4.20 前端性能监控

前端性能监控平台架构

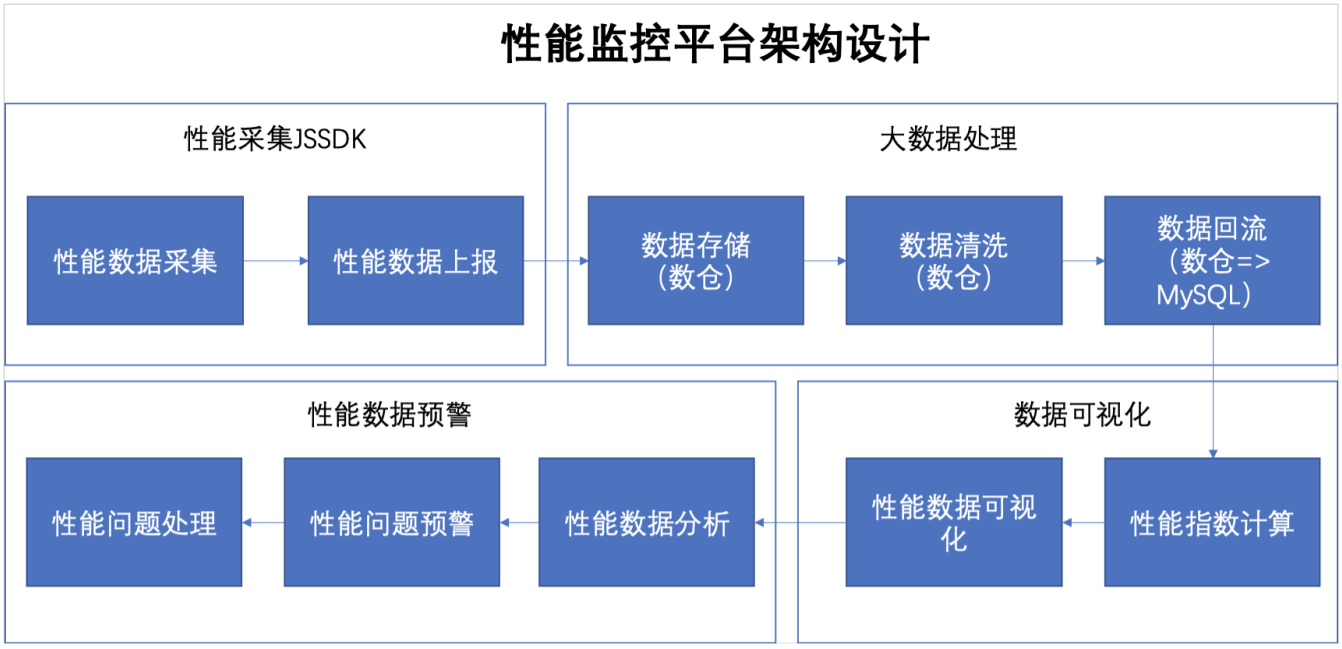
前端性能监控原理

- FP（First Paint）
- FCP（First contentful paint）
- LCP（LargestContentfulPaint）
- FMP（First Meaningful Paint）
- DCL（DOMContentLoaded）
- L（load）

