



GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE

WebAssembly在全民直播的应用

赵洋 全民直播 前端研发经理



GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE



关注msup
公众号获得
更多案例实践

GIAC 是中国互联网技术领域行业盛事，组委会从互联网架构最热门领域甄选前沿的有典型代表的技术创新及研发实践的架构案例，分享他们在本年度最值得总结、盘点的实践启示。

2018年11月 | 上海国际会议中心



高可用架构
改变互联网
的构建方式



GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE

目录

- 缘起
- WASM的基础使用
- WASM的实际项目应用
- 其他补充



GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE

缘起

在微信浏览器中播放FLV直播流



微信浏览器的限制和良好支持

- 直播流只允许HLS
- 延时在10s以上，实时性高的需求无法接受

- MSE无法使用
- JS编解码FLV到MP4不可行



- Canvas/WebGL支持良好
- 图像数据可以通过canvas/webgl的方式进行绘制

- WebAudio Api支持良好
- 音频流数据可以通过WebAudio Api进行操作



GIAC

全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

开源项目JSMPEG

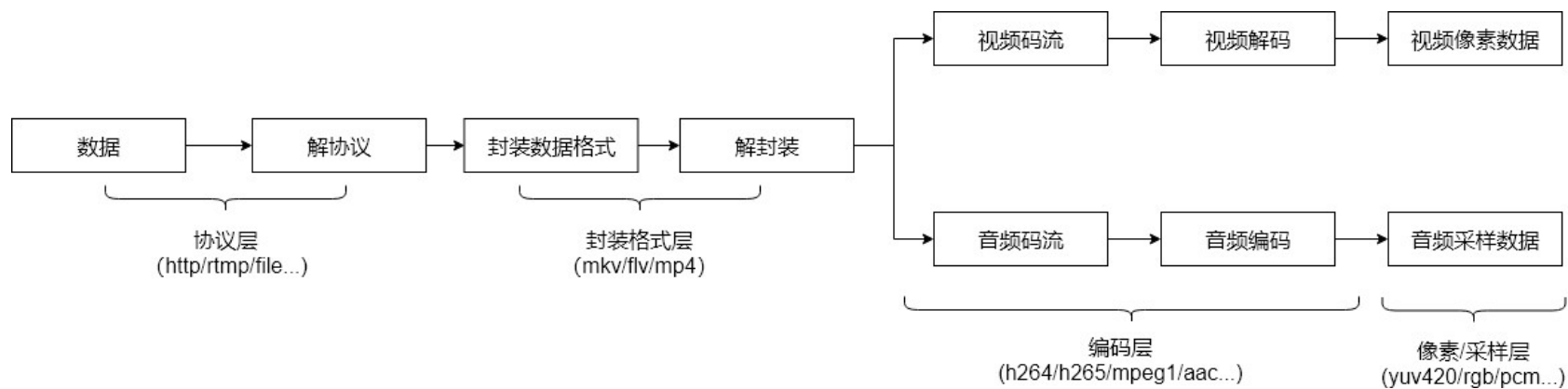
[phoboslab/jsmpeg](https://github.com/phoboslab/jsmpeg)



