Make IT AUTUMN 2021

Overview of Linux I/O capabilities

Speaker:

Urh Srečnik <*urh.srecnik@abakus.si*>

ORACLE Certified Professional
Oracle Database 12*c* Administrator

ORACLE Certified Professional
Java SE 8 Programmer

Abakus As na disku.

ORACLE Gold Partner

Abakus Plus d.o.o.

- Infrastructure Team
  - Services
    - OS & NET admin
    - DBA, Programming
  - Applications
    - Deja Vu
    - APPM
    - Arbiter

- Development Team
  - Enterprise Applications
  - Document Management
  - Newspaper Distribution
  - Flight Information System

References

NOVA BANKA

Aerodrom Ljubljana
Letališče Jožeta Pučnika Ljubljana

BANKA SLOVENIJE

BUTAN PLIN
hiša prijazne energije

METREL®
Measuring and Regulation Equipment Manufacturer

Mestna občina Ljubljana

MILENIJUM OSIGURANJE

NLB Vita
Življenjska zavarovalnica

MESTNA OBČINA KOPER
COMUNE CITTA DI CAPODISTRIA

gea
Tovarna olja Gea d.d.

Mercator

TelekomSlovenije

MERKUR

LASERLINE

ZAVOD ZA ŠPORT RS PLANICA

RAM 2

KONTROLA ZRAČNEGA PROMETA SLOVENIJE

jata emona
LJUBLJANA

ISKRATEL

iskra vzdrževanje strojegradnja

ISKRA ISD

SODO
SISTEMSKI OPERATER DISTRIBUCIJSKEGA OMREŽJA Z ELEKTRIČNO ENERGIJO

BERNARDIN GROUP
RESORTS & HOTELS

Hidria

Iskra®

PERLES
CHALLENGE OF PROFESSIONALS

HRANILNICA LON
Bančništvo na Ljubezniv Oseben Način

Gorenjska Banka

GENERALI GROUP
GENERALI Zavarovalnica

futuraplus

ELES
ELEKTRO-SLOVENIJA, d.o.o.
www.eles.si

DELO PRODAJA

datalab
Spremenite podatke v dobiček!

Sava TRADE

SKB
SOCIETE GENERALE GROUP

GOODYEAR DUNLOP
SAVA TIRES

Backup server

supports Oracle Databases and OLVM VMs

Abakus

- **Backup**
  takes no time
- **Recovery**
  data recovery is almost instant
- **Disk space**
  backed up data takes up minimal amount of disk space
- **Availibility**
  data is always available and always in view
- **Security**
  backed up data can not be deleted without support personnel intervention
- **Alternative uses**
  BI analysis / reporting / DB upgrade verification / R&D testing / seamless business continuation

www.abakus.si

Abakus

APPM

Abakus Plus
Performance
Monitor

- For Oracle Database Standard Edition
- Made by DBAs for DBAs
- Temporal performance comparison
- Resource allocation optimization
- Database performance tracking
- Performance bottleneck optimization

www.abakus.si

Linux ate my ram!

- https://www.linuxatemyram.com/

- Sometimes we do not want to populate the cache with one-time contents because we want other apps to keep their cached files.
- Great example would be a backup script

- Btw, from KVM hypervisor’s perspective,
  populated page cache inside VM is used (RSS) memory.

Live Demo

live demo recap

Limiting page cache usage

- Cgroups
- `sync; echo 3 > /proc/sys/vm/drop_caches`
- `dd if=random.iso iflag=nocache count=0`

*nocache uses **POSIX_FADVISE** to drop cache for whole file*

POSIX_FADV_DONTNEED

$ man **fadvise**, man **posix_fadvise**

```
#include <fcntl.h>
int posix_fadvise(int fd, off_t offset, off_t len, int advice);
```

Do not expect access in the near future. Subsequent access of pages in this range will succeed, but will result either in reloading of the memory contents from the underlying mapped file or zero-fill-in-demand pages for mappings without an underlying file.

live demo recap

POSIX_FADV_DONTNEED

	real	user	sys
always	9,12	0,65	8,19
after 16mb	8,87	0,60	8,05
after 32mb	8,88	0,61	8,00
after 64mb	8,63	0,62	7,80
after 256mb	8,37	0,58	7,63
never	7,37	0,62	6,36

live demo recap

FADV_SEQUENTIAL

*POSIX_FADV_NORMAL sets the **readahead** window to the default size for the backing device; POSIX_FADV_SEQUENTIAL doubles this size, POSIX_FADV_RANDOM disables file readahead entirely.*

