CONTACT CENTRE STAFF SCHEDULING AT PHILLIPS, HAGER & NORTH INVESTMENT MANAGEMENT LTD.

by

ANA MARIA KIM

B.Comm., The University of British Columbia, 1995

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIRMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

(Department of Commerce and Business Administration)

We accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

April 2000

© Ana Maria Kim, 2000

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Commerce and Bus. Admin

The University of British Columbia
Vancouver, Canada

Date Apr 27, 2000

DE-6 (2/88)

ABSTRACT

Phillips, Hager & North Investment Management Limited is a privately owned company providing investment management services to clients. The company has established a reputation for providing an exceptional level of customer service. In recent years, with the introduction of retail mutual funds, the number of clients has increased dramatically. Clients purchasing mutual funds have more extensive service requirements than the traditional institutional client. A call centre was formed to satisfy the heavier service requirements of mutual fund clients. However, the call centre has been unsuccessful in consistently achieving target service levels. A project was initiated to analyze the call centre's operations and its relationship to service levels. At the onset of the project, the call centre lacked the controls in place to enable targets to be met. A workforce management tool was developed and implemented to provide the call centre with a systemic mechanism for achieving target service levels. The tool has been a used by Contact Centre management to support its proposal to increase resources. Also, it currently being used to generate staff schedules.

TABLE OF CONTENTS

Abstract ii
List of Tables v
List of Figures vi
Acknowledgement vii

I. PROJECT BACKGROUND 1

II. CALL CENTRES 3
- 2.1 Architectural Framework 3
- 2.2 Terminology 4

III. CALL CENTRE MANAGEMENT 5
- 3.1 Establishing Service Levels 5
- 3.2 Forecasting Call Volume 6
- 3.3 Computing Staffing Requirements 6
- 3.4 Staffing Trade-Offs 7

IV. CONTACT CENTRE AT PH&N 8
- 4.1 Current Operations 8
- 4.2 Service Level Performance 11

V. ANALYSIS OF SERVICE LEVELS 13

VI. METHODOLOGY & APPROACH 16
- 6.1 Commercially Available Applications 16
- 6.2 Project Scope 17
  - *6.2.1 Analysis of the Existing Forecasting Model 17*
  - *6.2.2 Validation of Erlang-C Assumptions 17*
  - *6.2.3 Construction of an Efficient Scheduling Policy 18*
  - *6.2.4 Development of a Call Centre Workforce Management System 21*
  - *6.2.5 Implementation 21*

VII. ANALYSIS & RESULTS 22
- 7.1 Forecasting Model 22
- 7.2 Scheduling Heuristic 27

VIII. CONTACT CENTRE STAFF SCHEDULING TOOL 30
- 8.1 Main Menu 30
- 8.2 Preferences 31
- 8.3 Generating Schedules 32
- 8.4 Viewing Past Performance 35
- 8.5 Editing Default Values 37

IX. IMPLEMENTATION 38

X. CONCLUSION & FUTURE ISSUES 40

Bibliography 41
Appendix I Skills-Based Routing 43
Appendix II ACD Output 45
Appendix III Scheduling Heuristic 47
Appendix IV Scheduling Heuristic Code 52

LIST OF TABLES

Table 1: Staffing - Service Tradeoffs 7
Table 2: Shift and Lunch Combinations 8
Table 3: Summary of ACD Output on Agent Activity 10
Table 4: ACD Output on Queue Activity 11
Table 5: ACD Queue Report for December 1, 1999 13
Table 6: ACD Agent Report for December 1, 1999 14
Table 7: Abandoned Calls in 1999 18
Table 8: Results of Call Volume Predictions Presented to CSRs 25
Table 9: Average Call Length 26

LIST OF FIGURES

Figure 1: Call Centre Architecture 3
Figure 2: Total Service Time 4
Figure 3: Typical Breakdown of a Call Centre's Operating Costs 5
Figure 4: Staffing – Service Tradeoffs 7
Figure 5: Contact Centre Architecture 9
Figure 6: Service Levels Achieved During Week of November 1, 1999 11
Figure 7: Blocked Calls in October 1999 18
Figure 8: 1998 vs. 1999 Actual Average Call Volume per Period 22
Figure 9: 1998 vs. 1999 Actual Average Monthly Call Volume 23
Figure 10: 1998 vs. 1999 Actual Average Daily Call Volume 23
Figure 11: Adjusted Predictions 23
Figure 12: Average Monthly Predictions vs. Actual Call Volume 24
Figure 13: Average Daily Predictions vs. Actual Call Volume 24
Figure 14: Staffing Levels for August 9, 1999 26
Figure 15: Generated Schedule, Example 1 28
Figure 16: Generated Schedule, Example 2 29
Figure 17: Main Menu 30
Figure 18: Agent Preferences 31
Figure 19: View of Lunch Prefrences 31
Figure 20: Generating Forecasts 32
Figure 21: Modifying Call Volume Forecasts 33
Figure 22: Agent Details for Schedules 34
Figure 23: Shift and Lunch Restrictions 34
Figure 24: Schedule 35
Figure 25: Charting Past Performance 36
Figure 26: Predicted Calls vs. Actual Calls 36
Figure 27: Editing Default Values 37

ACKNOWLEDGEMENT

I would like to thank the Centre for Operations Excellence for giving me the extraordinary opportunity to work on an applied project for my thesis. I would like to thank Paul Hiom from the Centre for Operations Excellence for seeing a business need at Phillips, Hager & North Investment Management Ltd and being great project manager. I would also like to thank Stephen Jones from the Centre for Operations Excellence taking his time to give me input.

I would like to thank Professor Martin Puterman for supervising my thesis. My special thanks to him for convincing me to enter the program. I would like to thank Professor Shelby Brumelle who kindly agreed to be my external advisor.

I would like to thank Phillips, Hager & North for providing the support I needed to complete the project. Key people I would like to thank at Phillips, Hager & North are Ron Matthews, Pamela Brault, Jon Merrifield, Andrea Peterson, Scott Bisillion and Anna Catania.

I. PROJECT BACKGROUND

This thesis was carried out in the Centre for Operations Excellence (COE) in the Faculty of Commerce and Business Administration at the University of British Columbia. It reports the results of a project with COE partner, Phillips, Hager & North Investment Management Limited (PH&N).

PH&N is the second largest independent investment counseling firm in Canada, managing over $30 billion of assets [1]. The primary business activity is providing professional investment management services to clients.

PH&N has established a reputation for providing an exceptional level of customer service. However, a shift in the nature of PH&N's customer base has had a serious impact on customer service levels. PH&N's customer base traditionally included only institutional clients. Since the company was formed in 1964, the number of institutional clients has grown slowly to approximately 150. Institutional clients represent a small portion of PH&N's current customer base, but they own most of the assets under PH&N's management. In addition institutional clients rely entirely on PH&N for portfolio management, which results in a low frequency of contact. Because institutional clients generate a high level of revenue and have low service requirements, providing a high level of customer service has been uncomplicated.

The recent introduction of retail mutual funds has included a new type of client into PH&N's customer base. The minimum investment required for clients that are investing in retail mutual funds is only $25,000 in contrast to $250,000 for institutional clients. As a result, the number of clients investing in PH&N's portfolio of mutual funds has increased rapidly to approximately 15,000 in just a few years. Unlike institutional clients, mutual fund clients are responsible for managing their own portfolios. Mutual fund clients contact PH&N frequently for investment advice, account adjustments and account enquiries. The intense service requirements and significant number of mutual funds clients has made traditional means of providing high service levels costly.

PH&N's customer service strategy for mutual fund clients was to create a call centre and to implement Computer Telephony Integration (CTI) technology [2]. A call centre provides mutual fund clients with a single point of contact. CTI allows PH&N to maintain the tradition of knowing each customer on an individual basis even with mutual fund clients. As the name implies, CTI connects telephones with computers. CTI will retrieve information stored in a database about an incoming caller and send this to the call centre agent's screen before the agent accepts the call. Agents can read a brief history of the caller's past enquiries prior to answering the call then greet the customer by name.

Although high service levels were established for the call centre, targets are not being met consistently. This is undermining the company's customer service oriented reputation. More importantly, PH&N's most powerful marketing tool is being threatened.

A model, simulating the manner in which PH&N handled all client contact, was developed to evaluate possible changes to client contact architecture to improve customer service as well as to increase potential revenue generation (see Hiom [3]). The architecture encompassed more than just the call centre and incorporated more than the two client types previously discussed. The study indicated that the call centre lacked the controls in place to enable targets to be met.

As a result, the call centre's operations and its relationship to service levels were analyzed in depth. A workforce management tool to improve call centre efficiency and ultimately customer service was developed and presented to PH&N on November 3, 1999. Users were subsequently trained and the tool was implemented in December. The tool was not used for staff scheduling until March due to an unexpected increase in call volume. However, the call centre was able to use the tool to propose for more resources. The tool showed that at least twice as many agents would be required in order to meet the call centre's target service level for the increased call volume.

II. CALL CENTRES

This chapter provides an introduction to call centres. The first section presents the typical set of technology and framework used in operating a call centre. The next section covers call centre terminology to be used throughout the remainder of this thesis.

2.1 Architectural Framework

There are three general types of call centres: inbound, outbound and a combination of the two [4]. Inbound call centres consists of a group of agents accepting incoming calls typically to provide service such as placing orders, answering statement enquiries, etc. Agents in outbound call centres place outgoing calls mainly to sell a service or product.

Telephone lines connected to the call centre are called trunks. Personal Branch Exchange (PBX) is the hardware switching technology used to connect a trunk to an agent's telephone. An Automatic Call Distributor (ACD) is used to distribute each call to the next available agent as well as to collect and process call statistics [5]. Centrex ACDs have PBX functions built-in as part of the ACD software.

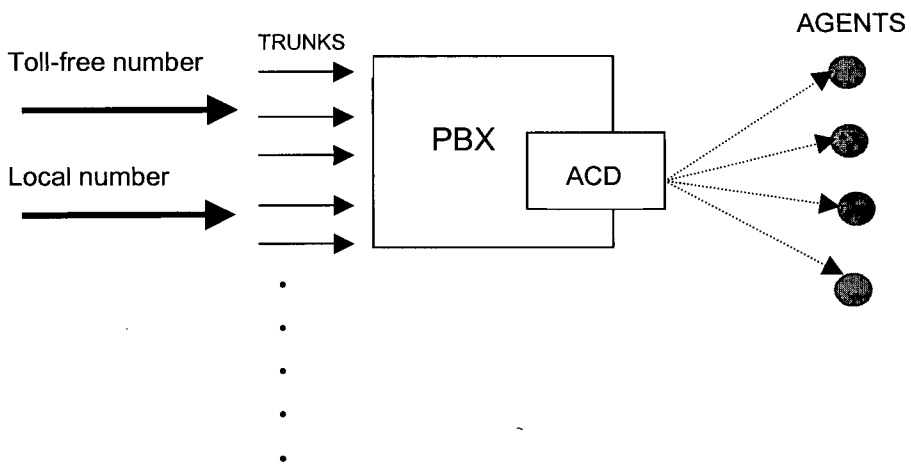


***Figure 1**: Call Centre Architecture*

Call centres that have agent groups with different skill sets can use skills-based routing, where the ACD is programmed to distribute a call to the queue serviced by agents whose skills are best matched to the caller's needs. An Interactive Voice Response unit (IVR) is typically required in addition to the ACD to perform the sorting of incoming calls. IVR technology is used to direct callers to the option that best describes his or her request using the buttons on a touch-tone phone. (See Appendix 1 for further information on skills-based routing).

Companies with more than one call centre, each providing the same services can combine the centres to service callers from one queue. This is called Network Automatic

Connection (NAC). For example, if a call was placed with the intention of reaching the local call centre but all agents are busy, NAC provides the capability of redirecting this caller to another, less busy call centre usually in a different geographic location.

2.2 Terminology

A caller will receive a busy signal if all trunks are in use. Such calls are referred to as blocked calls. Callers who receive a busy signal and do not attempt to call again at a later time are referred to as lost demand.

If all agents are busy, the caller is asked to wait for the next available agent by a recorded message. An available agent is one that is logged-in or connected to the ACD. The length of the queue is only as long as the number of trunks. Some callers, after receiving a trunk, may abandon the call prior to service. The rate of abandonment is one measure of dissatisfaction since it tends to increase with longer wait times.

The time from when the caller is connected to an agent to when the caller hangs up is referred to as talk-time. Talk time includes any time that the caller may have been placed on hold. Once the caller hangs up, the trunk line is released for other incoming calls. However, the agent may not be available at the same time. If the tasks resulting from the call are not completed during talk time, the agent will place his or her telephone's log-in status to "not-ready". Then the agent, though connected to the ACD, will not receive calls until he or she returns the log-in status to "ready". The time required to complete after-call work is referred to as wrap-up time. Call length includes wrap-up time.

