WORKSHOP MODCS 2016

Extending Mobile Device Autonomy Using Cloud Computing

Thiago Felipe da Silva Pinheiro

tfs3@cin.ufpe.br

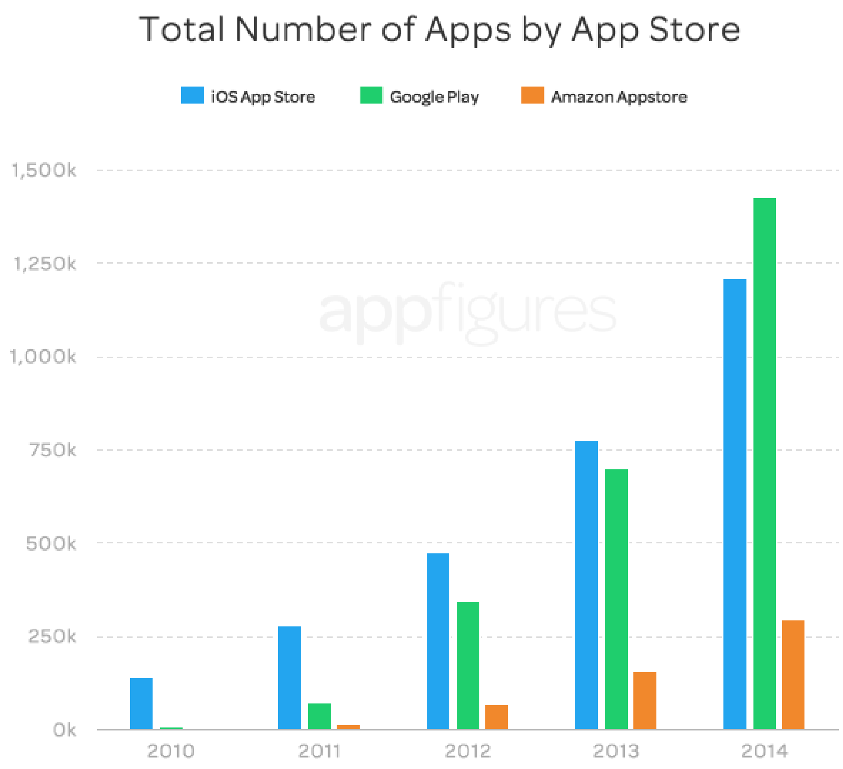
Advisor: Dr. Paulo Romero Martins Maciel