JSMPEG IS A PHOBOSLAB PRODUCT.
PUBLISHED UNDER THE PERMISSIVE
MIT LICENSE © 2017 D. SZABLEWSKI

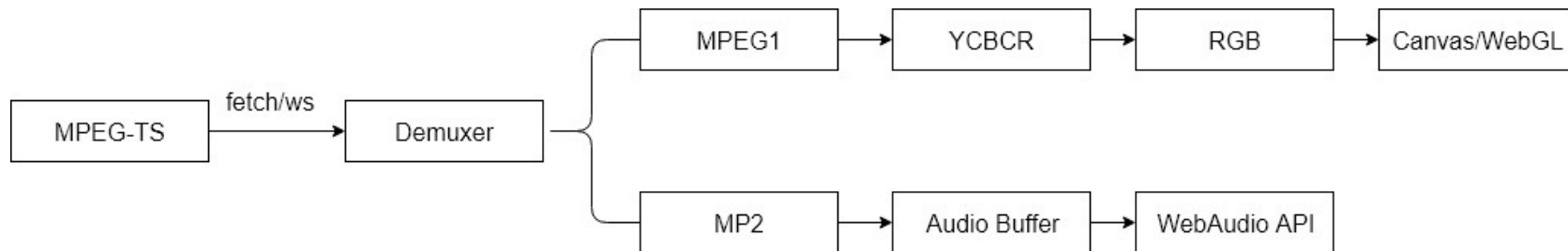


音视频解码的一般过程





JSMPEG执行过程





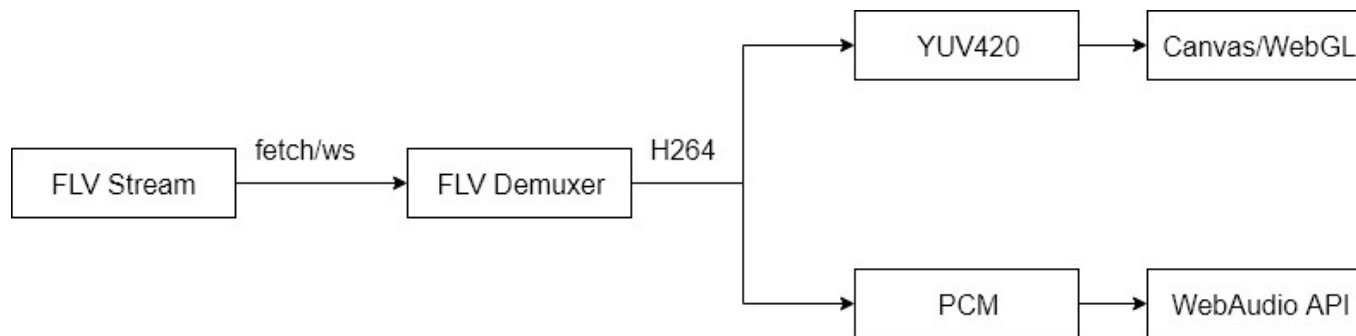
GIAC

全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

从FLV流中用JS解码出音视频像素/采样数据



FLV->H264->像素/采样数据



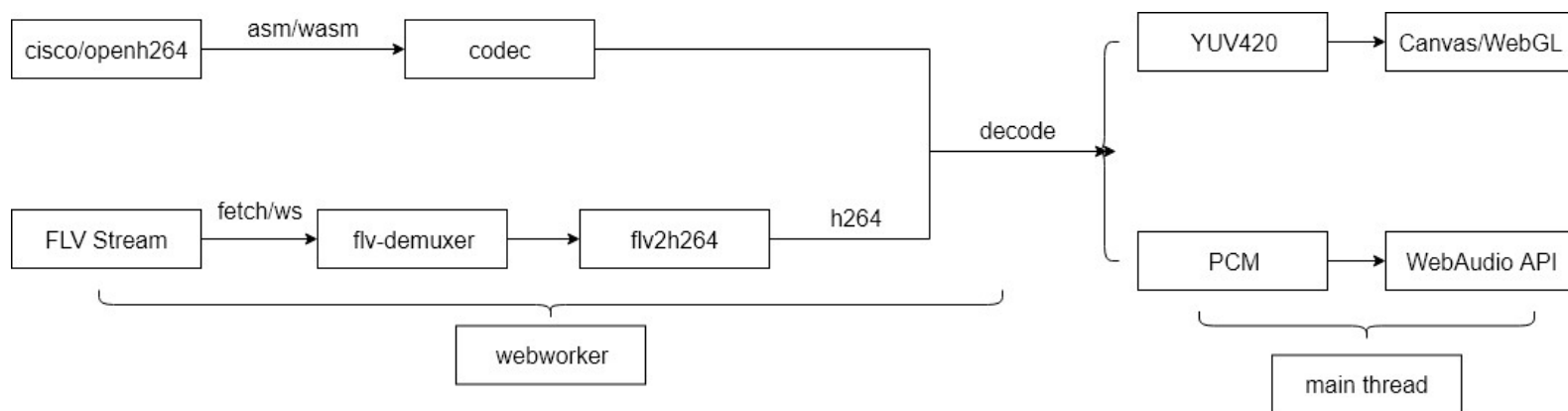


问题与疑问

- 没有成熟开源的H264编解码JS实现
- H264编解码过程复杂，JS实现在移动设备上是否可用



我们的解法





GIAC

全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

测试实例

www.BANDICAM.com

Google



Google Search

I'm Feeling Lucky





测试效果

机型	红米5A	小米6	iPhone6
操作系统	MIUI9	MIUI9	iOS10.3.3
是否支持asm.js	Y	Y	N
FPS最高	55	66	31
FPS最低	46	52	23
内存占用 (MB)	180-200	190-210	未知
CPU占用 (%)	16-21	9-12	未知

测试流参数 (baseline 320x180 128k)



