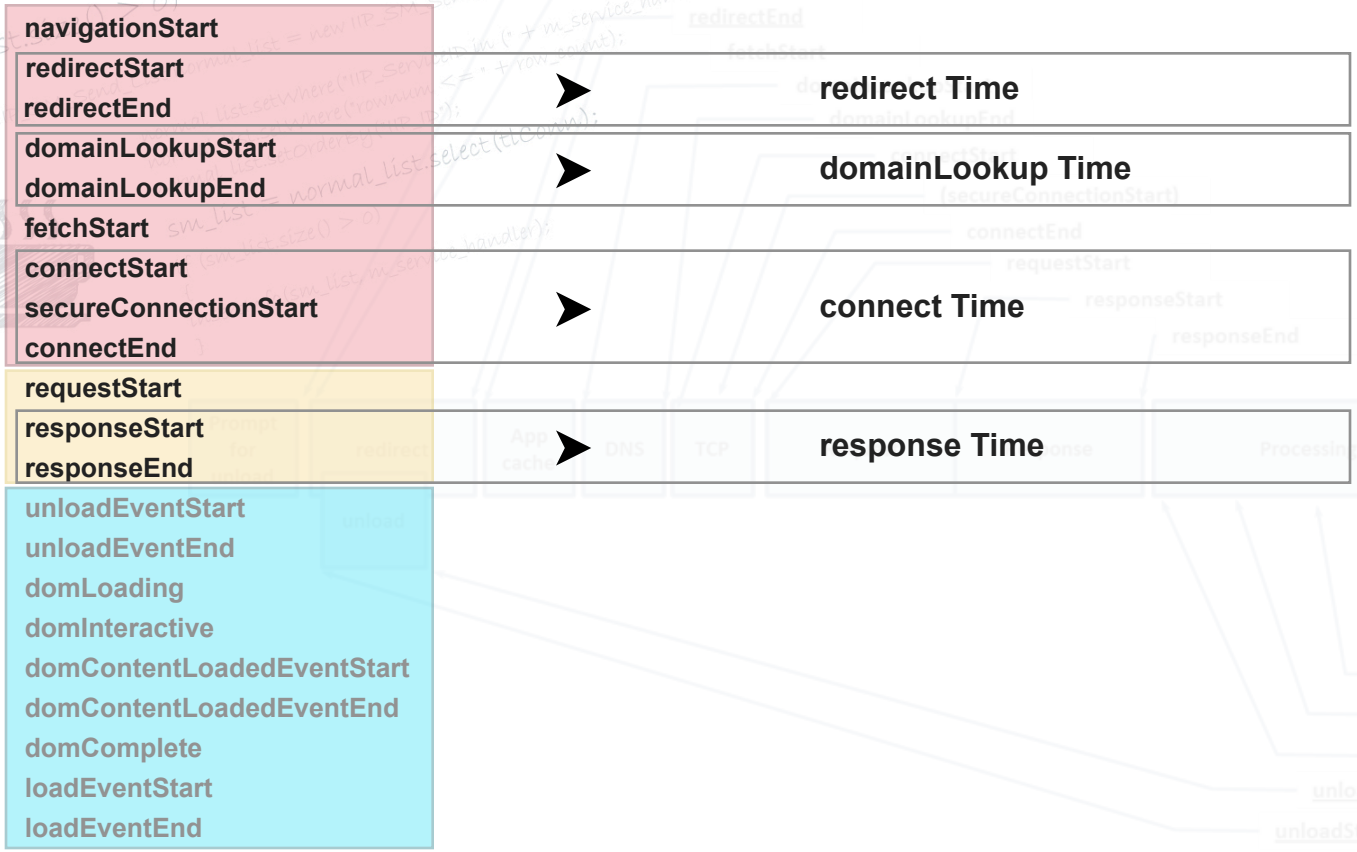
PerformanceTiming interface Attribute



This attribute must return the time immediately after the user agent receives the last byte of the current document or immediately before the transport connection is closed, whichever comes first. The document here can be received either from the server, relevant application caches or from local resources

