2022 William Carter Award Recognition and Ceremony

Minesh Patel

DSN'22, Baltimore, MD

28 June 2022

Honored to be a Recipient

- Recognition in honor of William Carter

- I am proud to build upon the direction he set

Thank You!

My Ph.D. adviser { **Onur Mutlu**

Thank You!

My Ph.D. adviser — **Onur Mutlu**

Defense committee — **Mattan Erez**, **Moinuddin Qureshi**, **Vilas Sridharan**, **Christian Weis**

Additional letter writers — **Stefan Saroiu**, **Jared Zerbe**

Thank You!

Award sponsors { **IEEE TC-FTG**, **IFIP WG 10.4**

Organizers { **Conference chairs**, **Selection committee**

Thank You!

Friends
Colleagues
Mentors

SAFARI group
Internships
School (CMU, ETH)
Research community

Family

Parents (Alpa, Hamen)
Sister (Shreya)

Self-Introduction

- Originally from **Houston, TX**
  - Bachelors at UT Austin in 2015 (EE + Physics)

- Ph.D. with **Onur Mutlu** from **ETH Zürich**
  - Started at CMU in 2015
  - Defended October 1, 2021 (graduated April 2022)
  - Focused on **memory systems reliability**

- Currently exploring options for what comes next
  - Broadly interested in **architecture/systems topics**
  - Thinking about work in **industry research**

Dissertation Overview

"Enabling Effective Error Mitigation in Modern Memory Chips that Use On-Die ECC"

Defended Oct. 2021 (ETH Zürich)
Deposited Apr. 2022 (DOI 10.3929/ethz-b-000542542)

Advisor:

Onur Mutlu (ETH Zürich)

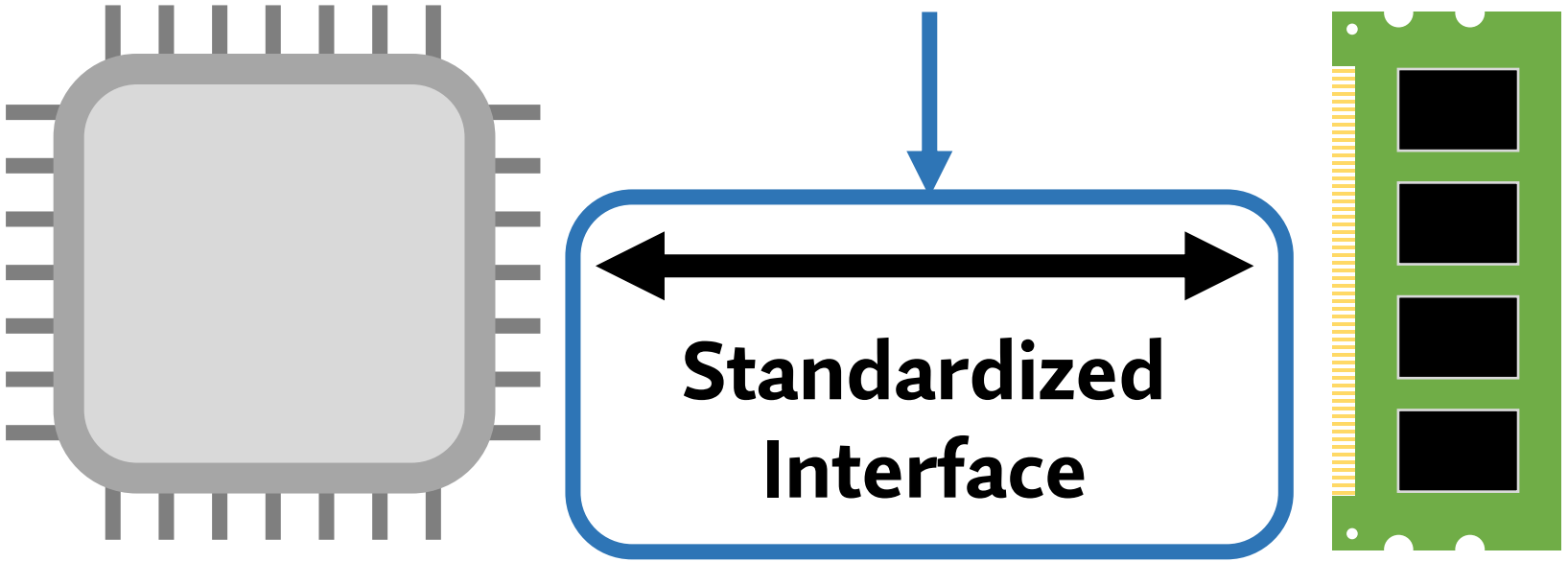
Co-Examiners:

Mattan Erez (UT Austin)
Moinuddin Qureshi (Georgia Tech)
Vilas Sridharan (AMD)
Christian Weis (TU Kaiserslautern)

ETH zürich

SAFARI

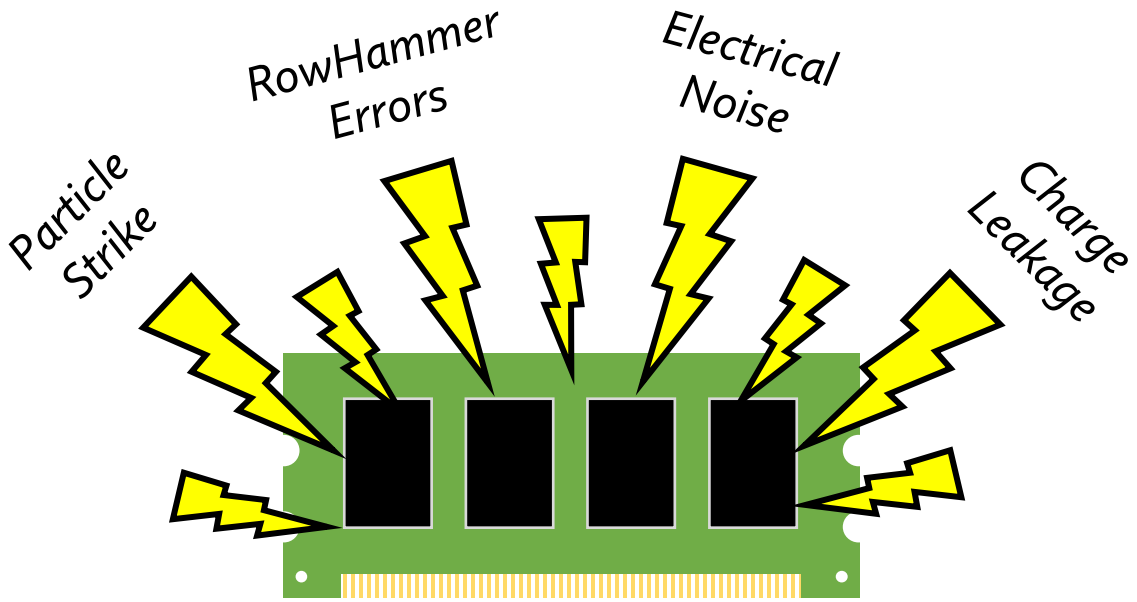
"Separation of Concerns"
between manufacturers



Processor

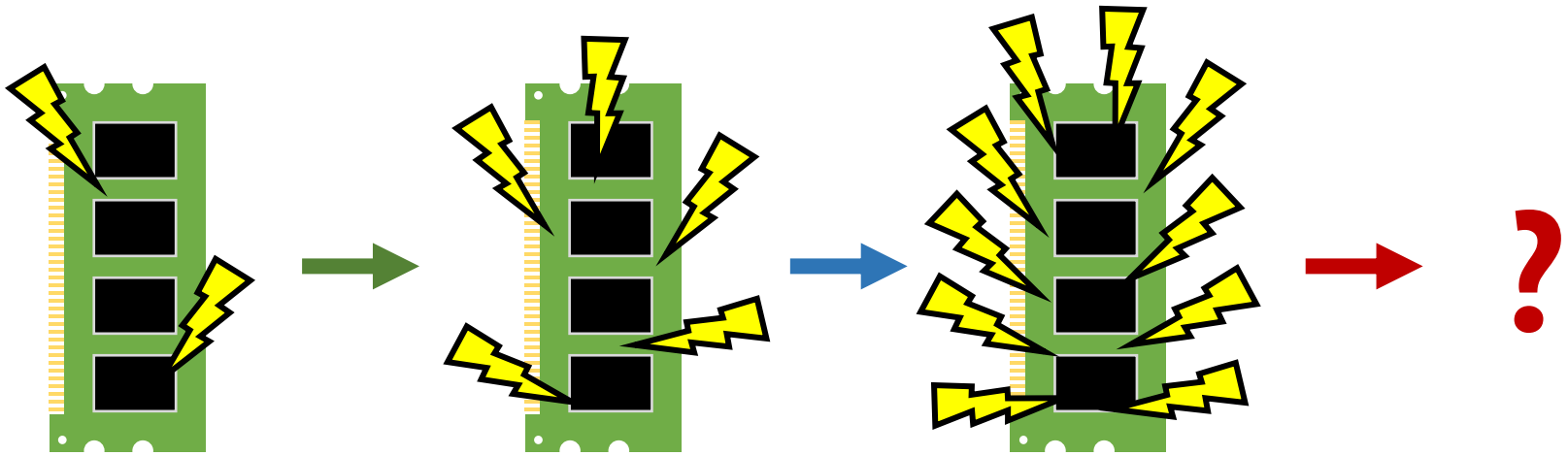
Main Memory (DRAM)

Enables each party to **solve** their own **design challenges**



Challenge:

DRAM suffers from **errors** that cause **data loss** or **system failure** if ignored



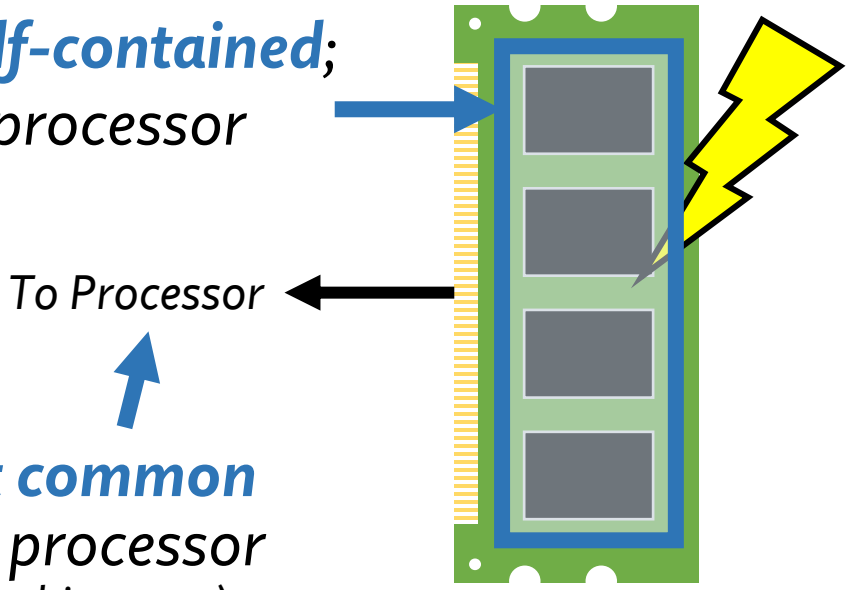
Manufacturers' primary goal is to increase storage density, but this **exacerbates** errors

1. **Increases costs** for manufacturers and consumers
2. **Limits** systems' overall potential for growth

Solution: Error Mitigation Techniques

Recently, DRAM manufacturers started using on-die error-correcting codes **(on-die ECC)**

Proprietary *and* ***self-contained***;
Invisible *to the processor*



Hides the ***most common*** *errors from the processor*
(e.g., random single-bit errors)

Simple and **low-cost**

Preserves **trade secrets** of DRAM manufacturers

Convenient for many **commodity systems**

Limited error correction capability for low cost

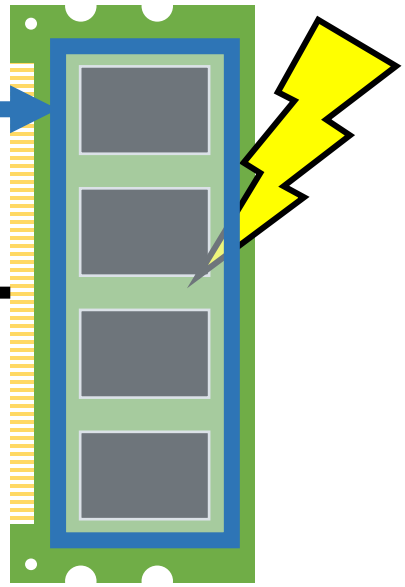
Partial error-correction can **complicate** system design and test

Proprietary *and* ***self-contained***;
Invisible *to the processor*

Addressing **all errors** in DRAM is **very expensive**

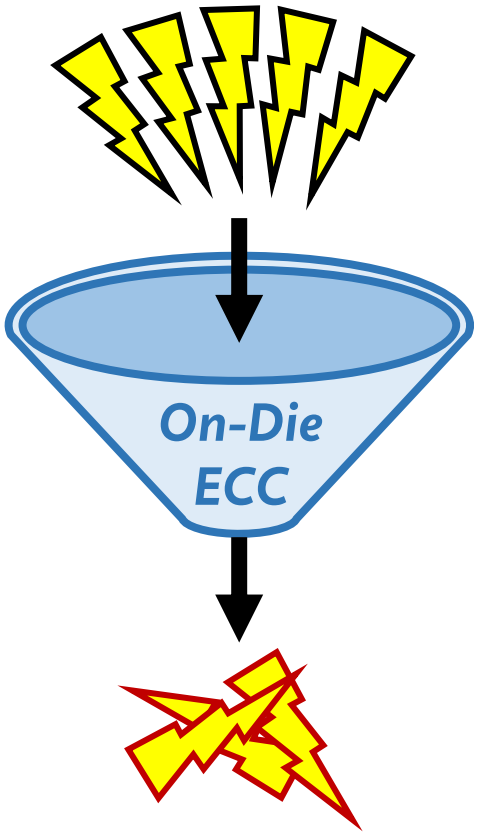
To Processor

Hides the ***most common*** *errors from the processor*
(e.g., random single-bit errors)



Problems Introduced by On-Die ECC

On-die ECC negatively impacts system **design** and **test** efforts



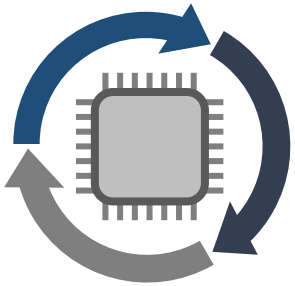
Predictable and/or **well-understood** errors due to **physical processes**

Unknown filtration from on-die ECC partially correcting the errors

Unpredictable, **obfuscated** errors that are hard to **understand** or **reason about**

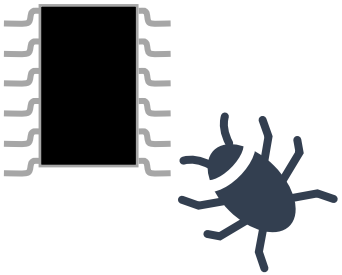
Parties Impacted by Obfuscated Errors

- **Anyone** who must understand error characteristics in the course of their work is potentially affected



Error-Mitigation Designers

Forced to make **limiting assumptions** (e.g., worst-case behavior) that lead to **inefficient designs**



Third-Party Testers

Hard to **debug observed errors** because on-die ECC conceals the **underlying cause**

Research Scientists

Experimental studies of DRAM technology characteristics polluted by **on-die ECC artifacts**

Thesis Statement

Exploit the interaction between **on-die ECC** and the **statistical characteristics** of memory errors

We can use **new memory testing techniques** to recover the **error characteristics** that **on-die ECC obfuscates**

Enable scientists and engineers to make **informed decisions** towards building robust systems

