An Evaluation Platform for Forensic Memory Acquisition Software

By

Stefan Voemel and Johannes Stuttgen

Presented At

The Digital Forensic Research Conference

DFRWS 2013 USA Monterey, CA (Aug 4th - 7th)

Motivation

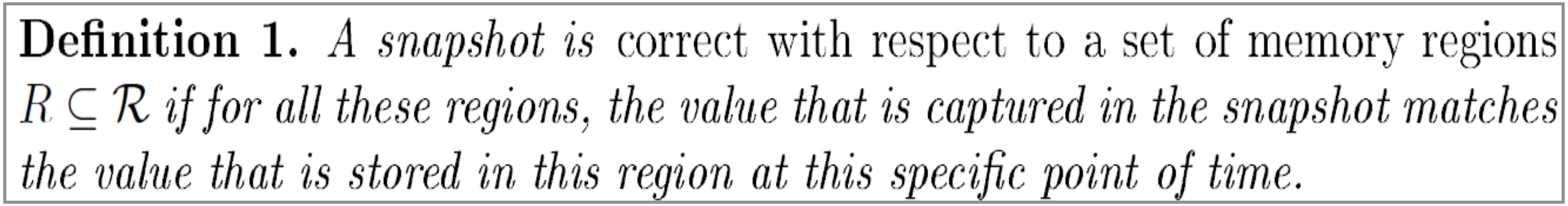
- Research in memory forensics has mostly focused on *analysis*-related aspects to date
- The respective base snapshot is frequently assumed to be „sound“ or „reliable“
- But what factors actually affect its „soundness“?
  - ➢ Once determined, how can we measure those factors?
  - ➢ To what degree do current acquisition approaches satisfy those factors?
  - ➢ In this talk: Methods for evaluating *software-based* imaging solutions

Acquisition Criteria

- Criteria for “Sound” Memory Imaging
  - Several criteria have been early identified by different authors
    - Works are mostly descriptive though and primarily illustrate weaknesses of existing technologies
    - More formal definition by Vömel and Freiling (2012)
  - Theory: The quality of a forensic memory snapshot is determined by its degree of *correctness*, *atomicity*, and *integrity*

Acquisition Criteria

- Correctness



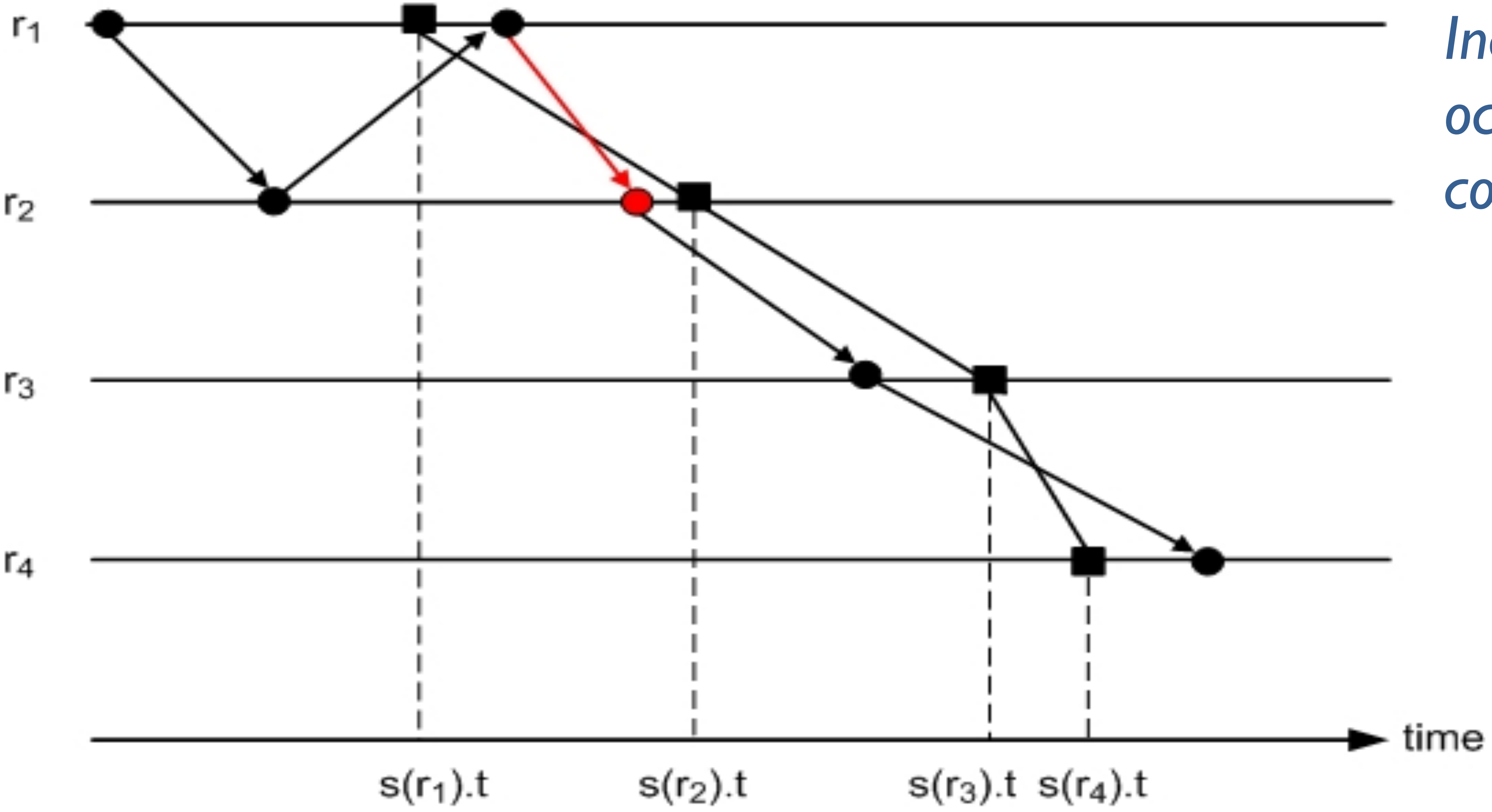
Definition 1. *A snapshot is* correct with respect to a set of memory regions $R \subseteq \mathcal{R}$ *if for all these regions, the value that is captured in the snapshot matches the value that is stored in this region at this specific point of time.*

