UCP and within constraints provided by the co-ordination editor 330. The editor 330 ensures that the designer of a new protocol does the following:

1. starts with the UCP;
2. identifies the parts of the UCP where it is necessary to add the extra complexity to realise the new protocol; and
3. defines and implements the new graph either by adding a sub-graph or by refinement of existing arcs or nodes of a predefined graph.
4. if any new nodes or arcs need to be defined the editor 330 provides the user templates for these.

Thus, for each of the agents identified in Step 1, the user selects and/or develops zero or more negotiation/co-ordination strategies which the agent can invoke, via the Co-ordination Editor 330.

4.5 Step 5: *Tasks Definition 520*

For each of the tasks identified in Step 2, the following user inputs have to be made for each agent, via the task description editor 335:

1. Determine the average cost and duration of performing the task.
2. Exhaustively list as variables 550 all the items (resources) that are required before the task can be performed.
3. Exhaustively list as variables 550 all the items (products) produced once the task is performed.
4. Determine and note all the constraints between the consumed items (No. 2 above) and the produced items (No. 3 above), and within each group.
5. Determine if the task can be performed by directly executing a domain function (*primitive tasks*) or whether it is in fact an abstract specification of a network of other tasks (i.e. it is a *summary task*).
6. If the task is primitive then provide the following two functions (a) one to execute to perform the task (a callback 555) and (b) one to compute the actual cost of the task once it has been performed. For summary tasks