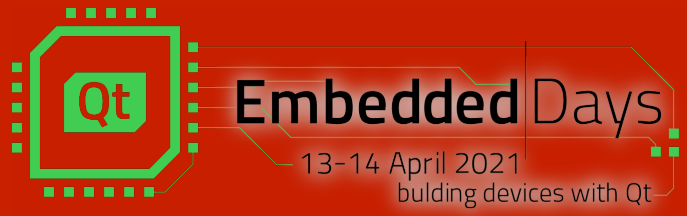
TUXERA



What does reliability mean to you?

When a device fails in the field, the consequences can be significant.

Sami Kassimäki
Product Manager
Tuxera

Agenda

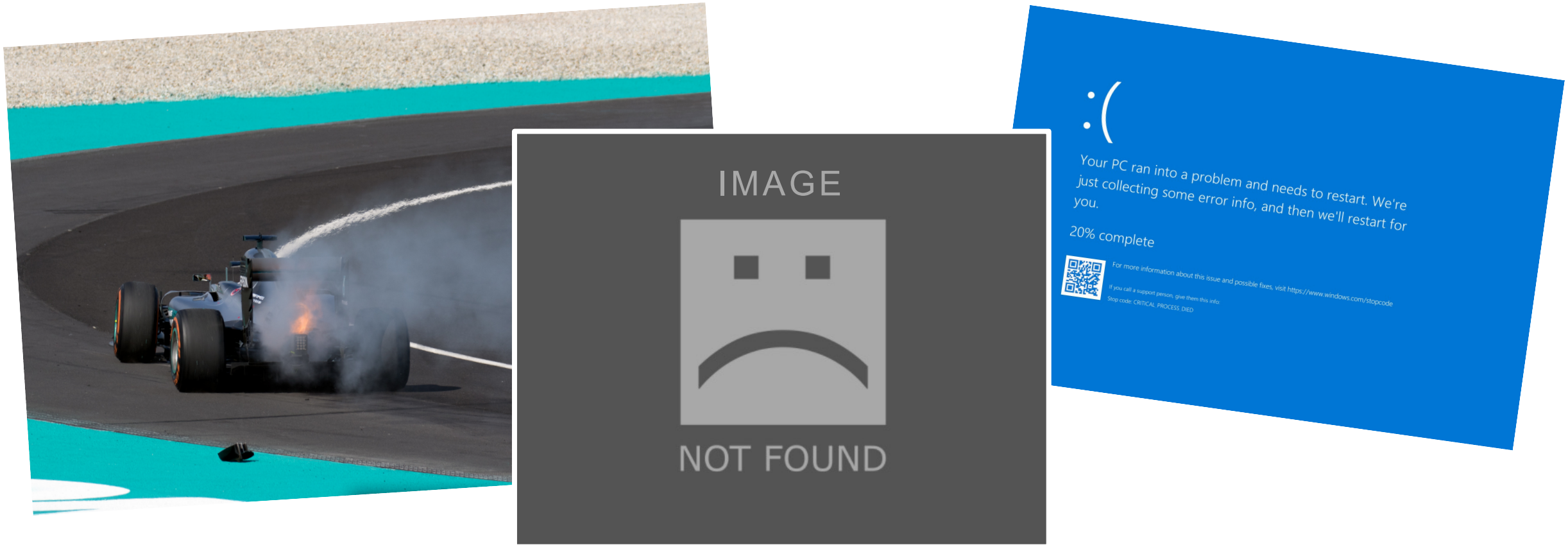
01 **Consequences of reliability failure**

02 **Understanding levels of reliability**

03 **Increasing reliability**
- Flash media
- Programming techniques
- File systems

When reliability fails

When reliability fails



Why reliability matters

- Field-failures are costly and resource draining
- User experience suffers
- Mission-critical data can be lost

TUXERA

What does reliability mean to you?

Hardware reliability

System integrity

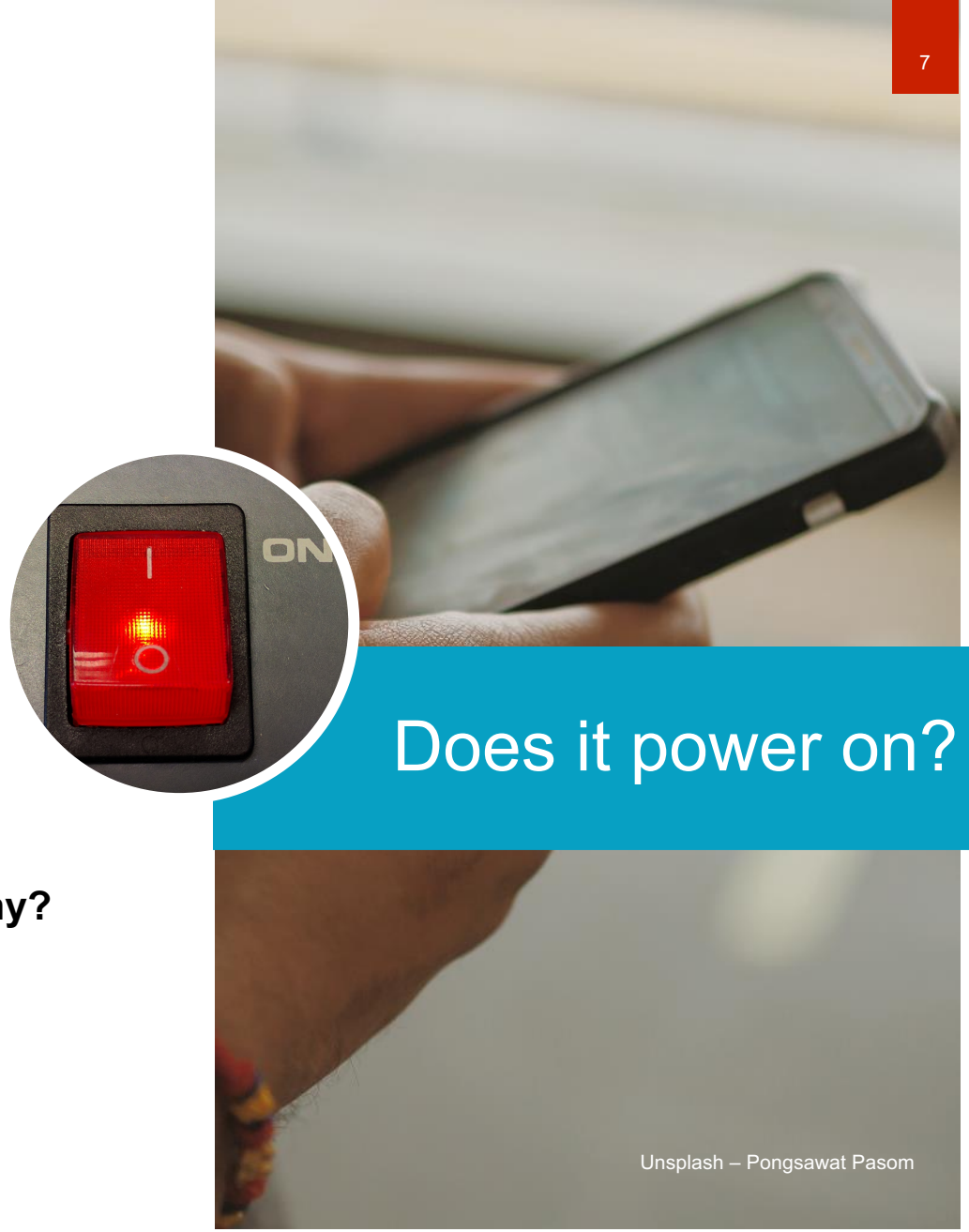
Data integrity

Hardware reliability

Does it power on?