W3C Navigation Timing API

PerformanceTiming interface Attribute



W3C Navigation Timing API

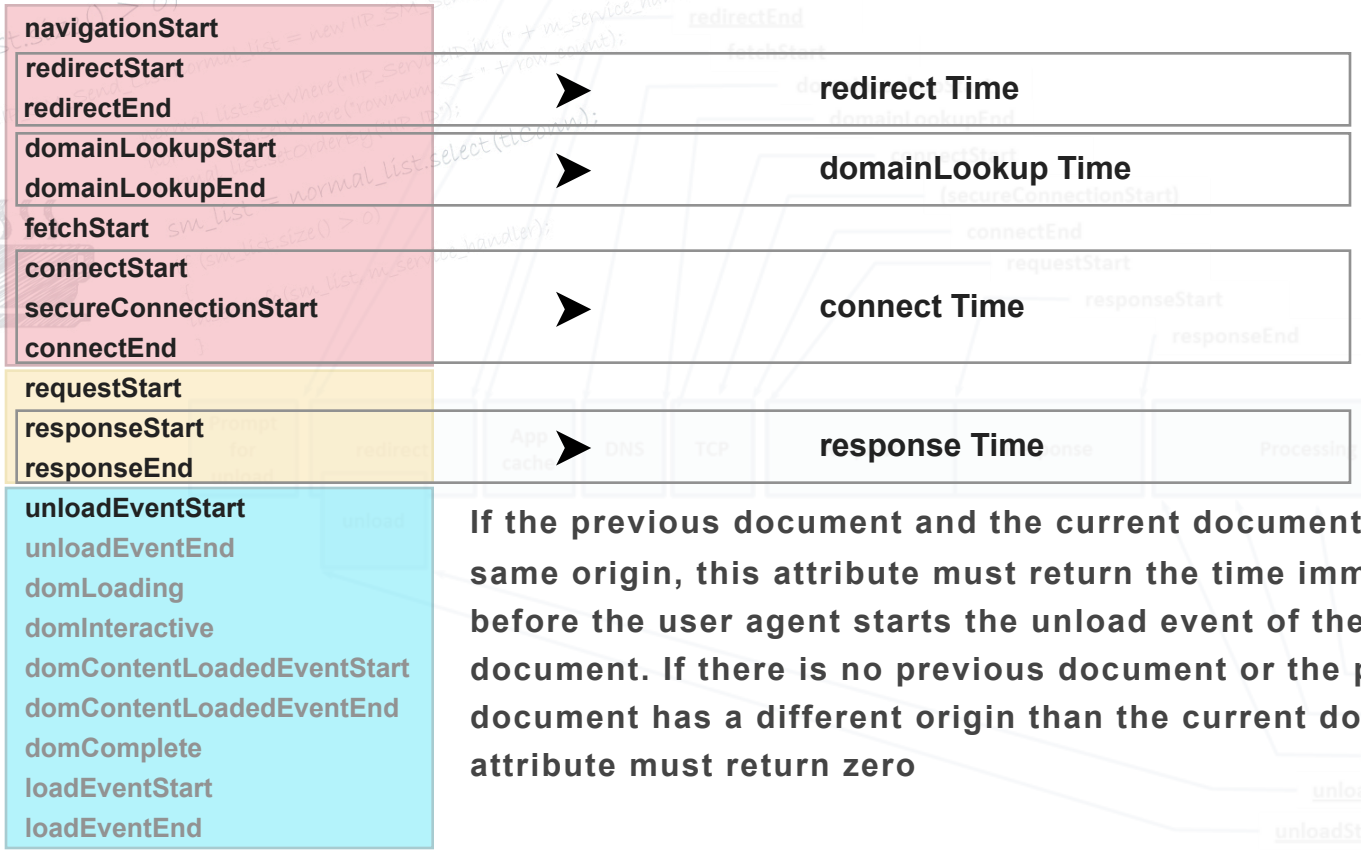
PerformanceTiming interface Attribute

优化原则：

- 尽早返回首字节内容
- 异步加载页面底部的HTML内容
- 适量压缩HTML并开启GZIP

W3C Navigation Timing API

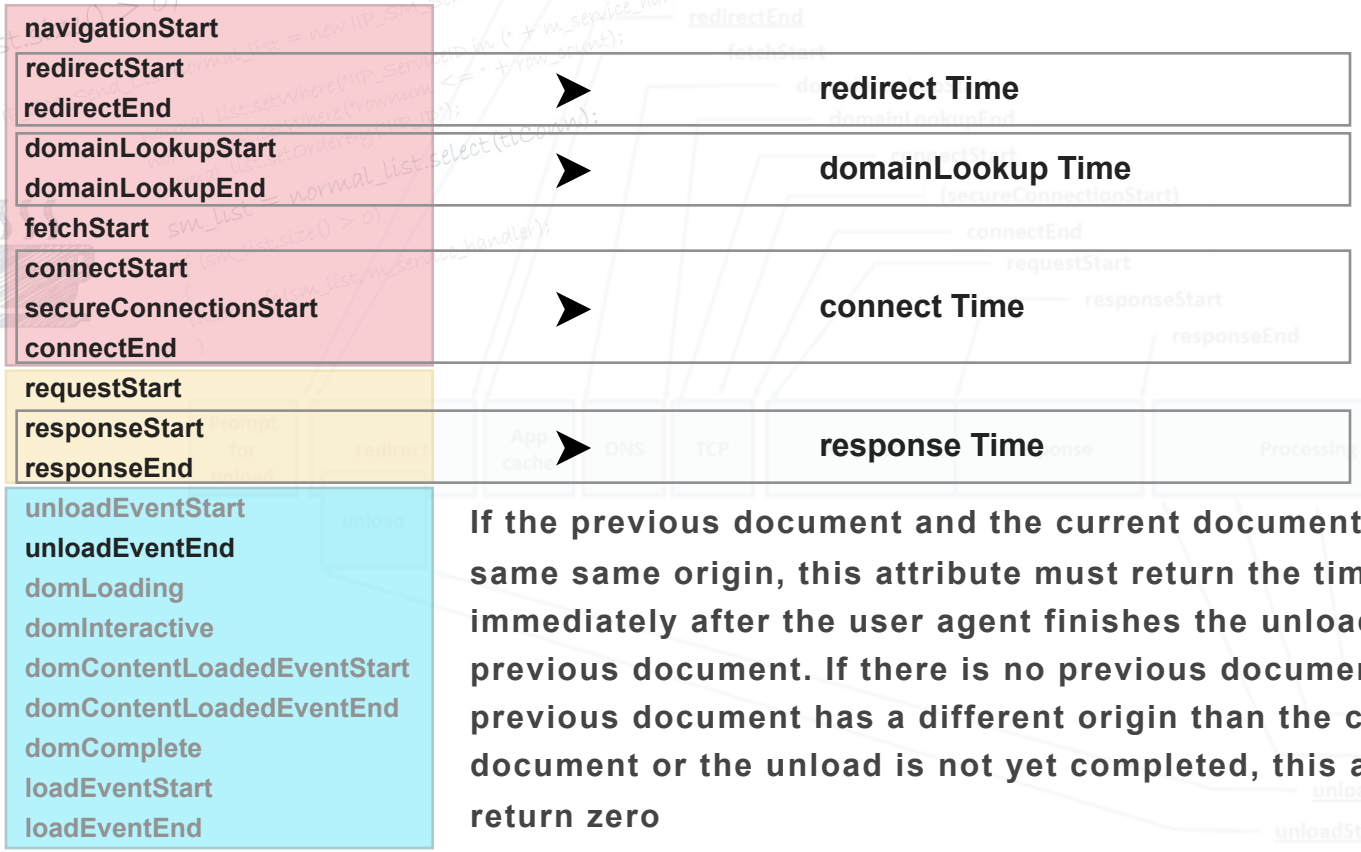
PerformanceTiming interface Attribute



If the previous document and the current document have the same origin, this attribute must return the time immediately before the user agent starts the unload event of the previous document. If there is no previous document or the previous document has a different origin than the current document, this attribute must return zero

