Case Study: Using VictorSchedule in a Quick Service Restaurant(QSR)

Mark is the owner of a popular sandwich franchise located in Broomfield, Colorado. His location is near several large office buildings, and as such lunch is the most important time of day for him. He also draws limited morning volume and a small evening volume from a nearby housing subdivision. His store is open from 8:00 in the morning until 8:00 at night during the week, but only during lunch hours on weekends due to limited traffic. Margins are tight, and Mark must make the most out of his employees and himself. He was looking for a tool that would help him create and distribute schedules more efficiently, and that also provides a way to track hours worked more easily. We introduced him to VictorSchedule, and he couldn't be happier.

This case study takes a look at how powerful VictorSchedule can be in a Quick Service Restaurant (QSR) environment. Mark previously used an Excel spreadsheet to create the schedule and simply printed out a copy and posted it behind the counter. The spreadsheet worked to some degree, but employees found it confusing and it required constant upkeep when staff changed. It also required Mark to manually schedule people to shifts, and as such was very error prone. Here we'll take a look at the simple setup that it took to get Mark up and running with VictoSchedule and the reasons he says he'll never go back to the spreadsheet again!

Step 1: Define the Qualifications

The first place to start when customizing VictorSchedule for your organization is with the Qualifications. Qualifications are what make a person a candidate for a particular task. Often times this is a little bit confusing at first, especially in settings like restaurants where there isn't really a formal qualification system (unlike in the healthcare industry). The best way to get started is to think of each of the tasks or shifts you need to schedule, and then think of which people you might assign for the task. When you have this list of people for each task, then think about what they have in common that makes them candidates to perform the task. Whatever that 'thing they have in common' is, should probably be captured as a Qualification. Keep in mind that people can have more than one qualification! In a Quick Service Restaurant there generally aren't many qualifications to worry about. Mark wanted to take advantage of some of VictorSchedule's more powerful features, however, so he defined more qualifications than QSR's traditionally define. This is part of the reason that makes Mark's environment a good Case Study. These are the qualifications that we defined for Mark's setup:

1. Over 18 Years Old: Regulations require someone be over 18 to operate some kitchen equipment, such as the slicer. Mark wanted to keep track of this since he often has minors employed in his shop.
2. Cash Register Trusted: Not all of Mark's employees are allowed to work the register when Mark is not at the shop. For certain shifts, he wants to ensure the right people work them.
3. Supervisor: Right now, Mark is the only one with this qualification. When he expands hours, however, he wants the flexibility to give this qualification to someone else so that his own workload is reduced.

4. Baseline Employee: Since not all of Mark's hires have another qualification, it was important to define a qualification that everyone has. This qualification is also useful for specifying that some people (such as Mark and other more senior employees) should NOT be used in shifts that don't require more qualifications.

Now that the qualifications were established, it was time for Mark to enter people into the staff directory. Mark has only 8 employees (besides himself), three of whom are considered full time.
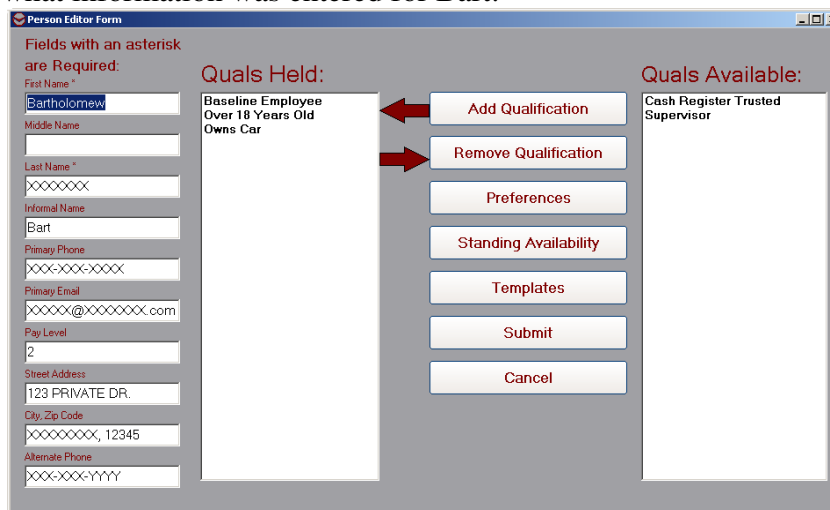
Step 2: Entering the Staff Members

Before entering information about people, it is useful to define exactly what information you want to store. VictorSchedule gives you four customizable fields, which are accessed from the Preferences / Settings area. Take a look at Screen Shot 1 to see what information Mark decided to store about his employees.