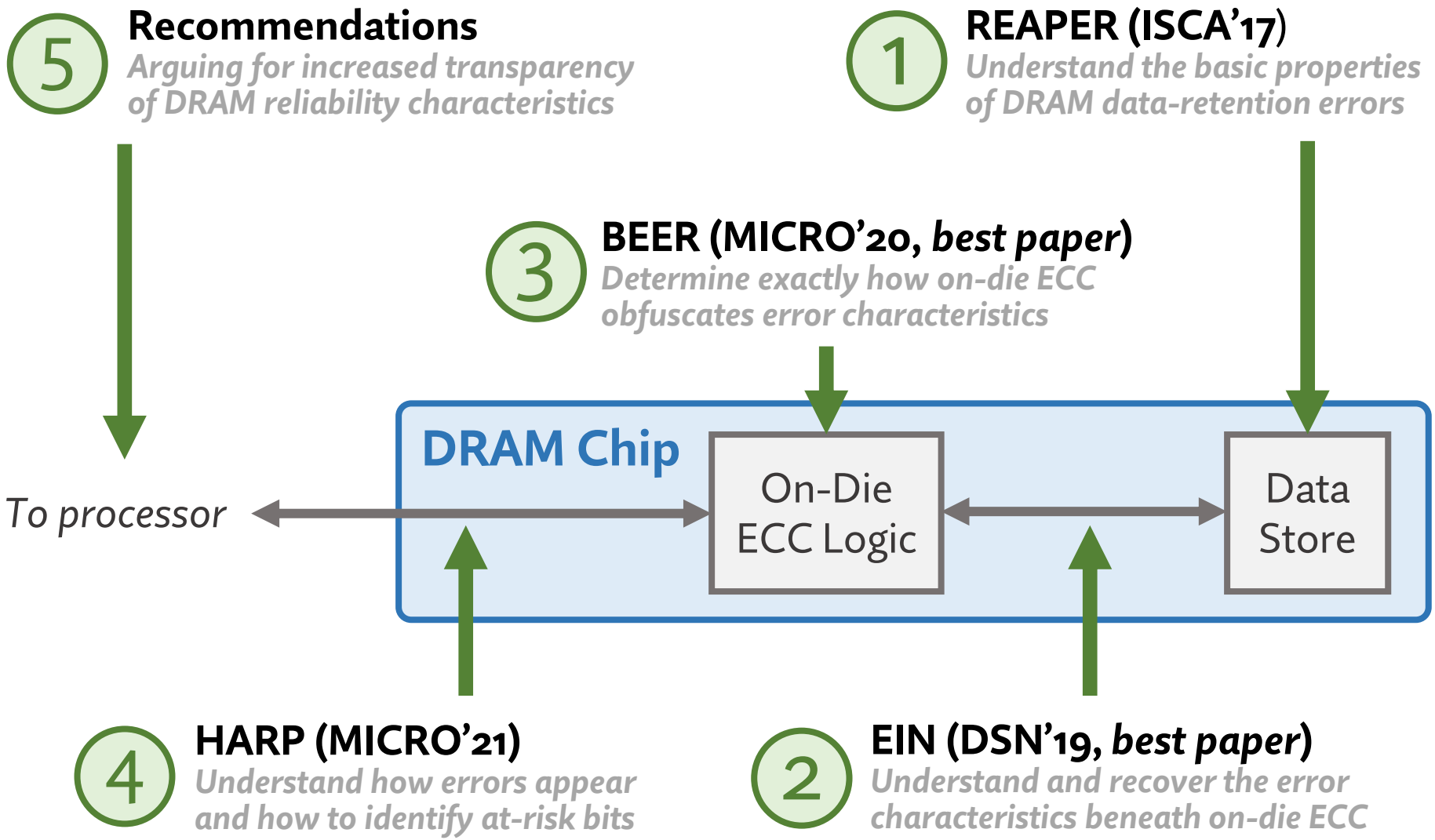
Thesis Statement (Verbatim)

1.3 Thesis Statement

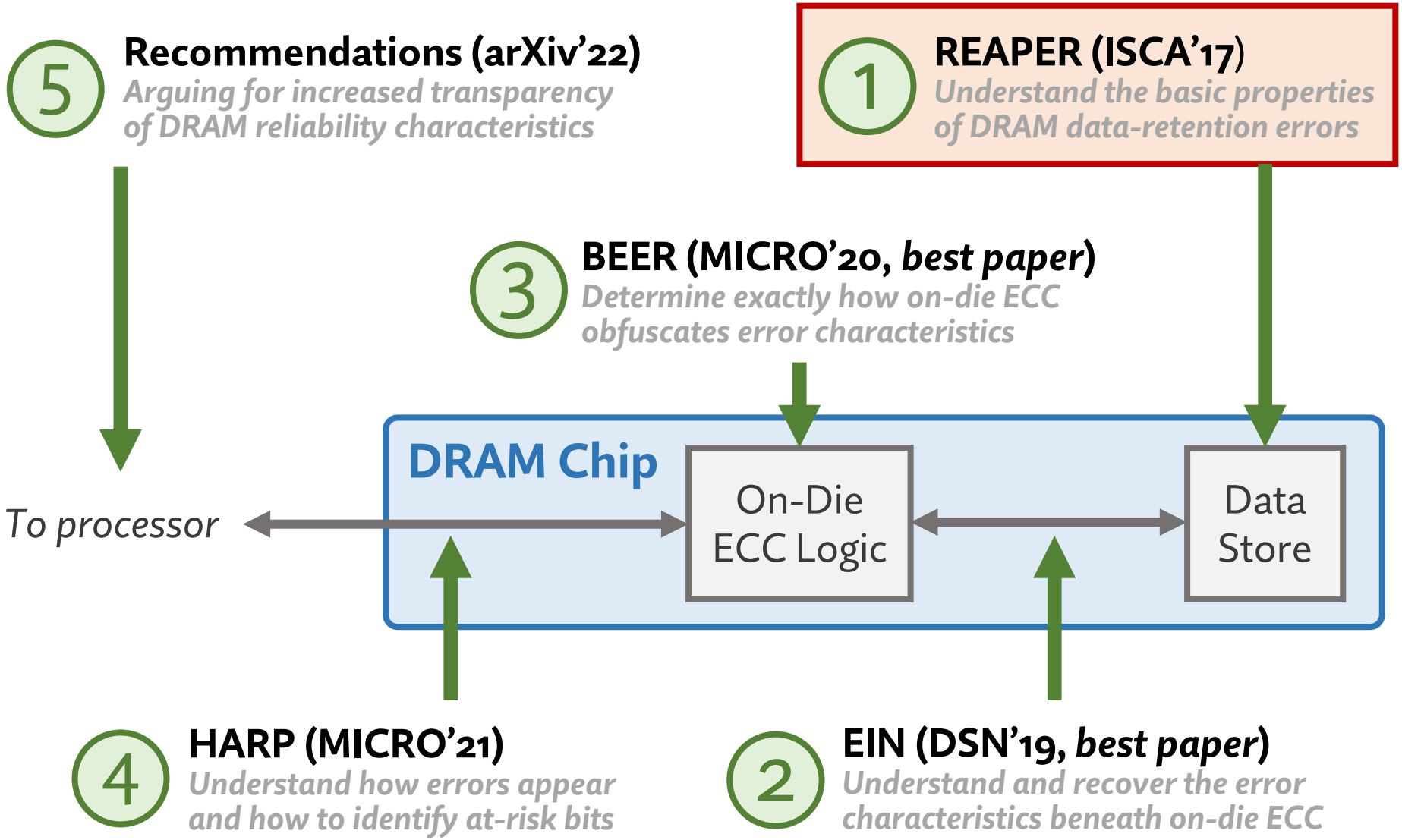
Our approach is encompassed by the following thesis statement:

> *The error characteristics that on-die ECC obfuscates can be recovered using new memory testing techniques that exploit the interaction between on-die ECC and the statistical characteristics of memory error mechanisms to expose physical cell behavior, thereby enabling scientists and engineers to make informed decisions towards building smarter and more robust systems.*

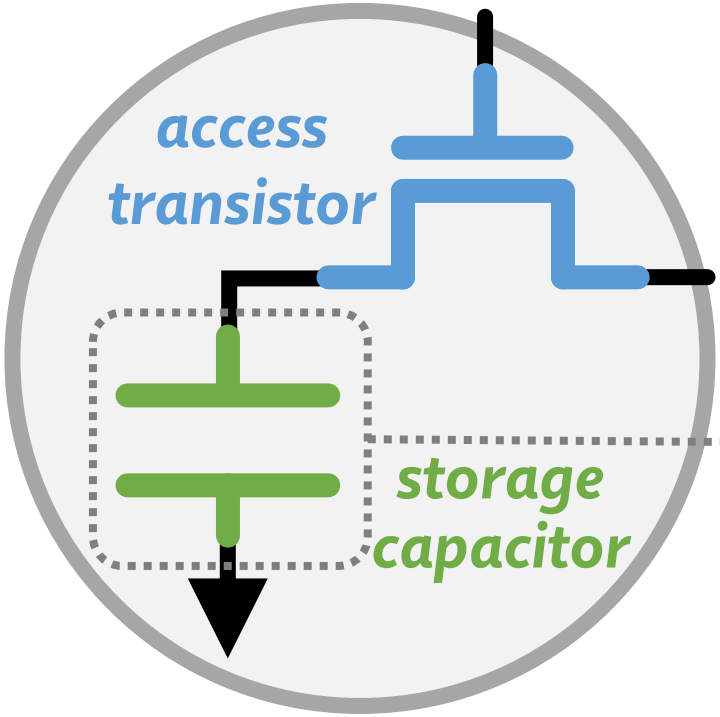
Core Contributions



Core Contributions



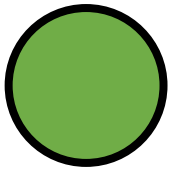
DRAM Cell



stores one bit of data

Data Encoding

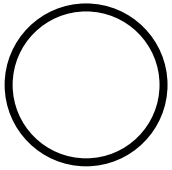
"charged"



= 1 or 0

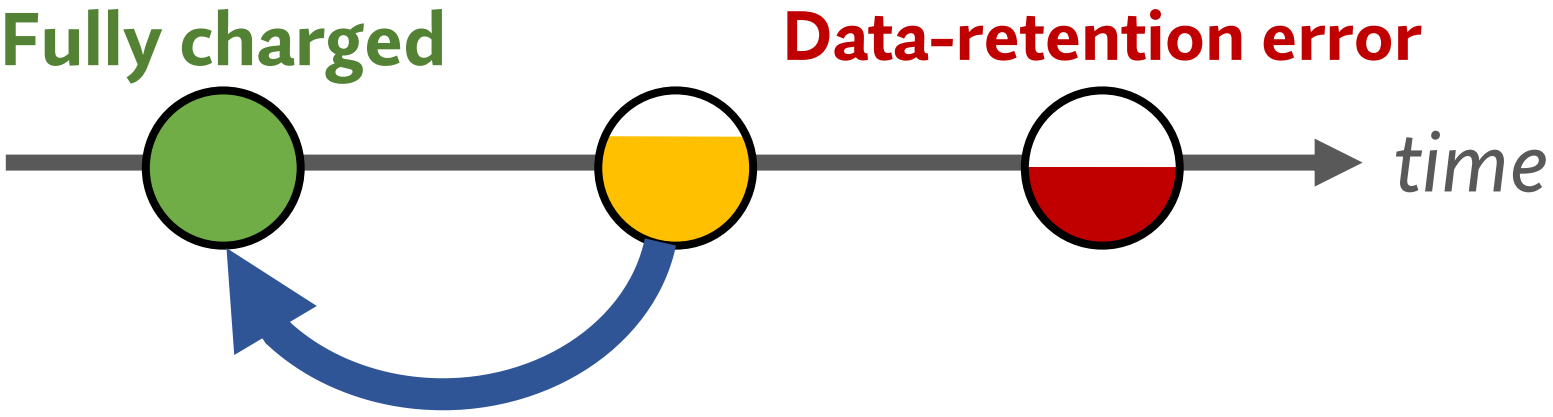
design-dependent

"discharged"



= 0 or 1

DRAM cells **leak charge** over time



DRAM Refresh

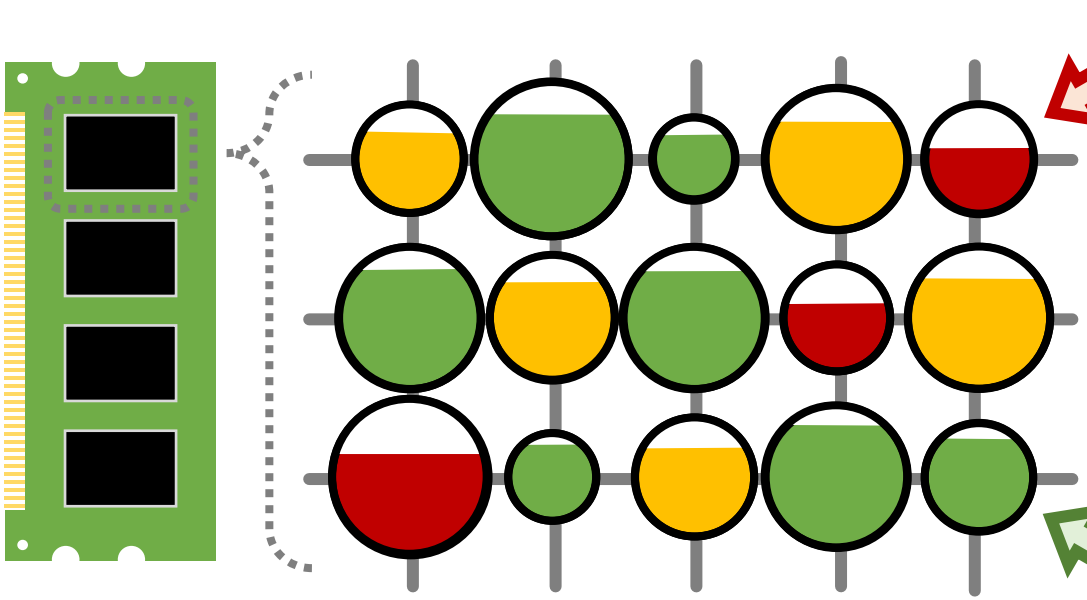
Periodically restores the charge of all cells to prevent data-retention errors

⇩

Significant performance and energy overhead

Making Refresh More Efficient

Only a **few cells** require frequent refreshing



fast-leaking

Hard to identify

1. *Process, voltage, temperature*
2. *Variable retention time*
3. *Data pattern dependence*

slow-leaking

Goal: **quickly** and **efficiently** identify the **error-prone** cells

Experimental Error Characterization

- We study the **data-retention error characteristics** in 368 real LPDDR4 DRAM chips

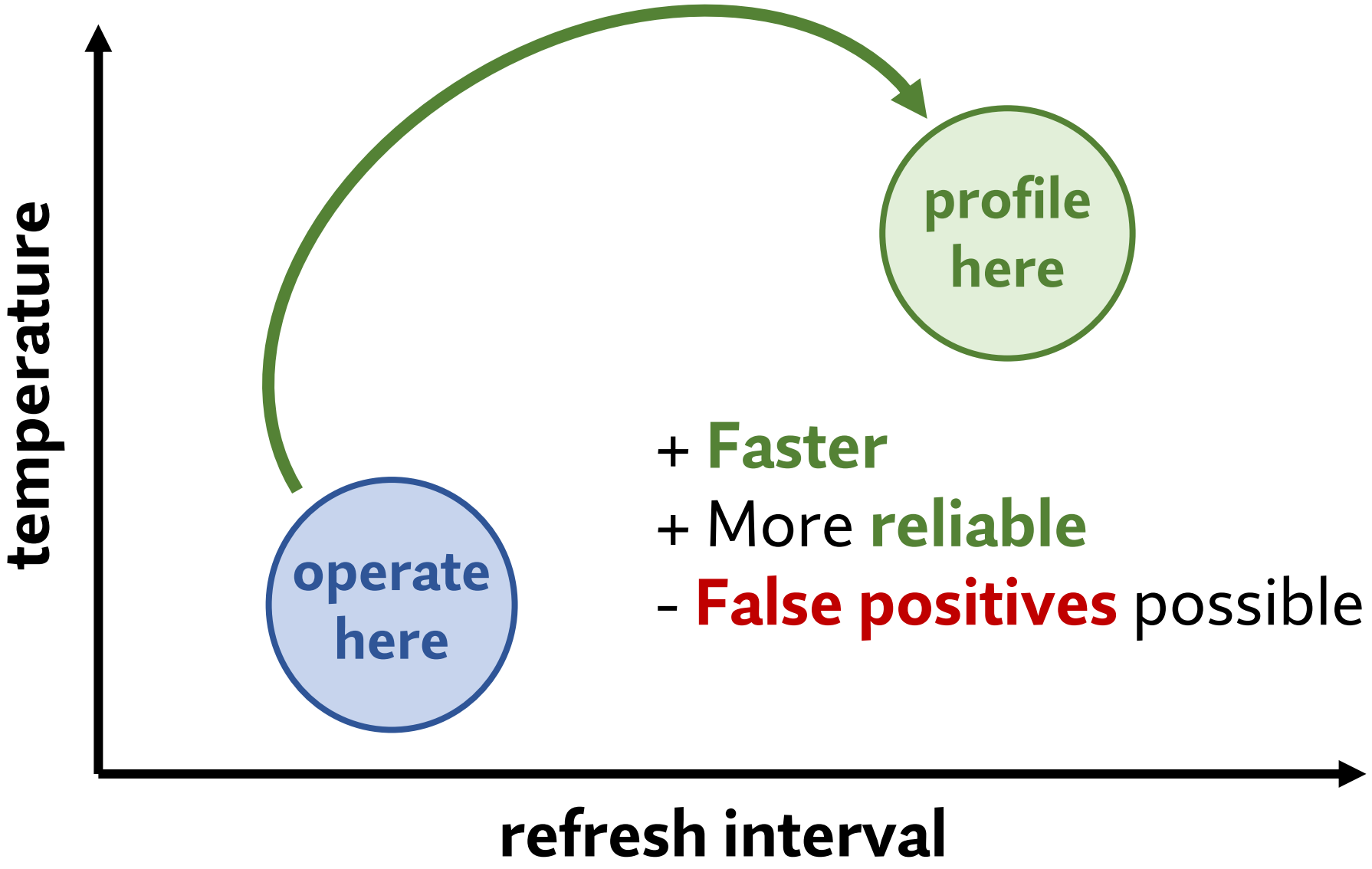
1

Cells are **more likely** to fail at an **increased** (1) refresh interval; or (2) temperature

