

Φυτώριο από το Μέλλον

Γεώργιος Τσατσάνιφος, Ph.D.,
Ηλεκτρονικός Μηχ/κός & Μηχ/κός Υπολογιστών

20 October 2017

Proposal number: HRL-20171020

EXECUTIVE SUMMARY

Εισαγωγή

Ως γνωστόν, τις τελευταίες δεκαετίες η αγορά βρίσκεται σε παγκόσμιο επίπεδο από μεγάλες εταιρείες λογισμικού και σχετικών υπηρεσιών σε πολλούς τομείς. Η φήμη και τα κέρδη που έχουν αποκομίσει είναι αδιαμφισβήτα παρά τις όποιες κάμψεις και πλήγματα έχουν στοιχίσει στον κλάδο, όπως για παράδειγμα η dot-com bubble όπου η ιλιγγιώδης άνοδος του τεχνολογικού δείκτη του αμερικανικού χρηματιστηρίου οδηγούμενη από την μόδα της εποχής οδήγησε σε μία μεγάλη κι άτακτη “κανονικοποίηση” του χώρου έπειτα. Σε αντιπαράθεση, την ίδια εποχή περίπου αρχίζει να γίνεται ευρέως γνωστό και το κίνημα του ανοιχτού λογισμικού ύστερα από την ανάμιξη αρκετών εταιριών, π.χ. Red Hat, Suse, αν και στην πραγματικότητα τις ρίζες του κινήματος του ελεύθερου λογισμικού τις εντοπίζει κανείς στη δεκαετία του '80 και τον Richard Stallman.

Πρόκειται για λογισμικό του οποίου ο πηγαίος κώδικας γίνεται διαθέσιμος στο ευρύ κοινό μαζί με το δικαίωμα για επέμβαση σε αυτόν, τον ελεύθερο διαμοιρασμό του αποτελέσματος, καθώς και την εκμετάλευσή του με οποιοδήποτε τρόπο δεν παραβιάζει τα προηγούμενα δικαιώματα, ήτοι δεν δύναται παράγωγο ελεύθερου λογισμικού να χαρακτηριστεί κλειστό και να γίνει προϊόν εκμετάλευσης κάποιου που δεν κάνει διαθέσιμο τον πηγαίο κώδικα. Πάνω σε ακριβώς αυτές τις αρχές μπορούν εταιρείες να πατήσουν και με ποικίλους τρόπους όπως (1) τροποποιώντας/βελτιώνοντας, (2) προσαρμόζοντας σε συγκεκριμένες ανάγκες κάποιου πελάτη, και (3) παρέχοντας υπηρεσίες, όπως εγκατάσταση και συντήρηση λειτουργικών συστημάτων, εφαρμογών, διακομιστών, κλειστών/ανοιχτών δικτυακών εφαρμογών του πελάτη σε ανοιχτού λογισμικού πλατφόρμες, κι ως αποτέλεσμα των δραστηριοτήτων αυτών να ανελιχθούν σε κερδοφόρες μονάδες.

Επιχειρηματικό Μοντέλο

Το παράδειγμα του ανοιχτού λογισμικού ακολουθεί κι η νεοσυσταθείσα “μικροσκοπική” start up ονόματι *HomeGrown Research*. Δεν πρόκειται όμως για ένα ποντίκι που ψάχνει να βρει τη θέση του ανάμεσα στους ελέφαντες. Η κάθετη εξειδίκευση των προϊόντων μας εξασφαλίζει τη μοναδικότητα του σήματός μας όσον αφορά την κατασκευή εφαρμογών κι ερευνητικών προϊόντων τα οποία δε διστάζουμε να τα συγκρίνουμε με τα αντίστοιχα άλλων μεγάλων μονάδων, ερευνητικών φορέων και πανεπιστημίων των οποίων οι μεγάλοι προϋπολογισμοί δεν δημιούργησαν τίποτε παραπάνω από μία τρύπα στο νερό, βλέπε #indexing από HomeGrown (<https://github.com/gtsat/indexing>) σε σχέση με το ChoroChronos του ΕΜΠ.

Πεδίο αναφοράς μας αποτελεί ο κλάδος που έχει γίνει γνωστός ως διαχείριση δεδομένων. Πρόκειται για έναν εξαιρετικά εφαρμοσμένο κλάδο της πληροφορικής, τεχνολογίας υπολογιστών κι εφαρμοσμένης έρευνας. Η εγκαθίδρυση κι επιτυχία μας “ποντάρει” σε συγκεκριμένες συμβάσεις που έχουμε κάνει για τον χώρο για διάστημα άνω της τελευταίας πενταετίας:

1. Σε αντίθεση με άλλους κλάδους των μηχανικών και της πληροφορικής που έχουν εξαιρετικά δυνατό θεωρητικό υπόβαθρο, όπως signal processing, μηχανική μάθηση, κ.α., στη διαχείριση δεδομένων τήνουν

να καταλήγουν λιγότερο καταρτισμένα άτομα που δεν θέλουν να εμπλέκονται είτε με μαθηματικά μοντέλα, είτε ιδιαίτερο διάβασμα, είτε ιδιαίτερη προσπάθεια εν γένει.

