

“Swift Strings”

ನಮಸ್ಕೆ @Paytm Travel 🚌 🚂 ✈️

SERVUS Cześć ಸ HÉ CIAO BONJOUR FIA PRONTO
HELLO SZERVUSZ BONGHJORNU OMING DAR OIá AIIó
HALOO ಹಿ DIA DUIT SALVE

- name[0]

+ name.prefix(l)

~~umbrella[5]~~ No random access. Why?

string1.count = 1

string2.count = 1

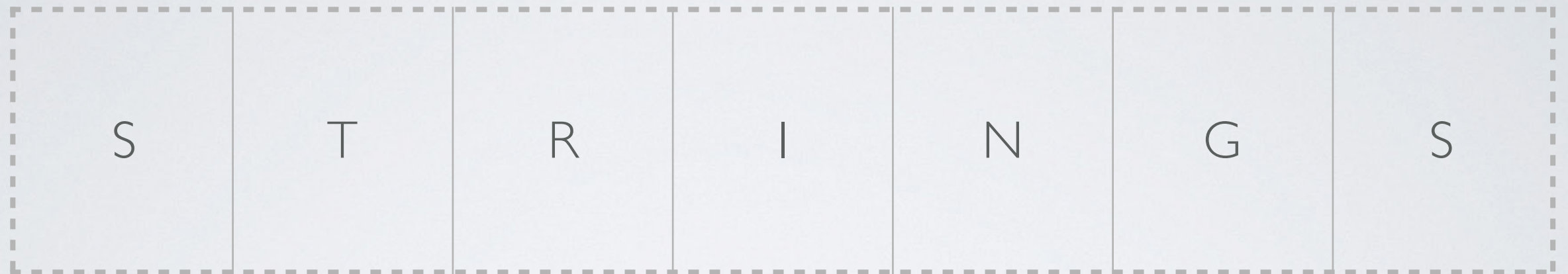
string3 = string1 + string2

string3.count = 1. Possible? Why

How String in Swift5 is efficient?

Why there is Substring type?

 is not visible in windows and android. Why?



Collection of characters

Literals

```
let greeting = "Hello"
```

Interpolation

```
let side = 5
```

```
let area = "Area of the square is \((side*side)\)"
```

swift/stdlib/public/core/String.swift

```
public struct String { }
```

```
public protocol StringProtocol :  
    BidirectionalCollection,  
    Comparable,  
    ExpressibleByStringInterpolation,  
    Hashable,  
    ...  
  
where Self.Element == Character,  
    ...  
Self.SubSequence : StringProtocol {}
```

