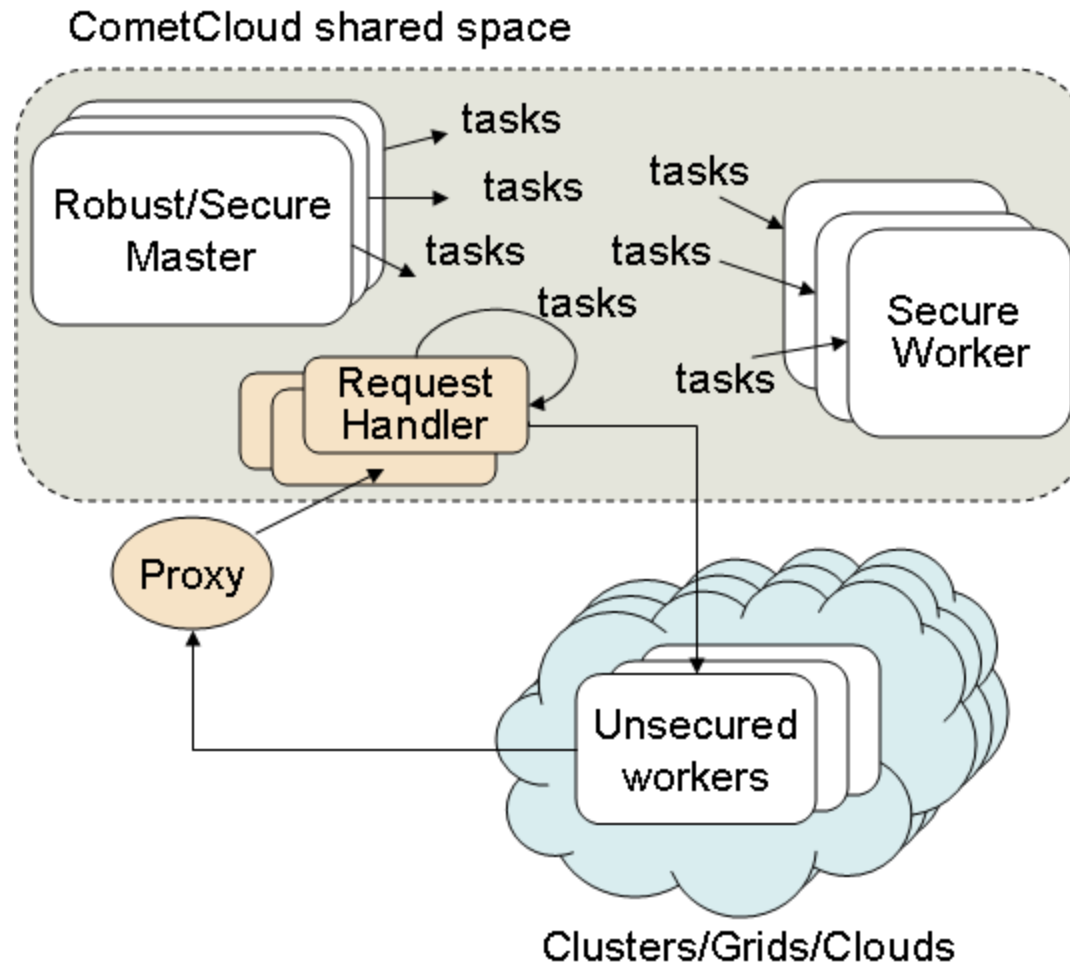


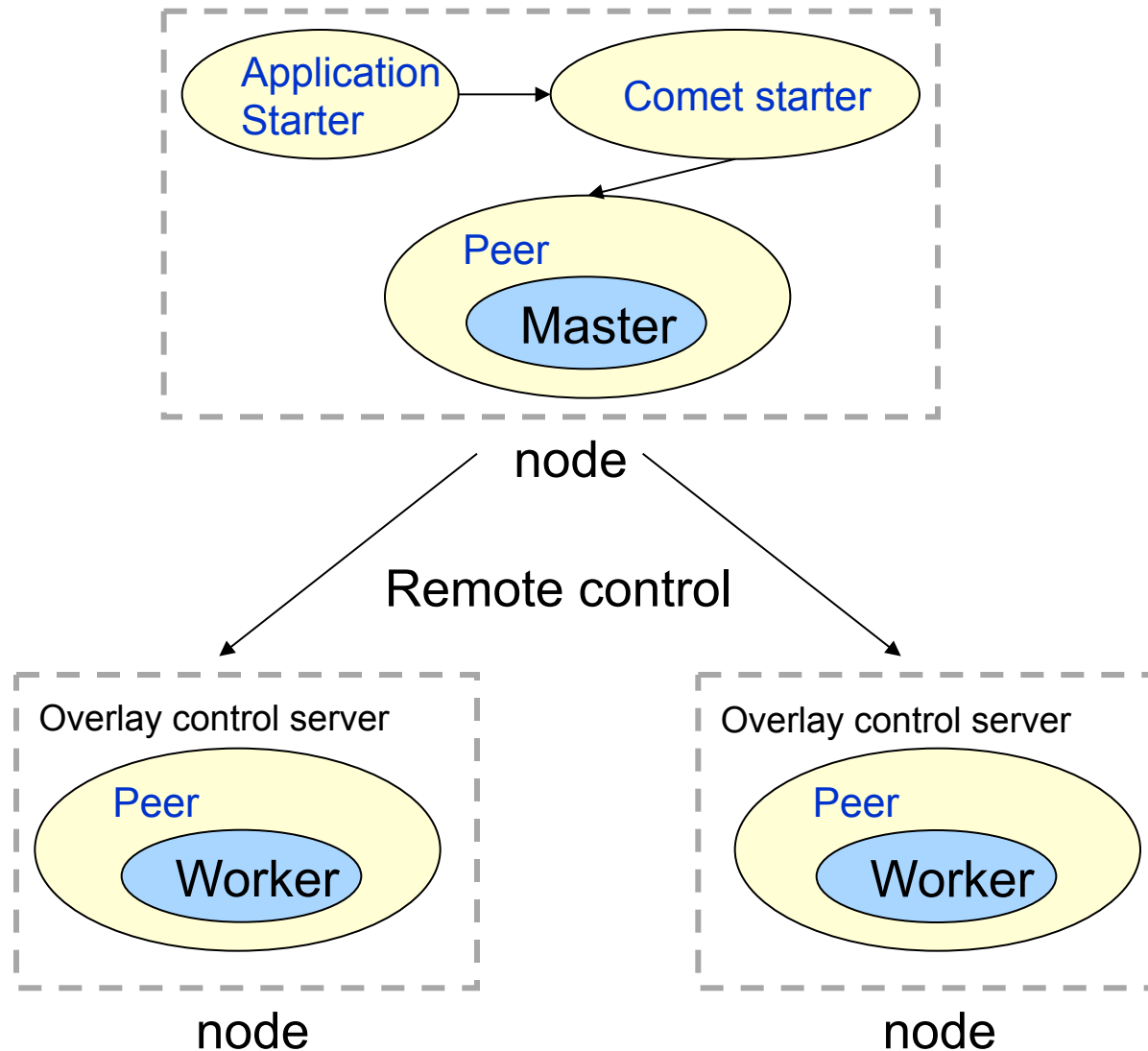


# Master Worker Programming Model

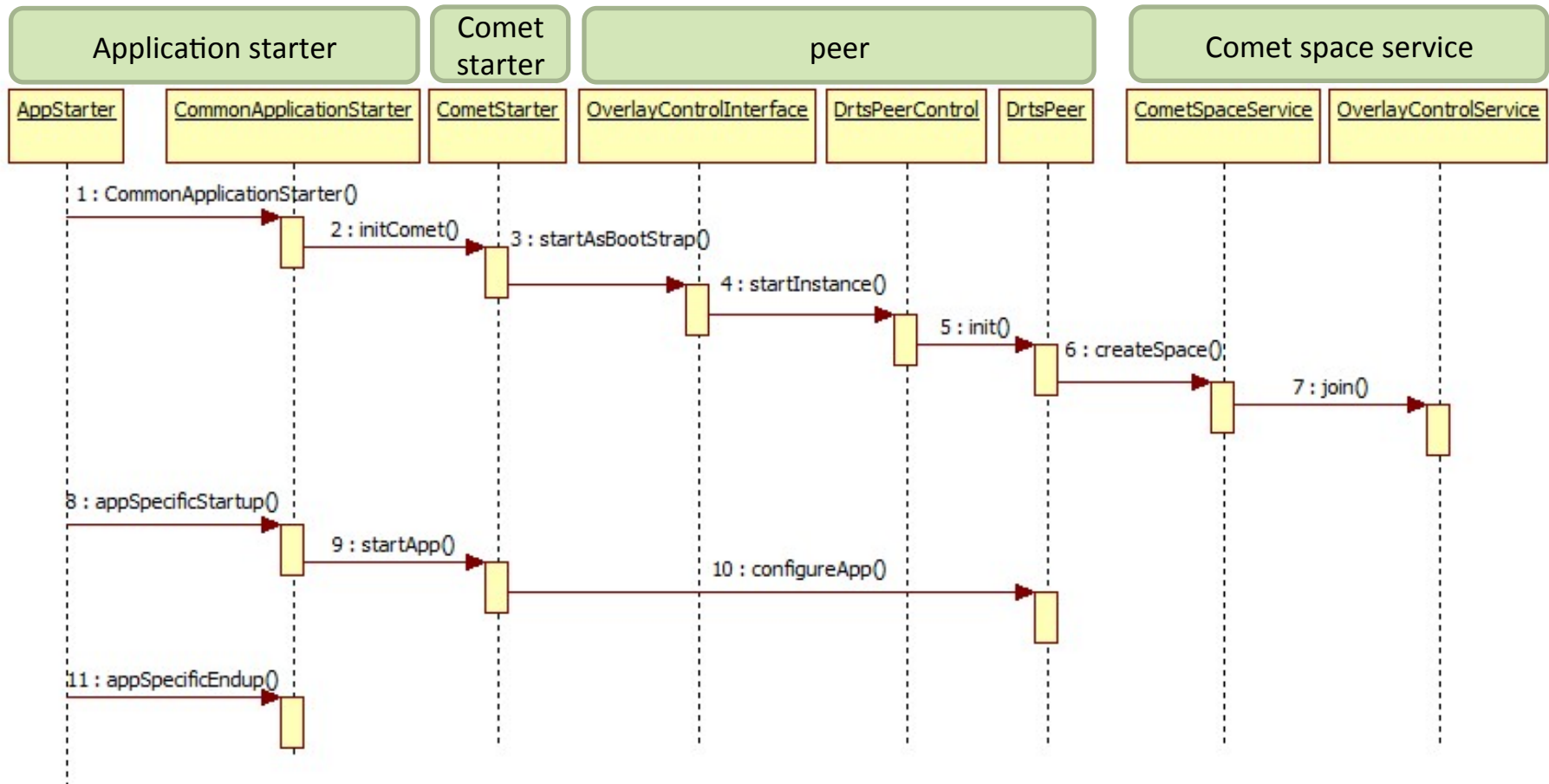
