UNIVERSITÀ
DI SIENA
1240

Analisi dei fenomeni transienti sul Sistema Operativo Android

STRUMENTI, PROFILAZIONE E ANALISI

▶ DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE E SCIENZE MATEMATICHE

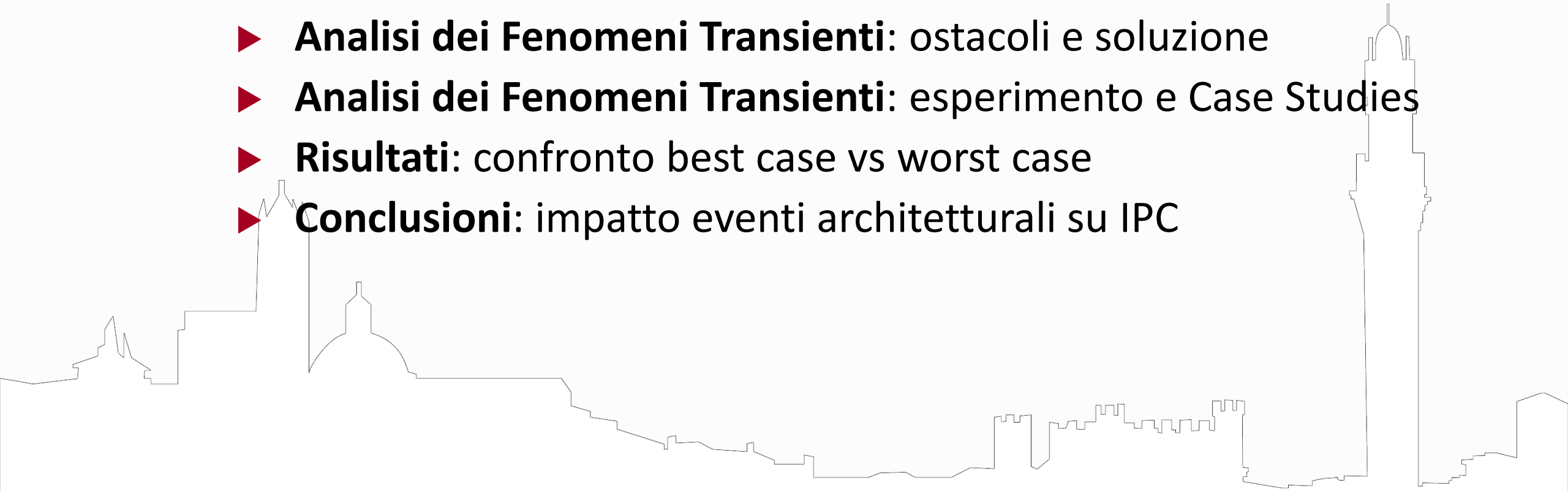
Emanuele Angiolilli, matricola 103629
RELATORE Prof. Sandro Bartolini

SIENA 16/10/2025

UNIVERSITÀ DI SIENA
1240

AGENDA

- **Introduzione**: la rilevanza e il problema
- **Strumenti di analisi**: Simpleperf e Perfetto
- **Analisi dei Fenomeni Transienti**: ostacoli e soluzione
- **Analisi dei Fenomeni Transienti**: esperimento e Case Studies
- **Risultati**: confronto best case vs worst case
- **Conclusioni**: impatto eventi architetturali su IPC



UNIVERSITÀ DI SIENA
1240

INTRODUZIONE

La rilevanza e il problema

Dal mercato emerge una forte pressione verso l'**efficienza** di esecuzione di più servizi sullo stesso calcolatore avente **architettura multicore.** Ciò è ancora più importante su un **dispositivo mobile**.

In condizioni di **multitasking** le risorse di un calcolatore saranno **contese tra più applicazioni**: è necessario un **cambio di contesto** ogniqualvolta si eseguirà un processo differente.