前端性能采集

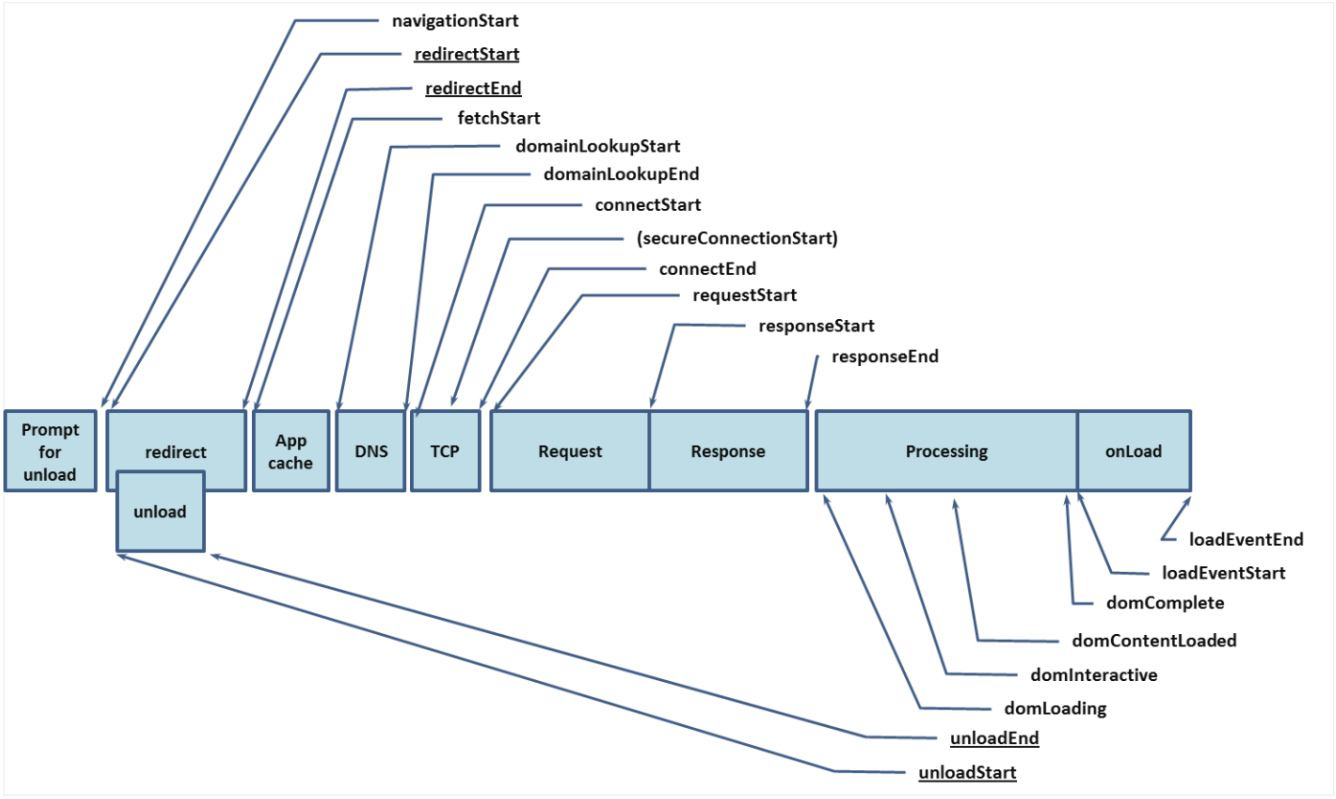
- PerformanceTiming
- 获取fp、fcp、fmp（PerformancePaintTiming）
- 获取更多性能指标：PerformanceObserver

