



# Configuration Optimization for Big Data Software

Pooyan Jamshidi

*Imperial College London*

DICE

Horizon 2020 Research & Innovation Action

Grant Agreement no. 644869

<http://www.dice-h2020.eu>



Funded by the Horizon 2020  
Framework Programme of the European Union

# Configuration Optimization (WP5)



- T5.1; Lead: IMP
- Contributors: XLAB, IEAT
- Experimentally produce an **optimal configuration**
- Parameters *not* assigned by WP3 optimization

# Big Data Application Configuration



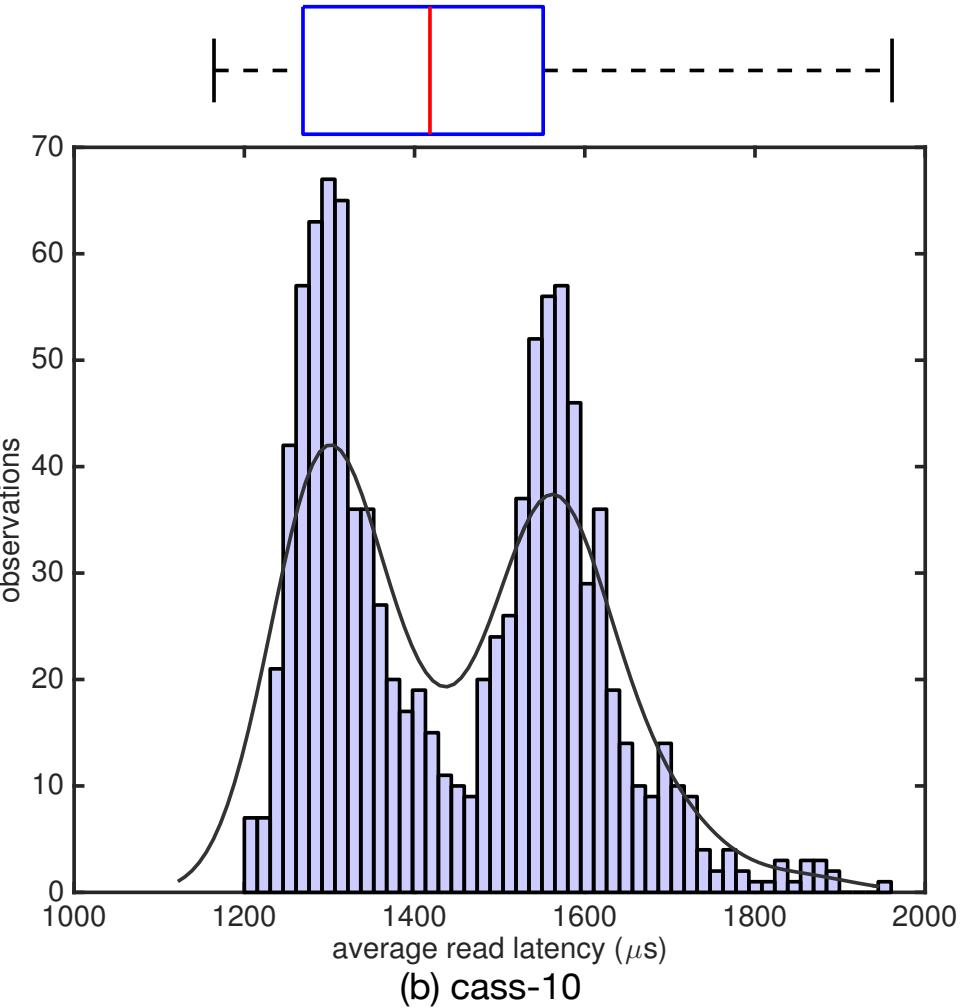
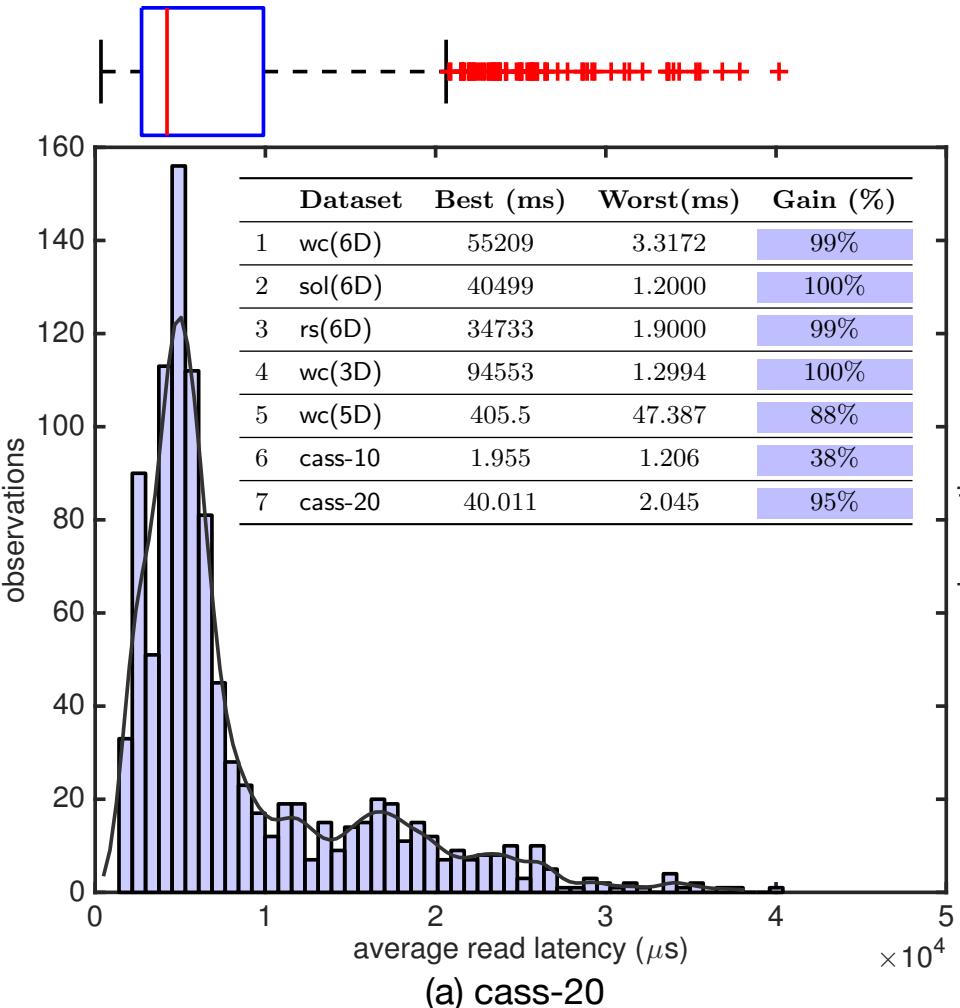
```
102 drpc.port: 3772
103 drpc.worker.threads: 64
104 drpc.max_buffer_size: 1048576
105 drpc.queue.size: 128
106 drpc.invocations.port: 3773
107 drpc.invocations.threads: 64
108 drpc.request.timeout.secs: 600
109 drpc.childopts: "-Xmx768m"
110 drpc.http.port: 3774
111 drpc.https.port: -1
112 drpc.https.keystore.password: ""
113 drpc.https.keystore.type: "JKS"
114 drpc.http.creds.plugin: org.apache.storm.security.auth.DefaultHttpCredentialsPlugin
115 drpc.authorizer.acl.filename: "drpc-auth-acl.yaml"
116 drpc.authorizer.acl.strict: false
117
118 transactional.zookeeper.root: "/transactional"
119 transactional.zookeeper.servers: null
120 transactional.zookeeper.port: null
121
122 ## blobstore configs
123 supervisor.blobstore.class: "org.apache.storm.blobstore.NimbusBlobStore"
124 supervisor.blobstore.download.thread.count: 5
125 supervisor.blobstore.download.max_retries: 3
126 supervisor.localizer.cache.target.size.mb: 10240
127 supervisor.localizer.cleanup.interval.ms: 600000
128
129
```