A seguito di **un cambio di contesto** si istaureranno **fenomeni transienti** che porteranno ad una **degradazione** delle **prestazioni**

▶ ***La tesi ha l'obiettivo di analizzare l'impatto di specifici eventi architetturali sulle prestazioni di un dispositivo mobile, a seguito di un cambio di contesto.***

*Un fenomeno transiente è **l'effetto** sull'architettura e sulle prestazioni di una applicazione in **conseguenza** di un cambio di contesto/interruzione.*

UNIVERSITÀ DI SIENA
1240

STRUMENTI DI ANALISI

Simpleperf e Perfetto

Dispositivo in Analisi: Google Pixel 7
SoC Tensor G2

- *4x Core Little Cortex-A55 (CPU0-3)*
- *2x Core Medium Cortex-A78 (CPU4-5)*
- *2x Core BIG Cortex-X1 (1.8GHz) (CPU6-7)*

▶ SIMPLEPERF: *tool nativo di Android per la profilazione della CPU. In base agli argomenti passati possiamo esplicitare gli eventi architetturali da monitorare.*

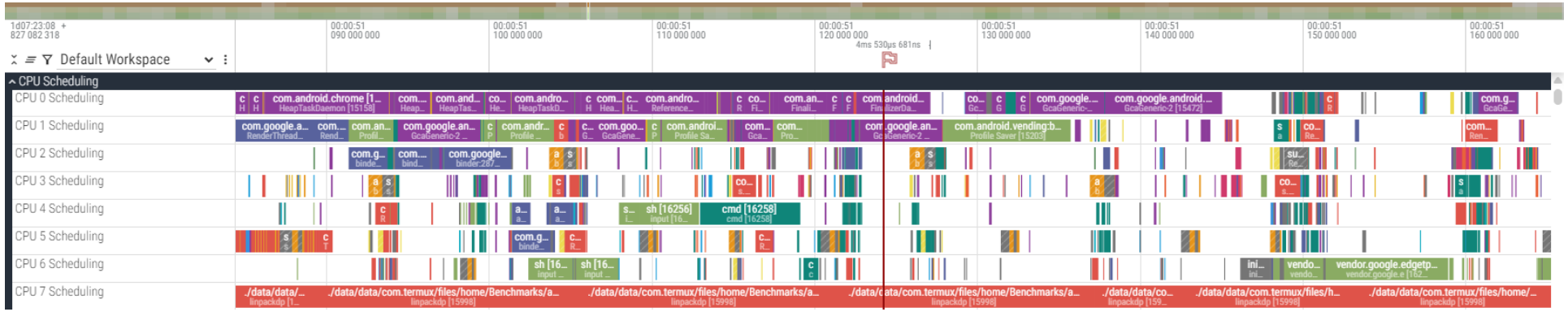
Esempi di uso

```
$ simpleperf -e instructions -a ./hello_world
```

```
Hello World!
Performance counter statistics:
#       count  event_name      # count / runtime
  11,140,764   instructions    # 864.079 M/sec
Total test time: 0.015108 seconds.
```

▶ PERFETTO: *strumento di profilazione integrato in Android in grado di registrare le informazioni in una traccia visualizzabile su linea temporale tramite un'interfaccia web interattiva.*

Esempio di traccia



UNIVERSITÀ DI SIENA
1240

ANALISI DEI FENOMENI TRANSIENTI

Gli ostacoli e le soluzioni

Si è voluto analizzare e quantificare l'impatto di determinati **eventi architetturali sulle prestazioni** di un'applicazione in condizione di elevati cambi di contesto

Multitasking limitato di Andorid	Simulazione di uno schermo secondario all'interno del dispositivo
Thermal Throttling	Limitare la frequenza di Clock del processore (UnderClock)
Assegnazione risorse di Android	Creazione di CPU Set personalizzati