- correctness basically means that the snapshot only contains “true” values
- trivial but necessary requirement
  - for instance, malicious software may try to impede or manipulate the acquisition process
  - errors in imaging applications may lead to incorrect acquisition results

Acquisition Criteria

- Atomicity

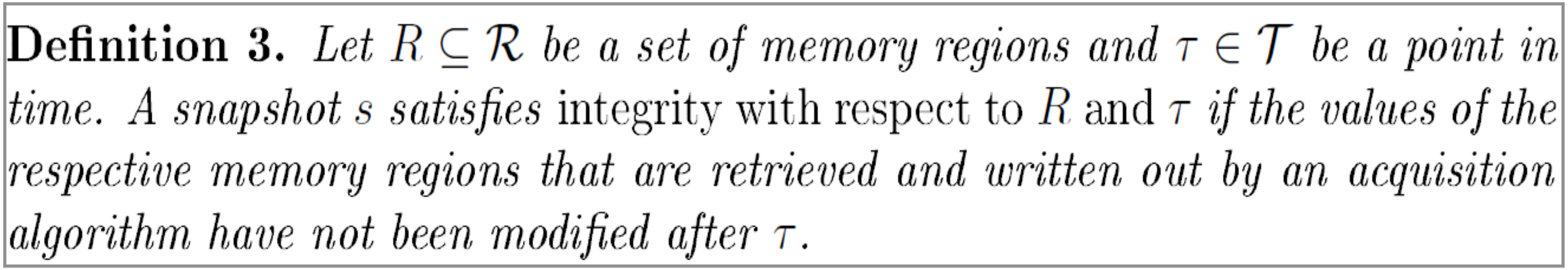
Definition 2. *A snapshot is* atomic with respect to $\mathcal{R}$ *if the cut through the corresponding space-time diagram is consistent.*



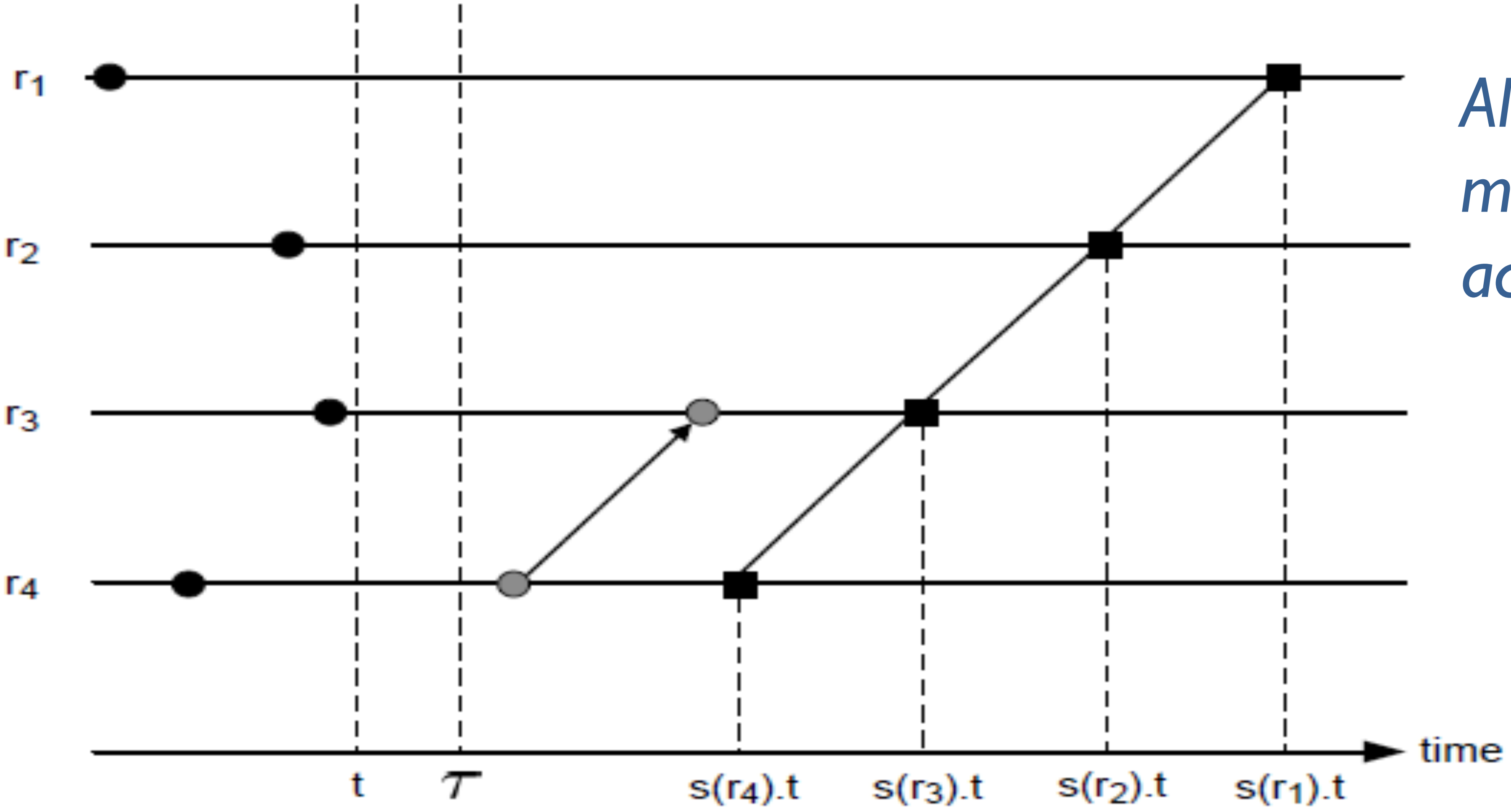
Inconsistencies may occur due to concurrent activity