live demo recap

stdio

- `open()`
- `read()`
- `close()`

stdio

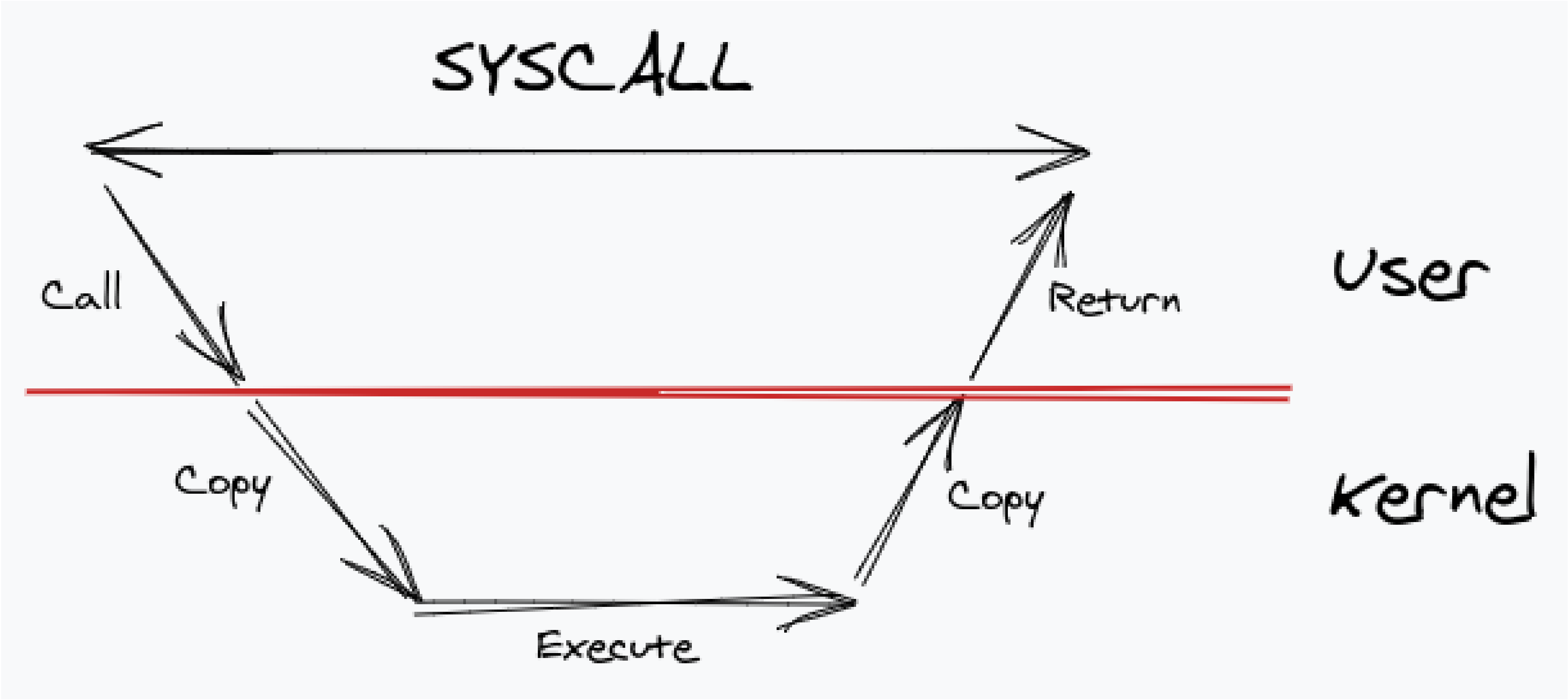
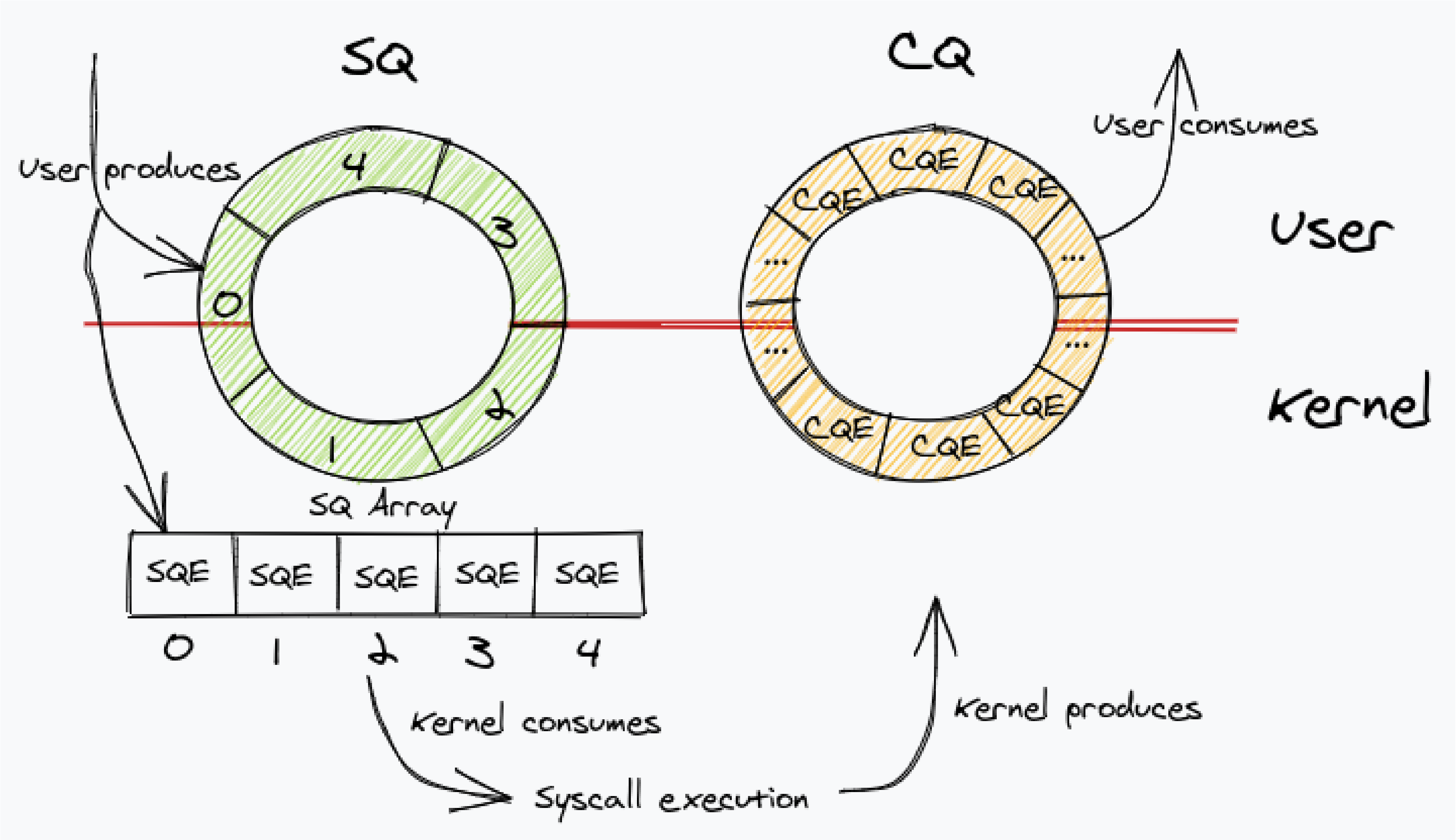