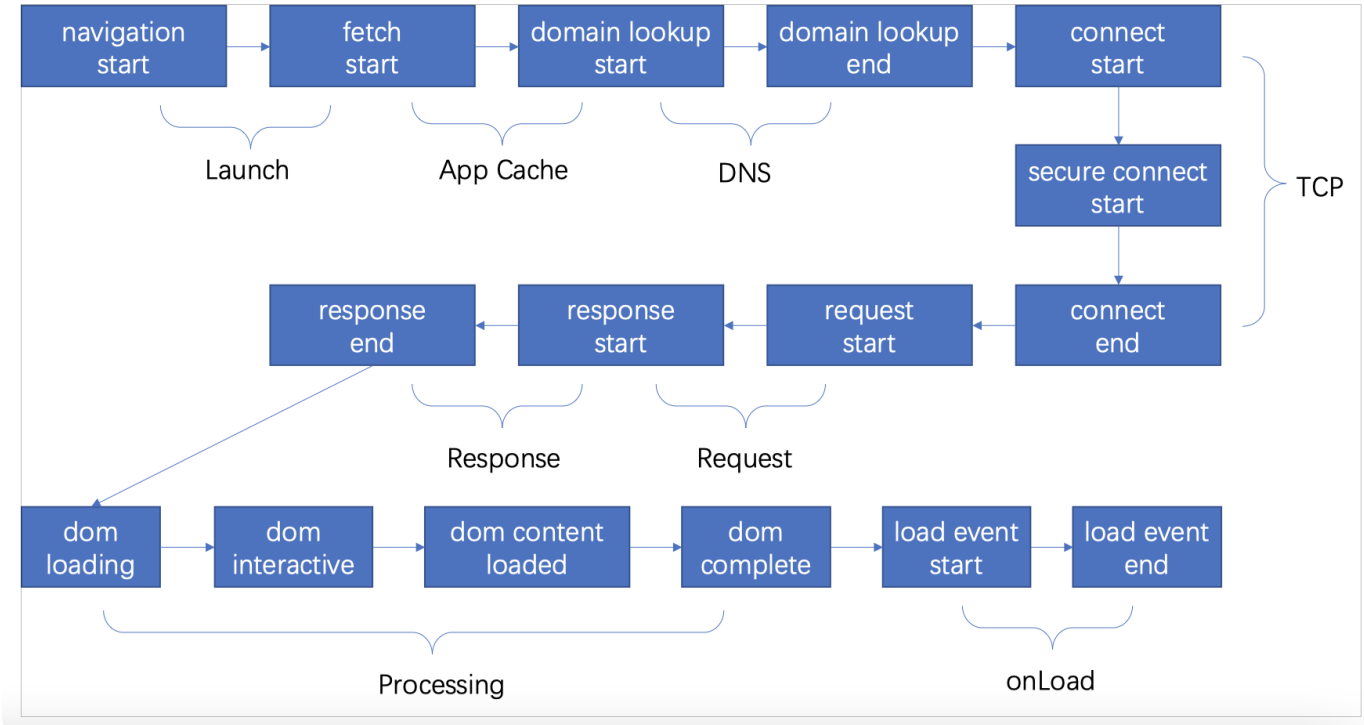
前端性能监控平台架构



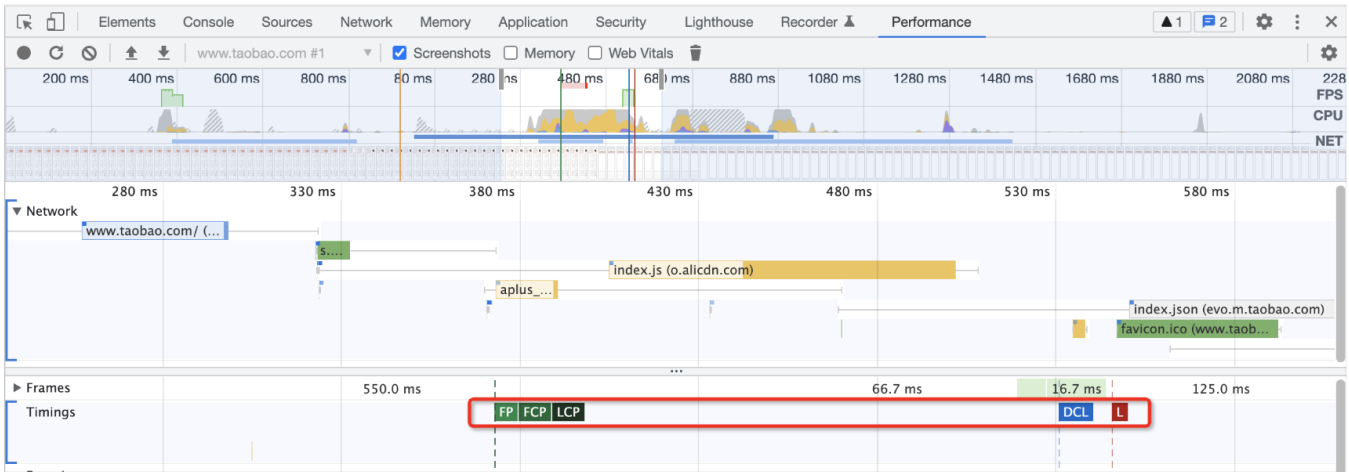
前端性能监控原理

https://developer.mozilla.org/zh-CN/docs/Web/API/Navigation_timing_API





Chrome Performance插件



FP（First Paint）

First Paint, part of the Paint Timing API, is the time between navigation and when the browser renders the first pixels to the screen, rendering anything that is visually different from what was on the screen prior to navigation. It answers the question "Is it happening?"

https://developer.mozilla.org/en-US/docs/Glossary/First_paint

FCP（First contentful paint）

First Contentful Paint (FCP) is when the browser renders the first bit of content from the DOM, providing the first feedback to the user that the page is actually loading. The question "Is it happening?" is "yes" when the first contentful paint completes.

https://developer.mozilla.org/en-US/docs/Glossary/First_contentful_paint

LCP（LargestContentfulPaint）

The LargestContentfulPaint interface of the Largest Contentful Paint API provides details about the largest image or text paint before user input on a web page. The timing of this paint is a good heuristic for when the main page content is available during load.