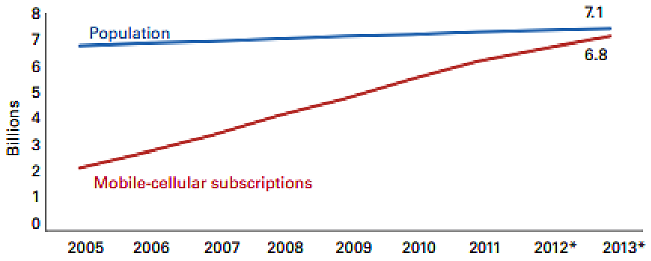
Centro de Informática U·F·P·E

Universidade Federal de Pernambuco

70 anos Tempos Transversos

Motivation

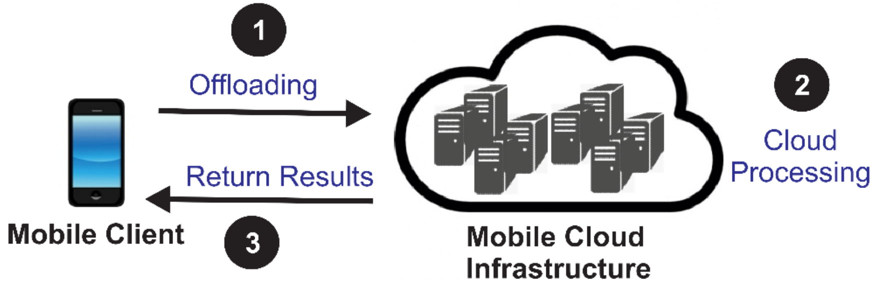


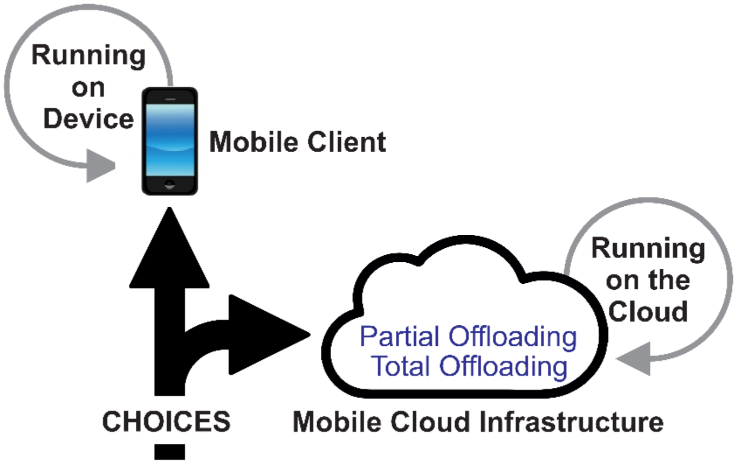


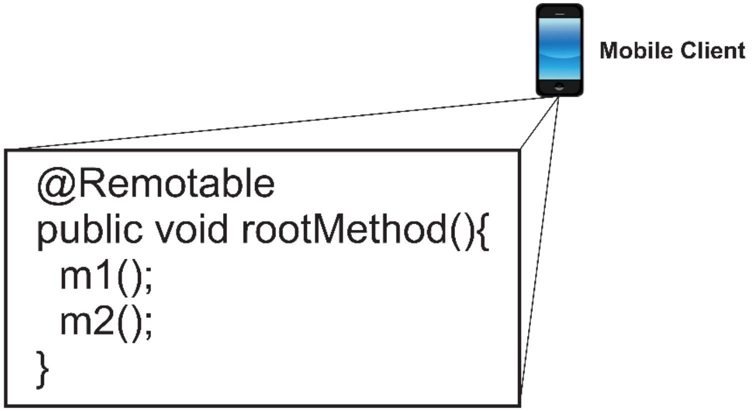
Source: ITU World Telecommunication /ICT Indicators database
Note: * Estimate

Motivation

How MCC Works?







```
@Remotable
public void rootMethod(){
  m1();
  m2();
}
```

```
<methods>
   <method name="m1" target="Device"/>
   <method name="m2" target="Cloud"/>
</methods>
```

Possibility	*m1()*	*m2()*
Scenario #1	mobile	mobile
Scenario #2	mobile	cloud
Scenario #3	cloud	mobile
Scenario #4	cloud	cloud

