Karlsruhe Institute of Technology

TimerMeter: Quantifying Timer Method Accuracy and Invocation Cost

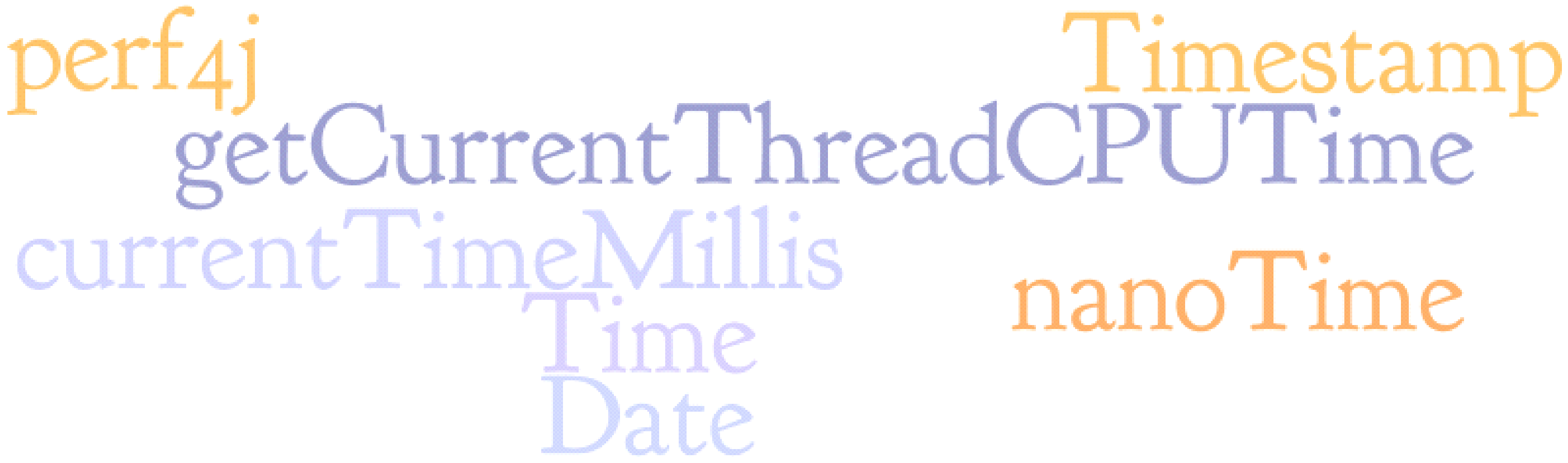
Michael.Kuperberg@kit.edu
March 24th, 2010

Forschungszentrum Karlsruhe in der Helmholtz-Gemeinschaft
Universität Karlsruhe (TH) Research University · founded 1825

Java Users Group Karlsruhe, Lightning talks

What do you use for timing in Java?

Karlsruhe Institute of Technology



Motivation

- What can you conclude from the following?

```
start = System.nanoTime();
yourMethodToBeBenchmarked();
duration System.nanoTime()-start; //e.g. 1955 ns
```

really 1955 ns?