2

Profiling involves a complex **tradeoff space:** (1) **speed**; (2) **coverage**; and (3) **false positives**

Reach Profiling



Evaluating Reach Profiling

1. **2.5x faster** than the **state-of-the-art baseline** for 99% coverage and a 50% false positive rate
   - **Even faster** (>3.5x) with more false positives (>100%)

2. Enables operating at **longer refresh intervals** by reducing the overall profiling overhead
   - 16.3% **end-to-end performance** improvement
   - 36.4% **DRAM power** reduction

The Reach Profiler (REAPER): Enabling the Mitigation of DRAM Retention Failures via Profiling at Aggressive Conditions

Minesh Patel[§‡] Jeremie S. Kim[‡§] Onur Mutlu[§‡]
[§]ETH Zürich [‡]Carnegie Mellon University

ABSTRACT

Modern DRAM-based systems suffer from significant energy and latency penalties due to conservative DRAM refresh standards. Volatile DRAM cells can retain information across a wide distribution of times ranging from milliseconds to many minutes, but each cell is currently refreshed every 64ms to account for the extreme tail end of the retention time distribution, leading to a high refresh overhead. Due to poor DRAM technology scaling, this problem is expected to get worse in future device generations. Hence, the current approach of refreshing all cells with the worst-case refresh rate must be replaced with a more intelligent design.

Many prior works propose reducing the refresh overhead by extending the default refresh interval to a higher value, which we refer to as the *target refresh interval*, across parts or all of a DRAM chip. These proposals handle the small set of failing cells that cannot retain data throughout the entire extended refresh interval via *retan*

KEYWORDS

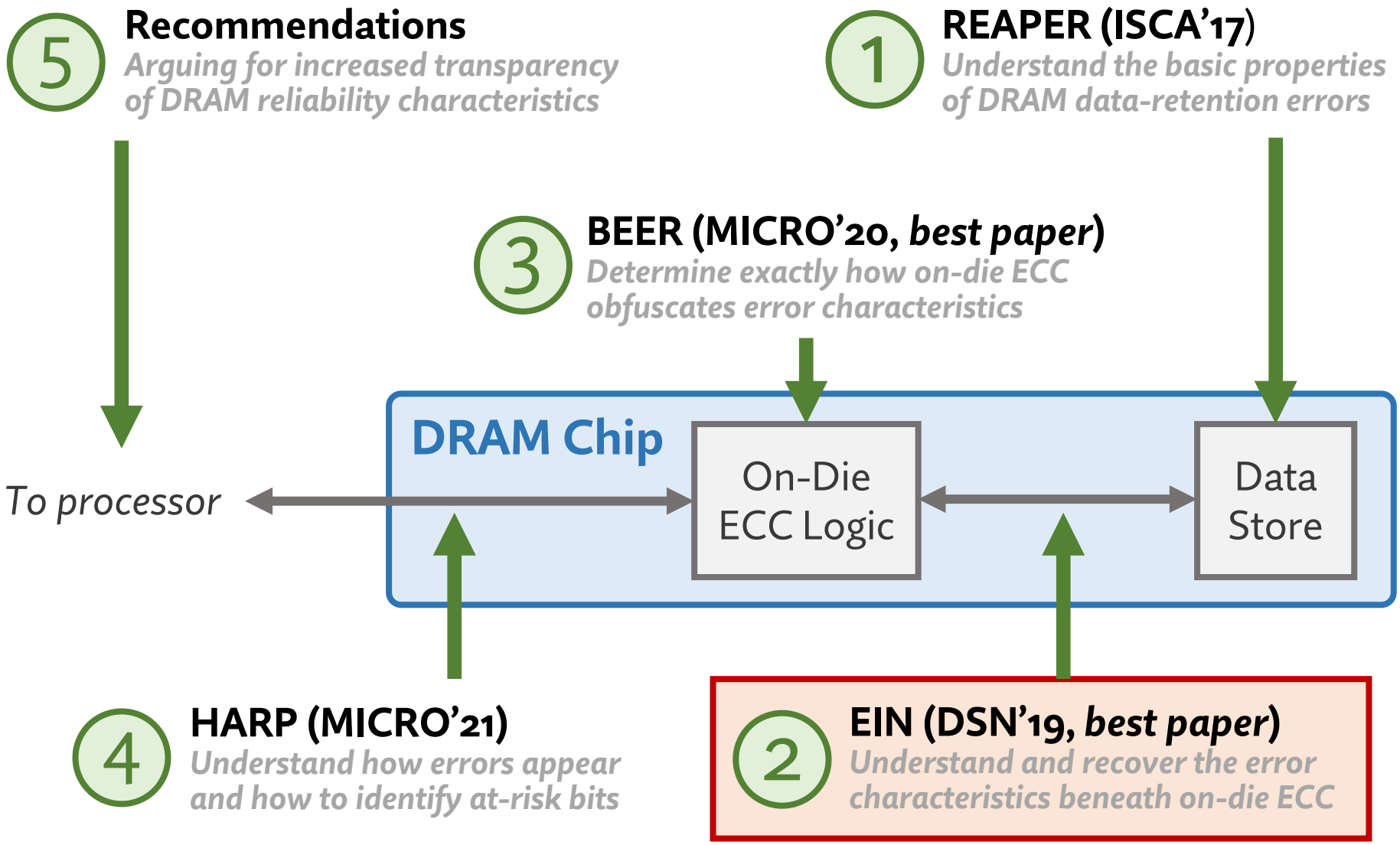
DRAM, refresh, retention failures, reliability, testing, memory

1 INTRODUCTION

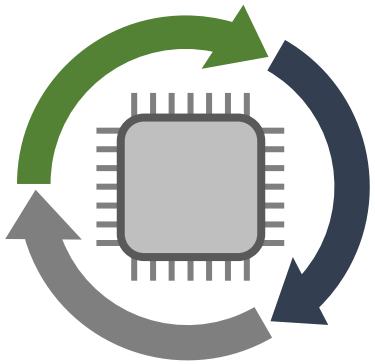
DRAM stores data in volatile capacitors that constantly leak charge and therefore requires periodic charge restoration to maintain data correctness. As cell capacitor sizes decrease with process scaling and the total number of cells per chip increases each device generation [36], the total amount of time and energy required to restore all cells to their correct value, a process known as DRAM *refresh*, scales unfavorably [23, 41, 63]. The periodic refresh of DRAM cell capacitors consumes up to 50% of total DRAM power [63] and incurs large performance penalties as DRAM cells are unavailable during refresh [23, 63, 73, 75].

Minesh Patel, Jeremie S. Kim, and Onur Mutlu,
"The Reach Profiler (REAPER): Enabling the Mitigation of DRAM Retention Failures via Profiling at Aggressive Conditions"
Proceedings of the 44th International Symposium on Computer Architecture (***ISCA***),
Toronto, Canada, June 2017.

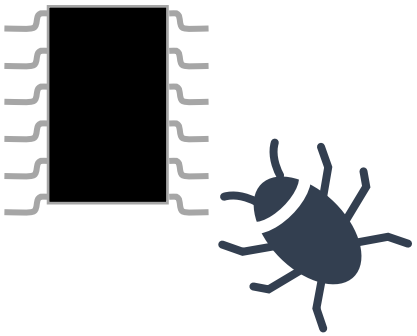
Core Contributions



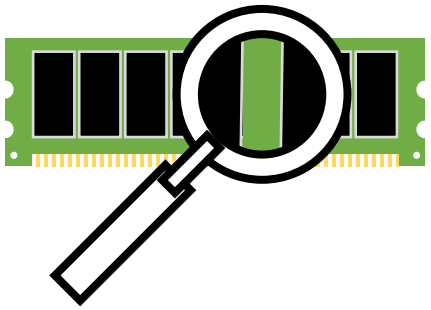
Third-Party DRAM Users



System Architects



Test Engineers



Research Scientists

Study **DRAM errors** to understand a DRAM chip's **reliability characteristics**

Expected error rates?
'Weak' cell locations?
Inter-chip variation?
Temperature dependence?
Statistical error properties?
Minimum operating timings?

Third-Party DRAM Users

System Architects

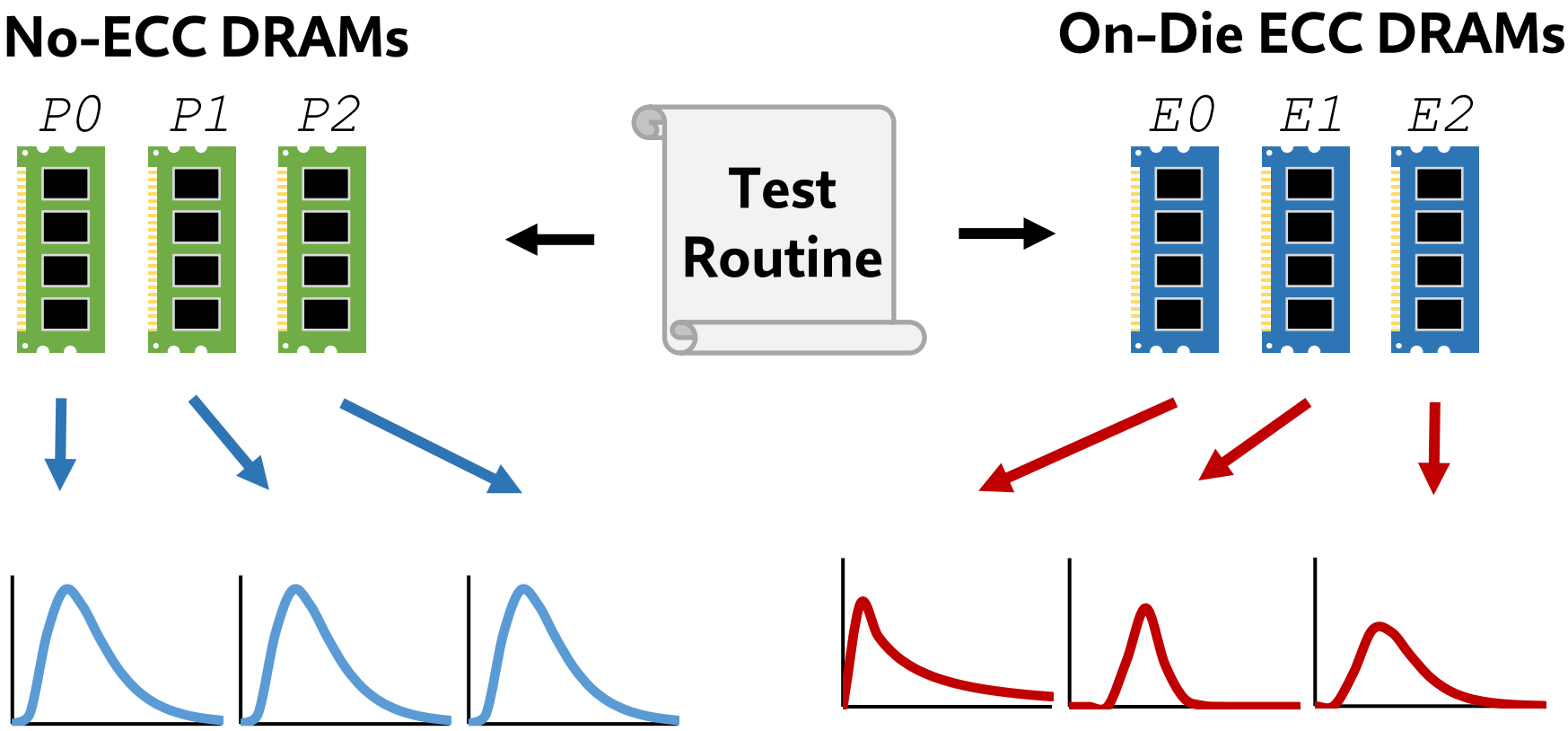
Test Engineers

Research Scientists

Study **DRAM errors** to understand
a DRAM chip's **reliability characteristics**

Gain **exploitable insights**
to improve performance, reliability, etc.

On-Die ECC Interferes with Studying Errors



Well-Understood Error Distributions

- *Based on physical properties of DRAM*
- *Easy to reason about and understand*

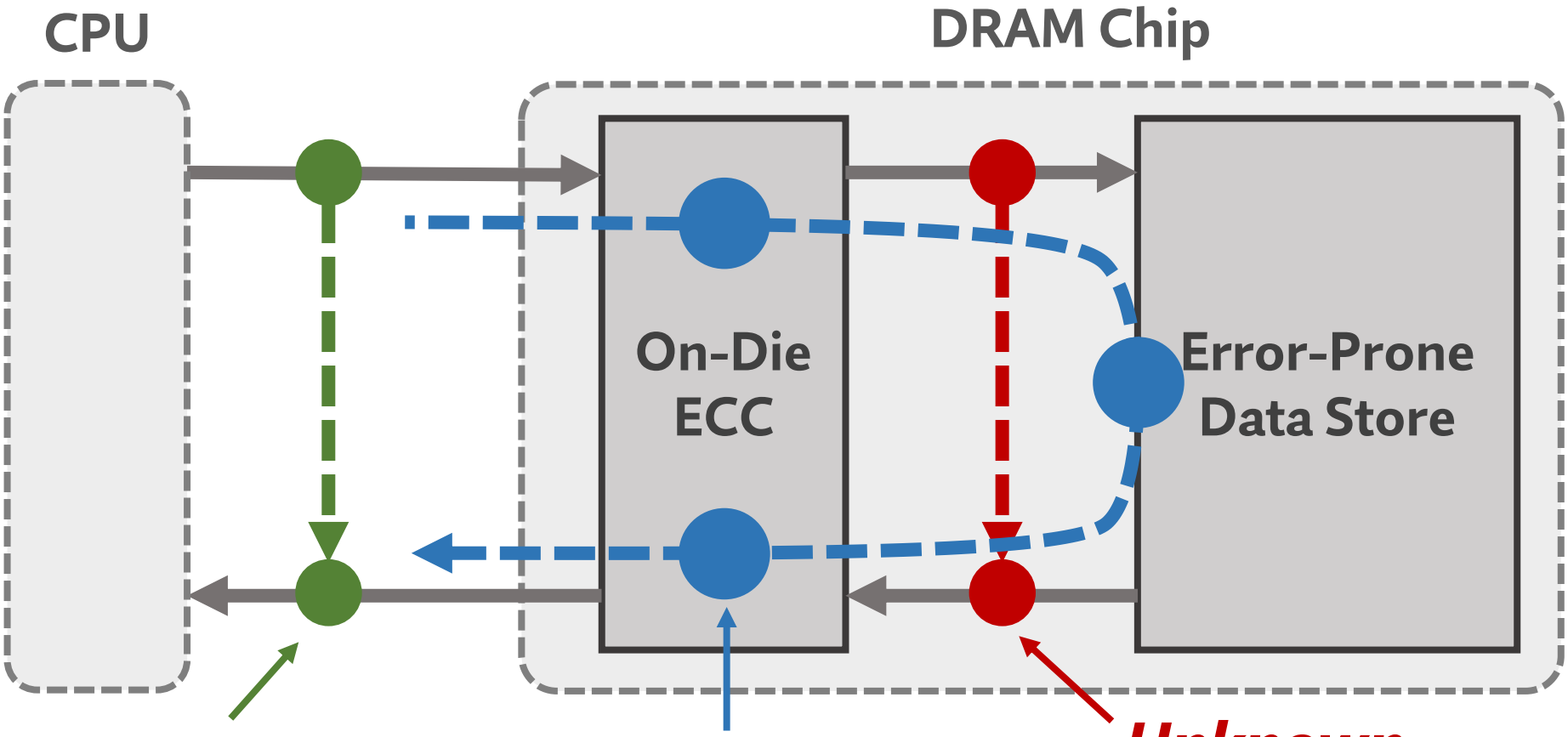
Unpredictable Error Distributions

- *Dependent on ECC implementation*
- *Hard to reason about and predict*

On-Die ECC Interferes with Studying Errors

Our goal:
Recover the **error characteristics** that on-die ECC **obfuscates**

Key Idea: Statistical Inference



Known
Directly observable

Inferable
By using **statistical methods**

Unknown
But **predictable** based on **well-understood** DRAM error mechanisms

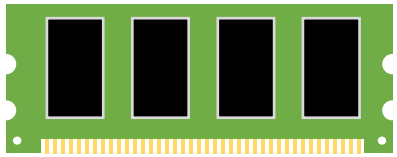
EIN: Error Inference Methodology

1 **Define Experimental Setup**
e.g., testing parameters, DRAM chips

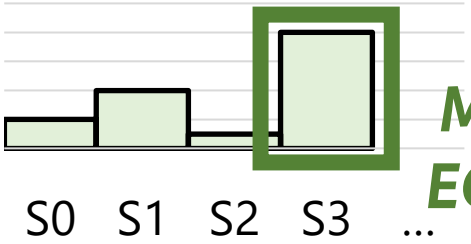
2 **Simulate Suspected ECCs**
e.g., Hamming, BCH, etc.

