# Public Library Computer Waiting Queues: Alternatives to the First-Come-First-Served Strategy

Stuart Williamson

**ABSTRACT**

*This paper summarizes the results of a simulation of alternative queuing strategies for a public library computer sign-up system. Using computer usage data gathered from a public library, the performance of these various queuing strategies is compared in terms of the distribution of user wait times. The consequences of partitioning a pool of public computers are illustrated as are the potential benefits of prioritizing users in the waiting queue according to the amount of computer time they desire.*

**INTRODUCTION**

Many of us at public libraries are all too familiar with the scene: a crowd of customers huddled around the library entrance in the morning, anxiously waiting for the doors to open to begin a race for the computers. From this point on, the wait for a computer at some libraries, such as the one we will examine, can hover near thirty minutes on busy days and peak at an hour or more. Such long waiting times are a common source of frustration for both customers and staff.

By far the most effective solution to this problem is to install more public computers at your library. Of course, when the space or money run out, this may no longer be possible. Another approach is to reduce the length or number of sessions each customer is allowed. Unfortunately, reducing session length can make completion of many important tasks difficult; whereas, restricting the number of sessions per day can result in customers upset over being unable to use idle computers.[1] Finally, faced with daunting wait times, libraries eager to make their computers accessible to more people may be tempted to partition their waiting queue by installing separate fifteen-minute "express" computers. A primary focus of this paper is to illustrate how partitioning the pool of public computers can significantly increase waiting times. Additionally, several alternative queuing strategies are presented for providing express-like computer access without increasing overall waiting times.

We often take for granted the notion that first-come-first-served (FCFS) is a basic principle of fairness. "I was here first," is an intuitive claim that we understand from an early age. However,

***Stuart Williamson*** (swilliamson@metrolibrary.org) is Researcher, Metropolitan Library System, Oklahoma City, Oklahoma.

the inefficiency present in a strictly FCFS queue is implicitly acknowledged when we courteously invite a person with only a few items to bypass our overflowing grocery cart to proceed ahead in the check-out line. Most of us would agree to wait an additional few minutes rather than delay someone else for a much greater length of time. When express lanes are present, they formalize this process by essentially allowing customers needing help for only a short period of time to cut in line. These line cuts are masked by the establishment of separate dedicated lines, i.e., the queue is partitioned into express and non-express lines.

One question addressed by this article is "is there a middle ground?" In other words, how might a library system set up its computer waiting queue to achieve express-lane type service without splitting the set of public internet computers into partitions that operate separately and in parallel? Several such strategies are presented here along with the results of how each performed in a computer simulation using actual customer usage data from a public library.

## STRATEGIES

Queuing systems are heavily researched in a number of disciplines, particularly computer science and operations research. The complexity and sheer number of different queuing models can present a formidable barrier to library professionals. This is because, in the absence of real-world data, it is often necessary to analyze a queuing system mathematically by approximating its key features with an applicable probability distribution. Unfortunately, applying these distributions entails adopting their underlying assumptions as well as any additional assumptions involved in calculating the input parameters. For instance, the Poisson distribution (used to approximate customer arrival rates) requires that the expected arrival rate be uniform across all time intervals, an assumption which is clearly violated when school lets out and teenagers suddenly swarm the computers.[2] Even if we can account for such discrepancies, there remains the difficulty of estimating the correct arrival rate parameter for each discrete time interval being analyzed.

Fortunately, many libraries now use automated computer sign-up systems which provide access to vast amounts of real-world data. With realistic data, it is possible to simulate various queuing strategies, a few of which will be analyzed in this article. A computer simulation using real-world data provides a good picture of the practical implications of any queuing strategy we care to devise without the need for complex models.

As is often the case, designing a waiting queue strategy involves striking a balance among competing factors. For instance, one way of reducing waiting times involves breaking with the FCFS rule and allowing users in one category to cut in front of other users. How many cuts are acceptable? Does the shorter wait time for users in one category justify the longer waits in another? There are no right answers to these questions. While simulating a strategy can provide a realistic picture of its results in terms of waiting times, evaluating which strategy's results are preferable for a particular library must be done on a case-by-case basis.

