


# bytefile layout

```
int sizeof(size_t),  
int numTax,  
size_t numPattern,  
int numPartitions,  
double gappyness
```

} header

int weights[numPattern]




```
int len1,  
char taxonName[len1],  
int len2,  
char taxonName[len2],  
...
```

} taxon names

```
partition1 {  
int states,  
int maxTipStates,  
size_t lower,  
size_t upper,  
size_t width, (unused)  
int dataType,  
int protModels,  
int autoProtModels,  
int protFreqs,  
boolean nonGTR,  
boolean optimizeBaseFrequencies,  
int numberOfCategories,  
int len,  
char partitionName[len],  
double frequencies[states]  
}  
partition 2 {  
....  
}  
....
```

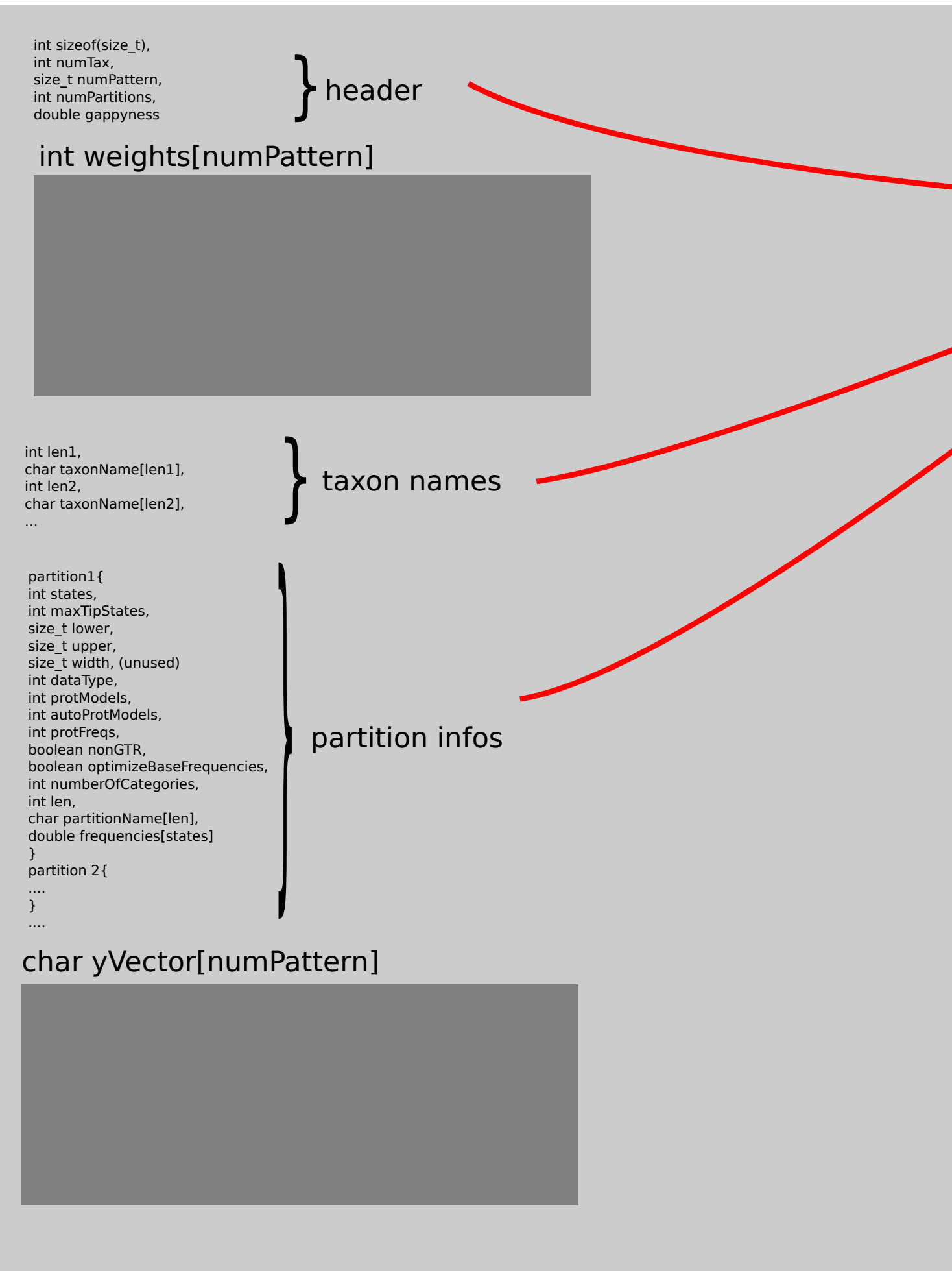
} partition infos

char yVector[numPattern]