2. Η εν λόγω ανωτέρω ευκολία τους γίνεται πρόδηλη ακόμα και σε αυτά καθεαυτά τα κακοφτιαγμένα προϊόντα τους. Αυτός ακριβώς είναι ο λόγος που το industry στον τομέα αυτό καθοδηγεί την έρευνα, αντί να γίνεται το αντίστροφο με την εφαρμογή “φρέσκων” ιδεών. Η πιο πρόσφατη έκφανση της συγκεκριμένης παρατήρησης ήταν η εδραίωση του Map-Reduce ως μοντέλου επεξεργασίας δεδομένων.
3. Στην πρωτοφανή διαφθορά των ατόμων που έχουν εδραιωθεί και διοικούν τον τομέα και τα όργανά του σε διεθνές επίπεδο, όπως για παράδειγμα τα κανάλια ημιδιαφανών διαδικασιών που χρησιμοποιούνται για τη δημοσίευση καινοτόμων εργασιών. Σε πόσους κλάδους γνωρίζετε ανθρώπους του top-tier είτε να έχουν συλληφθεί για κατοχή κοκαΐνης, είτε επανειλημμένα οι εργασίες τους να αποτελούν προϊόν κακής αντιγραφής “χωρίς περίσκεψιν, χωρίς λύπην, χωρίς αιδώ” των οποίων η κοινοτοπία να αποδεικνύεται με περισσή ευκολία; Τρανταχτά παραδείγματα φανατικών του HomeGrown Research είναι οι βιβλιογραφίες των “αξιότιμων” κων. καθηγητών Νίκου Μαμουλή και Δημήτρη Σαχαρίδη, με τις πλάτες και τις ευλογίες των κων. Ι. Ιωαννίδη και του κου. Τ. Σελλή αντίστοιχα.
4. Στην ιδιαίτερα κοστοβόρα συντήρηση του προαναφερθέντος κατεστημένου. Εν αντιθέσει, ένα “καλοκουρδισμένο” πειρατικό πλοίο μπορεί να συντηρηθεί από το περιβάλλον στο οποίο λειτουργεί και να προκαλέσει ένα άξιο απορίας αποτέλεσμα με ένα εξαιρετικά χαμηλό προϋπολογισμό.
5. Στο ότι γνωρίζουμε προσωπικά τα άτομα που στελεχώνουν το χώρο αυτόν, τις δυνατότητες και τις αδυναμίες τους και στο ότι δεν είναι σε θέση να αναπαράγουν τα δικά μας αποτελέσματα. Πρόκειται για νεποτιστικής νοοτροπίας μασώνους που χρίζουν άτομα σε θέσεις με τον ίδιο τρόπο που γίνεται στη στοά για τα προσκυνημένα ασπόνδυλα. Η παρατήρηση αυτή μας εξασφαλίζει ότι τα ανωτέρω σημεία θα ταλαιπωρούν τον κλάδο εις το διηνεκές κι έτσι γίνεται αναγκαία κι απαραίτητη η ύπαρξη ενός τόσο καθαρού μοντέλου όσο του δικού μας.

Καινοτομία

Δύο είναι τα πρώτα προϊόντα στα οποία στηρίζεται το φυτώριο των οποίων τον πηγαίο κώδικα κάνει διαθέσιμο στο ευρύ κοινό ενώ παράλληλα μπορούμε να παρέχουμε υπηρεσίες πάνω σε αυτά ή καινούριες υλοποιήσεις σε άλλες γλώσσες προγραμματισμού και διαφορετικές άδειες λογισμικού έναντι προσυμφωνημένου αντίτιμου. Λόγω περιορισμών στο μήκος της εργασίας αυτής θα αναφερθούμε στο πιο πρόσφατο εκ των HomeGrown προϊόντων και μόνο.

Το πακέτο #indexing λοιπόν, χρησιμοποιείται για την αποθήκευση σε blocks μεταβλητού μεγέθους κι ανάκτηση πολυδιάστατων μεγάλου όγκου δεδομένων στο σκληρό δίσκο συνοδευόμενο τόσο με στάνταρ λειτουργικότητα αλλά και τελεστές που δεν υπάρχουν σε ανταγωνιστικά πακέτα. Επιπλέον, επηρεασμένοι από τις διεπαφές των middlewares στη βιομηχανία έχουμε κατασκευάσει (1) μία RESTful query language, καθώς και (2) JSON-based Data Manipulation Language (DML) statements για τη διαχείριση των δεδομένων στο σκληρό δίσκο, προκειμένου να μπορεί να ικανοποιείται live κίνηση αλλά και να ενημερώνεται η τράπεζα δεδομένων στο δίσκο για τις όποιες δυναμικές αλλαγές.