Acquisition Criteria

- Integrity



Definition 3. *Let $R \subseteq \mathcal{R}$ be a set of memory regions and $\tau \in \mathcal{T}$ be a point in time. A snapshot s satisfies* integrity with respect to R and τ *if the values of the respective memory regions that are retrieved and written out by an acquisition algorithm have not been modified after τ.*



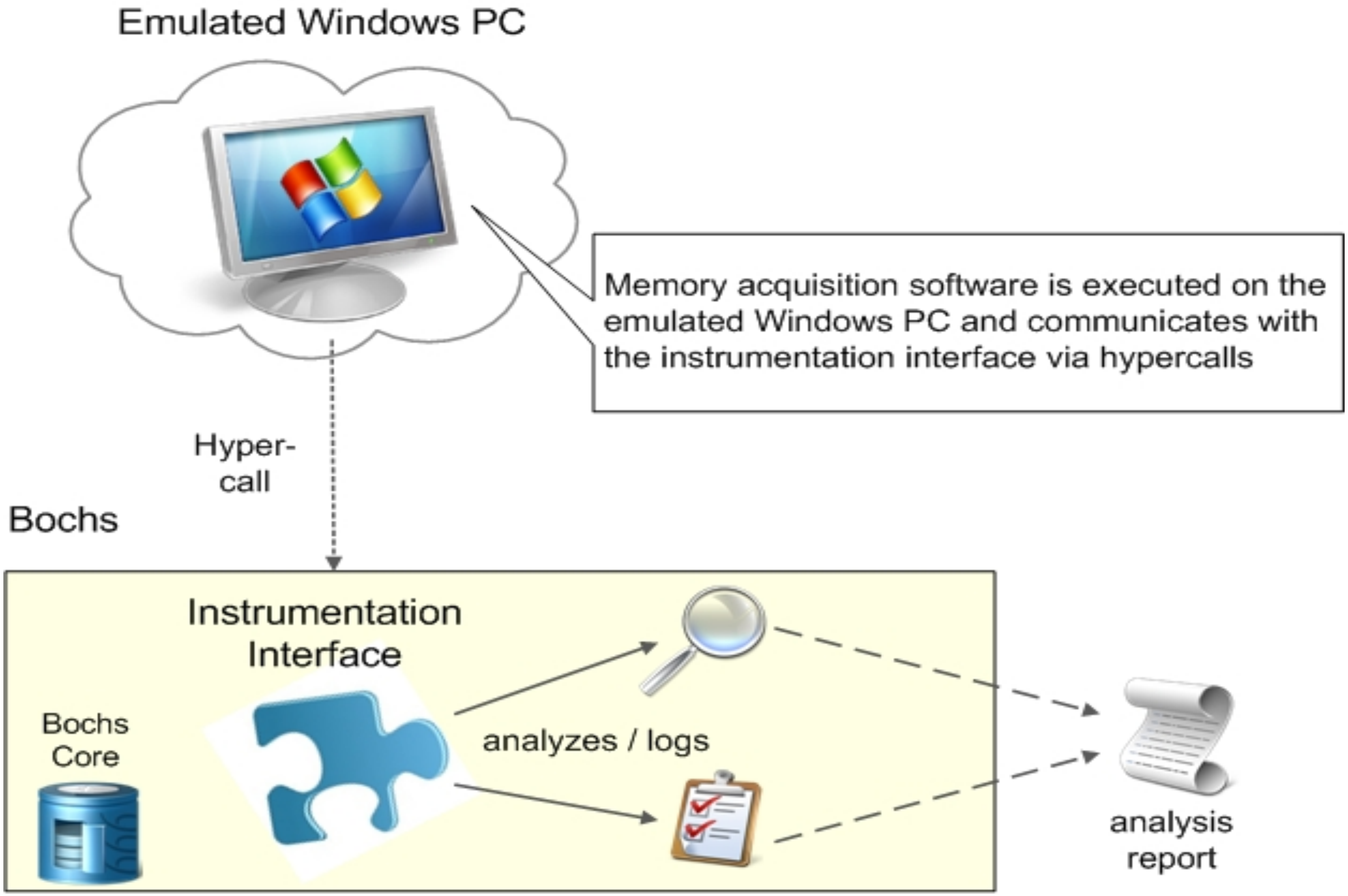
Allows observing the state of memory over the time of the acquisition process

Methodology

- Evaluation Methodology
  - We have developed an evaluation platform to determine the degree of correctness, atomicity, and integrity for Windows-based software imagers
    - Platform is based on a heavily customized version of the *Bochs* x86 PC emulator
    - White-box testing methodology
      - acquisition utilities need to be patched
      - important events (e.g., start of a page imaging operation) are communicated to the platform via a number of *hypercalls*

