

FragPicker: A New Defragmentation Tool for Modern Storage Devices

Jonggyu Park and Young Ik Eom

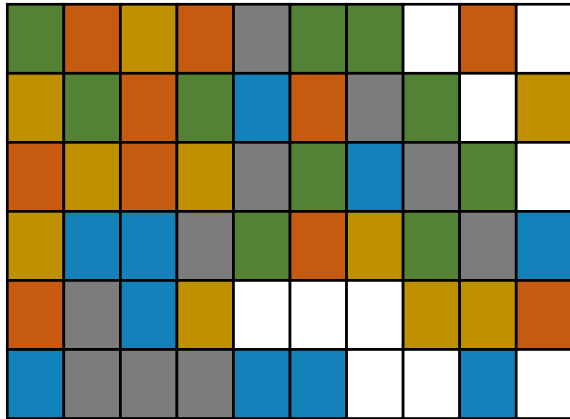
Sungkyunkwan University



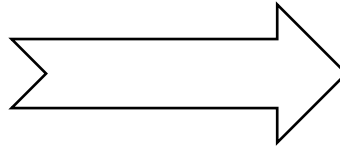
What is fragmentation?

a.k.a file fragmentation, filesystem fragmentation, and filesystem aging

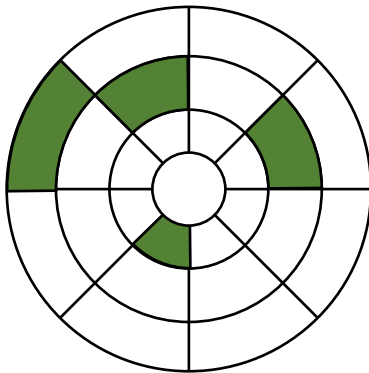
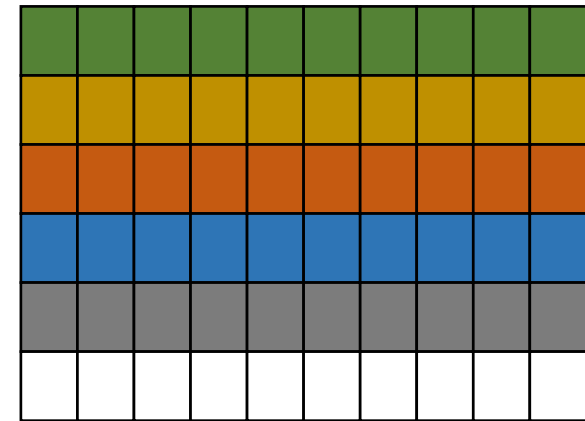
Filesystem view



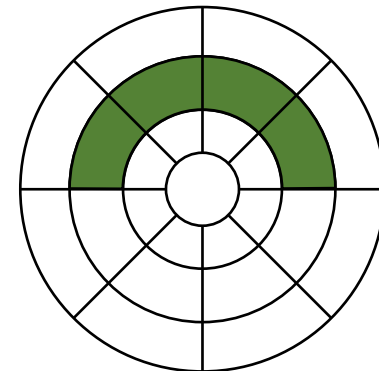
Defragment!



Filesystem view



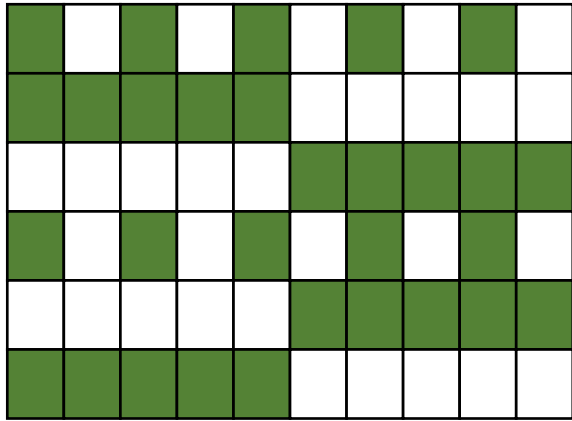
Storage view



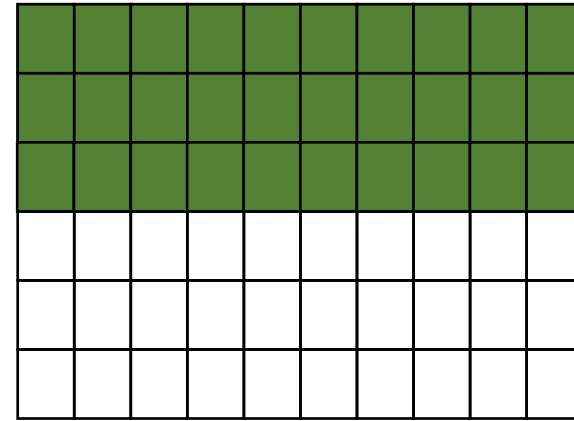
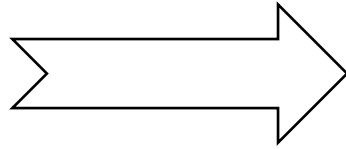
Storage view

Let's revisit defragmentation tools

Conventional defragmentation tools migrates the entire contents of files



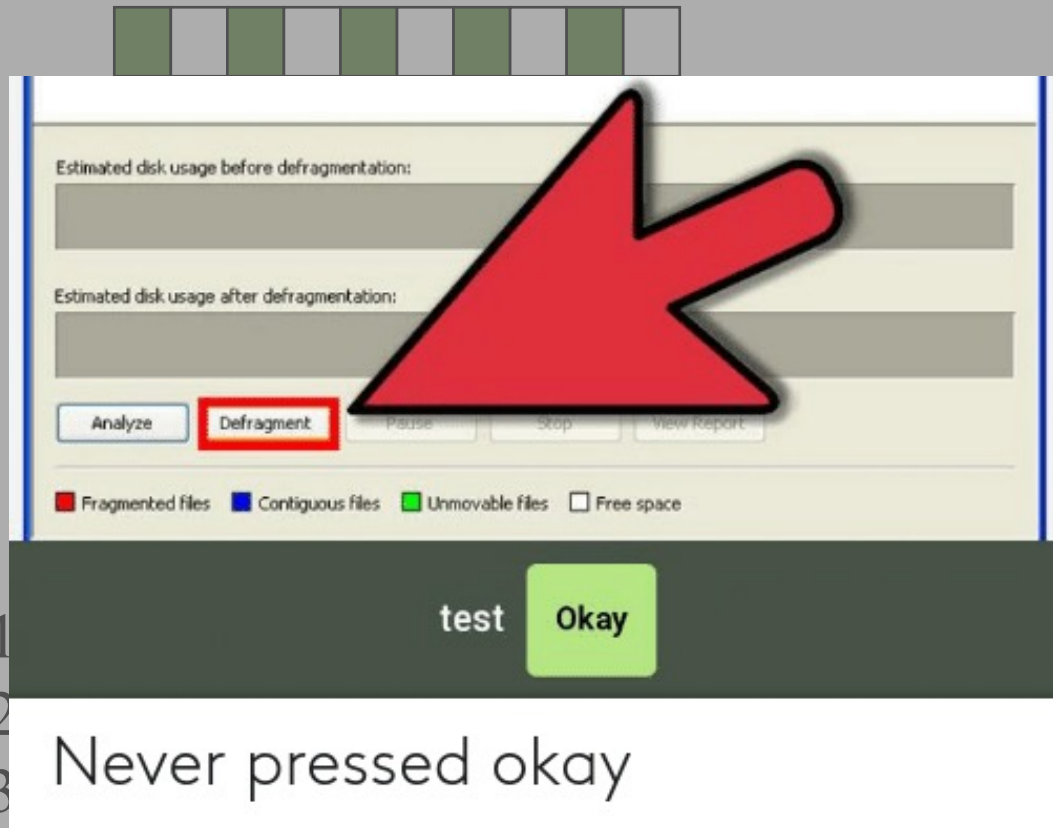
Defragment!