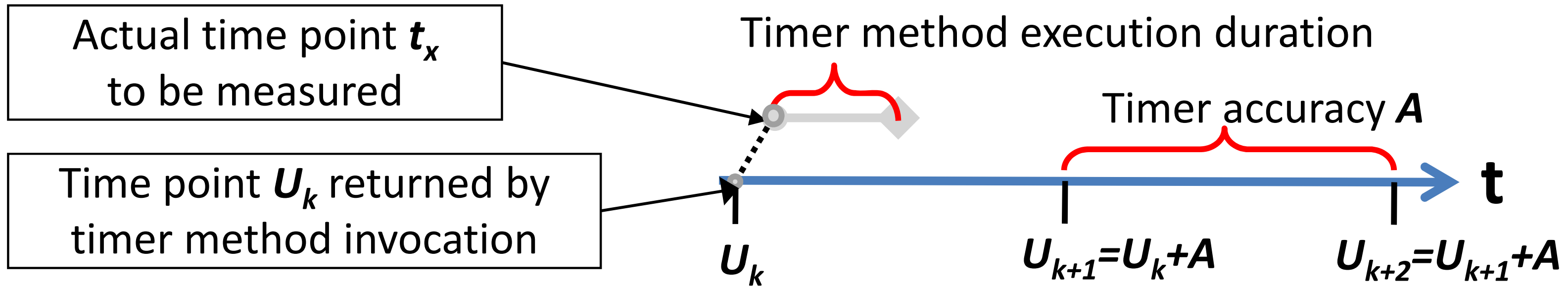
- **`java.lang.System.nanoTime()`**
  - from official javadocs: “*nanosecond precision, but not necessarily nanosecond accuracy*”
  - no API-provided means to obtain precision/accuracy
  - anecdotal evidence on the WWW, different results: e.g. accuracy „a few hundred nanoseconds“

Overview

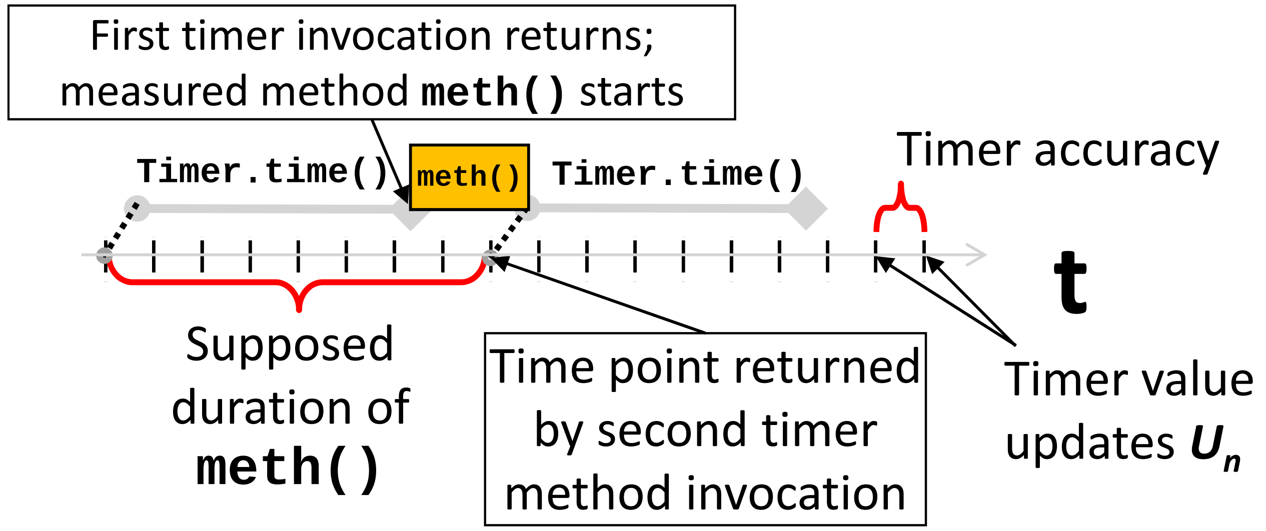
- ✓ Motivation
- Foundations
- Requirements
- Main Idea of TimerMeter
- Evaluation
- Conclusion
- (Related Work)

TimerMeter: Foundations (1)