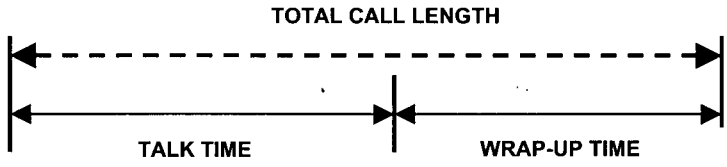


***Figure 2**: Total Service Time*

III. CALL CENTRE MANAGEMENT

An obvious goal of a service call centre is to improve customer service by being able to meet target service levels consistently. Another principal goal is to allocate staff efficiently. The majority of costs in operating a call centre are related to staffing [6]. Overstaffing results in spending money on unnecessary agents as well as lower staff-productivity.

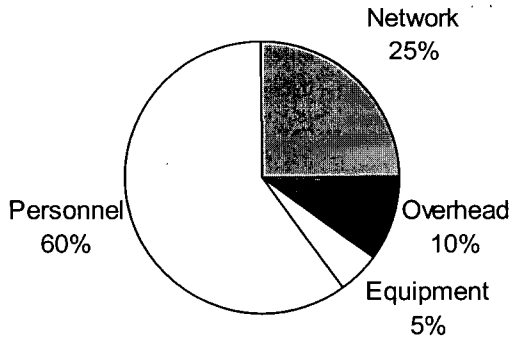


***Figure 3**: Typical Breakdown of a Call Centre's Operating Costs*

Understaffing impact service levels negatively. In order to avoid understaffing, call centres must determine staffing requirements. Two steps are involved. First target service levels must be established. Next, demand or call volume must be accurately forecasted to compute required staffing levels.

3.1 Establishing Service Levels

There are two widely used alternatives for measuring service level. One is Average Speed of Answer (ASA), which is simply the average delay of all calls. Delay is the length of time from when a call is received by the ACD to when it is connected to a server. Since not all calls are sent to the queue, this measure can be misleading. For example, suppose that the ASA for all calls received between 10AM and 11AM is only 10 seconds. This seems to be a reasonable length of time to be put on hold. However, if only 8% of calls actually had to wait for an agent then the average length of time that a caller who was sent to the queue waited, is 125 seconds.

The other method, which is very typical in call centres is to measure service level as a percent of calls, referred to as the Telephone Service Factor (TSF), answered within a certain number of seconds, referred to as the Target Answer Time (TAT). Determining the target level of service to be provided by the call centre is up to each company to define according to its service objectives.

3.2 Forecasting Call Volume

ACD technology collects call volumes among other statistics typically for each hour or half-hour period of the workday. A forecasting model can be developed using the historical call volume data provided by the ACD. Call centres commonly have period, daily, monthly and seasonal patterns that can be modelled to reasonably predict call volume by period for each day. Accurate call volume is necessary for computing required staffing levels.

3.3 Computing Staffing Requirements

Staffing requirements are computed using one of several techniques developed by Erlang. The Erlang-C formula is the standard method used for computing staffing requirements to service inbound calls in the ACD environment found at the Contact Centre [7]. The Erlang-C formula gives the probability that an arrival is delayed.

The Erlang-C formula is based on the Markovian delay system ($M/M/c/\infty$) which models the queuing process related to server requirements. The ($M/M/c/\infty$) system represents a queuing system with a Poisson arrival process, exponentially distributed service times, c servers, and an unlimited number of waiting positions. (See Gross and Harris [8] for derivation]).

An infinite queue assumption means that there will be a trunk available for every caller. In other words, the number of callers that receive a busy signal must be low in order for the Erlang-C formula to be a good approximation to the PH&N system. The Erlang-C model also assumes that the caller will wait in the queue for as long as necessary to be assisted. For this assumption to hold, the abandonment rate must be low.

The number of trunk lines utilized by the PBX can also affect customer service. A caller being serviced or waiting in the queue takes up a trunk line. An inbound call will be blocked if all trunks are in use; the inbound caller will receive a busy signal. Blockage rate can impact customer service levels.

Erlang developed another formula to calculate the number of required trunks (see Gross and Harris [8] for derivation), which is called the Erlang-B formula. This formula assumes an ($M/G/c/c$) queuing system and gives the blockage probability. Trunk service time is the amount of time a caller waits to be serviced plus the talk time. The agent's wrap-up time is excluded because the trunk line is released when the caller hangs up.

The Erlang-B formula implicitly assumes that a caller will not redial immediately after being blocked; otherwise the Poisson assumption fails. Although it is difficult to determine the percent of redialers, it goes up as blockage and waiting times increase [7].

3.4 Staffing Tradeoffs

Clearly customer service will improve by adding staff. However, adding more and more staff results in diminishing returns with respect to improved service levels. The opposite is true when subtracting staff; service levels decrease exponentially. The following example illustrates the tradeoffs in adding and subtracting agents.

Twenty calls may be forecasted for the peak period of 2pm to 3pm. The average call length, including wrap-up time, is 5.5 minutes. These values are typical of a peak period at PH&N's call centre. The workload hours or Erlangs, which is the unit of measure used by the Erlang-C formula, would then be: $\frac{\text{20 calls x 5.5 minutes}}{\text{60 minutes}} = 1.8 \text{ hours}$

The effects on ASA and agent occupancy of adding and subtracting agents based on 1.8 Erlangs or workload hours are detailed in the following table. The results below were taken partly from tables provided by TCS Management Group [5] and from their Call Centre Calculator [9].

Erlangs	No. of Agents	Staff:Workload Ratio	Average Delay of Delayed Calls	Probability that a Call is Delayed			ASA
1.8 hours	5	2.8	103 seconds	X	.0405	=	4 seconds
1.8 hours	4	2.2	150 seconds	X	.1285	=	19 seconds
1.8 hours	3	1.7	275 seconds	X	.3547	=	98 seconds
1.8 hours	2	1.1	1650 seconds	X	.8526	=	1407 seconds

***Table 1**: Staffing – Service Tradeoffs*

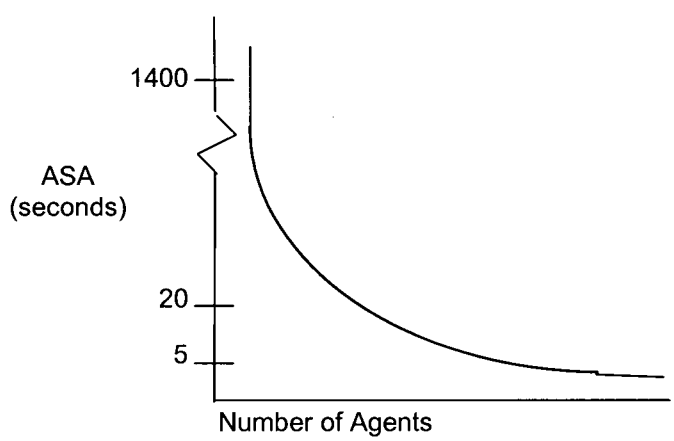


***Figure 4**: Staffing – Service Tradeoffs*

The table shows that as the staff to workload ratio approaches one, the ASA increases dramatically to 1407 seconds or 23 minutes. The blockage probability is also high at over 85%. Adding just one more agent would reduce the ASA to less than 2 minutes.

IV. CONTACT CENTRE AT PH&N

This chapter describes the operations then the service level performance of PH&N's call centre.

4.1 Current Operations

PH&N refers to its call centre as the Contact Centre. The Contact Centre services mutual fund clients and currently has seven agents. Each agent is referred to as Client Service Representative (CSR).

The Contact Centre operates weekdays from 7am to 5pm. Operating hours are covered by CSRs working one of three 7 hour shifts. CSRs have one hour for lunch. When it is taken depends on the shift. Each shift has more than one possible lunch start time. Every CSR is restricted to work one of the three shifts and a corresponding lunch option according to the department's policy. The shifts and each shift's set of allowable lunches are detailed below.

SHIFTS	LUNCH TIMES				
7:00am to 3:00pm	11:00 to Noon	11:30 to 12:30			
8:00am to 4:00pm		11:30 to 12:30	Noon to 1:00	12:30 to 1:30	
9:00am to 5:00pm				12:30 to 1:30	1:00 to 2:00

***Table 2**: Shift and Lunch Combinations*

All calls from mutual fund clients are answered by the ACD. The call is then transferred or routed to one of two queues using PBX technology and a voice recorded message. Callers seeking advice on PH&N's mutual funds such as structure, holdings, performance, and minimum investment are distributed to a Fund Advisor (FA) or to the queue serviced by only Fund Advisors if all agents in the group are busy. Fund Advisors form a revenue generation call centre rather than a service call centre like the Contact Centre. CSRs are responsible for handling all other enquiries from retail mutual fund clients. If all CSRs are busy, the incoming call is sent to the queue to wait for the next available CSR.

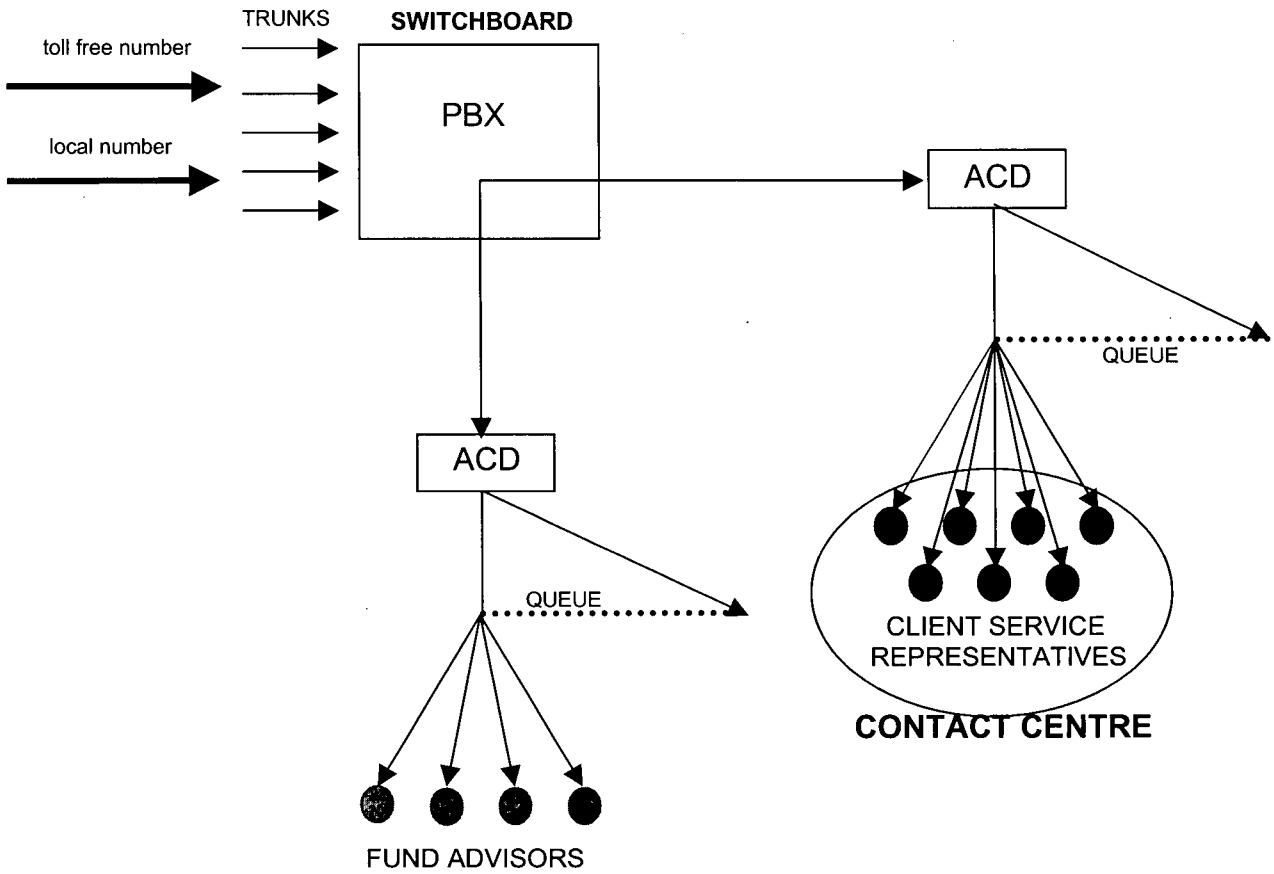


***Figure 5**: Contact Centre Architecture*

The ACD will only distribute calls to CSRs whose telephone is logged-in to the system and is placed on "ready" status.

A CSR's responsibilities are summarized below:

- Answering incoming calls are generally administrative in nature. Typical requests are:
  - statement transaction clarifications,
  - account status enquiries,
  - clarification on promotional material that was mailed with the monthly statement,
  - change of address requests, and
  - requests for product information.

- Responding to similar requests that were received through email or fax. These types of requests also usually require an outbound call to the client to confirm that the issue was resolved.

- Handling retail mutual fund clients that come to the office with a request for the Contact Centre. These clients do not call ahead of time and are referred to as walk-

ins. A CSR who is not currently logged-in will volunteer to perform this job. This duty is only assigned to one particular CSR during RRSP season when the number of walk-in retail mutual fund clients increases significantly. A CSR will be scheduled to work at reception solely to handle walk-ins for entire workdays at a time.

- Transaction orders are received by mail, fax or email and are sent to the Client Processing Department. However, CSRs are responsible for placing outbound calls to clients who have sent unclear transaction instructions.