- Basis for reliability
- Storage choices matter
- Misbehavior can be fatal
  - Storage device corruption
  - Memory corruption
  - Bus corruption

If a device doesn't power on, what's the cost to you and your company?

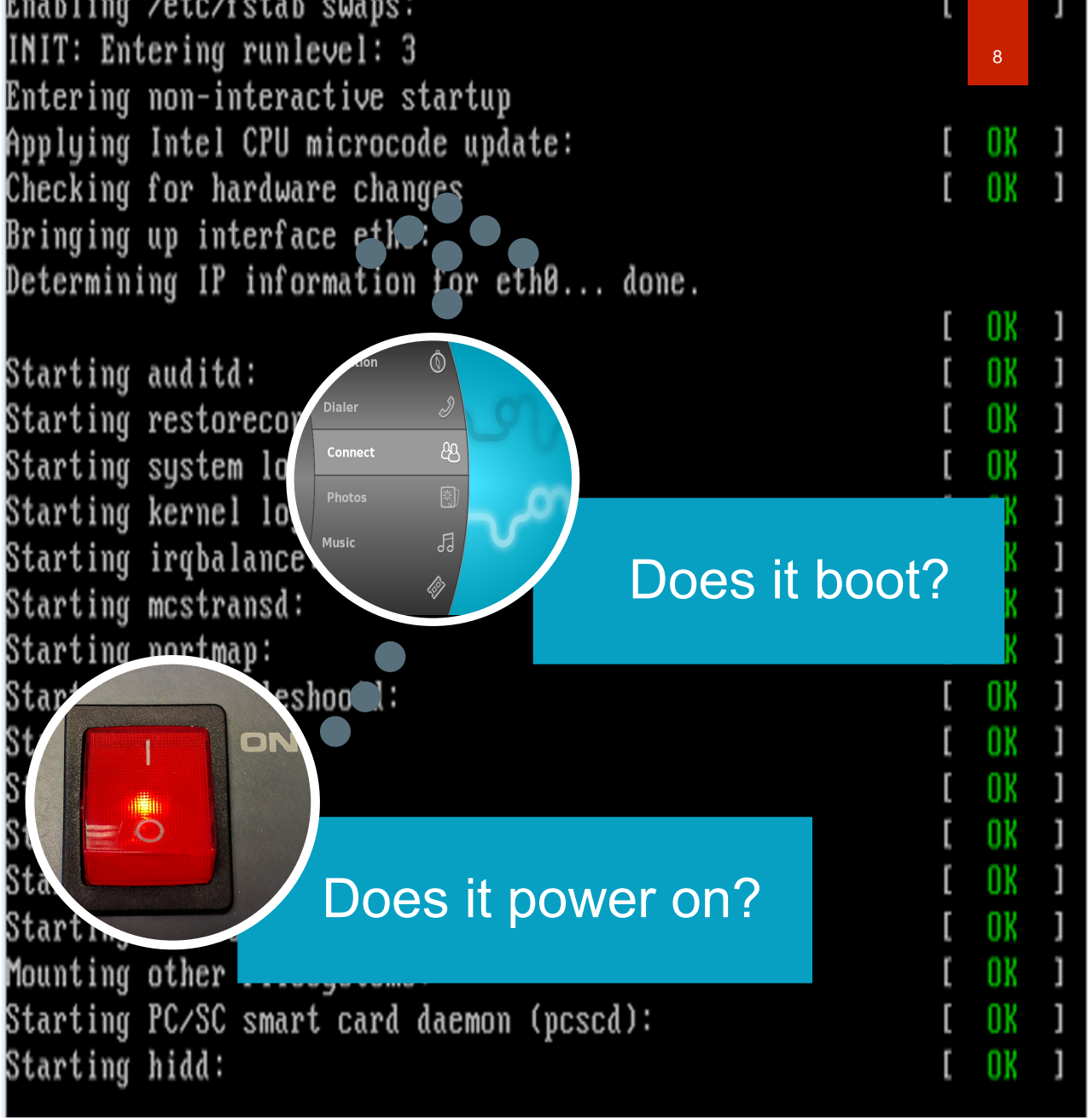


System Integrity

Does it boot?

- Metadata must always be in consistent state
- No media corruption due to:
  - Power failure
  - Unplugged storage

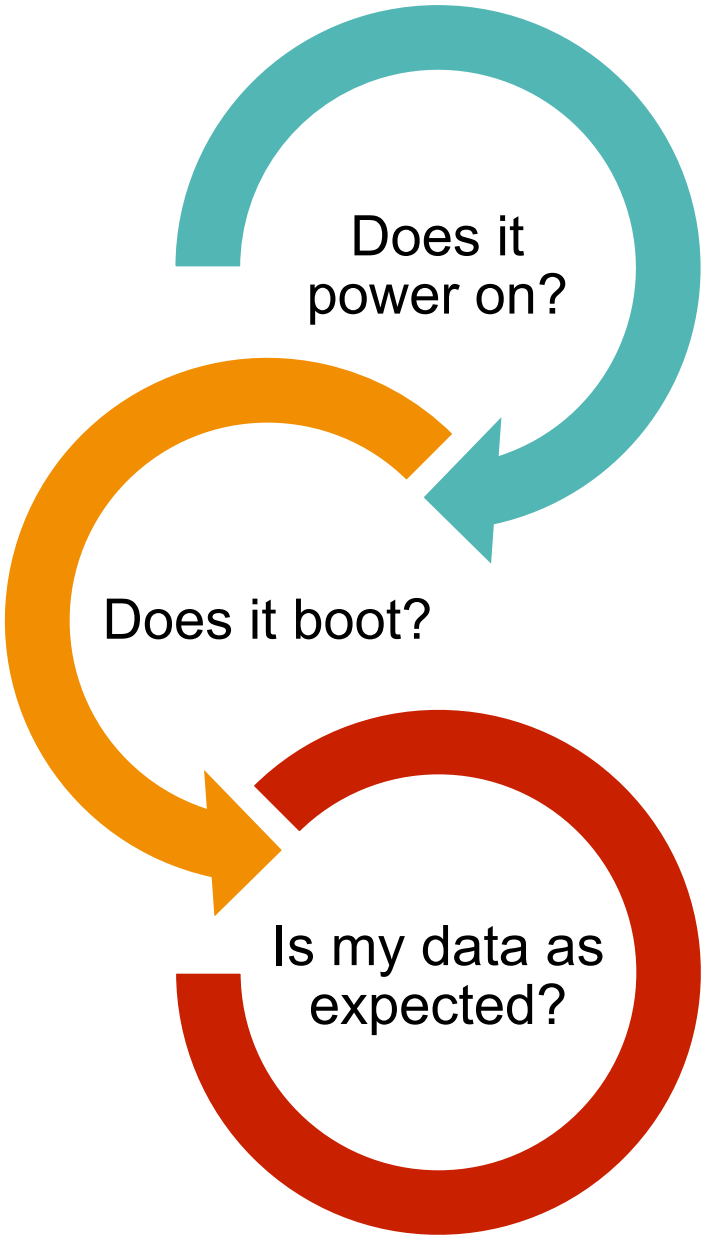
How do you prepare your devices for power interruptions?