W3C Navigation Timing API

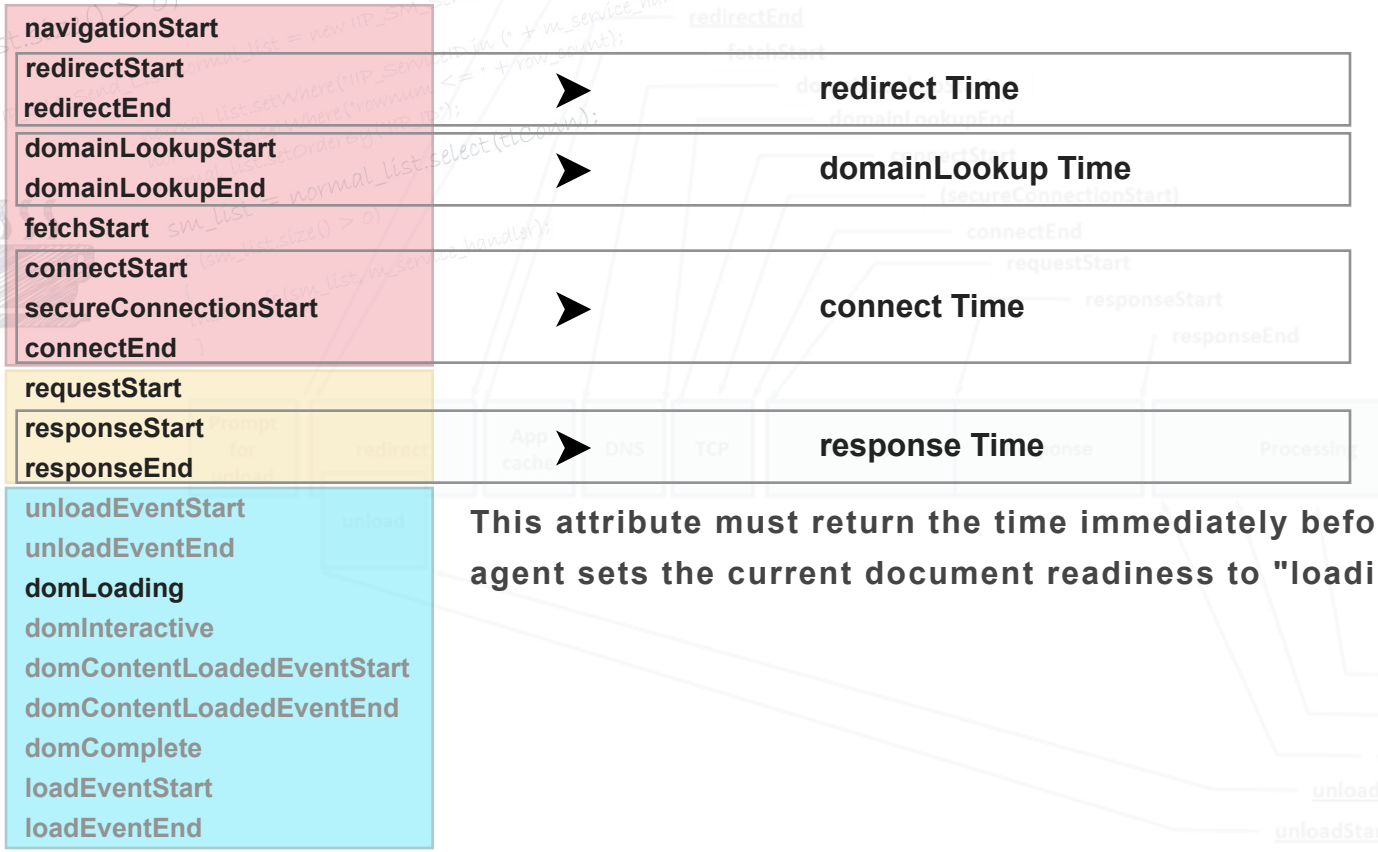
PerformanceTiming interface Attribute



If the previous document and the current document have the same same origin, this attribute must return the time immediately after the user agent finishes the unload event of the previous document. If there is no previous document or the previous document has a different origin than the current document or the unload is not yet completed, this attribute must return zero