# Performance Gain



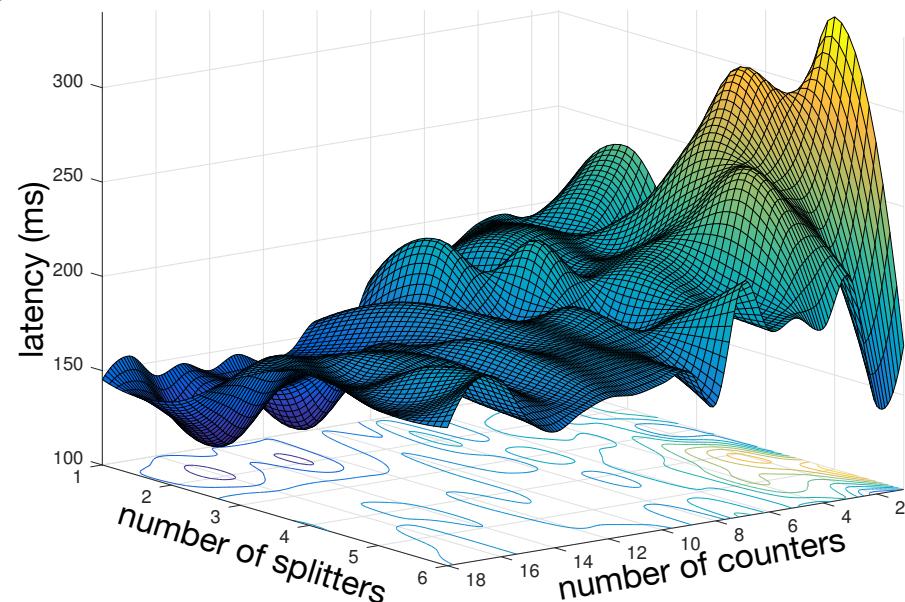
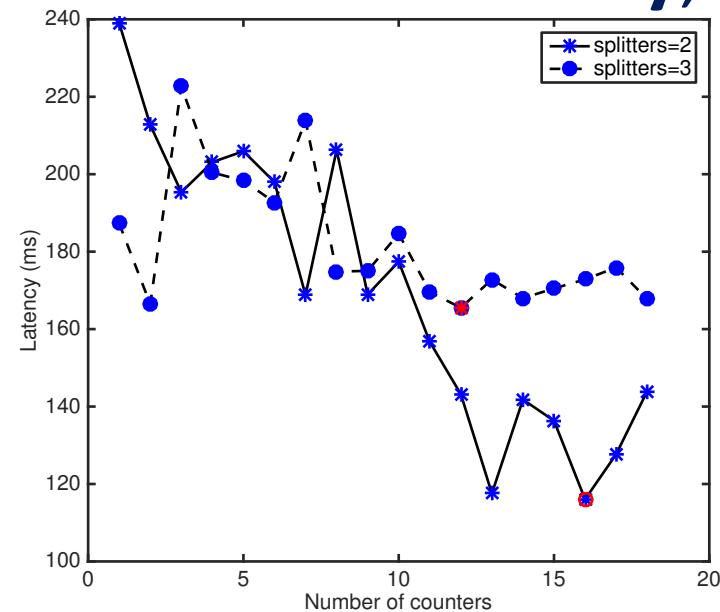
Different parameter settings cause very large variance in the performance of the system under test.