Screen Shot 1: These are the custom fields that Mark set up for his staff. These fields are available for each person in the directory.
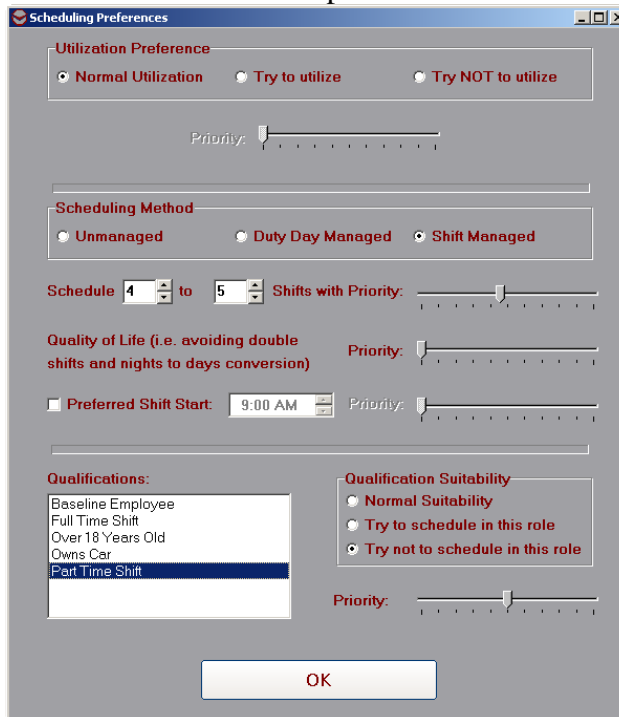
After setting up what information should be stored about each person, it was time to enter in all of Mark's employees. We'll look in detail at all the steps that went into defining one of them. To enter new people into VictorSchedule, click on 'Manage Directory' from the Control Panel. This brings up the Directory Manager form. Click on 'Add to Directory'. In Screen Shot 2 you can see what information was entered for Bart:

Screen Shot 2: The initial setup for one of Mark's employees, Bart. All personal data has been changed for this case study. Note that Bart is over 18 and owns a car, but is not allowed to work the cash register.

One key thing to keep in mind as you assign qualifications to people is that you should give a person all of the qualifications that you would ever consider using a person for, even as a last resort. When we define Preferences for a person we can set up how we want to use the person. Also, keep in mind that whatever email is entered in the 'Primary Email' address block will be used to send schedules, so make sure you enter the right address!

Once the basic information has been entered, it is time to enter the person's scheduling preferences. Here, a brief discussion of 'shift managed' versus 'duty day managed' is in order. For most applications, making people shift managed is most appropriate. This is because most organizations schedule by shifts. Some organizations, however, schedule in terms of independent tasks and meetings, (which are generally of shorter duration than shifts). The rule of thumb is that if your organization has defined shifts where a person is typically assigned to work one shift per day, then making personnel 'shift managed' is most appropriate. If your organization schedules people by individual meetings or tasks, then making them 'duty day managed' is usually most appropriate. One notable violation to this rule has been seen in the field of nursing, where shifts are often 24 hours or longer. In such a case, control over on and off-duty times is important, and duty-day managed nurses makes a lot of sense. For Mark's restaurant, however, the clear choice was to make all employees 'shift managed'. We'll also see how you can define additional tasks that must be performed within the shift.
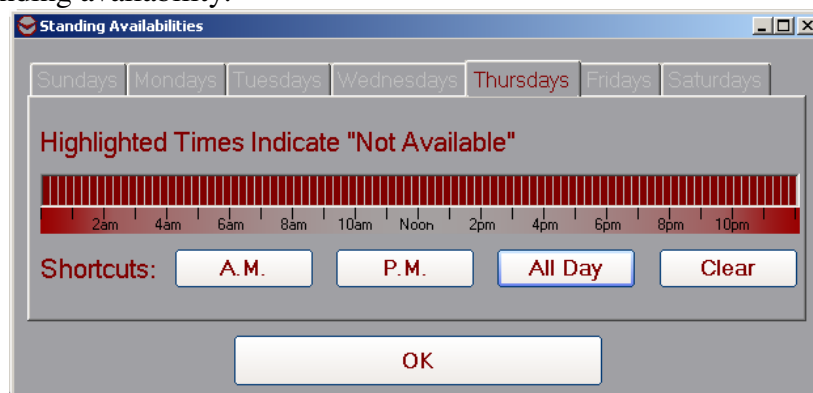