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

FMP（First Meaningful Paint）

First Meaningful Paint (FMP) is the paint after which the biggest above-the-fold layout change has happened and web fonts have loaded. It is when the answer to "Is it useful?" becomes "yes", upon first meaningful paint completion.

https://developer.mozilla.org/en-US/docs/Glossary/first_meaningful_paint

DCL（DOMContentLoaded）

The DOMContentLoaded event fires when the initial HTML document has been completely loaded and parsed, without waiting for stylesheets, images, and subframes to finish loading.

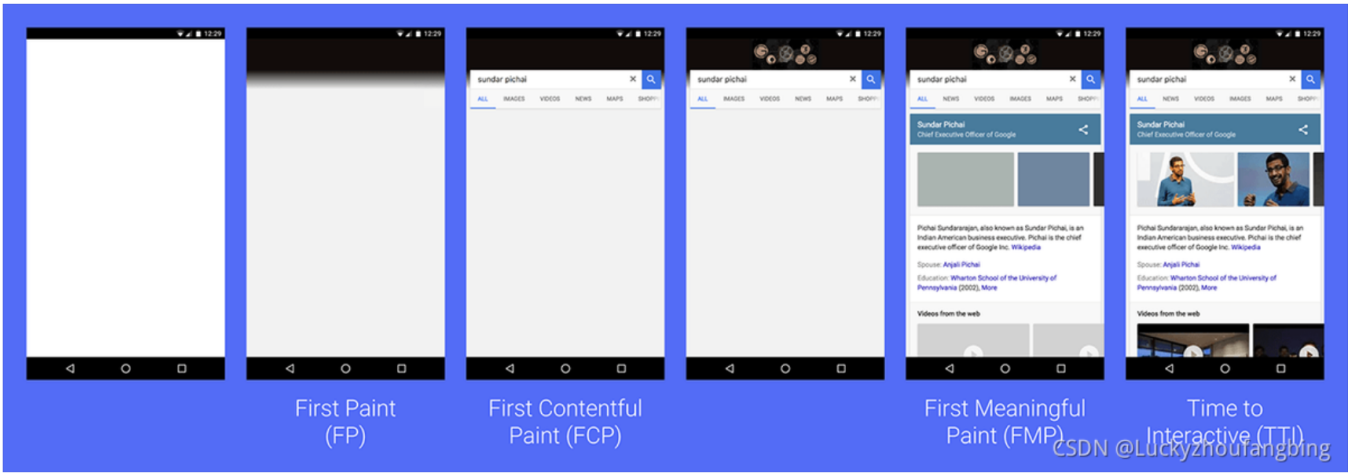
https://developer.mozilla.org/en-US/docs/Web/API/Window/DOMContentLoaded_event

L（load）

The load event is fired when the whole page has loaded, including all dependent resources such as stylesheets and images. This is in contrast to DOMContentLoaded, which is fired as soon as the page DOM has been loaded, without waiting for resources to finish loading.

https://developer.mozilla.org/en-US/docs/Web/API/Window/load_event

> 前端名词解释：https://github.com/mdn/content/tree/main/files/en-us/glossary



前端性能采集

PerformanceTiming

JavaScript 复制代码

```
window.addEventListener('load', e => {
  // PerformanceTiming
  const timing = window.performance.timing;
  const processingTiming = timing.domComplete - timing.domLoading;
  const dnsTiming = timing.domainLookupStart - timing.domainLookupEnd;
  console.log(processingTiming, dnsTiming);
  // PerformanceNavigationTiming
  const perfEntries = window.performance.getEntries();
  console.log(perfEntries);
});
```

获取fp、fcp、fmp（PerformancePaintTiming）

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

JavaScript 复制代码

```
// fp
const paint = window.performance.getEntriesByType('paint');
const fp = paint.find(e => e.name === 'first-paint').startTime;
const fcp = paint.find(e => e.name === 'first-contentful-paint').startTime;
console.log(fp, fcp);
```

获取更多性能指标：PerformanceObserver

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

JavaScript | 复制代码

```javascript
// PerformanceObserver
function observer_callback(list, observer) {
  list.getEntries().forEach(e => {
    console.log(e);
  });
}

let observer = new PerformanceObserver(observer_callback);
observer.observe({ entryTypes: ['paint', 'resource', 'mark'] });

window.performance.mark('own');
```