# bytefile layout

1. read header, taxa, partitions into ByteFile struct  
(use seekPos() to navigate in bytefile)



## ByteFile \*bFile



# bytefile layout

```
int sizeof(size_t),
int numTax,
size_t numPattern,
int numPartitions,
double gappyness
```

} header

```
int weights[numPattern]
```



```
int len1,
char taxonName[len1],
int len2,
char taxonName[len2],
...
```

} taxon names

```
partition1{
int states,
int maxTipStates,
size_t lower,
size_t upper,
size_t width, (unused)
int dataType,
int protModels,
int autoProtModels,
int protFreqs,
boolean nonGTR,
boolean optimizeBaseFrequencies,
int numberOfCategories,
int len,
char partitionName[len],
double frequencies[states]
}
partition 2{
....
}
....
```

} partition infos

```
char yVector[numPattern]
```

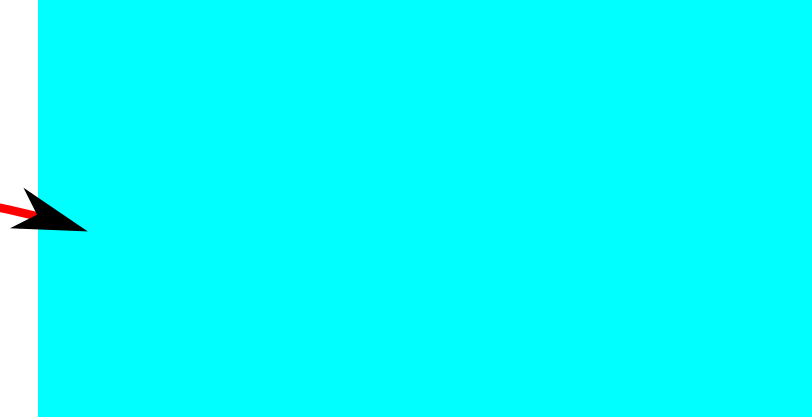


## 2. every process computes partition assignment

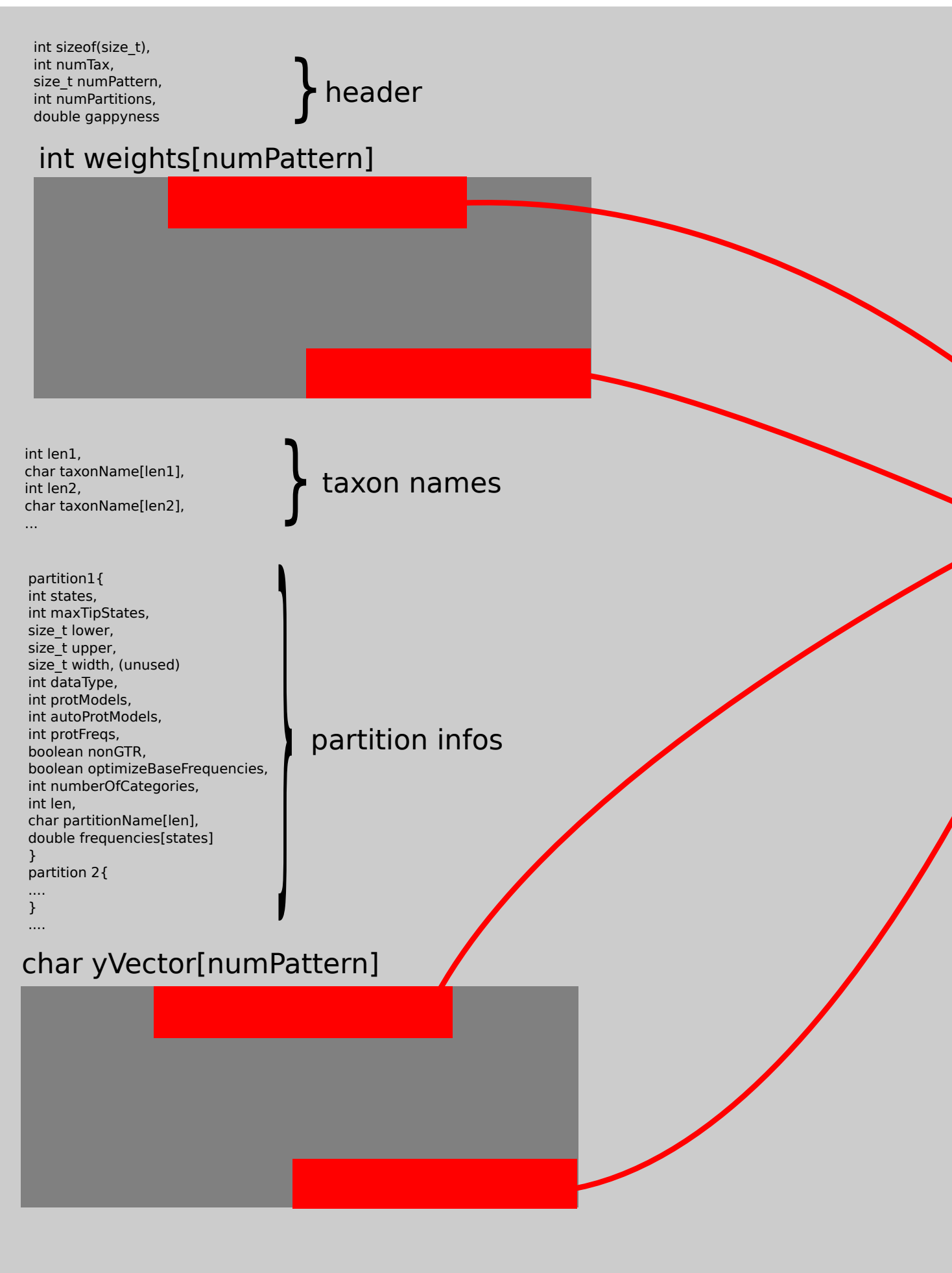
ByteFile \*bFile



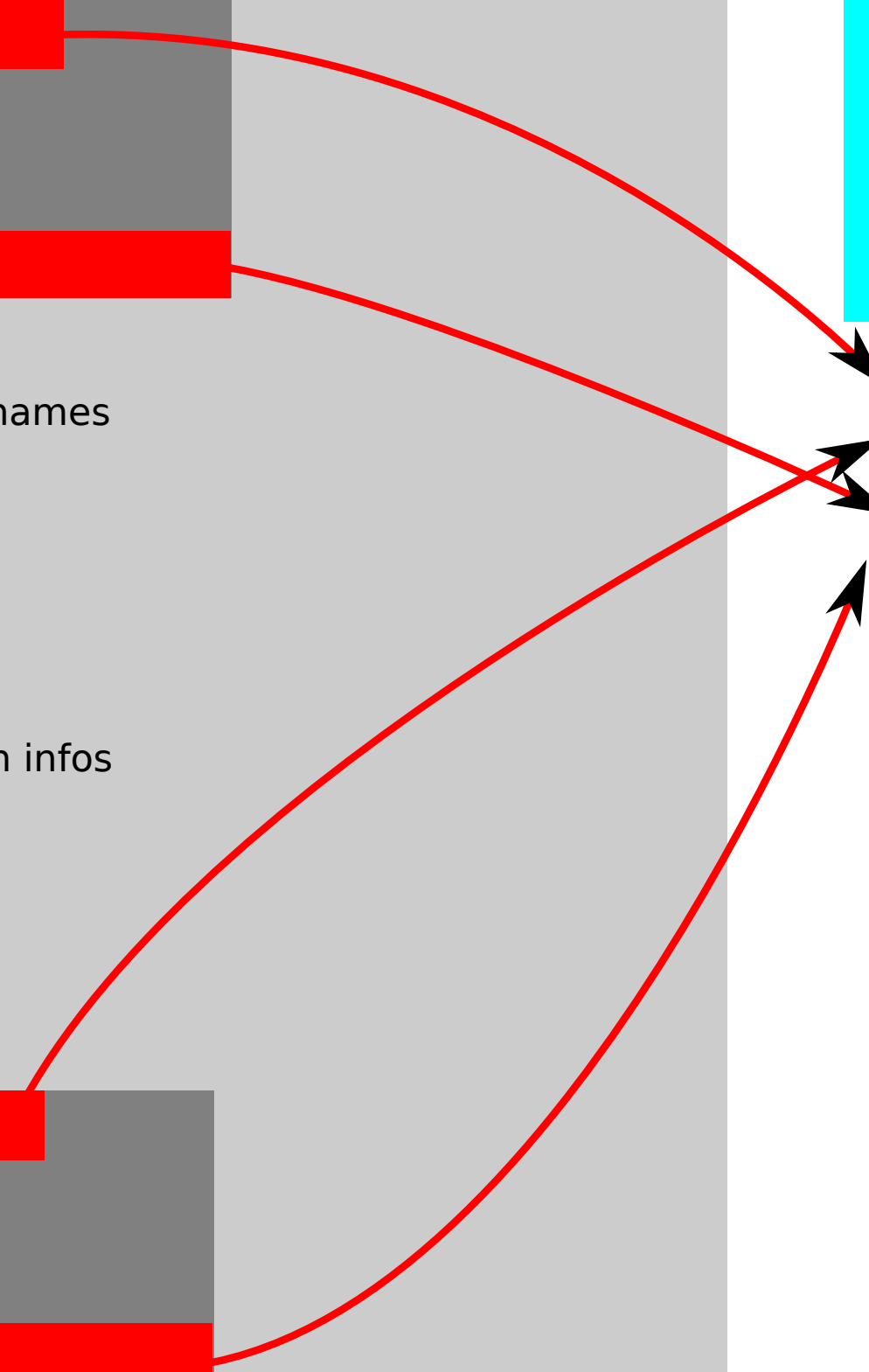
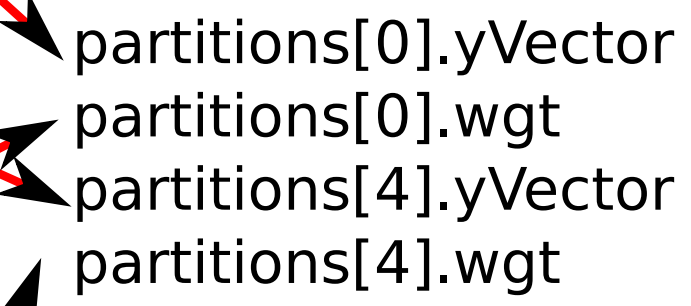
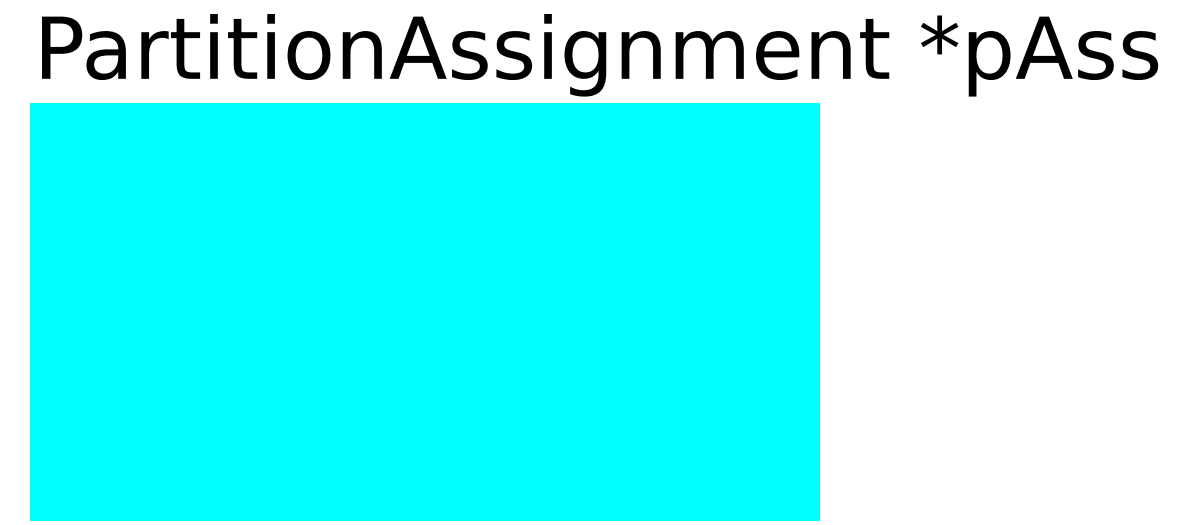
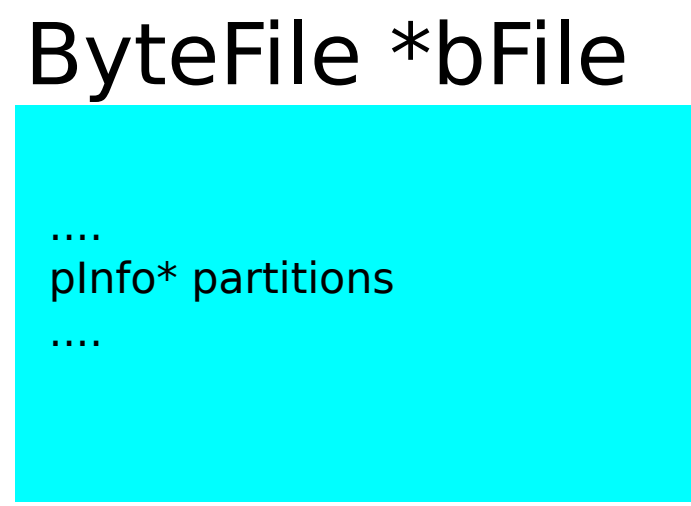
PartitionAssignment \*pAss