Screen Shot 3: Scheduling preferences for Bart, a typical employee for this case study. Note that we have defined the number of shifts for Bart to work as a full time employee. Since in this Case Study there is not a problem with rolling from night to day shifts, the Quality of Life score is not important to set. It is important, however, to set the suitability of Bart for working different types of shifts. Since Bart is a full-time employee, note above that we have set his suitability for working a Part Time Shift such that VictorSchedule will try to avoid using him for that role.

When defining scheduling preferences, we have the option to define a preferential use or disuse of each person. For this case study, we never had a need to do so, and all employees were left as 'Normal Utilization'. We also have the option to specify whether a person is 'Unmanaged', 'Duty Day Managed', or 'Shift Managed'. As discussed earlier, all employees in this case study were made 'Shift Managed'. Another choice we have when defining preferences is the 'Quality of Life' priority. Depending on where we set the priority controls how hard VictorSchedule will work to avoid double shifts and night-shift to day-shift transitions. Specifically the software will try to avoid assigning shifts with less than a 12 hour turn time between them. For this Case Study, there is enough downtime between shifts (and AM and PM shifts overlap, so nobody could be scheduled for both) the Quality of Lift score is not important.

The last thing to mention about setting up scheduling preferences is the 'Qualification Suitability' area. In this example, Bart has normal suitability for all of his qualifications except for 'Part Time Shift' qualification. For this qualification we specify that we should avoid scheduling Bart in part-time shifts. We do a similar thing for each employee depending on whether they are a full-time worker or a part-time worker. For some employees we also defined that they should not be used in a Cash Register role, for example. Doing this means that VictorSchedule will try to find someone else unless nobody is available. This is a powerful mechanism to allow VictorSchedule to choose the most suitable workers first, and only use others as a last resort. All of this can be done according to priorities you control.

Once we had setup the scheduling preferences for Bart, it was time to define his baseline availability. To do this, click on 'Standing Availability' from the Person Editor form. On this form, there is a tab for each day of the week. To define when a person is available and not available, simply highlight the times that they are unavailable on a given day. Whenever a new schedule is created, people will be marked as unavailable according to their standing availability settings. You can of course edit, change, and make exceptions to their availabilities at any time. We'll discuss this a little more later. Take a look at Screen Shot 4 to see an example of how we set up Bart's standing availability.

Screen Shot 4: An example of setting standing availability for a person. In this example, Bart is marked as not available on Thursdays. When defining standing availabilities, keep in mind that if you have a shift that spans days (i.e. a late shift) a person must be marked available for the entire shift to be considered. This is not a factor in the QSR setting of this Case Study.

Now we have a pretty typical person defined for this environment. Here's where Mark started to make good use of templates. By selecting the 'Templates' button on the Person Editor, you can save the current person as a template to be used when defining future people. By saving a person as a template, you save all of their preferences and qualifications. This is very handy when you have people with similar roles and preferences in your organization. Next time around, all Mark has to do is select 'Templates' and then apply the template of his choice to the person he's editing. Note that standing availabilities are NOT saved as part of a person template, since those generally are different for everyone.

Adding the remaining people in the organization is much like adding the first one, but the process gets much faster through the use of templates. All in all, Mark defined 4 person templates covering different roles as well as part time versus full time employees. He would use those templates as the baseline when entering in a new person and then just tweak as necessary. It took about 15 minutes for Mark to enter in all of his employees. Since this QSR scheduling situation does not have any resources that need to be scheduled, we are now ready to start defining schedules.

Step 3: Defining the Shifts

Much like most QSR setups, Mark's restaurant has only a few shift schedule patterns. He uses one pattern for weekdays, and another pattern for weekends. Here are the shifts that Mark has defined for week days:

- AM First Off: 8:00 – 2:00, if things are slow this individual gets sent home first. Usually taken by a part time employee.
- AM Full Shift: 8:00 – 4:00, generally filled by a full time employee.
- Weekday Supervisor: 10:00 – 6:00, always filled by Mark.
- PM First Off: 11:00 – 3:00, filled by a part-time worker in general.
- PM Full Shift: 12:00 – 8:00, generally filled by a full time employee.