UNIVERSITÀ DI SIENA
1240

THERMAL THRTOTTLING

Il focus

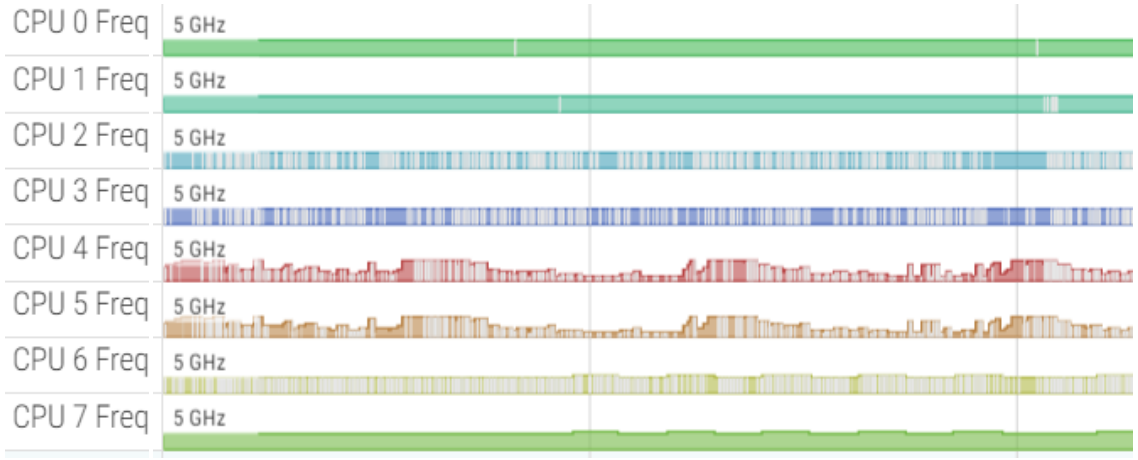
PROBLEMA

Eseguire prolungati test porta il dispositivo a surriscaldarsi. Il sistema è costretto a limitare la frequenza di Clock della CPU → **ANALISI IMPREVEDIBILE**

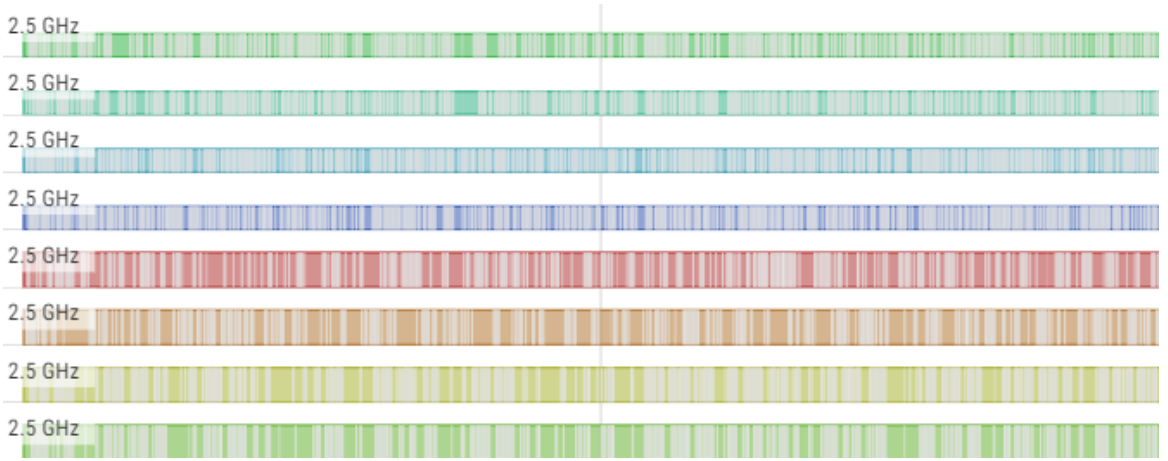
SOLUZIONE

Soluzione: eseguire un UnderClock della CPU limitando la frequenza di Clock per avere **temperatura e frequenze stabili nel tempo** → ***ANALISI ACCURATA***

Frequenza CPU Prima dell'UnderClock



Frequenza CPU Dopo l'UnderClock



UNIVERSITÀ DI SIENA
1240

ASSEGNAZIONE RISORSE ANDROID

Il focus

In Android le app possono trovarsi in 3 stati in base all'interazione dell'utente:

- FOREGROUND
- BACKGROUND
- TOP-APP

In base allo stato in cui si trova, all'app sono assegnati determinati Core definiti da CPUSet omonimi.

PROBLEMA

Dispositivo controllato da Script: all'applicazione analizzata non verranno assegnate le risorse desiderate.

SOLUZIONE

Creazione CPUSet Custom per assegnare in **modo stabile** i Core desiderati all'app da profilare.