几点思考

- ASM/WASM在移动设备上性能已经可堪重用
- ASM/WASM复用现有C/C++库极为方便
- ASM/WASM有着较为良好的兼容性



GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE

WASM的基础使用

如何从0开始编写WASM



WASM的特点





GIAC

全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

支持编写WASM的语言

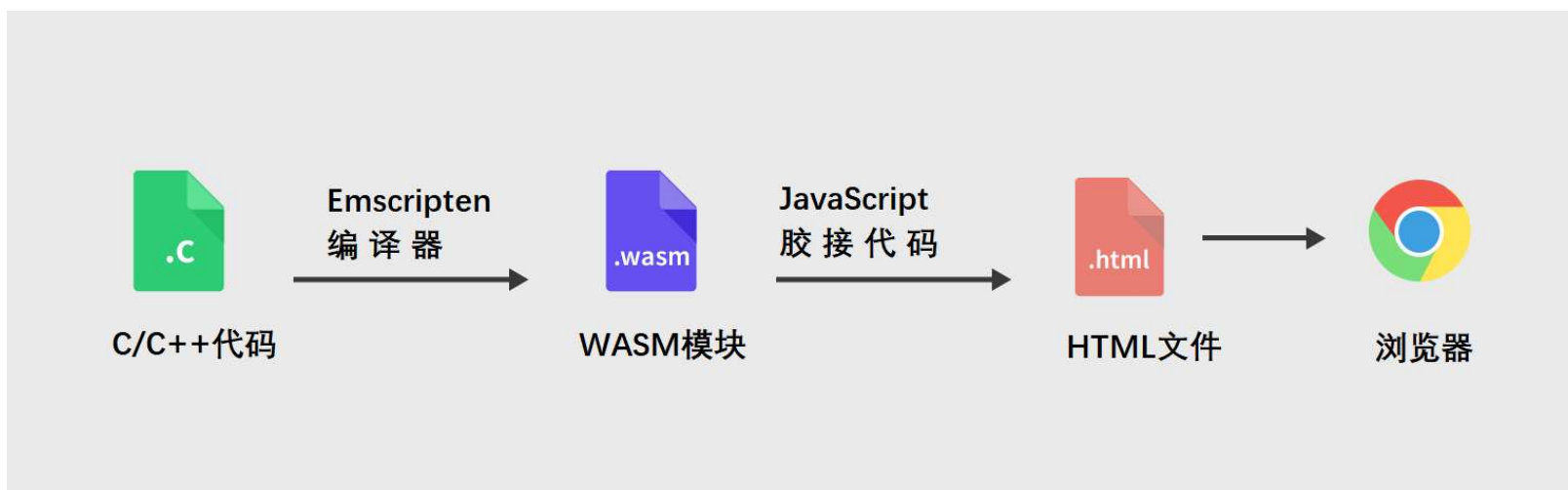




GIAC

全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

编译过程





编写C++代码

```
extern "C" {  
    int sum(int *arr, int length){  
        int ret = 0;  
        for(int i = 0; i < length; i++){  
            ret += arr[i];  
        }  
        return ret;  
    }  
}
```



使用Emscripten编译C++代码

```
#!/usr/bin/env bash
set -x
rm -rf ./build
mkdir ./build

em++ --std=c++11 -s WASM=1 -Os \
    --memory-init-file 0 --closure=1 \
    -s NO_FILESYSTEM=1 -s DISABLE_EXCEPTION_CATCHING=0 \
    -s ELIMINATE_DUPLICATE_FUNCTIONS=1 -s LEGACY_VM_SUPPORT=1 \
    --llvm-lto 1 -s "EXTRA_EXPORTED_RUNTIME_METHODS=['ccall']" \
    -s EXPORTED_FUNCTIONS="['_sum']" \
    ./sum.cpp -o ./build/index.html
```



调用WASM代码

```
function _arrayToHeap(typedArray) {  
    const numBytes = typedArray.length * typedArray.BYTES_PER_ELEMENT;  
    const ptr = Module._malloc(numBytes);  
    const heapBytes = new Uint8Array(Module.HEAPU8.buffer, ptr, numBytes);  
    heapBytes.set(new Uint8Array(typedArray.buffer));  
    return heapBytes;  
}  
  
function _freeArray(heapBytes) {  
    Module._free(heapBytes.byteOffset);  
}
```



调用WASM代码

```
Module.sum = function(intArray) {  
  const heapBytes = _arrayToHeap(intArray);  
  const ret = Module.ccall(  
    "sum",  
    "number",  
    ["number", "number"],  
    [heapBytes.byteOffset, intArray.length]  
  );  
  _freeArray(heapBytes);  
  return ret;  
};  
  
Module.sum(new Int32Array([1, 2, 3, 4])); // output: 10
```



GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE

WASM的实际项目应用

实际应用中的坑和解决办法



GIAC

全球互联网架构大会

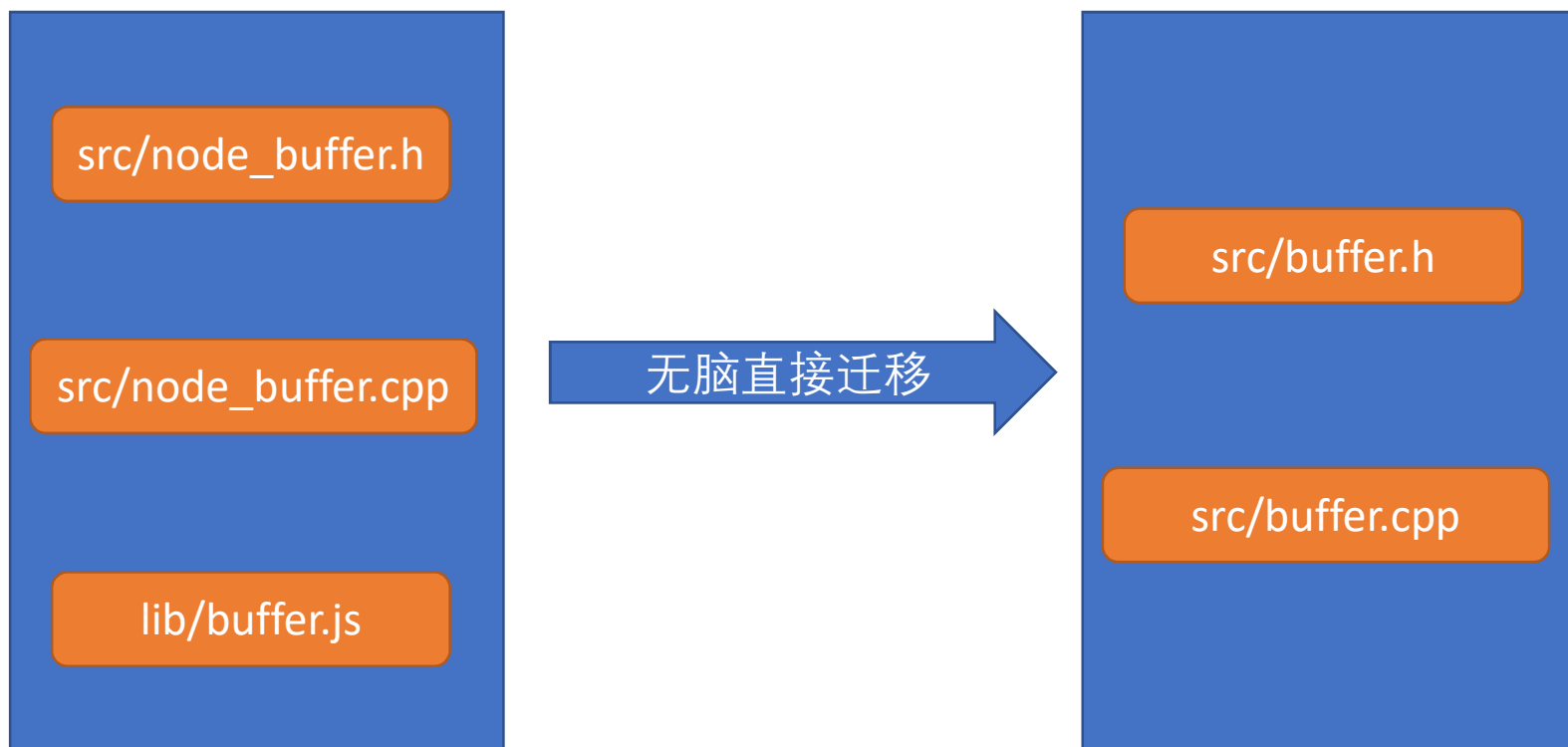
GLOBAL INTERNET ARCHITECTURE CONFERENCE

为什么使用WASM



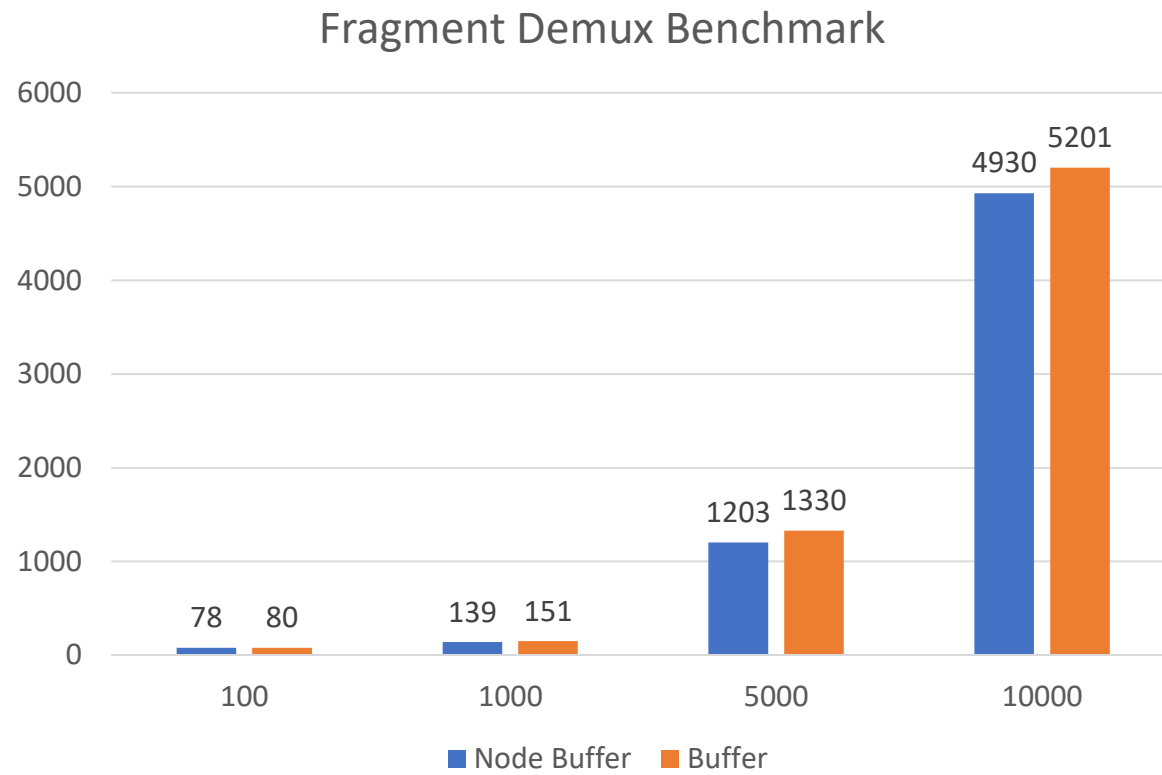