Έστω για παράδειγμα ότι θέλουμε να δημιουργήσουμε ένα μεγάλο heapfile από δισδιάστατα δεδομένα που αντιστοιχούν στο οδικό δίκτυο των ΗΠΑ. Ο τρόπος που γίνεται αυτό φαίνεται στην παρακάτω εικόνα, όπου

HOMEGROWN RESEARCH LABS

δίνοντας ως είσοδο το αρχικό αρχείο με τα δεδομένα, τον αριθμό των διαστάσεων και το επιθυμητό μέγεθος των blocks, δημιουργούμε στον σκληρό δίσκο μία δενδρική δομή σε binary format.

```
gtsat@mackenzie:~/HomeGrown/github/src> ./create#rtree

Copyright (C) 2016 George Tsatsanifos <gtsatsanifos@gmail.com>

#indexing comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to redistribute it under certain conditions.

** ERROR - Please specify the dimensionality of the domain...
** ERROR - Please specify a block-size...
** ERROR - Please specify a dataset to be indexed...
** ERROR - Please specify a filepath for the produced heapfile...
** Usage:
    ./create#rtree [option] [parameter]
    -d --dims      : The number of dimensions.
    -b --block     : The desired size of each block.
    -a --dataset   : The path to the datafile.
    -t --tree      : The path to the binary heap-file.
gtsat@mackenzie:~/HomeGrown/github/src> time ./create#rtree -d 2 -b 4096 -a ../../data/dimacs/USA-road-d.USA.co -t USA.b4096.rtree

Copyright (C) 2016 George Tsatsanifos <gtsatsanifos@gmail.com>

#indexing comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to redistribute it under certain conditions.

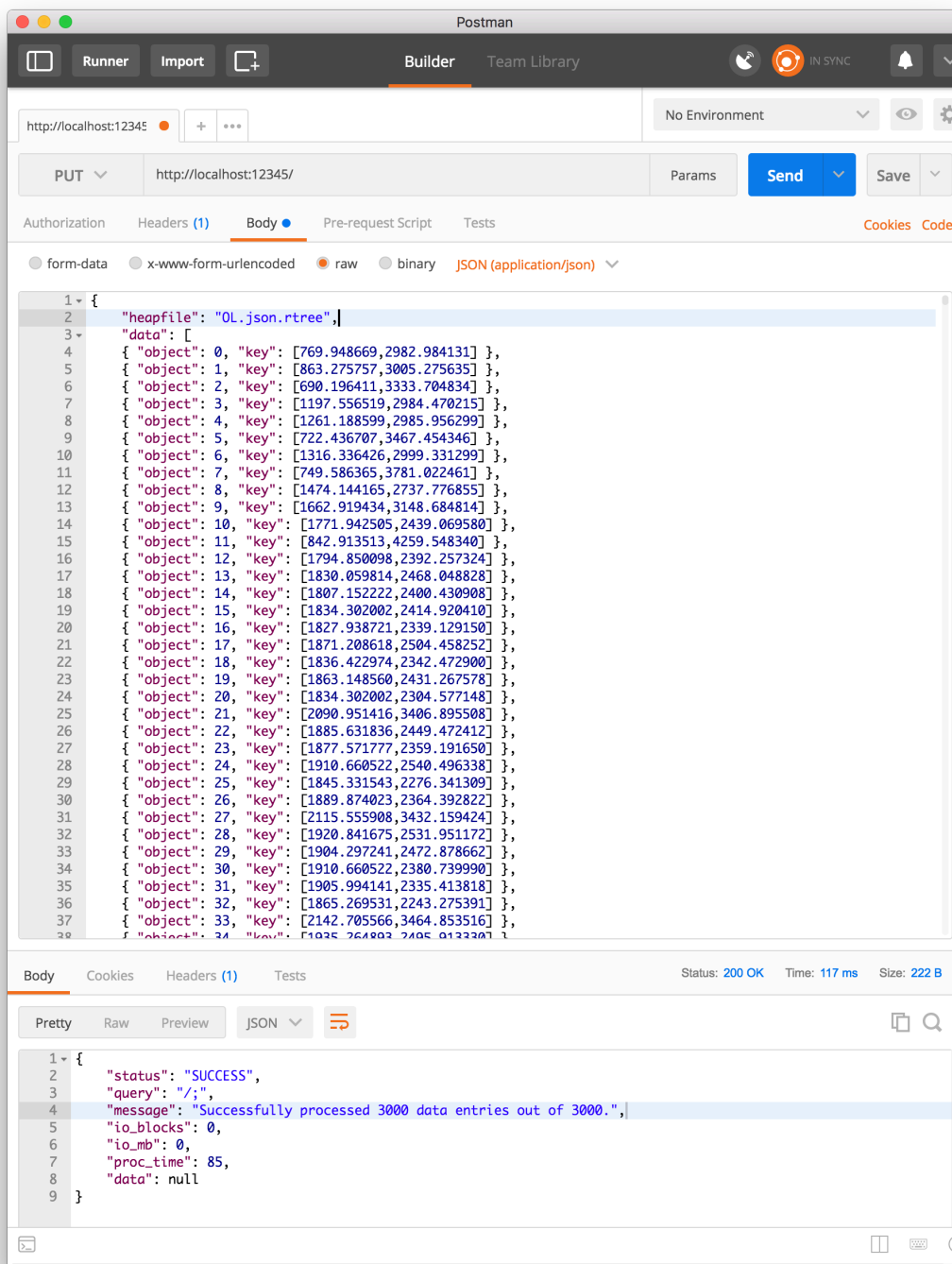
** WARNING - [USA.b4096.rtree] Cannot open file 'USA.b4096.rtree' for reading...
** WARNING - [USA.b4096.rtree] Now flushing tree hierarchy. Overall 23947347 entries are indexed overall.
** WARNING - [USA.b4096.rtree] Done flushing tree hierarchy. Overall 96678 dirty blocks were found!

real    37m3.688s
user    36m42.802s
sys      0m14.918s
gtsat@mackenzie:~/HomeGrown/github/src> du -hs ../../heapfiles/USA.b*
1.6G    ../../heapfiles/USA.b1024.rtree
929M    ../../heapfiles/USA.b2048.rtree
9.7G    ../../heapfiles/USA.b256.rtree
1.1G    ../../heapfiles/USA.b4096.rtree
2.9G    ../../heapfiles/USA.b512.rtree
gtsat@mackenzie:~/HomeGrown/github/src>
```