Monte-Carlo Simulation
https://github.com/CMU-SAFARI/EINSim

3 **Real-Chip Experiments**
with unknown ECC scheme



4 **Perform Inference**
Maximum-a-priori (MAP) estimation



Most Likely ECC Scheme

Applying EIN to Real Chips

- Apply EIN to **314 real LPDDR4 DRAM** chips

- Show that EIN can **infer** both:
  - The **ECC scheme** to be a (136, 128) Hamming code
  - **Raw bit error rates** of data-retention errors

- EIN works **without**:
  - Visibility into the ECC mechanism
  - Disabling ECC
  - Tampering with the hardware

Best Paper Award, DSN 2019

Understanding and Modeling On-Die Error Correction in Modern DRAM: An Experimental Study Using Real Devices

Minesh Patel[†] Jeremie S. Kim[‡†] Hasan Hassan[†] Onur Mutlu[†‡]

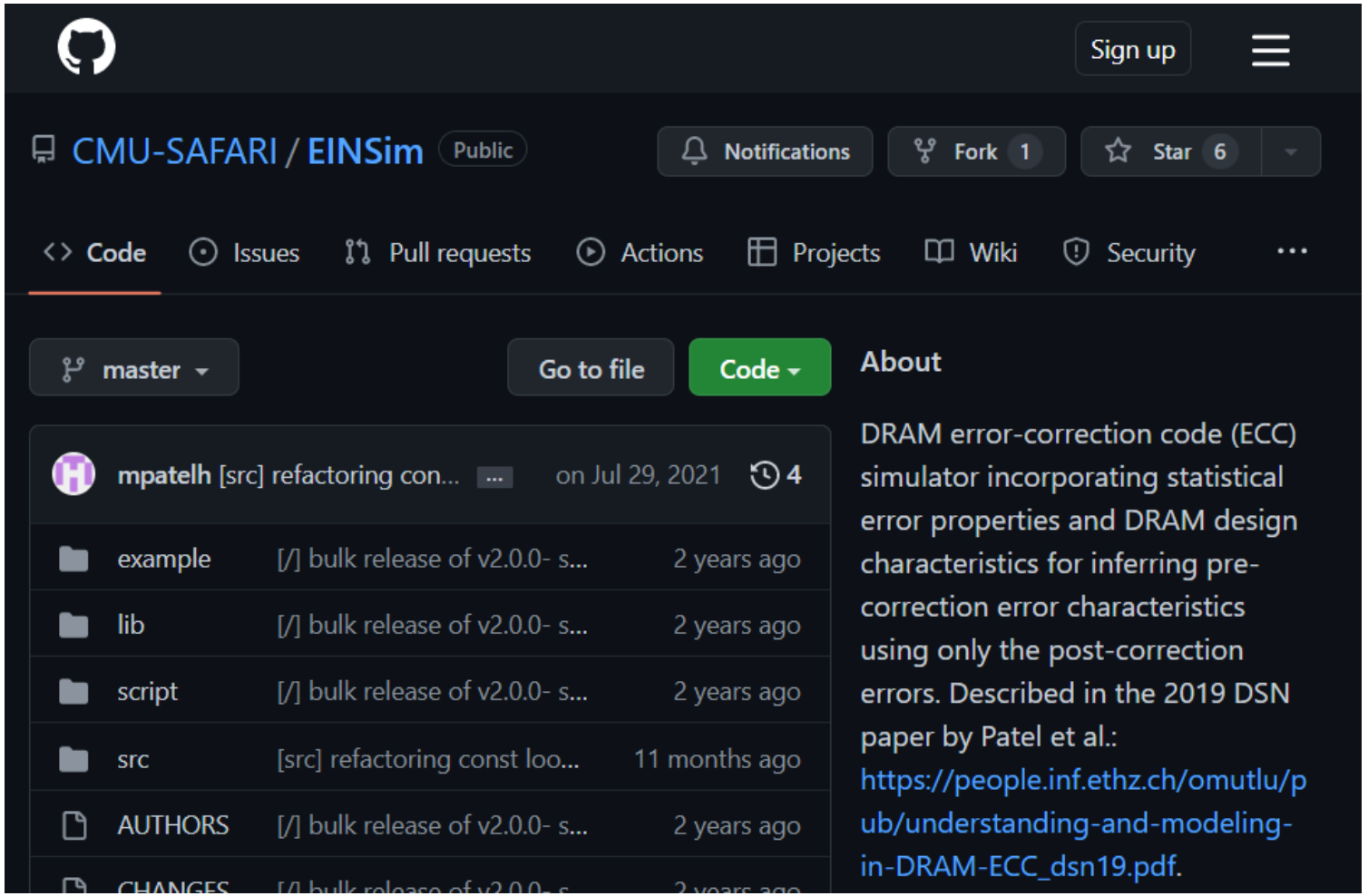
[†]*ETH Zürich* [‡]*Carnegie Mellon University*

Experimental characterization of DRAM errors is a powerful technique for understanding DRAM behavior and provides valuable insights for improving overall system performance, energy efficiency, and reliability. Unfortunately, recent DRAM technology scaling issues are forcing manufacturers to adopt on-die error-correction codes (ECC), which pose a significant challenge for DRAM error characterization studies by obfuscating raw error distributions using undocumented, proprietary, and opaque error-correction hardware. As we show in this work, errors observed in devices with on-die ECC no longer follow expected, well-studied distributions (e.g., lognormal retention times) but rather depend on the particular ECC scheme used.

physical DRAM error mechanisms (e.g., charge leakage, circuit interference). The resulting errors directly reflect the effects of the error mechanisms, providing researchers with insight into the physical properties that underlie DRAM operation (e.g., data-retention, circuit timings, data-pattern sensitivity). Researchers can then exploit these insights to develop new mechanisms that improve DRAM and overall system efficiency.

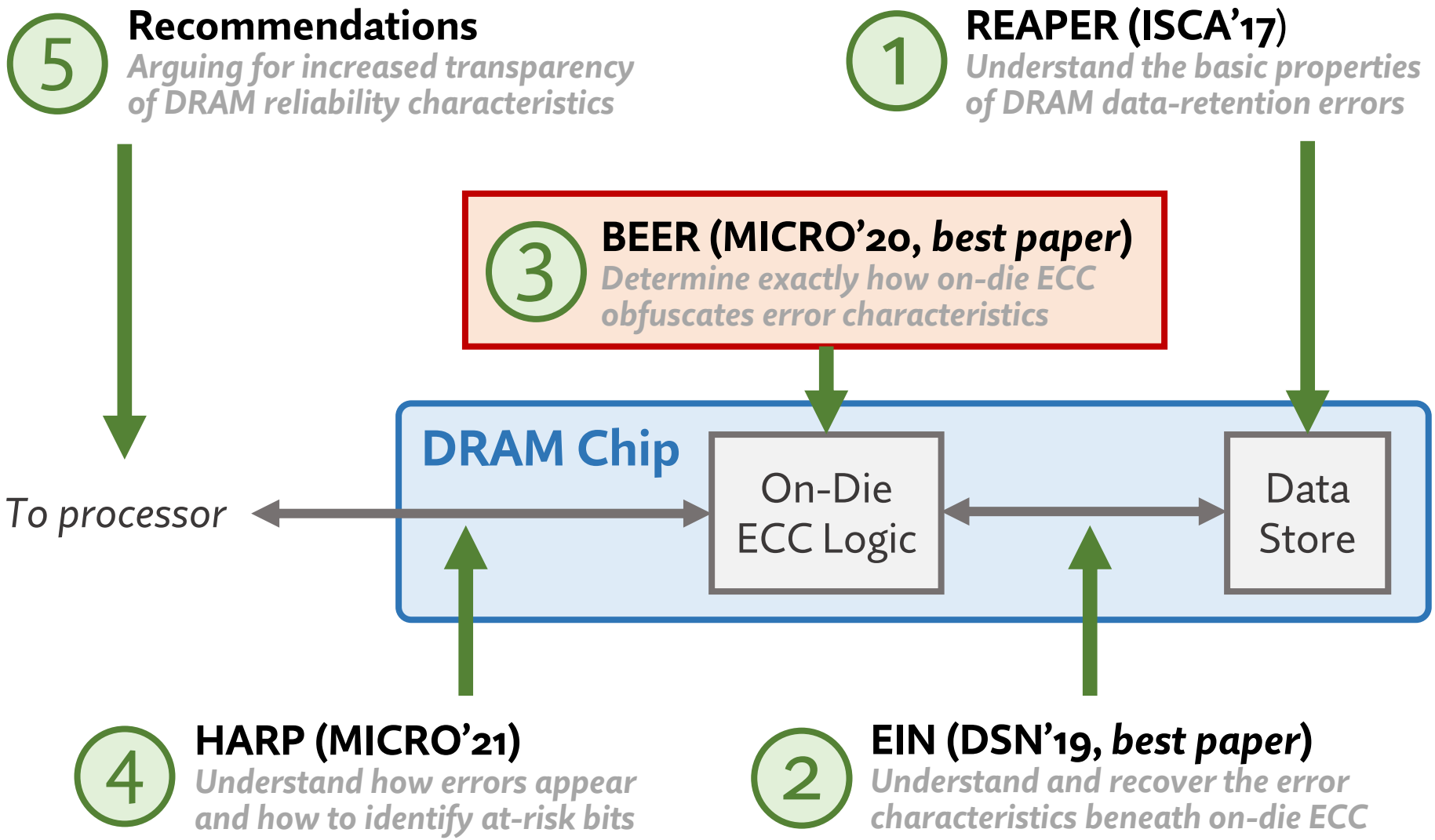
Unfortunately, continued DRAM technology scaling heralds grave reliability concerns going forward primarily due to increasing single-bit error rates that reduce manufacturing yield [28,37,49,73,82,85,86,92,93,106,114,115]. While manufacturers traditionally use redundant circuit elements (e.g., rows,

Minesh Patel, Jeremie S. Kim, Hasan Hassan, and Onur Mutlu,
"Understanding and Modeling On-Die Error Correction in Modern DRAM: An Experimental Study Using Real Devices"
Proceedings of the 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (***DSN***), Portland, OR, USA, June 2019.



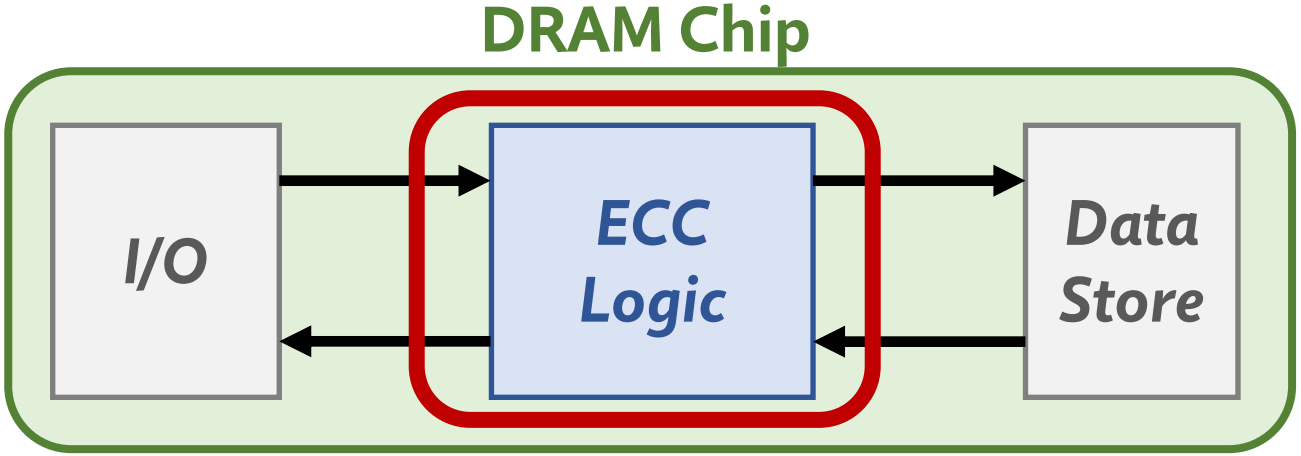
https://github.com/CMU-SAFARI/EINSim

Core Contributions



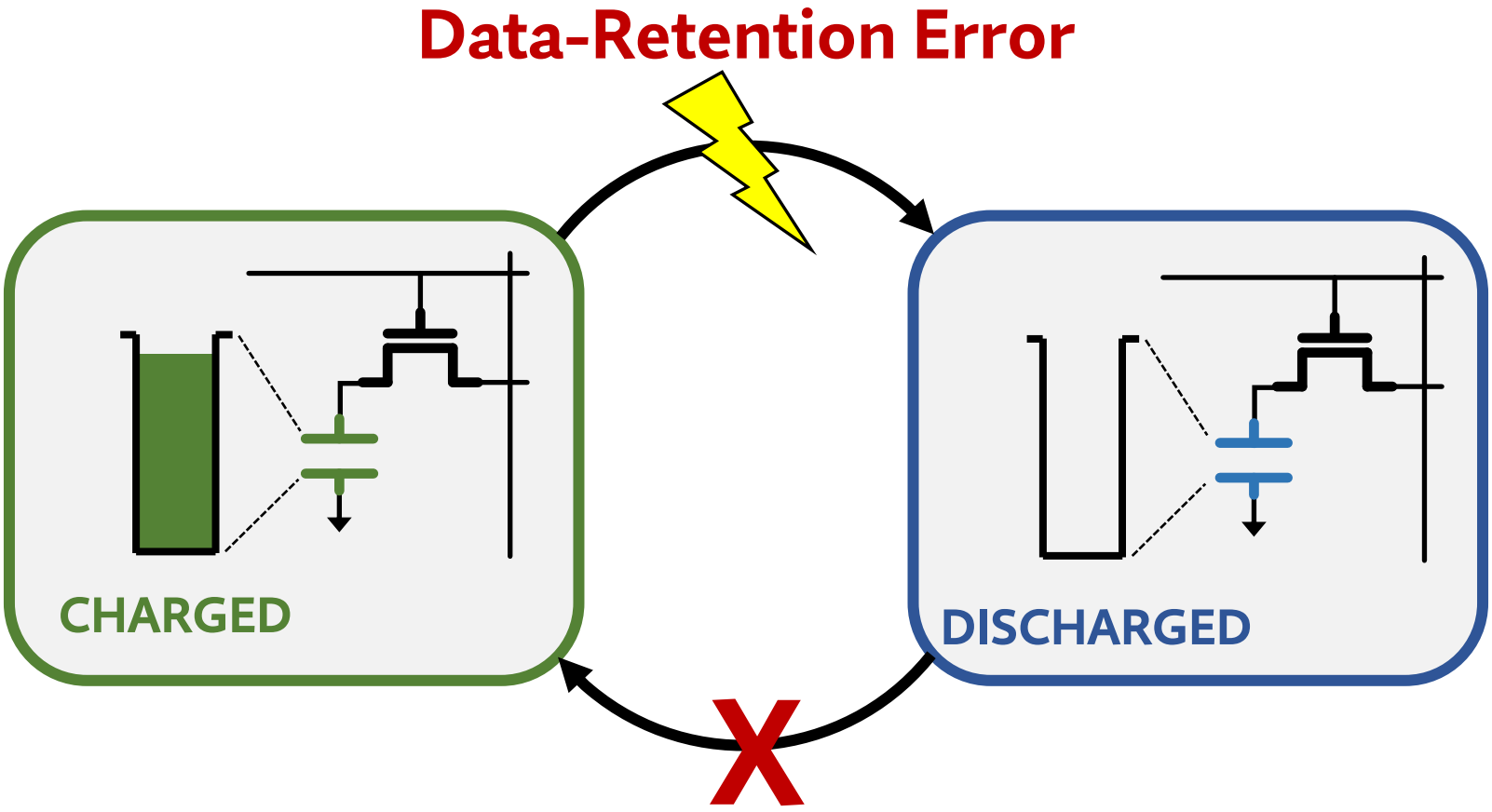
Our goal:

