Checkpoint Report
Original Schedule
Achieved Goals are highlighted in green

Week #	Tasks
April 10th - April 16th	• Refresh ourselves on directory-based cache coherence and various protocols • Initial Design • Familiarize ourselves with Intel Pintool • Develop the basic test and correctness harness
April 17th - April 23rd	• Implement a simple cache • Implement the MSI and MESI protocol • Start on developing an evaluation harness
April 24th - April 30th	• Implement MESIF protocol • Detailed analysis of cache behaviour of test memory traces
May 1st - May 10th	• User-Friendly output explaining the cache behaviour of any input executable • If goals achieved, work on stretch goals • Project wrap-up

*We are in the process of finishing up the MSI implementation, the design took us longer than we expected. That being said, we should finish MSI/MESI by the end of the week.

Revised Schedule

Divided into 3 day blocks

Time Block	Task	Team Member
April 25th-27th	• Implement protocolHandler, MSIProtocolHandler, and directory classes • Change current code to reflect more realistic timing • Integrate different pieces	both
April 28th - 30th	• Implement MESIProtocolHandler • Test correctness of simulator	both
May 1st - 3rd	• Implement MESIFProtocolHandler • Test correctness of simulator	both
May 4th - 6th	• Detailed comparison of memory traces with MSI/MESI/MESIF protocols • Analyze and compare cache behavior of memory traces using snooping based and directory based cache coherence protocols	both
May 7th - 9th	• Develop a GUI for the end user • Nice to have - work on parallelizing the simulator	both
May 10th - 12th	• Wrap - up the sequential simulator • Nice to have -Work on parallel simulator	both

- One to two paragraphs, summarize the work that you have completed so far. (This should be easy if you have been maintaining this information on your project page.)

  We were able to get and parse memory trace files using Contech, and feed the traces into our simulator. With object-oriented design in mind, we spent a fair amount of time coming up with a good representation of different entities in our project such as directory, protocol, processor, cache, etc. We also designed a messaging scheme to simulate communication between entities. We carefully made reasonable assumptions while building our simulator regarding sequential consistency, memory mapping and latency, and so on. We have completed most part of the code for our first version of the simulator, which runs MSI protocol.

- Describe how you are doing with respect to the goals and deliverables stated in your proposal. Do you still believe you will be able to produce all your deliverables? If not, why? What about the "nice to haves"? In your checkpoint writeup we want a new list of goals that you plan to hit for the Parallelism competition.

  We are a little behind our original schedule in completing a working version of simulator. This is due to the fact that we spent considerable time sketching out our object-oriented, modularized, and scalable design, which we think is worthwhile. After planning out and adjusting our tasks for the next two weeks, we believe that we will be able to produce all our deliverables because our initial schedule was relatively light towards the second half of the project period. Our “nice to haves” basically involves parallelize our simulator, as right now it’s designed to run sequentially.

- What do you plan to show at the parallelism competition? Will it be a demo? Will it be a graph?

  - We plan to run various taskGraphs under different protocols and compare the cache health of the given program under the different protocols via graphs.
  - Secondly, we plan to compare and analyse the results we get with the results from the snooping based protocol from previous years and study which protocol is more useful when. The final demo will have a defined “which” and “when”.

- Do you have preliminary results at this time? If so, it would be great to included them in your checkpoint write-up.

  We do not have any simulation results yet as we are still in the process of integrating our individual modules. Here's our github repo https://github.com/Ridhii/contech-repo and our code is in /backend/Simulator/

- List the issues that concern you the most. Are there any remaining unknowns (things you simply don't know how to solve, or resource you don't know how to get) or is it just a matter of coding and doing the work? If you do not wish to put this information on a public web site you are welcome to email the staff directly.

  Concerns -
  - We are yet to finish the implementation for MSI. The design took us a little longer than we thought.
  - Timing Analysis - In order to reflect "real time" it takes for any kind of operation to complete we accept the latency parameters from the user - however, we anticipate that this will make our simulation really slow.
  - Ideally, we plan to parallelize the simulator and we have designed our simulator to be as modular and easy to change as possible .Still, we acknowledge that there will be scenarios that we haven't thought about as of yet and will have to solve then.