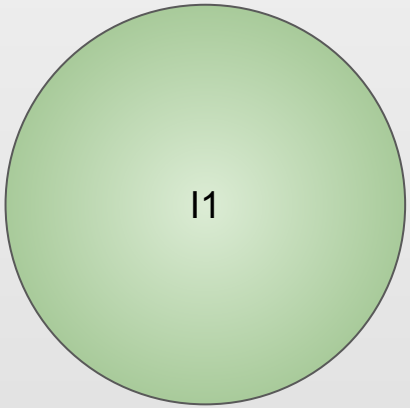
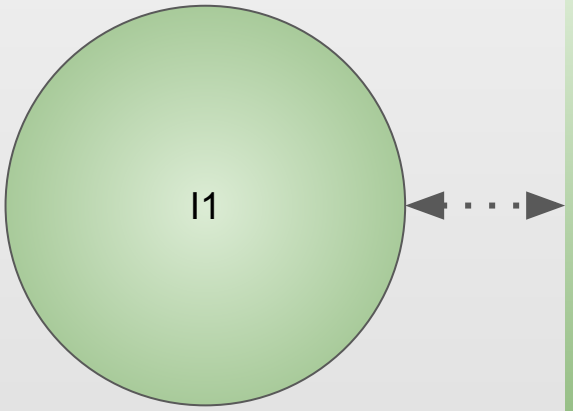
Memory Management

Jonas H

Item Class



Item Class



Example:
id = 2
price = 100 # Divide by 100 to get $
quantity = 100
holdQuanity = 5 # Quantity in carts

Item Class & Customer Item Class

Store wide
E.G. How much do we have in stock?

Customer wide
E.G. How much do we have in cart?

Item Class & Customer Item Class

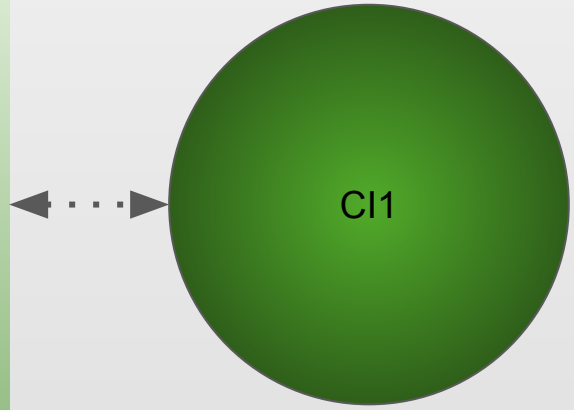
Store wide info
E.G. How much do we have in stock?

Store wide info
E.G. How much do we have in cart?

Fewer files and modification at runtime

Customer Item Class

Example:
Item = *I1
purchaseQuantity = 100
purchasePrice = 101 # Price purchased at, used for viewing old orders



Items

Live (map)

Name: I1 Index: N/A Value: &I1	Name: I2 Index: N/A Value: &I3	…	Name: In Index: N/A Value: &In

No limit on quantity?

Save (JSON)

Name: I6 Value: I6 stuff
Name: I6 Value: I6 stuff
…
Name: Ij Value: Ij stuff

Items

Live (map)

Name: I1 Index: N/A Value: &I1	Name: I2 Index: N/A Value: &I2	…	Name: In Index: N/A Value: &In

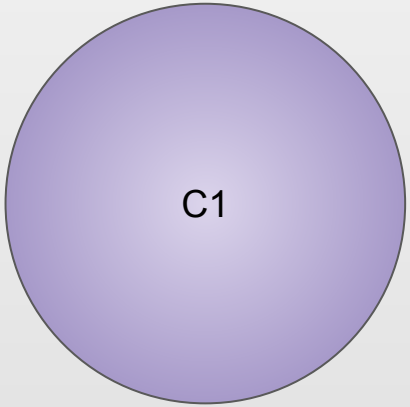
Save (JSON)