Determine **exactly how** on-die ECC obfuscates errors (i.e., its parity-check matrix)

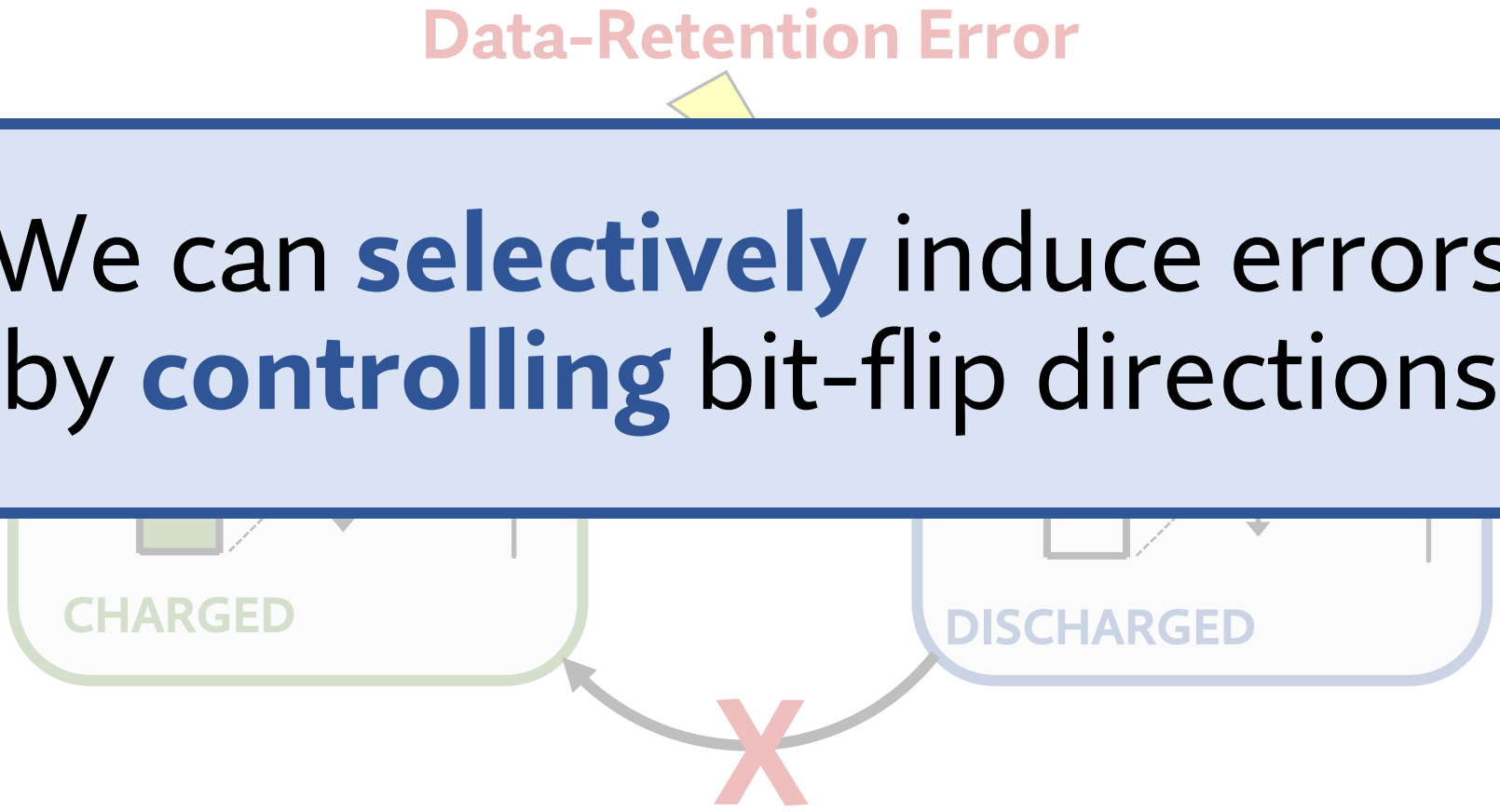


- **BEER:** Reveals **how** on-die ECC scrambles errors
- **BEEP:** Enables **inferring** raw bit error locations

Key idea: disabling DRAM refresh induces data-retention errors **only** in CHARGED cells

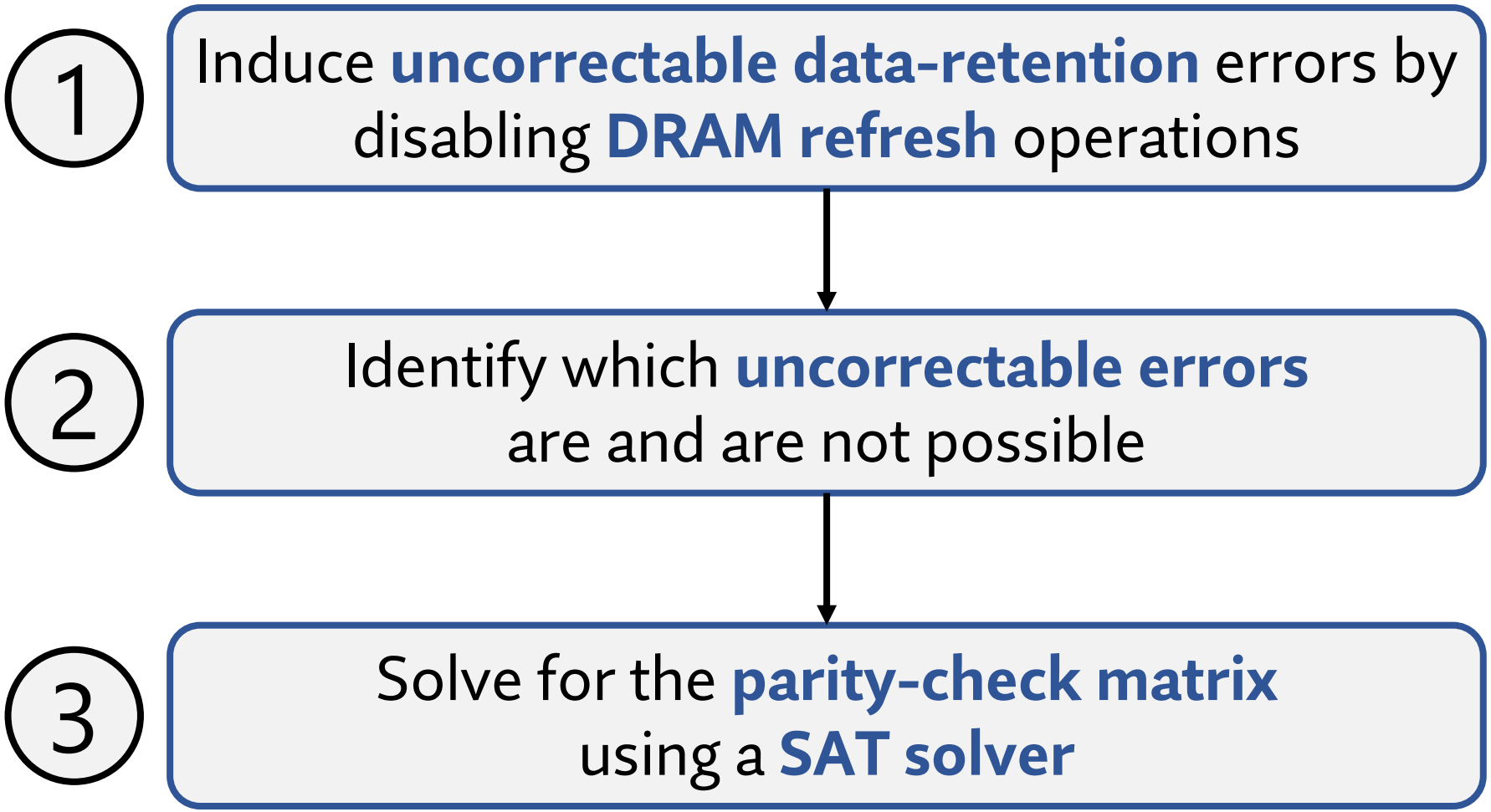


Key idea: disabling DRAM refresh induces data-retention errors **only** in CHARGED cells



We can **selectively** induce errors by **controlling** bit-flip directions

BEER Testing Methodology



Using BEER in Practice

- BEER determines the parity-check matrix **without**:
  (1) hardware support or tools
  (2) prior knowledge about on-die ECC
  (3) access to ECC metadata (e.g., syndromes)

- Open-source C++ tool on GitHub

  https://github.com/CMU-SAFARI/BEER

Experimental demonstration

80 LPDDR4 DRAM chips
(3 major manufacturers)

Two-Part Evaluation

Simulation of correctness and practicality

Over 100,000 representative ECC codes
of varying word lengths (4 – 247 bits)

1. Different manufacturers appear to use **different parity-check matrices**
2. Chips of the same model appear to use **identical** parity-check matrices

Two-Part Evaluation

1. BEER works for **all** simulated test cases
2. BEER is **practical** in both runtime and memory usage

Best Paper Award, MICRO 2020

Bit-Exact ECC Recovery (BEER): Determining DRAM On-Die ECC Functions by Exploiting DRAM Data Retention Characteristics

Minesh Patel[†] Jeremie S. Kim[‡†] Taha Shahroodi[†] Hasan Hassan[†] Onur Mutlu[†‡]

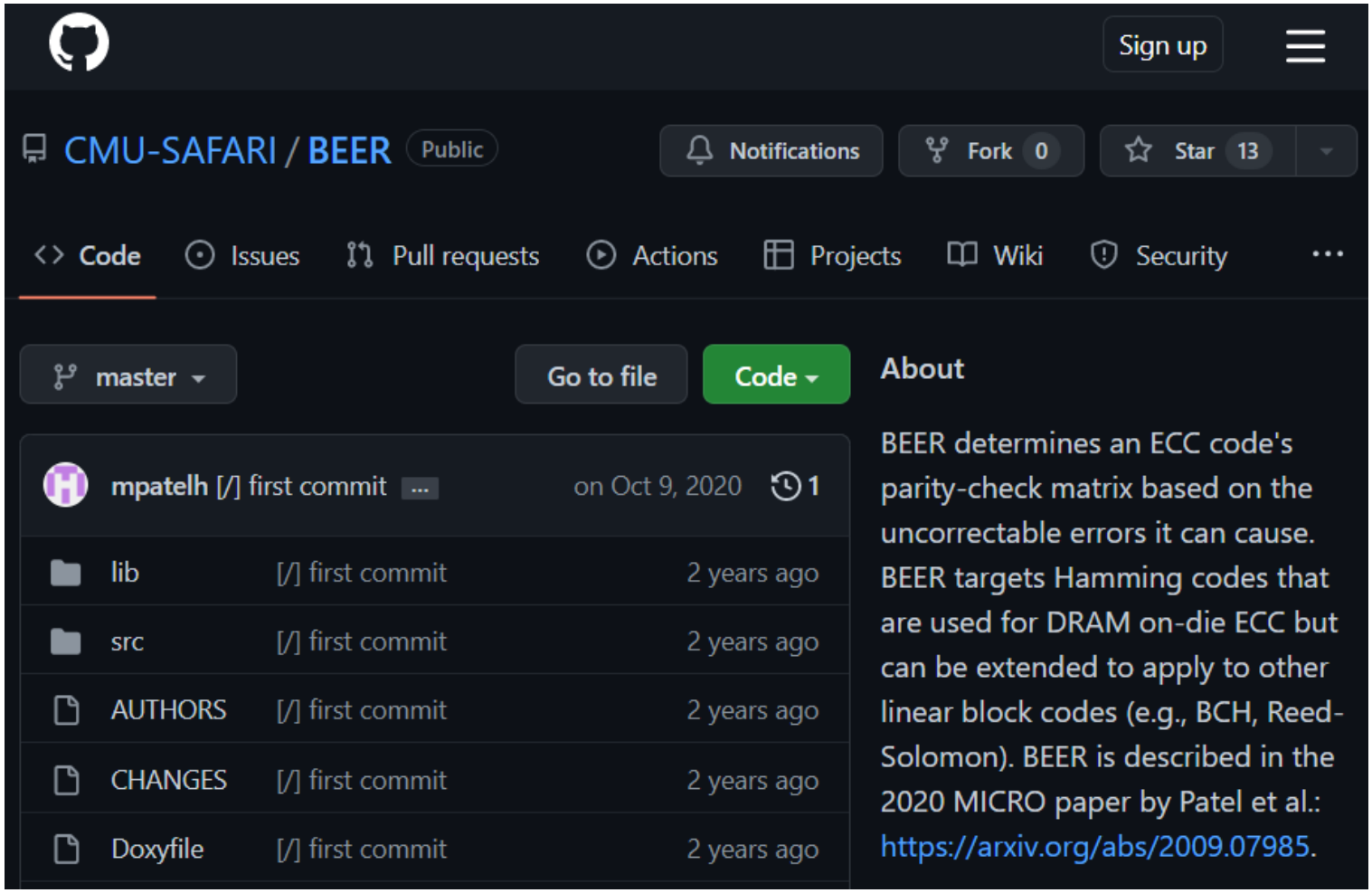
[†]*ETH Zürich* [‡]*Carnegie Mellon University*

Increasing single-cell DRAM error rates have pushed DRAM manufacturers to adopt on-die error-correction coding (ECC), which operates entirely within a DRAM chip to improve factory yield. The on-die ECC function and its effects on DRAM reliability are considered trade secrets, so only the manufacturer knows precisely how on-die ECC alters the externally-visible reliability characteristics. Consequently, on-die ECC obstructs third-party DRAM customers (e.g., test engineers, experimental researchers), who typically design, test, and validate systems based on these characteristics.

entirely within the DRAM chip [39, 76, 120, 129, 138]. On-die ECC is *completely invisible* outside of the DRAM chip, so ECC metadata (i.e., parity-check bits, error syndromes) that is used to correct errors is hidden from the rest of the system.