1. Significantly degrades the performance of co-running applications
2. The additional writes reduce the lifespan of modern storage devices.
3. Time-consuming

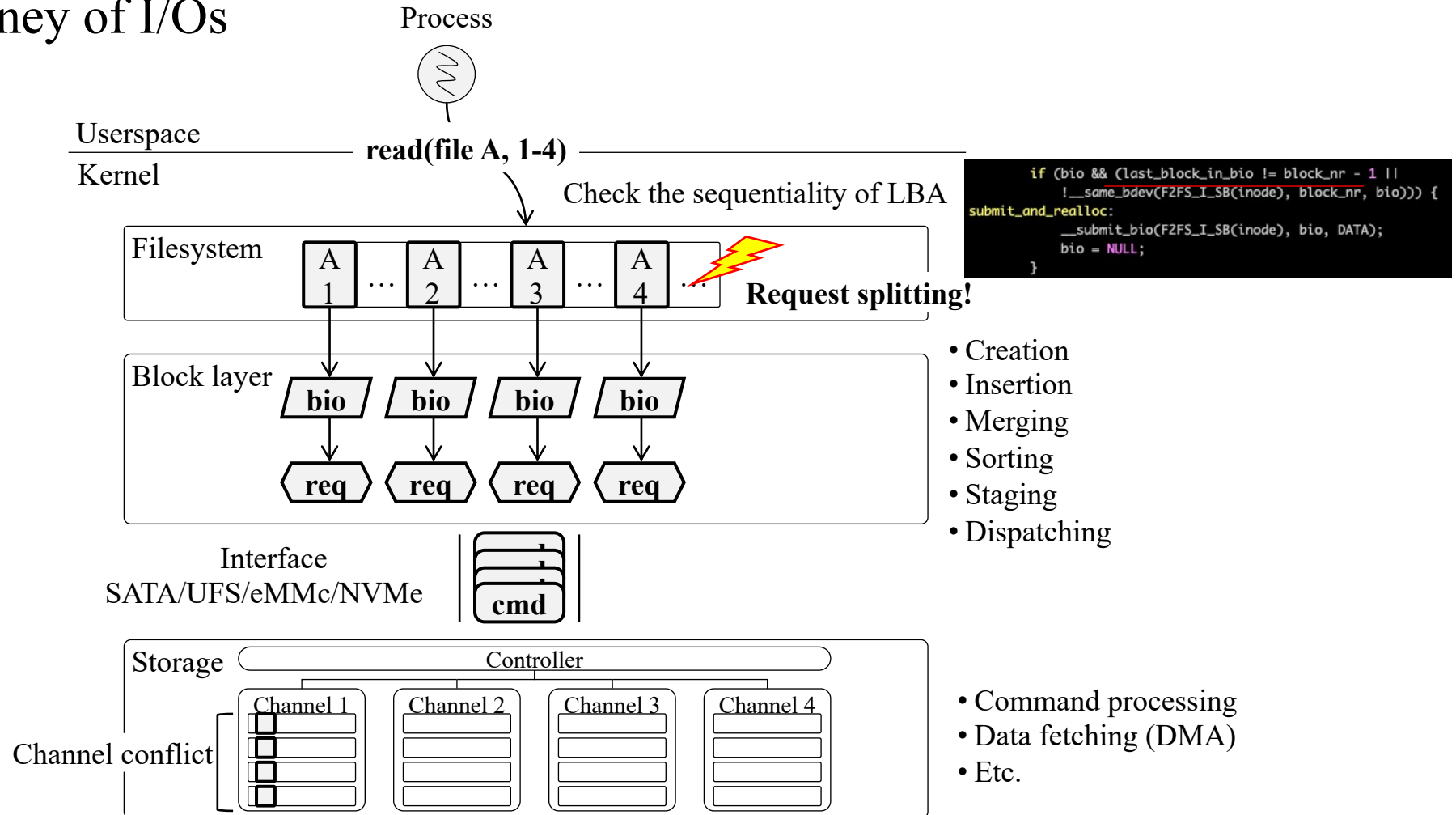
Let's revisit defragmentation tools

Conventional defragmentation tools migrates the entire contents of files



Fragmentation on modern storage devices

Let's follow the journey of I/Os

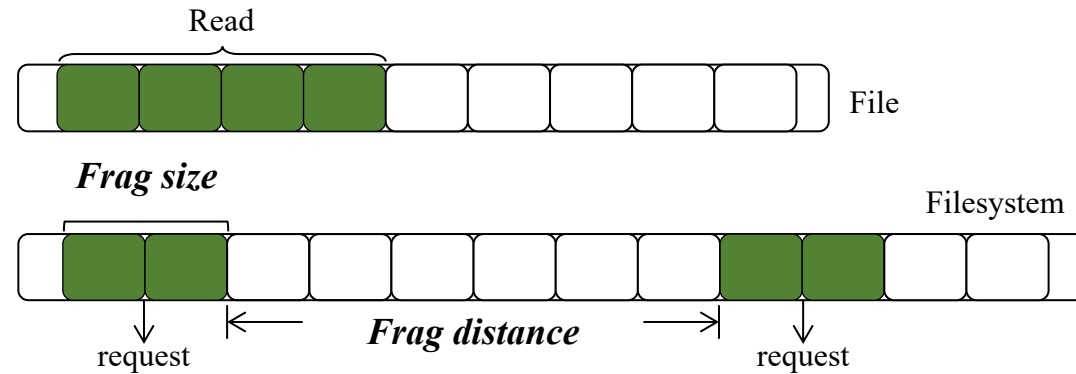


Let's verify this “scientifically”

New terminology

Frag size: *the size of each fragment that are contiguous in terms of LBA*

Frag distance: *the distance between two consecutive fragments*



Let's verify this “scientifically”

Evaluation Setup.