迁移NodeJS代码





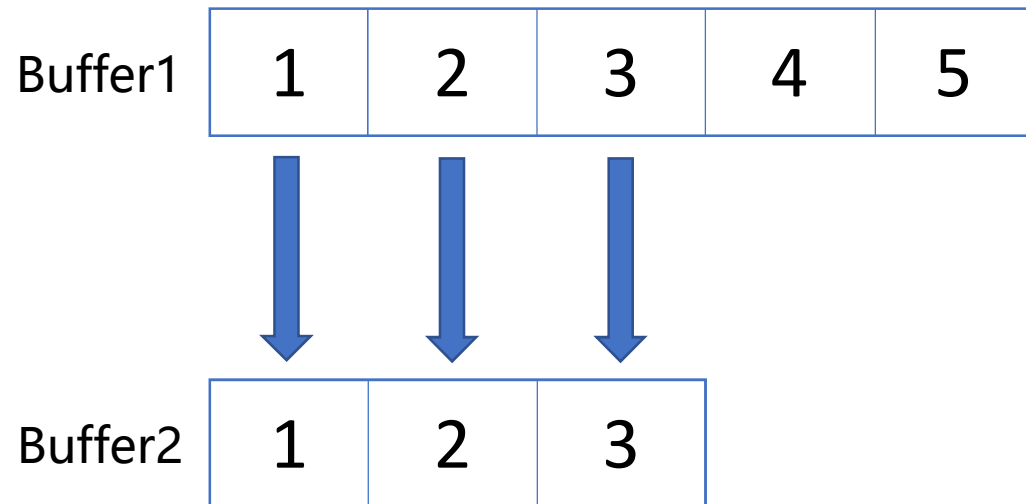
Benchmark





Buffer

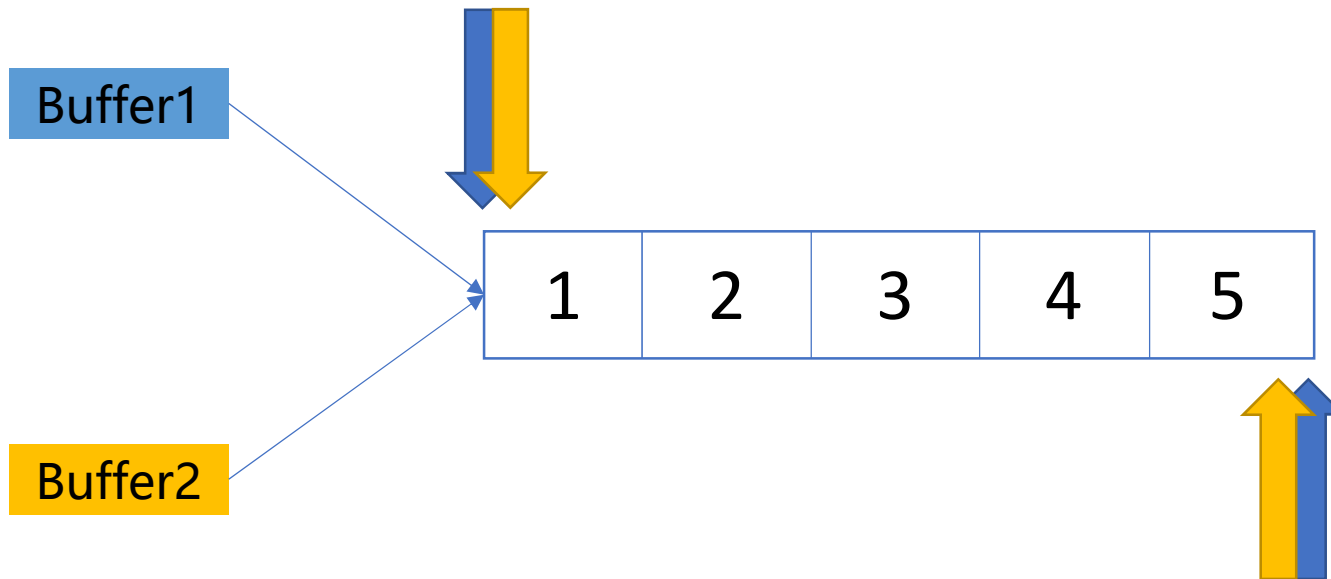
```
auto ptr = new uint8_t[5]{1,2,3,4,5};  
Buffer buf1(ptr);  
auto buf2 = buf1.slice(0, 3);
```





