https://mlab.com/databases/interiarprototype

Run some kinda mongoDBtest

https://www.youtube.com/watch?v=8GRUwftKAps

Nick said something about trying to just throw in like a million users into it and see if it works??

Can’t remember what else he said…

<https://stackoverflow.com/questions/6783212/how-to-load-100-million-records-into-mongodb-with-scala-for-performance-testing> on inserting random stuff.

Mongo bulk insert should be used I guess.

Adding 10k tiny documents with a for loop not using bulk insert took about 2 min 30 seconds and used about 800KB

Using **var** bulk = db.items.initializeUnorderedBulkOp();

bulk.insert( { item: "abc123", defaultQty: 100, status: "A", points: 100 } );

bulk.insert( { item: "ijk123", defaultQty: 200, status: "A", points: 200 } );

bulk.insert( { item: "mop123", defaultQty: 0, status: "P", points: 0 } );

bulk.execute();

from <https://docs.mongodb.com/manual/reference/method/Bulk.insert/> did 10 thousand basically instantly.

One million documents using this method took about 35 seconds and used about 77MB.

Swapped to using python, see youtube link at top and in MiscTechStuff on googledrive the python file. However with this python method 1 million files took about 2 mins 20 and used about 140MB.

NOTE IT MAYBE JUST TOOK LONGER BECAUSE the documents had more information??

With comparable info it still took about 1 min 45.