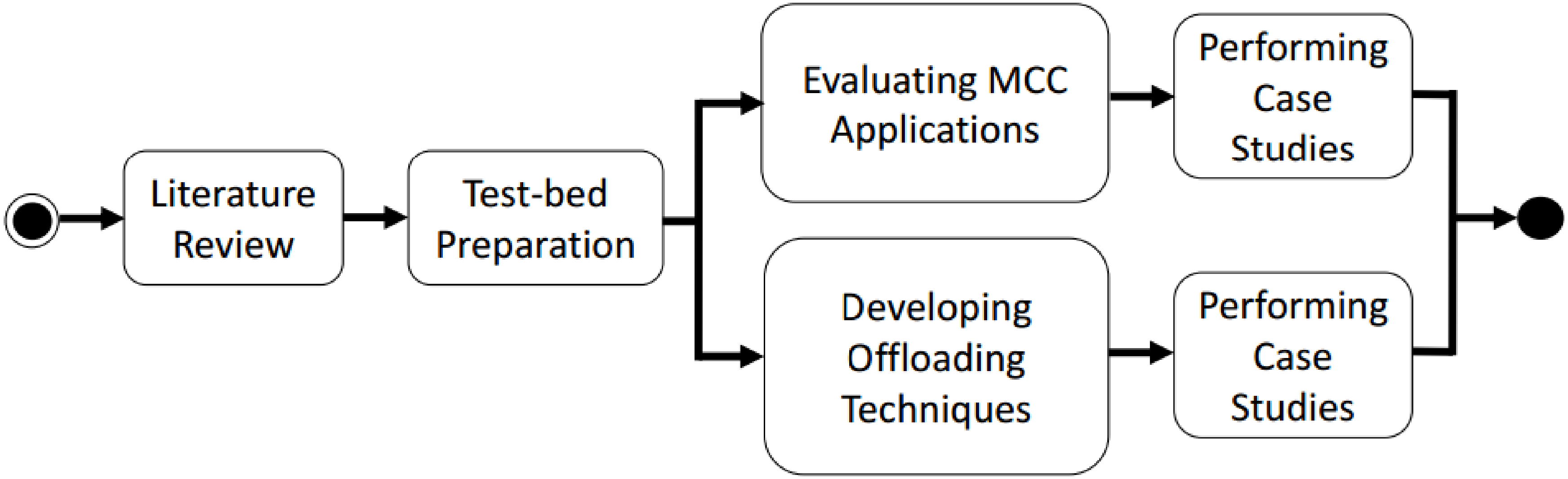
General Objective

The main objective of this research is to conceive, design and implement methods applied to mobile cloud computing to support performance improvement and extending mobile device autonomy focusing on partial method-call offload.

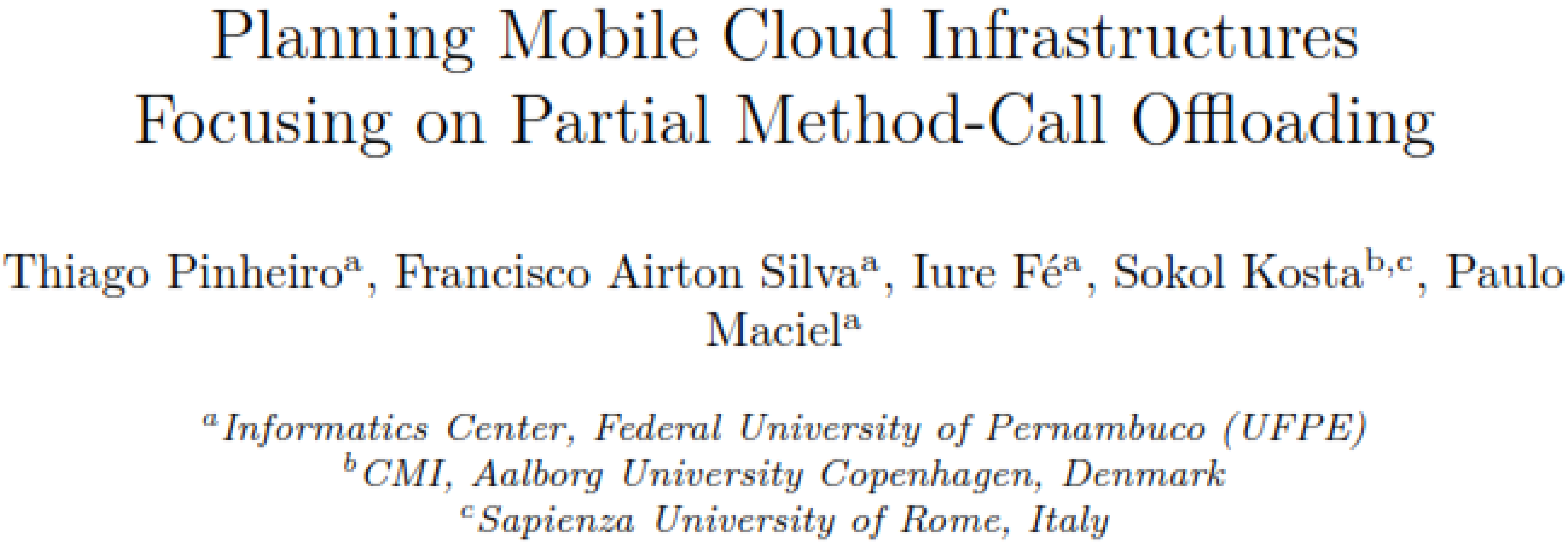
Specific Objectives

1. Conceive strategies to support MCC application performance evaluation.
2. Propose MCC application performance models.
3. Design, implement and improvement tools to evaluate MCC applications.
4. Conceive, design and implement mobile application offloading techniques aiming at performance and autonomy improvement focusing on partial offloading.