Data Integrity

Does it have complete and correct user data?

- User data always in consistent state
  - No partial files
  - User data is not overwritten
- Understanding what is or isn't committed is crucial



TUXERA

Increasing reliability

Hardware

Considerations with Flash media

Programming techniques

Added mechanisms

File systems

Helping with complexity

Considerations with flash

Flash storage concerns

Flash memory wear-out

Flash memory lifetime limited by number or write cycles

Max number write cycles (P/E cycles) reached = system failure!

Concerns

- Extended lifetime requirement up to 10-20 years
- Applications are becoming increaingly write-intensive
- Expensive to replace flash memory
- Potential damage to brand reputation due to system failure

Flash storage concerns

Write amplification factor

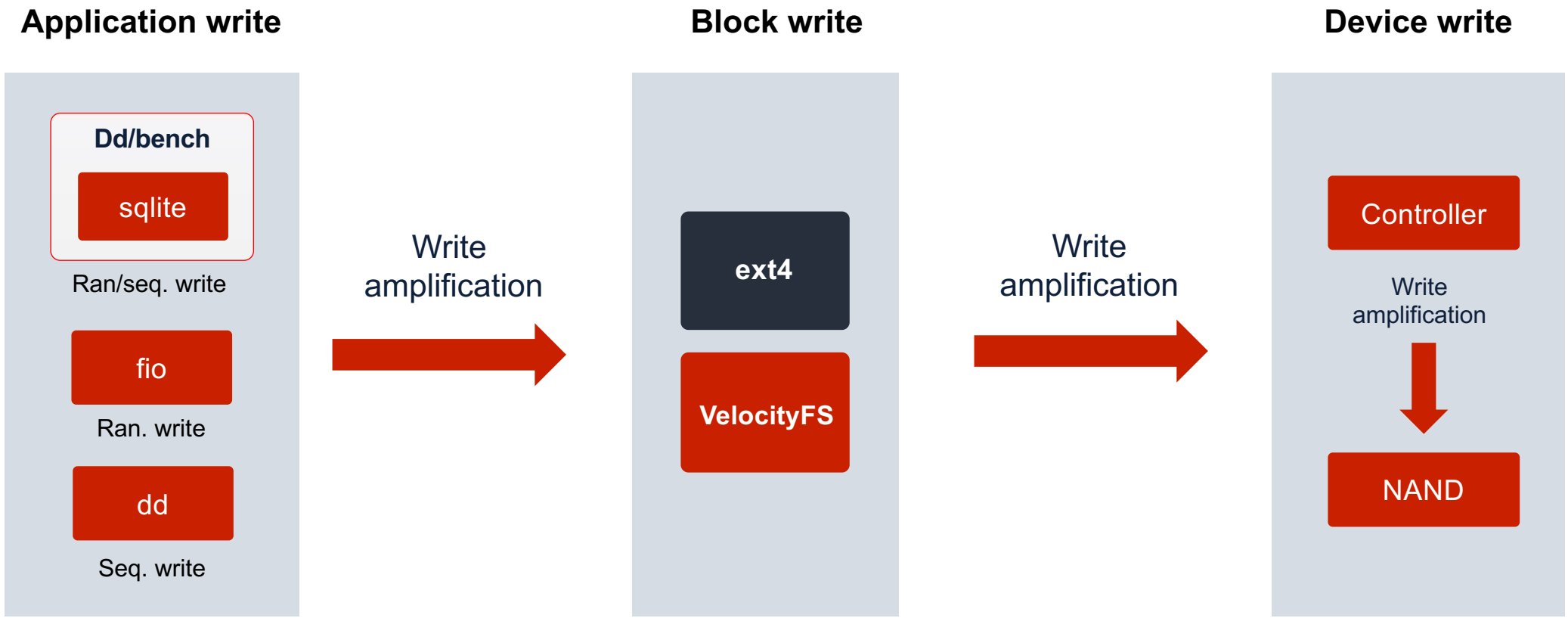
An undesirable phenomenon where the actual amount of physical information written is more than the logical amount intended.

Concerns

- Leads to flash memory wear-out
- Can cause critical system failure
- Sluggish performance (phones, automotive)

$$\text{Write Amplification Factor} = \frac{\text{Device write}}{\text{Application write}}$$

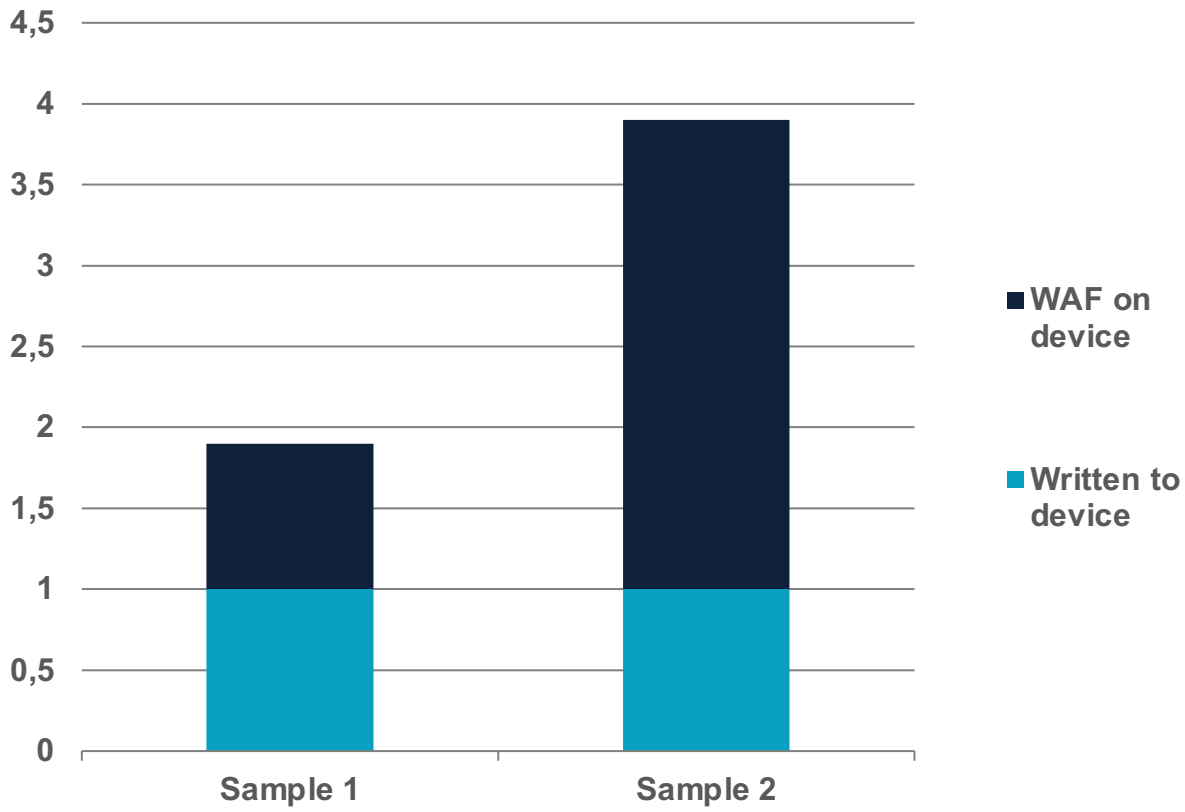
How write amplification multiplies



Validating flash lifetime

Workloads are getting increasingly write-intensive

- Managed NAND behaves differently
- Same workload – different results
- Validating Flash Lifetime to highest standards
- Exceed your customer expectations