- Effect of accuracy on measurements

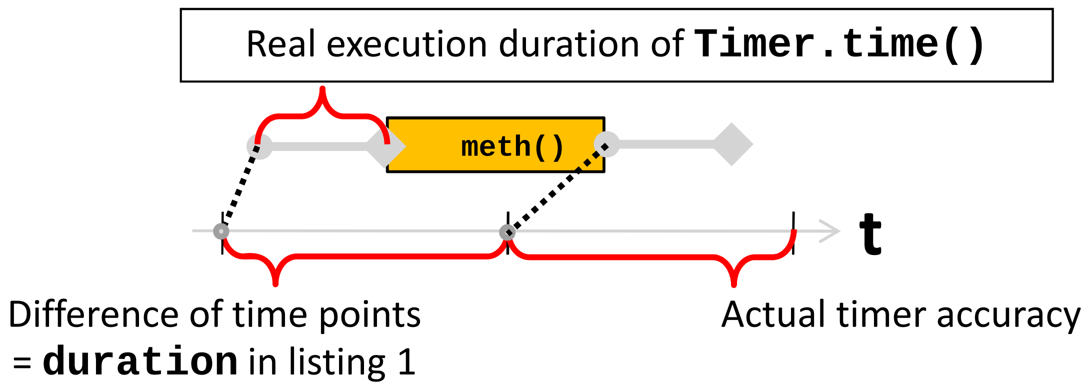

Actual time point t_x to be measured
Time point U_k returned by timer method invocation
Timer method execution duration
Timer accuracy A
t
U_k
$U_{k+1}=U_k+A$
$U_{k+2}=U_{k+1}+A$

- **Case 1**: accuracy < execution duration


First timer invocation returns; measured method **meth()** starts
Timer.time()
meth()
Timer.time()
Timer accuracy
t
Supposed duration of **meth()**
Time point returned by second timer method invocation
Timer value updates U_n

TimerMeter: Foundations (2)

- **Case 2**: accuracy ≥ execution duration


Real execution duration of Timer.time()
meth()
t
Difference of time points
= duration in listing 1
Actual timer accuracy

- Thus, we need to know both accuracy and execution duration!
- It‘s hard to disentangle the measurement overhead from what‘s being measured

TimerMeter: Requirements

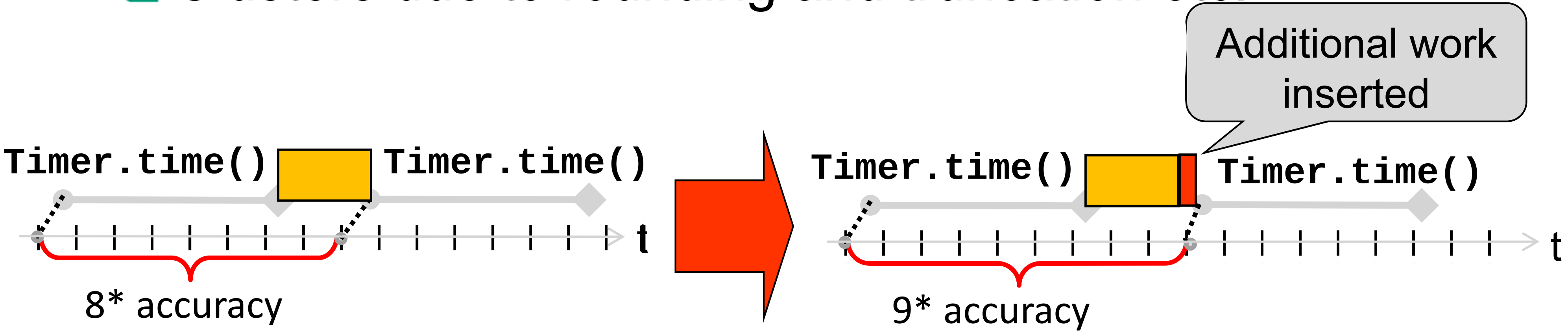
- Context: timers for fine-grained benchmarks and measurements ($\rightarrow$ less than a microsecond)
- Usual approach is „take the best available timer“
  - Java: many timer methods available, incl. 3rd-party
  - The choice is often not clear, not justifiable – or wrong

- Thus: **we need to know timer accuracy / resolution**
  - HW-specific and OS-specific
  - they contribute to the measured timing values
  - they impact the statistical quality of results

Solution: next slides!