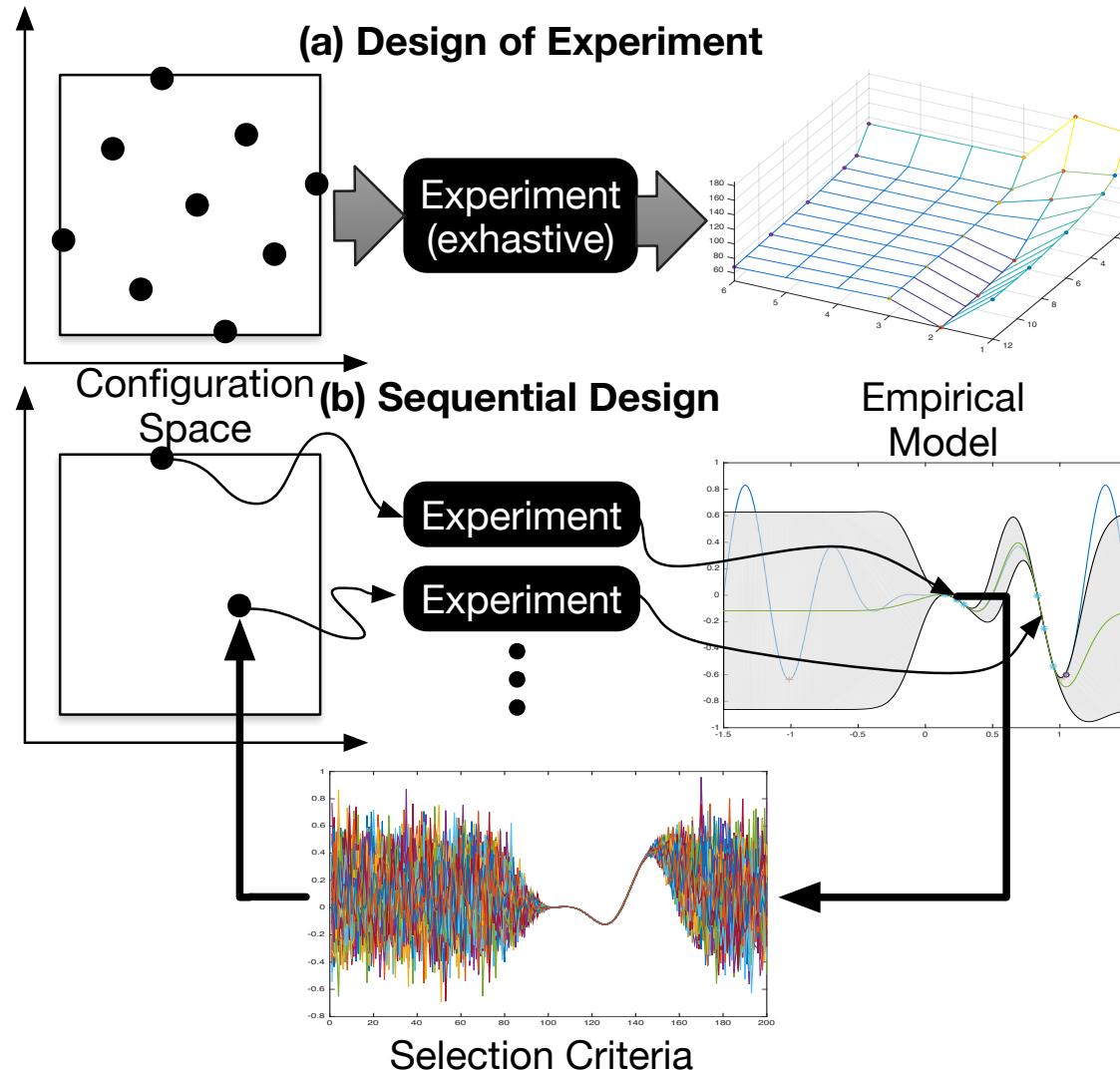
# Configuration Optimization features



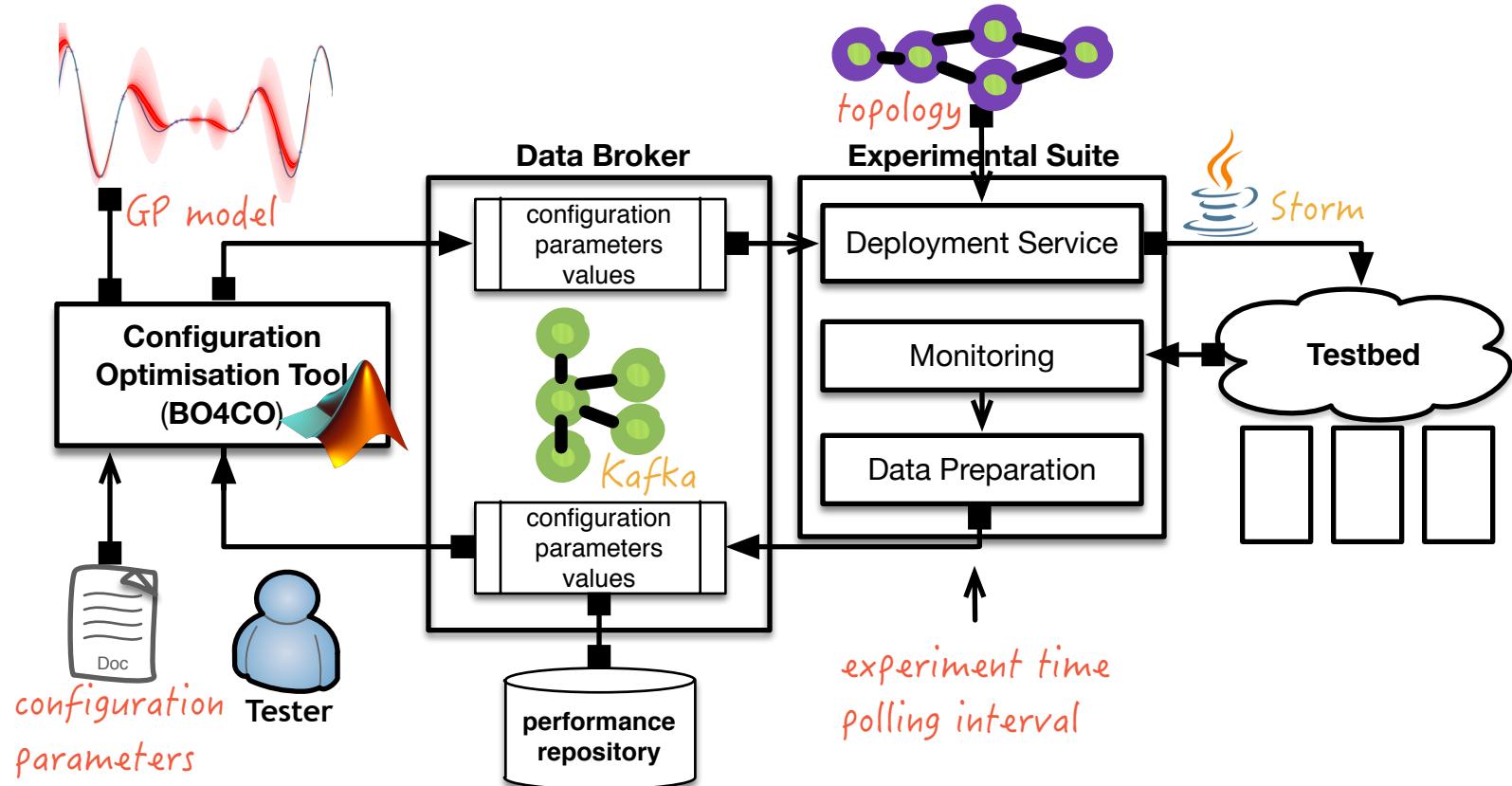
- A traditionally time-costly operation considerably **reduced in duration**
- Configurations get **more efficient with each build**
- Support for **Apache Storm** and **Cassandra**
- Metrics: **latency, throughput**