By defining shifts in this way, Mark ensures heavy coverage during the lunch rush and lighter coverage during lighter periods. In addition, he has the ability to send the 'First Off' shifts home early when times are especially slow.
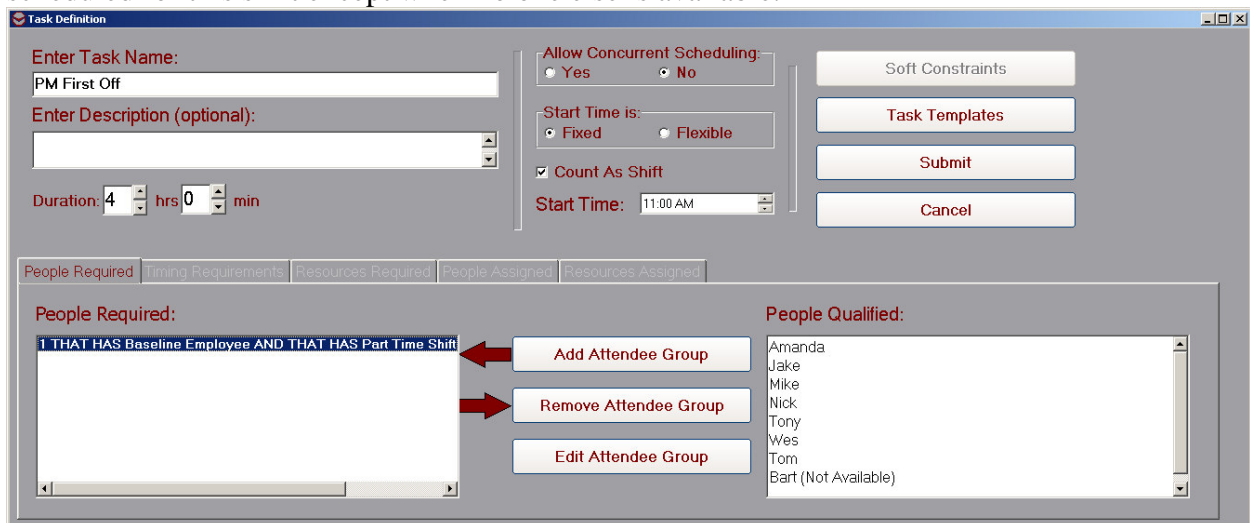
Mark also assigned some additional duties, however, which is what makes this QSR setup different than many other small restaurants. Specifically, Mark wanted to ensure that he always has someone (other than himself) that is over 18 years old on hand to work the slicer, and that he always has someone besides him to work the cash register. To account for these additional duties, Mark defined some 'Concurrent' tasks as follows:

- Cash Register 1: 8:00 – 11:00.
- Slicer 1: 8:00 – 12:00.
- Cash Register 2: 11:00 – 2:00.
- Cash Register 3: 2:00 – 8:00.
- Slicer 2: 4:00 – 6:00.

By defining the times for the duties as he has, these tasks can be assigned to people in different shifts as appropriate. It is important to remember, however, that when using this technique you should ensure that the 'Efficiency Premium' is set to a mid-level value at least. You can find the efficiency premium in the Building Schedules tab of the Preferences / Settings area. The

efficiency premium controls how hard VictorSchedule should work to ensure that the minimum number of people possible are used to fill a schedule. Setting the efficiency premium has an effect not only on Concurrent Tasks, but also on tasks with multiple required qualifications. If Mark, for example, had defined a single shift with one requirement for someone more than 18 and a separate requirement for someone Cash Register Trusted, setting the efficiency premium would control how hard VictorSchedule would work to find a single person that had both qualifications (thus requiring one person to fill the shift instead of two). This would potentially be another way that Mark could have accomplished nearly the same result he achieved by using Concurrent Tasks. Mark, however, wanted to use the Concurrent Task feature, and so we set up his shifts to use this paradigm. When doing so, the efficiency premium causes the Concurrent Tasks to be automatically assigned the same people for both the concurrent task and an existing shift. See Screen Shot 8 below for an example of how this works.