# Master/Worker model



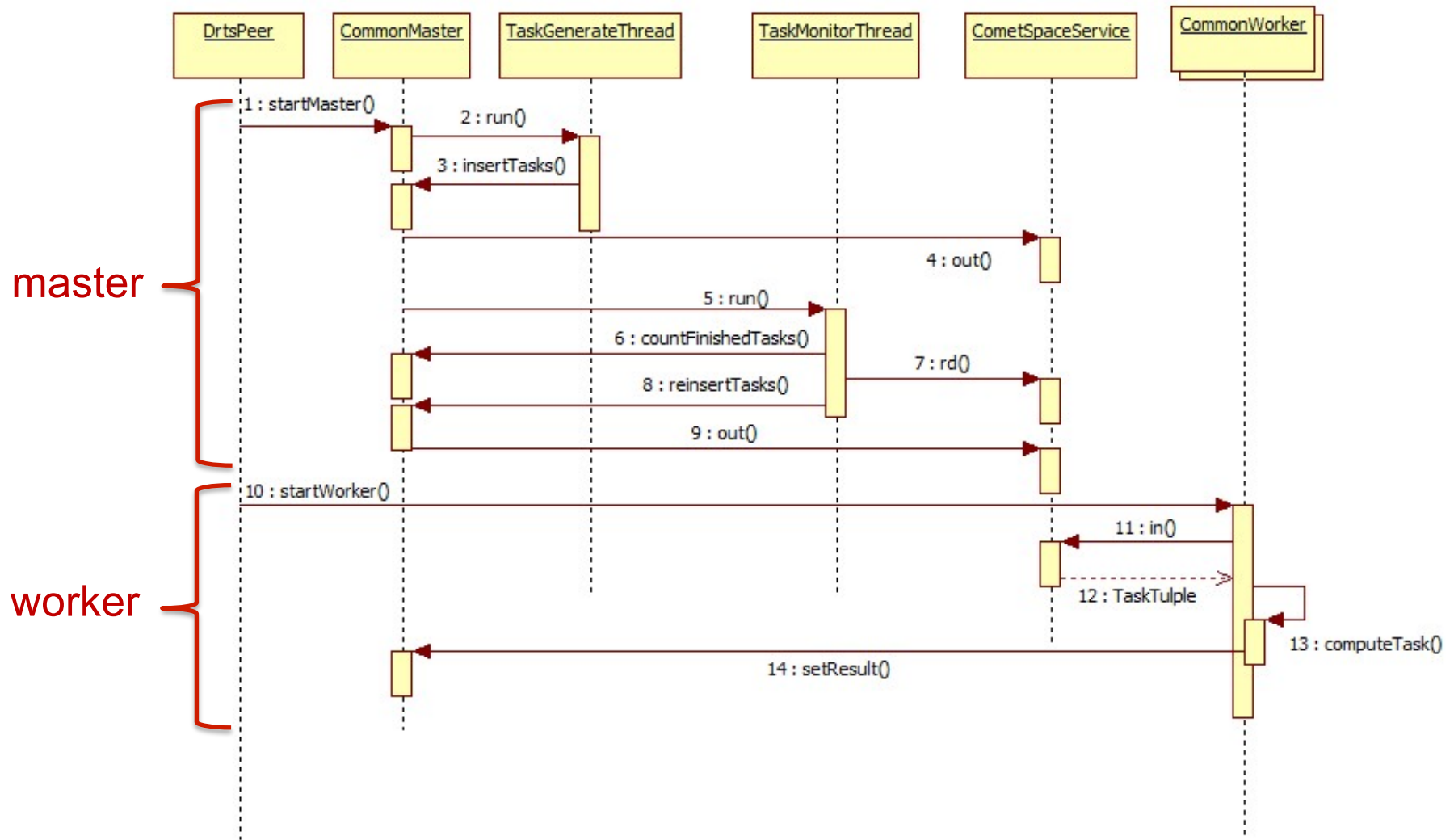
# Flow to start master/worker



# Call Diagram – Starting