- CSRs are also assigned to departmental projects both individually and in groups.

As the summary of duties above describes, the Contact Centre cannot neatly be described as either an inbound or outbound call centre. The Contact Centre is more a combination of both. The following figure depicts how CSRs spend their workday based on ACD output. A portion of the outbound calls is placed while CSRs are logged-in during "not ready" time. Table 2 below is a summary of ACD output for the week November 1, 1999 to November 5, 1999. See Appendix 2 for a sample of actual agent ACD output on agent activity.

DATE	TOTAL # OF INBOUND CALLS	TOTAL # OF OUTBOUND CALLS PLACED	AVERAGE TOTAL LOGGED-IN TIME PER AGENT	TIME SPENT ON OTHER TASKS
Nov 1, 1999 Mon	169	119	4 hrs 44 min	2 hrs 16 min
Nov 2, 1999 Tue	166	123	4 hrs 42 min	2 hrs 18 min
Nov 3, 1999 Wed	120	164	5 hrs 16 min	1 hrs 44 min
Nov 4, 1999 Thu	138	117	5 hrs 29 min	1 hrs 31 min
Nov 5, 1999 Fri	110	84	4 hrs 12 min	2 hrs 48 min

***Table 3**: Summary of ACD Output on Agent Activity*

Logged-in time includes time that telephones are on "not ready" status, which is the wrap-up time. As the last column in Table 2 shows, a lot of time is spent on work that does not result from incoming calls. The amount of time that each CSR spends logged-in to the ACD also varies each day.

There is a Client Services Manager who is responsible for the Client Processing Department, which contains the Contact Centre. However, there is no immediate supervisor for the Contact Centre. CSRs have a lot of autonomy in how they balance their workload. When the volume of incoming calls is high, CSRs currently doing work that is not related to incoming calls will log-in to the ACD. On days when certain CSRs have a lot of other work to do, the remaining CSRs will handle the bulk of the inbound calls.

4.2 Service Level Performance

The Contact Centre's target is to answer 80% of incoming calls within 20 seconds, which in call centre terminology is TSF 80, TAT 20. The ACD report on queue activity will let the Contact Centre know which period target service levels were reached. These reports show that target service levels are not being reached consistently.

Below is a segment of the ACD report on queue activity for October 15, 1999. See Appendix 2 for a sample of actual ACD output on the Contact Centre's queue activity.

Period Ending	Agents Available	Required Agents	Calls Accepted	TSF %
7:30	1	1	5	100
8:00	2	2	7	100
8:30	3	3	6	100
9:00	2	3	17	***52***
9:30	4	4	15	***60***
10:00	4	4	8	***62***
10:30	4	4	10	88
11:00	4	4	12	***69***
11:30	3	3	16	81
12:00	4	3	17	***41***
12:30	4	3	10	100
13:00	3	2	8	87
13:30	3	3	8	87
14:00	2	2	7	100
14:30	2	3	5	***60***
15:00	3	2	6	***66***
15:30	3	2	2	100
16:00	3	3	10	100
16:30	3	3	2	100
17:00	2	2	1	100
Total			172	76

***Table 4**: ACD Output on Queue Activity*

As the last column in the table shows, at least 80% of incoming calls were not answered within 20 seconds in seven of the twenty periods. At the end of every period the ACD uses the Erlang-C formula to compute the number of agents the call centre should have had, based on the call volume and average call length. These values are report in the third column, "Required Agents". This column suggests that the Contact Centre was understaffed for two periods and overstaffed for 5 periods.

To display the inconsistency in service level provided, TSF for TAT of 20 seconds is shown below for the first week of November for each half-hour of the workday. . The service level varies considerably throughout the day. However, service levels are more likely to be under target during the daily call volume peaks around 9-11 and 1-3.

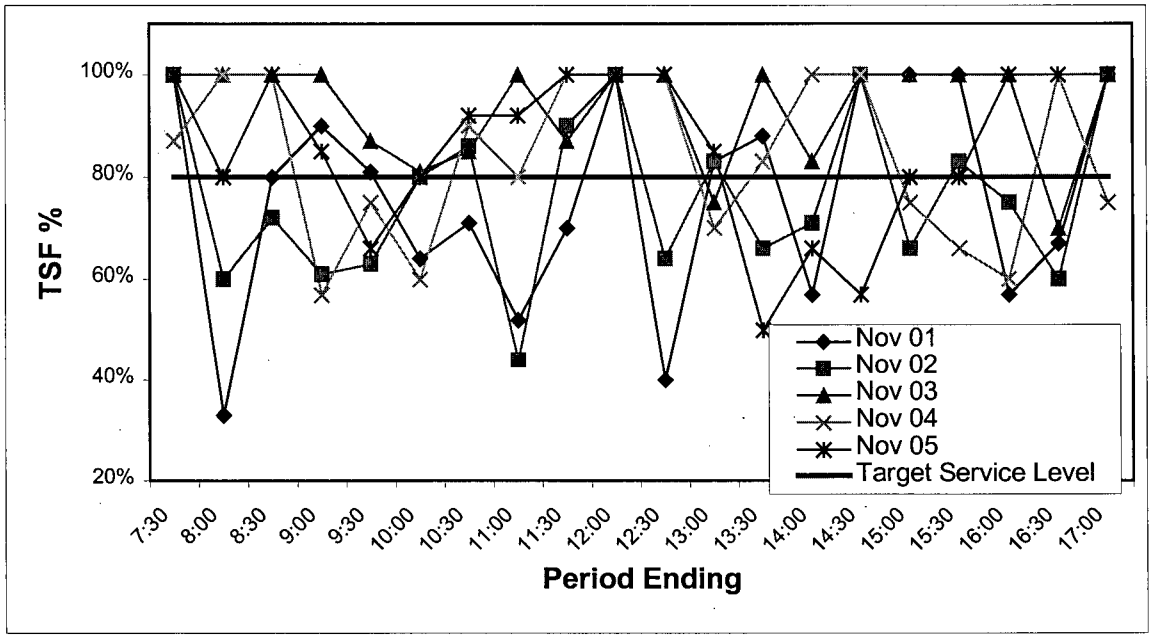


***Figure 6**: Service Levels Achieved During Week of November 1, 1999*

V. ANALYSIS OF SERVICE LEVELS

Studying the Contact Centre pointed to a combination of four possible reasons for poor service level performance as reported by the ACD at the end of each day.

Many times for periods where a low service level was provided, the ACD will report that the number of available agents was equal to or greater than the required number of agents, based on the Erlang-C formula using post hoc data for the period of concern. Contact Centre management wanted to know why this was happening so frequently.

There are two reasons for this. One is that the number of available agents provided by the ACD report is actually an average of the number of agents that had been logged-in to the ACD for that period. Thus, there may have been fewer or more agents logged-in over the duration of the period. CSRs do not schedule themselves to log-in for half-hour blocks. For a period where the ACD reports four agents being available, there could have been just one agent logged-in for a 5-minute interval. Due to the random nature of telephone call arrivals, several calls could have been accepted by the ACD during this interval. These few calls can severely impact the overall TSF for the period.

The other reason is that the number of available agents reported by the ACD can be exaggerated. Analysis of the ACD agent report revealed that in reality, the agents available column does not accurately represent the true number of agents available on average. The ACD includes all phones that had been logged-in even if the phone had not taken a single call. There are several phones outside the Contact Centre that are capable of logging in to the ACD. Often, the log-in button is accidentally pressed on one of these phones. It may also go unnoticed for most of the day because the log-in status is "not ready" by default and thus, the telephone will not receive any inbound calls destined for the Contact Centre. It is not uncommon for this to happen several times a week. An ACD agent report for December 1, 1999 is shown below.

Agent Name	Agent ID (Phone)	Calls Answered	Talk Time Total	Hold Time Total	Wrap-up Time Total	NonACD Time	Logged-in Time
	9251	***0***	***0:00***	***0:00***	***10:54:04***	***0:00***	***10:54:04***
Agent 1	1021	34	1:10:56	2:44	2:59:48	39:28	6:17:16
Agent 2	1002	11	45:48	1:34	1:28:40	40:48	3:04:12
Agent 3	1014	46	2:59:58	9:58	1:48:40	1:20:28	6:46:44
Agent 4	1032	46	1:44:30	30:42	3:18:04	42:04	7:10:46
Agent 5	1018	54	1:24:46	8:08	2:58:36	1:13:00	7:11:46
Agent 6	1030	47	2:11:04	16:16	1:32:30	1:19:28	6:04:52
Agent 7	1031	35	1:33:10	7:16	1:55:04	1:05:58	6:43:08
		273	11:50:12	1:16:38	26:55:26	7:01:14	54:12:48

***Table 5**: ACD Agent Report for December 1, 1999*

The report indicates that telephone 9521, which does not belong to the Contact Centre, had been logged-in the entire day and had not taken a single call. Each of the values in the "Agents Available" column of the ACD queue report should be subtracted by one.

The corresponding ACD queue report for December 1, 1999 is shown below. Subtracting one from the agents available column shows that in fact, for many of the periods with low TSFs, the Contact Centre was understaffed.

Period Ending	Agents Available	***Actual*** Agents Available	Required Agents	TSF
7:30	4	3	3	100
8:00	***4***	***3***	***4***	***62***
8:30	4	3	5	100
9:00	6	5	4	60
9:30	7	6	6	78
10:00	8	7	7	37
10:30	***7***	***6***	***7***	***33***
11:00	7	6	6	69
11:30	***5***	***4***	***5***	***38***
12:00	***4***	***3***	***4***	***58***
12:30	5	4	4	66
13:00	***4***	***3***	***4***	***81***
13:30	***4***	***3***	***4***	***37***
14:00	5	4	4	50
14:30	***4***	***3***	***4***	***69***
15:00	***5***	***4***	***5***	***21***
15:30	***3***	***2***	***3***	***30***
16:00	3	2	4	35
16:30	***3***	***2***	***3***	***20***
17:00	2	1	3	36

***Table 6**: ACD Queue Report for December 1, 1999*

Because there is no immediate supervisor, each agent is responsible for ensuring that the Contact Centre runs smoothly. As a result, a CSR's decision to log-in is typically made too late and the decision to log-off is made too early. These are the other two causes for the Contact Centre's poor service level performance.

Noticing that the volume of incoming calls is high tends to be subjective. A CSR logged-in to the ACD may notice that the call volume is high because an inbound call continues to arrive immediately after placing the telephone on the "ready" status or another CSR may let everyone know. CSRs that are already logged-in may decide to stay logged-in for a little longer even though he or she may have originally intended to log-off to do other tasks unrelated to the call centre. Other CSRs currently doing other work may log-in to help. Due to the random nature of call arrivals, the number of required staff hours has to be greater than the actual workload hours. In order to provide a service level of TSF 80%, agent occupancy must be especially low. In other words, agents will appear idle much of the time. Deciding to log-in when the agents currently logged-in appear busy is too late.

There is a system in place that will display the number of callers waiting to be assisted. This number may lead the CSR to log-off too early. A CSR, who has other tasks unrelated to the call centre, may decide to log off as soon as he or she notices that the queue length was zero. However, a zero queue length does not mean the call centre is less busy. A queue can be zero several times during a peak period, again due to the random nature of telephone call arrivals.

The autonomy that CSRs have in deciding when to log-in or off encourages a team culture; however, this autonomy is an inefficient scheduling policy when trying to reach target service levels. The Contact Centre's goal is to consistently provide a TSF of 80% within TAT 20. However, the Contact Centre lacked the controls in place to achieve the service level systematically.

VI. METHODOLOGY & APPROACH

The purpose of this project was to develop and provide PH&N with a call centre management application that would improve call centre efficiency and ultimately improve customer service.

6.1 Commercially Available Applications

Scheduling tools for call centres are widely available commercially. Some of the leading companies providing call centre workforce management software are TCS Management Group Inc. [9], Taske Technology Inc. [10], Portage Communications Inc.[11], and Pipkins Inc. [12].

All commercially available call centre workforce management software includes modules capable of [13]:

- Forecasting call volume.
  The forecasting model uses data provided by the ACD and is built on period by period, day of week, monthly call volume variables as well as possible seasonal and yearly trend patterns.

- Computing staffing requirements.
  For a revenue-generating call centre, staffing levels can be determined based on profit maximization [14]. The Erlang-C formula is used for a service call centres. Currently, simulation is used to approximate staffing requirements for call centres which use skills-based routing [15], [16] (see Appendix 1 for further details).

- Generating efficient staff schedules.
  Schedules are generated based on staffing requirements as well as schedule constraints and staff preferences. This module automates the process of selecting an efficient staffing schedule.

The schedule generation module is usually optional. It is much more difficult for vendors to customize this function which makes it an expensive option. Many vendors suggest that the need to automate scheduling is a function of the number of agents. However, the need to automate can also be a function of the complexity of scheduling as is the case at PH&N's Contact Centre.

Vendors developing software for service call centres assume the centres to be only inbound in nature, which they typically are. As a result a limited number of fixed-length shifts devoted to handling inbound calls is also assumed. However, the Contact Centre at PH&N is an inbound and outbound call centre and more. Each CSR is logged-in for only a portion of the day. The number of hours each CSR logs-in varies every day. Also, the number of hours can be distributed flexibly throughout the day. Once a scheduling policy is developed, automating the scheduling process can eliminate a time consuming tedious task.