TUXERA

Programming techniques

Programming techniques for increased reliability

TUXERA

Read-only partitions and files

Read-only partitions

- Helps with file system level corruption
- Doesn’t prevent media corruption

Read-only files

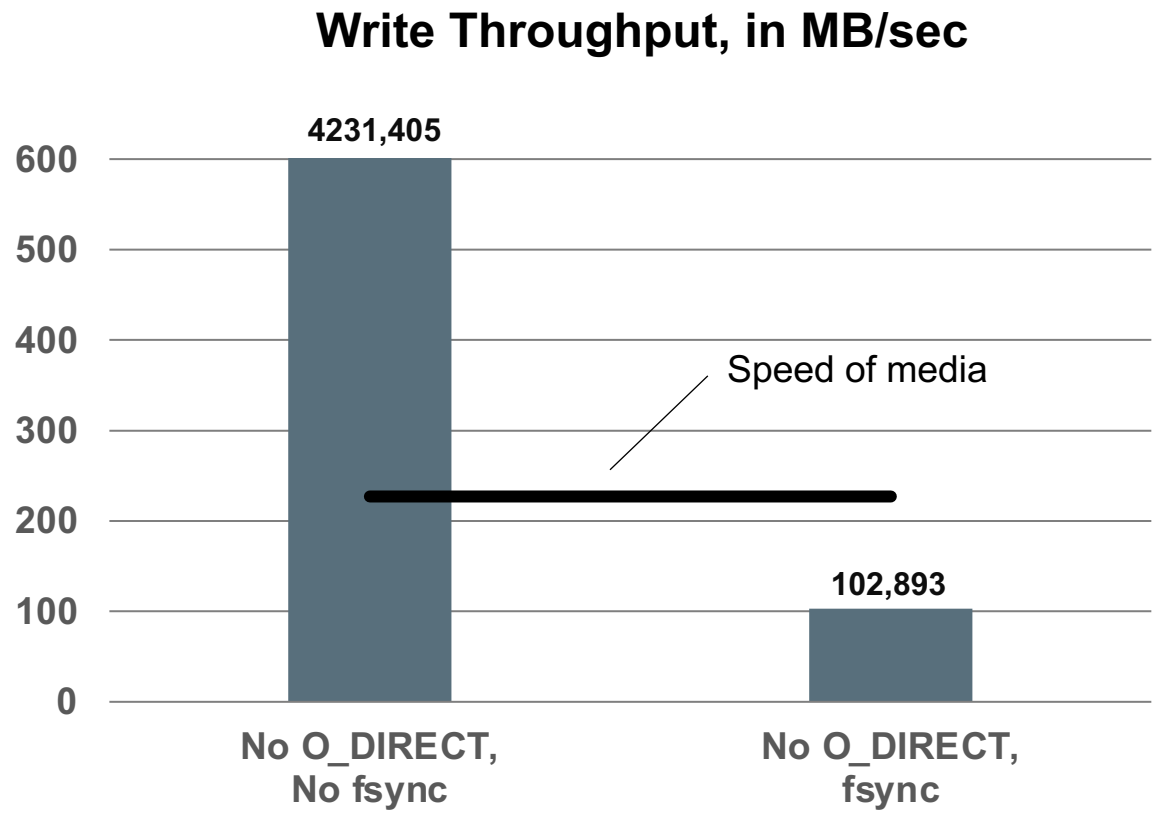
- File system might get corrupted

TUXERA

Flushing and fsync()

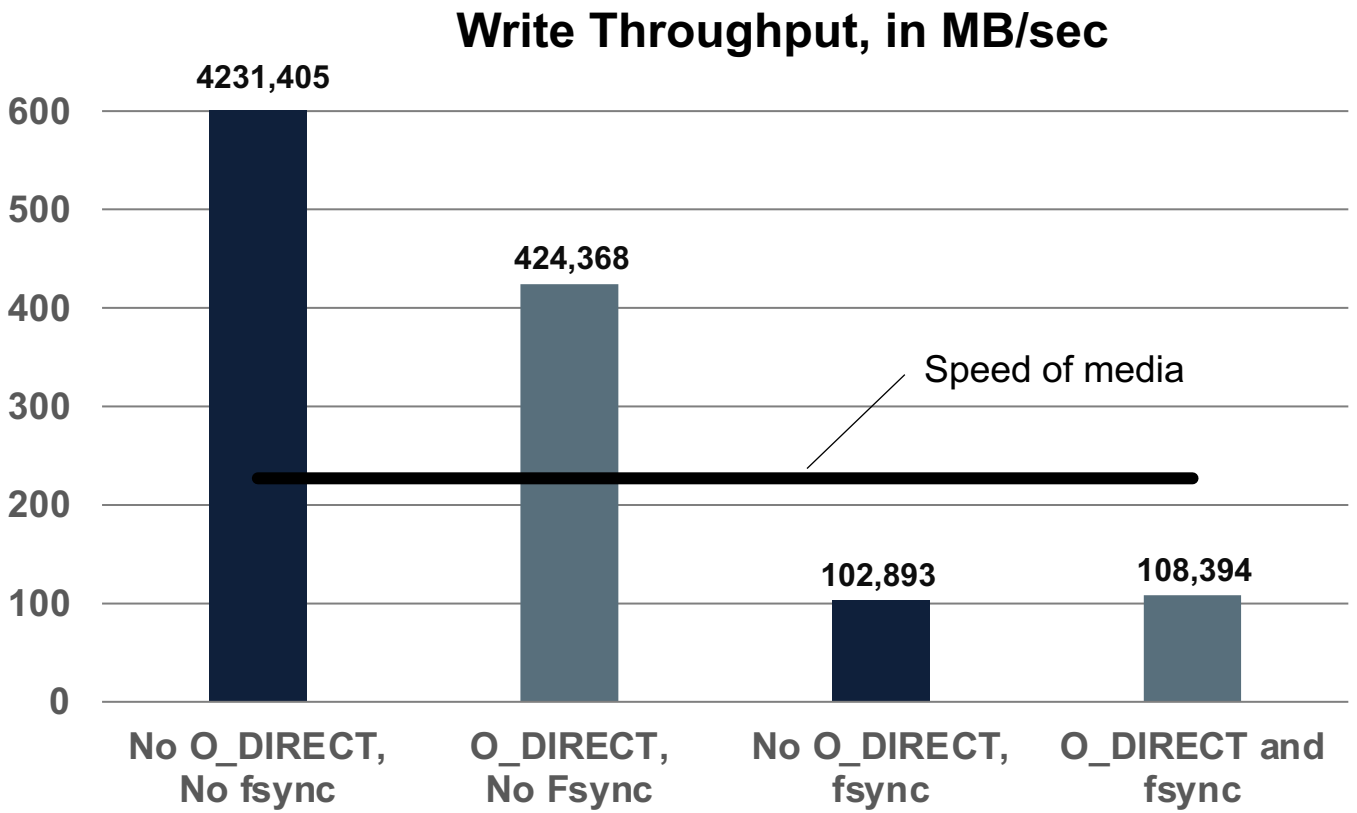
Transfers all modified data to permanent storage

