Number of accesses per snapshot

Figures 1 describe, the average number of accesses to a snapshot during the workload. We present the results for disk images of 50 and 500 snapshots. We can observe that with vanilla, the number of accesses per snapshoht is much larger compared to direct-access Stella ►*il faut voir avec Alain la nomenclature à suivre tout au long du papier. Il va sûrement la définir dès l'intro et c'est la même que tu devras employer pour qu'on te suive dans la lecture du papier. que ce soit pour les noms des versions (vanilla, direct access) ou des disques (chain, image, disk image)*◄. This is explained by the fact that, to access a specific cluster, vanilla needs to access each cluster in all previous intermediate snapshots to find which one contains this cluster. In contrast, direct-access directly accesses the index of the snapshot that contains the target cluster.

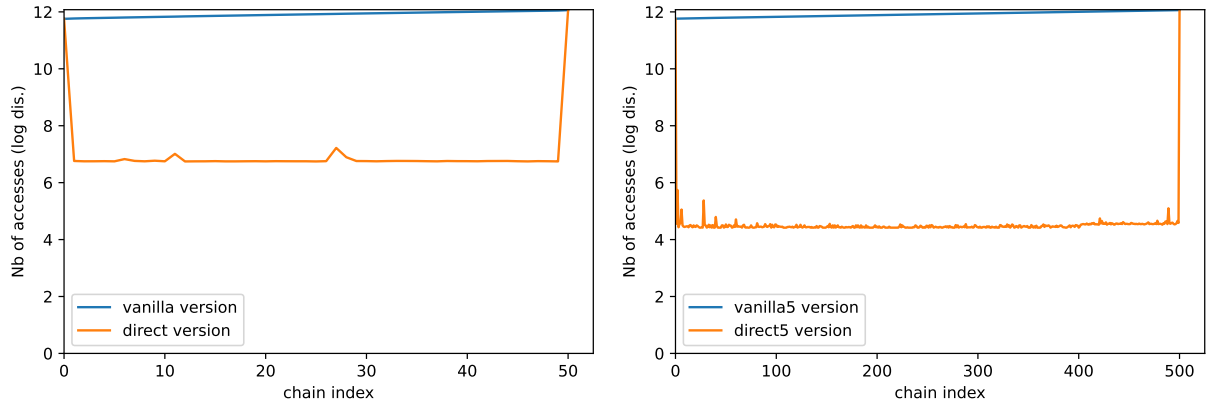

Figure 1. Accesses per sanspshots for a chain of 50 snapshots (on the left) and 500 snapshots (on the right).

Events when accessing a snapshot

Accessing a snapshot can results in one of the following events :

- normal => the cluster is allocated in the current snapshot (after we resolved an unallocated event Stella ►*ça veut dire quoi que le normal survient toujours après le "unallocated" ? et quand on trouve donc directement le cluster c'est quoi l'évt dans ce cas ?)*◄
- unallocated => the cluster is not allocated Stella ►*allocated ou present ? c'est quoi la diff ?*◄ in the current snapshot
- cache missed (Stella ►*je pense qu'il faudrait changer ça partout et appeler ça miss sans le "ed" car ce à quoi on est habitué et qui sonne normal comme tlb miss*◄) => the cluster metadata are or not in the cache
- cache hit => the cluster metadata are present in the cache

We can observe that with vanilla, the number of unallocated to access a given cluster (i.e., to obtain a *normal* event) increases exponentially with the numer of snapshots. While with direct-access, we almost always have the same number of *unallocated* and *normal* events regardless the number of snapshots. This is due to the fact that with direct-access, an *unallocated* event is not propagated all over the chain but rather resloved in the snapshot where it occurs (Stella ►*qu'est ce que tu entends pas "propagated" on ne comprend pas. tu n'nas pas dit pour vanilla qu'on propageait l'évt donc un peu perdu*◄).

Memory footprint

Memory footprint and throughput comparison between vanilla and direct-access during dd workload.

Startup duration

This section evaluates the startup duration of a VM for both vanilla and direct-access.

Cache *hit* and *missed* treatment times

Stella ►*il faut expliquer tout ce qui précédait là (la diff entre hit et miss) dans la section où tu parles des events en reformulant mieux*◄ We perfom this experiment on an HDD hard disk (as used by most of cloud providers), and an SSD hard disk, supposing that we are in the best cases of hard disk speed.