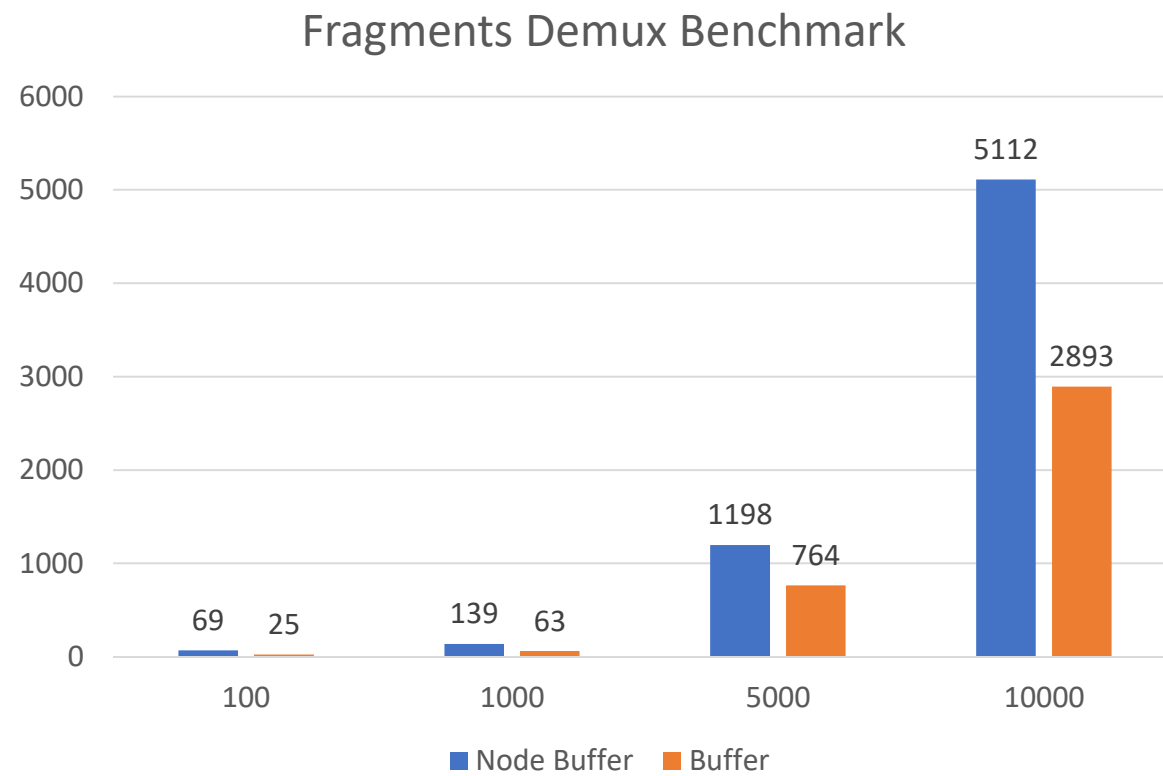
COW Buffer

```
auto ptr = new uint8_t[5]{1,2,3,4,5};  
Buffer buf1(ptr);  
auto buf2 = buf1.slice(0, 3);
```