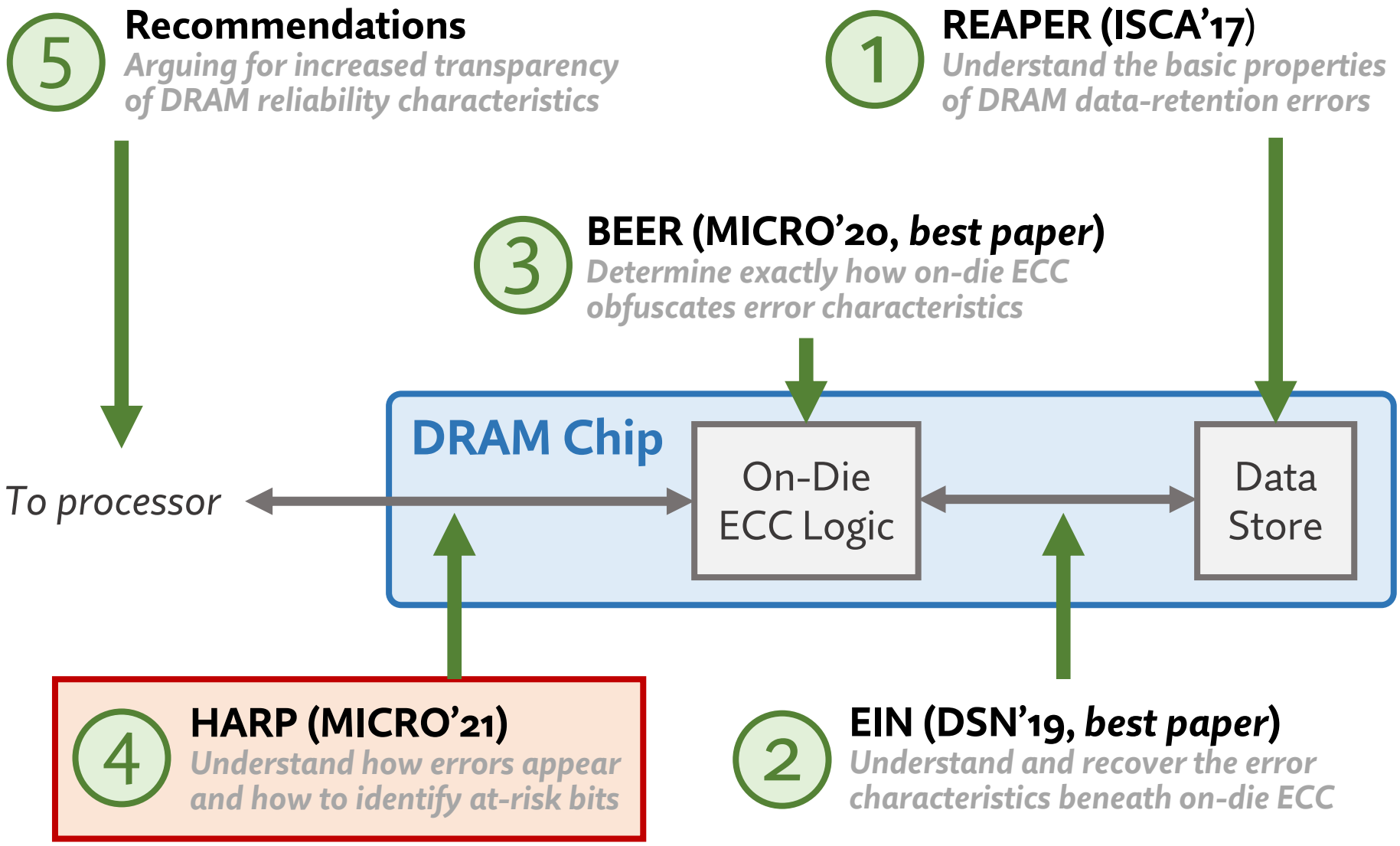
Prior works [60, 97, 98, 120, 129, 133, 138, 147] indicate that existing on-die ECC codes are 64- or 128-bit single-error correction (SEC) Hamming codes [44]. However, each DRAM manufacturer considers their on-die ECC mechanism's design and implementation to be highly proprietary and ensures not to reveal its details in any public documentation, including DRAM

Minesh Patel, Jeremie S. Kim, Taha Shahroodi, Hasan Hassan, and Onur Mutlu, **"Bit-Exact ECC Recovery (BEER): Determining DRAM On-Die ECC Functions by Exploiting DRAM Data Retention Characteristics"** *Proceedings of the 53rd International Symposium on Microarchitecture (**MICRO**)*, Virtual, October 2020.

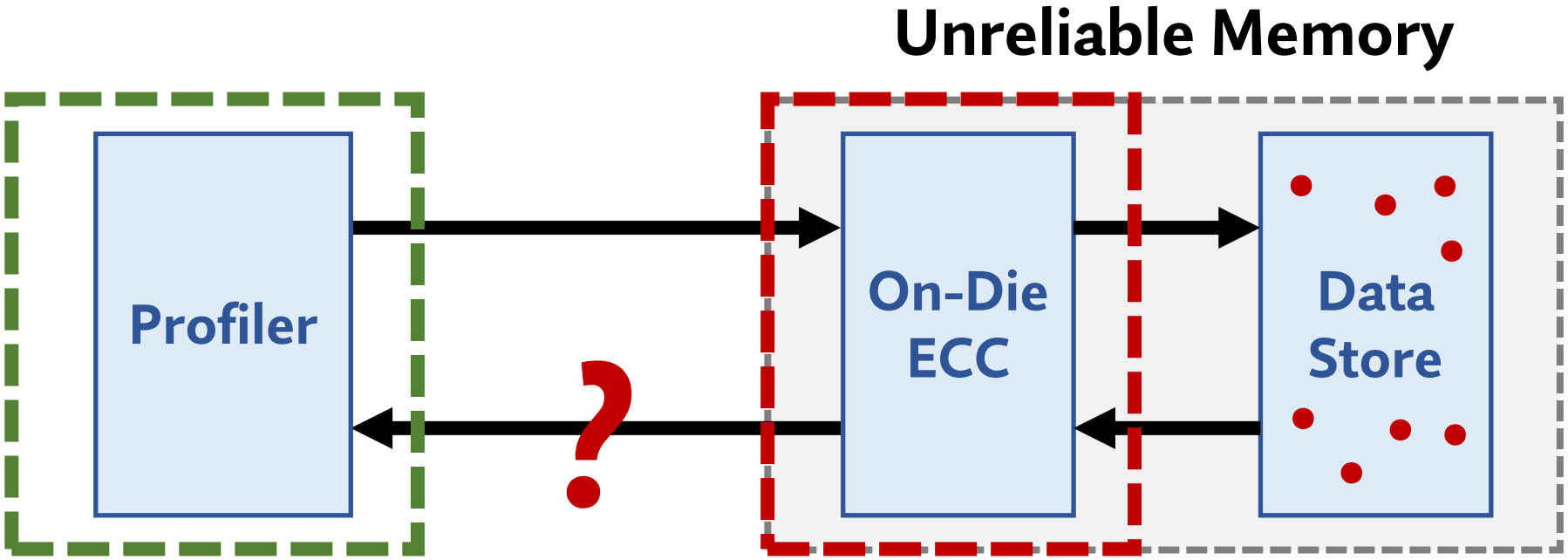


https://github.com/CMU-SAFARI/BEER

Core Contributions



Profiling a Memory Chip with On-Die ECC



Which bits are at risk of error?

On-die ECC changes how errors appear to the profiler

Goal: **understand** and **address** any challenges that on-die ECC introduces for error profiling

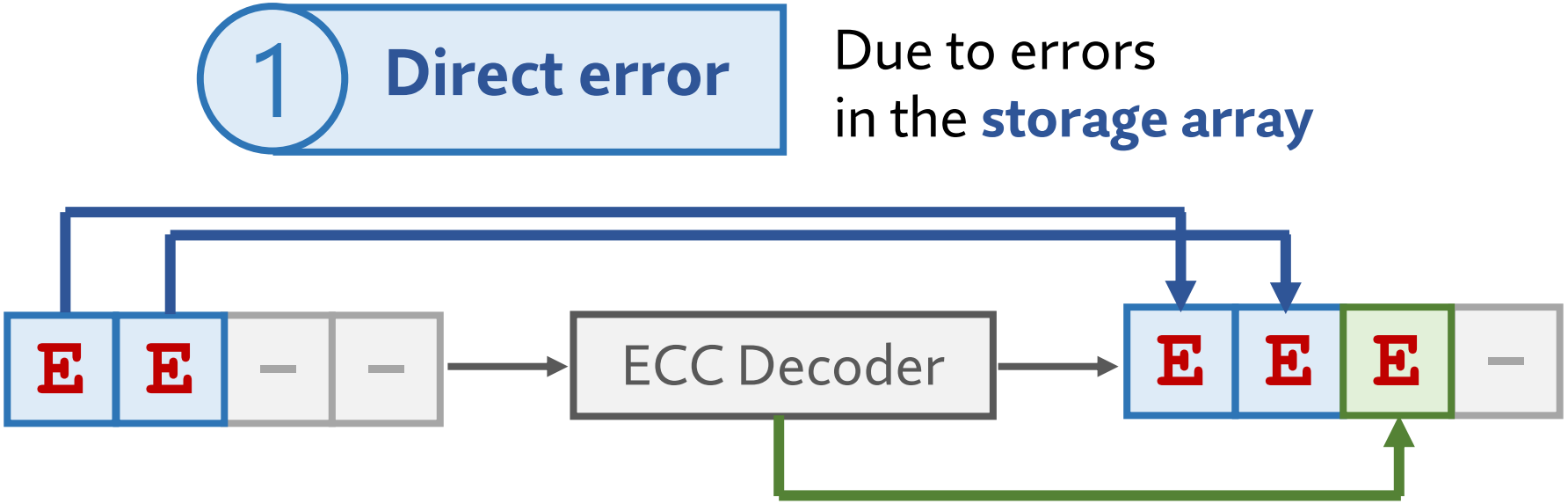
Challenges Introduced by On-Die ECC

1. **Exponentially increases** the total number of at-risk bits

2. Makes it **harder to identify** individual at-risk bits

3. **Interferes** with commonly-used data patterns for memory testing

Key Observation: Two Sources of Errors



Upper-bounded by the ECC algorithm

Key Observation: Two Sources of Errors

1 **Direct error**

Due to errors in the **storage array**

Key Idea: **decouple** profiling for **direct** and **indirect** errors

2 **Indirect error**

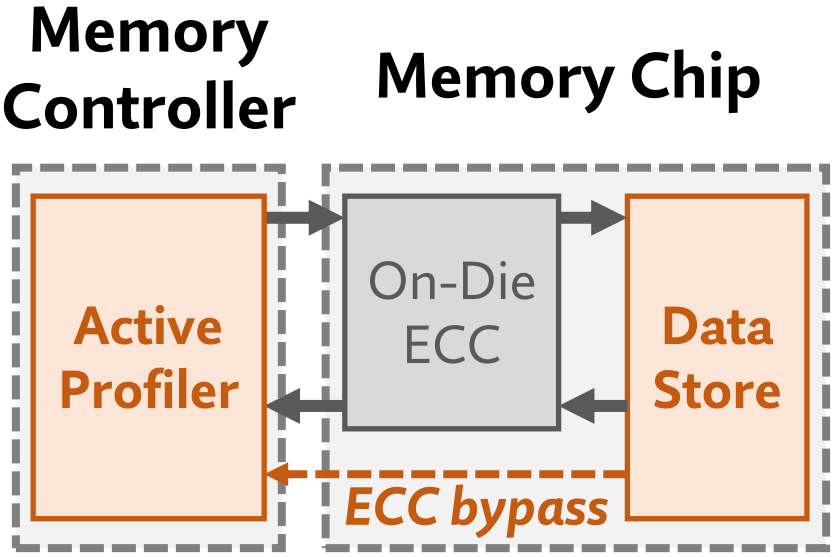
Artifact of the on-die ECC algorithm

Upper-bounded by the ECC algorithm

Hybrid Active-Reactive Profiling (HARP)

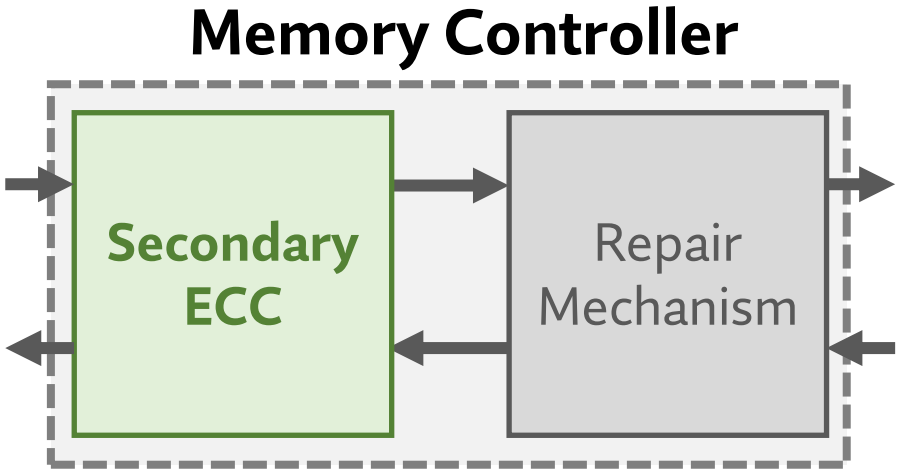
1 Active Profiling

Quickly identifies **all direct errors** with **existing** profiling techniques using an on-die ECC **bypass path**

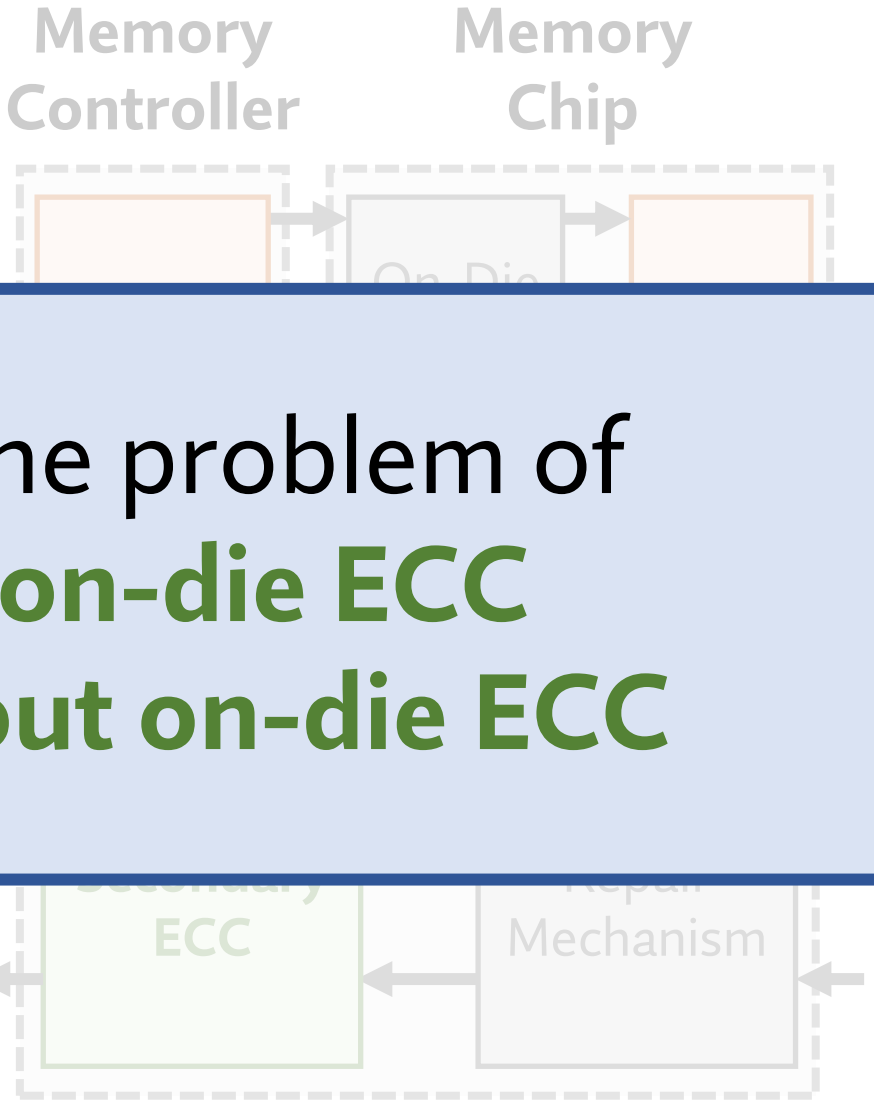


2 Reactive Profiling

Safely identifies **indirect errors** using **secondary ECC** at least as strong as on-die ECC



Hybrid Active-Reactive Profiling (HARP)



HARP **reduces** the problem of profiling **with on-die ECC** to profiling **without on-die ECC**

Evaluations

1. HARP improves **coverage** and **performance** relative to two state-of-the-art baseline profiling algorithms
   - E.g., **20.6-62.1% faster** to achieve 99th-percentile coverage for 2-5 raw-bit errors per on-die ECC word

2. HARP **outperforms** the best-performing baseline in a case study of mitigating data-retention errors
   - E.g., **3.7x faster** given a per-bit error probability of 0.75

We conclude that HARP **overcomes** all three profiling challenges

HARP: Practically and Effectively Identifying Uncorrectable Errors in Memory Chips That Use On-Die Error-Correcting Codes