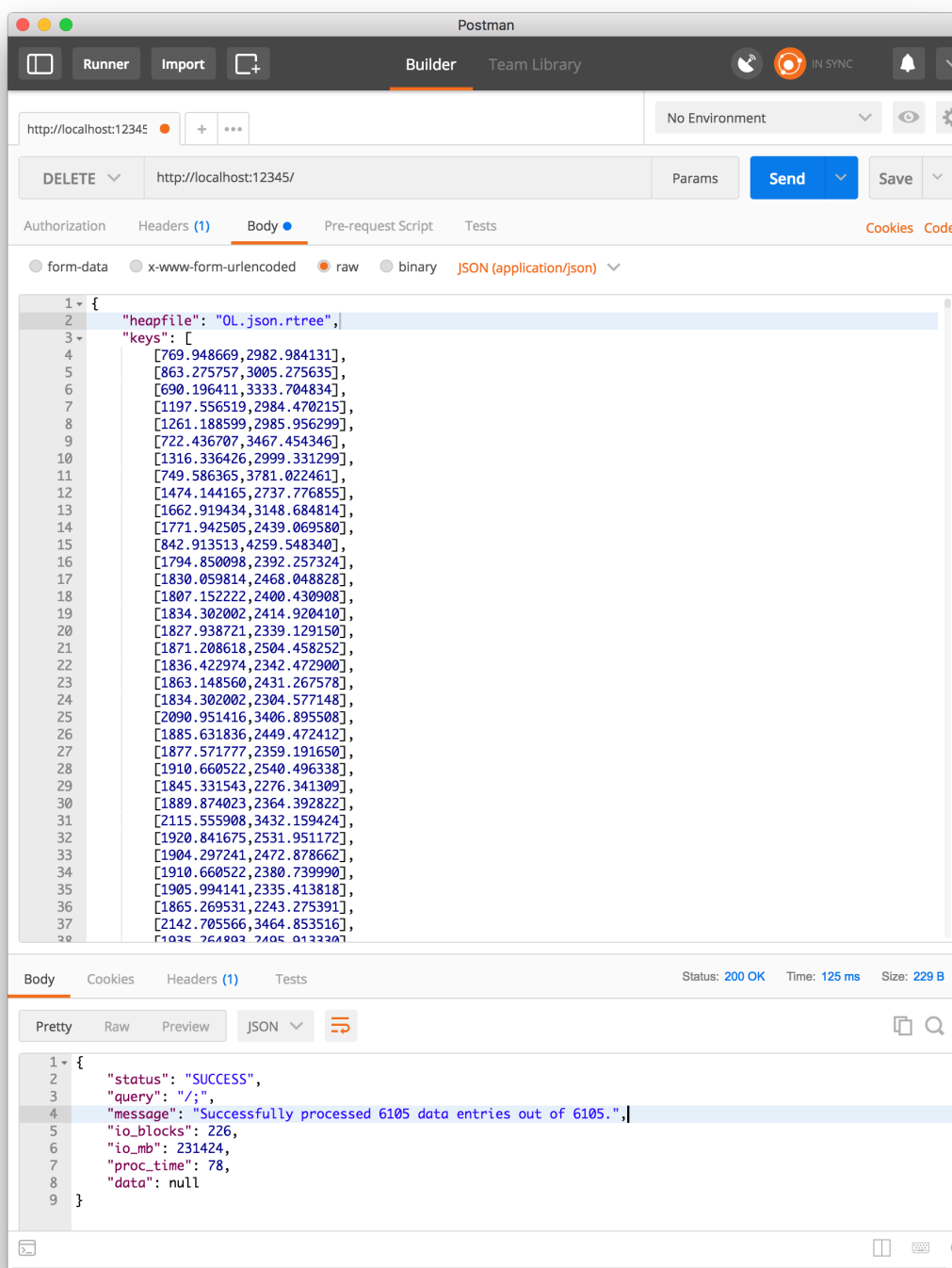
Προσδιορίζοντας εν συνεχεία στον #server έναν οποιοδήποτε φάκελο ο οποίος εμπεριέχει heapfiles μπορούμε να διατυπώσουμε ερωτήματα σε μορφή HTTP GET requests με τον ακόλουθο τρόπο για να πάρουμε τα 15 κοντινότερα ζεύγη κόμβων από δύο ξεχωριστά οδικά δίκτυα που αντιστοιχούν σε αυτά των πολιτειών στη δυτική κι ανατολική ακτή αντίστοιχα, όπως δείχνουμε παρακάτω:

```
Postman
Runner Import Builder Team Library
http://localhost:12345 GET http://localhost:12345/WEST.b4096.rtree/EAST.b4096.rtree/15
No Environment Params Send Save
Pretty Raw Preview
{
  "data": [
    {
      "rid": 0, "objects": [2723139, 3334592], "keys": [[-100000000.000000, 40365616.000000], [-78999752.000000, 40412552.000000]], "mindistance_ordered": 21000300.451413, "mindistance_pairwise": 21000300.451413, "avgdistance_ordered": 21000300.451413, "avgdistance_pairwise": 21000300.451413, "maxdistance_ordered": 21000300.451413, "maxdistance_pairwise": 21000300.451413 },
      {
        "rid": 1, "objects": [2723139, 3334588], "keys": [[-100000000.000000, 40365616.000000], [-78999752.000000, 40412552.000000]], "mindistance_ordered": 21000298.397822, "mindistance_pairwise": 21000298.397822, "avgdistance_ordered": 21000298.397822, "avgdistance_pairwise": 21000298.397822, "maxdistance_ordered": 21000298.397822, "maxdistance_pairwise": 21000298.397822 },
      {
        "rid": 2, "objects": [3804712, 1467966], "keys": [[-1000000040.000000, 35241000.000000], [-78999928.000000, 35157072.000000]], "mindistance_ordered": 21000290.116977, "mindistance_pairwise": 21000290.116977, "avgdistance_ordered": 21000290.116977, "avgdistance_pairwise": 21000290.116977, "maxdistance_ordered": 21000290.116977, "maxdistance_pairwise": 21000290.116977 },
      {
        "rid": 3, "objects": [3804712, 422569], "keys": [[-100000048.000000, 35241000.000000], [-78999848.000000, 35180628.000000]], "mindistance_ordered": 21000288.512894, "mindistance_pairwise": 21000288.512894, "avgdistance_ordered": 21000288.512894, "avgdistance_pairwise": 21000288.512894, "maxdistance_ordered": 21000288.512894, "maxdistance_pairwise": 21000288.512894 },
      {
        "rid": 4, "objects": [2599005, 2060376], "keys": [[-100000024.000000, 3716624.000000], [-78999736.000000, 37169768.000000]], "mindistance_ordered": 21000288.235348, "mindistance_pairwise": 21000288.235348, "avgdistance_ordered": 21000288.235348, "avgdistance_pairwise": 21000288.235348, "maxdistance_ordered": 21000288.235348, "maxdistance_pairwise": 21000288.235348 },
      {
        "rid": 5, "objects": [3288771, 2853783], "keys": [[-100000000.000000, 38079084.000000], [-78999800.000000, 38063616.000000]], "mindistance_ordered": 21000280.996059, "mindistance_pairwise": 21000280.996059, "avgdistance_ordered": 21000280.996059, "avgdistance_pairwise": 21000280.996059, "maxdistance_ordered": 21000280.996059, "maxdistance_pairwise": 21000280.996059 },
      {
        "rid": 6, "objects": [3288770, 2853783], "keys": [[-1000000040.000000, 38072082.000000], [-78999800.000000, 38063616.000000]], "mindistance_ordered": 21000280.048647, "mindistance_pairwise": 21000280.048647, "avgdistance_ordered": 21000280.048647, "avgdistance_pairwise": 21000280.048647, "maxdistance_ordered": 21000280.048647, "maxdistance_pairwise": 21000280.048647 },