Benchmark





GIAC

全球互联网架构大会

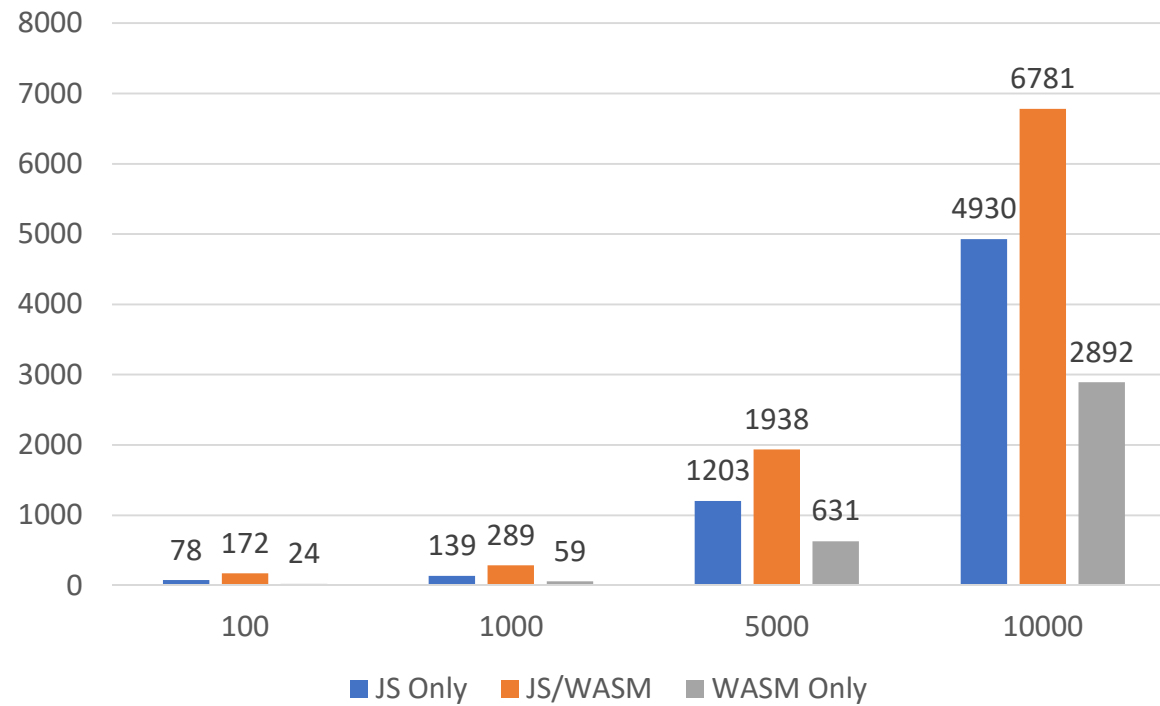
GLOBAL INTERNET ARCHITECTURE CONFERENCE

JS和WASM进行频繁通信调用会造成大量的性能消耗



Benchmark

Fragments Demux Benchmark



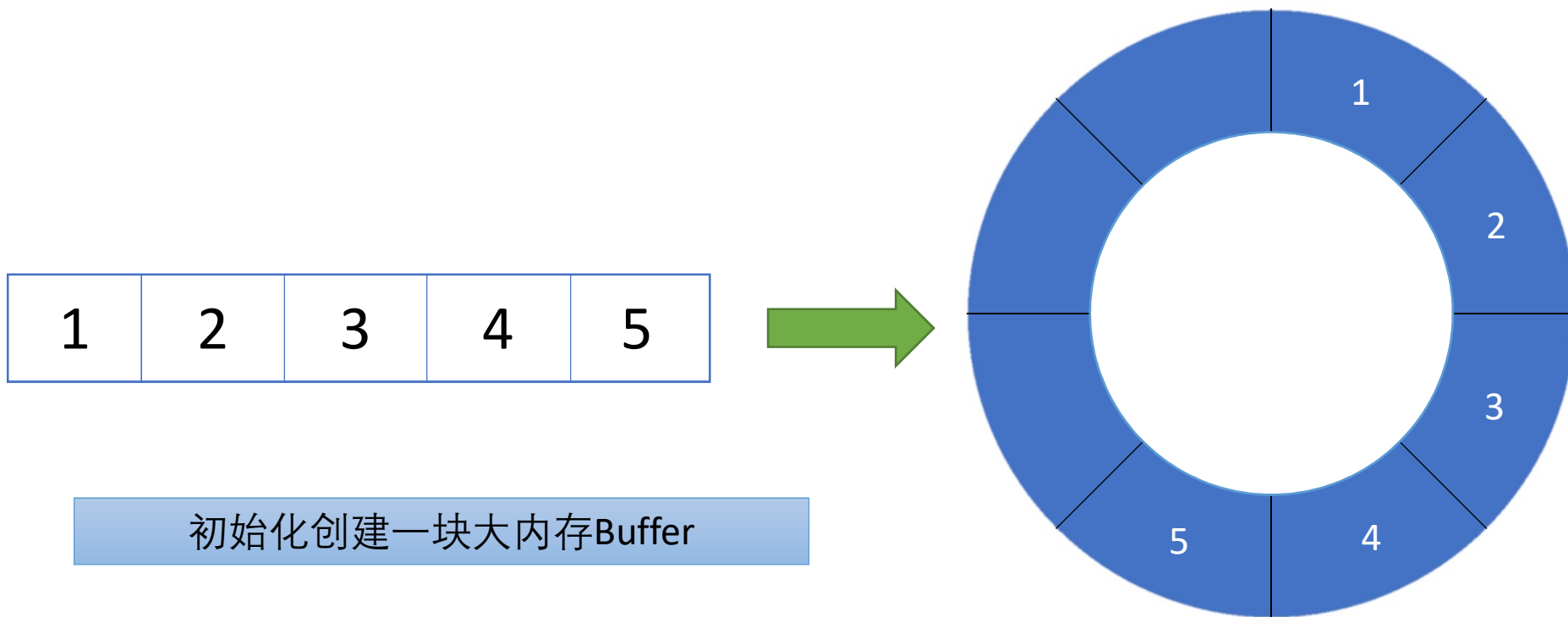


Ring Buffer

```
Module.sum = function(intArray) {  
  const heapBytes = _arrayToHeap(intArray);  
  const ret = Module.ccall(  
    "sum",  
    "number",  
    ["number", "number"],  
    [heapBytes.byteOffset, intArray.length]  
  );  
  _freeArray(heapBytes);  
