

Case Study: Using VictorSchedule in a Library

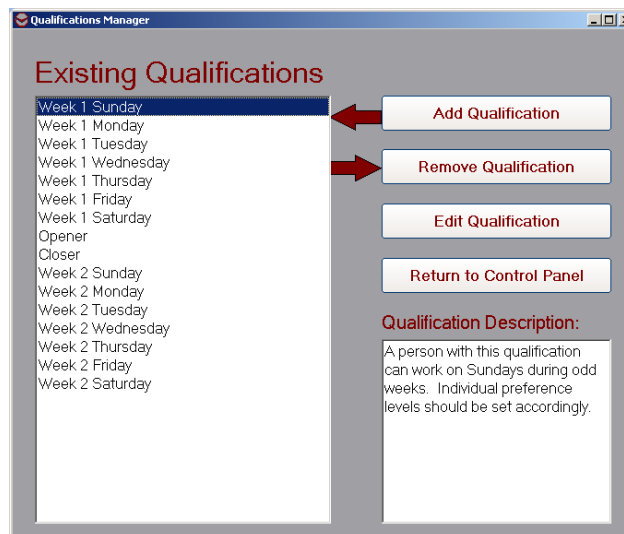
This case study illustrates how VictorSchedule can completely replace popular online scheduling services at a fraction of the cost. Not only that, VictorSchedule shows off powerful features that other solutions simply cannot match. These features allowed one mid-sized library to be more flexible in its scheduling and dramatically reduce the time taken to create schedules.

This case study is interesting because it shows the flexibility of VictorSchedule for getting the job done. No two organizations do their scheduling in the same way, but VictorSchedule can adapt to almost any scenario. This particular library was using an online scheduling system that didn't meet all of their needs, but was being used because accessing schedules online was deemed so important that they accepted a compromise. The compromise was that the system would not generate schedules automatically for them, and thus ended up requiring a significant amount of manual effort each week to release a schedule.

VictorSchedule met all of their scheduling requirements, and reduced the time spent getting a schedule published online from an average of 12 person-hours to less than a half hour per week. Let's see how it was done, and how some of VictorSchedule's advanced features paid huge benefits.

Step 0: Defining the Scheduling Problem

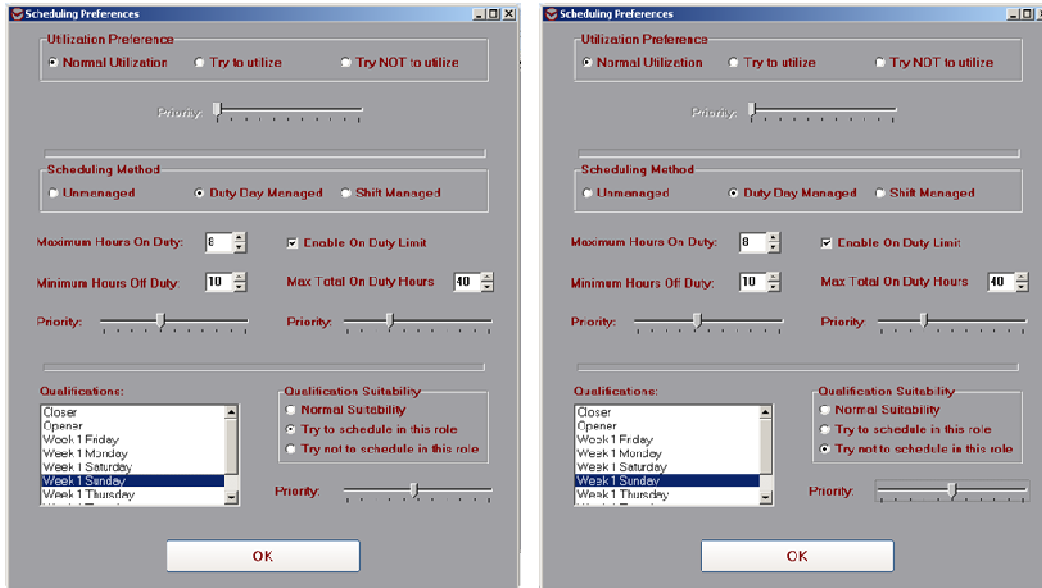
This library has a pool of salaried workers on the reference staff, each with a nearly set working schedule. They rotate some shifts on a bi-weekly basis to ensure that people get the desired number of shifts and to give workers variety in the days they work. During each day, there are a number of desks that need to be manned. The workers like to rotate desks every hour or so to keep things more interesting.