Methodology

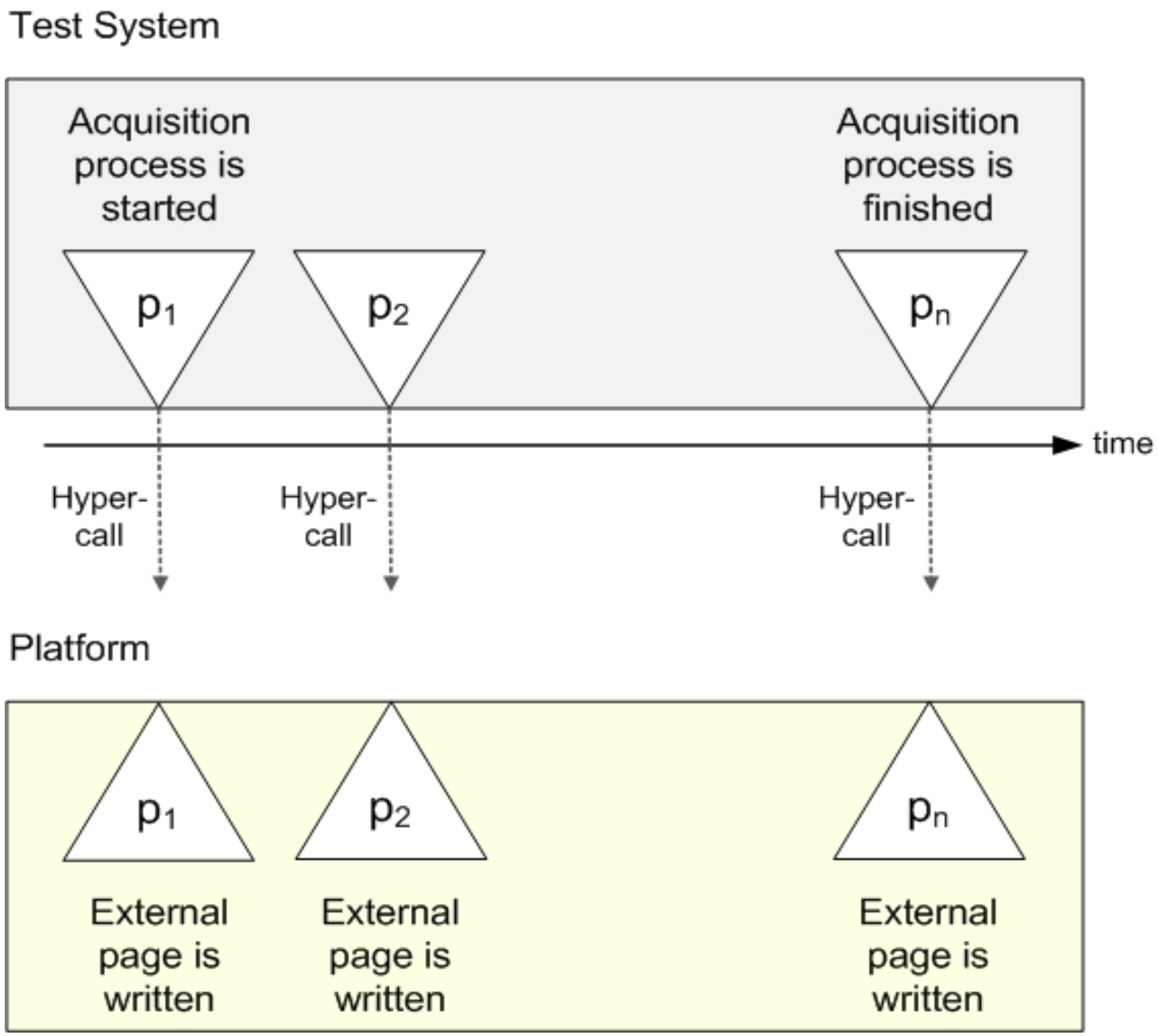
- Overview of the Platform Architecture



Memory acquisition software is executed on the emulated Windows PC and communicates with the instrumentation interface via hypercalls