W3C Navigation Timing API

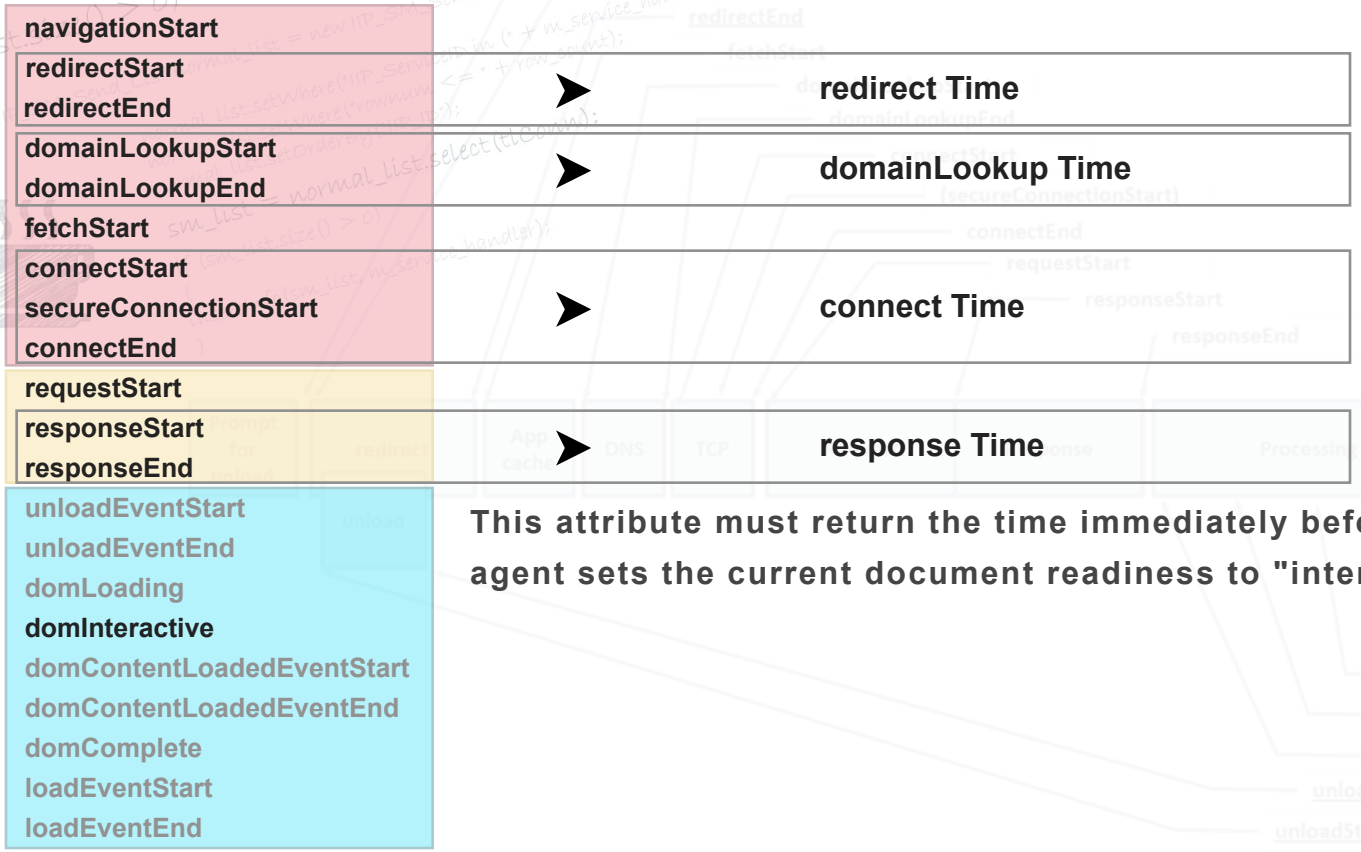
PerformanceTiming interface Attribute



This attribute must return the time immediately before the user agent sets the current document readiness to "loading"

W3C Navigation Timing API

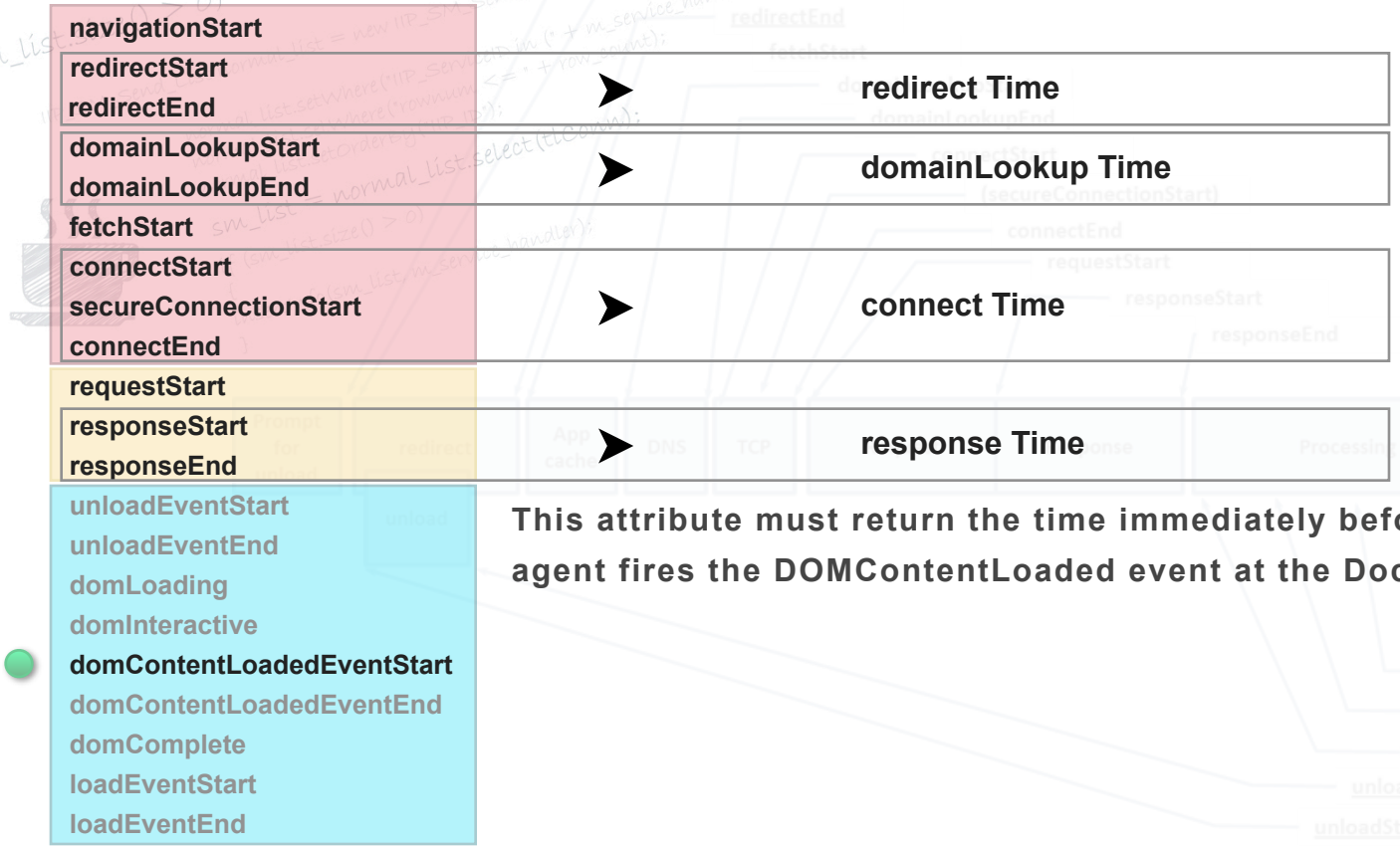
PerformanceTiming interface Attribute



This attribute must return the time immediately before the user agent sets the current document readiness to "interactive"