In addition to the standard FCFS strategy with a single pool of computers and the same FCFS strategy implemented with one computer removed from the pool to serve as a dedicated fifteen-

minute express computer (referred to as FCFS-15), we will consider for comparison three other well-known alternative queuing strategies: Shortest-Job-First (SJF), Highest-Response-Ratio-Next (HRRN), and a variant of Shortest-Job-First (SJF-FB) which employs a feedback mechanism to restrict the number of times a given user may be bypassed in the queue.[3]

The three alternative strategies all require advance knowledge or estimation of how long each particular computer session will last. In our case, this means customers would need to indicate how long of a session they desire upon first signing up for a computer. Any number of minutes is acceptable so we will limit the sign-up options to four categories in fifteen-minute intervals: fifteen minutes, thirty minutes, forty-five minutes, and sixty minutes. Each session will then be initially categorized into one of four priority classes (P1, P2, P3, and P4) accordingly. As the data will show, customers selecting shorter sessions are given a higher priority in the queue and will thus have a shorter expected waiting time.

It should be noted that relying on users to choose their own session length presents its own set of problems. It is often difficult to estimate how much time will be required to accomplish a given set of tasks online. However, users face a similar difficulty in deciding whether to opt for a dedicated fifteen-minute computer under the FCFS-15 system. The trade-off between use time and wait time should provide an incentive for some users to self-ration their computer use, placing an additional downward pressure on wait times. However, user adaptations in response to various queuing strategies are outside the scope of this analysis and will not be considered further.

The Shortest-Job-First (SJF) strategy functions by simply selecting from the queue the user in the highest priority class. The amount of time spent waiting by each user is only considered as a tie breaker among users occupying the same priority class. Our results demonstrate that the SJF strategy is generally best for minimizing overall average waiting time as well as for getting customers needing the least amount of computer time online the fastest.

The main drawbacks of this strategy are that these gains come at the expense of more line cuts and higher average and maximum waiting times for the lowest priority users—those needing the longest sessions (sixty minutes). There is no limit to how many times a user can be passed over in the queue. In theory, this means that such a user could be continually bypassed and never be assigned a computer during the day.

The SJF-FB strategy is a variant of SJF with the addition of a feedback mechanism that increases the priority of users each time they are cut in line. For instance, if a user signs up for a sixty-minute session, he/she is initially assigned a priority of 4. Suppose that shortly after, another user signs up for a thirty-minute session and is assigned a priority of 2. The next available computer will be assigned to the user with the priority 2. The bypassed user's priority will now be bumped up by a set interval. In this simulation an interval of 0.5 is used so the bypassed user's new priority becomes 3.5. As a result, users beginning with a priority of 4 will reach the highest priority of 1 after being bypassed six times and will not be bypassed further. This effectively restricts the maximum number of times a user can be cut in front of at six.

The final alternative strategy, Highest-Response-Ratio-Next (HRRN), is a balance between FCFS and SJF. It considers both the arrival time and requested session length when assigning a priority to each user in the queue. Each time a user is selected from the queue, the response ratio is re-calculated for all users. The user with the highest response ratio is selected and assigned the open computer. The formula for response ratio is:

$$ResponseRatio = 1 + \frac{(TimeNow - ArrivalTime)}{SessionLength}$$

This allows users with a shorter session request to cut in line, but only up to a point. Even customers requesting the longest possible session move up in priority as they wait, just at a slower pace. This method produces the same benefits and drawbacks as the SJF strategy; but the effects of both are moderated, and the possibility of unbounded waiting is eliminated. Still, although the expected number of cuts will be lower using HRRN than with SJF, there is no limit on how many times a user may be passed over in the queue.