      {
        "rid": 7, "objects": [4461243, 3318828], "keys": [[-100000056.000000, 40355800.000000], [-78999968.000000, 40280880.000000]], "mindistance_ordered": 21000221.856165, "mindistance_pairwise": 21000221.856165, "avgdistance_ordered": 21000221.856165, "avgdistance_pairwise": 21000221.856165, "maxdistance_ordered": 21000221.856165, "maxdistance_pairwise": 21000221.856165 },
      {
        "rid": 8, "objects": [4461243, 3318827], "keys": [[-100000056.000000, 40355800.000000], [-78999968.000000, 40281572.000000]], "mindistance_ordered": 21000219.396823, "mindistance_pairwise": 21000219.396823, "avgdistance_ordered": 21000219.396823, "avgdistance_pairwise": 21000219.396823, "maxdistance_ordered": 21000219.396823, "maxdistance_pairwise": 21000219.396823 },
      {
        "rid": 9, "objects": [2723139, 3318828], "keys": [[-100000000.000000, 40365616.000000], [-78999968.000000, 40280880.000000]], "mindistance_ordered": 21000202.955941, "mindistance_pairwise": 21000202.955941, "avgdistance_ordered": 21000202.955941, "avgdistance_pairwise": 21000202.955941, "maxdistance_ordered": 21000202.955941, "maxdistance_pairwise": 21000202.955941 },
      {
        "rid": 10, "objects": [2723139, 3318827], "keys": [[-1000000040.000000, 40355800.000000], [-78999968.000000, 40281572.000000]], "mindistance_ordered": 21000200.175116, "mindistance_pairwise": 21000200.175116, "avgdistance_ordered": 21000200.175116, "avgdistance_pairwise": 21000200.175116, "maxdistance_ordered": 21000200.175116, "maxdistance_pairwise": 21000200.175116 },
      {
        "rid": 11, "objects": [4460591, 3318828], "keys": [[-100000040.000000, 40351084.000000], [-78999968.000000, 40280880.000000]], "mindistance_ordered": 21000189.346927, "mindistance_pairwise": 21000189.346927, "avgdistance_ordered": 21000189.346927, "avgdistance_pairwise": 21000189.346927, "maxdistance_ordered": 21000189.346927, "maxdistance_pairwise": 21000189.346927 },
      {
        "rid": 12, "objects": [4460591, 3318827], "keys": [[-100000040.000000, 40351084.000000], [-78999968.000000, 40281572.000000]], "mindistance_ordered": 21000187.044961, "mindistance_pairwise": 21000187.044961, "avgdistance_ordered": 21000187.044961, "avgdistance_pairwise": 21000187.044961, "maxdistance_ordered": 21000187.044961, "maxdistance_pairwise": 21000187.044961 },
      {
        "rid": 13, "objects": [4461242, 3318828], "keys": [[-100000000.000000, 40360828.000000], [-78999968.000000, 40280880.000000]], "mindistance_ordered": 21000184.182138, "mindistance_pairwise": 21000184.182138, "avgdistance_ordered": 21000184.182138, "avgdistance_pairwise": 21000184.182138, "maxdistance_ordered": 21000184.182138, "maxdistance_pairwise": 21000184.182138 },
      {
        "rid": 14, "objects": [4461242, 3318827], "keys": [[-100000000.000000, 40360828.000000], [-78999968.000000, 40281572.000000]], "mindistance_ordered": 21000181.559086, "mindistance_pairwise": 21000181.559086, "avgdistance_ordered": 21000181.559086, "avgdistance_pairwise": 21000181.559086, "maxdistance_ordered": 21000181.559086, "maxdistance_pairwise": 21000181.559086 }
    ]
  },
  "status": "SUCCESS",
  "query": "WEST.b4096.rtree/EAST.b4096.rtree/15;",
  "message": "Successful operation.",
  "io_blocks": 65,
  "io_mb": 0.254,
  "proc_time": 361
}
```