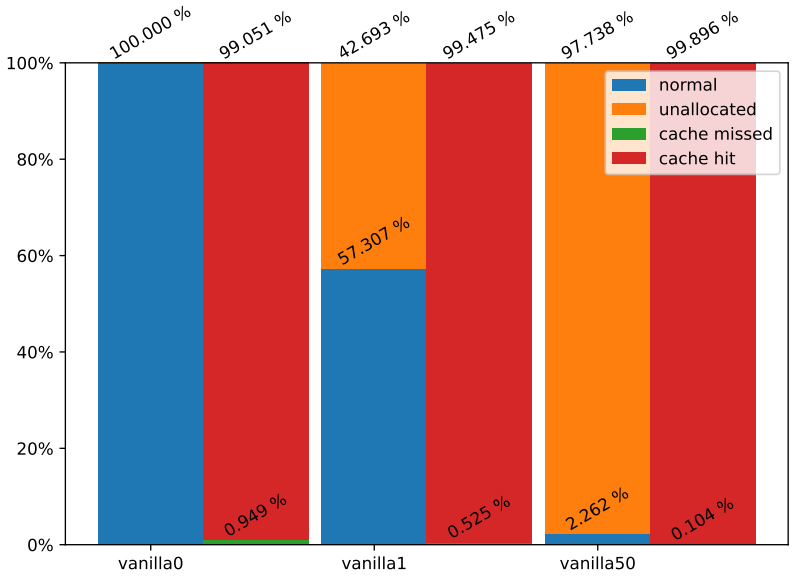

Figure 2. Number of events (unallocated, missed, hit, normal) with vanilla for chains of 0, 1, and 50 snapshots. Stella ►*je me dis que si tu voulais expliquer pq le nb d'accès avec vanilla (fig. précédente) alors ça aurait été bien de prendre les mm tailles de chaines*◄

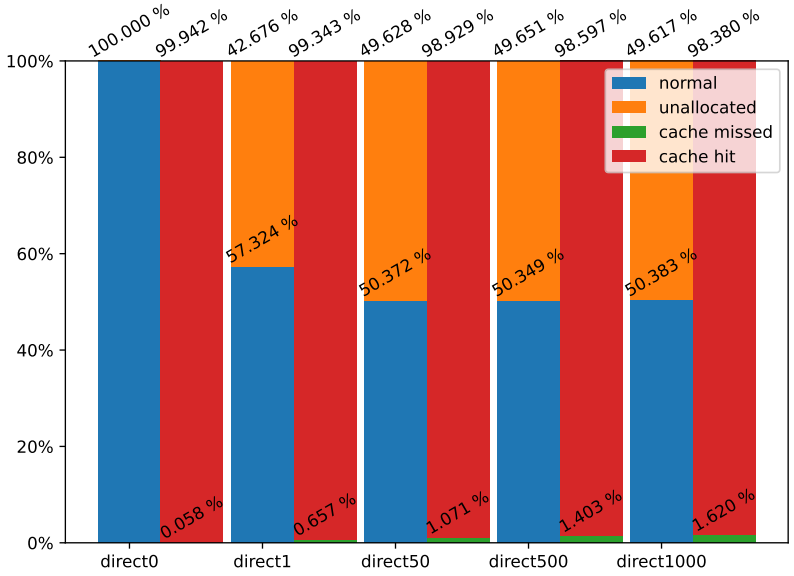

Figure 3. Number of events (unallocated, missed, hit, normal) with direct-access.

Cache Missed treatment times

Van/Dir x : Vanilla/Direct-access run on a chain of x snapshots. ***base*** *for a chain without snapshots.* Stella ►**ici tu parles de base alors que dans une figure précédente tu as utilisé 0 pr parler d'image sans snap il faut uniformiser cela**◄

We can make the following observations. (1) Obviously, when there is no snapshot, there are no differences between the 2 versions and times of treatment are the same. (2) When the chain length increases, treatment of a cache missed takes generally about 500*ms* with vanilla against less than 300*ms* with direct-access, for a chain of 100 snapshots. Stella ►***essaie de mieux***