- Writing through cache or flushing disk cache
- Some applications use frequent fsync()
  - Hurts performance
  - Increases write amplification
- Often used to ensure correct ordering
  - Works, but also commits to permanent storage



O_DIRECT

- Data must be on the media for reliability
- O_DIRECT bypasses Linux kernel caches
- Speed of the media is inflexible
  - Buffering if capture rate is higher
- Buffering allows preprocessing
  - Compression & Encryption

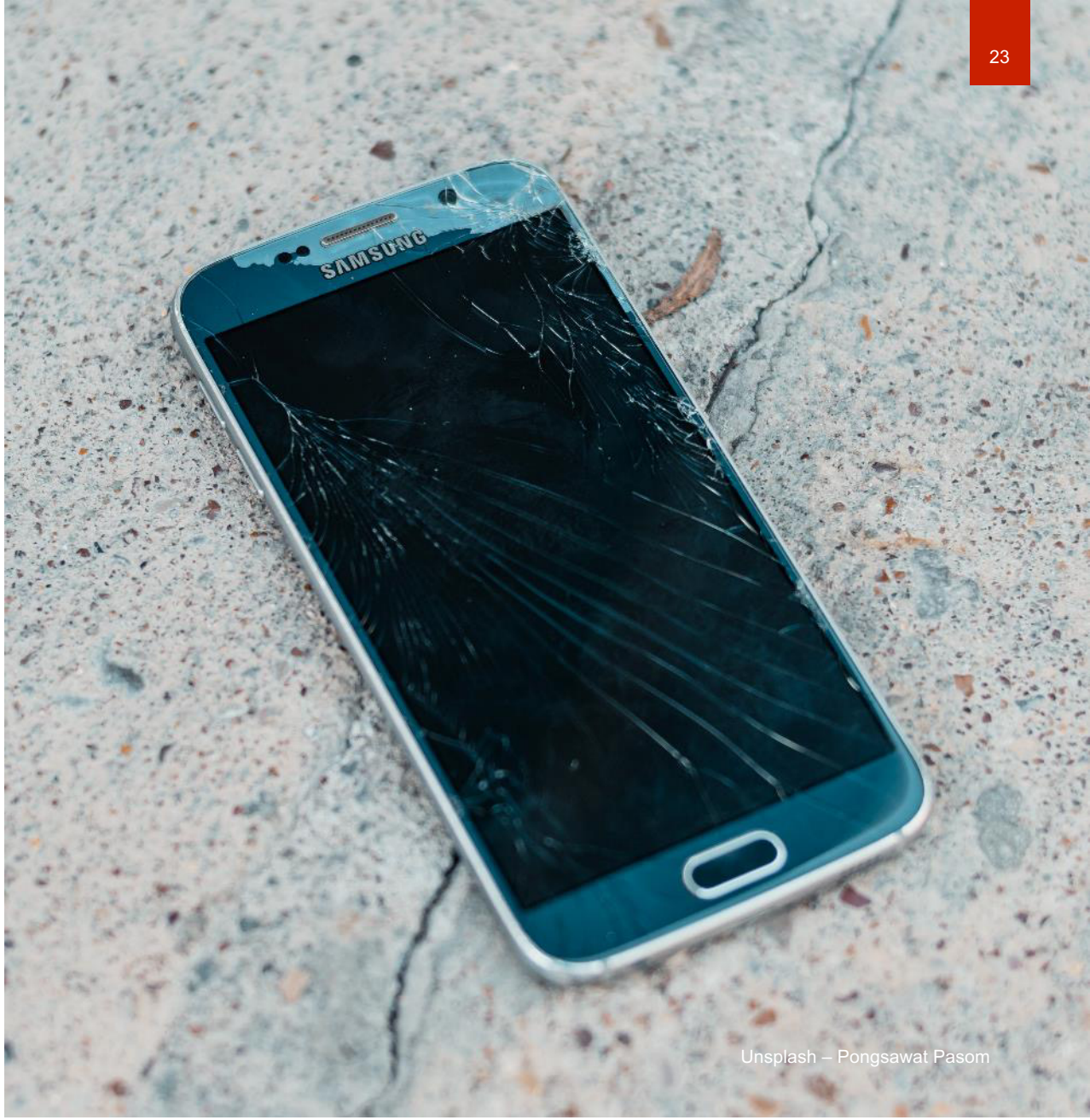


File systems

Limitations of FAT

Originating from desktop environment

- No powerloss protection
- Corruption on whole partition
- No support for security features