The response ratio formula can be generalized by scaling the importance of the waiting time factor. For instance in the modified response ratio below, increasing values of $x > 1$ will cause the strategy to more resemble FCFS, and decreasing values of $0 < x < 1$ will more resemble SJF.

$$ResponseRatio' = 1 + \frac{(TimeNow - ArrivalTime)^x}{SessionLength}$$

One could experiment with different values of *x* to find a desired balance between the number of line cuts and the impact on average waiting times for customers in the various priority classes. This won't be pursued here, and *x* will be assumed to be 1.

## METHODOLOGY

The data used in this simulation come from the Metropolitan Library System's Southern Oaks Library in Oklahoma City. This library has eighteen public Internet computers that customers can sign up for using proprietary software developed by Jimmy Welch, Deputy Executive Director/Technology for the Metropolitan Library System. The waiting queue employs the first-come-first-served (FCFS) strategy. Customers are allotted an initial session of up to sixty minutes but may extend their session in thirty-minute increments so long as the waiting queue is empty. Repeat customers are also allowed to sign up for additional thirty-minute sessions during the day, provided that no user currently in the queue has been waiting for more than ten minutes (an indication that demand for computers is currently high). Anonymous usage data gathered by the system in August 2010 was compiled to produce the information about each customer session shown in table 1.

| Sign-up Time | Log-in Delay | Session Length | Subsequent Session | Sign-up Abandoned |
|---|---|---|---|---|
| 8/3/2010 6:53:57 PM | 0.87 | 60.03 | False | False |
| 8/3/2010 6:54:48 PM | 8.17 | - | False | True |
| 8/3/2010 7:01:19 PM | 1.03 | 29.82 | True | False |
| 8/3/2010 7:02:10 PM | 0.92 | 59.57 | False | False |

**Table 1. Session Data (units in minutes)**

The information about each session required for the simulation includes the time at which the user arrived to sign up for a computer, the number of minutes it took the user to log in once assigned a computer, how many minutes of computer time were used, whether or not this was the user's first or a subsequent session for the day, and finally, whether the user gave up waiting and abandoned his/her place in the queue. Users are given eight minutes to log in once a computer station is assigned to them before they are considered to have abandoned the queue. Once this data has been gathered, the computer simulation runs by iterating through each second the library is open. As user sign-up times are encountered in the data, they are added to the waiting queue. When a computer becomes available, a user is selected from the queue using the strategy being simulated and assigned to the open computer. The customer occupies the computer for the length of time given by their associated log-in delay and session length. When this time expires, customers are removed from their computer and the information recorded during their time spent in the waiting queue is logged.

## RESULTS

There were 7,403 sign-ups for the computers at the Southern Oaks Library in August 2010. Each of these requests is assigned a priority class based on the length of the session as detailed in table 2. The intended session length of users choosing to abandon the queue is unknown. Abandoned sign-ups are assigned a priority class randomly in proportion to the overall distribution of priority classes in the data so as not to introduce any systematic bias into the results. Even though their actual session length is zero, these users participate in the queue and cause the computer eventually assigned to them to sit idle for eight minutes until it is re-assigned. Customers signing up for a subsequent session during the day are always assigned the lowest priority class (P-4) regardless of their requested session length. This is a policy decision to not give priority to users who have already received a computer session for the day.

| Session Length (min) | Priority Class | Count |
|---|---|---|
| Abandoned Sign-up [0] | Proportionally Assigned | 743 |
| (0 – 15] | P-1 | 448 |
| (15 – 30) | P-2 | 682 |
| (30 – 45] | P-3 | 755 |
| (45 – 60] | P-4 | 1,872 |
| [> 60] | P-4 | 1,314 |
| Subsequent Session [any] | P-4 | 1,589 |

**Table 2. Assignment of Priority Classes**

Figure 1 displays the average waiting time for each priority class during the simulation (bars) along with the total number of sessions initially assigned to each class (line). It is immediately obvious from the chart that each alternative strategy excels at reducing the average wait for high priority (P1) users. Also observe how removing one computer from the pool to serve exclusively as a fifteen-minute computer drastically increases the FCFS-15 average wait times in the other priority classes. Clearly, removing one (or more) computer from the pool to serve as a dedicated fifteen-minute station is a poor strategy here for all but the 519 users in class P-1. Losing just one of the eighteen available computers nearly doubles the average wait for the remaining 6,884 users in the other priority classes.