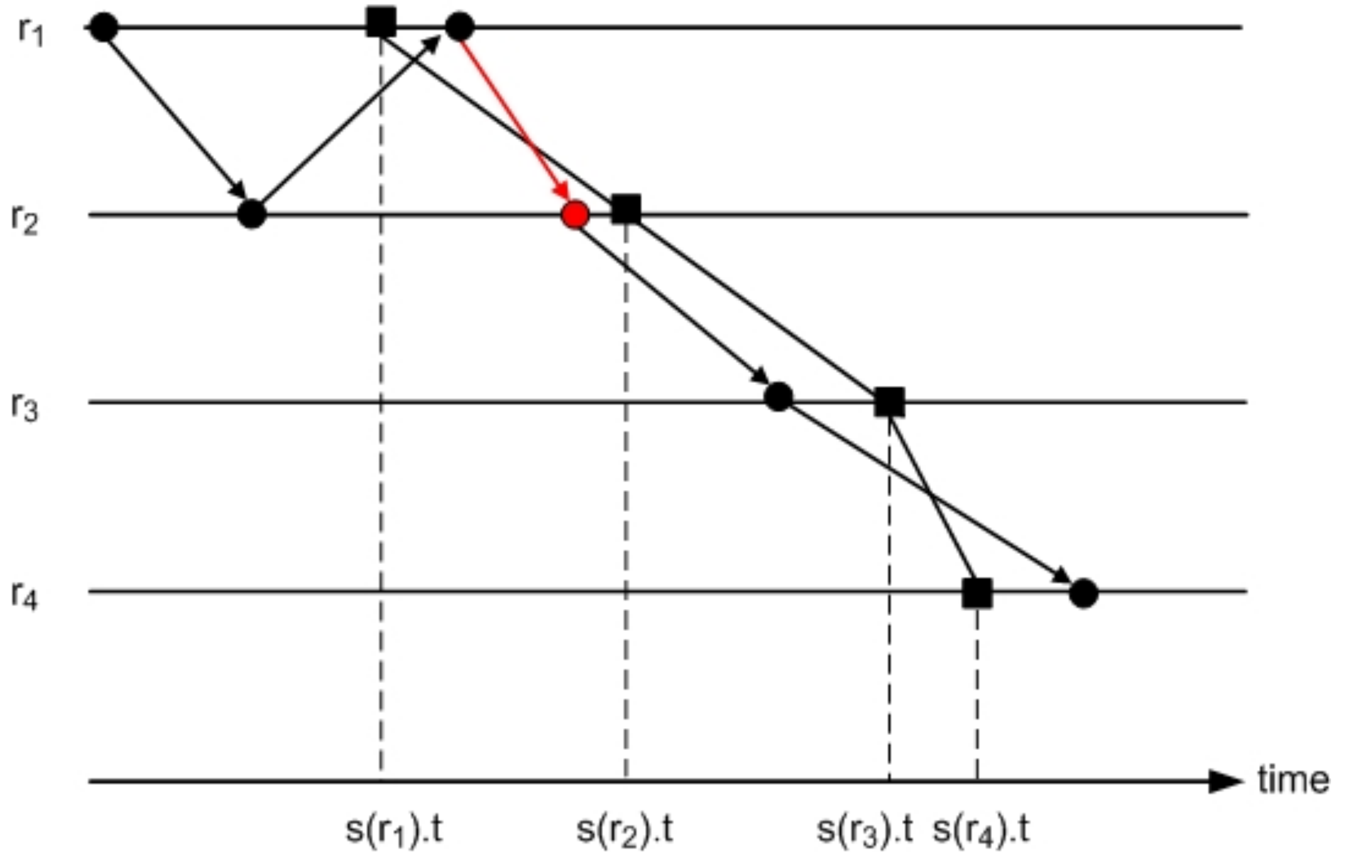
Methodology

- Measuring Correctness
  - Idea: Create an external memory snapshot in parallel to the acquisition phase
    - match the external snapshot with the image of the acquisition program to identify possible differences
    - permits verifying the size and contents of the created memory image



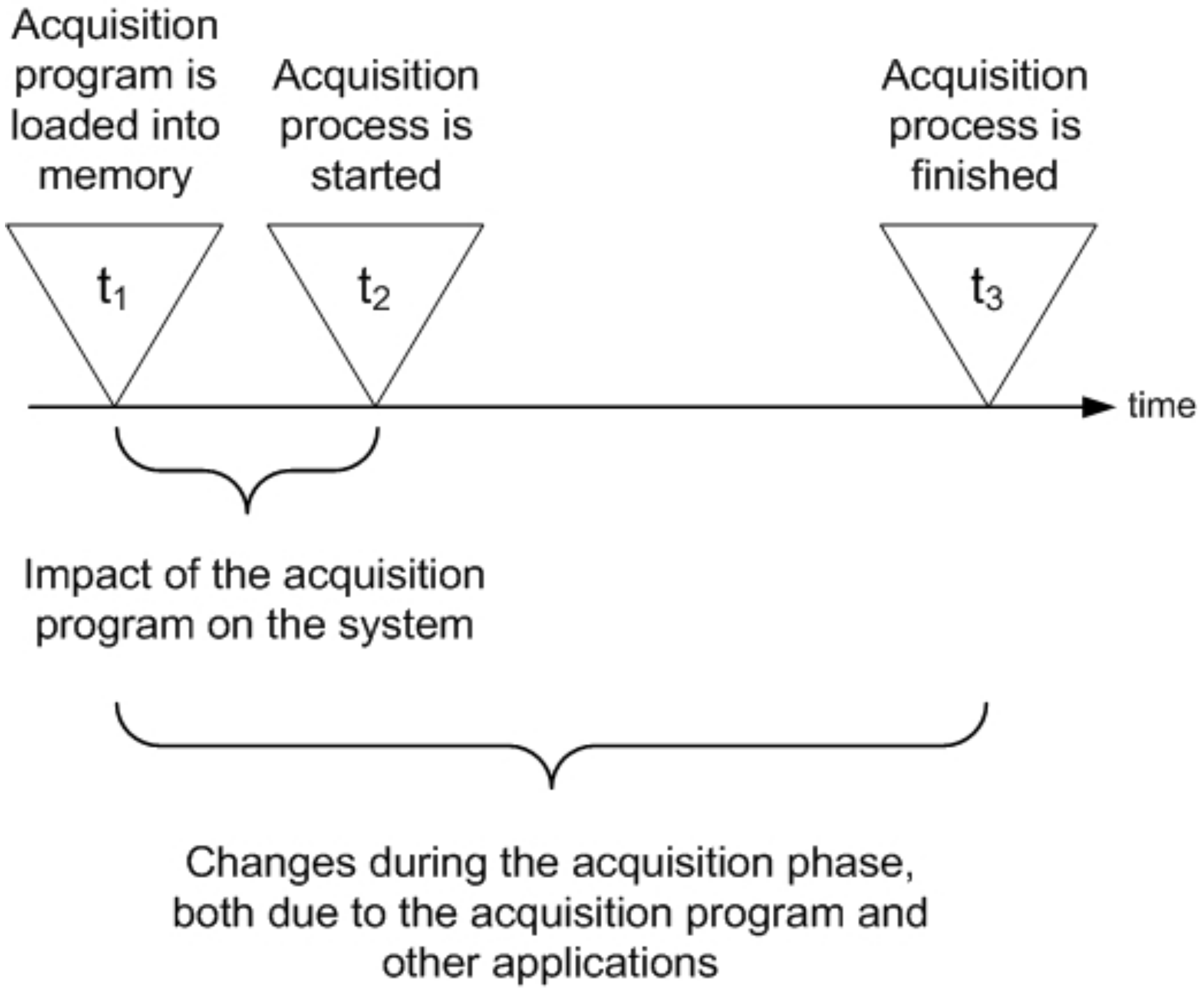
Methodology

- Measuring Atomicity
  - Idea: Use an indirect approach and attempt to quantify the degree of atomicity *violations*
    - requires monitoring the memory operations of all running threads during the acquisition phase
    - Problem: We do not know if the individual memory operations are causally related
    - Quantify *potential* atomicity violations as an upper bound