Comet



# Call Diagram – Master/Worker

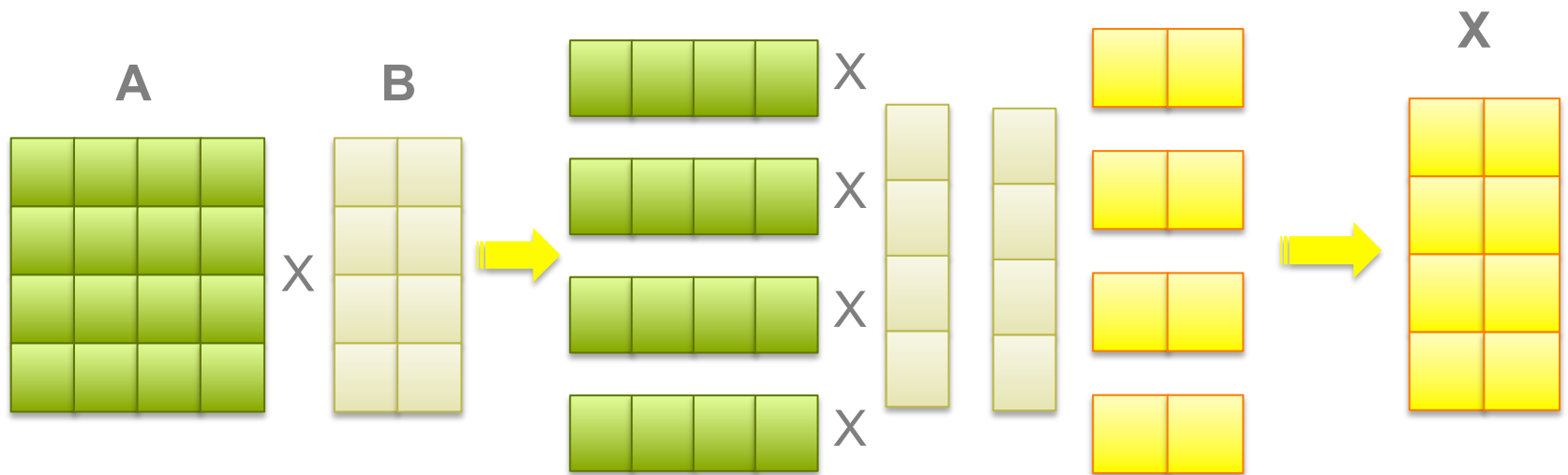


# Source code – template

---

- Starting source package
  - `tassl.automate.programmodel.masterworker.sample`
- Two pairs of master/worker
  - `SimpleMaster.java`, `SimpleWorker.java`
  - `AppMaster.java`, `AppWorker.java`
- Core implementation you should do
  - `TaskTuple`
  - Master: `insertTask`, `setResult`
  - Worker: `computeTask`