Name: I6 Value: I6 stuff
Name: I6 Value: I6 stuff
…
Name: Ij Value: Ij stuff

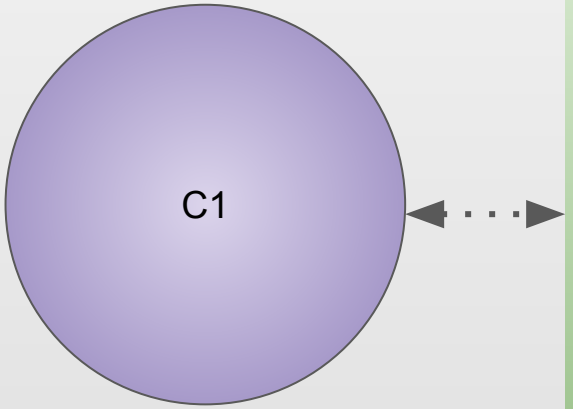
No limit on quantity?

A few things accessed frequently

Customer Class



Customer Class



Example:
customerId = 2
string firstName = Jonas
string lastName Hemmett
int credit = 0 # Store credit
phoneNumber = 0000000000
pendingTransaction = &T1 # Pointer
transactions = {Tn's ID} # Past Transactions

Customers

Customers (map)

Name: C1 Index: N/A Value: &C1	Name: C2 Index: N/A Value: &C2	Name: C3 Index: N/A Value: &C3

Save (JSON)

Name: C4 Value: C4 stuff
Name: C5 Value: C6 stuff
…
Name: Cn Value: Cn stuff

Customers

Customers (map)

Name: C1 Index: N/A Value: &C1	Name: C2 Index: N/A Value: &C2	Name: C3 Index: N/A Value: &C3

Save (JSON)

Name: C4 Value: C4 stuff
Name: C5 Value: C6 stuff
…
Name: Cn Value: Cn stuff

I might have a lot of customers, and I want to move the oldest one from memory. But maps don’t keep index!

Customers

Customers (map)

Name: C1 Index: N/A Value: &C1	Name: C2 Index: N/A Value: &C2	Name: C3 Index: N/A Value: &C3

Ages of Customer (queue)

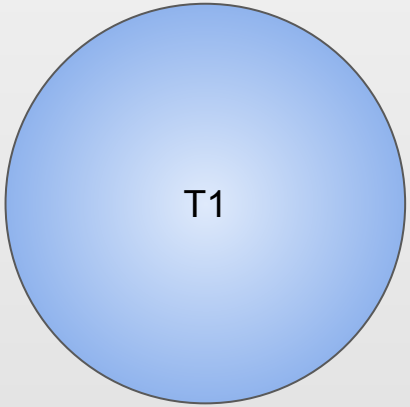
Name: N/A Index: 0 Value: C1's ID	Name: N/A Index: 1 Value: C2's ID	Name: N/A Index: 2 Value: C3's ID

Save (JSON)

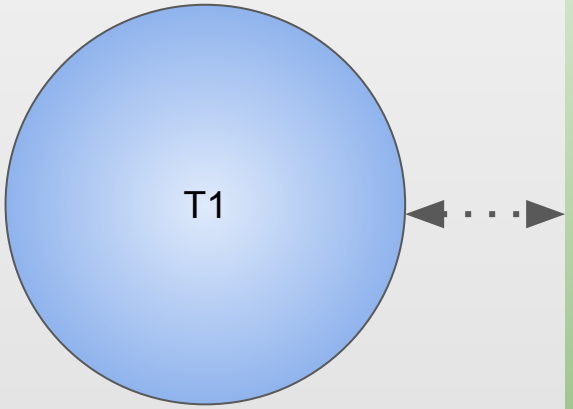
Name: C4 Value: C4 stuff
Name: C5 Value: C6 stuff
…
Name: Cn Value: Cn stuff

I might have a lot of customers, and I want to move the oldest ones from memory. But maps don't keep index! That's why I have a corresponding queue to keep track of when customers were last accessed

Transaction Class



Transaction Class



Example:

totalCost = 0
completed # Boolean
int id = 3
customerId = 2 # Who's transaction
paymnetPortal = &P1
usersItems = {I1, In} # Stores the user's instance not the store's

Transactions

Live (map)

Name: T1 Index: N/A Value: &T1	Name: T2 Index: N/A Value: &T3	…	Name: Tn Index: N/A Value: &Tn

Completed (map)

Name: T3 Index: N/A Value: &T3	Name: T4 Index: N/A Value: &T4	Name: T5 Index: N/A Value: &T5

Ages of Completed (queue)

Name: N/A Index: 0 Value: T3’s ID	Name: N/A Index: 1 Value: T4’s ID	Name: N/A Index: 2 Value: T5’s ID

Save (JSON)

Name: T6 Value: T6 stuff
Name: T6 Value: T6 stuff
…
Name: Tj Value: Tj stuff

Transactions

Wouldn't iterators work?