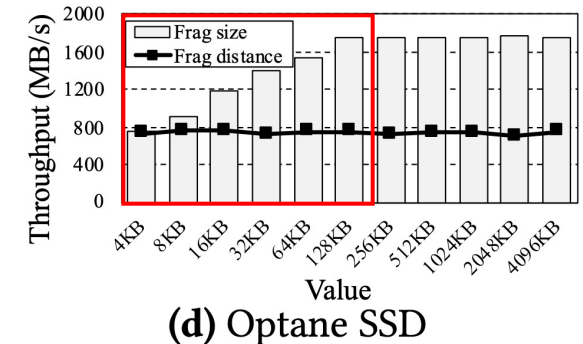
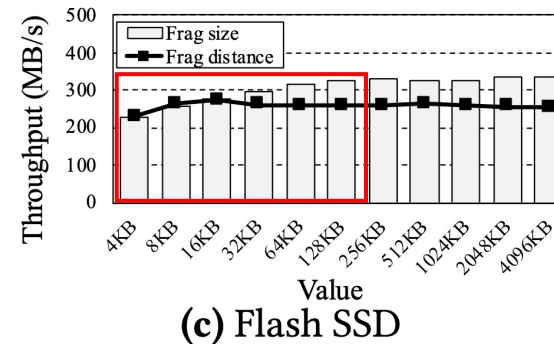
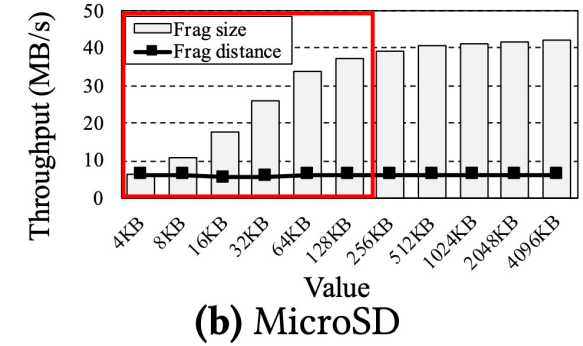
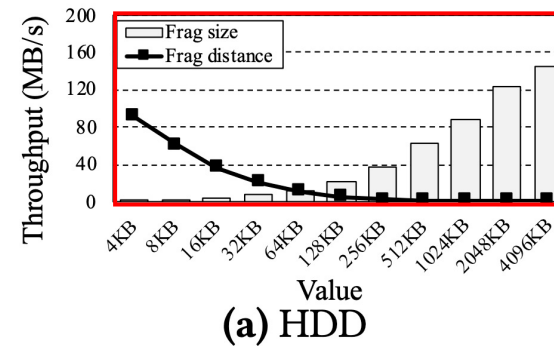
- 128KB O_DIRECT sequential read on F2FS
- Varying frag size/frag distance

* **Request splitting** occurs when frag size < 128KB

Observations

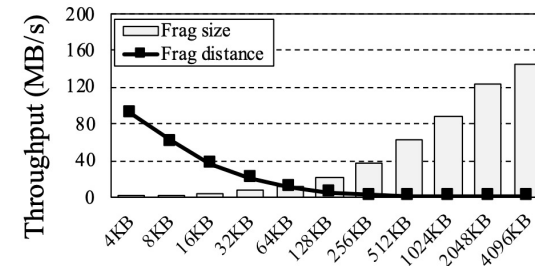
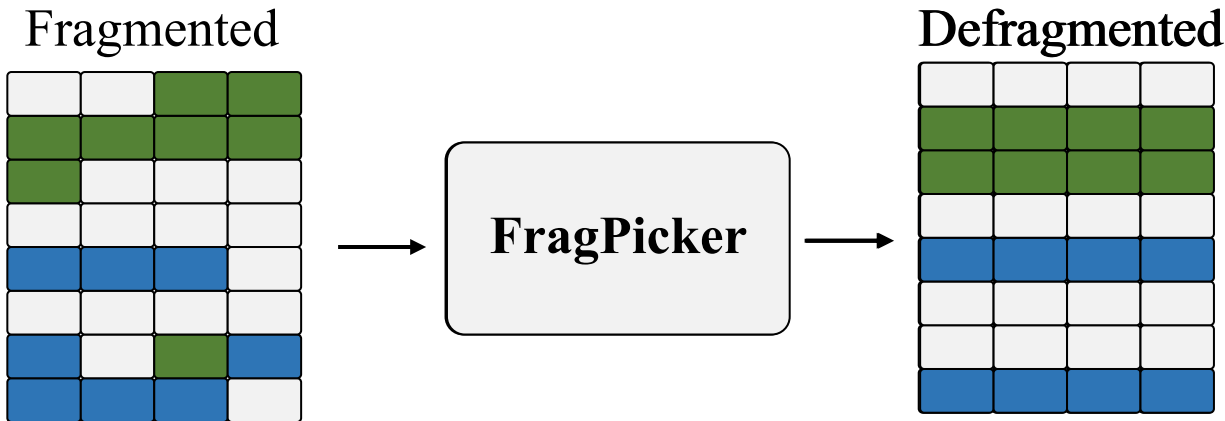
1. The performance of HDD is **sensitive to both frag size and distance** (due to seek time)
2. Flash/Optane storage is **sensitive to frag size** when it's less than 128KB.
3. Flash/Optane storage is **not sensitive to frag size ($\geq 128\text{KB}$) and frag distance.**

The performance of Flash/Optane storage, mostly depends on **the occurrence of request splitting** and is irrelevant to the distance between fragments.

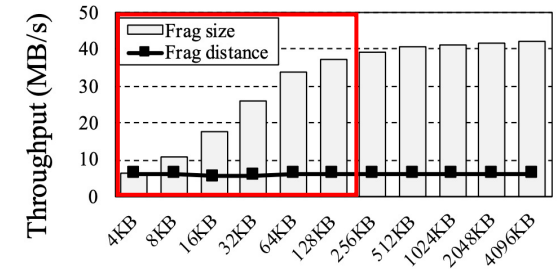


FragPicker for modern storage devices

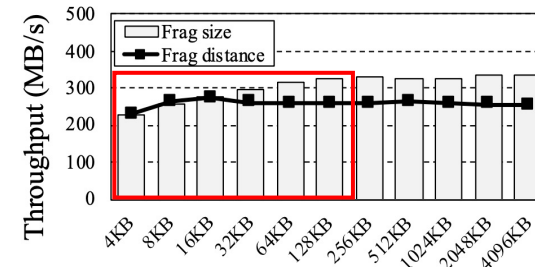
Let's ignore ~~frag distance~~ and do our best to prevent request splitting!