Since Mark took advantage of the 'Qualification Suitability' feature of VictorSchedule, it was important that he defined shifts so as to utilize the power of this feature. Screen Shot 5 below shows how we defined the PM First Off shift. Note that it specifies as a staffing requirement that someone with the Part Time Shift qualification be scheduled for the task. Recall that (in Screen Shot 3) we specified that a full time employee such as Bart should not be scheduled for this shift except when no one else is available.
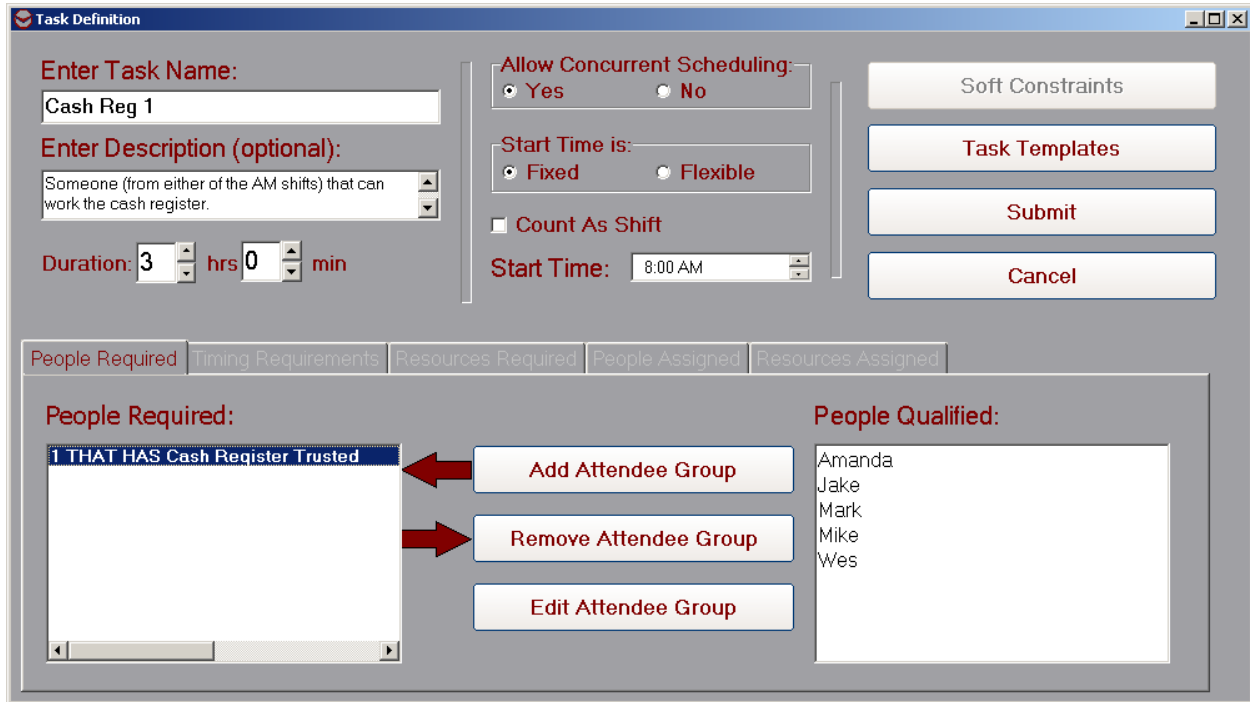

Screen Shot 5: The definition of the PM First Off shift. Note that 'Count As Shift' is selected so that VictorSchedule can manage the shift counts for people. The start time is set as Fixed, and 'Allow Concurrent Scheduling' is set to No. Also note that the requirements for this shift specify someone who is capable of working Part Time Shifts must be assigned to this task. When coupled with individual scheduling preferences (specifically the suitability setting), this is a very powerful and flexible way to specify who should be scheduled for particular shifts.

All of the shifts for Mark's restaurant were created in much the same fashion. The Concurrent Tasks described above, however, are slightly different. Screen Shot 6 shows how the Cash Reg 1' task was defined. Note that it is specified as 'Concurrent', meaning that a person can (and should) be scheduled for something else at the same time as this shift. Because of the time this task was defined (i.e. from 8:00 to 11:00) this causes VictorSchedule to try to assign someone to either of the two morning shifts who has the 'Cash Register Trusted' qualification so that they can fulfill the shift requirement as well as the Cash Register Concurrent requirement.