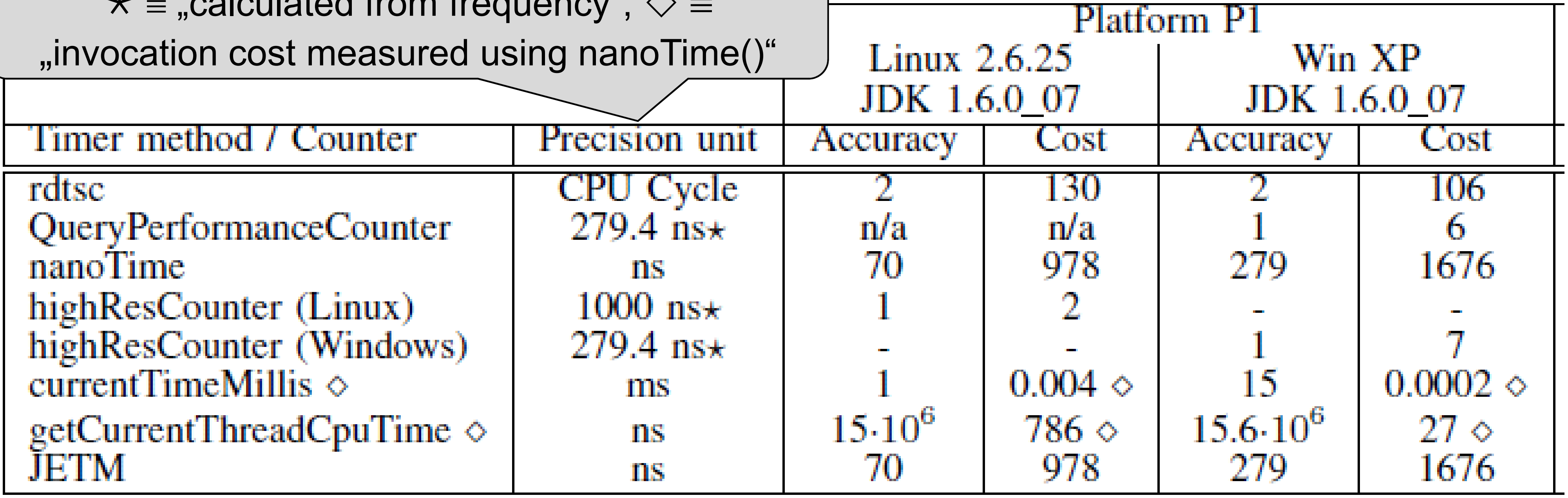
TimerMeter: The Main Idea

- Central idea of TimerMeter: gradually and slowly increase work between timer invocations
  - so that the measured interval increases by an accuracy at some point (cf. paper)
  - works for Accuracy < InvocationCost and A ≥ IC
- Implementation is more complex
  - Clusters due to rounding and truncation etc.