Purchasing commercially available software requires a fairly substantial investment. Microsoft EXCEL, which is familiar and readily available to all CSRs, was used to develop a workforce management tool customized to the needs of the Contact Centre.

6.2 Project Scope

The development of the tool involved the following phases:

- analysis of the existing forecasting model,
- validation of Erlang-C assumptions,
- construction of an efficient scheduling policy,
- development of a call centre workforce management software system, and
- implementation.

6.2.1 Analysis of the Existing Forecasting Model

A forecasting model was previously developed for PH&N's Contact Centre [3]. However, it is not being used mainly because it lacks a user interface. The model was developed as part of another study without the original intention of becoming a tool for end-users.

The model uses Poisson regression and is based on one year (from December 15, 1997 to December 9, 1998) of call volume data collected by the ACD for every half-hour interval of each operating day. The model includes the variables common to most call centre forecasting models, period, day of week, and month. There is also a time trend variable to capture the steady increase of call volumes per day during the RRSP term starting in mid-January and ending with the end of the RRSP season.

The effectiveness of this forecasting model's ability to predict *future* call volume patterns was analyzed with 1999 data.

6.2.2 Validation of Erlang-C Assumptions

In order to insure accuracy of the Erlang-C formula, we must be sure that callers will wait as long as necessary to be assisted and that few calls will receive a busy signal must hold. The Erlang-C approximation is sensitive to these two assumptions unlike others such as the assumption of exponentially distributed service times [7].

The abandonment and blockage rate must be low. The number of blocked calls is calculated from ACD queue reports as the difference between calls accepted and calls answered.

Both the abandonment and blockage are quite low at the Contact Centre. The rates were only calculated for the month of October. However Jon Merrifield, who is contracted by PH&N and responsible for ACD output, verified that these rates have been historically low. The abandonment rate for the month of October was less than 2%. The blockage rate for the same period was also less than 2 percent.

Total Calls	Number of Calls Abandoned	Abandonment Rate
2502	47	1.9%

***Table 7**: Abandoned Calls in October 1999*

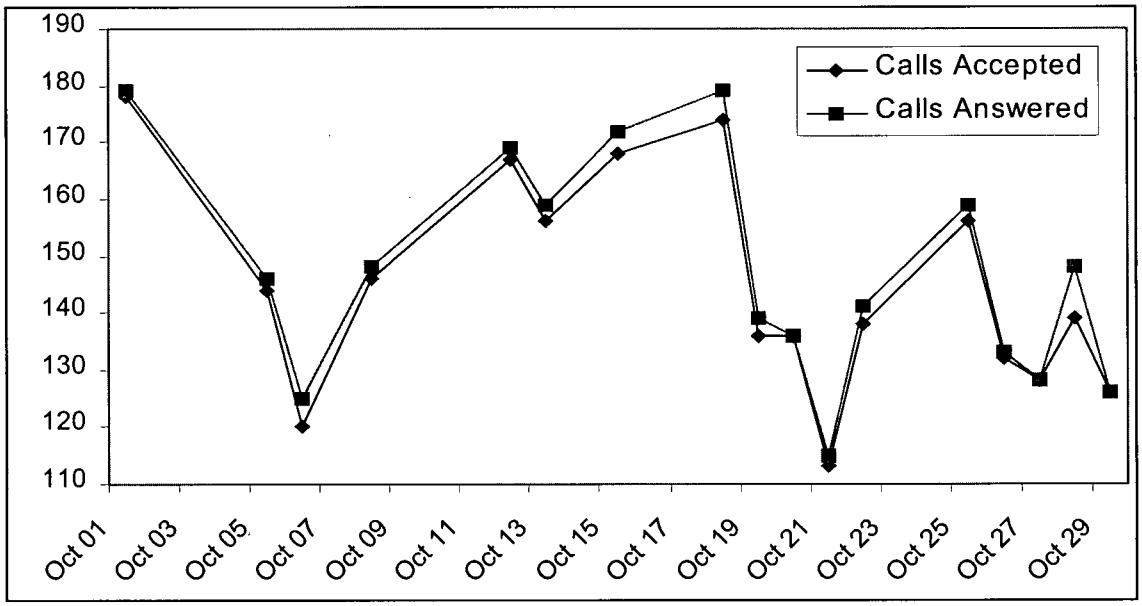


***Figure 7**: Blocked Calls in October 1999*

6.2.3 Construction of an Efficient Scheduling Policy

An efficient scheduling policy was defined to be one that meets computed staffing requirements as well as a list of preferences and constraints detailed in a meeting with the Client Services Manager and several CSRs

Schedule Constraints

The following is a list of constraints and rules that a scheduling policy must satisfy when generating a staff schedule for a future day:

- CSRs must be assigned to work one of the three shifts (7am to 3pm, 8am to 4pm, and 9am to 5pm) unless the CSR has a shift restriction,

- CSRs must be assigned to one of the lunch options available for the shift he or she is scheduled to work on, unless the CSR has a lunch restriction,

A CSR who has a shift or lunch restriction has a separate shift or lunch that he or she must be assigned to. For example, a CSR may have a doctor's appointment at 1pm on the day being scheduled and will only be in the office from 7am to noon with no break for lunch.

Typically, each CSR will work 7 hours a day with a 1-hour lunch break. Since the ACD collects data for each half-hour of the operating day, it was decided that the schedule would also schedule CSRs on a half-hour basis rather than on an hourly basis. Thus each CSR has 14 periods a day of work but not all of it will be spent logged-in to the ACD; some time must be allocated for other work.

Each CSR must be able to indicate how many periods of the assigned shift that he or she will be available to log-in to the ACD as long as the total number of periods that all CSRs are available is greater than or equal to the total number of periods computed to handle incoming calls. This was determined by the CSRs and the Client Services Manager as preferable over dividing the total number of periods required equitably across all CSRs. More experienced CSRs tend to have more other work to do, and as a result, less time available to log-in to the ACD to handle inbound calls.

Schedule Preferences

Schedule preferences as determined by the Contact Centre Manager should aim to minimize the number of shifts required to meet staffing requirements. This preference is more desirable than maximizing total agents' shift and lunch preferences.

The schedule must also select shifts to best fit each CSRs preferences. With lunch breaks, it was determined that assigning lunch breaks to minimize the total number of people away each period was more important than assigning lunch breaks according to CSR preferences.

Once a shift and lunch break has been assigned to each CSR, each period will be scheduled either for other work or to log-in to the ACD. It was decided that either task should be scheduled for at least two consecutive periods at a time.

A desirable schedule is one that satisfies all the constraints and maximizes the preferences while meeting staffing requirements.

Literature Review

The staff scheduling problem can be divided into two sub problems. The first problem is to determine the number of staff required to meet minimum service levels. Computing agent requirements is a problem well covered by queuing theory. The

Erlang-C delay system is the model used for computing agent requirements in service call centers, see Harris, Hoffman, and Saunders [7] for further reference.

The second part of the staff scheduling problem is the task of constructing efficient daily work schedules that satisfy agent requirements. There is a significant amount of literature covering this problem. Integer programming models are commonly applied to scheduling problems. The three typical scheduling problems are covered in an overview by Morris and Showalter [17]. More specifically, the shift problem of assigning agents to a select number of daily shift start times is found in Bechtold and Jacobs [18] with a method for handling breaks. Baker [19] covers the days-off problem of assigning work and nonwork days across the week. Lastly Bechtold, Brusco, and Showalter [20] cover the tour scheduling problem, which is the integration of both the shift and days-off scheduling problems. An article reference for the scheduling of telephone sales operators in particular is Gaballa and Peace [21].

Most of the literature covers situations where each individual works an equal amount of time whether daily or weekly. At PH&N's call centre, each CSR can have more than one shift for logging-in to the ACD each day of variable length in order to meet the required staffing level. Staff scheduling at PH&N's call centre could also be linked to the days-off scheduling problem where the issue of assigning work and nonwork days can be changed to assigning work and nonwork half-hour periods. However literature for the days-off problem focuses on which two days to assign off work out of the week rather than the issue of deciding whether to assign the agent on or off the phones each period for each agent.

Rafael Sanegre [4], a past student of the COE, developed an integer program to schedule the call centre at the Workers' Compensation Board (WCB) which is very similar to the Contact Centre. Agents at the WCB call centre also spent only a portion of each day logged-in to the ACD. However, unlike at the Contact Centre, every agent was required to log-in to the ACD for exactly 3.5 hours scheduled in one continuous block. Because this log-in time was not variable, all combinations of shifts, lunches and log-in time, also called tricks in call centre scheduling terminology, were enumerated. An assignment problem was used to maximize the agents' preferences for the possible tricks.

Because of the limitations in the technology that was used to develop the Contact Centre's workforce management tool, the cost of an IP solver as well as the many criteria involved in defining an optimal schedule, a heuristic rather than an integer program was developed to satisfy schedule constraints and preferences.

6.2.4 Development of a Call Centre Workforce Management System

The tool was developed to be capable of providing all the functions found in commercially available applications. A module that allows users to track past performance of the forecasts with actual data was also included.

Since the forecasting model is built on historical data, the tool allows users to modify forecasts to be used for generating schedules. For example, a new product promotion mailed to clients along with statements will increase call volume temporarily in the days following the mailing.

In addition, the tool allows users to modify schedules after they have been generated. This provides flexibility to involve preferences that cannot be modelled easily. For example, an agent may be indifferent to taking lunch at 12:30 or 1 if he or she has been assigned to the 9 to 5 shift. If neither lunch will affect the desirability of the overall schedule, the tool may select 1pm. However, the agent may prefer to take lunch at 12:30 since a co-worker whom he or she frequently takes lunch with was assigned to take lunch then.

6.2.5 Implementation

The success of the project lies in the implementation. Ultimately, the end-users must be able to use the tool as well as to understand its purpose and value. An interface was developed keeping in mind the skills and needs of all end-users or CSRs and the basics of human computer interaction [22]. Implementation also involved user training.

Steps were taken while the tool was being developed to maintain the focus on the implementation of the tool as a business solution. A test run of call volume predictions was provided to convince the Contact Centre of the forecasting model's validity near the onset of the project. As well, end-users provided input on sample schedules while developing the final scheduling heuristic.

VII. ANALYSIS & RESULTS

Results from the analysis of the forecasting model as well as the scheduling policy developed are provided below.

7.1 Forecasting Model

The forecasting model was developed using 1998 data. An accuracy analysis of the forecasting model was done as part of the previous project. Although the results were favourable, it only measured the model's "goodness of fit". The Contact Centre was concerned with the model's accuracy of future forecasts.

In June of 1999, Jon Merrifield who is contracted by PH&N, provided the call volume data from the ACD necessary to complete the analysis. The data was collected and processed in an Excel spreadsheet and covered the periods from Monday, January 5, 1999 to Friday, May 22, 1999.

The Mean Average Percentage Error (MAPE) of the predicted call volumes by period provided by the model for the same interval, Jan. 5 to May 22, was 67.1%. This figure seemed high. Further analysis showed that there was a systematic bias. The Mean Error (ME) was –4.6, indicating that the model was over-forecasting by approximately 5 calls per period. In 1999, actual call volume per period was consistently lower than in 1998, see Figures 7 to 9 below. The graphs compare the average values from Jan to May.

The improvement in statement accuracy due to a new system implemented at the end of 1998 is credited for the lower call volumes in 1999.