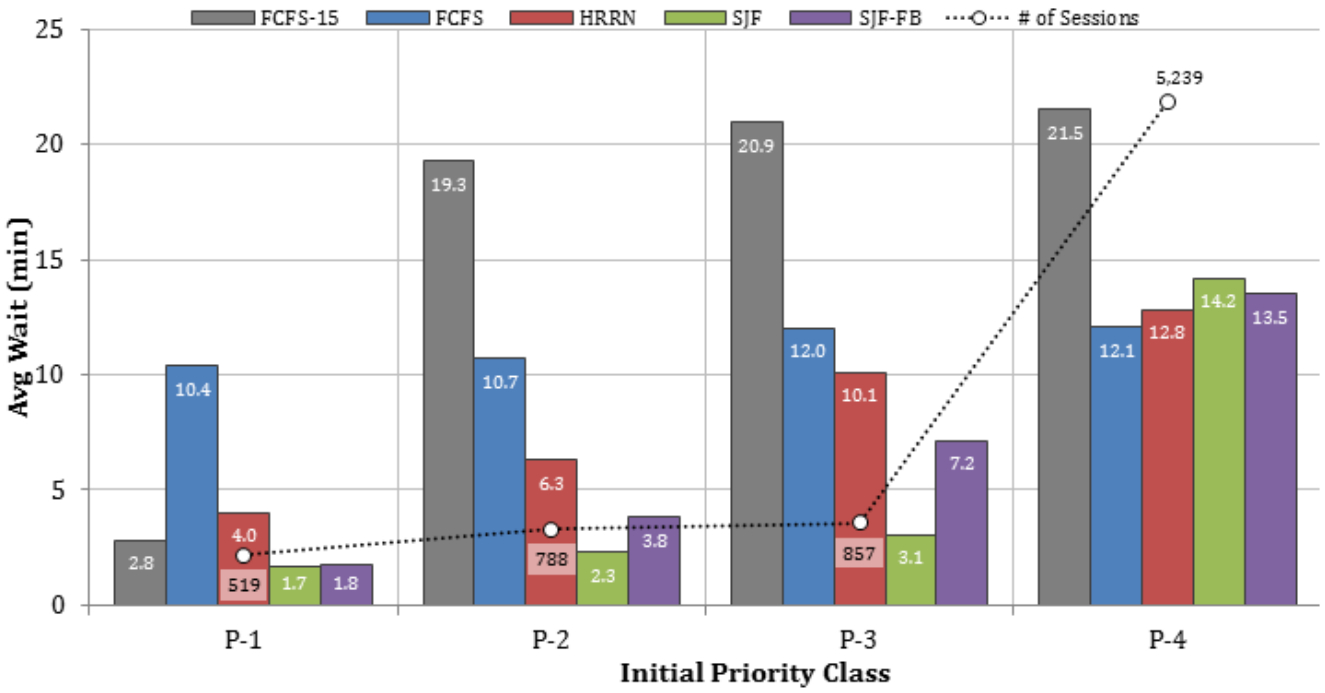


**Figure 1. Average User Wait Minutes by Priority Class**

By contrast, note that the reduced average wait times for the highest priority users in class P-1 persist in classes P-2 and P-3 for the non-FCSC strategies. The SJF strategy produces the most dramatic reductions for the 2,164 users not in class P-4. However, for the 5,239 users in class P-4, the SJF strategy produced an average wait time that was 2.1 minutes longer than the purely FCFS strategy. The HRRN strategy achieves lesser wait time reductions than SJF in the higher priority classes, but HRRN increased the average wait for users in class P-4 by only 0.7 minutes relative to FCFS. The average wait using the SJF-FB strategy falls in between that of SJF and HRRN for each priority class while guaranteeing users will be cut at most six times.