W3C Navigation Timing API

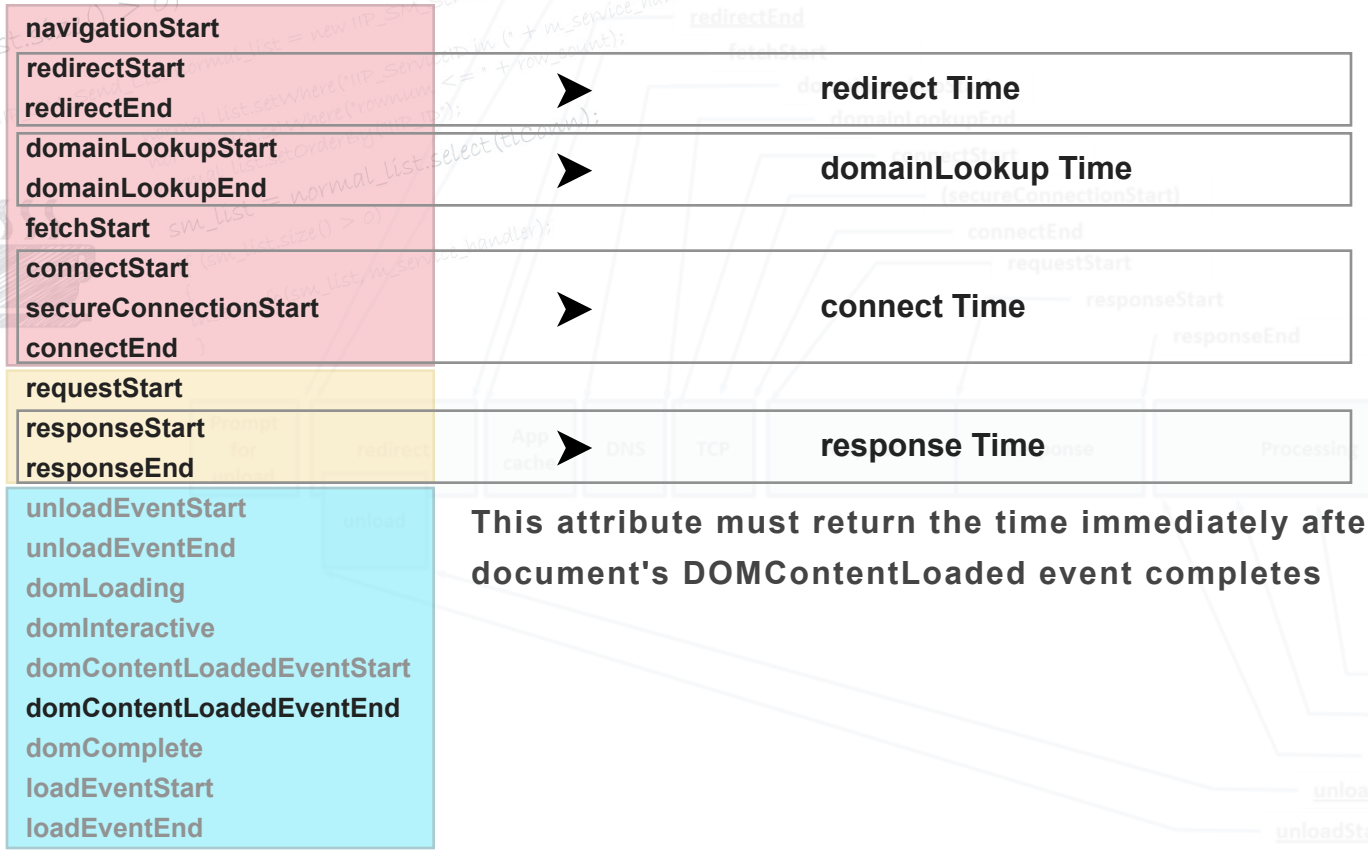
PerformanceTiming interface Attribute



This attribute must return the time immediately before the user agent fires the DOMContentLoaded event at the Document