Screen Shot 1. The Qualification system utilized by the library. Each Qualification represents the ability to work on a particular day during each bi-weekly period. The additional qualifications (Opener and Closer) are assigned only to some senior librarians, and are always used in conjunction with other qualifications as Additional Criteria.

Step 1: Defining the Qualifications

An extremely flexible qualification system was created for this library. Since the workers are almost interchangeable, each qualification was used to represent availability to work on a particular day. Screen Shot 1 below illustrates the qualification system utilized in this case study. By using this qualification system, each person could be assigned a relative preference for each day in the scheduling rotation. We'll see how in a bit.



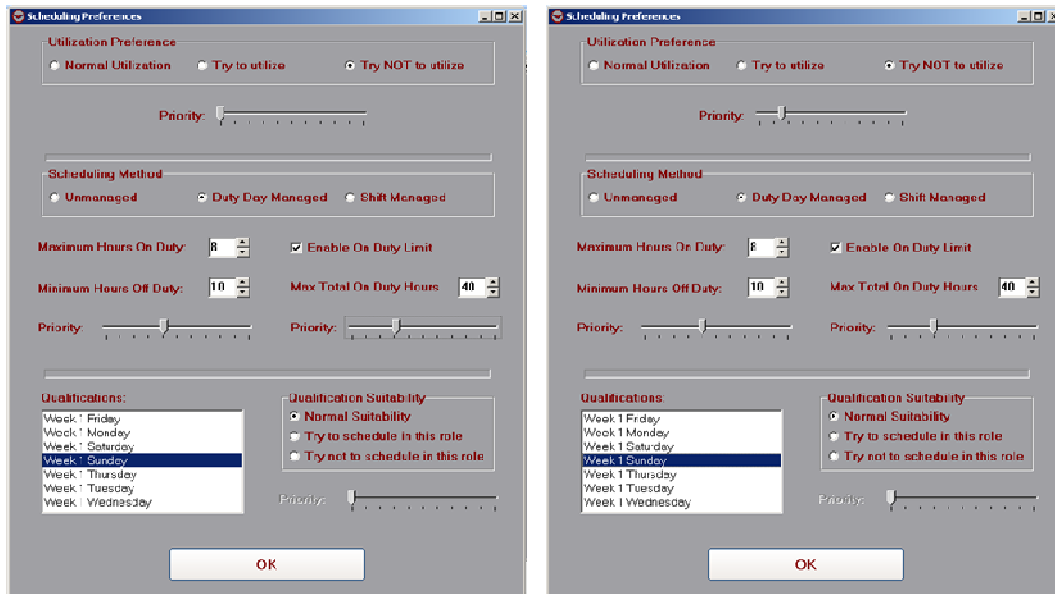
Screen Shot 2: These preference forms indicate the differences between the preferences defined for workers who either are or are not supposed to be scheduled on Sunday of the Odd-week scheduling rotation. As can be seen, the only differences are for the specific qualification 'Week 1 Sunday'. On the left (the person who is designated as a worker in this rotation), we see that they have a medium priority suitability to be used in the selected role. On the right, we see a similar preference for disuse in the selected role. This tells VictorSchedule how to utilize each of these people.