An examination of the maximum wait times for each priority class in figure 2 illustrates how the express lane itself can be a bottleneck. Even with a dedicated fifteen-minute express computer under the FCFS-15 strategy, at least one user would have waited over half an hour to use a computer for fifteen minutes or less. In all but the highest priority class (P-2 through P-4), the FCFS-15 strategy again performs poorly with at least one user in each of these classes waiting over ninety minutes for a computer.

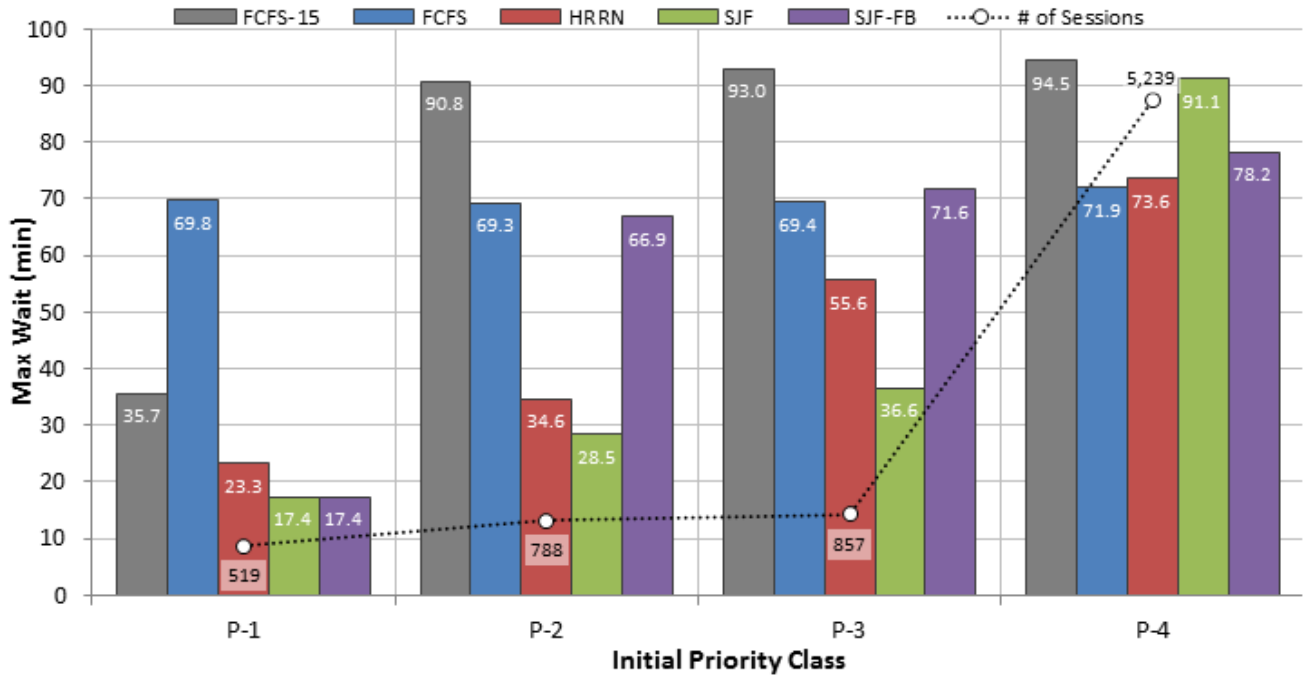


**Figure 2. Maximum User Wait Minutes by Priority Class**

Capping the number of times a user may be passed over in the queue under the SFJ-FB strategy makes it less likely that members of classes P-2 and P-3 will be able to take advantage of their higher priority to cut in front of users in class P-4 during periods of peak demand. As a result, the SJF-FB maximum wait times for classes P-2 and P-3 are similar to those under the FCFS strategy. This was not the case in the breakdown of SJF-FB average waiting times across priority classes in figure 1.