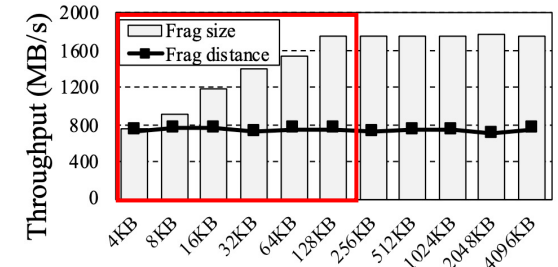
(a) HDD



(b) MicroSD



(c) Flash SSD



(d) Optane SSD

FragPicker for modern storage devices

Key Challenges

1. Which data to migrate?

2. How to migrate?

FragPicker for modern storage devices

Key Challenges

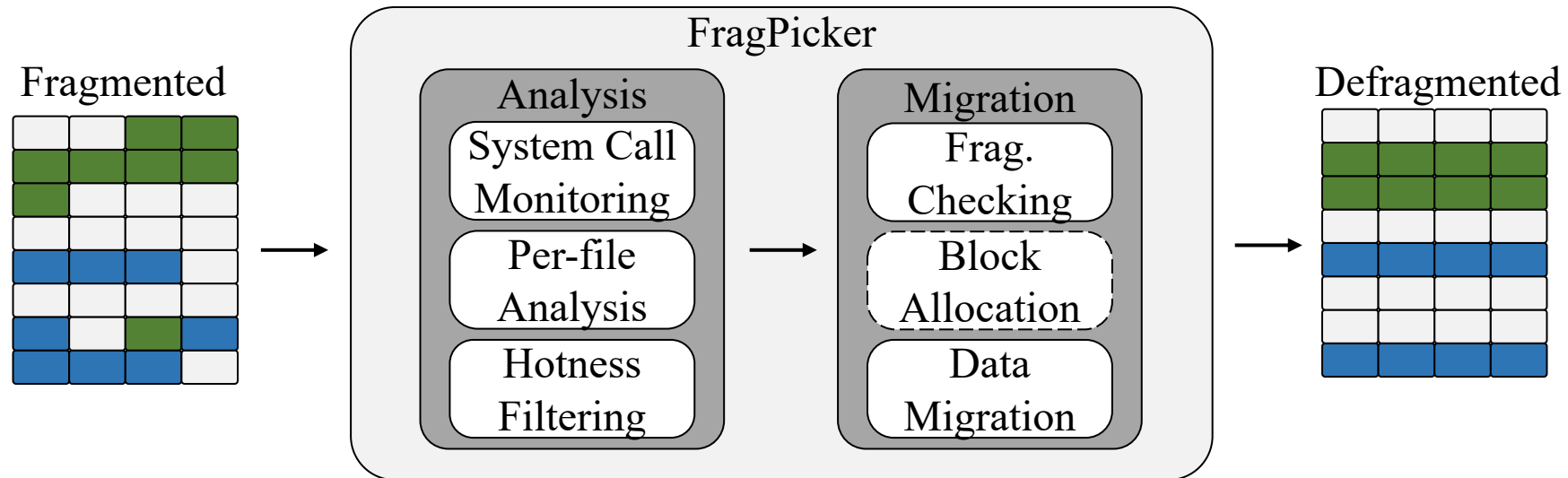
1. Which data to migrate?

➔ I/O Analysis to find the best pieces of data blocks to prevent request splitting

2. How to migrate?

➔ Filesystem-agnostic migration while minimizing the amount of writes

FragPicker for modern storage devices



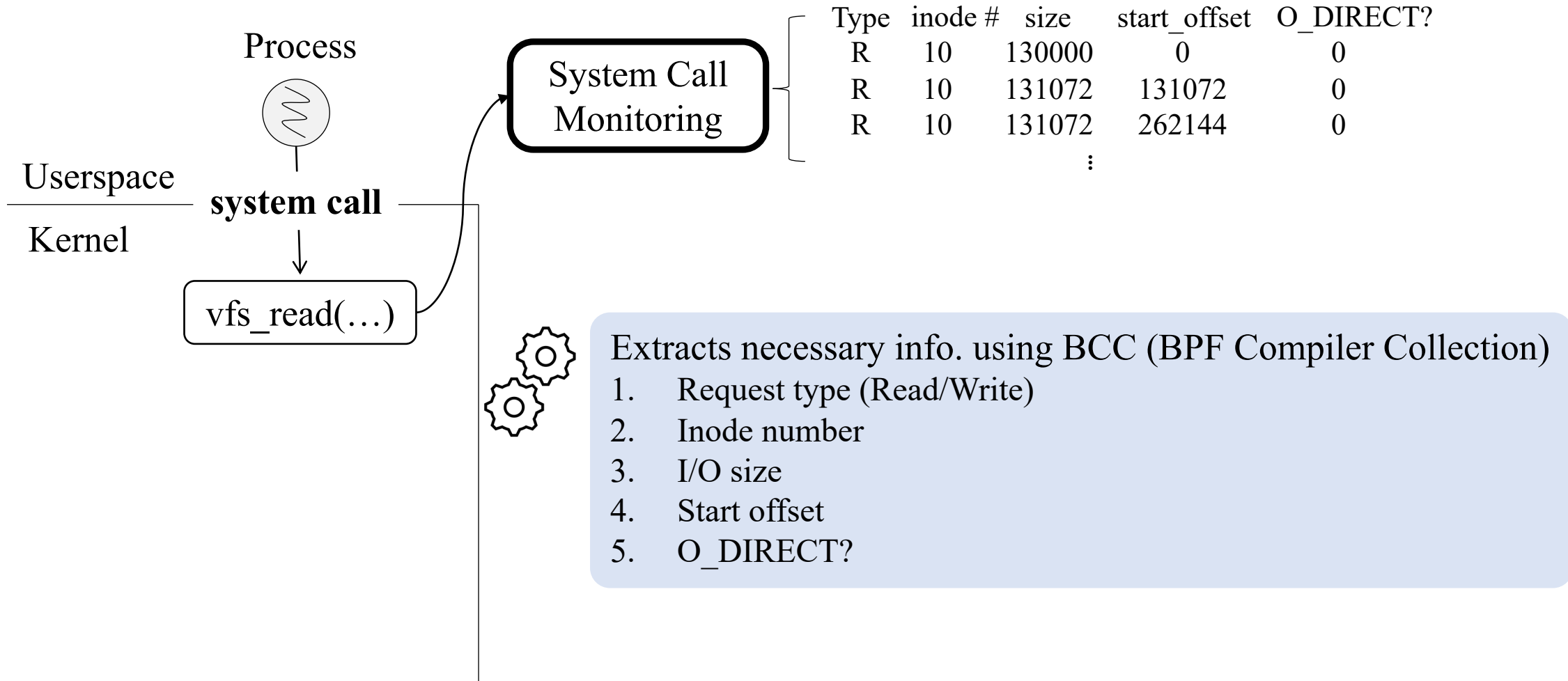
❖ Analysis Phase

- Monitors system calls related I/Os
- Analyzes I/O characteristics per file
- Filters I/Os with hotness