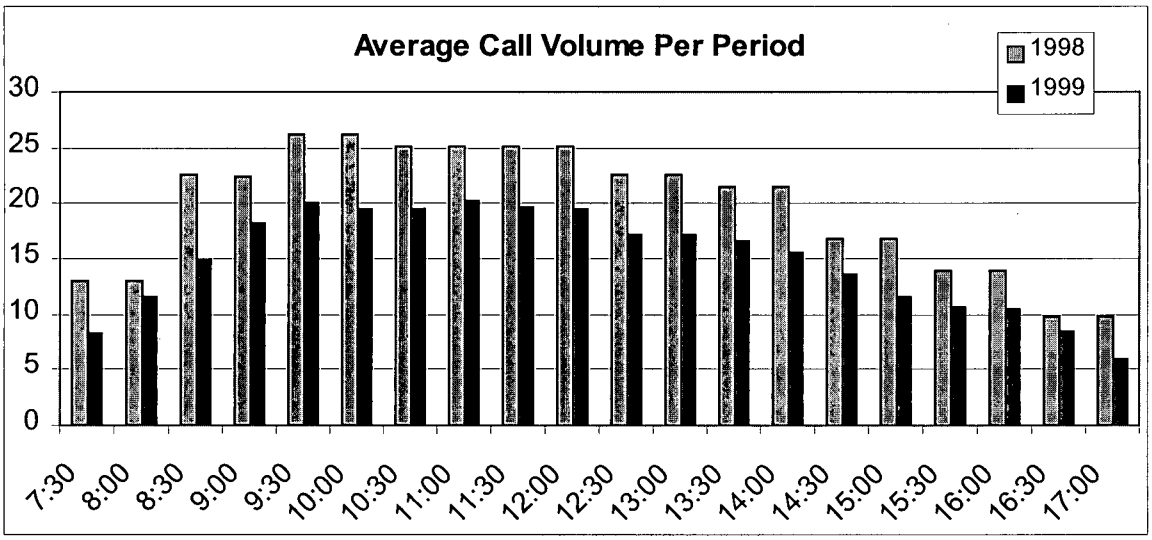
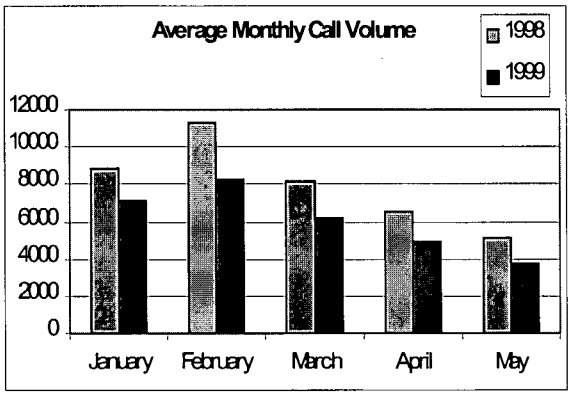
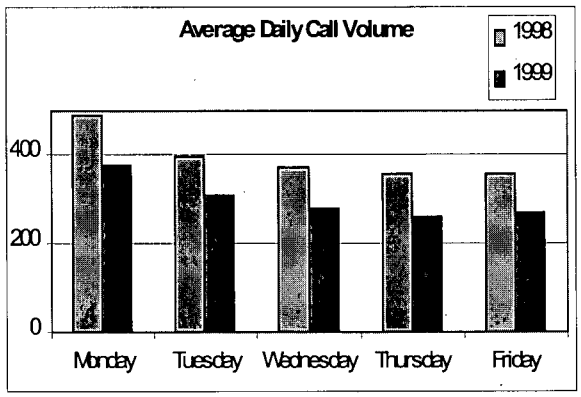


***Figure 8**: 1998 vs. 1999 Actual Average Call Volume per Period*





***Figures 9 & 10**: 1998 vs. 1999 Actual Average Monthly and Daily Call Volume*

Adjusting the predictions for this systematic error reduces the MAPE considerably. The adjusted predicted call volumes by period closely patterned actual 1999. Average call volumes for 1999 versus predicted is shown below. The pattern is consistently close when analyzed month by month.

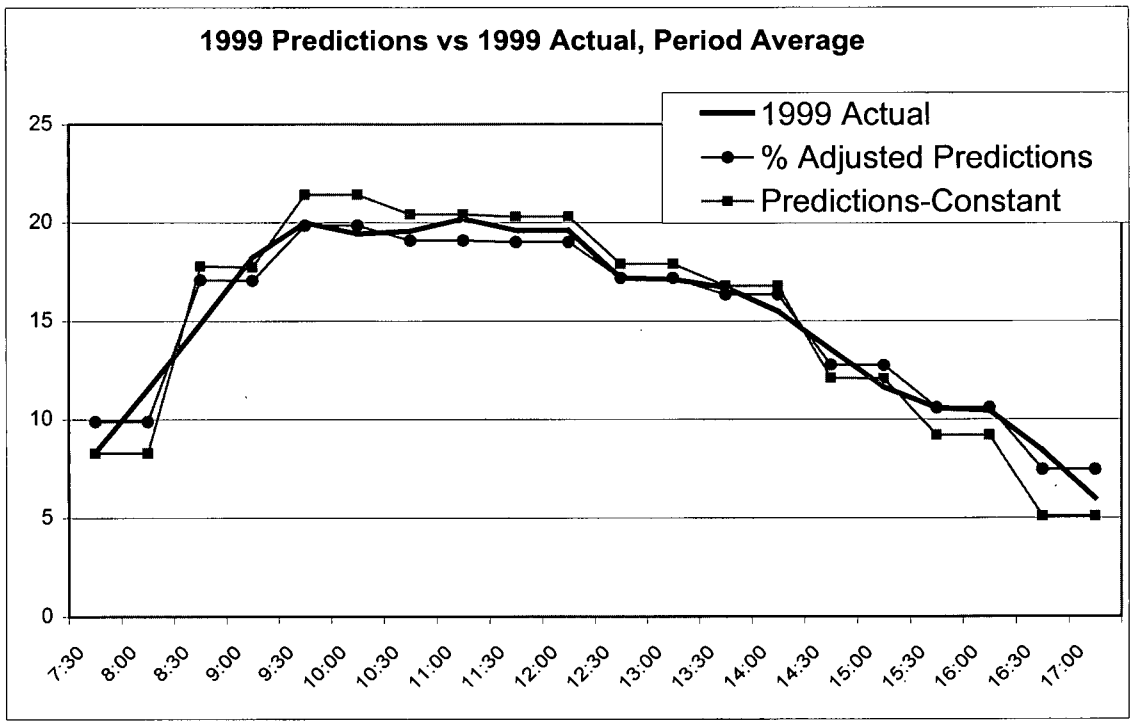


***Figure 11**: Adjusted Predictions*

The graph above shows predictions adjusted by subtracting a constant equal to the ME and by multiplying a percent equal to the total actual call volume over predicted call volume for Jan. to May 1999. This percent was 76.5%. The MAPE of the percent

adjusted predictions, 42.1%, was slightly higher than the MAPE of the predictions reduced by a constant, 38.8%. As Figure 10 shows, the percent-adjusted predictions followed the call volume pattern more consistently. Because this was the case and agent requirements would be relatively insensitive to the minimal difference in MAPE between the two types of adjustments, the forecasting model was modified to provide predictions multiplied by 76.5%.

The following graphs show that the predictions based in 1998 are consistent with actual 1999 monthly and daily patterns as well

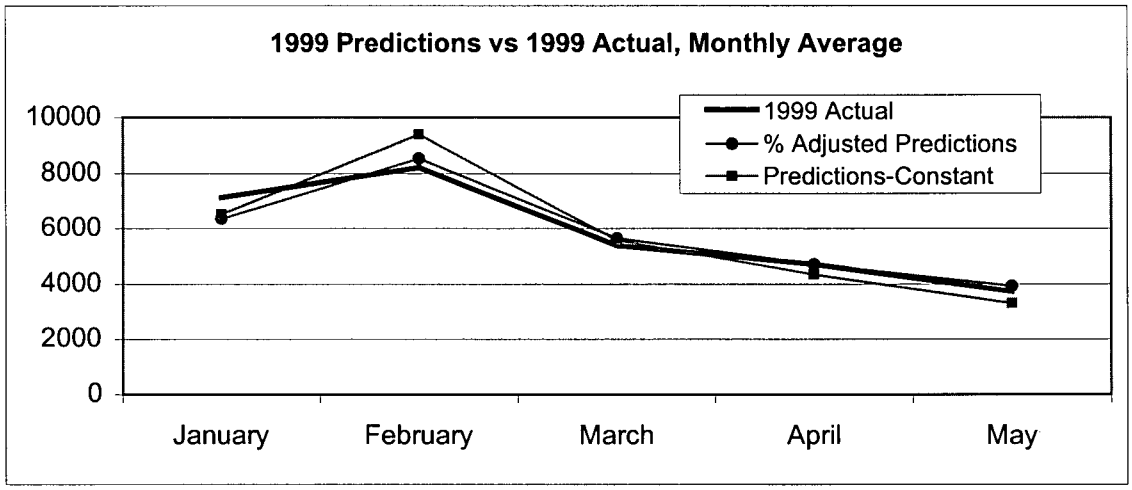


***Figure 12**: Average Monthly Predictions vs Actual Call Volume*

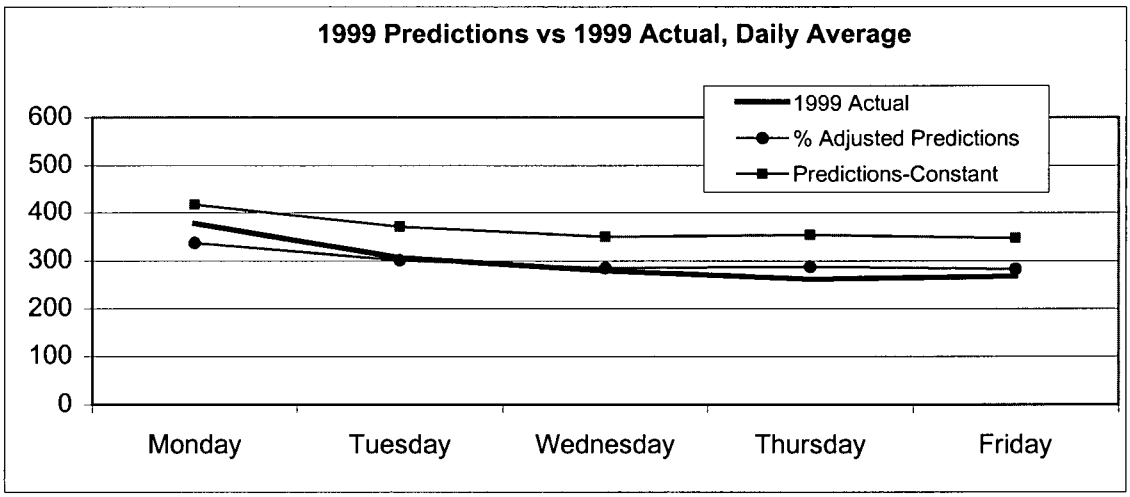


***Figure 13**: Average Daily Predictions vs Actual Call Volume*

Had there been more time, rerunning the regression would have been desirable. However the analysis thus far show that only levels have changed; patterns remain

consistent. Modifying the predictions would provide fairly accurate predictions. A report containing these results was presented to the Client Services Manager on July 27, 1999. In order to gain the confidence of the ultimate users of the forecasting model, a "test run" was initiated shortly after. On Friday, July 30, 1999 and Aug. 6, 1999 the CSRs were presented reports detailing the model's call volume predictions by period for each day of the following week. The reports included required staffing levels based on call volume forecasts. At the end of each week, similar reports comparing the forecasts with the resulting actual data were given. The following tables summarize the results for the two weeks.

WEEK 1 SUMMARY

		Total Calls		Total Agent Hours		Total Agent Hours
		Actual	Forecasted	Required	Forecasted	Logged-in to ACD
Tues	**3-Aug**	151	156	26	30	42
Wed	**4-Aug**	147	143	28	28.5	41.5
Thurs	**5-Aug**	118	138	26.5	26.5	43
Fri	**6-Aug**	99	134	23	26.5	33.5

WEEK 2 SUMMARY

		Total Calls		Total Agent Hours		Total Agent Hours
		Actual	Forecasted	Required	Forecasted	Logged-in to ACD
Mon	**9-Aug**	169	172	29.5	34	45
Tue	**10-Aug**	170	156	27	31	42.5
Wed	**11-Aug**	152	143	28.5	29	44
Thurs	**12-Aug**	129	135	28	29	46.5

***Table 8**: Results of Call Volume Predictions Presented to CSRs*

No report was provided for Aug. 2 since the office was closed due to a holiday. As well no follow-up report was provided for Aug 12 because actual values were not available to compare.

The third and fourth columns, actual and forecasted total calls, in Table 8 compare forecasted call volume with subsequent actual call volume. The adjusted forecasts proved convincing. Total agent hours forecasted were consistent with total agent hours required based on subsequent actual call volume

The sixth column, forecasted total agent hours, shows the agent hours required based on applying the Erlang-C formula to the forecasted call volume. The last column, total agent hours logged-in to ACD, shows the number of actual agent hours. The fifth column, required total agent hours, shows the number of hours that were actually required based on actual call volume. Because call volume forecasts were close to actual, the table shows that using the forecasts to compute staffing requirements can lead to a significant reduction in required agent hours.

The total call length used to compute predicted agent hours was 5.4 minutes, an average of July data. This average was used in the staffing level calculations for every period and day. In reality total call length will differ slightly period by period and day to day. The ACD does not provide data required to calculate average call length on a period by period basis. Staffing levels are generally insensitive to a range this small.

Average call length is shown for the last week in November where data was available (on a daily basis since period by period data is not available).

Date	Average Call Length
Wed, Nov 24	5 min 35 sec
Thurs, Nov 25	5 min 22 sec
Fri, Nov 26	5 min 16 sec
Mon, Nov 29	5 min 33 sec
Wed, Dec 1	5 min 14 sec

***Table 9**: Average Call Length*

The graph below charts staffing levels period by period for August 9. Not just on August 9 but for each of the days in the "test" period, the Contact Centre was overstaffed in handling inbound calls.