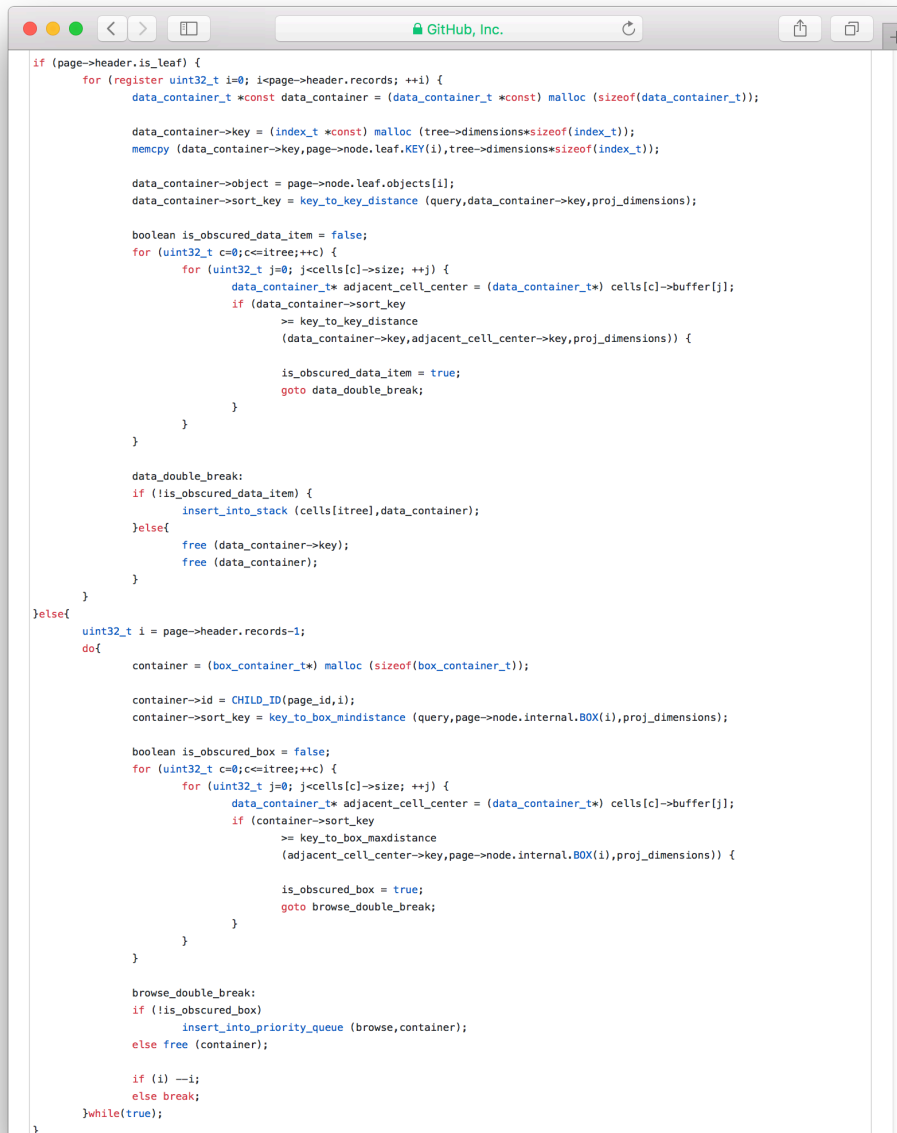
Υπενθυμίζουμε ότι οι αποκρίσεις του #server είναι σε JSON μορφή, ένα εξαιρετικά δημοφιλές πρότυπο για την επικοινωνία συστημάτων, κι άρα μπορούν να καταναλωθούν και να επεξεργασθούν περαιτέρω από άλλες εφαρμογές για οιοδήποτε σκοπό και τρόπο, π.χ. λόγω της φύσης των συγκεκριμένων δισδιάστατων δεδομένων, την απεικόνισή τους σε έναν browser μέσω δημοφιλών APIs όπως είναι αυτά των GoogleMaps και OpenStreetMaps. Αν τώρα όμως θέλουμε να προσθέσουμε δυναμικά μερικούς κόμβους ακόμα στην ίδια τράπεζα δεδομένων ή και σε διαφορετική θα το κάνουμε on-line στέλνοντας στον #server ένα HTTP PUT request του οποίου το σώμα είναι διαμορφωμένο σε JSON μορφή με τον παρακάτω τρόπο για να λάβουμε την απόκριση επιτυχίας ή αποτυχίας του αιτήματος και να τη διαχειριστούμε αναλόγως.