▶ *Un CPUSet è una funzionalità del Kernel Linux che permette di vincolare un gruppo di processi ad un insieme di Core della CPU*

UNIVERSITÀ DI SIENA 1240

ANALISI DEI FENOMENI TRANSIENTI

Esperimento|esempio di caso articolato

Con **Simpleperf** verrà profilata l'applicazione principale in 2 condizioni:

- TRANSITORIO → **cambio di contesto** tra due app secondarie **(i disturbi)**
- REGIME → app secondaria in Foreground da almeno 1 sec

L'app da profilare rimarrà fissa sul display virtuale mentre i **disturbi** si **alterneranno** sul display principale. Verranno **variati i Core assegnati** sia **ai disturbi** che **all'app principale** e **confrontati** i **risultati** nelle 2 condizioni.

APPLICAZIONE PRINCIPALE DA PROFILARE

Reason why:

- **Notevole stress** sulla CPU pur non avendo interazione con l'utente
- Utilizzo costante/prevedibile delle risorse nel tempo

UNIVERSITÀ DI SIENA
1240

ANALISI DEI FENOMENI TRANSIENTI

Script BASH per estrazione automatica dei dati

```
adb shell settings put global overlay_display_devices
   null
sleep 0.3
adb shell settings put global overlay_display_devices
   1920x1080/225
sleep 0.3
DisplayId=‘adb shell dumpsys display | grep "mDisplayId="
    | tail -n 1|cut -d= -f2‘
adb shell am start -S -n com.mojang.minecraftpe/.
   MainActivity --display $DisplayId --windowingMode 1
```

Creazione di un **display virtuale** secondario e **avvio dell’app** principale su display desiderato.

```
PID=$(adb shell "pidof com.mojang.minecraftpe")
adb shell "su -c \"echo $PID > /dev/cpuset/Custom/tasks\"
   "
adb shell "su -c \"pstree -p $PID | sed ’s/.*(\(.*\))/\1/
   ’ > /dev/cpuset/Custom/tasks\""
adb shell su -c taskset -ap FF ’$(pidof com.mojang.
   minecraftpe)’
```

Assegnazione dei **Core** desiderati all’app principale usando un **CPUSet Custom**.

```
PID=$(adb shell su -c perfetto --background -c /data/
   local/tmp/config_long.pbtx --txt -o /data/misc/
   perfetto-traces/trace_transientemultitask$(date -d "
   today" +"%Y%m%d%H%M").perfetto-trace )
```

Avvio della **traccia** di Perfetto

UNIVERSITÀ DI SIENA
1240

ANALISI DEI FENOMENI TRANSIENTI

Script BASH per estrazione automatica dei dati

```
for j in {0..6}
do
    (adb shell "su -c 'simpleperf stat  -e branch-
       misses,raw-dtlb-walk,raw-itlb-walk --app com.
       mojang.minecraftpe --duration 1' " | awk '{
       print $1}' | sed 's|[^0-9]||g' | grep . &&
       echo ) >> outputtransitorio&
    sleep 0.1
    adb shell input swipe 500 2400  1300 2400
    sleep 2
    (adb shell "su -c 'simpleperf stat  -e branch-
       misses,raw-dtlb-walk,raw-itlb-walk --app com.
       mojang.minecraftpe --duration 1' " | awk '{
       print $1}' | sed 's|[^0-9]||g' | grep . &&
       echo ) >> outputregime&
    sleep 2
done
```

Profilazione dell'app principale in condizione di **Regime e Transitorio**. Verranno estratti gli eventi architetturali d'interesse.

```
adb shell "su -c 'chown shell:shell /data/misc/perfetto-
   traces/*'"
adbsync pull /data/misc/perfetto-traces/ /home/angiolilli
   /Desktop/Tesi/tracce/perfetto-traces
python3 perftransitorio.py outputtransitorio
python3 perftransitorio.py outputregime
```

Estrazione della **traccia** di Perfetto e **media** sui dati.