W3C Navigation Timing API

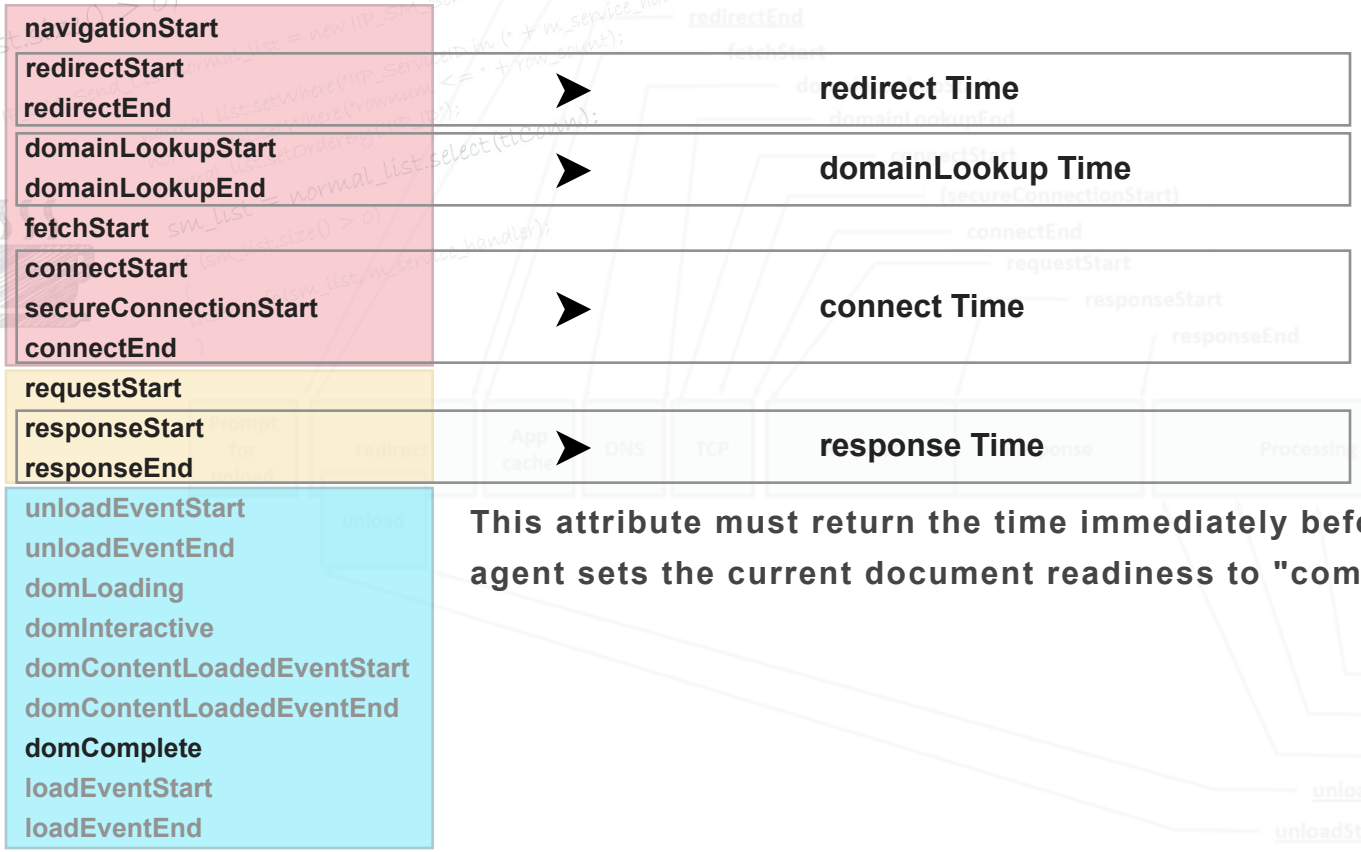
PerformanceTiming interface Attribute



This attribute must return the time immediately after the document's DOMContentLoaded event completes

W3C Navigation Timing API

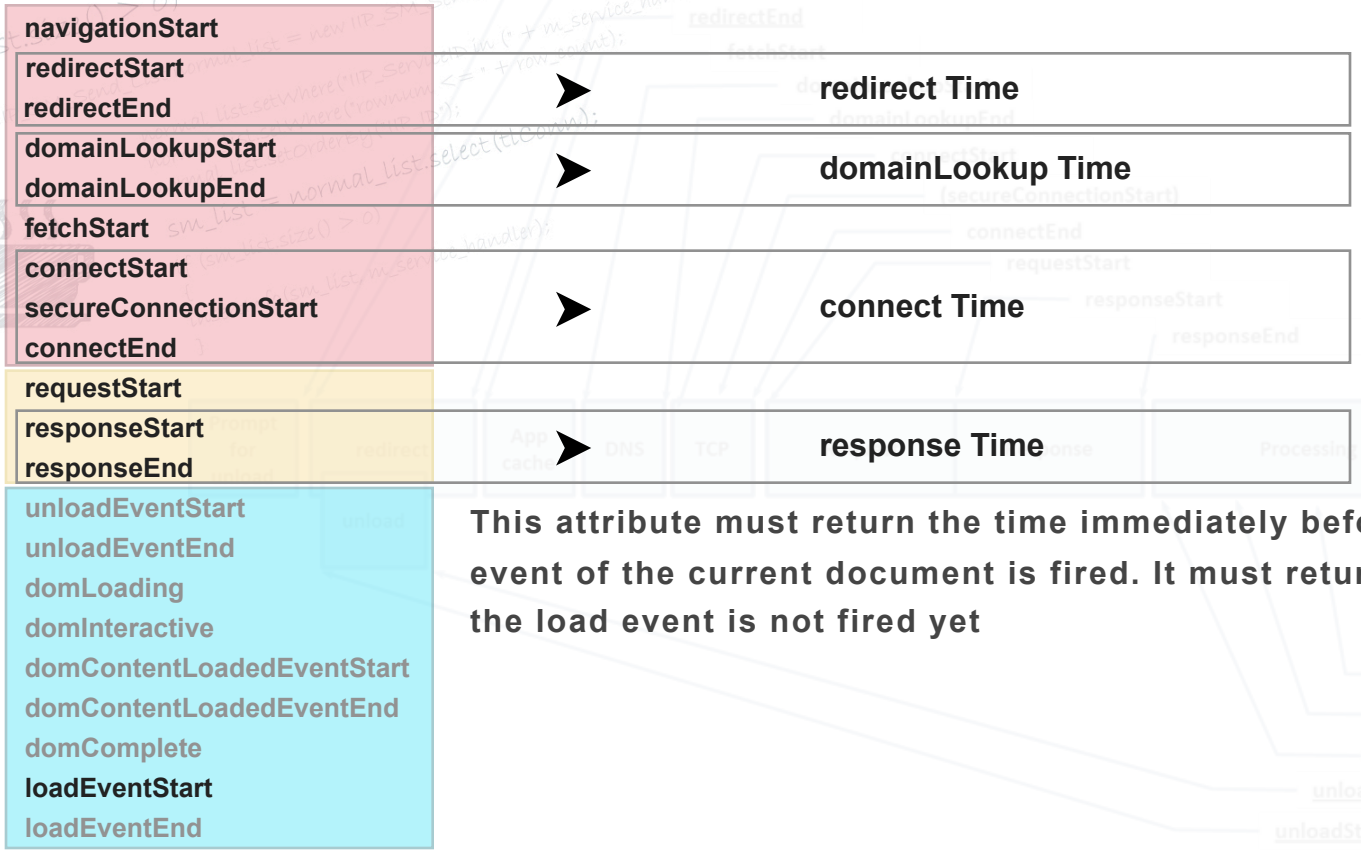
PerformanceTiming interface Attribute



This attribute must return the time immediately before the user agent sets the current document readiness to "complete"