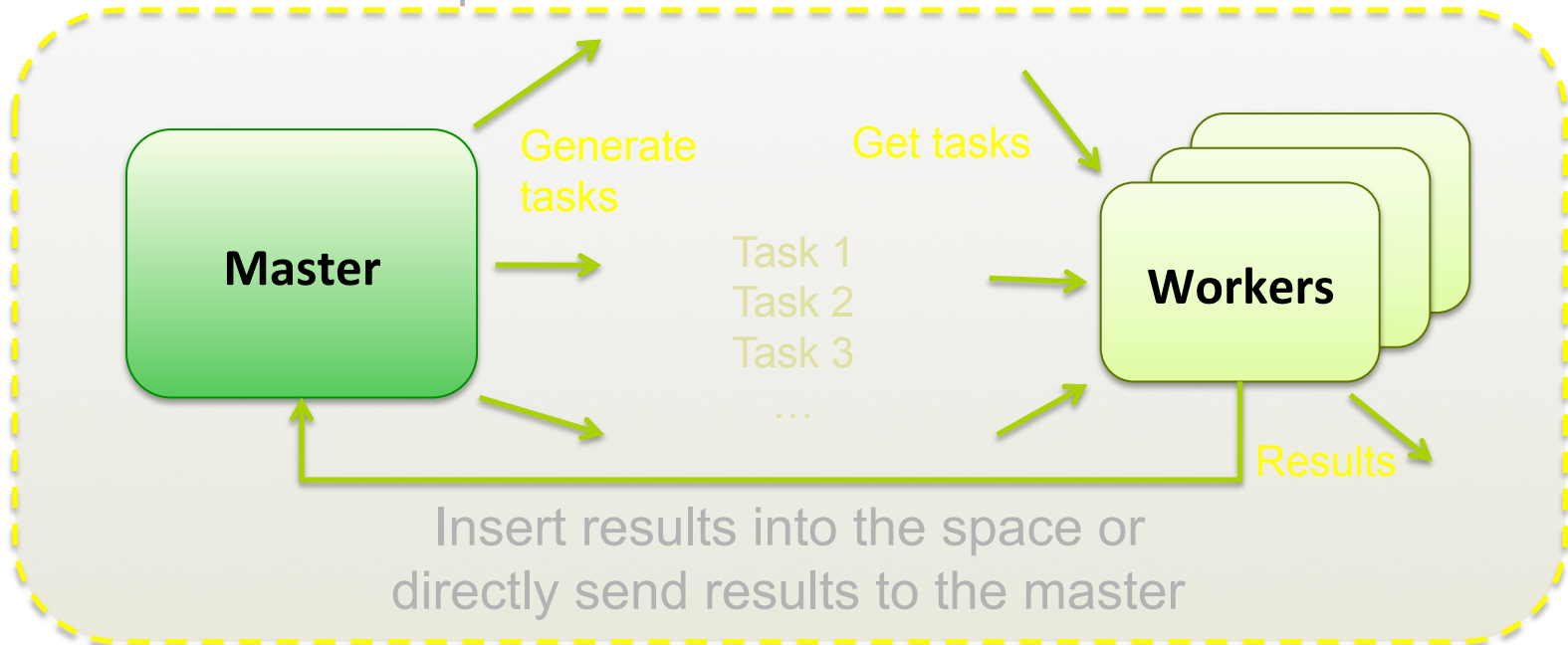
# Matrix Multiplication



# Master Worker

- Master ■ Read matrices dimension  $\mathbf{A}_{n,q}$  and  $\mathbf{B}_{q,m}$  from input file
- Master ■ Generate matrices  $\mathbf{A}_{n,q}$  &  $\mathbf{B}_{q,m}$  randomly
- Master ■ Create  $n \times m$  tasks and insert into the space
  - Each task contains  $1, n \times n, 1$  vectors
- Worker ■ Pull and execute one task at a time
- Worker ■ Perform vector multiplication & return sum
- Master ■ Gather results to matrix  $\mathbf{C}$  & print to screen

Comet shared space





# Configuration

---

- Overlay configuration files
  - nodeFile, portFile, exceptionFile
    - Comet.NodeType MASTER|WORKER|REQUEST\_HANDLER
- Properties files
  - chord.properties, squid.properties, comet.properties
  - and your additional application properties
- Examples in
  - CometCloud-lite\simple\_run

# Configuration

---

- chord.properties
  - chord.ID\_BITS
- squid.properties
  - squid.SPACE\_DIMENSIONS
  - squid.BIT\_LENGTH
  - squid.D0.KEY\_TYPE
- comet.properties
  - MasterClass
  - WorkerClass
  - TaskClass
  - RoutingKeys

# Run procedure – Secure worker

---

- Run overlay control server in each node
  - java -cp \$CLASSPATH  
tassl.automate.overlay.OverlayControlServer 4444
- Run your application starter
  - java -cp \$CLASSPATH  
tassl.automate.programmodel.masterworker.sample.App  
Starter -nodeFile nodeFile -portFile portFile -exceptionFile  
exceptionFile -propertyFile chord.properties -propertyFile  
squid.properties -propertyFile comet.properties -  
propertyFile your\_app.properties

# Run procedure – Isolated (unsecured) worker

---

- ❑ Run overlay control server on one or more request handler nodes
- ❑ Start a proxy with RequestHandlerList
  - `java -cp $CLASSPATH tassl.automate.cloudburst.RequestHandlerProxy`
- ❑ Run your application starter
  - This will start your master and request handlers
- ❑ Start isolated workers
  - Set `IsolatedProxy` in `comet.properties`
  - `java -cp $CLASSPATH tassl.automate.cloudburst.CloudBurstStarter -propertyFile comet.properties -propertyFile`

# Questions?

---