Minesh Patel
ETH Zürich

Geraldo F. Oliveira
ETH Zürich

Onur Mutlu
ETH Zürich

ABSTRACT

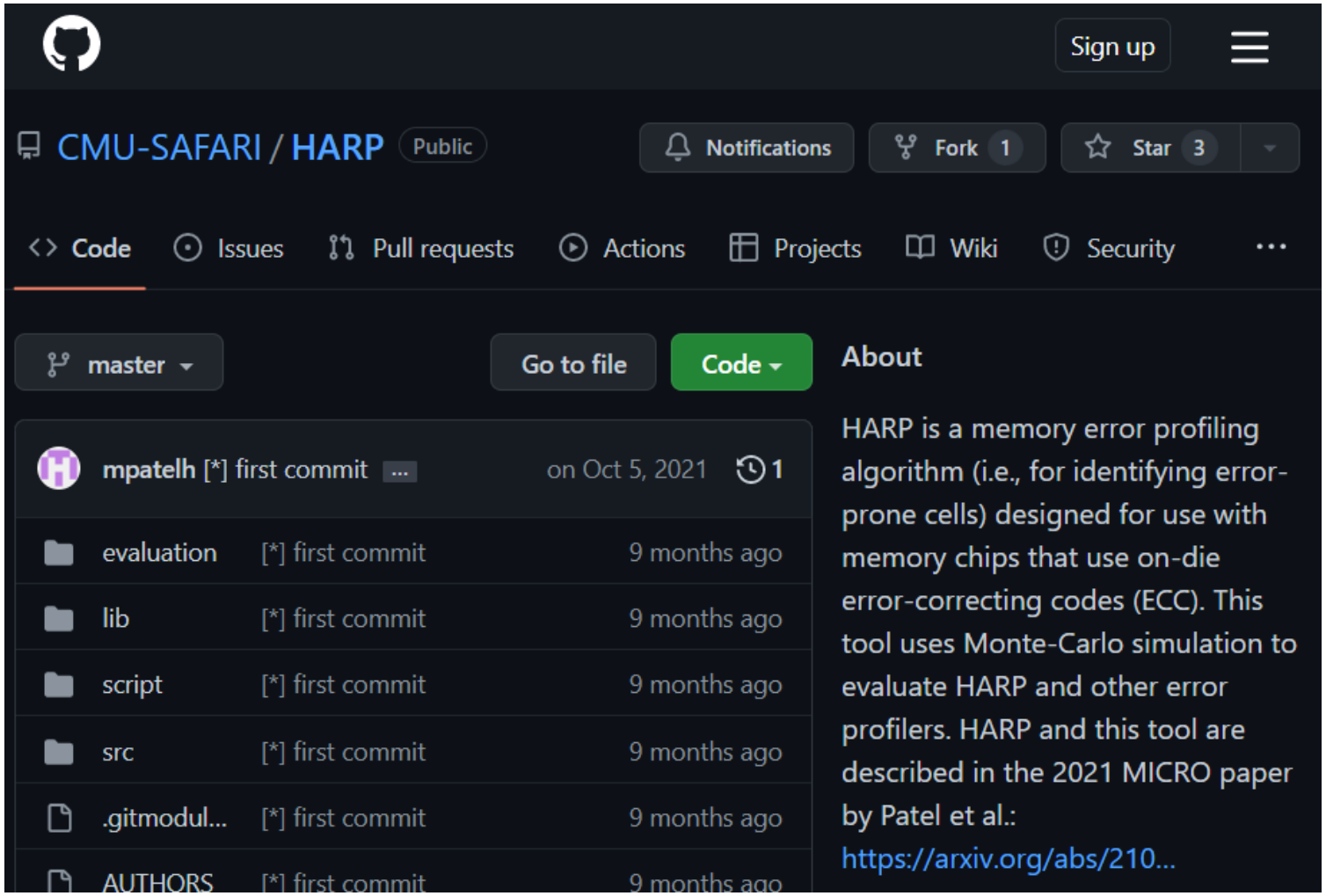
Aggressive storage density scaling in modern main memories causes increasing error rates that are addressed using error-mitigation techniques. State-of-the-art techniques for addressing high error rates identify and repair bits that are at risk of error from within the memory controller. Unfortunately, modern main memory chips internally use on-die error correcting codes (on-die ECC) that obfuscate the memory controller's view of errors, complicating the process of identifying at-risk bits (i.e., error profiling).

profiler impacts the system's overall bit error rate (BER) when using a repair mechanism to tolerate DRAM data-retention errors. We show that HARP identifies all errors faster than the best-performing baseline algorithm (e.g., by 3.7× for a raw per-bit error probability of 0.75). We conclude that HARP effectively overcomes the three error profiling challenges introduced by on-die ECC.

CCS CONCEPTS

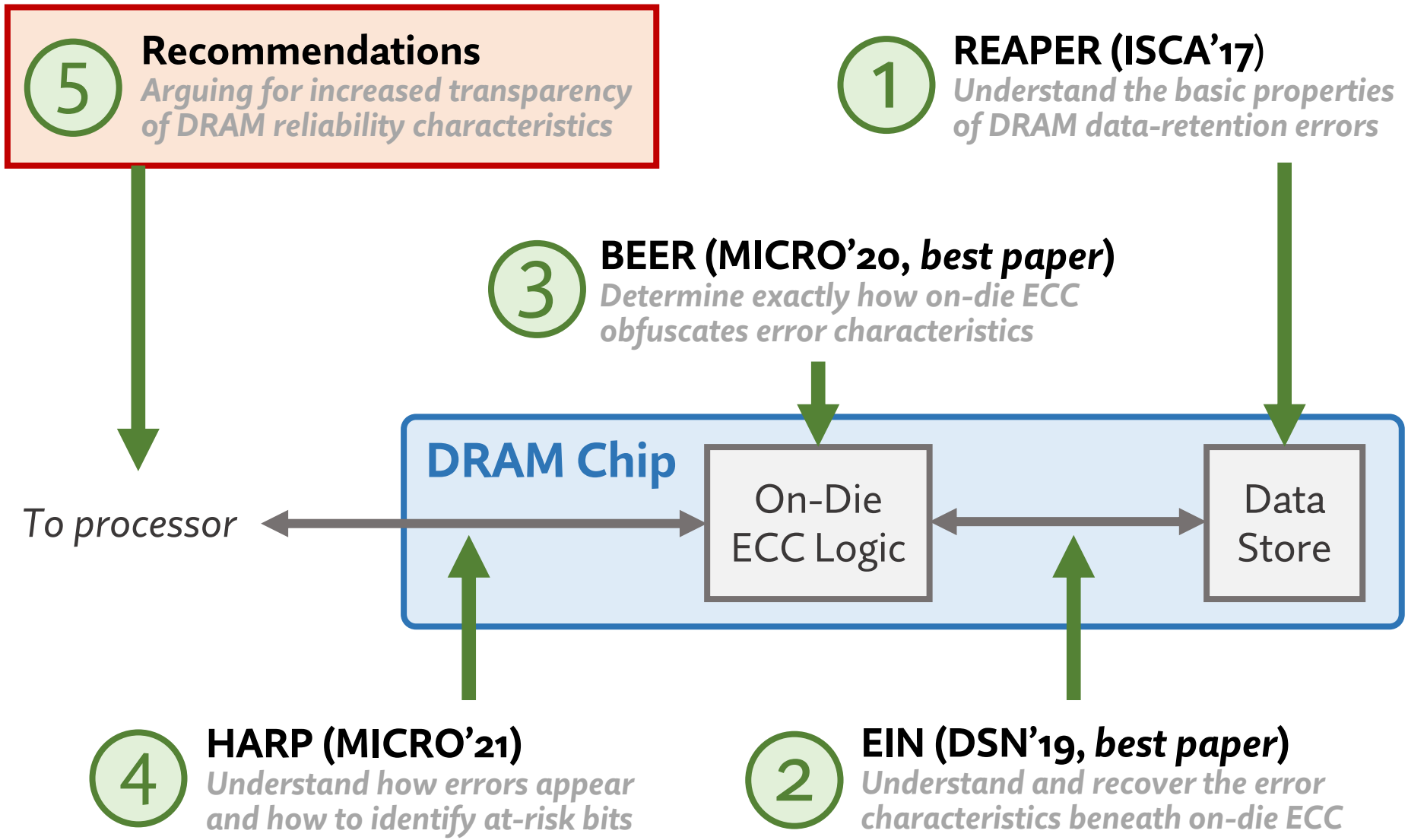
• **Computer systems organization** → **Dependable and fault-**

Minesh Patel, Geraldo F. de Oliveira Jr., and Onur Mutlu,
"HARP: Practically and Effectively Identifying Uncorrectable Errors in Memory Chips That Use On-Die Error-Correcting Codes"
Proceedings of the 54th International Symposium on Microarchitecture ***(MICRO)***, Virtual, October 2021.

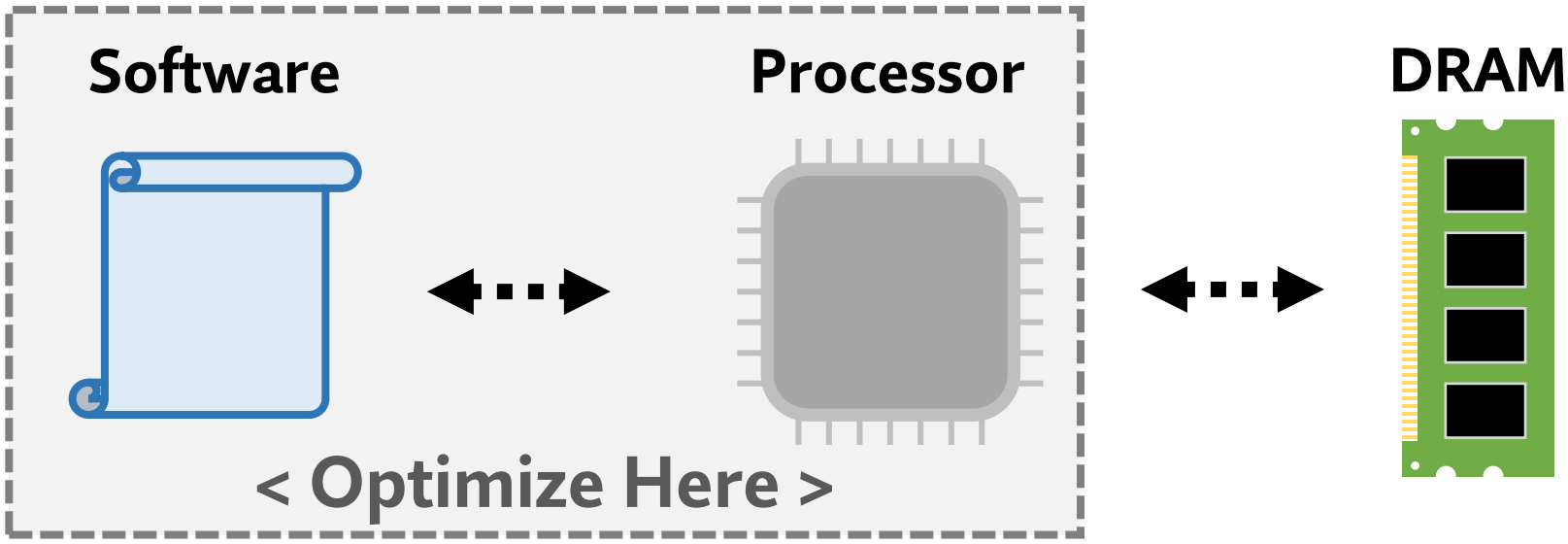


https://github.com/CMU-SAFARI/HARP

Core Contributions



Many Ways to Exploit Commodity DRAM



Reduce timing/voltage margins
e.g., Access and refresh timings

Use system-level error mitigations
e.g., ECC, redundancy, replication

Use security enhancements
e.g., RowHammer and Cold-Boot defenses

Cost
Security
Reliability
Performance
Energy/Power

Many Ways to Exploit Commodity DRAM

Unfortunately, **adopting** these proposals typically relies on **unavailable information** about **DRAM reliability characteristics**

(e.g., design characteristics, testing practices, error behavior)

e.g., ECC, redundancy, replication

Use security enhancements

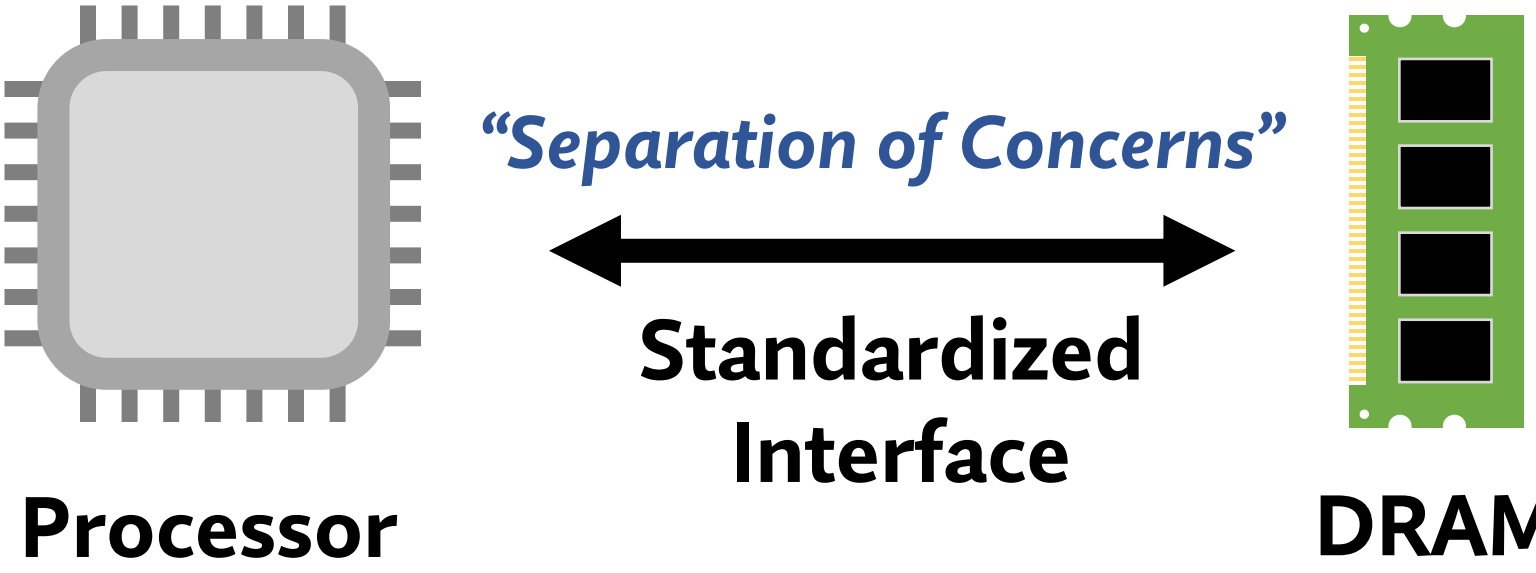
e.g., RowHammer and Cold-Boot defenses

Performance

Energy/Power

Source of the Problem

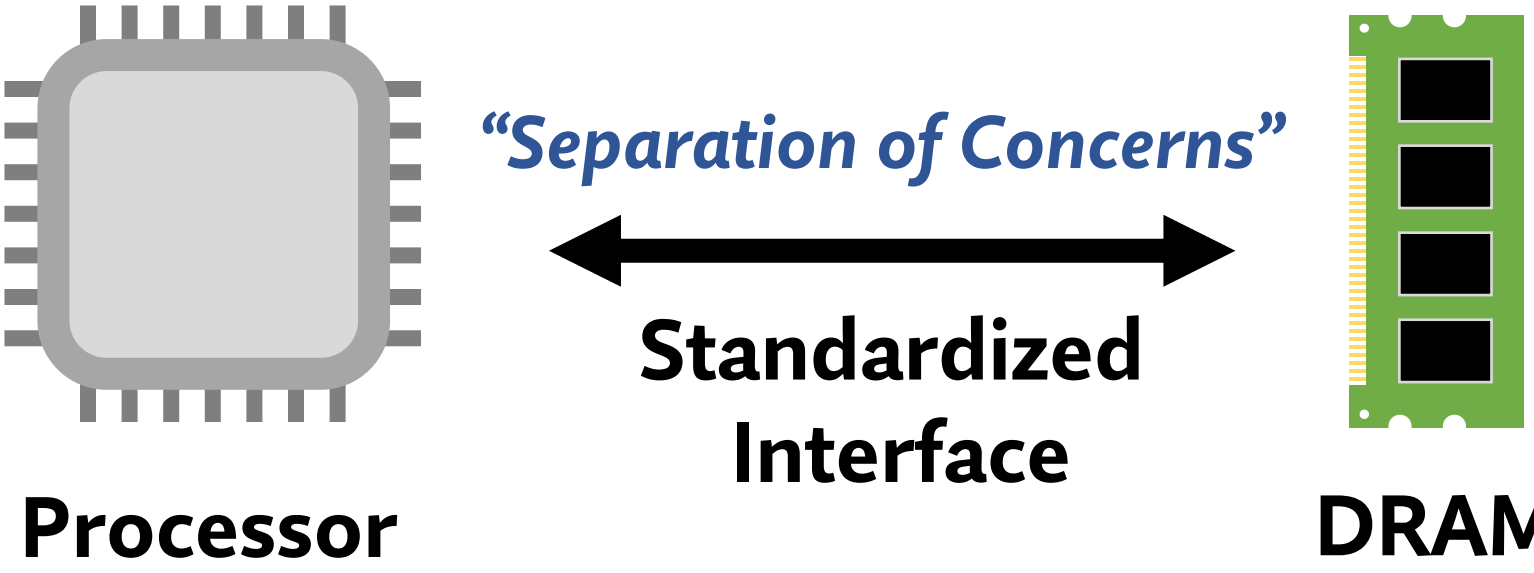
- Commodity DRAM specifications do not provide this information **by design**