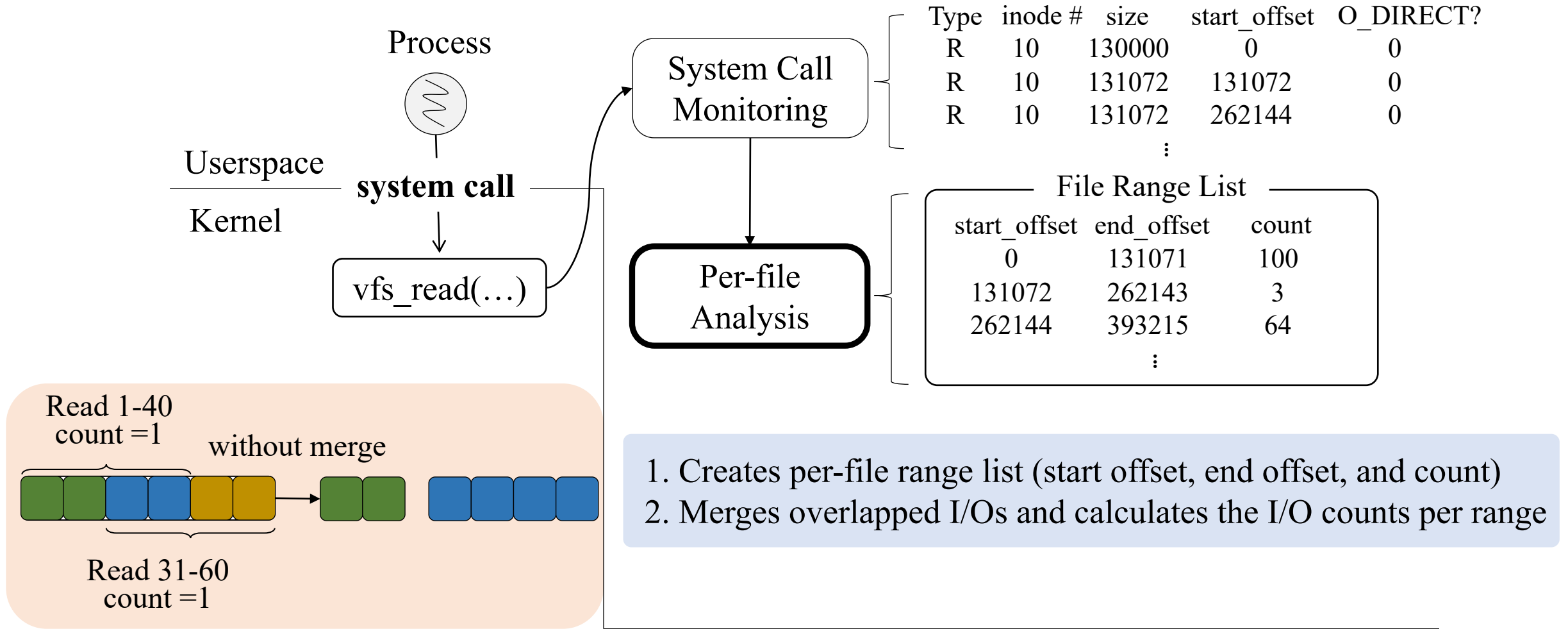
❖ Migration Phase

- Checks the fragmentation state
- Allocates contiguous blocks
- Migrates fragmented data into a new space