# bytefile layout



3. process only reads data assigned to it (exa\_fread/exa\_fseek)



# bytefile layout

```
int sizeof(size_t),  
int numTax,  
size_t numPattern,  
int numPartitions,  
double gappyness  
} header
```

```
int weights[numPattern]
```

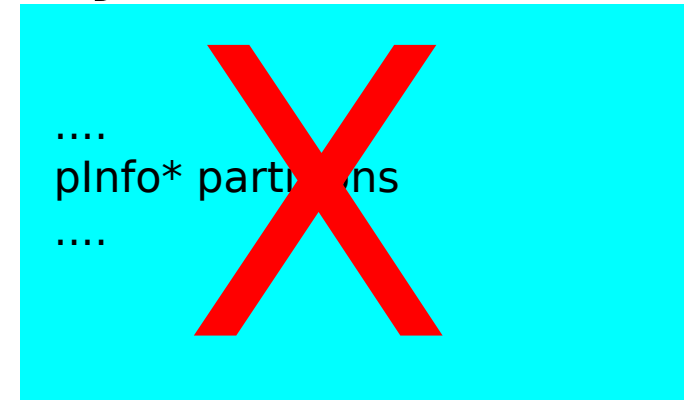
```
int len1,  
char taxonName[len1],  
int len2,  
char taxonName[len2],  
...
```

```
partition1 {  
int states,  
int maxTipStates,  
size_t lower,  
size_t upper,  
size_t width, (unused)  
int dataType,  
int protModels,  
int autoProtModels,  
int protFreqs,  
boolean nonGTR,  
boolean optimizeBaseFrequencies,  
int numberOfCategories,  
int len,  
char partitionName[len],  
double frequencies[states]  
}  
partition 2 {  
....  
}  
....
```

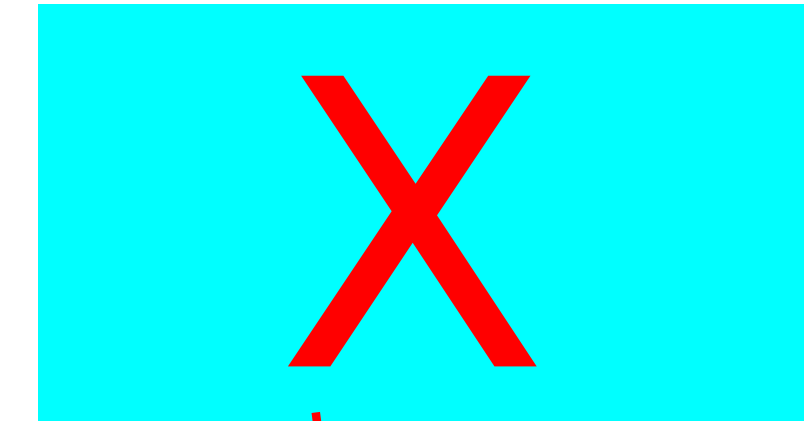
```
char yVector[numPattern]
```

## 4. tree struct is initialized; bFile and pAss are deleted

ByteFile \*bFile



PartitionAssignment \*pAss



tree \*tr