Research Methodology



Case study



Planning Mobile Cloud Infrastructures Focusing on Partial Method-Call Offloading

Thiago Pinheiro[a], Francisco Airton Silva[a], Iure Fé[a], Sokol Kosta[b,c], Paulo Maciel[a]

[a]*Informatics Center, Federal University of Pernambuco (UFPE)*
[b]*CMI, Aalborg University Copenhagen, Denmark*
[c]*Sapienza University of Rome, Italy*

Case study: Planning Mobile Cloud Infrastructures Focusing on Partial Method-Call Offloading

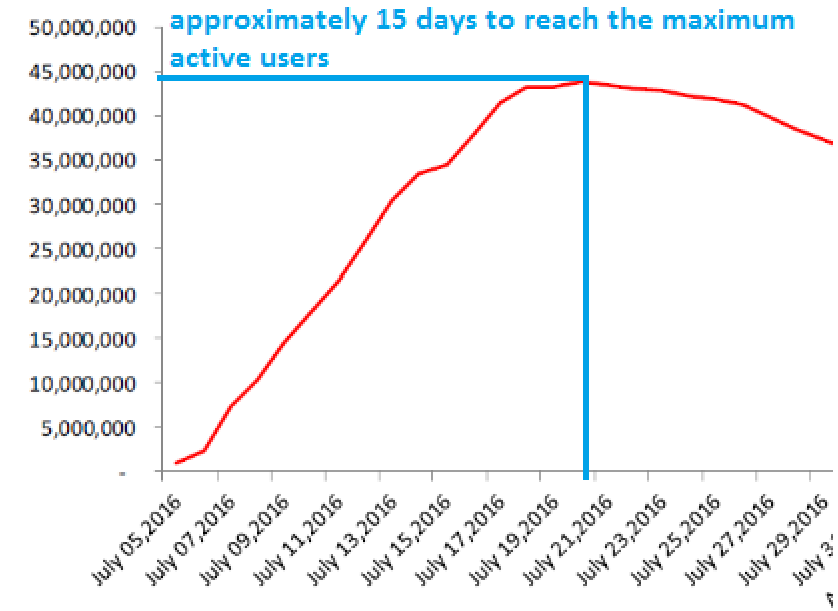
This case study sought to answer the following question:

Considering a set of method-call offloading possibilities how to decide which of them brings better results in terms of both, performance and financial costs?

Table 5: Prices of Amazon EC2 per Transfered Bytes [27]

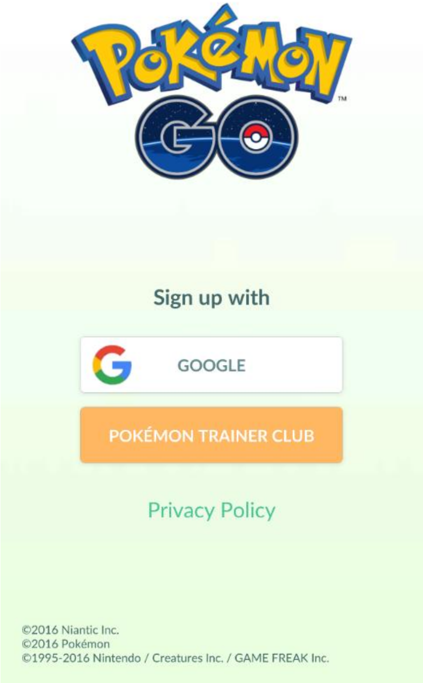
Data Transfer OUT To Internet	Price/GB
First 10 TB / month	$0.155
Next 40 TB / month	$0.115
Next 100 TB / month	$0.090
Next 350 TB / month	$0.065

Figure 1: *Pokémon Go* Worldwide Daily Active Users

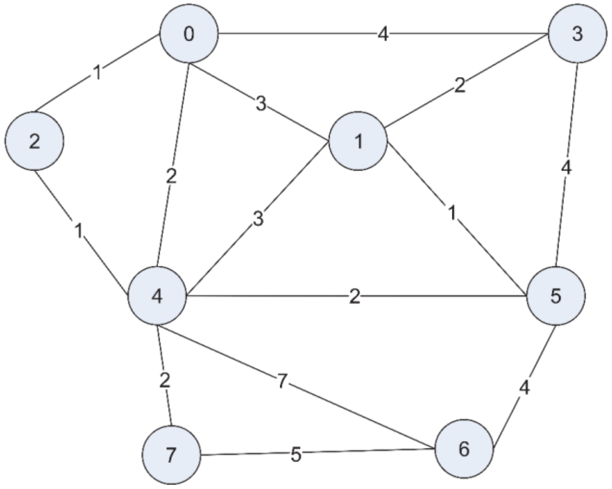


approximately 15 days to reach the maximum active users

Case study: Planning Mobile Cloud Infrastructures Focusing on Partial Method-Call Offloading