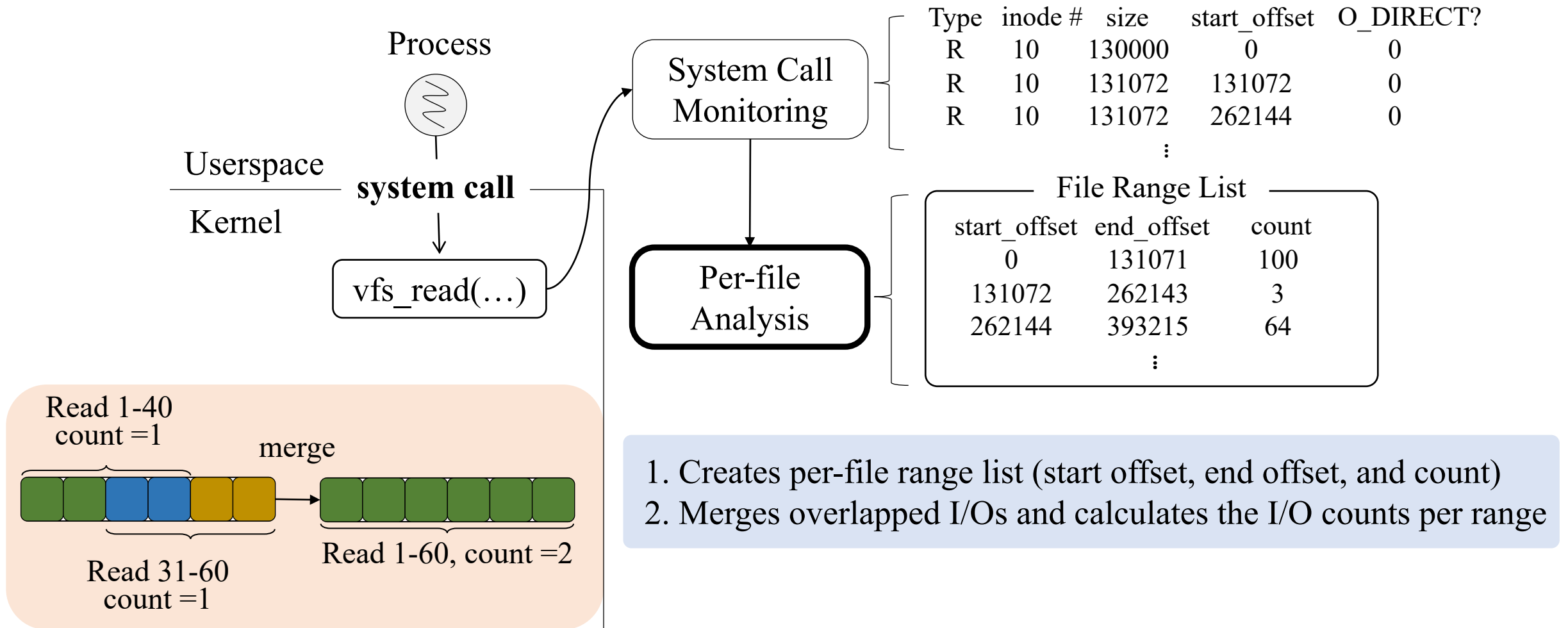
The analysis phase of FragPicker



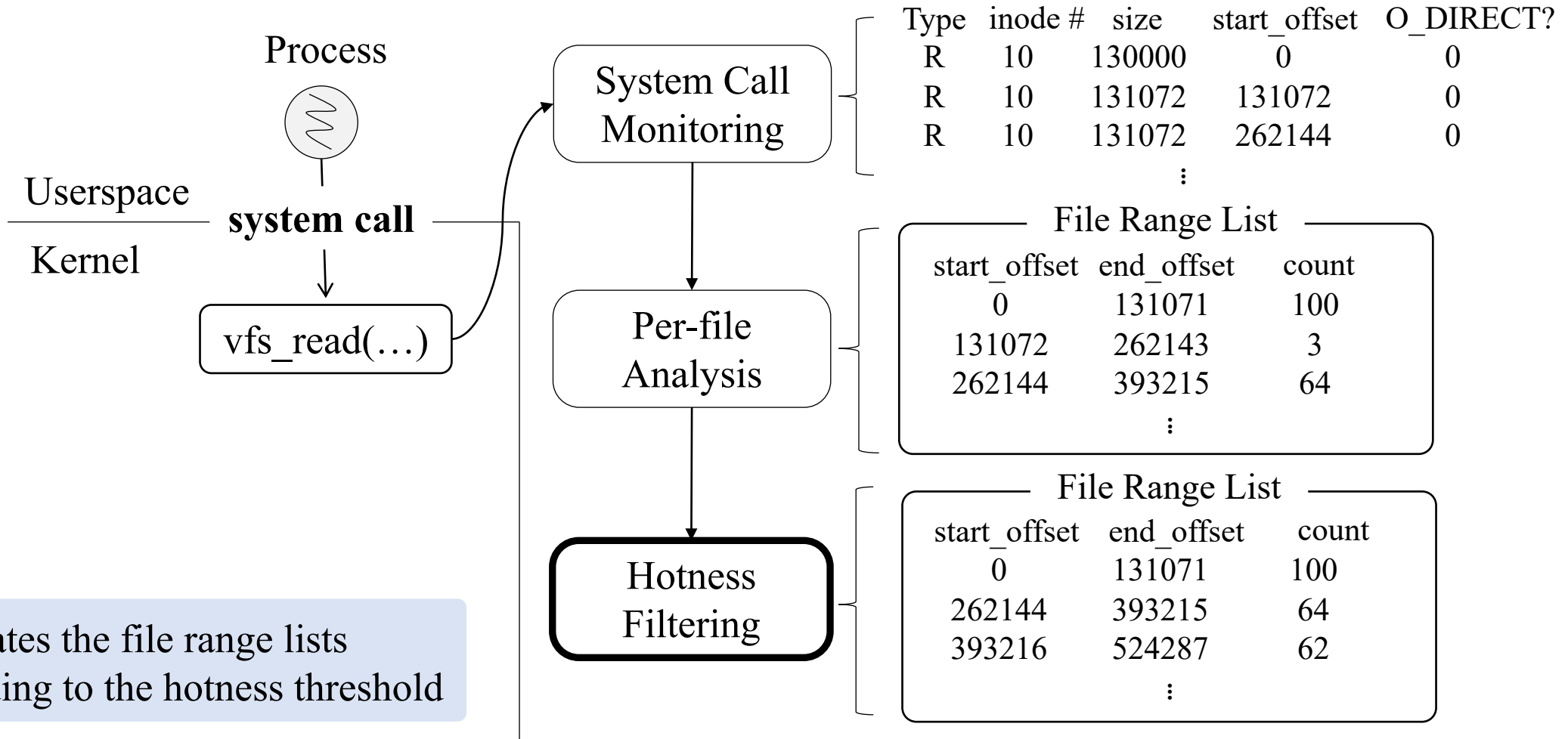
The analysis phase of FragPicker



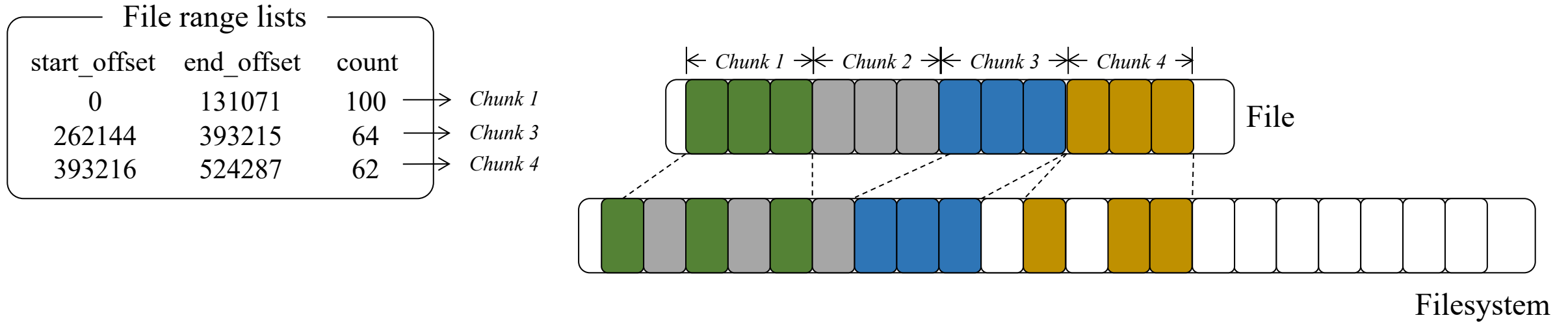
The analysis phase of FragPicker



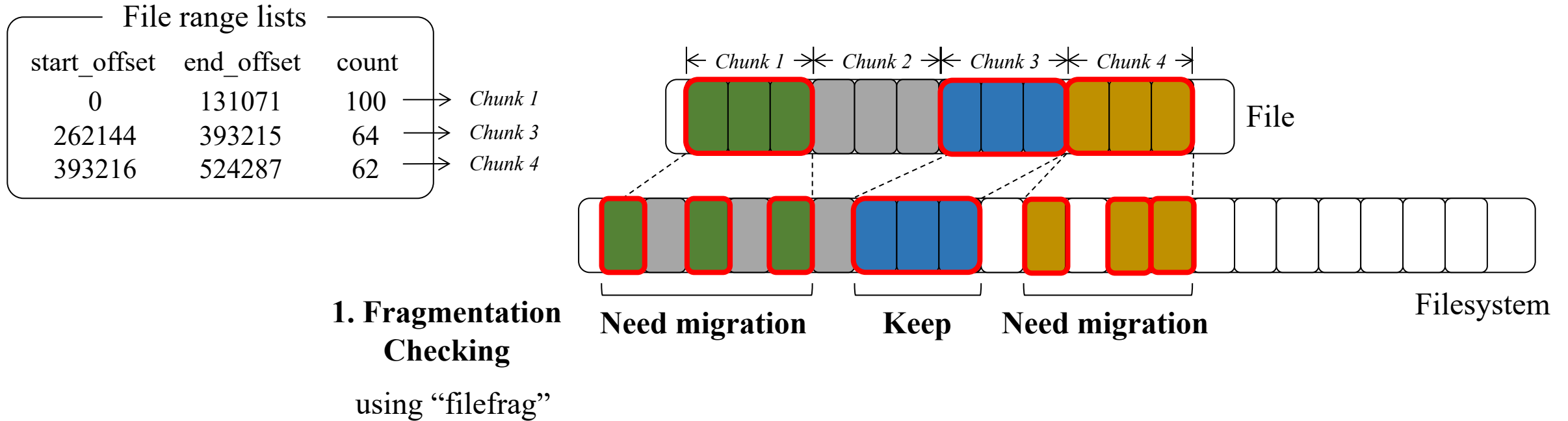
The analysis phase of FragPicker



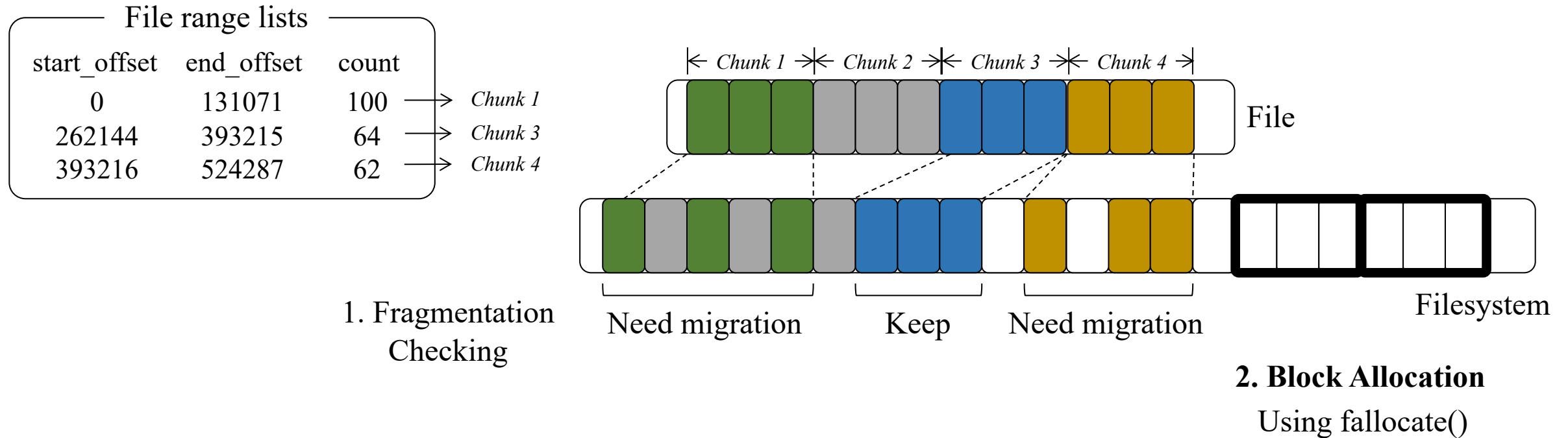
The migration phase of FragPicker



The migration phase of FragPicker

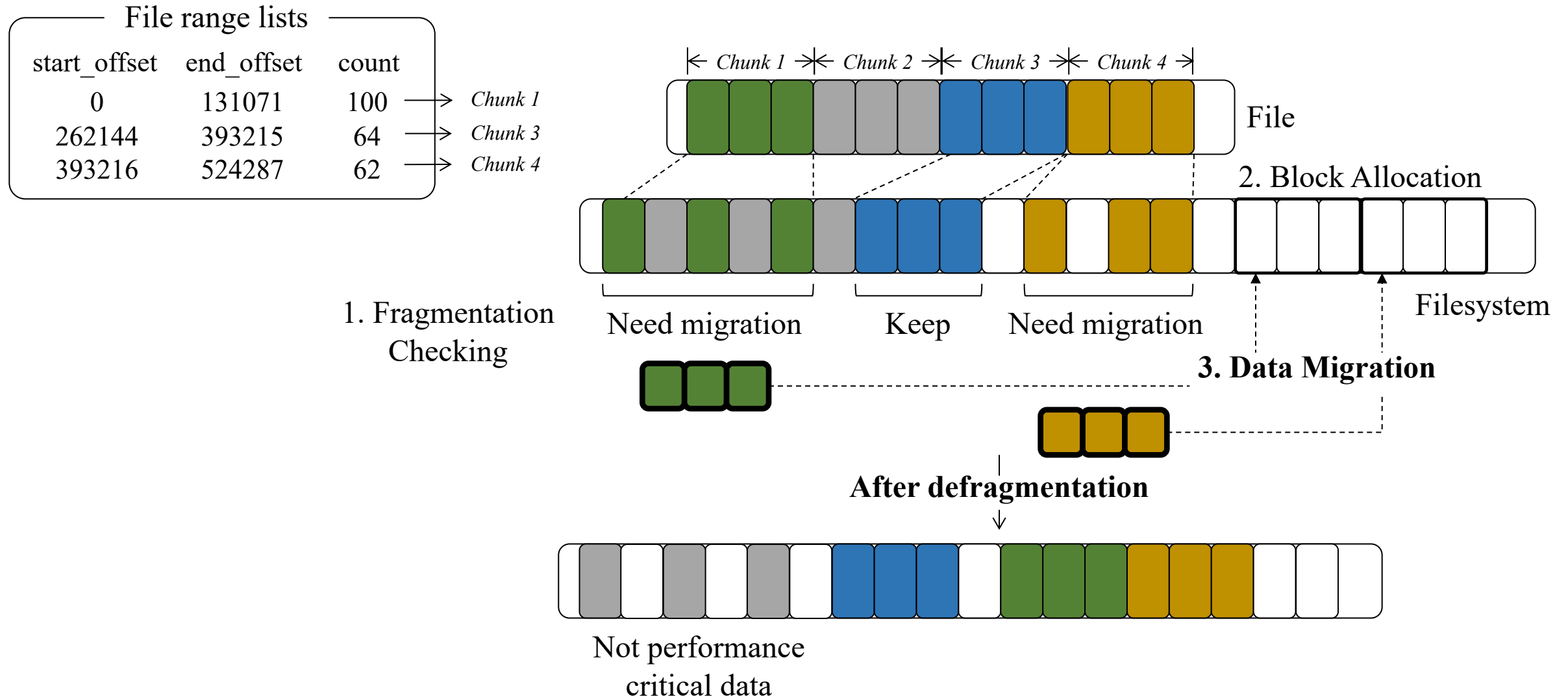


The migration phase of FragPicker



In the case of out-place update filesystems,
FragPicker can skip this process

The migration phase of FragPicker



Evaluation

Storage

1. Samsung SATA Flash SSD 850 PRO 256GB
2. Intel NVMe Optane SSD 905P 960GB



Workloads

1. Sequential read/update
2. Stride read/update

Objectives

1. Does FragPicker reduce the **amount of writes** for defragmentation?
2. Does FragPicker achieve a similar level of **performance gain**, compared with conventional tools?

