

We want to show that: „Every symbol from a given alphabet can be represented as a bitstring. This representation is one-to-one and only scales logarithmically in alphabet size.“ Consider an alphabet A with n symbols where each symbol $a_0, a_1, \dots, a_{n-1}$ can be represented by a unique integer. Because we can use the index of an element as its integer counterpart, at least one such representation exists

We can uniquely represent an integer c as a bitstring b through the following algorithm: Zero all bits in b . Find the largest x such that 2^x divides c . Set b_x to one and set c to $c - 2^x$. Continue until c is zero. The length of such a bitstring is determined by the x of the first iteration, so the first x such that $2^x \mid c$. The length of the resulting bitstring will thus be $\lfloor \log_2(c) \rfloor$, hence the logarithmic scaling in the size of the alphabet, 2.