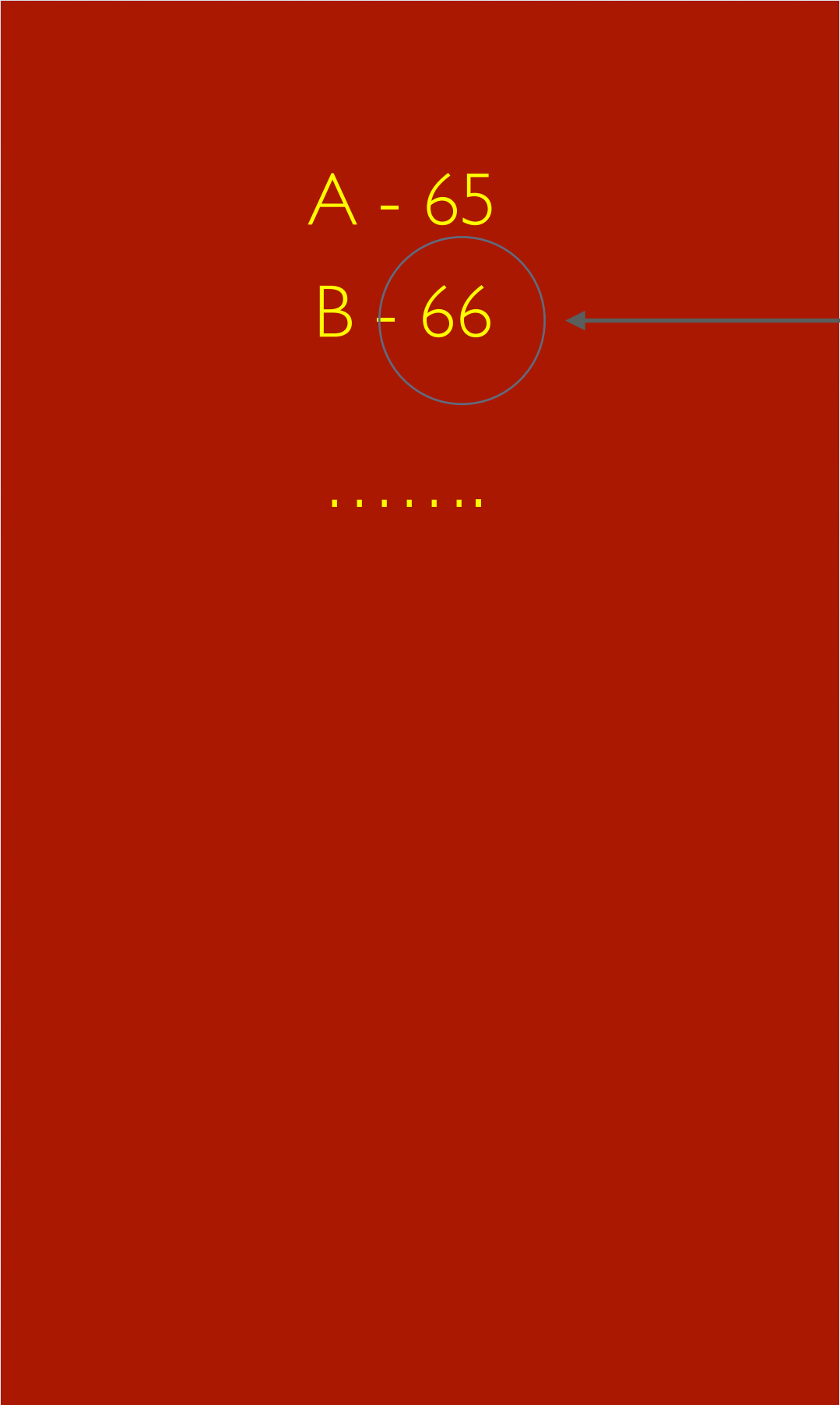
```
public struct Substring { }
```




11110000 10011111
10011000 10000000

ASCII

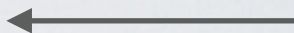
0	011 0000	48
1	011 0001	49
A	100 0001	65
B	100 0010	66
Z	101 1010	90
a	110 0001	97
b	110 0010	98



A - 65

B - 66

.....

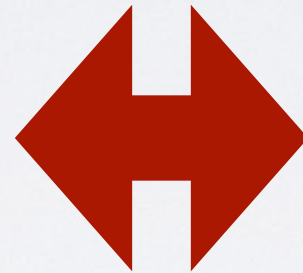


Code space



Code point

We are building the
go-to destination for
all your travel needs



0008700101 000970011400101
000980011700105001080010000
1050011000103
001160010400101
0010300111000450011600111
001000010100115001160010500
110000970011600105001110011
0 001020011100114
000970010800108
00121001110011700114
001160011400097001180010100
108
0011000101001010010000115

Unicode

Unicode provides a unique number for every character,

no matter what the platform,
no matter what the program,
no matter what the language.

Unicode

137,993 characters covering 150 modern and
historic scripts

The Unicode Consortium

Full Members (**Voting**)



NETFLIX

ORACLE



Institutional Members (**Voting**)



Berkeley
University of California

Supporting Members (**Voting**)





Unicode

U+50 U+61 U+79 U+74 U+6D U+20 U+2708

Decimal

80 97 121 116 109 32 9992

Binary

1010000 1100001 1111001 1110100 1101101 100000
10011100001000



11111011000000001

17 bits

U+1F601

32 bits



16*2 bits



8*4 bits





|||||0||00000000| 17 bits

UTF-32



UTF-16



UTF-8





|||||0||00000000|

17 bits

UTF-32

128513

UTF-16

55357

56833

UTF-8

240

159

152

129

```
let smile = "😊"
```

```
//32 bit unicode
```

```
for unicodeScalar in smile.unicodeScalars{  
    print(unicodeScalar.value)  
}
```

```
//16 bit code units
```

```
for codeUnit in smile.utf16{  
    print(codeUnit)  
}
```

```
//8 bit code units
```

```
for codeUnit in smile.utf8{  
    print(codeUnit)  
}
```

```
let smile = "😊"
```