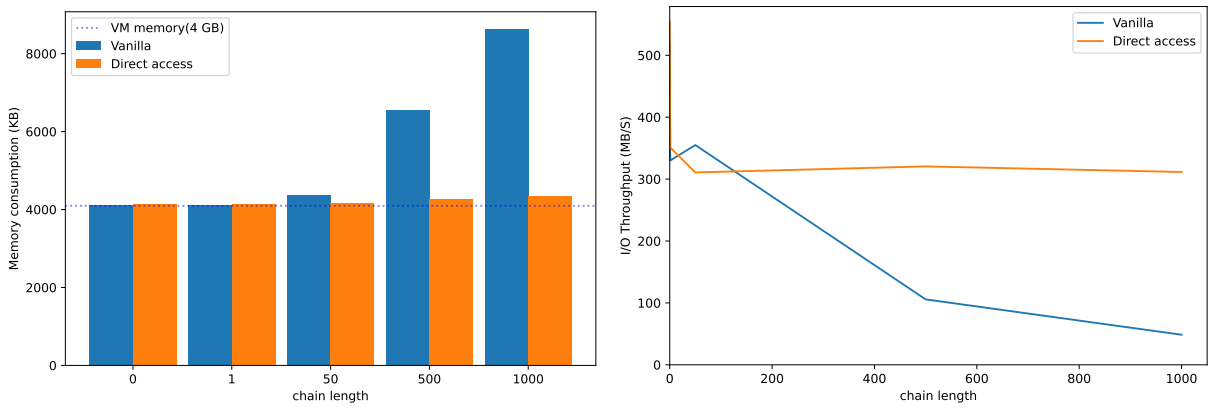

Figure 4. Memory consumption and Throughput (I/O) during the workload

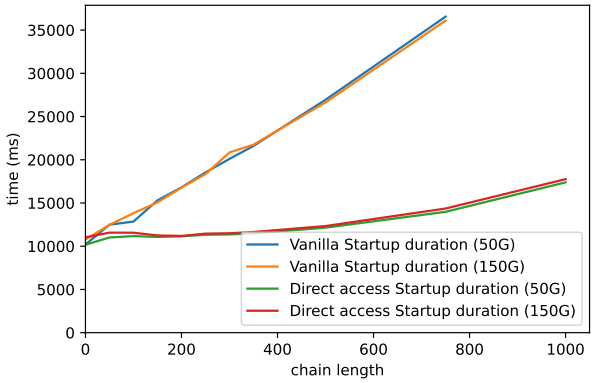

Figure 5. Startup, vanilla and direct-access version on 50G and 150G disk

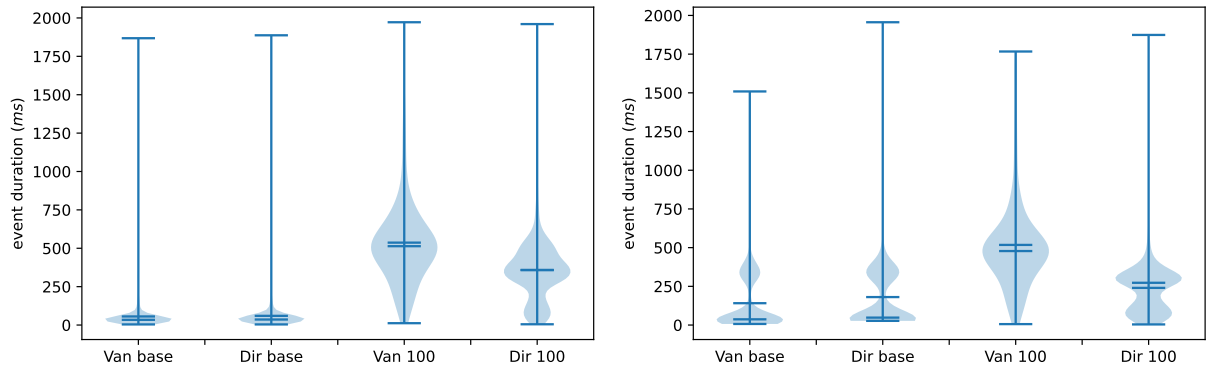

Figure 6. On the left, boxplot presenting time to resolve cache missed with images file stored on a HDD hard disk. On the right, the same with images file on an SSD hard disk

expliquer le paragraphe suivant je ne comprends pas : est ce un plus pour le travail de le soulever ? (vu que tu as dit que c'était pas notre but) : si oui mieux formuler, sinon l'omettre◂