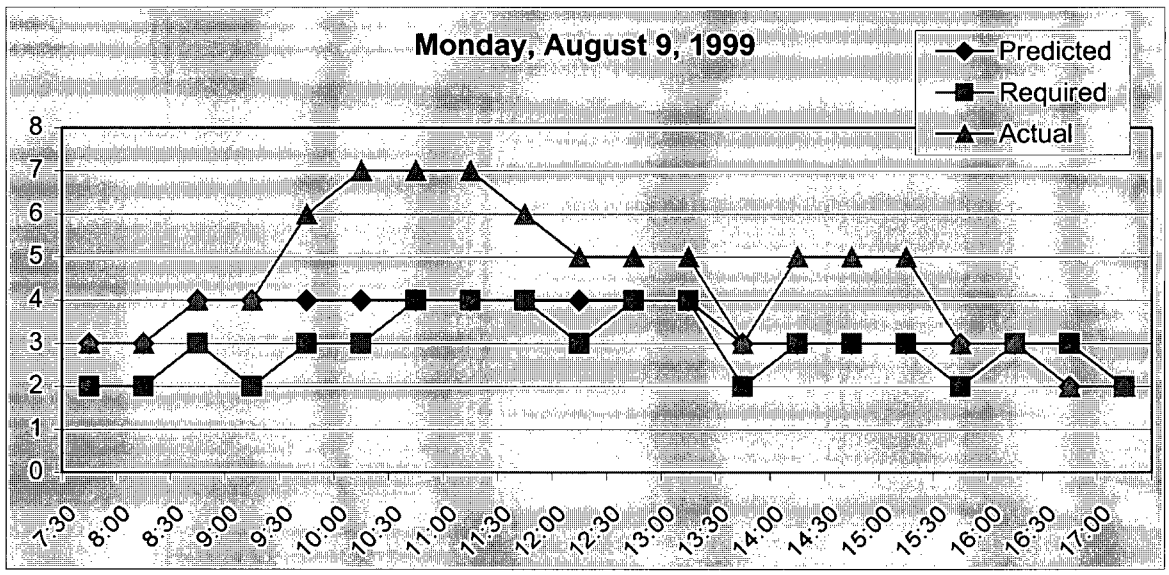


***Figure 14**: Staffing Levels for August 9, 1999*

Even though the Contact Centre was clearly overstaffed (see plot of Actual staffing level in figure above), seven of the twenty periods on August 9 did not meet target service levels. Overstaffing wastes time that CSRs could have used to do other tasks. Since

most of a CSR's duties outside of the call centre require placing outbound calls, much of it cannot be done while the CSR is logged-in. Placing outbound calls will inhibit inbound calls.
On August 9, 34 agent hours were forecasted and 45 agent hours were actually used (see Table 8). If the Contact Centre had used forecasts provided by the model to compute staffing required, then 45 minus 34 hours could have been devoted to other work which would have increased overall productivity. CSRs that are logged-in cannot do other work because almost all tasks unrelated to incoming calls require placing outgoing calls, which would inhibit the CSR from receiving a potential call.

7.2 Scheduling Heuristic

The number of periods each agent is available to log-in is variable up to a maximum of 14 periods. There are also flexible shift start times and lunch times. In addition, the scheduling policy includes many preferences of differing priorities such as agent preferences for shifts and lunch times as well as schedule preferences. Using heuristics supported the flexibility in considering several preferences of varying priority or equal priority.

Generally, the heuristic starts by assigning shifts. The minimum number agents required for each of the three shifts is predetermined by the required staffing levels in the first and last four periods and during the mid-day peak interval. CSRs are assigned to these shifts according to preferences.

Lunch times are assigned next. CSRs are each scheduled to his or her first preference for lunch initially. A iterative method is used to move agents to a less preferred lunch time if it will reduce the number of agents away for lunch each period overall.

Lastly, each period of the day must be assigned to either ACD work (log-in) or non-ACD work. The work assignment heuristic developed is a complex combination of priorities, rules, and procedures to meet schedule constraints as well as to best meet overall schedule preferences.

The heuristics are described in detail in Appendix 3. The programming code is used provided in Appendix 4. A sample schedule generated automatically using the tool is shown below. 1 means the CSR is scheduled to log-in, 0 means the CSR is scheduled to do other work, L means a lunch period, and X is outside of the CSR's shift.

December 6, 1999

MINIMUM TOTAL HALF HOUR BLOCKS REQUIRED: 52

	7:30	8:00	8:30	9:00	9:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30	16:00	16:30	17:00	AGENT'S HALF HOUR BLOCKS: AVAILABLE	USED	LEFT
Paul Hiorn	1	1	1	1	0	0	0	0	L	L	1	1	1	1	0	0	X	X	X	X	8	8	0
Luke Smith	1	1	1	1	0	0	0	0	L	L	1	1	1	0	0	0	X	X	X	X	7	7	0
Julie Lyons	X	X	1	1	1	1	1	0	0	1	1	L	L	1	1	0	0	0	X	X	9	9	0
Elizabeth Kim	X	X	0	0	1	1	1	1	0	0	L	L	0	0	0	0	0	0	X	X	4	4	0
Ana Maria Kim	X	X	0	0	1	1	1	1	1	L	L	1	1	1	1	0	0	0	X	X	9	9	0
Junior Bains	X	X	X	X	0	0	0	1	1	1	0	0	L	L	0	1	1	1	1	1	8	8	0
George Burns	X	X	X	X	0	0	0	0	1	1	0	0	L	L	0	1	1	1	1	1	7	7	0
REQUIRED:	2	2	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2	2	2	2	52	52	0
STILL NEEDED:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			

***Figure 15**: Generated Schedule, Example 1*

The final heuristic developed provided successful results. In the above schedule, all of the constraints as well as preferences were met.

Understaffing can occur and tends to occur in the periods ending at 1, 1:30 and 2. This is caused when the early shift agents are only available to log-in for a limited number of blocks and the blocks have already been scheduled to log-in before these periods. These agents are crucial in these periods since agents assigned later shifts will be away on lunch at this time.

Originally, schedules were generated to focus on lumping all blocks for logging-in to the ACD sequentially if at all possible. This usually led to the situation described above. The heuristic was adjusted to stop assigning morning shift agents to log-in after the period ending at 9 until they have come back from lunch unless needed to satisfy staffing requirements. It is unlikely that morning shift agents would be required since all agents are in after 9. This modification also forces splits in the remaining shifts as well but considerably reduces understaffing in the latter lunch periods.

The Client Services Manager was consulted on this issue. She indicated that although each agent had their own individual preferences, splits were actually desirable. The reason for this was that a portion of many tasks could not be completed until many periods after the task had been initiated. For example, the agent may need to wait for an answer to an inquiry that was placed with another department.

A schedule that did not meet the preference of assigning at least two blocks to one work type is shown below. Agent 4 is assigned to log-in for only one period at 2:30 without even having at least two periods logged-off the ACD. She would have preferred to log-in for three consecutive periods. However, this would push another agent into the same situation. The scheduler could manually assign this agent to log-in in the period ending 2:30. An agent currently assigned to log-in could log-off one period earlier. This will smooth out the schedule but will impact the agents' available blocks.

December 17, 1999

MINIMUM TOTAL HALF HOUR BLOCKS REQUIRED: 51

	7:30	8:00	8:30	9:00	9:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30	16:00	16:30	17:00	AGENT'S HALF HOUR BLOCKS: AVAILABLE	USED	LEFT
Mary Glavas	1	1	1	1	0	0	0	0	L	L	1	1	1	0	0	0	X	X	X	X	7	7	0
Luke Smith	1	1	1	1	0	0	0	0	L	L	1	1	1	0	0	0	X	X	X	X	7	7	0
Junior Bains	X	X	1	1	1	1	0	0	1	1	0	L	L	1	1	0	0	0	X	X	8	8	0
Elizabeth Kim	X	X	0	0	1	1	1	1	0	0	L	L	1	1	0	1	0	0	X	X	7	7	0
Ana Maria Kim	X	X	0	0	1	1	1	1	1	L	L	0	0	1	1	0	0	0	X	X	7	7	0
Julie Lyons	X	X	X	X	0	0	1	1	1	1	0	0	L	L	0	1	1	1	1	0	8	8	0
George Burns	X	X	X	X	0	0	0	0	0	1	1	1	L	L	0	0	1	1	1	1	7	7	0
REQUIRED:	2	2	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2	2	2	1	51	51	0
STILL NEEDED:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			

***Figure 16**: Generated Schedule, Example 2*

In the heuristics, a function generating random integers within a specified range is used when it is necessary to make fair selections. Thus the final schedule produced may vary somewhat each time the program is run. Each schedule will impact agents differently and the final schedule will be a partial function of the user. Ultimately all CSRs will share the responsibility of staff scheduling, most likely for one month at a time.

Because the forecasting model does not include a weekly variable, one week of schedules can be used for the entire month. This alleviated concerns that CSRs had about frequently changing shift assignments.

VIII. CONTACT CENTRE STAFF SCHEDULING TOOL

This section provides an overview of the tool that was implemented at the Contact Centre. The functionality and interface of the tool was developed with EXCEL 97 since this is the version currently operated by PH&N. Approximately seven thousand lines of code were built in develop the required functionality. It is capable of providing call volume forecasts on a period by period basis through an interface developed for the modified forecasting model, computing agent requirements using the Erlang-C delay system, generating schedules based on the policy developed, and finally allows users to track past performance of call volume and staffing requirement forecasts.

8.1 Main Menu

The initial screen is shown below. There are five options available to the user:

1. Generate future work schedules.
2. Track past call volume and staffing requirement forecasts.
3. Input and edit each agent's shift and lunch preferences.
4. Edit default values used throughout the tool.
5. Exit the program.

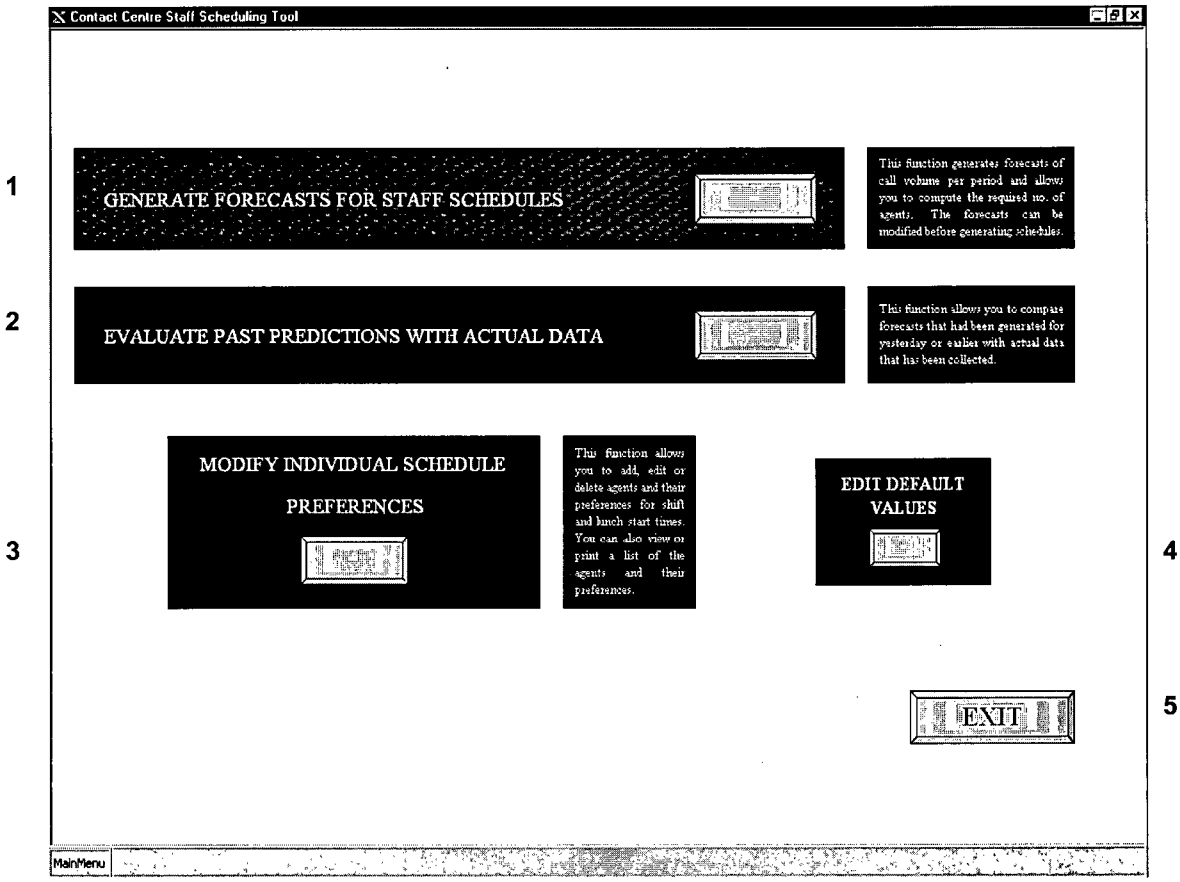


Figure 17: *Main Menu*

8.2 Preferences

All agents who will be a part of the schedule must input their names and preferences. This can be done with button 3.