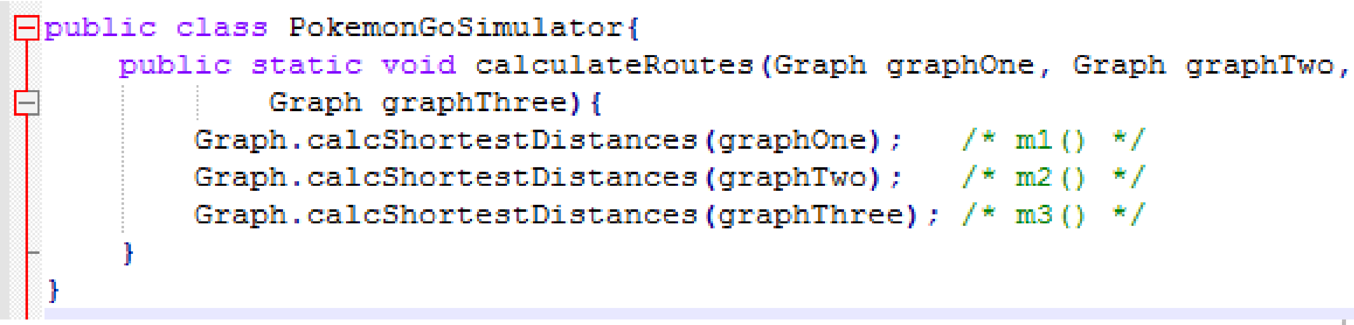
©2016 Niantic Inc.
©2016 Pokémon
©1995-2016 Nintendo / Creatures Inc. / GAME FREAK Inc.



Example Graph of Pokémon Go Simulator

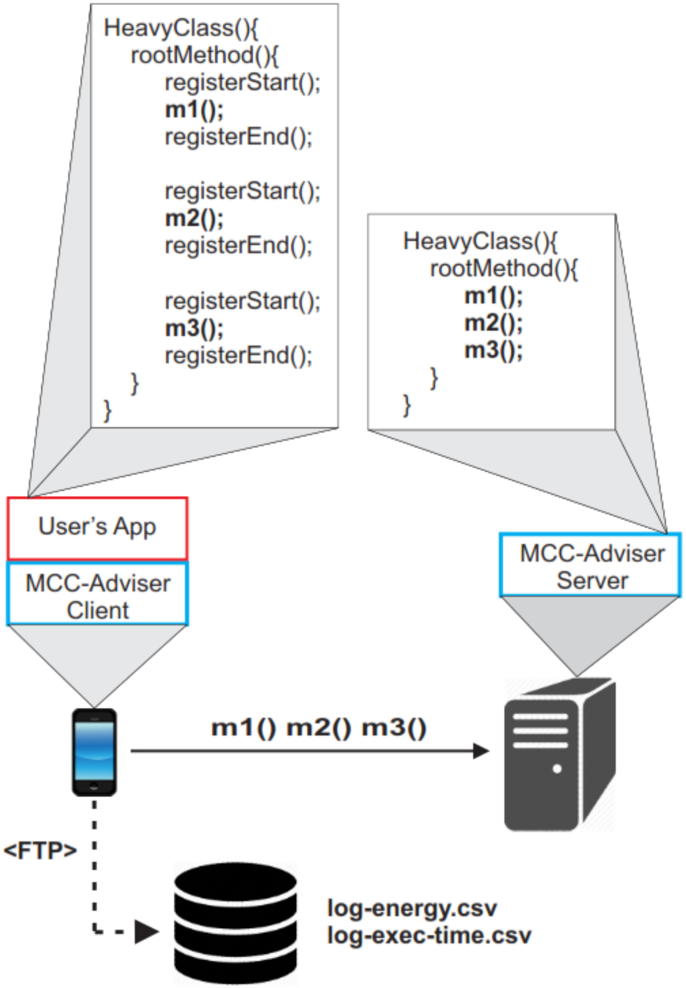
Table 2: Scenarios to Methods Calls Executions