By making use of this feature, Mark is able to ensure that VictorSchedule will automatically schedule people with the desired qualifications without having to tie the qualifications to a particular shift.
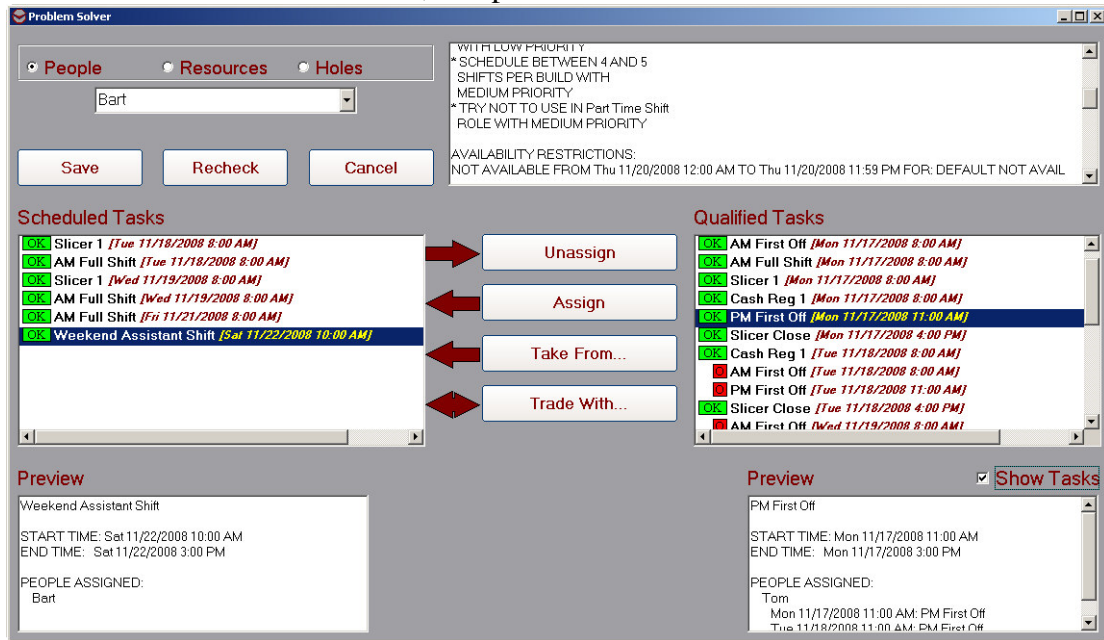

Screen Shot 6: Another example of a task defined in this case study. This shift is different from others that have been discussed previously because it is a 'Concurrent' task, i.e. people can (and should) be scheduled for this task while they are performing another task (such as their assigned shift). Note that the 'Allow Concurrent Scheduling' selection is 'Yes', and that the task should NOT be counted as a shift. In order to make use of the concurrent task functionality of VictorSchedule, it is important to set the Efficiency Premium in the Building Schedules tab of the Preferences area.