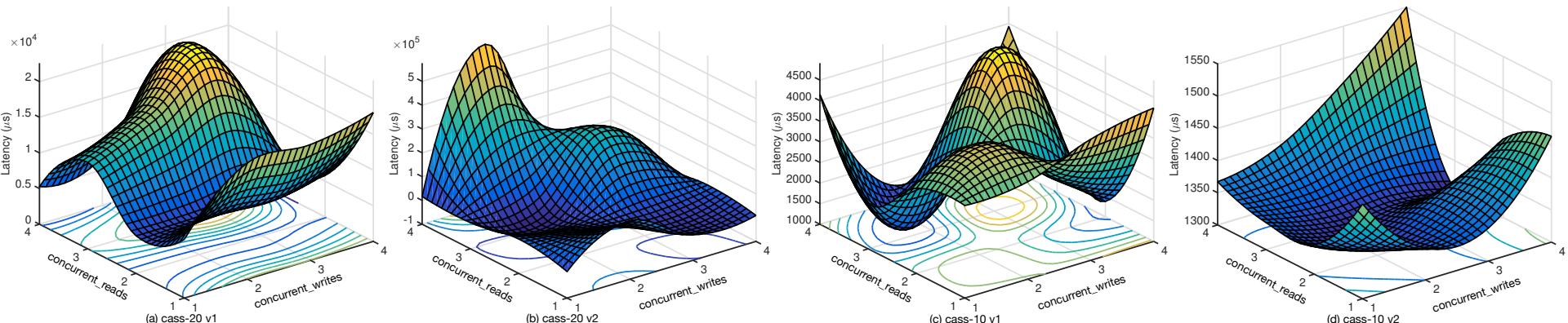
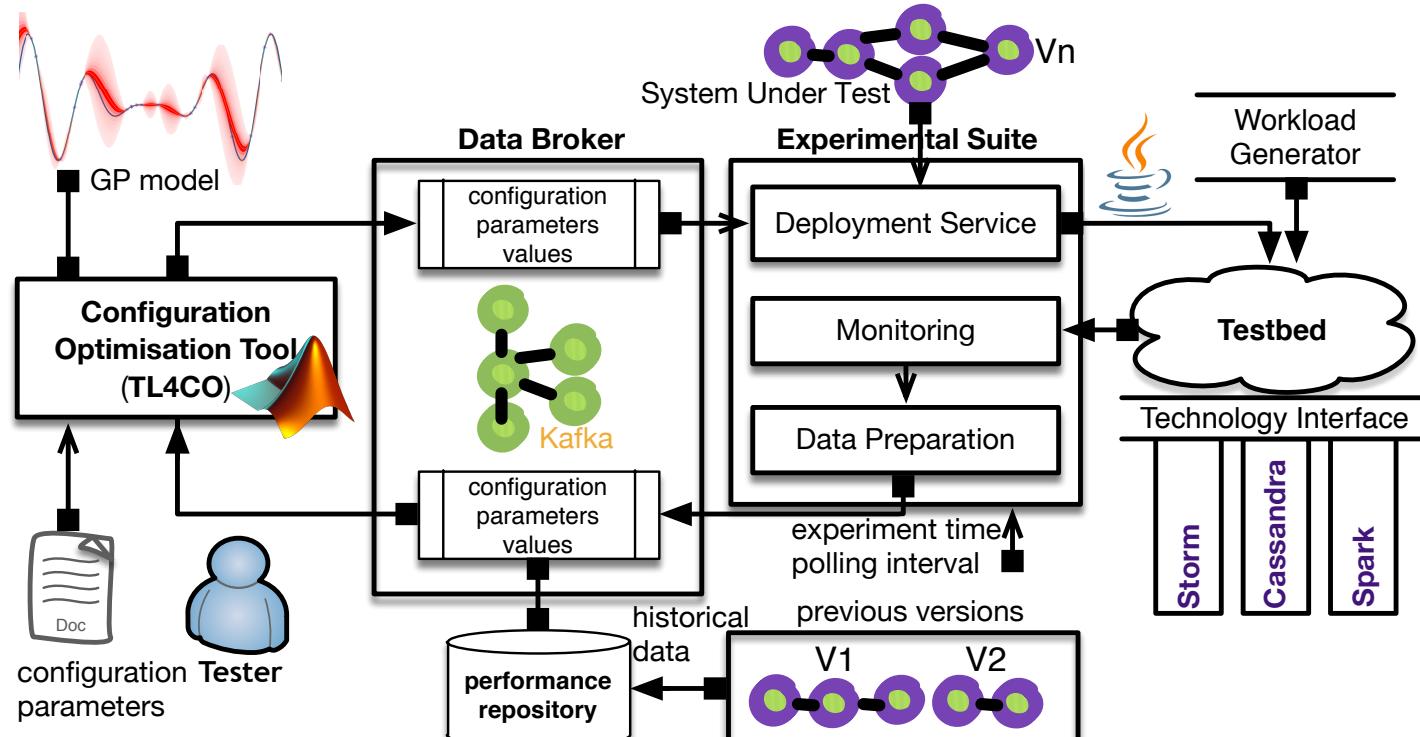
# Overview of our approach