Κατ' αντιστοιχίαν, προκειμένου να αφαιρέσει μία εφαρμογή δυναμικά δεδομένα από την τράπεζα δεδομένων στο σκληρό δίσκο θα πρέπει να χρησιμοποιήσουμε το παρακάτω παράδειγμα αιτήματος ως ένα HTTP DELETE request, όπου η σύγχρονη απόκριση του συστήματος επιτυχίας ή αστοχίας διαχειρίζεται από την εφαρμογή που κάνει χρήση των #indexing υποδομών που έχουμε κατασκευάσει ως HomeGrown Research.



Επιπλέον, μεταξύ των υποστηριζόμενων τελεστών που υποστηρίζονται αυτή τη στιγμή από την προτεινόμενη γλώσσα ονόματι #QL είναι (1) πολυδιάστατα ερωτήματα εύρους, (2) κοντινότερων γειτόνων για τον ίδιο ή και μικρότερο αριθμό διαστάσεων από αυτόν της τράπεζας δεδομένων, (3) εύρεσης των σημείων από διαφορετικές τράπεζες δεδομένων που βρίσκονται εντός συγκεκριμένης απόστασης, (4) εύρεση των σημείων που βρίσκονται σε μεγαλύτερη απόσταση από αυτήν που προσδιορίζεται στο ερώτημα, (5) τα k κοντινότερα σημεία εκ διαφορετικών heapfiles, (6) τα k μακρύτερα, (7) ερωτήματα αντίστροφων κοντινότερων γειτόνων για αυθαίρετο αριθμό από heapfiles, (8) ερωτήματα κορυφογραμμής, ενώ χωρίς να συγκαταλέγονται ακόμα στη γλώσσα επερωτήσεων #QL σε αυτή τη φάση, παρέχονται προγραμματιστικά APIs για (9) ερωτήματα κορυφογραμμής από διαφορετικά domains, (10) και τέλος, για διαφόρους τύπους result diversification ως προς το περιεχόμενο. Ολόκληρη η υλοποίηση είναι ανοιχτού κώδικα και βρίσκεται on-line στο github για όποιον ενδιαφέρεται με χρήση άδειας λογισμικού GNU Public License (GPL) version 3, ως δείχνουμε παρακάτω στιγμιότυπο από το github repository.