Table 3 breaks down waiting times for each queuing strategy according to the overall percentage of users waiting no more than the given number of minutes. Here we see the effects of each strategy on the system as a whole, instead of by priority class. Notice that the overall average wait times for the non-FCFS strategies are lower than those of FCFS. This indicates that the total reduction in waiting times for high-priority users exceeds the additional time spent waiting by users in class P-4. In other words, these strategies are globally more efficient than FCFS. Notice, too, in table 3 that the non-FCFS strategies achieve significant reductions in the median wait time compared with FCFS.

| % of Users | FCFS-15 | FCFS | HRRN | SJF | SJF-FB |
|---|---|---|---|---|---|
| Average | 19.89 | 11.80 | 11.20 | 10.76 | 10.93 |
| 10% | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 20% | 0.28 | 0.00 | 0.00 | 0.00 | 0.00 |
| 30% | 7.84 | 2.33 | 2.08 | 0.65 | 0.86 |
| 40% | 12.45 | 5.53 | 4.83 | 2.38 | 3.05 |
| 50% (Median) | 17.30 | 8.42 | 7.58 | 5.15 | 6.27 |
| 60% | 22.10 | 11.40 | 10.55 | 9.00 | 9.92 |
| 70% | 27.20 | 15.27 | 14.34 | 13.55 | 14.22 |
| 80% | 33.42 | 21.15 | 19.88 | 20.36 | 20.46 |
| 90% | 43.20 | 29.79 | 29.03 | 29.78 | 29.95 |
| 100% (Max) | 94.47 | 71.90 | 73.58 | 91.13 | 78.18 |

**Table 3. Distribution of Wait Times by Strategy**

After demonstrating the impact that breaking the first-come-first-served rule can have on waiting times, it is important to examine the line cuts that are associated with each of these strategies. Line cuts are recorded by each user in the simulation while waiting in the queue. Each time a user is selected from the queue and assigned a computer, remaining users who arrived prior to the one just selected note having been skipped over. By the time they are assigned a computer, users have recorded the total number of times they were passed over in the queue.