Live (map)

Name: T1 Index: N/A Value: &T1	Name: T2 Index: N/A Value: &T3	…	Name: Tn Index: N/A Value: &Tn

Completed (map)

Name: T3 Index: N/A Value: &T3	Name: T4 Index: N/A Value: &T4	Name: T5 Index: N/A Value: &T5

Ages of Completed (queue)

Name: N/A Index: 0 Value: T3's ID	Name: N/A Index: 1 Value: T4's ID	Name: N/A Index: 2 Value: T5's ID

Save (JSON)

Name: T6 Value: T6 stuff
Name: T6 Value: T6 stuff
…
Name: Tj Value: Tj stuff

Transactions

Wouldn't iterators work?
YES!

Live (map)

Name: T1 Index: N/A Value: &T1	Name: T2 Index: N/A Value: &T3	…	Name: Tn Index: N/A Value: &Tn

Completed (map)

Name: T3 Index: N/A Value: &T3	Name: T4 Index: N/A Value: &T4	Name: T5 Index: N/A Value: &T5

Ages of Completed (queue)

Name: N/A Index: 0 Value: T3's ID	Name: N/A Index: 1 Value: T4's ID	Name: N/A Index: 2 Value: T5's ID

Save (JSON)

Name: T6 Value: T6 stuff
Name: T6 Value: T6 stuff
…
Name: Tj Value: Tj stuff

Transactions

Wouldn't iterators work?
YES!
But I don't need them

Live (map)

Name: T1 Index: N/A Value: &T1	Name: T2 Index: N/A Value: &T3	…	Name: Tn Index: N/A Value: &Tn

Completed (map)

Name: T3 Index: N/A Value: &T3	Name: T4 Index: N/A Value: &T4	Name: T5 Index: N/A Value: &T5

Ages of Completed (queue)

Name: N/A Index: 0 Value: T3's ID	Name: N/A Index: 1 Value: T4's ID	Name: N/A Index: 2 Value: T5's ID

Save (JSON)

Name: T6 Value: T6 stuff
Name: T6 Value: T6 stuff
…
Name: Tj Value: Tj stuff

Transactions

Wouldn't iterators work?
YES!
But I don't need them

Live (map)

Name: T1 Index: N/A Value: &T1	Name: T2 Index: N/A Value: &T3	…	Name: Tn Index: N/A Value: &Tn

Completed (map)

Name: T3 Index: N/A Value: &T3	Name: T4 Index: N/A Value: &T4	Name: T5 Index: N/A Value: &T5

Ages of Completed (queue)

Name: N/A Index: 0 Value: T3's ID	Name: N/A Index: 1 Value: T4's ID	Name: N/A Index: 2 Value: T5's ID

Save (JSON)

Name: T6 Value: T6 stuff
Name: T6 Value: T6 stuff
…
Name: Tj Value: Tj stuff

They would help if I was removing elements that were not at index 0

Transactions

Wouldn't iterators work?
YES!
But I don't need them

Live (map)

Name: T1 Index: N/A Value: &T1	Name: T2 Index: N/A Value: &T3	…	Name: Tn Index: N/A Value: &Tn

Completed (map)

Name: T3 Index: N/A Value: &T3	Name: T4 Index: N/A Value: &T4	Name: T5 Index: N/A Value: &T5

Ages of Completed (queue)

Name: N/A Index: 0 Value: T3's ID	Name: N/A Index: 1 Value: T4's ID	Name: N/A Index: 2 Value: T5's ID

Save (JSON)

Name: T6 Value: T6 stuff
Name: T6 Value: T6 stuff
…
Name: Tj Value: Tj stuff

They would help if I was removing elements that were not at index 0.
But I'm not, at least for now