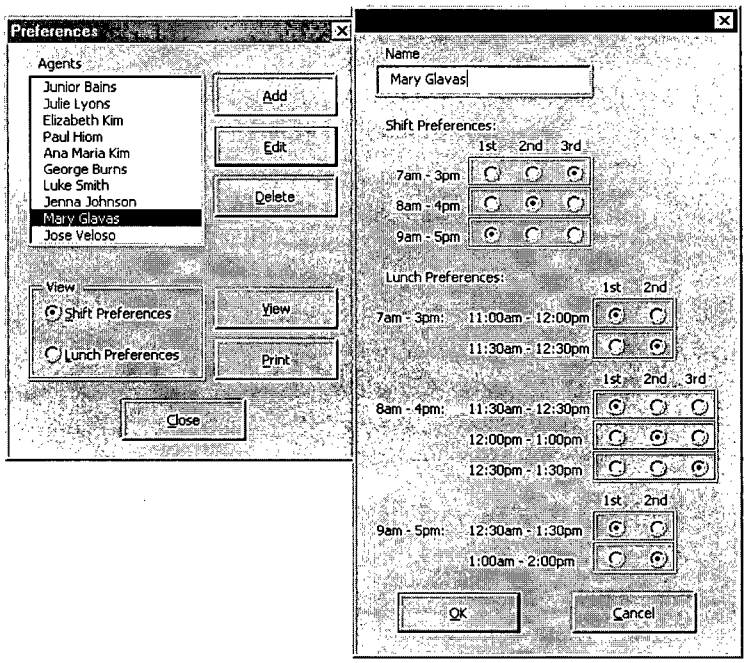
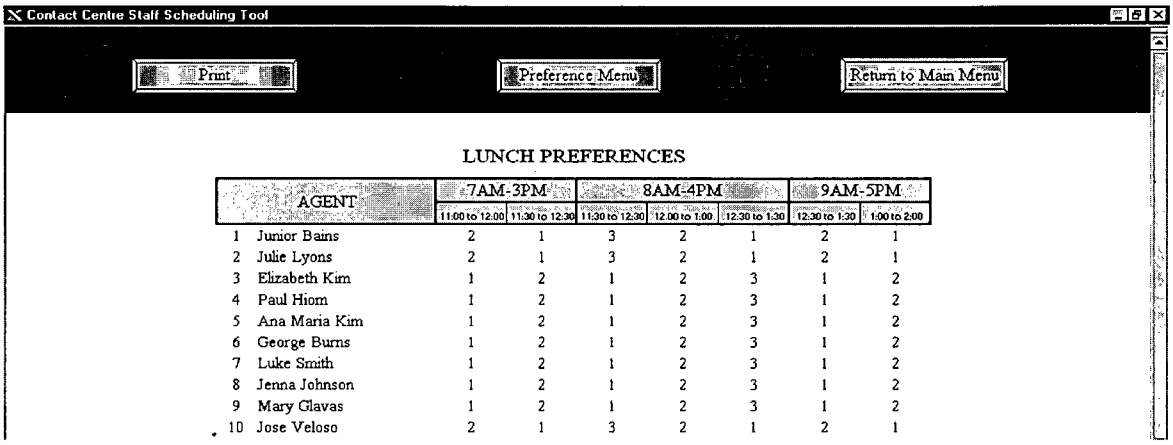


***Figure 18**: Agent Preferences*

Once each agent's preferences have been input, the data can be viewed or printed using the buttons on the Preferences form above. A partial screen of the lunch preferences view is show below. A similar screen exists for shift preferences.



LUNCH PREFERENCES

	AGENT	7AM-3PM		8AM-4PM			9AM-5PM	
		11:00 to 12:00	11:30 to 12:30	11:30 to 12:30	12:00 to 1:00	12:30 to 1:30	12:30 to 1:30	1:00 to 2:00
1	Junior Bains	2	1	3	2	1	2	1
2	Julie Lyons	2	1	3	2	1	2	1
3	Elizabeth Kim	1	2	1	2	3	1	2
4	Paul Hiorn	1	2	1	2	3	1	2
5	Ana Maria Kim	1	2	1	2	3	1	2
6	George Burns	1	2	1	2	3	1	2
7	Luke Smith	1	2	1	2	3	1	2
8	Jenna Johnson	1	2	1	2	3	1	2
9	Mary Glavas	1	2	1	2	3	1	2
10	Jose Veloso	2	1	3	2	1	2	1

***Figure 19**: View of Lunch Preferences*

8.3 Generating Schedules

The user can generate schedules using button 1 found on the main menu. However the user is prompted to generate forecasts first. The call volume forecasts are provided and charted along with forecasted staffing requirements. Creating this step enables users to use judgement to modify forecasts rather than creating schedules automatically without this step.

Future forecasts can be generated for one week or for one day, which can be today. Default values required to compute agent requirements are provided but can be changed manually by the user. Default values can also be edited using button 4 on the main menu.

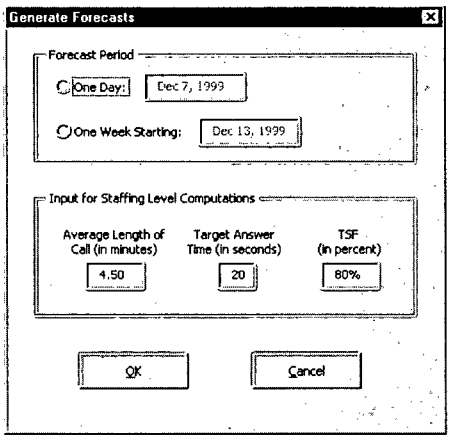


***Figure 20**: Generating Forecasts*

The resulting screen is shown below. In addition to modifying call volume forecasts, the user can chart a past day's call volume to compare (data must be inputted manually), as well as print and save the forecasts.

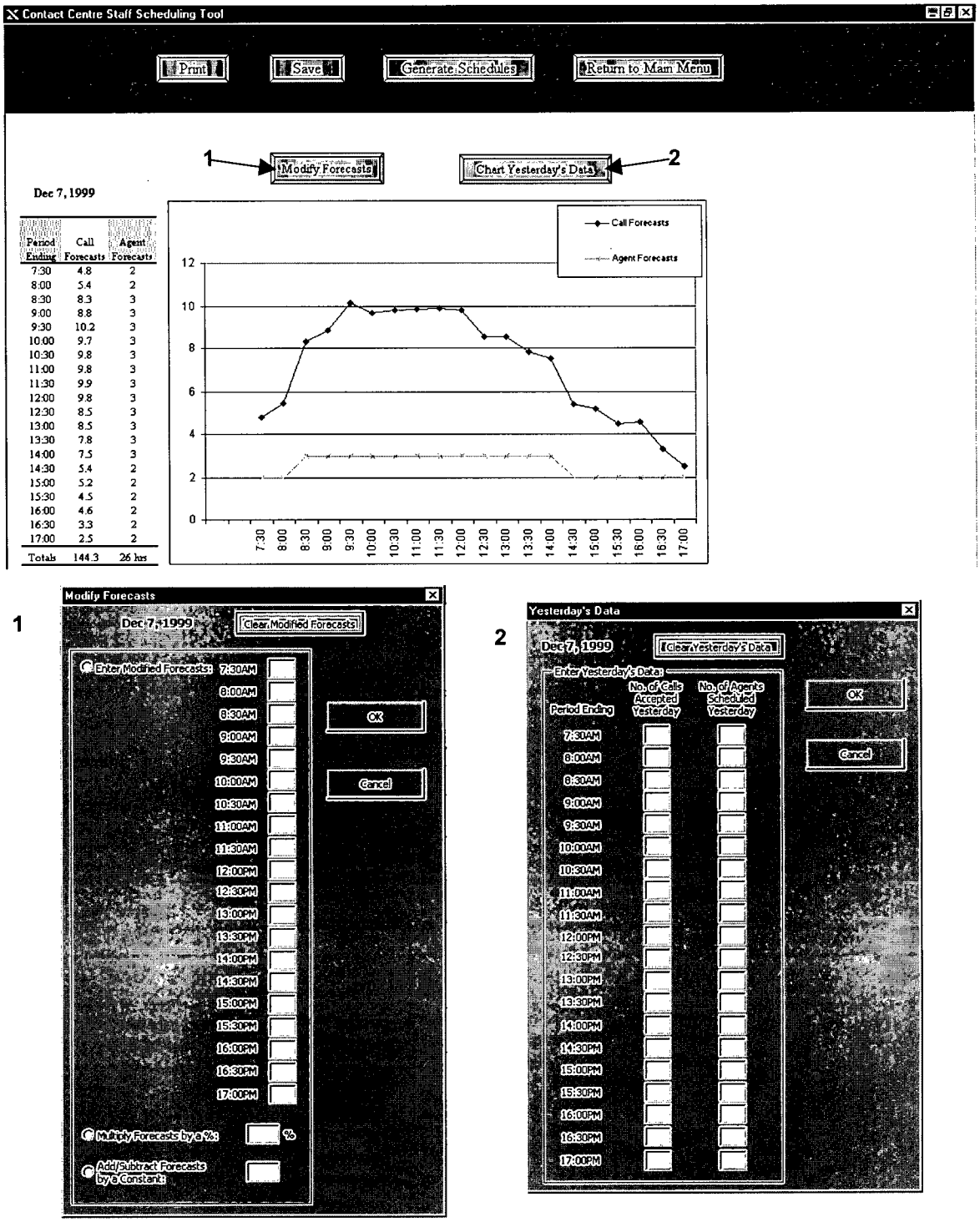


***Figure 21**: Modifying Call Volume Forecasts*

Once the user is satisfied with the forecasts, schedules can be generated. The button is found on the top of the screen and will prompt the user to indicate which agents will be working on the forecasted day (the form is shown below) as well as the number of blocks each agent will be available to handle inbound calls (partial screen is shown below). The

tool will automatically divide the total minimum half-hour blocks required to handle inbound calls equitably among CSRs. However the user has the option of adjusting these numbers.

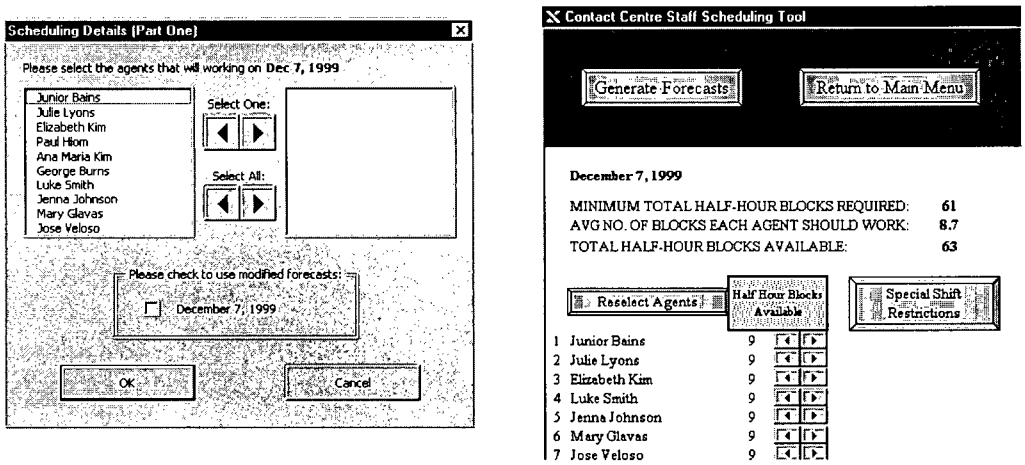


***Figure 22**: Agent Details for Schedules*

The user can also input shift and lunch restrictions. A shift restriction may be needed for an agent who will be leaving for the day at 1pm and will not be taking a lunch.