UNIVERSITÀ DI SIENA 1240

ANALISI DEI FENOMENI TRANSIENTI

Casi di studio

Sono stati analizzati i dati acquisiti dall'esperimento per cercare di **determinare** quanto un rallentamento prestazionale sia **correlato** ad un aumento di metriche di **eventi architetturali** «costosi». Per fare ciò è stato **confrontato il caso Regime e Transitorio**.

- INDICE PRESTAZIONI → Instructions-per-Cycle (IPC)
- EVENTI ARCHITETTURALI → Cache MISS, Branch Predictor MISS, TLB Walks

I **casi di studio** considerati sono:

- 8 Core a Minecraft
- 6 Core (Little + Medium) a Minecraft

★☆☆

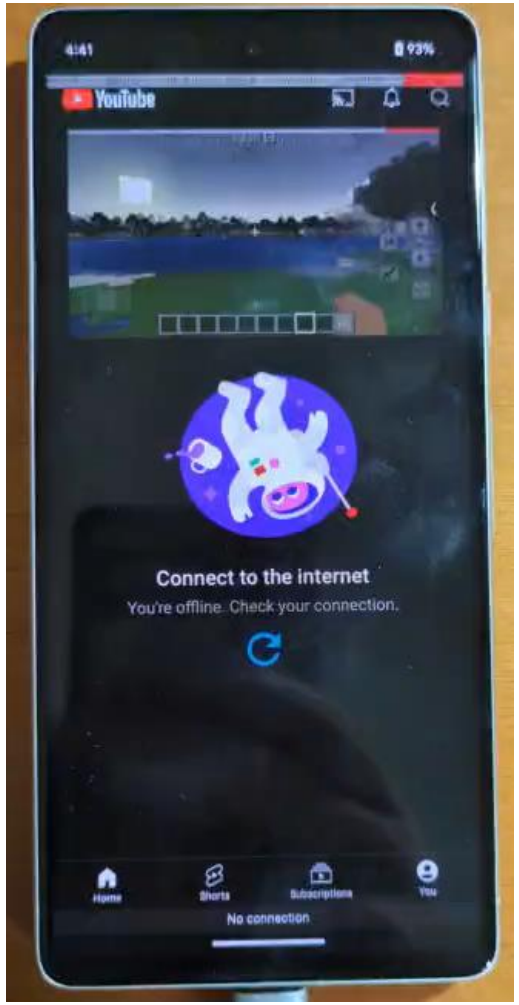
Core assegnati ai disturbi:

- 8 Core
- 6 Core (Little + Medium)
- 4 Core (Medium + BIG)
- 4 Core (Little)
- 2 Core (BIG)

UNIVERSITÀ DI SIENA
1240

ANALISI DEI FENOMENI TRANSIENTI

Dati ottenuti dalla profilazione



Assegnazione Risorse

Eventi architetturali

Tabella 2.1: TRANSITORIO Minecraft 8 CORE

Core ai Disturbi	IPC	Cache Miss	BP Miss	dTLB Walks	iTLB Walks
6Core (Little+Medium)	0,63051	12097080	5824525	3446941	958701
4Core (Medium+Big)	0,43787	11621773	6404868	2875303	966963
2Core (Big)	0,50451	13366729	6584369	4069180	1458827
4Core (Little)	0,65137	11889481	5684443	3780750	1199767
8Core	0,48514	11967453	6559637	3620424	1179062

Tabella 2.2: REGIME Minecraft 8 CORE

Core ai Disturbi	IPC	Cache Miss	BP Miss	dTLB Walks	iTLB Walks
6Core (Little+Medium)	0,60777	12093133	5841520	3821997	1085514
4Core (Medium+Big)	0,54114	12312899	6340615	3593321	1041016
2Core (Big)	0,53569	13619468	6594781	3298328	1075237
4Core (Little)	0,61380	12838224	6036021	4047769	1211779
8Core	0,56006	12738853	6232228	3921615	1111333

UNIVERSITÀ DI SIENA
1240

RISULTATI

Caso di Studio| 8 Core a Minecraft

Caso Migliore: 6 Core (Little + Medium) ai disturbi

Stato	IPC	Cache Miss	BP Miss	dTLB Walks	iTLB Walks
Regime	0.60777	12,093,133	5,841,520	3,821,997	1,085,514
Transitorio	0.63051	12,097,080	5,824,525	3,446,941	958,701
Variazione %	**-3.61%**	**-0.03%**	**+0.29%**	**+10.88%**	**+13.23%**