Image from https://mattermost.com/blog/iouring-and-go/

uring



Images from https://mattermost.com/blog/iouring-and-go/

live demo recap

aio

- `open()`
- `io_setup()`
- `io_prep_pread()`
- `io_submit()`
- `io_getevents()`
- `io_destroy()`
- `close()`

live demo recap

uring

- `open()`
- `io_uring_queue_init()`
- `io_uring_get_sqe()`
- `io_uring_prep_read()`
- `io_uring_sqe_set_data()`
- `io_uring_submit()`
- `io_uring_wait_cqe()`
- `io_uring_cqe_get_data()`
- `io_uring_cqe_seen()`
- `io_uring_queue_exit()`
- `close()`

uring vs aio

- https://kernel.dk/io_uring.pdf
- aio only supports async IO for O_DIRECT
- aio may block for metadata operations
- uring is newer »replacement« for aio
    - more features, »faster«

- aio requires kernel ~2.6 or newer
- uring requires kernel ~5.6 or newer

Dirty Pages

- **`vm.dirty_background_[ratio|bytes]`**
  how many dirty pages before sync starts
- **`vm.dirty_[ratio|bytes]`**
  how many dirty pages before i/o is blocked until sync frees up the required space
- **`vm.dirty_expire_centisecs`**
  how long can a dirty page be in cache before sync starts
- **`vm.dirty_writeback_centisecs`**
  how often should kernel check if something needs to be done

fsync()

- **fsync()**, **sync()** causes all pending modifications to filesystem metadata and cached file data to be written to the underlying filesystems.
- **fsync(int fd)**, **syncfs(int fd)** is like sync(), but synchronizes just the filesystem containing file referred to by the open file descriptor fd.

- Usage example: Oracle redolog files

sync_file_range()

```
SYNC_FILE_RANGE_WAIT_BEFORE
  | SYNC_FILE_RANGE_WRITE
  | SYNC_FILE_RANGE_WAIT_AFTER
```

will ensure that all pages in the specified range which were dirty when sync_file_range() was called are committed to disk.

fallocate()
FALLOC_FL_PUNCH_HOLE

Specifying the FALLOC_FL_PUNCH_HOLE flag deallocates space (i.e., creates a hole) in the byte range starting at offset and continuing for len bytes.

Within the specified range, partial file system blocks are zeroed, and whole file system blocks are removed from the file. After a successful call, subsequent reads from this range will return zeroes.

AS

Abakus

As na disku.

http://www.abakus.si/