Evaluation (Flash SSD)

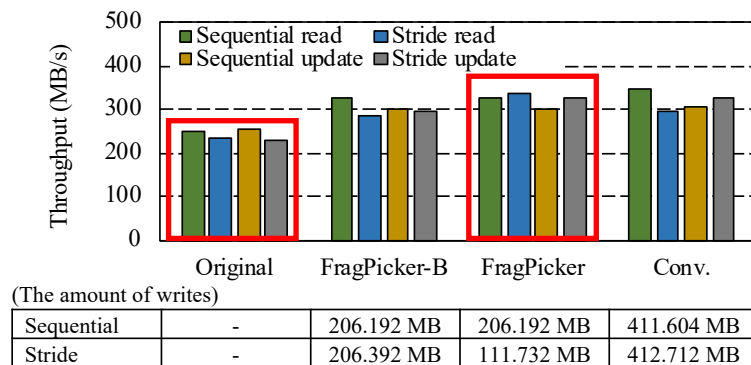
1) Write amount

FragPicker reduces the amount of writes by around 50% (sequential) and 75% (stride)

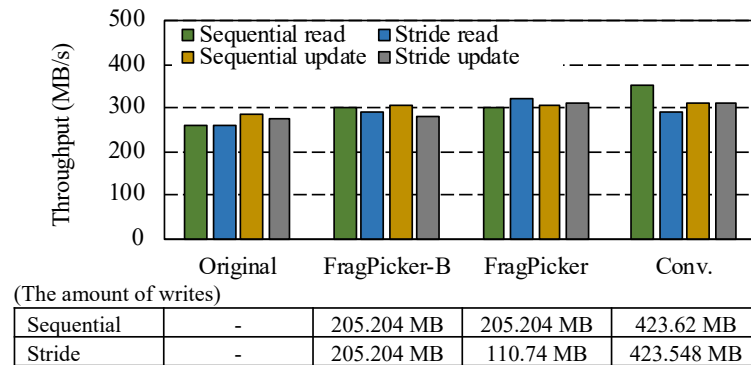
2) Performance

FragPicker improves sequential/stride I/O by around 30% and 42%, respectively

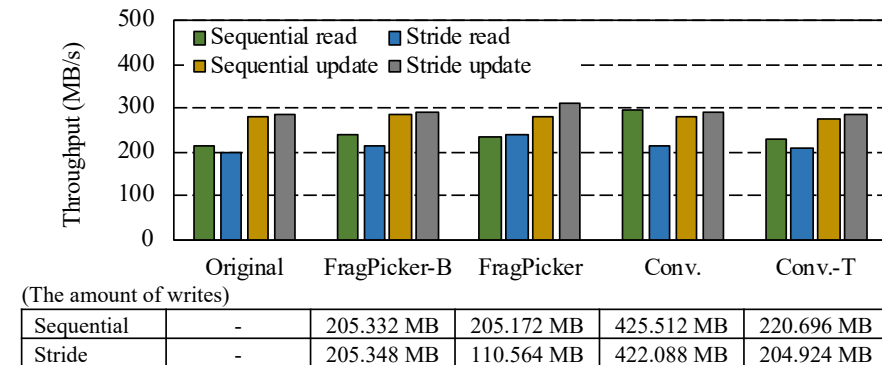
Regarding stride read performance, FragPicker outperforms conventional ones by 15%



Ext4



F2FS



Btrfs

Evaluation (Optane SSD)

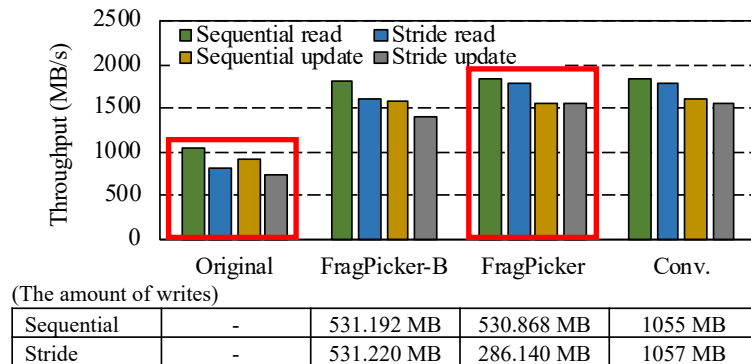
1) Write amount

FragPicker reduces the amount of writes by around 50% (sequential) and 75% (stride)

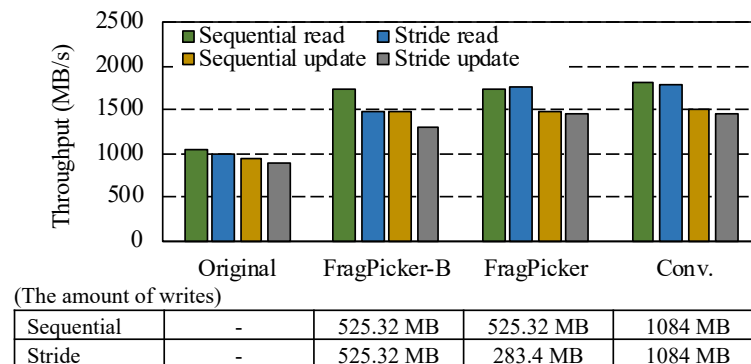
2) Performance

FragPicker improves sequential/stride I/O by around 75% and 120%, respectively

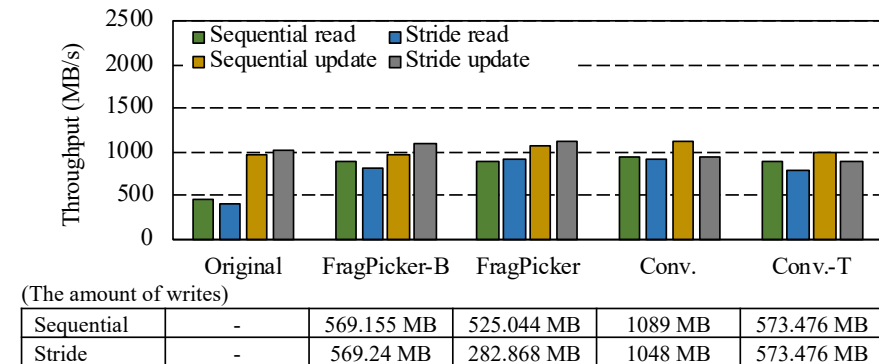
* High performance devices suffer from kernel overheads



Ext4



F2FS



Btrfs

More experiments in our paper!

- Database workloads
 - RocksDB YCSB-C
 - SQLite
- Fileserver workloads
- Hotness filtering test
- ...

Conclusion

- FragPicker
 - A new defragmentation tool for modern storage devices
 - No kernel modification
 - Filesystem agnostic
- Requirements for FragPicker
 - eBPF support
 - Filesystem-level consistency
 - fallocate and filefrag
 - Filesystem support for providing a contiguous LBA region for a single write op.

Thank you

Source code: <https://github.com/jonggyup/FragPicker>

Contact: jonggyu@skku.edu