Step 4: Using VictorSchedule

Once Mark had defined all of the people and shifts for his restaurant, it was time to start putting VictorSchedule to use. Mark keeps a notebook to track his employees' off-time requests, and before building the schedule he enters this information in the Availabilities area. Then, he creates a week's worth of schedules. This is very easy to do since most of his weeks are the same schedule, and he has created a multi-day template for it. Then, he copies the schedules he's working on to the Schedule Workbench, clicks 'Auto Populate', and in less than a minute he has a completed schedule ready to go.
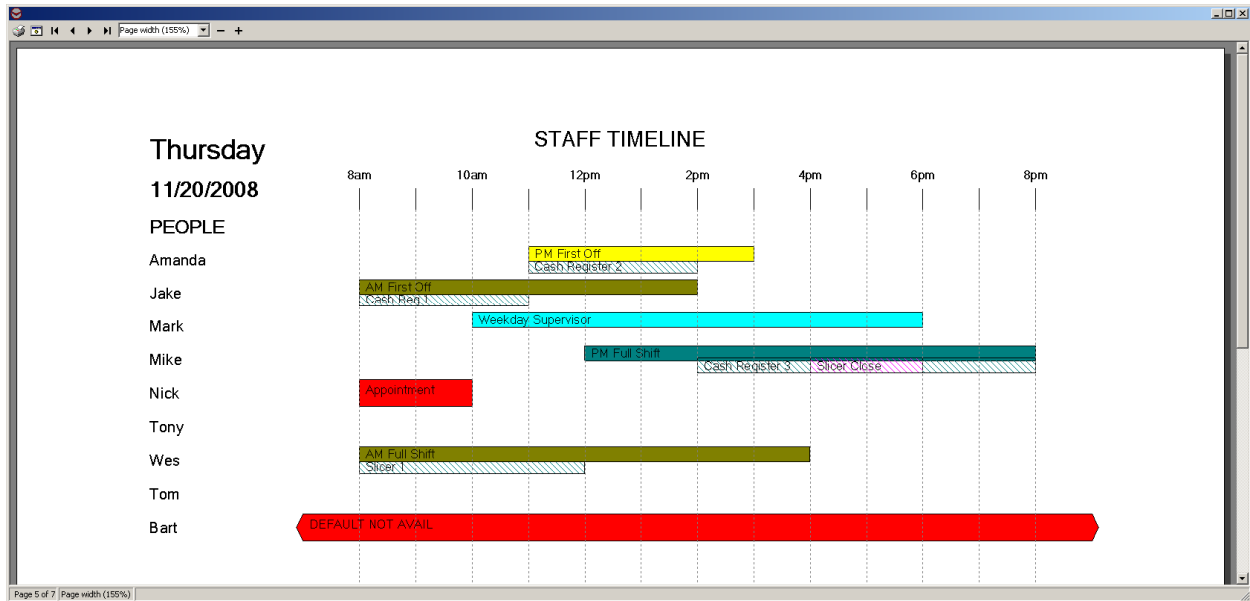
Before publishing the schedule, Mark takes a look at the Problem Solver view. He quickly scrolls through each person to ensure that they have a suitable workload. If there are any tweaks he wants to make, he can make them very quickly in this view as well. Screen Shot 7 shows the Problem Solver view, and how easy it is for Mark to make trades and reassignments. In one convenient view, Mark can see what everyone is scheduled for, what they are qualified for (but not scheduled for), who is available for trading for any shift, as well as people's availability restrictions and shift preferences.

After making sure people are scheduled the way he wants, Mark then selects 'Publish All' from the Schedule Workbench form. This allows him to create all the scheduling products he needs. First, he selects the 'Individual Schedules' tab and prints up an individualized schedule for each of his employees. He puts this product in their cubby hole. Some of his employees also have email, and he can quickly email their individualized schedules to them. Last, he selects the 'Timeline View' tab, and prints out a schedule timeline for the week.

Screen Shot 7: The Problem Solver form looks a little bit intimidating at first, but it doesn't take long for people to love it! A wealth of information is available in an instant. Above, for example, we can see the Bart's workload as well as when he's not available. We can see what other tasks he's qualified for (with icons indicating if he's available and if the task overlaps something he's already scheduled for). For each task he's qualified for, you can see at a glance who is currently scheduled for it and what else they're scheduled for. Trading shifts is just a button click, and any problems or warnings are displayed as appropriate.

The timeline view is extremely useful for Mark during the week, an example of which can be seen in Screen Shot 8. Mark keeps a printed copy of each day's timeline in his office, and can quickly see what each of his employees is doing at all times. This helps him make decisions as far as who to send home early if times are slow, and (on occasion) who to call for assistance if he's feeling understaffed due to a catering job or some other event arising. He also takes notes on the timeline view for the precise times that his employees work. At the end of the week, he transfers these notes back into VictorSchedule so that the shifts accurately reflect any changes that were made during the week (i.e. what actually happened with the schedule). This allows Mark to generate a report on hours worked for each person, which he can very simply export to a spreadsheet. With a few simple macros in Excel, his payroll is done too!

Thursday

11/20/2008

STAFF TIMELINE

Screen Shot 8: The timeline view is of immense help to Mark as the schedule is executing. He can see at a glance what each of his employees is scheduled for, including off-time. Many other products are available for viewing the schedule as well. Each tab of the View Products form represents a type of output that can be produced. VictorSchedule provides an email notification system, individual schedules, group schedules, HTML output, and summary report products in addition to the timeline shown above. Each product is fully customizable.

Summary

The bottom line is that VictorSchedule is an excellent scheduling solution for Mark in his QSR franchise. He saves hours per week, the products are much easier to interpret than his old scheduling method, and the hours-worked data is available for easy export into a spreadsheet application. Mark is very happy, and we think you'll be happy with VictorSchedule too. Try it absolutely free for 30 days with no risk whatsoever. Just visit our website at http://victorware.com/victorschedule.html. Click the Try It link, and take the pain out of scheduling.