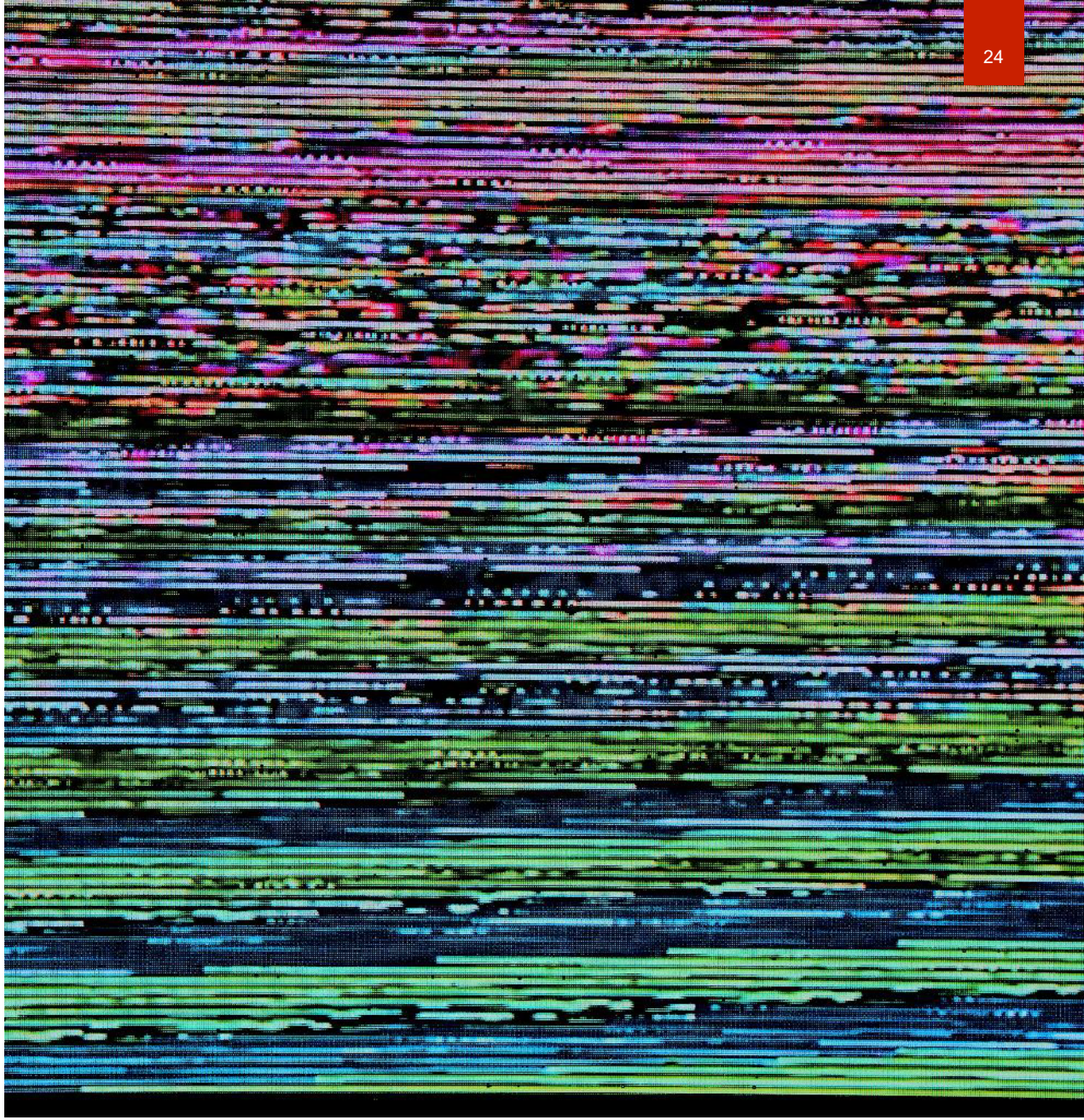
Unsplash – Pongsawat Pasom

TUXERA

Journaling File Systems

Reliability with additional structure

- Metadata changes written to journal
- Unclean shutdown – journal replay
  - Restoring correct media state
- Downsides
  - Inconsistent mount time
  - Additional overhead
  - Data Corruption can still occur



TUXERA

Transactional File System

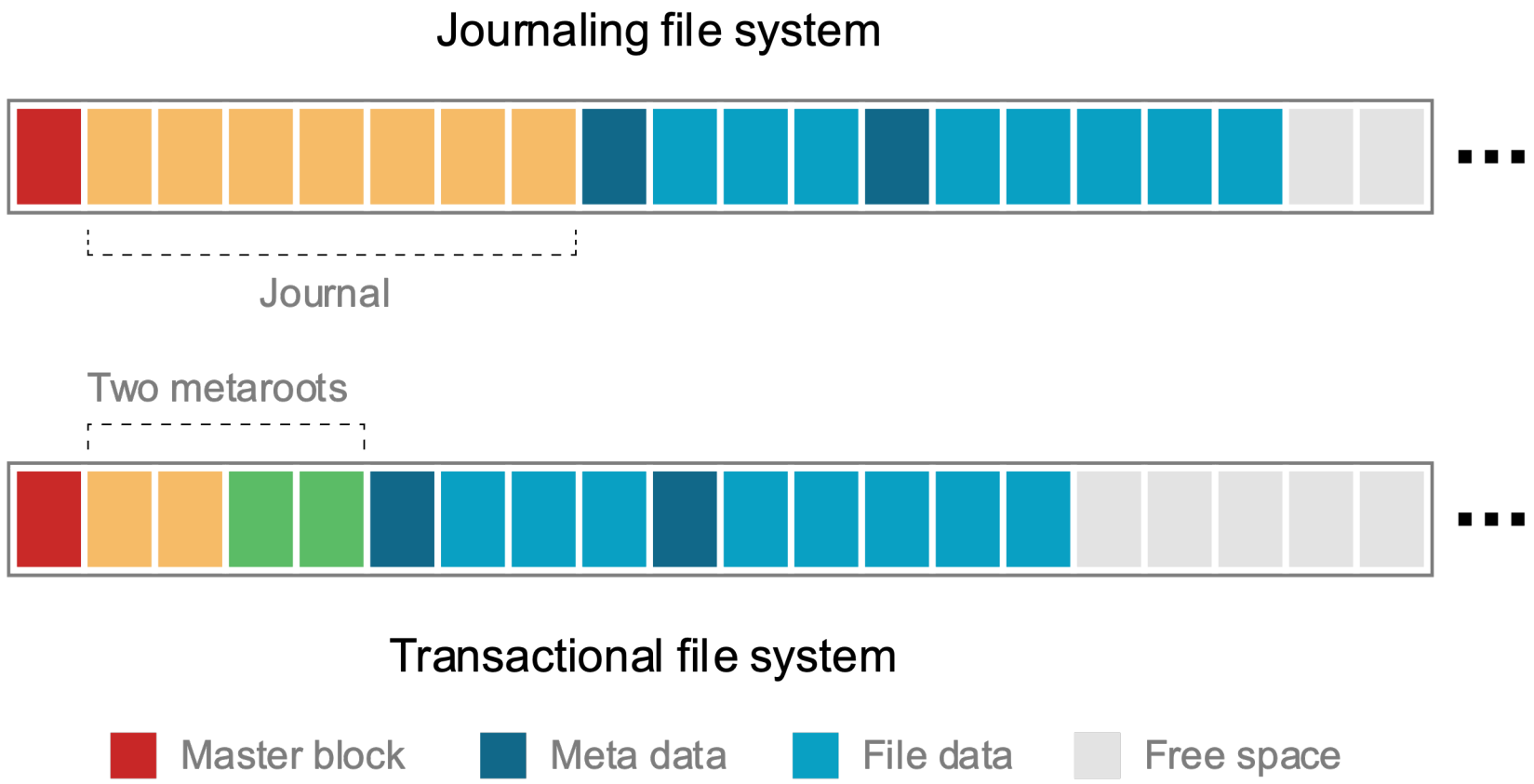
- Less overhead than journaling file system
  - Fast consistent mount time
- Writes changes only to the working state within transaction point

	FAT	Transactional
Lost Clusters	Fixable (chkdsk)	Cannot happen
Cross-Linked Files	Fixable (chkdsk)	Cannot happen
Invalid files/folders	Fixable (chkdsk)	Cannot happen

