guest_memfd 1G page support

For 2025-02-06 guest_memfd bi-weekly upstream call

Contact ackerleytng@google.com if you have questions/suggestions!

Overview

- At creation time, guest_memfd is configured to use 1G pages
- Pages will be allocated from hugetlb
- Pages will be split when used as shared
- Pages will be merged before returning them to hugetlb

When do we split the folio?

- When getting a new folio for the filemap, split if any of the offsets are shared
- When converting from private to shared

- Invariant: folios are always split when any sub-folio is marked shared

When do we merge the folio?

- Only in the `folio_put()` callback
- The `folio_put()` callback is the only place where we know no new refcounts will be taken on all the folios to be merged. We know that because either
  - Folio is used by the guest and we don’t expect refcounts to be taken on private pages
  - Shared folio refcounts have been dropped by the host and has reached 0

- Info needed to merge: only need original size of folio to merge folio
  - First folio will be determined by aligning to nearest page size
  - This info will be stored in lookup table keyed by physical address

Purposes of the `folio_put()` callback

- Transitioning mappability from `NONE` to `GUEST`
- Merging the folio if it is ready for merging
- Keeping subfolio around (even if `refcount == 0`) until folio is ready for merging or return it to hugetlb

When do we register the `folio_put()` callback?

- During shared to private conversion
  - Set mappability to NONE
  - Register `folio_put()` callback
  - Drop filemap refcounts to trigger callback
    - If folio has transient refcounts, callback will be delayed
- During folio truncation
  - Truncation also drops filemap refcounts, so callback is registered to
    - Do necessary merging
    - Keep folios from being freed
    - Free merged folios back to hugetlb

A lot depends on the `folio_put()` callback

=> Look at state machine that captures refcount changes in detail

State machine legend

- Mappability
  - GUEST: Only the guest can fault this page
  - ALL: Both guest and host can fault this page
  - NONE: Neither the guest nor host can fault this page
  - N/A: Folio not associated with inode (truncated), so there is no mappability information
- Refcount
  - fm: filemap's refcounts
  - vma: refcounts taken when this page is mapped
    - If vma refcounts are present, page is mapped and mapcount > 0
  - tt: Transient refcounts, any other (speculative) refcounts that are taken
- In filemap
  - True: this folio is associated with an inode's filemap
  - False: this folio has been truncated and removed from the filemap
- Split vs Merged