Scenario	*m1()*	*m2()*	*m3()*
#1	cloud	cloud	cloud
#2	cloud	cloud	device
#3	cloud	device	cloud
#4	cloud	device	device
#5	device	cloud	cloud
#6	device	cloud	device
#7	device	device	cloud
#8	device	device	device



```
public class PokemonGoSimulator{
    public static void calculateRoutes(Graph graphOne, Graph graphTwo,
            Graph graphThree){
        Graph.calcShortestDistances(graphOne);   /* m1() */
        Graph.calcShortestDistances(graphTwo);   /* m2() */
        Graph.calcShortestDistances(graphThree); /* m3() */
    }
}
```

Case study: Planning Mobile Cloud Infrastructures Focusing on Partial Method-Call Offloading

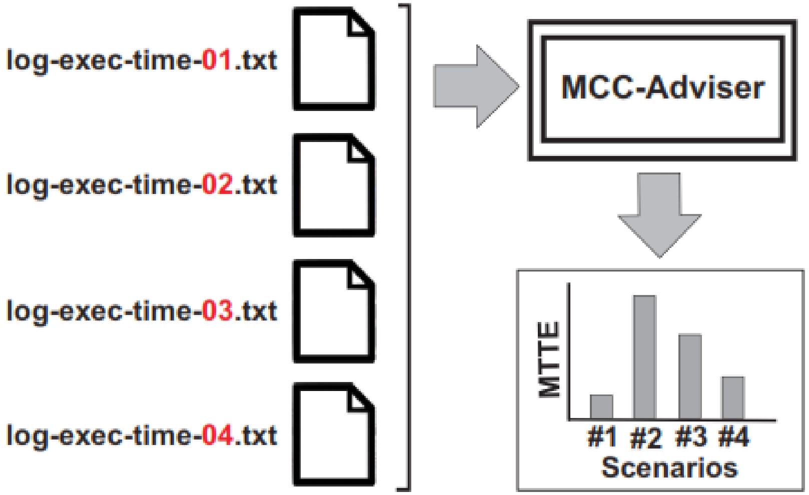


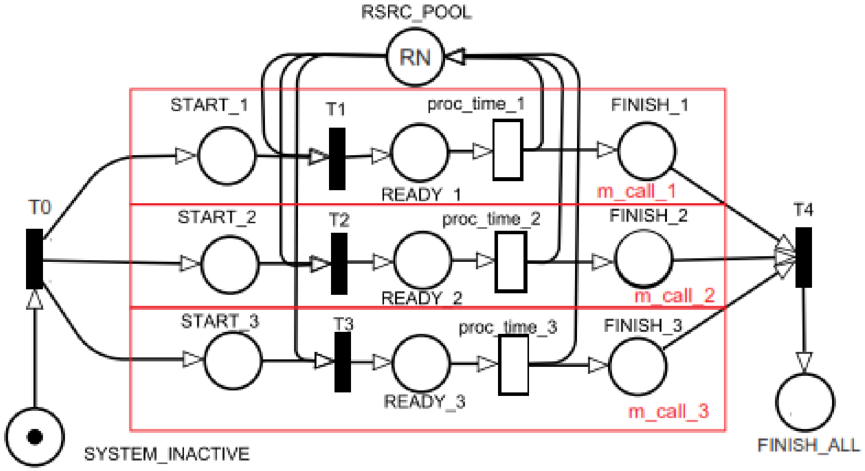
```
HeavyClass(){
  rootMethod(){
      registerStart();
      m1();
      registerEnd();

      registerStart();
      m2();
      registerEnd();

      registerStart();
      m3();
      registerEnd();
  }
}
```

```
HeavyClass(){
  rootMethod(){
      m1();
      m2();
      m3();
  }
}
```

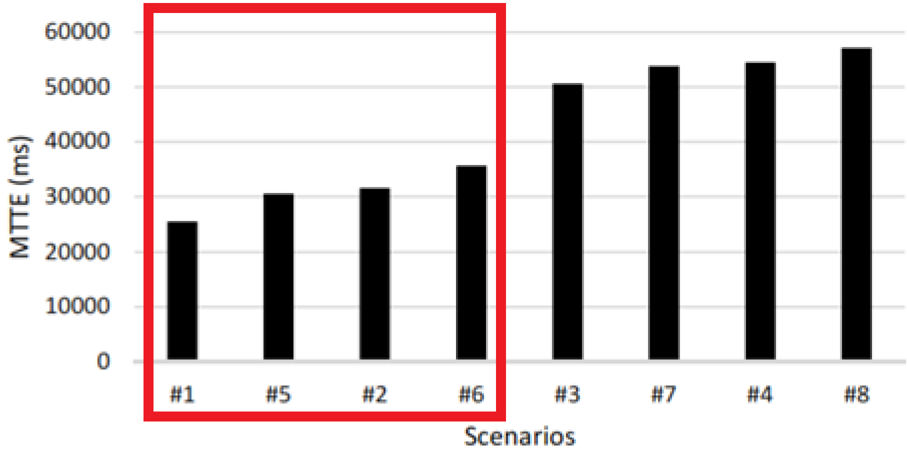




Case study: Planning Mobile Cloud Infrastructures Focusing on Partial Method-Call Offloading

