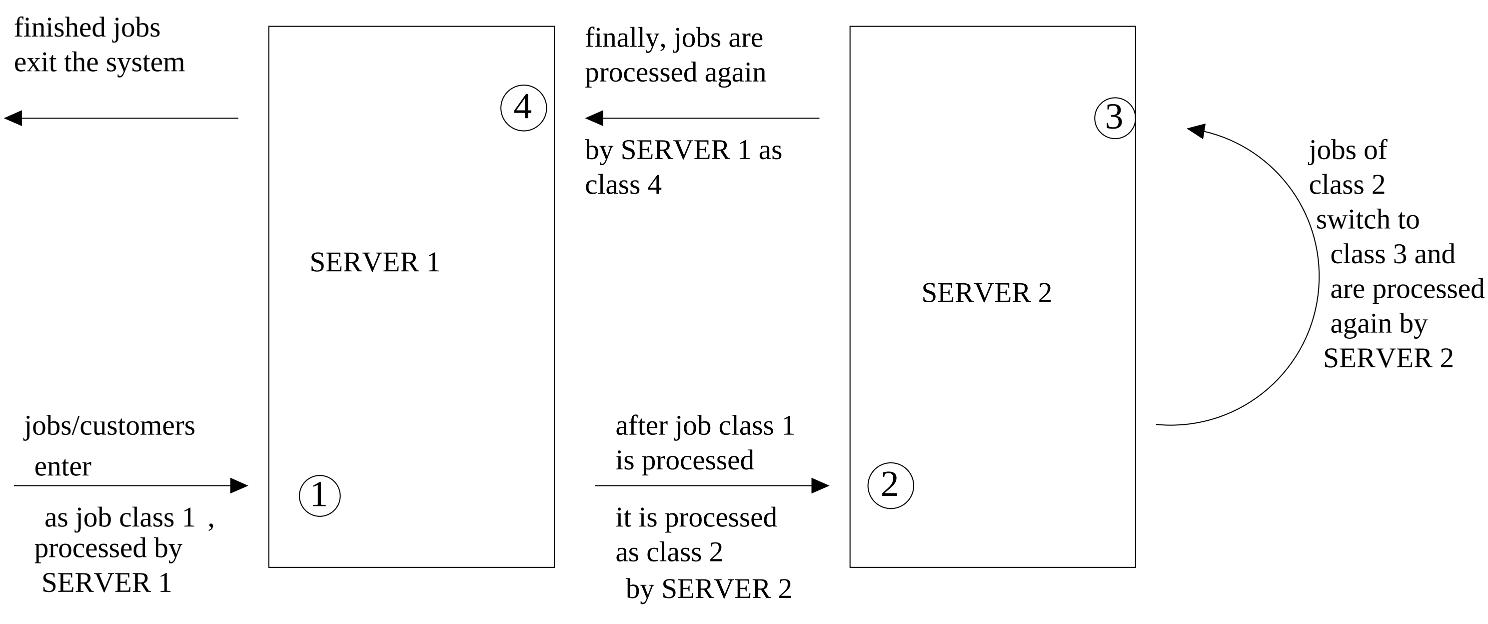
Optimal scheduling policies in networks with multi-task servers

In this project the students will develop models for queueing networks in which "customers" may return to a server with requests for different tasks or "jobs". Examples of such systems occur in manufacturing processes and design of disk controllers. In semi-conductor device manufacture, where there are many steps in the manufacturing process, the same machine is used for several of these steps. Similarly, disk controllers are dedicated processors for efficient reading and writing data to a hard disk; the efficiency depends on implementing efficient algorithms for storage, so that writing data results in an additional service request for applying the smart algorithm.

In both of these examples, a bottleneck in the system can occur because of feedback. If one server processes jobs quickly, but these same jobs then return later for additional processing, there is then a competition between new jobs and old jobs requiring further service. If old jobs have priority for service, then the new jobs will have to wait a long time, unless old jobs are processed very quickly. A typical system will consist of a network of such multi-task servers, so that the potential for bottlenecks in the processing flow can easily occur. The figure shows a schematic for a simple system in which an overload can occur at one point in the system, even when the overall load on the system is not high. This overload can occur over a long time when job class 4 has a priority over job class 1, and the ratio of processing times is sufficiently large.



A crucial element in the design of an efficient processing network is assigning optimal policies for service priorities. The choice of the optimal process depends on restrictions on the servers, such as ability to process jobs simultaneously or switching costs, the measure of efficiency, such as throughput or wait time, as well as the structure of the network itself.

The goal of the project will be to build simple models which illustrate the the bottleneck problems described above. By viewing these models as sub-networks in a larger network, the team will develop optimal priorities for job processing in these small sub-networks, as a basis for constructing an overall optimal policy.

There are a variety of approaches that can be used to model these systems. One can view the process as a stochastic system with random arrival and service times. In this case methods used in queueing networks and Markov chains can be used. Alternatively, one can view the system as a fluid process, in which a concentration of jobs or customers "flow" through the network. Then the models are based on differential equations, either deterministic or stochastic. Students will use tools from probability, differential equations, dynamical systems, and optimization to build the models. They will analyze and simulate the systems to evaluate the effects of different priority policies.