A screenshot of a web browser displaying a GitHub repository. The browser's address bar shows 'GitHub, Inc.' and the page contains C code. The code is a function that processes query results, likely for a spatial query engine. It includes logic for allocating memory for data containers, calculating distances between points, and managing a stack of results. The code is written in C and uses standard data types like 'uint32_t' and 'boolean'. It also includes comments in Greek, such as 'is_obscured_data_item' and 'is_obscured_box'. The code is organized into several blocks, with some parts enclosed in 'do-while' loops and 'goto' statements for control flow. The overall structure suggests a complex data processing pipeline for handling spatial queries on a distributed dataset.

```
if (page->header.is_leaf) {
    for (register uint32_t i=0; i<page->header.records; ++i) {
        data_container_t *const data_container = (data_container_t *) malloc (sizeof(data_container_t));

        data_container->key = (index_t *) malloc (tree->dimensions*sizeof(index_t));
        memcpy (data_container->key, page->node.leaf.KEY(i), tree->dimensions*sizeof(index_t));

        data_container->object = page->node.leaf.objects[i];
        data_container->sort_key = key_to_key_distance (query, data_container->key, proj_dimensions);

        boolean is_obscured_data_item = false;
        for (uint32_t c=0; c<=itree;++c) {
            for (uint32_t j=0; j<cells[c]->size; ++j) {
                data_container_t* adjacent_cell_center = (data_container_t*) cells[c]->buffer[j];
                if (data_container->sort_key
                    >= key_to_key_distance
                        (data_container->key, adjacent_cell_center->key, proj_dimensions)) {

                    is_obscured_data_item = true;
                    goto data_double_break;
                }
            }
        }

        data_double_break:
        if (!is_obscured_data_item) {
            insert_into_stack (cells[itree], data_container);
        } else {
            free (data_container->key);
            free (data_container);
        }
    }
} else {
    uint32_t i = page->header.records-1;
    do {
        container = (box_container_t*) malloc (sizeof(box_container_t));

        container->id = CHILD_ID(page_id, i);
        container->sort_key = key_to_box_mindistance (query, page->node.internal.BOX(i), proj_dimensions);

        boolean is_obscured_box = false;
        for (uint32_t c=0; c<=itree;++c) {
            for (uint32_t j=0; j<cells[c]->size; ++j) {
                data_container_t* adjacent_cell_center = (data_container_t*) cells[c]->buffer[j];
                if (container->sort_key
                    >= key_to_box_maxdistance
                        (adjacent_cell_center->key, page->node.internal.BOX(i), proj_dimensions)) {

                    is_obscured_box = true;
                    goto browse_double_break;
                }
            }
        }

        browse_double_break:
        if (!is_obscured_box)
            insert_into_priority_queue (browse, container);
        else free (container);

        if (i) --i;
        else break;
    } while(true);
}
```

BUDGET

“Money is a terrible master but an excellent servant.” — P.T. Barnum

Τέλος, παρουσιάζουμε μία συνοπτική εικόνα του προϋπολογισμού του HomeGrown Research Lab όπως αυτός έχει διαμορφωθεί έως σήμερα. Υπενθυμίζουμε ότι πρόκειται για μερικής απασχόλησης επιχείρηση φυτώριο.

Description	Quantity	Unit Price	Cost
Salaries	21	€ 1.000	€ 21.000
Tenancy costs & related expenses	21	€ 400	€ 8.400
Promotion - Website costs	1	€ 400	€ 400
Hardware/Office equipment	1	€ 2.000	€ 2.000
Total			€ 31.800