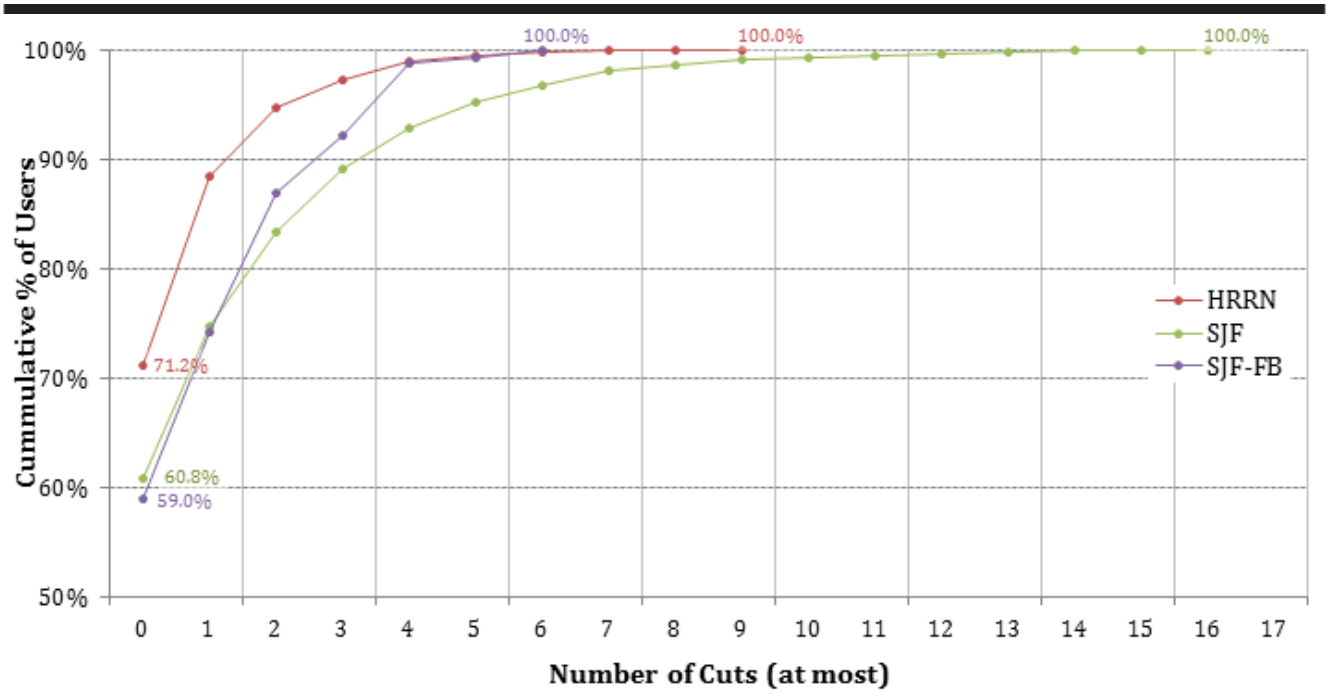


**Figure 3. Cumulative Distribution of Line Cuts by Queuing Strategy**

Figure 3 displays the cumulative percentage of users experiencing no more than the listed number of cuts for each non-FCFS strategy. The majority of users are not passed over at all under these strategies. However, there is a small minority of users that will be repeatedly cut in line. For instance, in our simulation, one unfortunate individual was passed over in the queue sixteen times under the SJF strategy. This user waited ninety-one minutes using this strategy as opposed to *only* fifty-nine minutes under the familiar FCFS waiting queue. Most customers would become upset upon seeing a string of sixteen people jump over them in the queue and get on a computer while they are enduring such a long wait. The HRRN strategy caused a maximum of nine cuts to an individual in this simulation. This user waited seventy-three minutes under HRRN versus only fifty-five minutes using FCFS.

Extreme examples such as those above are the exception. Under the HRRN and SJF-FB strategies, 99% of users were passed over at most four times while waiting in the queue.

## CONCLUSION

We have examined the simulation of several queuing strategies using a single month of computer usage data from the Southern Oaks Library. The relative performance difference between queuing strategies will depend on the supply and demand of computers at any given location. Clearly, at libraries with plenty of public computers for which customers seldom have to wait, the choice of queuing strategy is inconsequential. However, for libraries struggling with waiting times on par with those examined here, the choice can have a substantial impact.

In general, however, these simulation results demonstrate the ability of non-FCFS queuing strategies to significantly lower waiting times for certain classes of users without partitioning the pool of computers. These reductions in waiting times come at the cost of allowing high priority users to essentially cut in line. This causes slightly longer wait times for low priority users; but, overall average and median wait times see a small reduction.

Of course, for some customers, being passed over in line even once is intolerable. Furthermore, creating a system to implement an alternative queuing strategy may present obstacles of its own. However, if the need to provide for quick, short-term computer access is pressing enough for a library to create a separate pool of "express" computers; then, one of the non-FCFS queuing strategies discussed in this paper may be a viable alternative. At the very least, the FCFS-15 simulation results should give one pause before resorting to designated "express" and "non-express" computers in an attempt to remedy unacceptable customer waiting times.

## ACKNOWLEDGMENTS

The author would like to thank the Metropolitan Library System, Kay Bauman, Jimmy Welch, Sudarshan Dhall, and Bo Kinney for their support and assistance with this paper as well as Tracey Thompson and Tim Spindle for their excellent review and recommendations.

## REFERENCES

1. J. D. Slone, "The Impact of Time Constraints on Internet and Web Use," *Journal of the American Society for Information Science and Technology* 58 (2007): 508–17.

2. William Mendenhall and Terry Sincich, *Statistics for Engineering and the Sciences* (Upper Saddle River, NJ: Prentice-Hall, 2006), 151–54.

3. Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, *Operating System Concepts* (Hoboken, NJ: Wiley, 2009), 188–200.