Async-re.py

12. query/exchange count

1. Launch BigJobs

BigJob
(bigjob.py)

2. Post Job Description

3. Launch BigJob Agents

BigJob Agent 1

BigJob Agent 1   4. Poll for jobs

Advert Service

5. Launch Replica Agents as Subjobs

5. Launch Replica Agents as Subjobs

9. Post results

Replica Agent 1

Replica Agent 2

....

Replica Agent N-1

Replica Agent N

10. Query for matching replicas

6, 13. Launch   6,13. Launch

6, 13. Launch   6, 13. Launch

Replica 1

Replica 2

Replica N-1

Replica N

7. Replica Done

7. Replica Done

8. collect results

Replica 1

Replica 2

Replica N-1

Replica N

Resource 1   11. Exchange   Resource N

Co-ordination: Red
Communication: Black
Run:
Green

Decentralized Control flow:
i. Async-re.py is used to set the job descriptions and launch the BigJob.py(BigJob Manager) which posts the job description to the advert service.
ii. As and when a BigJob becomes active, BigJob.py launches the BigJob agents.
iii. The bigjob agents poll for jobs in the advert server and allocate the resources to the subjobs
iv. The subjob in this case is the replica agent. The replica agents are now launched
v. The replica agents launch their respective replicas and monitor them
vi. the replica agent collects the results and posts them to the advert server.
vii. the replica agents also query the advert server and if they find a matching replica, they negotiate an exchange
viii. the async-re.py continuously queries/ updates the exchange count and when the required number of exchanges complete, the experiment ends.

Time for each step:
1: 2 seconds
2: 1 second
3: partial queue time
(waits till the BJs get active
4: 1.5 seconds per job
5: included in 4
6: 0.3 seconds
7: run time
8: 0.5 seconds
9: included in 8
10: 1 sec
11: 1 second
12: 0.3 sec0nds