

72 bits = 9 bytes

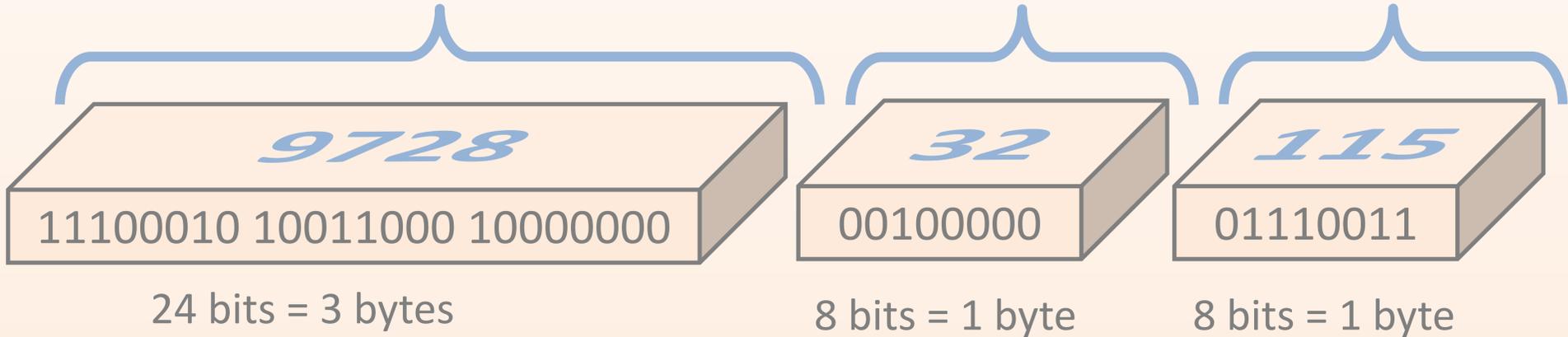
"☀️ sunny" =

11100010 10011000 10000000 00100000 01110011
01110101 01101110 01101110 01111001

{☀️}

{ }

{S}

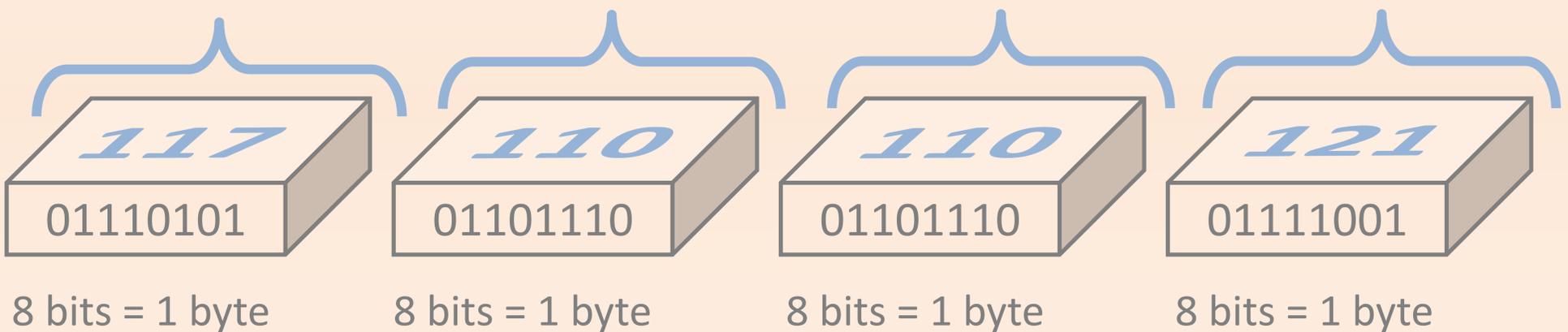


{u}

{n}

{n}

{y}



Size of text according to UTF-8. It shows the size of text "☀️ sunny" under UTF-8. This further includes the code points (the whole number associated with a symbol), the corresponding bit sequences and the actual symbols associated with these abstract representations. Note that boxes indicate abstract regions of physical memory. The space character and the letters that make up the word "sunny" take up a total of 6 bytes, however, the sun symbol is new and is encoded in 3 bytes instead of 1 byte. This observation is in fact very important. Usually, the most necessary symbols were those that were first introduced as characters in the development of computers over time. Consequently, time precedence of characters is directly proportional to their frequency of occurrence in data. Thus, preservation of the initial encoding for the most frequent symbols dictates the conservation of file size. UTF-8 characters can be represented by 1 byte for older legacy symbols, up to 4 bytes for newer symbols. This is one of the main reasons why UTF-8 is crucial to the future of technology when compared to other character encodings.