On media layout

Journaling file system
- ext2, ext3, ext4
- NTFS
- UBIFS

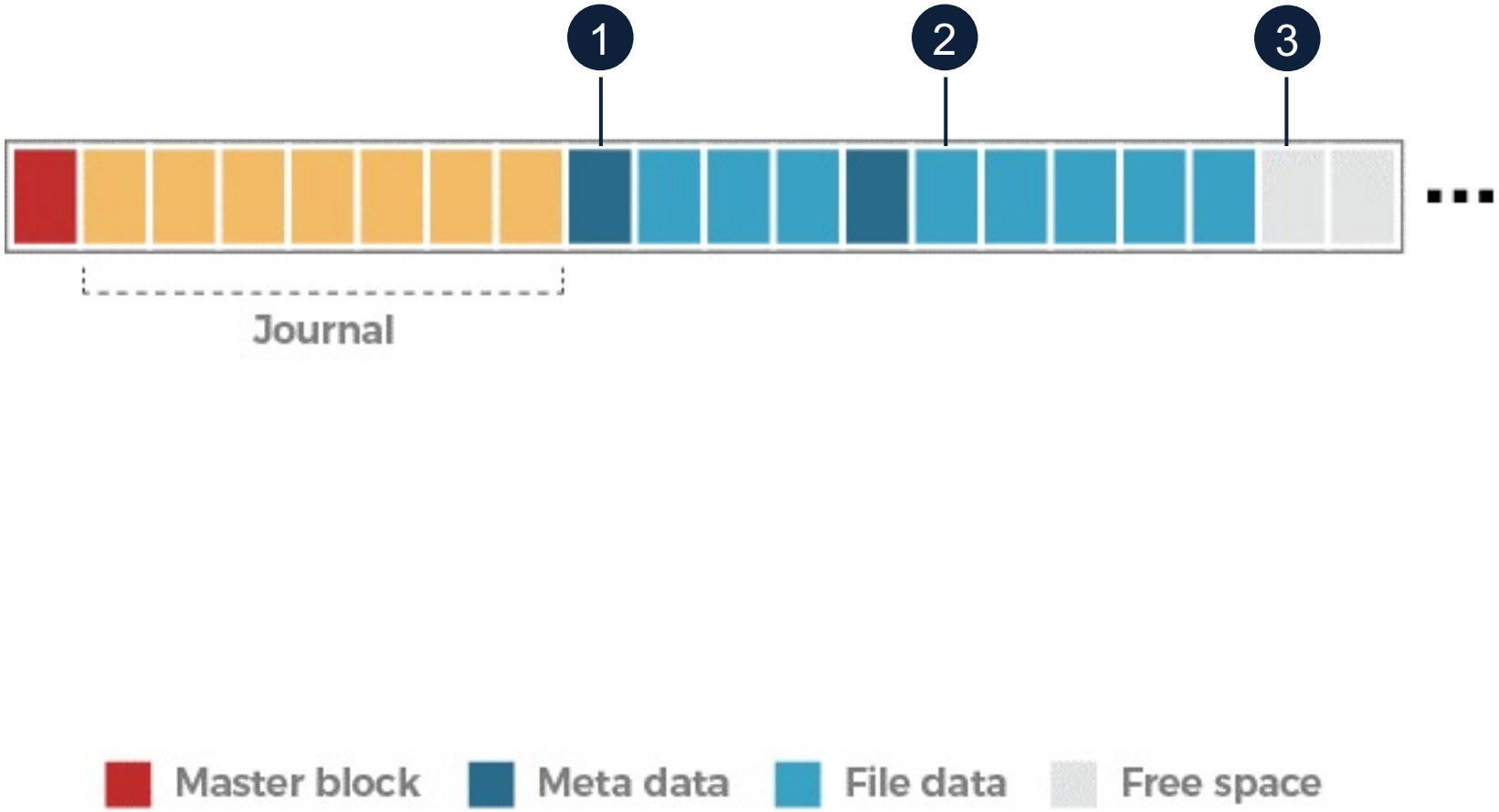
Transactional file system
- Reliance Nitro
- Reliance Edge



Journaling example

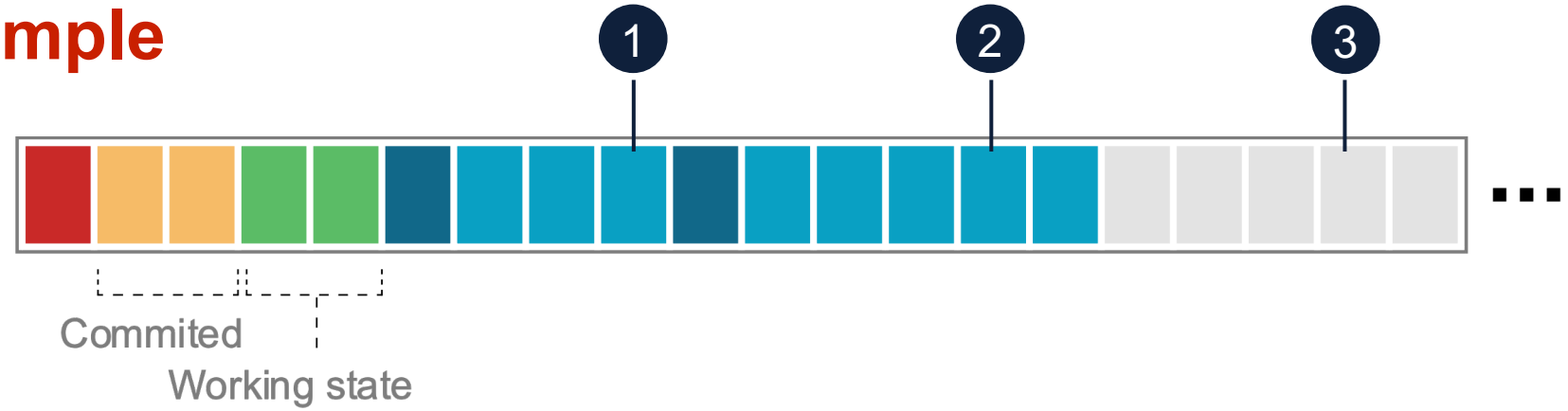
1. Delete a file
2. Modify a file
3. Create a file

No guarantees!



Transaction point example

1. **Delete a file**
2. **Modify a file**
3. **Create a file**

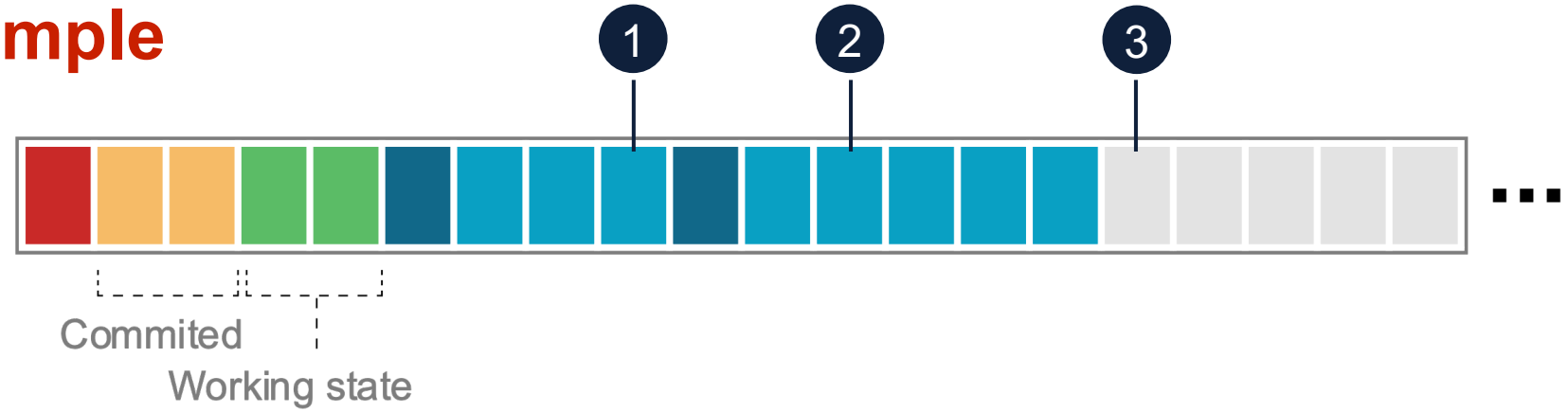


Success!

TUXERA

Transaction point example

1. **Delete a file**
2. **Modify a file**
3. **Create a file**



Interrupted!