W3C Navigation Timing API

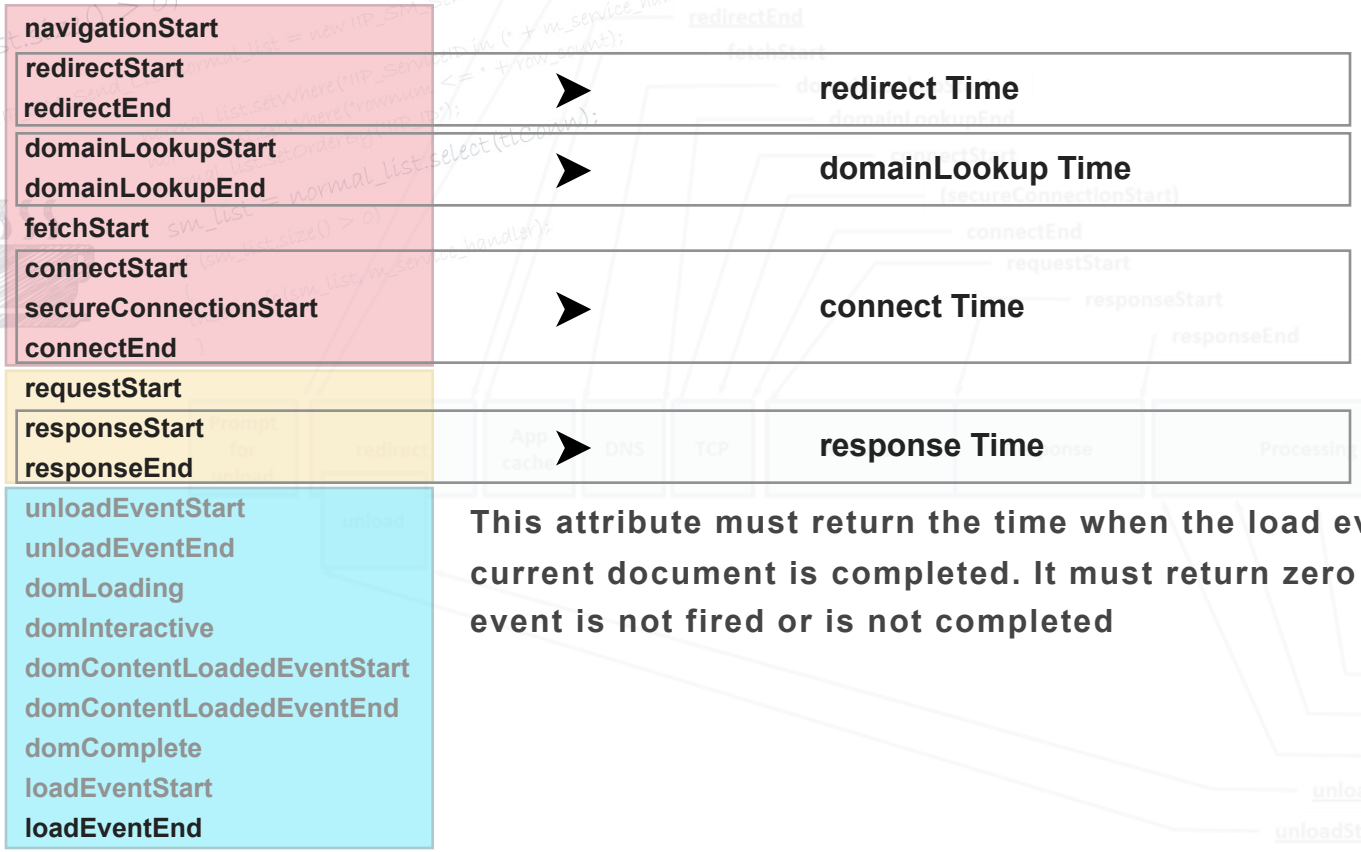
PerformanceTiming interface Attribute



This attribute must return the time immediately before the load event of the current document is fired. It must return zero when the load event is not fired yet

W3C Navigation Timing API

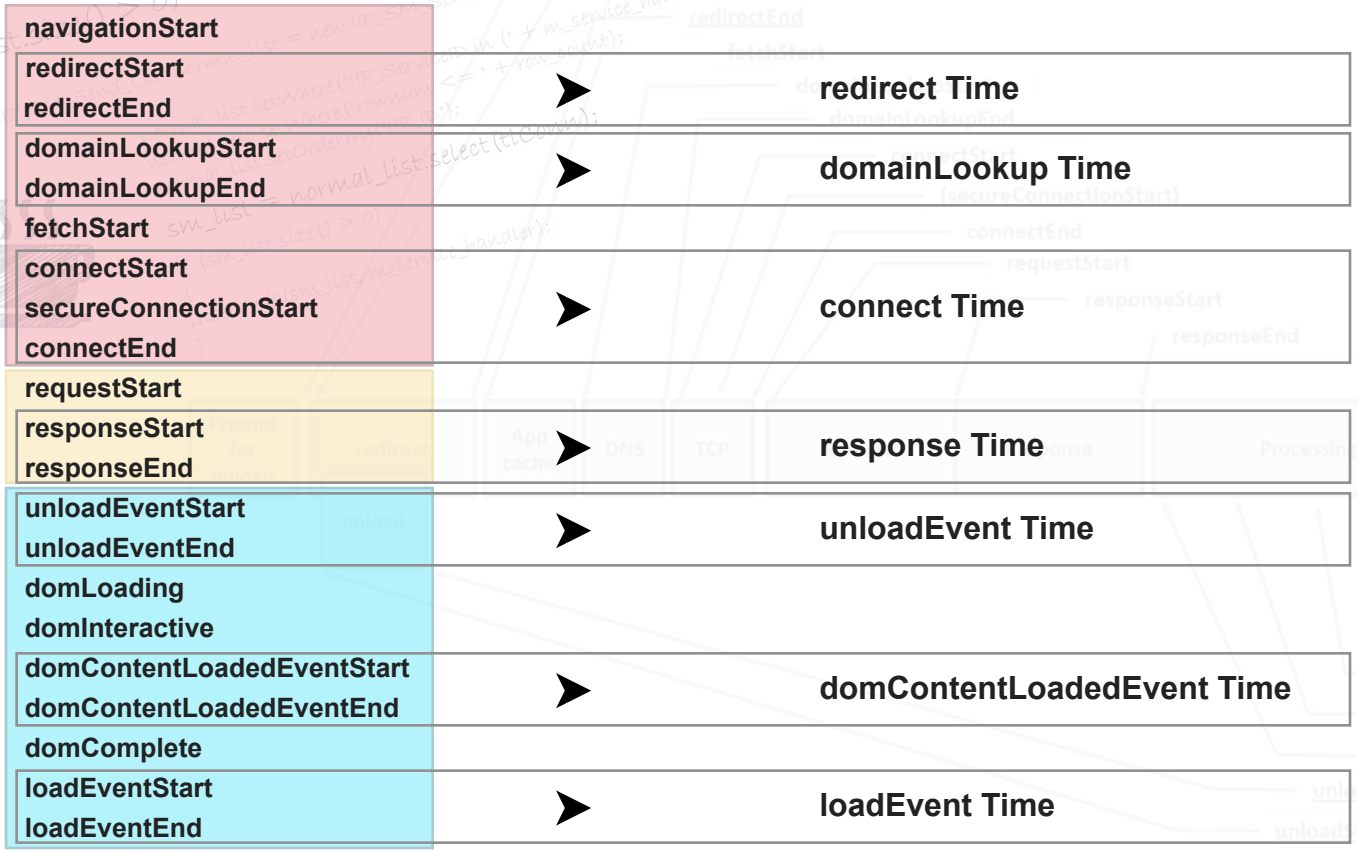
PerformanceTiming interface Attribute



This attribute must return the time when the load event of the current document is completed. It must return zero when the load event is not fired or is not completed