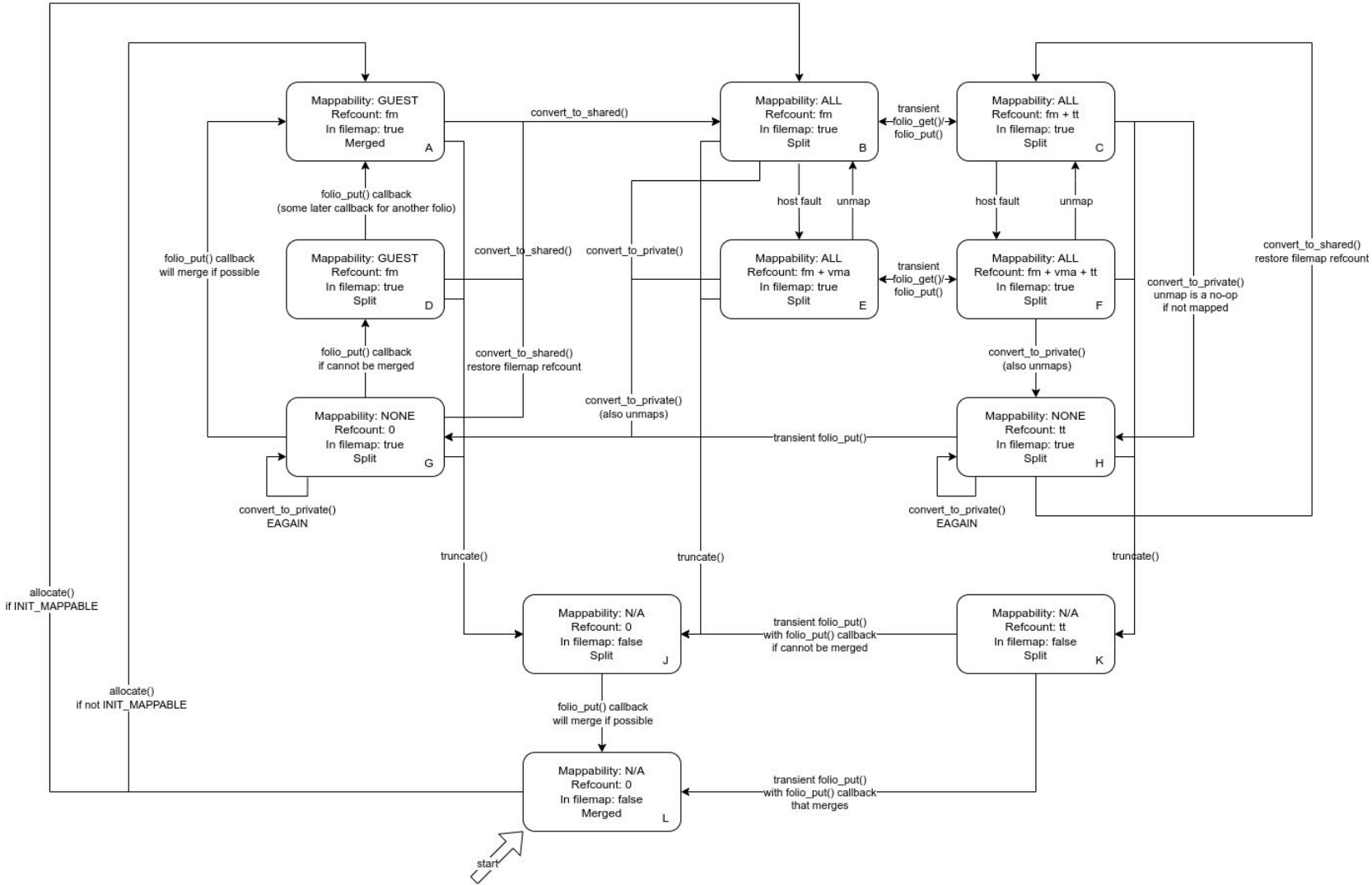
Mappability: GUEST
Refcount: fm
In filemap: true
Merged
A
convert_to_shared()
Mappability: ALL
Refcount: fm
In filemap: true
Split
B
transient
folio_get()/
folio_put()
Mappability: ALL
Refcount: fm + tt
In filemap: true
Split
C
folio_put() callback
(some later callback for another folio)
host fault
unmap
host fault
unmap
folio_put() callback
will merge if possible
Mappability: GUEST
Refcount: fm
In filemap: true
Split
D
convert_to_shared()
convert_to_private()
Mappability: ALL
Refcount: fm + vma
In filemap: true
Split
E
transient
folio_get()/
folio_put()
Mappability: ALL
Refcount: fm + vma + tt
In filemap: true
Split
F
convert_to_shared()
restore filemap refcount
convert_to_private()
unmap is a no-op
if not mapped
folio_put() callback
if cannot be merged
convert_to_shared()
restore filemap refcount
convert_to_private()
(also unmaps)
convert_to_private()
(also unmaps)
Mappability: NONE
Refcount: 0
In filemap: true
Split
G
transient folio_put()
Mappability: NONE
Refcount: tt
In filemap: true
Split
H
convert_to_private()
EAGAIN
convert_to_private()
EAGAIN
truncate()
truncate()
truncate()
allocate()
if INIT_MAPPABLE
Mappability: N/A
Refcount: 0
In filemap: false
Split
J
transient folio_put()
with folio_put() callback
if cannot be merged
Mappability: N/A
Refcount: tt
In filemap: false
Split
K
allocate()
if not INIT_MAPPABLE
folio_put() callback
will merge if possible
Mappability: N/A
Refcount: 0
In filemap: false
Merged
L
transient folio_put()
with folio_put() callback
that merges
start

Diagram drawn with draw.io

- SVG: https://lpc.events/event/18/contributions/1764/attachments/1409/3702/guest-memfd-state-diagram-split-merge-2025-02-06.drawio.svg
- XML for editing in draw.io: https://lpc.events/event/18/contributions/1764/attachments/1409/3703/guest-memfd-state-diagram-split-merge-2025-02-06.drawio.xml