Step 2: Entering the Employees

Each employee at this library is relatively interchangeable. The chief difference is that certain senior librarians are designated as 'Openers' and 'Closers'. These people were given the corresponding additional qualifications defined in Screen Shot 1. The other difference between the employees is when they are able (and / or willing) to work. For this case study, these differences were captured by using the Qualification Suitability functionality of VictorSchedule. Each person was given a qualification for every possible day that they could be scheduled to work. After this, preferences were defined specifying the priority to give for either using a person or not using a person on a given day. This can be seen illustrated in Screen Shot 2.

One additional interesting thing about this case study is the library's use of substitute workers. This library has a pool of substitutes that it uses when regular workers are not available. The library seeks to minimize the use of substitutes, but will generally use a substitute rather than calling a person in on a day that they are not normally scheduled to work. This concept was captured by defining two Substitutes and giving them all qualifications except for

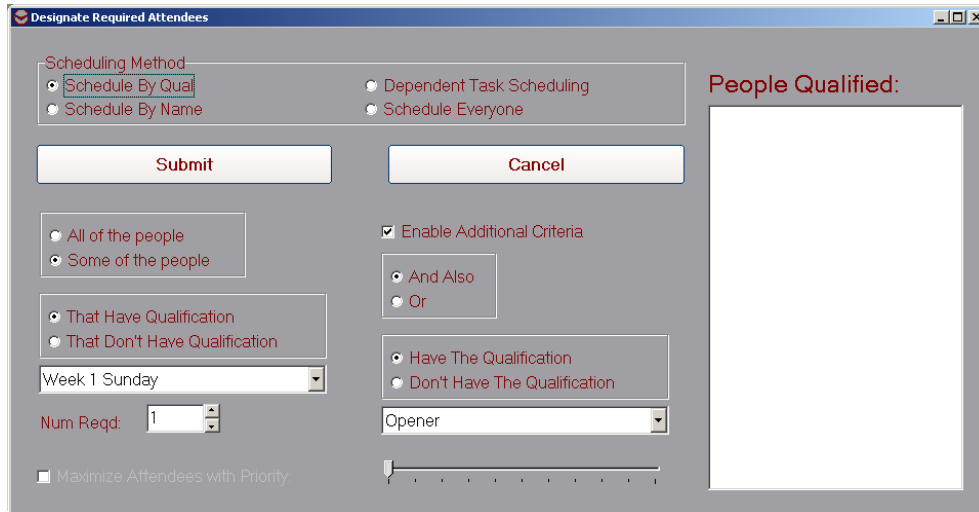
the Opener and Closer qualifications. Then, preferences were set up in such a way that the first sub would always be used before the second sub, and the second sub would be used before someone would be required to work on their off day. If both subs are given the same priority, VictorSchedule will not know that one sub should be preferred to having two subs on a given day. Screen Shot 3 shows how the preferences were set up for the two substitutes in relation to each other, as well as in relation to the preferences assigned to normal workers.



Screen Shot 3: These preference forms indicate the differences in the preferences specified for the substitutes. On the left is the first substitute person. We see that they are given a general preference for disuse, with a very low priority. On the right we see the preferences defined for the second substitute. Again, they are given a general preference for disuse, but with a higher priority. By defining priorities in this way, VictorSchedule will attempt to use Sub 1 before using Sub 2. Since the priorities are lower than those given to the workers in Screen Shot 2, a Sub will be used before someone is used on a day that they don't want to work. This is as desired by the library management team.