W3C Navigation Timing API

PerformanceTiming interface Attribute



W3C Navigation Timing API

PerformanceTiming interface Attribute

优化原则：

- 将CSS放在页面顶部
- 优化CSS和JS的顺序
- 控制DOM Tree数量
- 异步加载部分静态资源
- 延迟执行

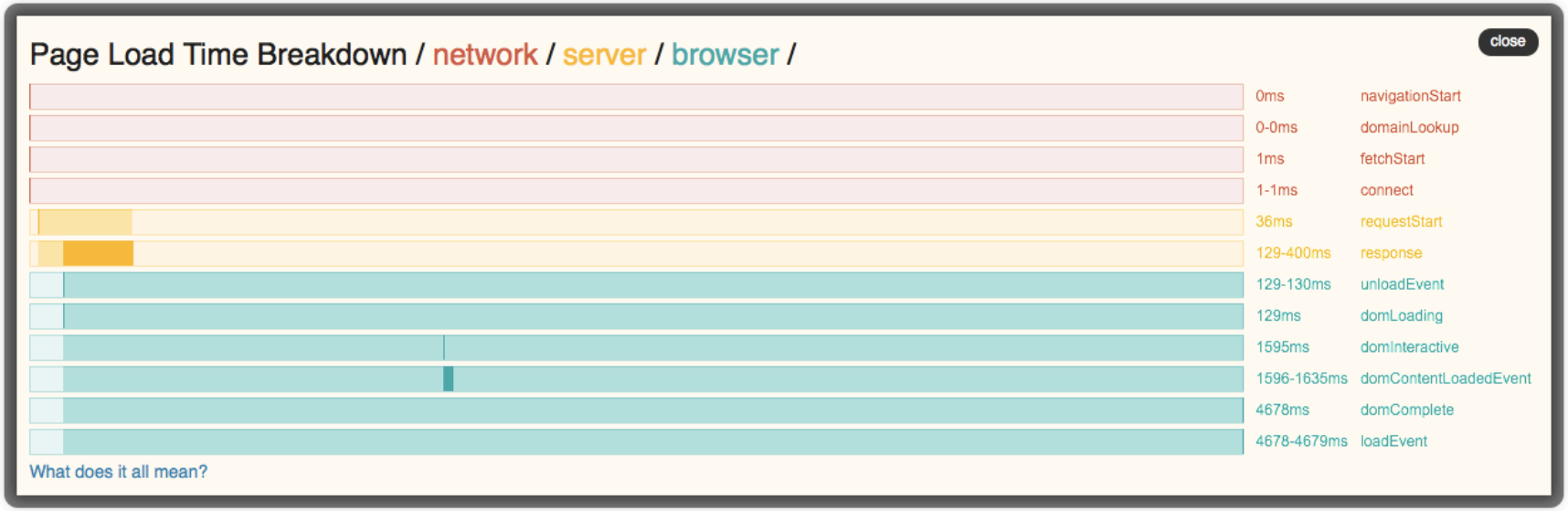
W3C Navigation Timing API

PerformanceTiming interface Attribute

1351410372170	navigationStart	0			
0	redirectStart	0	➤	0	redirect Time
0	redirectEnd	0			
1351410372170	domainLookupStart	0	➤	0	domainLookup Time
1351410372170	domainLookupEnd	0			
1351410372171	fetchStart	1			
1351410372171	connectStart	1			
0	secureConnectionStart	0	➤	0	connect Time
1351410372171	connectEnd	1			
1351410372206	requestStart	36			
1351410372299	responseStart	129	➤	271	response Time
1351410372570	responseEnd	400			
1351410372299	unloadEventStart	129	➤	1	unloadEvent Time
1351410372300	unloadEventEnd	130			
1351410372299	domLoading	129			
1351410373765	domInteractive	1595			
1351410373766	domContentLoadedEventStart	1596	➤	39	domContentLoadedEvent Time
1351410373805	domContentLoadedEventEnd	1635			
1351410376848	domComplete	4678			
1351410376848	loadEventStart	4678	➤	1	loadEvent Time
1351410376849	loadEventEnd	4679			

W3C Navigation Timing API

可视化工具：**Breaking Down onLoad**



http://kaaes.github.com/timing/

现状

现状

标准制定进度：

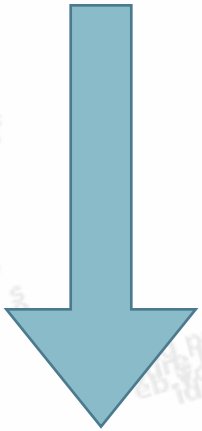
Proposed Recommendations

2012 7.26 成为建议推荐标准

现状

标准制定进度：

1	Submissions	
2	Notes	
3	Working Groups	2010 08.18
4	Working Drafts	2010 10.26
5	Candidate Recommendations	2011 03.15
6	Proposed Recommendations	2012 07.26
7	Recommendations	



现状

浏览器支持情况：

standards mode

6 +

7 +

4.0 +

参考资料

- http://www.w3.org/TR/navigation-timing/
- http://www.w3.org/2010/08/webperf.html
- http://www.w3.org/2010/webperf/
- http://www.w3.org/Consortium/Process/
- http://www.w3.org/TR/html5/history.html#prompt-to-unload-a-document
- http://msdn.microsoft.com/en-us/library/ff975075
- https://developers.google.com/speed/docs/best-practices/rules_intro
- https://developer.mozilla.org/en-US/docs/API/navigationTiming

- http://kaaes.github.com/timing/
- http://kaaes.github.com/timing/info.html
- http://oldj.net/article/measuring-the-user-latency/
- http://66.7percentangel.com/2011/12/breaking-down-onload-event-performance-bookmarklet/

谢 谢