TimerMeter: Evaluation: Results

⋆ ≡ „calculated from frequency“, ◇ ≡ „invocation cost measured using nanoTime()“

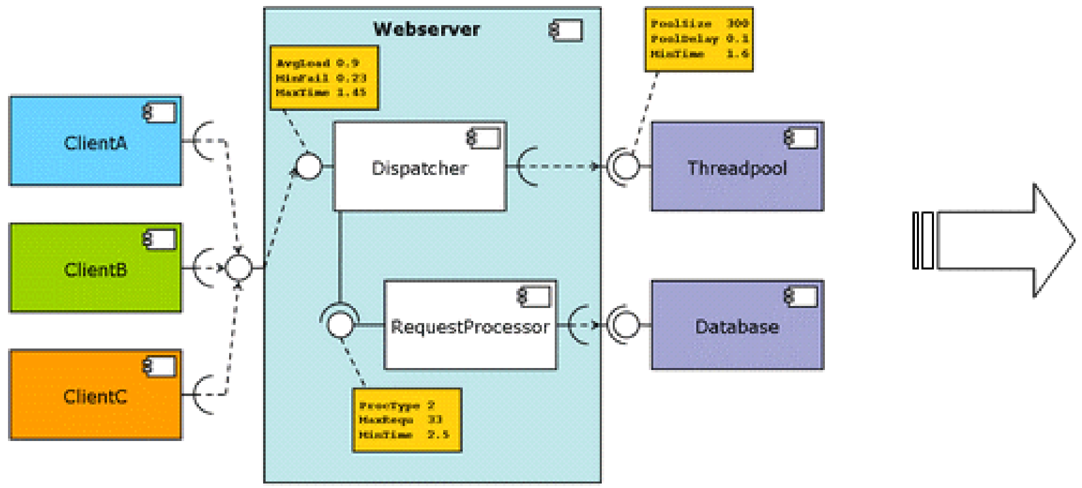


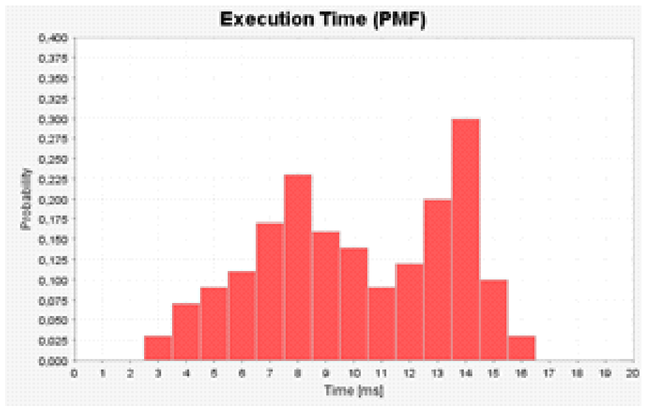
		Platform P1			
		Linux 2.6.25 JDK 1.6.0_07		Win XP JDK 1.6.0_07	
Timer method / Counter	Precision unit	Accuracy	Cost	Accuracy	Cost
rdtsc	CPU Cycle	2	130	2	106
QueryPerformanceCounter	279.4 ns⋆	n/a	n/a	1	6
nanoTime	ns	70	978	279	1676
highResCounter (Linux)	1000 ns⋆	1	2	-	-
highResCounter (Windows)	279.4 ns⋆	-	-	1	7
currentTimeMillis ◇	ms	1	0.004 ◇	15	0.0002 ◇
getCurrentThreadCpuTime ◇	ns	$15 \cdot 10^6$	786 ◇	$15.6 \cdot 10^6$	27 ◇
JETM	ns	70	978	279	1676

- Note the difference of OSes for `nanoTime()`
- Note the accuracy of `getCurrentThreadCpuTime`
- JETM: very similar to `nanoTime()`: accuracies correspond to (rounded) value of 1 HPET counter tick

The Larger Context: http://www.palladio-approach.net

- Model-based Architecture Performance Evaluation





- Research on software architectures (KIT and FZI): benchmarking, reverse engineering, reliability, etc.
- Development using Eclipse: GEF, GMF, GEF etc.
- Student theses (master, bachelor) and jobs available

TimerMeter: Conclusions

- TimerMeter: a novel, easily portable algorithm
  - For transparently quantifying accuracy and invocation cost of timer methods on <u>your</u> platform
  - Available under EPL. Details, docs etc.: **http://bit.ly/TimerMeter** or **https://sdqweb.ipd.kit.edu/wiki/TimerMeter**
- The accuracy of a timer method can differ by 10x or more depending on OS (e.g. `currentTimeMillis`)
  - `nanoTime()`: accuracy „only“ 70 ns up to 279 ns
  - `nanoTime()`: invocation cost: 978 ns up to 1676 ns