Step 4: Defining the Schedules

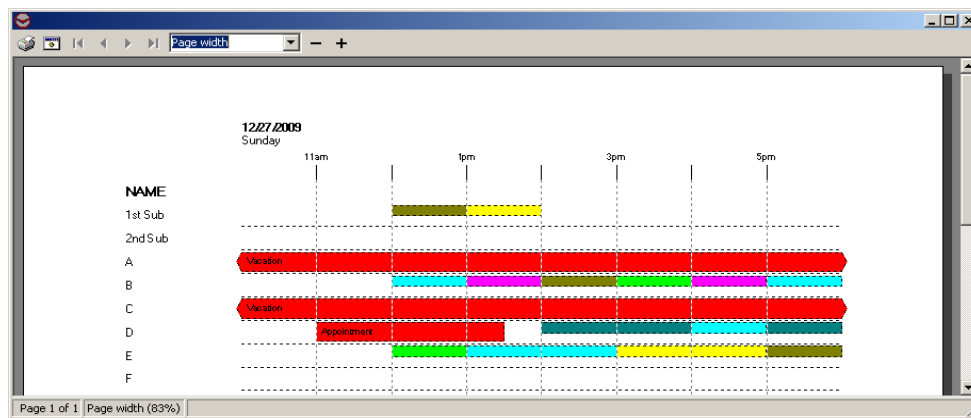
As mentioned before, each day is divided into a number of hour-long tasks corresponding to an hour manning one of the reference areas in the library. Thus on Sundays, when the library is only open for 6 hours and there are 3 desks to man, there are 18 tasks corresponding to desk manning. In addition, certain tasks are defined for some workers to perform additional duties, such as meetings and management tasks (such as building schedules). For the most part, these tasks are very straightforward, and can be generated fairly quickly through the use of templates. Required Qualifications are assigned to each task according to the day. Screen Shot 4 shows how qualifications are defined for one typical opening task.



Screen Shot 4: The People Required designation for a typical opening task on a Sunday. This qualification is interpreted as “1 person that has the qualification *Week 1 Sunday* and also has the qualification *Opener*”. Since this task has the *Week 1 Sunday* role associated with it, the qualification suitability preferences as defined in Screen Shots 2 and 3 are utilized by VictorSchedule in deciding who should be scheduled for each particular task.

Step 5: Building the Schedules

Now that all the tasks have been defined, it is time to test how well we’ve captured all of the desired scheduling behavior. One week of scheduling for this library takes about 2 minutes to be automatically built by VictorSchedule. The resulting schedules are exactly as desired, as can be seen in Screen Shot 5.



Screen Shot 5: This shows one Sunday’s schedule during the holiday season. Person’s A, B, C, D, and E are normally scheduled to work Sundays. Two of them are on vacation, however, and one of them has an appointment, however, leaving the library short staffed. As can be seen, VictorSchedule has chosen to utilize Sub 1 instead of calling in someone (such as Person F) on their off-day. Other people are fully utilized during this busy time. Library management rates this behavior ‘Exceptional’, and was exactly what they were looking for in a scheduling product. What used to take many hours to produce has been reduced to a few minutes a week.

Step 6: Publishing the Schedules

The library management team takes a few minutes on Wednesdays to enter in people's availability restrictions for the following week. Then, they utilize a template to generate the schedules for the week. After that, they generate the schedules using VictorSchedule's AutoPopulate functionality. Then, draft schedule products (such as the Timeline View in Screen Shot 5) are printed out and posted in the Staff room. This process generally takes about 20 minutes to complete by a proficient scheduler (once the base templates have been created once). Any changes are made to the schedules (usually using the Problem Solver view), and then final schedules are published on Thursday at Close of Business.

Final schedules are published to the web in just a few minutes and a few mouse clicks. This is done from the Web tab of the Products form. Using an existing Google[®] account, all of the library's schedules can be posted (with private visibility) on the Google[®] Calendar service. This provides many advanced features free of charge to the library, such as automated notifications, reminders, and RSVP tracking. Any changes that are made during schedule execution can be re-published to the web, and only the affected parties are notified of the differences. Also, each employee can be granted different levels of access to the library's calendars on Google[®], and some changes are trusted to the employees such that management does not have to be involved in all details of the scheduling process.

The Bottom Line

VictorSchedule allows this library to completely replace its old online subscription based scheduling service for a fraction of the price. Between the time saved each week on scheduling, and the monthly subscription price of their old service, VictorSchedule paid for itself within the first month of its use. We think you'll be happy with VictorSchedule, too. Try it absolutely free for 30 days with no risk whatsoever. Just visit us at <http://victorware.com/victorschedule.html>. Click the Try It link, and take the pain out of scheduling.