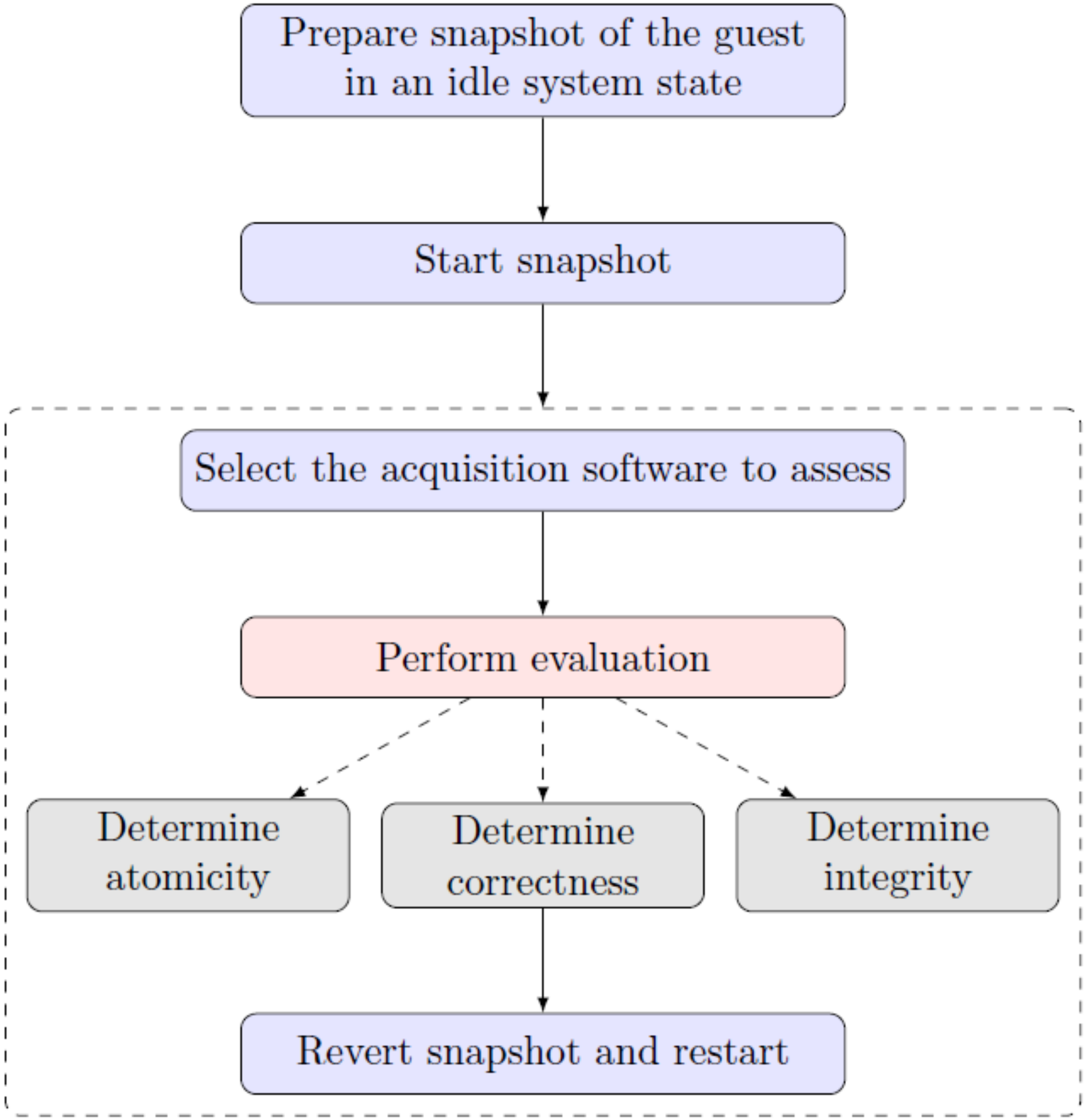
Methodology

- Measuring Integrity
  - Idea: Create external snapshots of system memory at specific point of times
    - match the created snapshots in a second step to determine the level of differences
    - permits determining how much memory was changed during the acquisition phase and due to loading the acquisition program into RAM


Acquisition program is loaded into memory
Acquisition process is started
Acquisition process is finished
t_1
t_2
t_3
time
Impact of the acquisition program on the system
Changes during the acquisition phase, both due to the acquisition program and other applications

Evaluation

- Evaluation Procedure


Prepare snapshot of the guest in an idle system state
Start snapshot
Select the acquisition software to assess
Perform evaluation
Determine atomicity
Determine correctness
Determine integrity
Revert snapshot and restart

- Evaluation of three open source imaging applications
  - win32dd, mdd, WinPMEM
- 90 test runs for each imager
- All tests initially started from an idle system state
  - Memory sizes between 512 MB and 2 GB
  - Each test required between 6.87 and 22.37 GB of disk space

Evaluation

- Correctness Evaluation
  - (older, open source version of) *win32dd* and *mdd* initially acquired the physical address space incorrectly