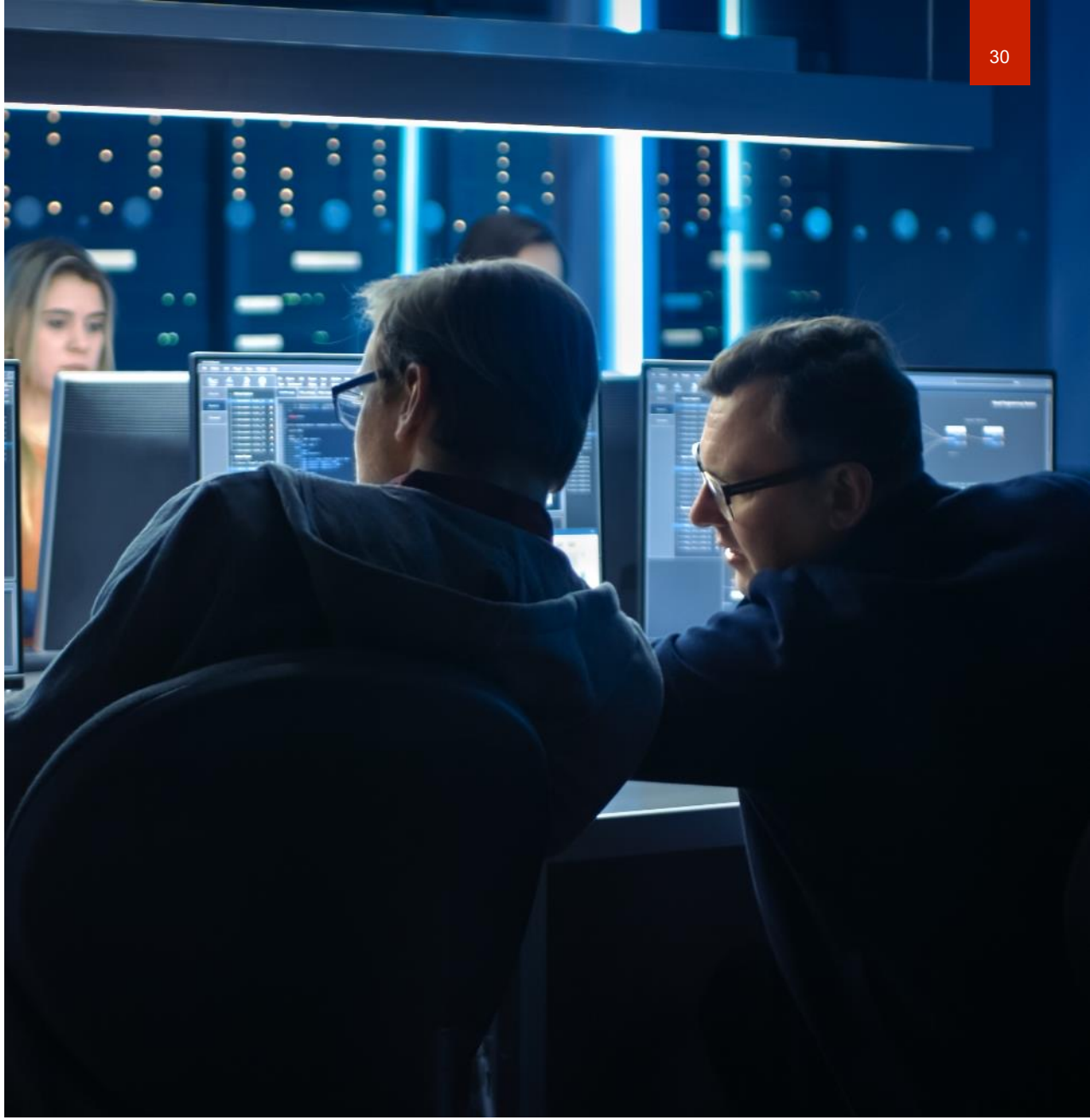
TUXERA

Effective Testing

- Performance is use case specific
- Power interruptions are commonplace for embedded devices
- Testing for System and Data Integrity
- Corner cases
  - Device full
  - Lots of files

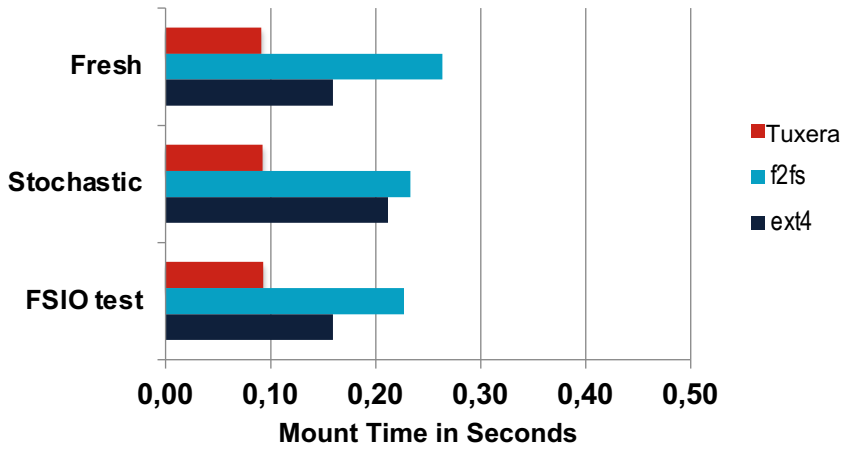
- No media corruption due to
  - Power failure
  - Unplugged storage

How you prepare your devices for power interruptions?

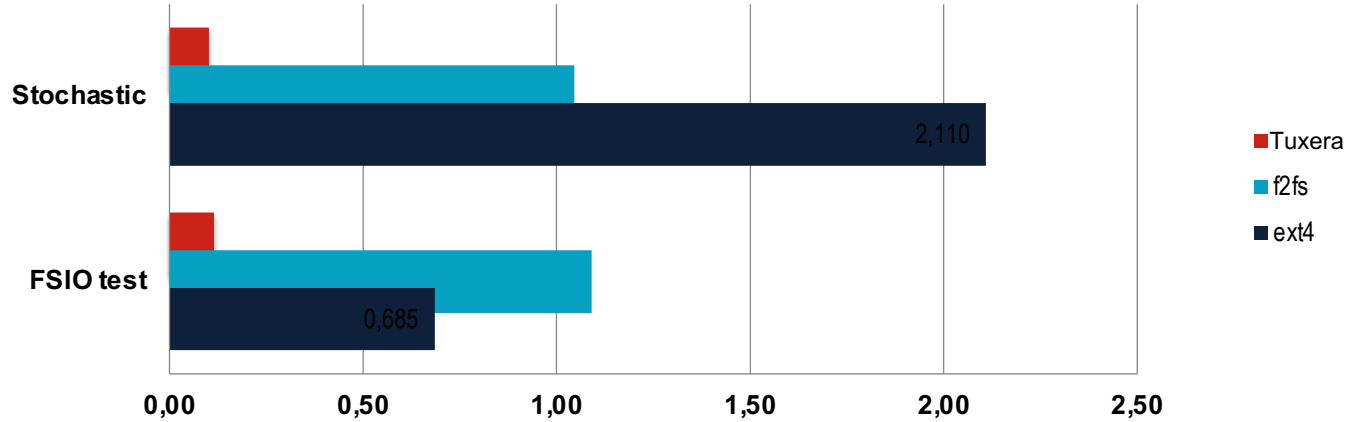


Reliability mechanisms impact mount time

After normal dismount



After power interruption



TUXERA

Summary

Summary

- Reliability requires Software and Hardware
  - Testing interaction is key for long lifetime
  - Power interruption testing is crucial for long-term reliability
- Programming techniques can help, but adds complexity
- File systems can help with dealing with complexity

Download our whitepaper on reliability

Questions & answers

sami@tuxera.com

TUXERA