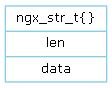
Nginx Infrastructure

# 概述

暂无

# ngx\_str\_t

实现一个字符串。

* len

字符串的长度，不包含结尾NULL。有了这个字段，就不用每次调用strlen()计算字符串的长度。

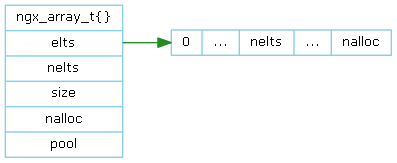
这是提高性能的PREPROCESS原则的应用。

* data

指向字符串，不保证字符串以NULL结尾。

它通常是大字符串中的一个子串，为了减少或避免内存Copy，只是指针引用，所以不能保证以NULL结尾。

# ngx\_array\_t

实现一个数组，优点是下标寻址速度快，缺点是当申请的空间不够用时，需整体重新申请整体复制。

* elts

指向数组内存空间。

* nelts

数组有多少个有效元素。

* size

单个元素大小。

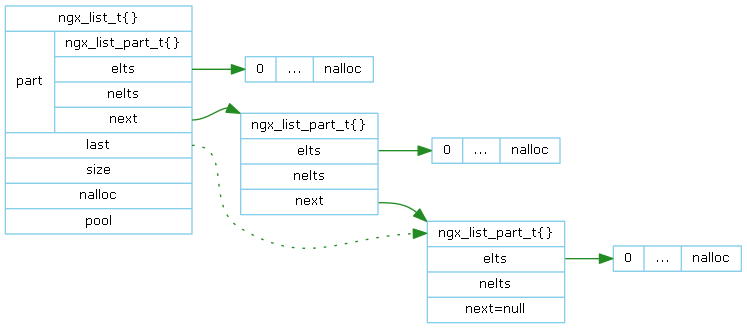
* nalloc

已申请了多少个元素空间。

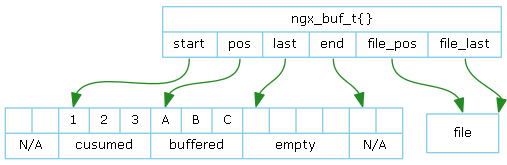
* pool

内存池。

# ngx\_list\_t



# ngx\_buf\_t



实现一个缓冲区。

* start / end

start指向缓冲区内存的开始（闭区间），end指向结尾（开区间）。

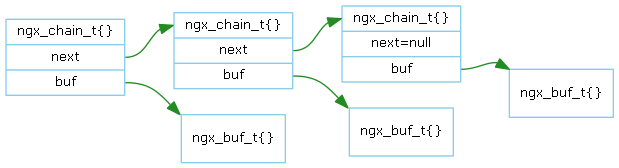
* pos

pos指向已处理的结尾。

* last

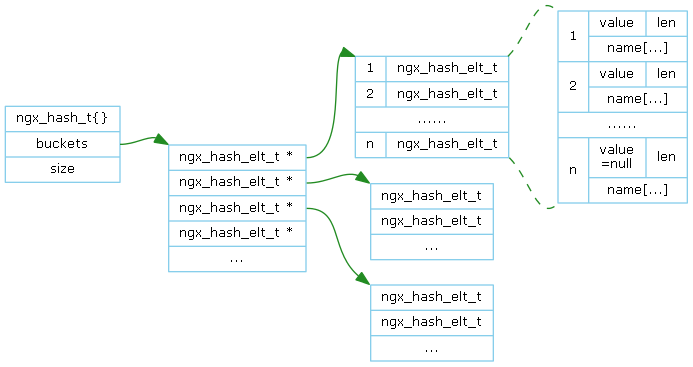
last指向已读入的结尾。

# ngx\_chain\_t



# nginx hash

## ngx\_hash\_t

ngx\_hash\_t实现一个静态HASH表。

* buckets

指向HASH数组

* size

HASH数组的大小

ngx\_hash\_elt\_t表示一个元素。

* value

元素值。NULL值有特殊的含义，表示KLV结束。

* len

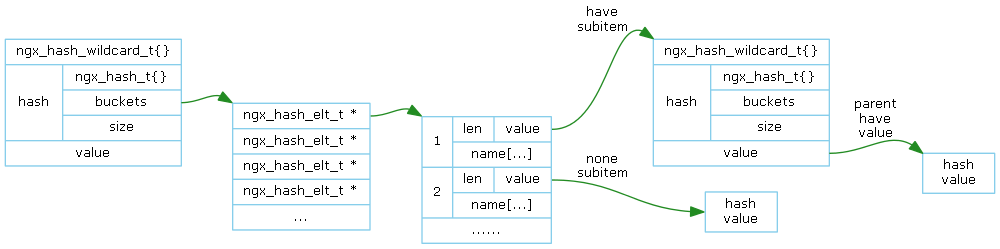
Name的长度

* name

可变长度的字符串Key

Nginx HASH表的实现不采用通常的单向链表的方式，而是采用KLV（key-length-value）方式。相比于链表方式，KLV可以减少TLB计算量。

## ngx\_hash\_wildcard\_t



# Prototype

暂无

# Latest revision

https://github.com/lingjf/nginx\_analyse/blob/master/doc/