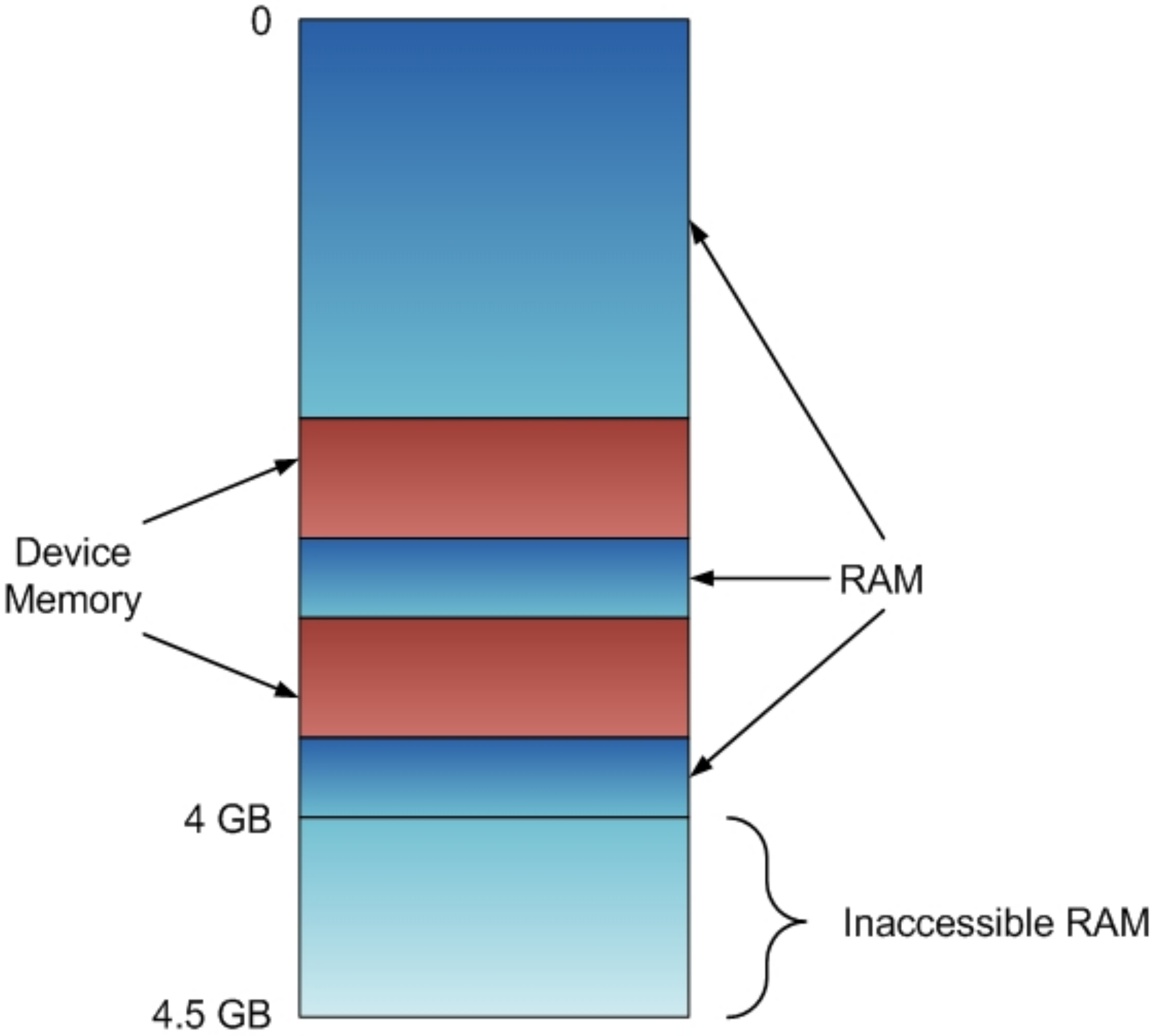
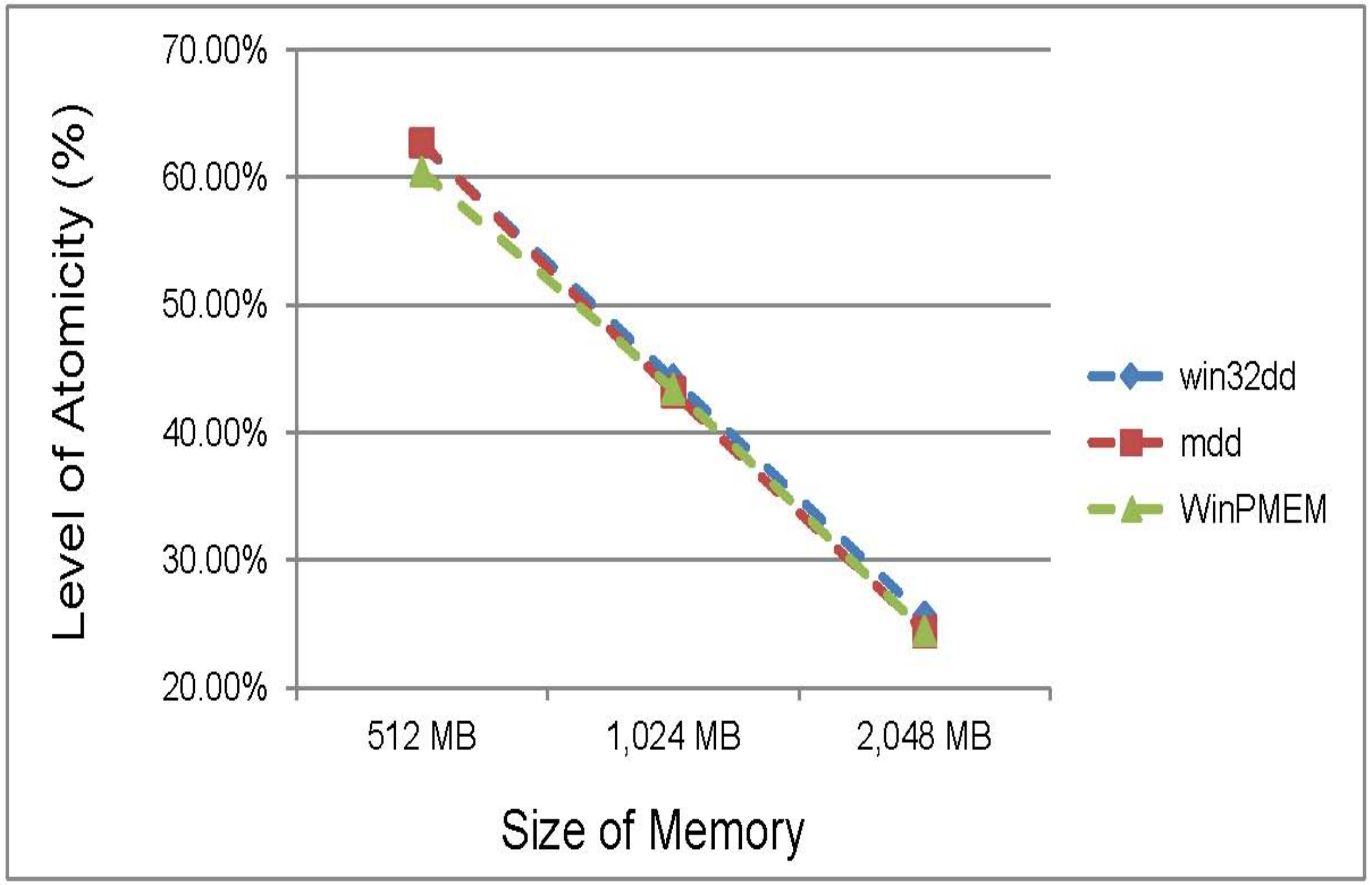