=

```
let smile = "\u{1f601}"
```

```
let kannadaLetter = "\u{ca4}"
```

```
print(kannadaLetter) //ತ
```

```
print(kannadaLetter.count) //1
```

```
print(kannadaLetter.unicodeScalars.count) //1
```

```
let kannadaLetter = "\u{ca4}" + "\u{ccd}"
```

```
print(kannadaLetter) //ತೆ
```

```
print(kannadaLetter.count) //1
```

```
print(kannadaLetter.unicodeScalars.count) //2
```



```
let kannadaLetter = "\u{ca4}\u{ccd}"
```

Grapheme Cluster

\u{1F44D}



\u{1F3FF}+\u{1F3FB}



\u{1F3FF}+\u{1F3FF}



`string.count`
is not always equal
`string.unicodeScalars.count`.

count is $O(n)$

avoid `string.count > 0`

Use `string.isEmpty == false`

protocol **RandomAccessCollection**

Random-access collections can move indices any distance and measure the distance between indices in **$O(1)$** time.

String is a collection but is not a
RandomAccessCollection

So subscripting using index is not allowed

~~umbrella[5]~~

Index API

```
let umbrella = "☂Umbrella"
```

```
let indexOfR = umbrella.index(umbrella.startIndex,  
offsetBy: 4)
```

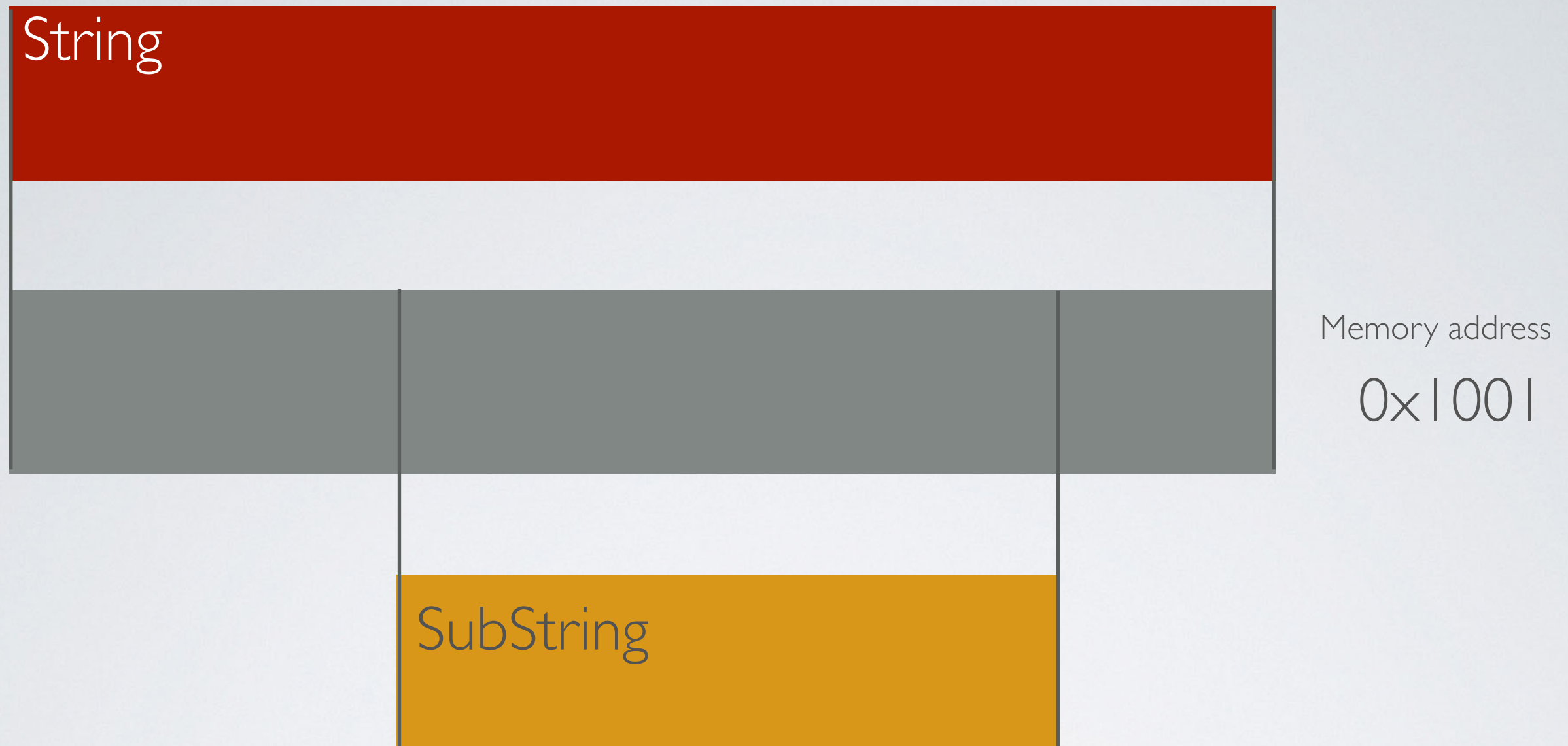
```
umbrella[indexOfR]
```

```
let umbrella = "☂Umbrella"  
let index0fU = umbrella.index(umbrella.startIndex,  
offsetBy:1)
```

```
let umbrellaWithoutEmoji =  
umbrella[index0fU..  
umbrella.endIndex]
```

```
umbrellaWithoutEmoji is String      False
```