# BO4CO Architecture



# TL4CO (DevOps compatible)



# Experiments (Storm+Cassandra)



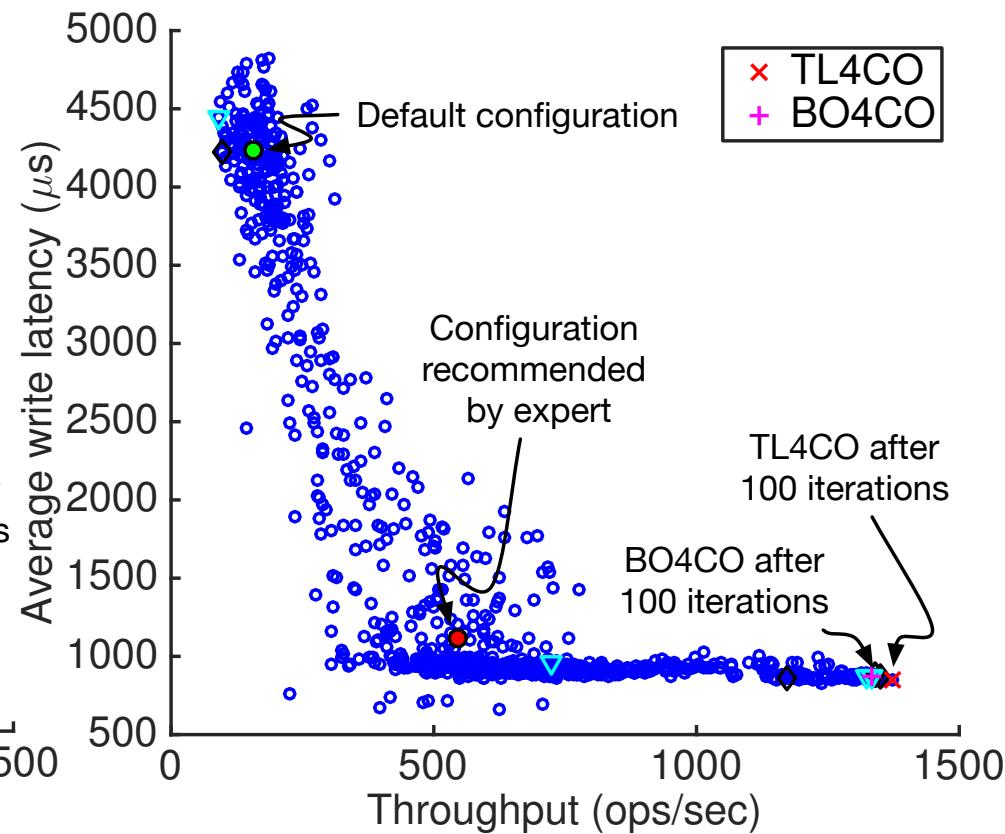
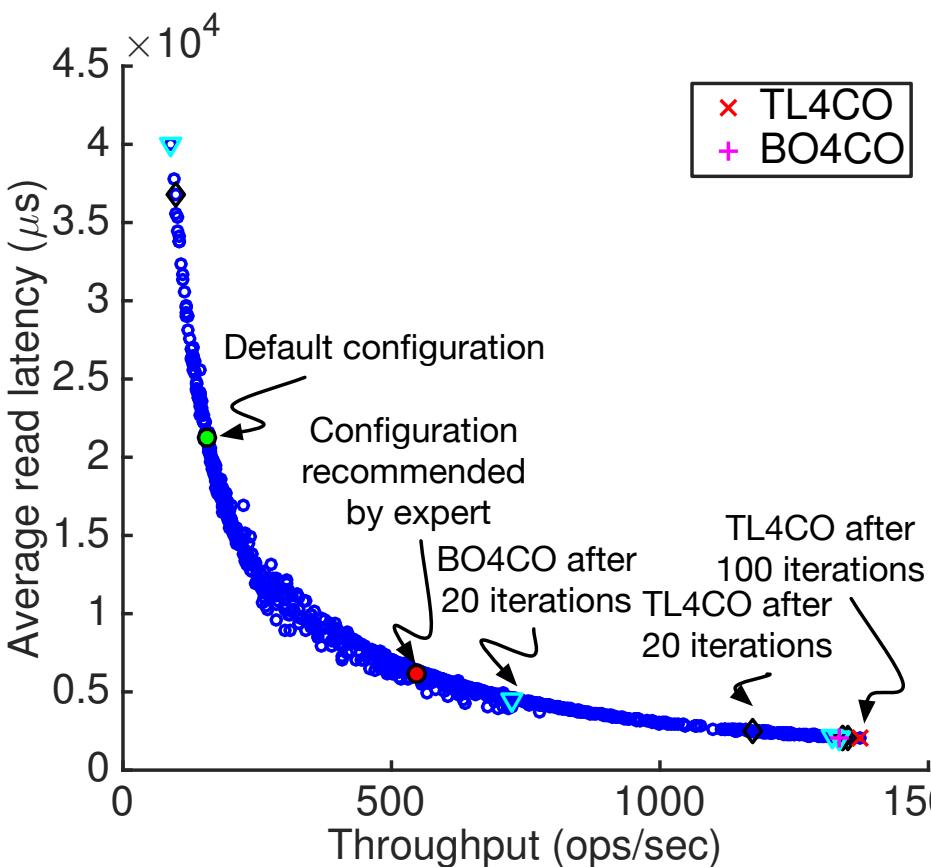
- Over 4 months (24/7) of experimental efforts
- on 6 different cloud platforms with 4 benchmark systems (3 Storm, 1 NoSQL)