- ✓ *IPC transitorio > Regime*
- ✓ *Fluidità elevata costante*
- ✓ *Affinità Core Big nel Transitorio*

Caso Peggiore: 2 Core (BIG) ai disturbi

Stato	IPC	Cache Miss	BP Miss	dTLB Walks	iTLB Walks
Regime	0.53569	13,619,468	6,594,781	3,298,328	1,075,237
Transitorio	0.50451	13,366,729	6,584,369	4,069,180	1,458,827
Variazione %	**+6.18%**	**+1.89%**	**+0.16%**	**-18.94%**	**-26.29%**

- ▪ *Elevato stress memoria*
- ▪ *Prestazioni mediocri*

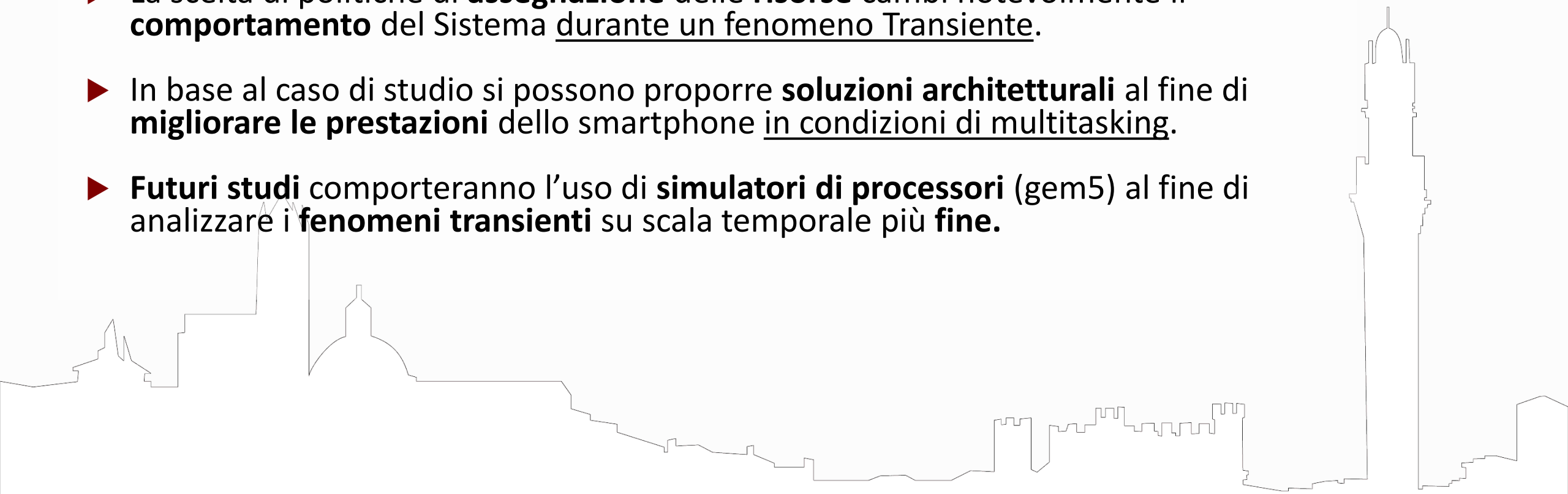
UNIVERSITÀ DI SIENA
1240

CONCLUSIONI

Impatto eventi architetturali su IPC

Dall'analisi svolta si è osservato come:

- La scelta di politiche di **assegnazione** delle **risorse** cambi notevolmente il **comportamento** del Sistema durante un fenomeno Transiente.
- In base al caso di studio si possono proporre **soluzioni architetturali** al fine di **migliorare le prestazioni** dello smartphone in condizioni di multitasking.
- **Futuri studi** comporteranno l'uso di **simulatori di processori** (gem5) al fine di analizzare i **fenomeni transienti** su scala temporale più **fine.**



UNIVERSITÀ
DI SIENA
1240

GRAZIE PER L'ATTENZIONE