Achieved results:

SLA = maximum MTTE around 36000 ms



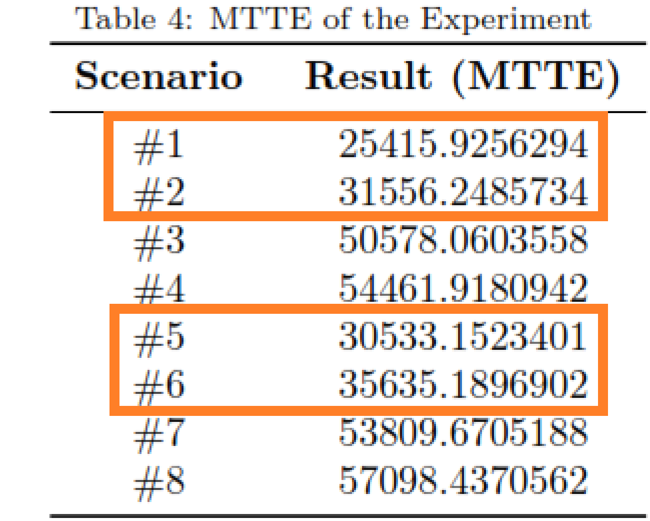
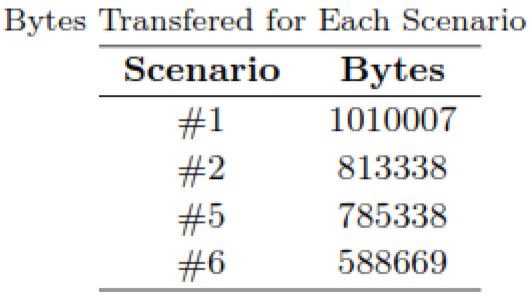
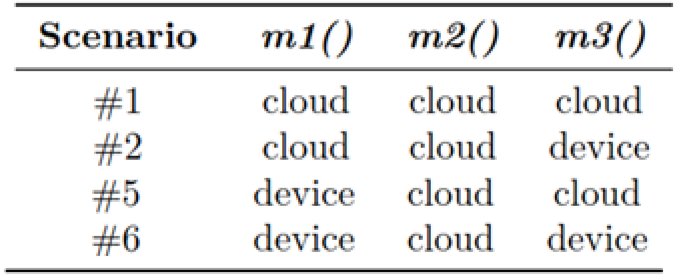


Table 4: MTTE of the Experiment

Scenario	Result (MTTE)
#1	25415.9256294
#2	31556.2485734
#3	50578.0603558
#4	54461.9180942
#5	30533.1523401
#6	35635.1896902
#7	53809.6705188
#8	57098.4370562


Bytes Transfered for Each Scenario

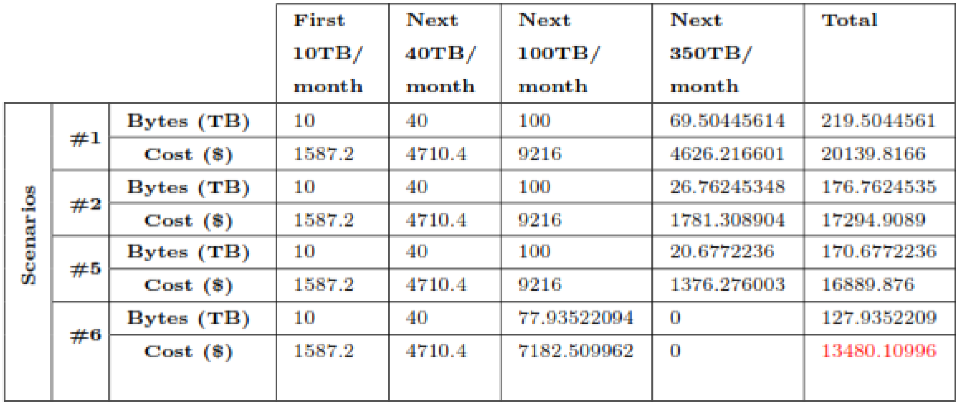
Scenario	Bytes
#1	1010007
#2	813338
#5	785338
#6	588669


