

Wrapping up unowned UTF8 C strings: CStringView

WebKit Contributors Meeting 2025
28-29 October 2025

Xabier Rodríguez Calvar
calvaris@igalia.com

Problem

- Some GStreamer APIs return `const char*`.
- We used StringView in our WebKit helpers.
- StringView was span-ed.
- We began to have weird memory warnings in ASan.
- C string null terminators were gone!
- We need to efficiently handle C strings.

Possible solutions

- Keep using StringView:
 - `rawCharacters` is unreliable.
 - String copy when retrieving the characters through CString
- Use spans:
 - No good management of the null terminator and the length
- Something new

New Class

- Different names
 - UnownedString
 - CStringView
- Different implementations:
 - Alias of ASCIILiteral
 - Own class
- Problems with encoding handling
- Big vs small class

Disclaimer

I might be lying!

CStringView

- Own class
- `char8_t` to ensure compiler UTF8 compliance
- Few methods
 - Constructors
 - Null terminator vs length
 - `utf8()` for null terminated `const char*`
 - spans
 - emptiness
- Helpers implemented in StringCommon and StdLibExtras

Next steps

- CString should get UTF8 compliance guaranteed at compile time.
- Maybe renaming CStringView accordingly

Thanks!