	Dataset	Parameters	Size	Testbed
1	wc(6D)	1-spouts: {1,3}, 2-max_spout: {1,2,10,100,1000,10000}, 3-spout_wait: {1,2,3,10,100}, 4-splitters: {1,2,3,6}, 5-counters: {1,3,6,12}, 6-netty_min_wait: {10,100,1000}	2880	C1
2	sol(6D)	1-spouts: {1,3}, 2-max_spout: {1,10,100,1000,10000}, 3-top_level: {2,3,4,5}, 4-netty_min_wait: {10,100,1000}, 5-message_size: {10,100,1e3,1e4,1e5,1e6}, 6-bolts: {1,2,3,6}	2866	C2
3	rs(6D)	1-spouts: {1,3}, 2-max_spout: {10,100,1000,10000}, 3-sorters: {1,2,3,6,9,12,15,18}, 4-emit_freq: {1,10,60,120,300}, 5-chunk_size: {1e5,1e6,2e6,1e7}, 6-message_size: {1e3,1e4,1e5}	3840	C3
4	wc(3D)	1-max_spout: {1,10,100,1e3, 1e4,1e5,1e6}, 2-splitters: {1,2,3,4,5,6}, 3-counters: {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18}	756	C4
5	wc+rs	1-max_spout: {1,10,100,1e3, 1e4,1e5,1e6}, 2-splitters: {1,2,3,6}, 3-counters: {1,3,6,9,12,15,18}	196	C4
6	wc+sol	1-max_spout: {1,10,100,1e3, 1e4,1e5,1e6}, 2-splitters: {1,2,3,6}, 3-counters: {1,3,6,9,12,15,18}	196	C4
7	wc+wc	1-max_spout: {1,10,100,1e3, 1e4,1e5,1e6}, 2-splitters: {1,2,3,6}, 3-counters: {1,3,6,9,12,15,18}	196	C4
8	wc(5D)	1-spouts: {1,2,3}, 2-splitters: {1,2,3,6}, 3-counters: {1,2,3,6,9,12}, 4-buffer-size: {256k,1m,5m,10m,100m}, 5-heap: {"-Xmx512m", "-Xmx1024m", "-Xmx2048m"}	1080	C5
9	wc-c1	1-spout_wait: {1,2,3,4,5,6,7,8,9,10,100,1e3,1e4,1e4}, 2-splitters: {1,2,3,4,5,6}, 3-counters: {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18}	1343	C1
10	wc-c3	1-spout_wait: {1,2,3,4,5,6,7,8,9,10,100,1e3,1e4,1e4}, 2-splitters: {1,2,3,4,5,6}, 3-counters: {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18}	1512	C3
11	cass-10	1-trickle_fsync: {false,true}, 2-auto_snapshot: {false,true}, 3-con_reads: {16,24,32,40}, 4-con_writes: {16,24,32,40}, 5-file_cache_size_mb: {256,384,512,640}, 6-con_compactors: {1,3,5,7}	1024	C6
12	cass-20			

# How does a “default” setting compare to the worst and optimum configuration in terms of performance?



Default parameter settings perform poorly. The expert’s prescription is also far from optimal on individual problem instances.



Interested to know more about this?



please email us at  
[p.jamshidi@imperial.ac.uk](mailto:p.jamshidi@imperial.ac.uk)

Or

[g.casale@imperial.ac.uk](mailto:g.casale@imperial.ac.uk)