- However, the **opportunity cost** of preserving this status quo is **increasing**
  - Technology scaling **exacerbates** refresh, RowHammer, etc.
  - Many old and new proposals for **leveraging this opportunity**

Source of the Problem

- Commodity DRAM specifications do not provide this information **by design**



Proposal: revisit **DRAM specifications** to improve **information transparency**

Two-Step Plan for Transparency

- **No change** to DRAM hardware or design
  - Just provide **information** so that system designers can make better **informed decisions** and **reason about** their designs

1. Short-term: convey basic information

- Whatever the manufacturers feel is practical to do so
- Possibly develop a crowdsourced database
- E.g., basic design properties that can be reverse-engineered

2. Long-term: rethink DRAM standards

- Incorporate transparency of reliability-related topics
- E.g., error models, testing guidelines

A Case for Transparent Reliability in DRAM Systems

Minesh Patel[†] Taha Shahroodi[‡†] Aditya Manglik[†] A. Giray Yağlıkçı[†]
Ataberk Olgun[†] Haocong Luo[†] Onur Mutlu[†]

[†]*ETH Zürich* [‡]*TU Delft*

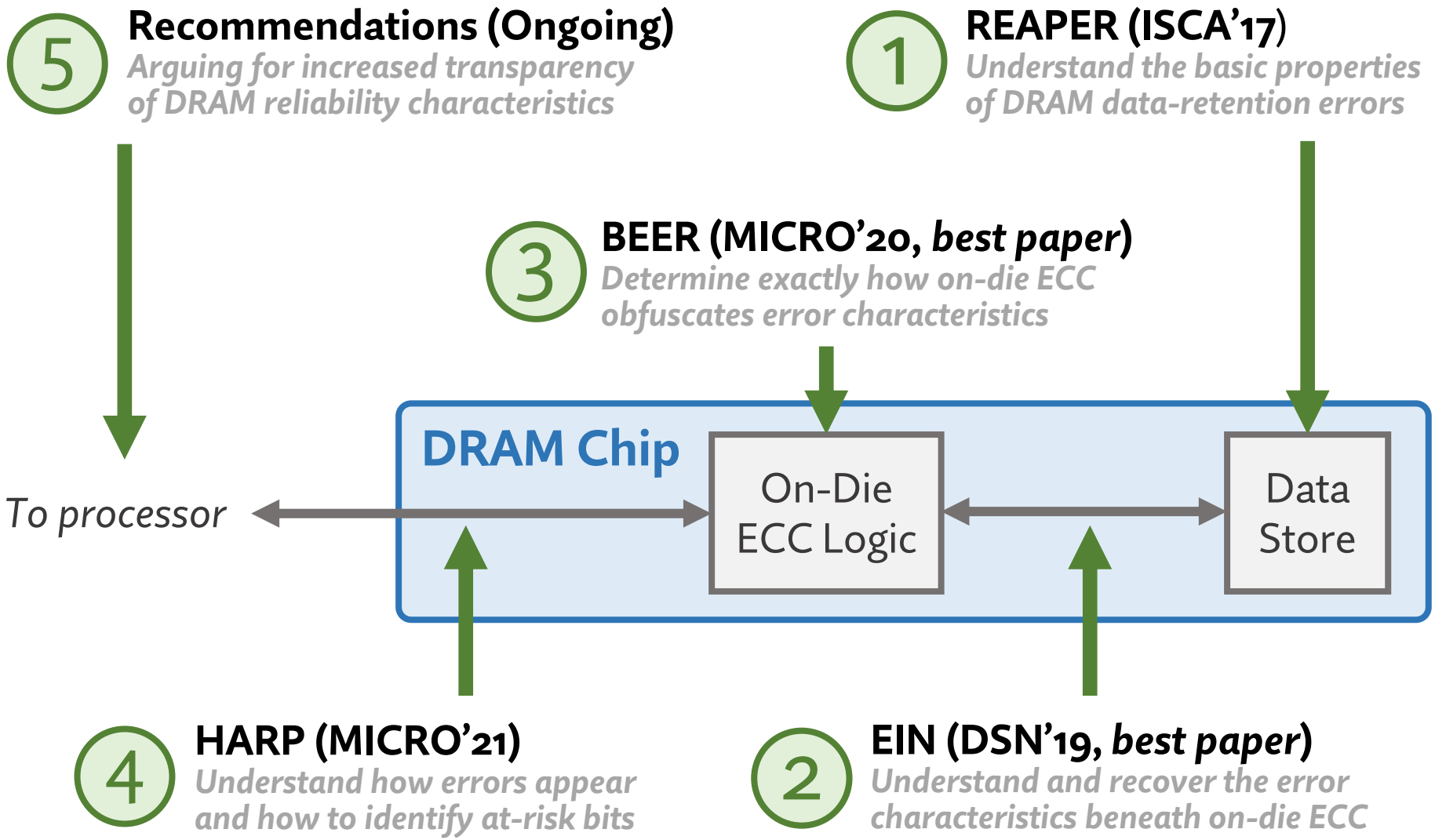
Mass-produced commodity DRAM is the preferred choice of main memory for a broad range of computing systems due to its favorable cost-per-bit. However, today's systems have diverse system-specific needs (e.g., performance, energy, reliability) that are difficult to address using one-size-fits-all general-purpose DRAM. Unfortunately, although system designers can theoretically adapt commodity DRAM chips to meet their particular design goals (e.g., by exploiting slack in access timings to improve performance, or implementing system-level RowHammer mitigations), we observe that designers today lack the necessary insight into commodity DRAM chips' reliability characteristics to implement these techniques in practice.

who purchase, test, and/or integrate commodity DRAM chips (e.g., cloud system designers, processor and system-on-a-chip (SoC) architects, memory module designers, test and validation engineers) are free to focus on the particular challenges of the systems they work on instead of dealing with the nuances of building low-cost, high-performance DRAM.

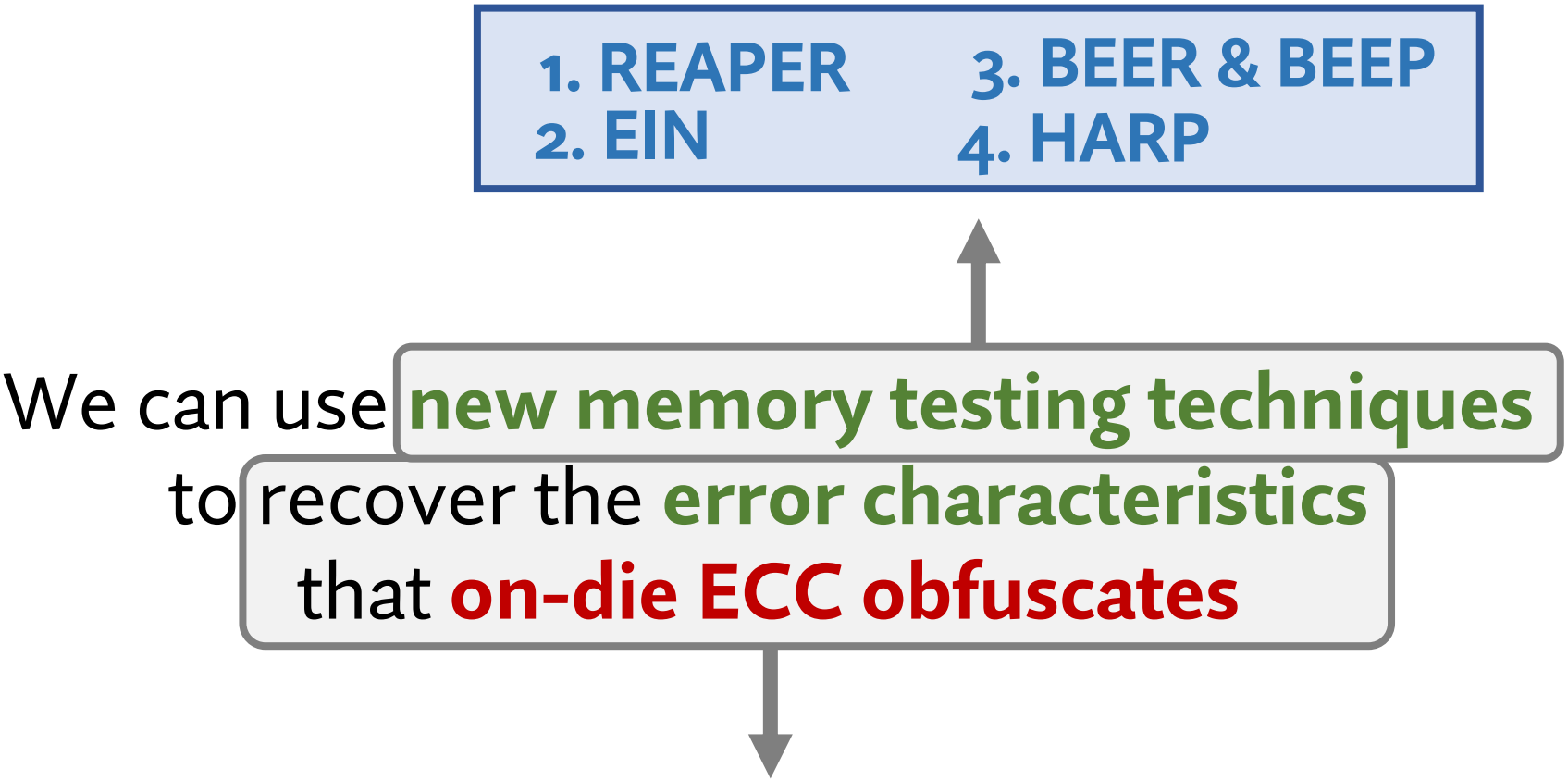
To ensure that system designers can integrate commodity DRAM chips from any manufacturer, the DRAM interface and operating characteristics have long been standardized by the JEDEC consortium [8]. JEDEC maintains a limited set of *DRAM standards* for commodity DRAM chips with different target applications, e.g., general-purpose DDR*n* [9–11], bandwidth-

Minesh Patel, Taha Shahroodi, Aditya Manglik, A. Giray Yaglikci, Ataberk Olgun, Haocong Luo, Onur Mutlu,
"A Case for Transparent Reliability in DRAM Systems"
arXiv, April 2022.

Core Contributions



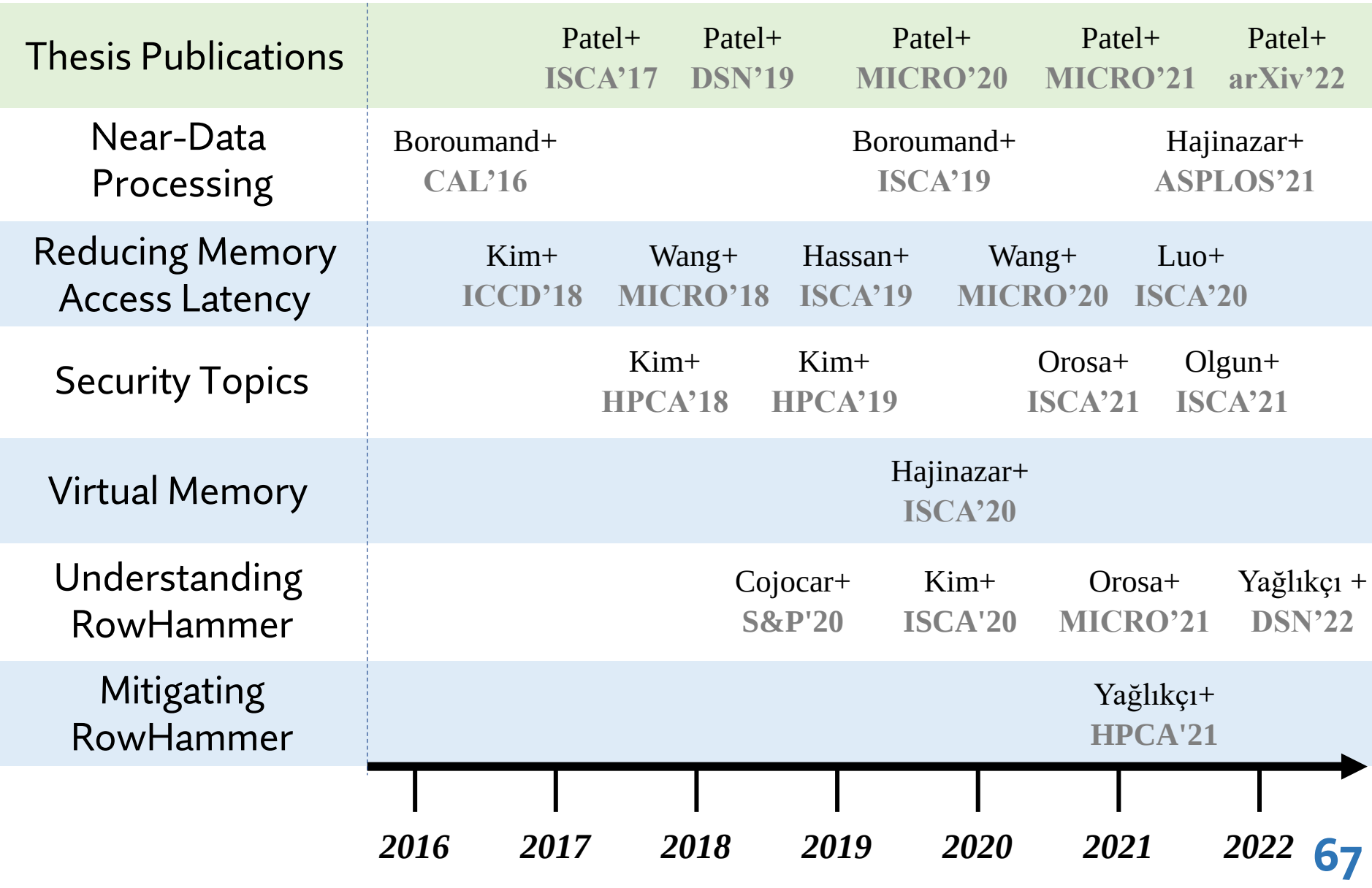
Thesis Statement



Future Research Directions

- **Extending the techniques** that we propose
  - Other ECC types and error mechanisms
  - Emerging ECC architectures and memory technologies
- **Using the information** that our techniques reveal
  - Improved system-level error mitigation mechanisms
  - Better diagnostic techniques for errors in the field
- **Devising alternatives** to on-die ECC
  - Different on-die ECC architectures
  - Cooperative on-die ECC and secondary error mitigation
- **Improving transparency** of DRAM reliability

Other Significant Works



2022 William Carter Award Recognition and Ceremony

Minesh Patel

DSN'22, Baltimore, MD

28 June 2022

List of Publications

1. Minesh Patel, Jeremie S. Kim, and Onur Mutlu, "**The Reach Profiler (REAPER): Enabling the Mitigation of DRAM Retention Failures via Profiling at Aggressive Conditions**," *Proceedings of the 44th International Symposium on Computer Architecture (ISCA)*, Toronto, Canada, June 2017.

2. Minesh Patel, Jeremie S. Kim, Hasan Hassan, and Onur Mutlu, "**Understanding and Modeling On-Die Error Correction in Modern DRAM: An Experimental Study Using Real Devices**," *Proceedings of the 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, Portland, OR, USA, June 2019. Best paper award.

3. Minesh Patel, Jeremie S. Kim, Taha Shahroodi, Hasan Hassan, and Onur Mutlu, "**Bit-Exact ECC Recovery (BEER): Determining DRAM On-Die ECC Functions by Exploiting DRAM Data Retention Characteristics**," *Proceedings of the 53rd International Symposium on Microarchitecture (MICRO)*, Virtual, October 2020. Best paper award.

4. Minesh Patel, Geraldo F. Oliveira, and Onur Mutlu, "**HARP: Practically and Effectively Identifying Uncorrectable Errors in Main Memory Chips That Use On-Die ECC**," *To appear in the Proceedings of the 54th International Symposium on Microarchitecture (MICRO)*, Virtual, October 2021.

5. Minesh Patel, A. Giray Yaglikci, Aditya Manglik, Taha Shahroodi, and Onur Mutlu, "**A Case for Transparent Reliability in DRAM Systems**," *arXiv:2204.10378*, April 2022.