Scenario	*m1()*	*m2()*	*m3()*
#1	cloud	cloud	cloud
#2	cloud	cloud	device
#5	device	cloud	cloud
#6	device	cloud	device

Case study: Planning Mobile Cloud Infrastructures Focusing on Partial Method-Call Offloading

Achieved results:

Table 8: Transfered Bytes and Respective Costs for Each Scenario



			First 10TB/ month	Next 40TB/ month	Next 100TB/ month	Next 350TB/ month	Total
Scenarios	#1	Bytes (TB)	10	40	100	69.50445614	219.5044561
		Cost ($)	1587.2	4710.4	9216	4626.216601	20139.8166
	#2	Bytes (TB)	10	40	100	26.76245348	176.7624535
		Cost ($)	1587.2	4710.4	9216	1781.308904	17294.9089
	#5	Bytes (TB)	10	40	100	20.6772236	170.6772236
		Cost ($)	1587.2	4710.4	9216	1376.276003	16889.876
	#6	Bytes (TB)	10	40	77.93522094	0	127.9352209
		Cost ($)	1587.2	4710.4	7182.509962	0	13480.10996

30000 active users in one month

Choosing #6 rather #1 allows a saving of :

US$ 6.659,71 / per month

US$ 79.916,52 / per year

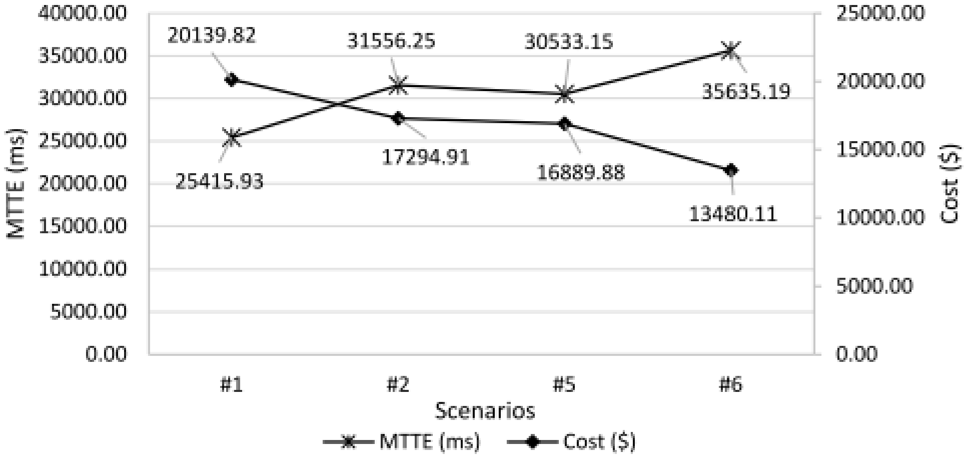
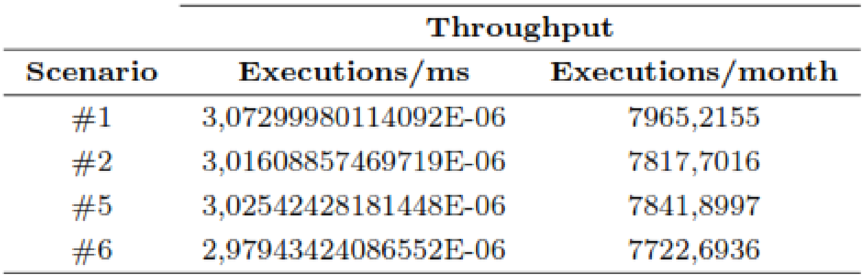


Figure 14: Comparing MTTE and Cost.

Table 7: Throughput for Each User in Each Scenario



	Throughput	
Scenario	Executions/ms	Executions/month
#1	3,07299980114092E-06	7965,2155
#2	3,01608857469719E-06	7817,7016
#5	3,02542428181448E-06	7841,8997
#6	2,97943424086552E-06	7722,6936

Thank You