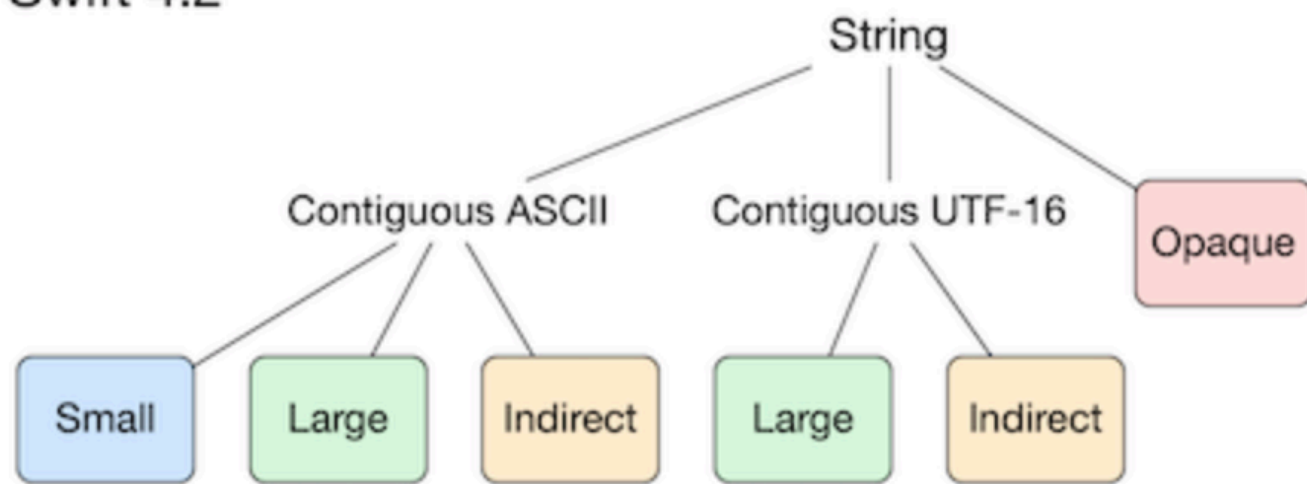
```
umbrellaWithoutEmoji is Substring  True
```



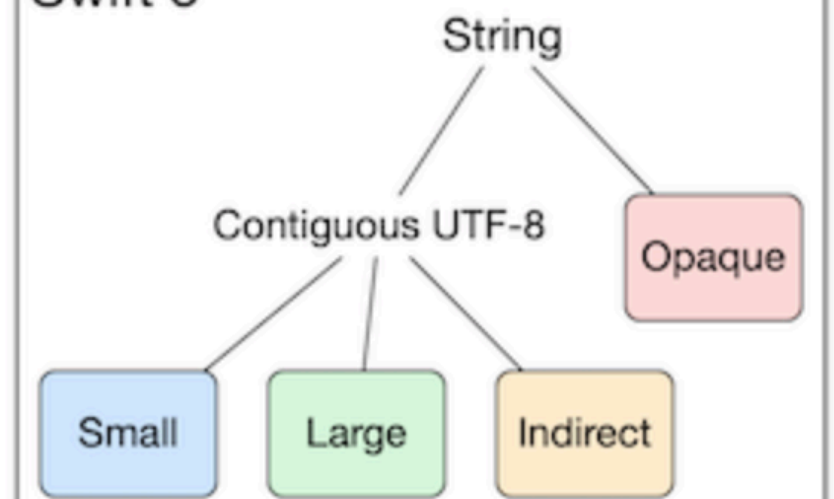
Substring shares its storage with the original string

Swift 5

Swift 4.2



Swift 5



Benefits

- Unified Storage Representation
- C Interoperability
- Unicode Small Strings
- Server-side and Systems programming

Thank you