Image: http://blogs.technet.com/b/markrussinovich/archive/2008/07/21/3092070.aspx

- regions of device memory were ignored
  - offset mapping is corrupted
- after patching, all three utilities created *correct* snapshots, both in size and contents
- non-accessible regions are zeroed out

Evaluation

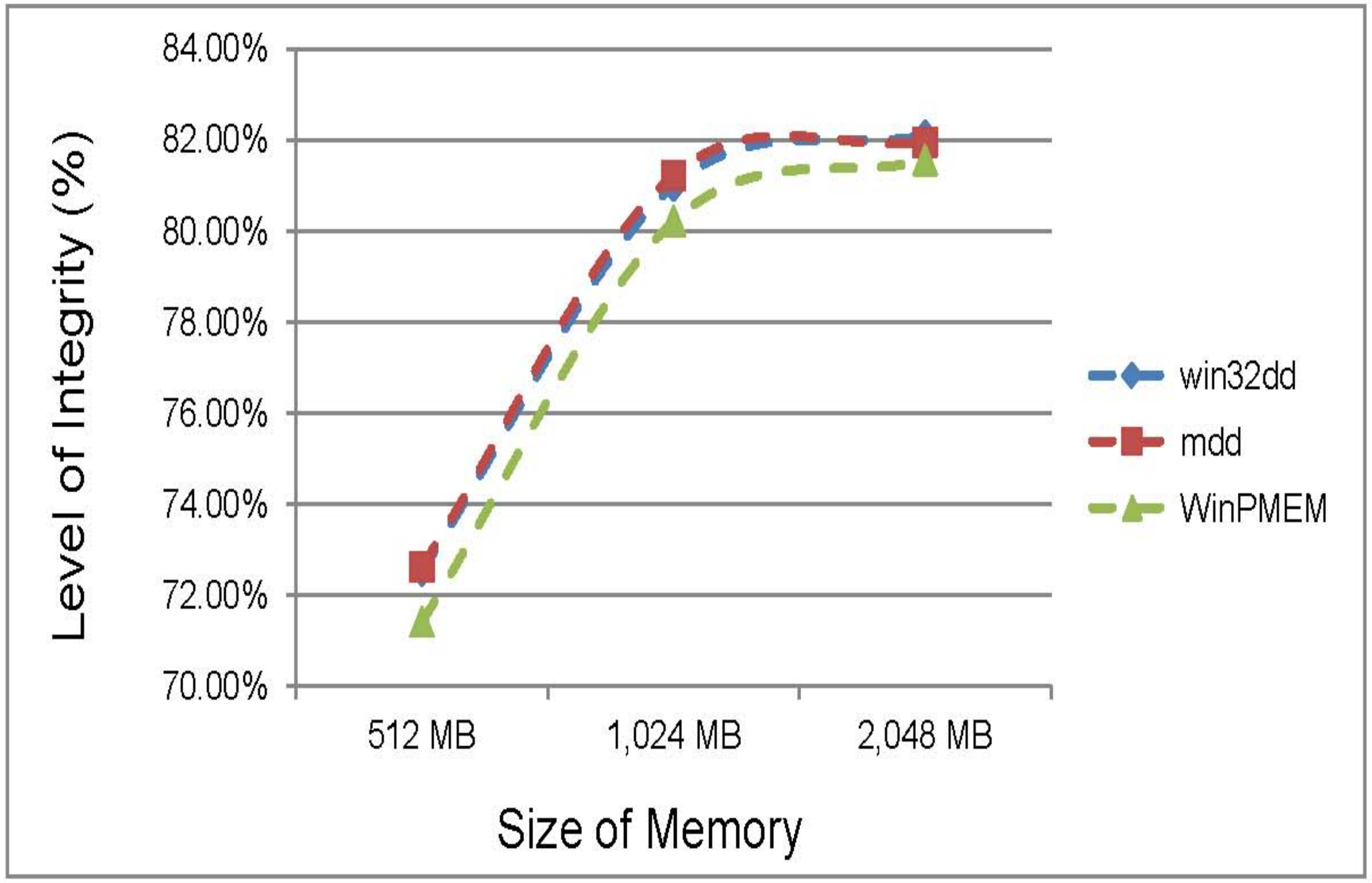
- Atomicity Evaluation
  - the level of atomicity rapidly decreased with larger memory sizes



- Theory: With longer imaging periods, it gets increasingly difficul to keep the image file free from smearing
- Open research: In how far do inconsistencies truly affect later memory analysis?
- Inconsistencies are counter-intuitive to classic perceptions of "forensic soundness" though

Evaluation

- Integrity Evaluation
  - the level of integrity slightly increased with larger memory sizes



- Theory: On a system with constant load, proportionally less amounts of space are required with higher memory capacities
- Still: About one fifth of memory is changed during the acquisition phase
- Results are similar to earlier experiments by other authors

Summary

- Summary and Outlook
  - ➢ Rigorous testing and evaluation of acquisition solutions has been widely neglected so far
  - ➢ We now have a mechanism for verifying the correctness of imaging applications and estimating their level of atomicity and integrity
    - ➢ Experiments have been performed under "laboratory" conditions
    - ➢ Next step: How reliably do acquisition solutions work in the presence of an intelligent adversary?

In case of any questions, please feel free to contact:

Stefan Vömel

http://www1.informatik.uni-erlangen.de/
stefan.voemel@cs.fau.de

i1 Chair for IT Security Infrastructures

FAU FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Discussion

- Discussion of the Evaluation Approach
  - Evaluation of a software imager requires minor patching
    - white-box testing methodology
    - so far, we have only evaluated open source solutions
  - Evaluation is limited to x86 32-bit applications and systems with a maximum memory capacity of 2 GB
    - restrictions are due to the underlying Bochs engine
  - Level of atomicity and impact of an acquisition program can only be estimated based on upper bounds