  return ret;  
};  
  
Module.sum(new Int32Array([1, 2, 3, 4])); // output: 10
```



Ring Buffer



初始化创建一块大内存Buffer

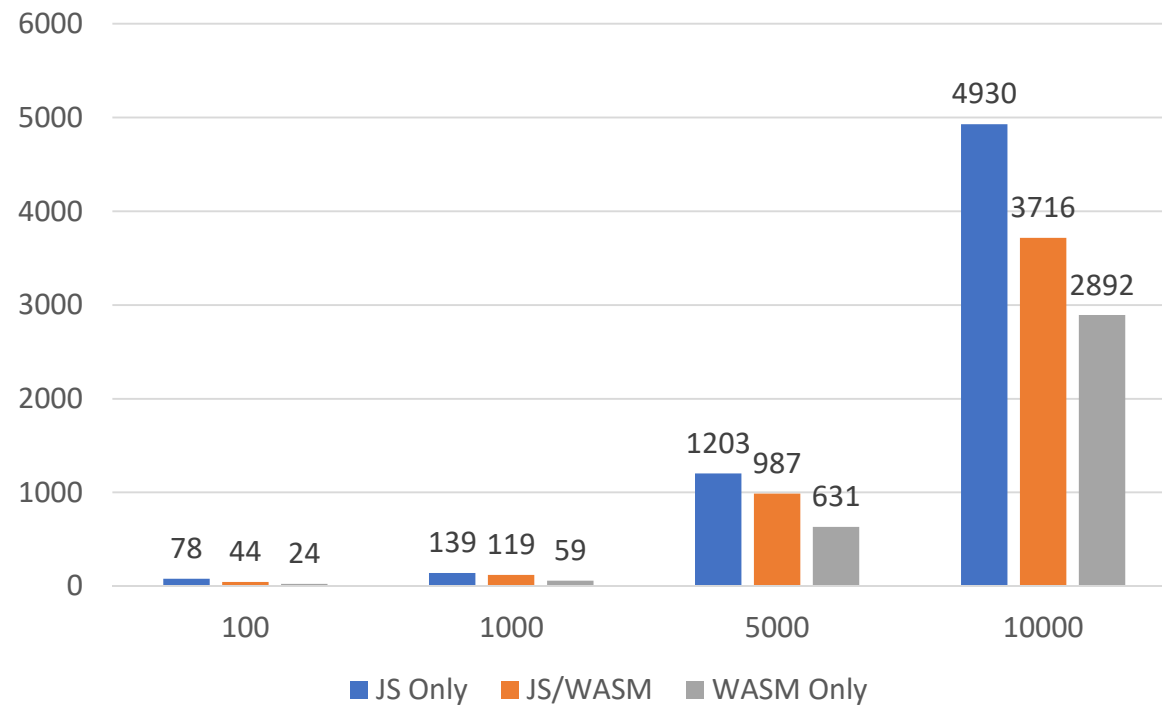
自增ID + 数据大小 + 具体数据

将Buffer视作环进行复用



Benchmark

Fragments Demux Benchmark





GIAC

全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

其他的补充

关于WASM的安全性



GIAC

全球互联网架构大会
GLOBAL INTERNET ARCHITECTURE CONFERENCE

W A S M 的 安 全 性 其 实 只 是 一 层 纱

GIAC

全球互联网架构大会

GLOBAL INTERNET ARCHITECTURE CONFERENCE



关注公众号获得
更多案例实践