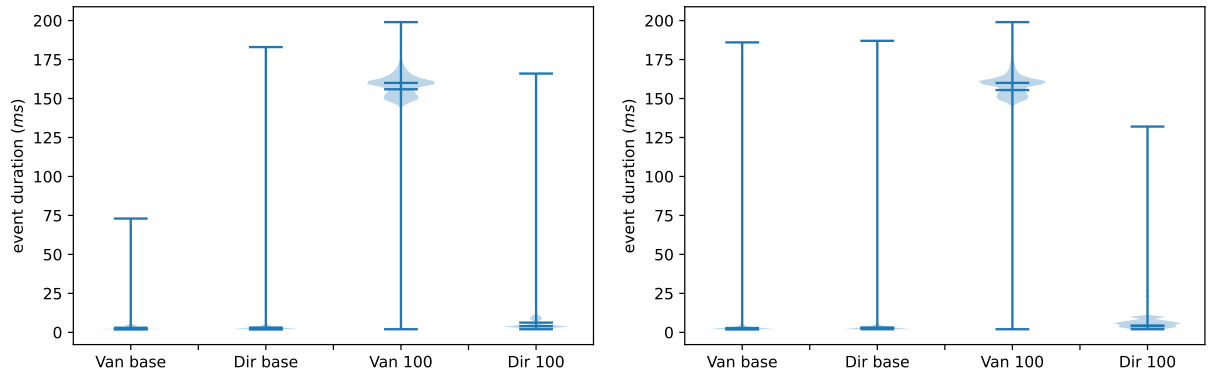

Figure 7. On the left, boxplot presenting time to get value in the cache with images file stored on a HDD hard disk. On the right, the same with images file on an SSD hard disk

Cache hit times

A hit time is not affected by the type of the disk (HDD or SSD) as it is done in-memory. Hits are often short events in general (at the order of μs). But when scaling by using a chain of 100 snapshots for example, vanilla takes much longer time because each hit knocks on all the intermediate cache levels. With direct-access, a hit only knocks on the target cache level. This is why regardless the number of snapshots, the time of a hit event with direct-access is almost the same.

Performance evaluation

We vary the cache size of the VM while starting it and launch a random workload (e.g., fio). The goal is to compare both solution when using the same amount of cache.

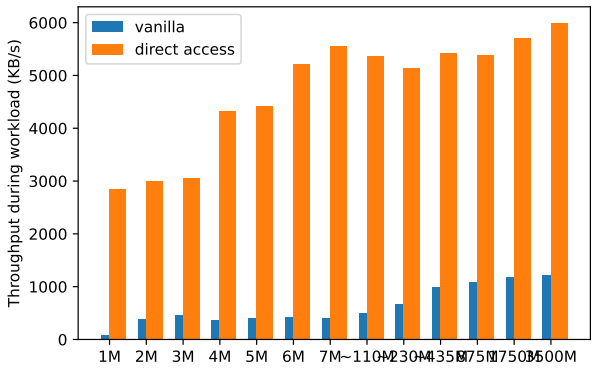

Figure 8. Cache variation while running a random read workload on a chain of 500 snapshots

We observe that direct-access always has a better throughput compared to vanilla. And when the cache size becomes greater than 4 MB, direct-access performance becomes almost constant. This is due to the fact that 4 MB is the minimal cache size required by direct-access for disk size and the chain length Stella ►*il faut dire à l'intro de la section où tu décris l'expériment*◄.

Cache deduplication in vanilla

This experiment demontrates the cache deduplication with vanilla. We count for each snapshot, the cluster metadata that are present in the cache and compute the amount of waste memory. Stella ►*explique ici ce que tu entends par "perte", j'imagine que c'est le fait que ce soit répliqué alors que ce n'est pas nécessaire*◄ The experiment is done with a chain of 100 snapshots and minimal cache size required is 7 MB. Fig. ?? presents the memory wasted (duplication) for fio (random read) workload from the entire VM life.

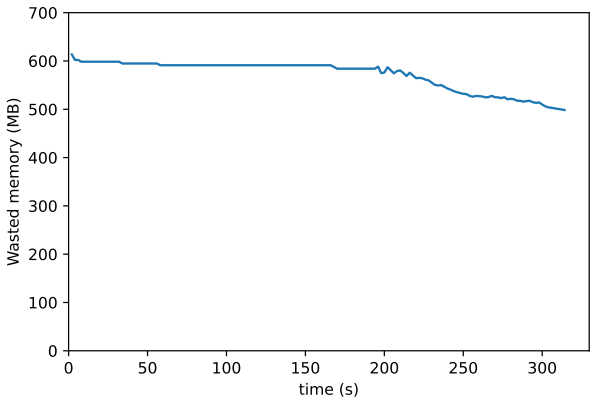

Figure 9. Startup, vanilla and direct-access version on 50G and 150G disk