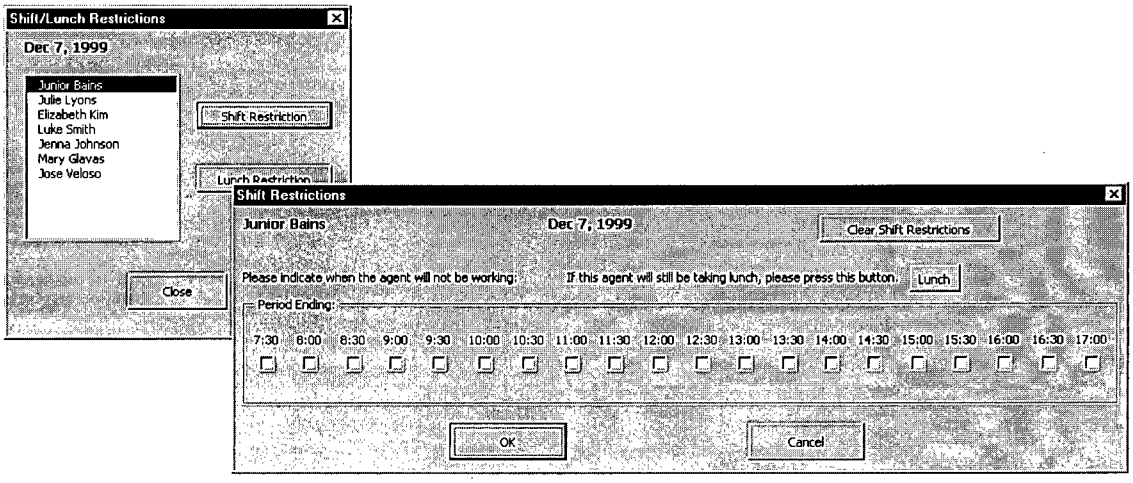
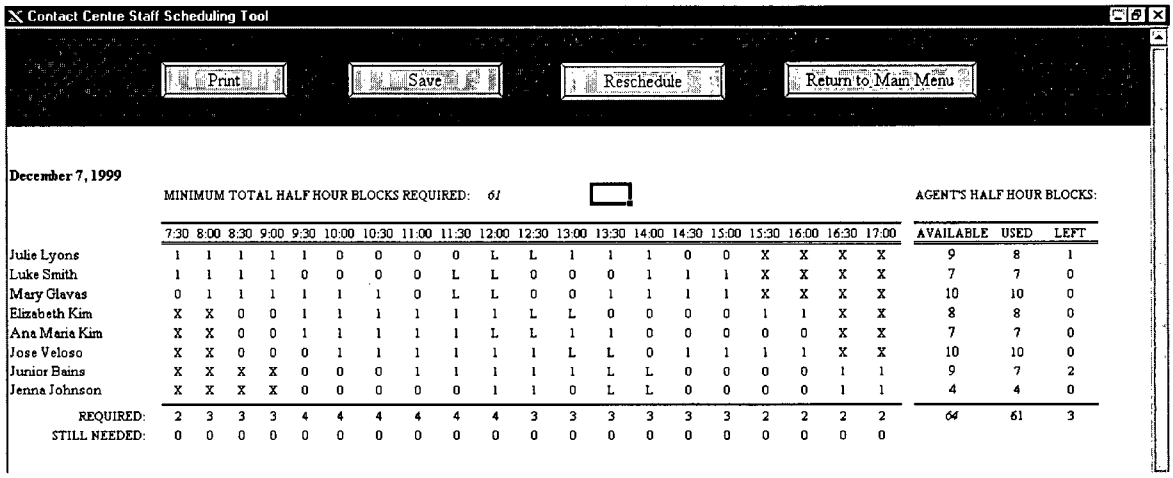


***Figure 23**: Shift and Lunch Restrictions*

Finally a schedule can be generated (see Figure 24).



Contact Centre Staff Scheduling Tool

Print | Save | Reschedule | Return to Main Menu

December 7, 1999

MINIMUM TOTAL HALF HOUR BLOCKS REQUIRED: 61

AGENT'S HALF HOUR BLOCKS:

	7:30	8:00	8:30	9:00	9:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30	16:00	16:30	17:00	AVAILABLE	USED	LEFT
Julie Lyons	1	1	1	1	1	0	0	0	0	L	L	1	1	1	0	0	X	X	X	X	9	8	1
Luke Smith	1	1	1	1	0	0	0	0	L	L	0	0	0	1	1	1	X	X	X	X	7	7	0
Mary Glavas	0	1	1	1	1	1	1	0	L	L	0	0	1	1	1	1	X	X	X	X	10	10	0
Elizabeth Kim	X	X	0	0	1	1	1	1	1	1	L	L	0	0	0	0	1	1	X	X	8	8	0
Ana Maria Kim	X	X	0	0	1	1	1	1	1	L	L	1	1	0	0	0	0	0	X	X	7	7	0
Jose Veloso	X	X	0	0	0	1	1	1	1	1	1	L	L	0	1	1	1	1	X	X	10	10	0
Junior Bains	X	X	X	X	0	0	0	1	1	1	1	1	L	L	0	0	0	0	1	1	9	7	2
Jenna Johnson	X	X	X	X	0	0	0	0	0	1	1	0	L	L	0	0	0	0	1	1	4	4	0
REQUIRED:	2	3	3	3	4	4	4	4	4	4	3	3	3	3	3	3	2	2	2	2	64	61	3
STILL NEEDED:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			

***Figure 24**: Schedule*

5.4 Viewing Past Performance

Button 2 on the main menu allows the user to go back and chart the forecasts against the subsequent actual data. Various series can be charted and the user is given the choice to change them while he or she is using this particular module. If the user had modified the forecasts, he or she can manually input these values.

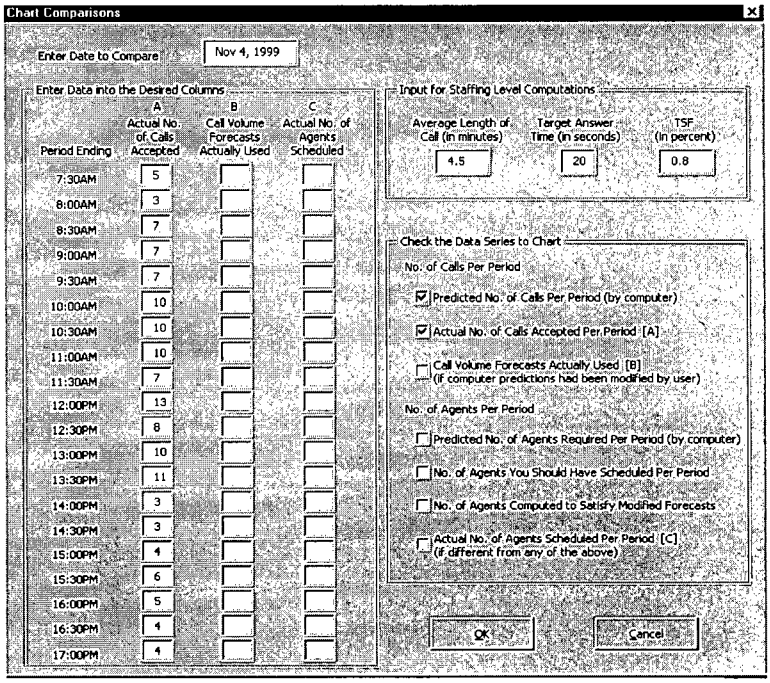


***Figure 25**: Charting Past Performance*

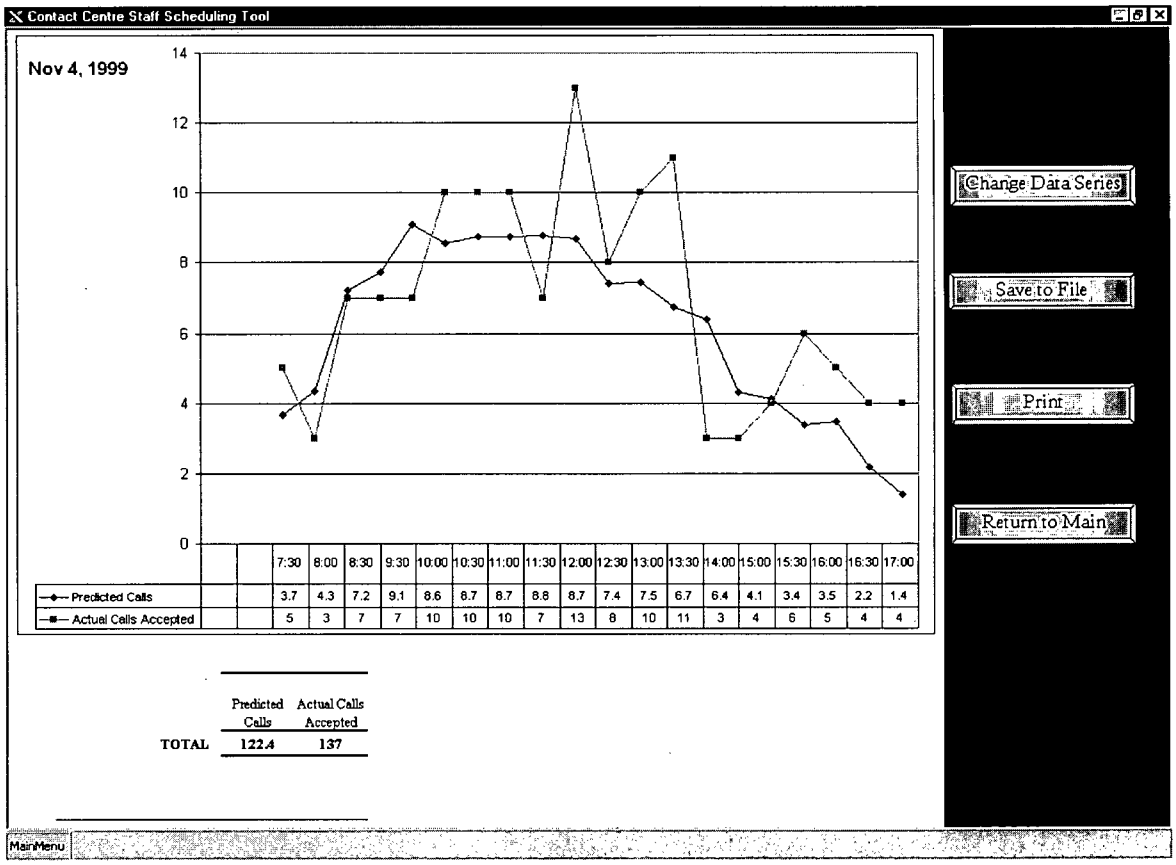


***Figure 26**: Predicted Calls vs. Actual Calls*

5.5 Edit Default Values

The screen for editing several default values that are used several times throughout the tool follows. This screen is reached by button 4 on the main menu.

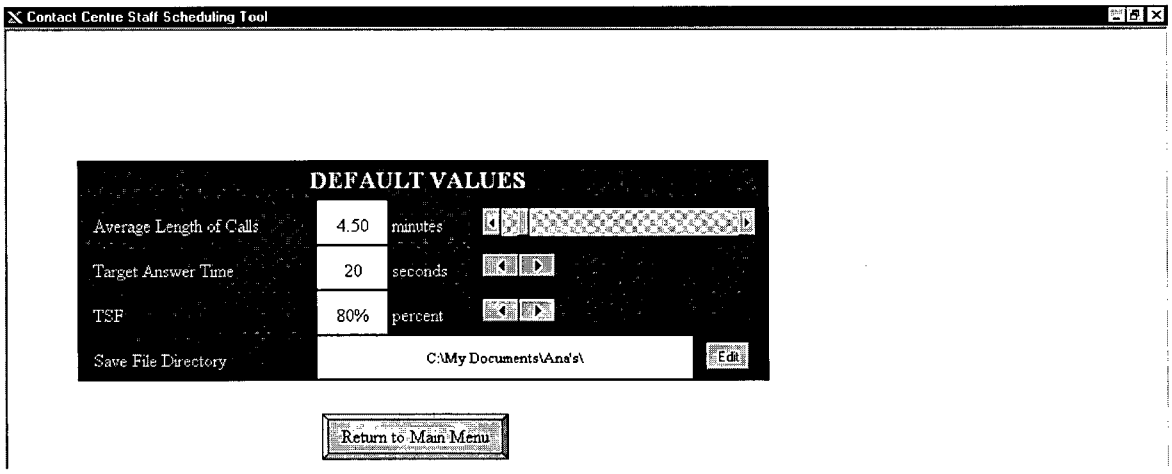


***Figure 27**: Editing Default Values*

IX. IMPLEMENTATION

In order for the project to be considered complete the tool must be successfully implemented. For application development, end user involvement throughout the development process is critical for the ultimate implementation. This issue is a small part of an emerging area of study known as human computer interaction. Specific to application development, there are four basic steps to consider in achieving a successful implementation [22]:

1. Identify the user.
2. Set software functionality.
3. Design the user interface to the product.
4. Test the user interface.

The project sponsor was the Client Services Manager. However, the main users of the tool were intended to be three of the more experienced CSRs who have been involved in the development process. The "test run" of forecasts conducted very early on in the project stage helped to "buy-in" the users of the tool's ultimate functionality. As well, sample schedules were provided with an early version of the scheduling policy to answer any questions and receive any suggestions that CSRs and the Client Services Manager had.

Once the desired functionality was developed, an interface that would meet the needs of the user also needed to be developed. The previously developed forecasting model was not being used simply because it did not have an interface. Human Computer Interaction specialists suggest cognitive ergonomics should be considered in developing user interfaces which is mainly common sense. Examples of this can be seen in the tool. The same buttons are used throughout the application and they are always placed in a similar setting (blue panel). Using EXCEL to develop the application greatly facilitated the ability to create a friendly user interface since the users were already familiar with EXCEL and the way it uses forms to input external data.

The interface also needs to be robust. Although the program looks very much like an EXCEL file, all of the original menus and any other functionality not relevant to the tool have been hidden. Screens cannot be modified nor can any cell be selected except for the worksheet that provides the schedule. The users are not restricted to buttons on the scheduling worksheet to allow complete flexibility in manually adjusted schedules. However the worksheet is recreated each time a schedule is generated and expunged once the user leaves the screen to ensure its consistency. Error checks are built in throughout the tool to let users know when input is invalid. Invalid inputs can be weekend dates for forecasting and dates or numeric values in incorrect format. Prior to generating a schedule, the tool will let the user know if too few agents were selected to work, the total of each agent's available log-in blocks does not meet total minimum staffing requirements, or there are not enough agents available a minimum number of blocks to satisfy staffing requirements.

Since the tool was built with EXCEL, it required no additional hardware or software to implement the tool. The users have had two training sessions and they are currently testing the tool.

X. CONCLUSION & FUTURE ISSUES

Although the application was implemented in December, it was not used for scheduling staff until the following March. PH&N has moved from using its reputation as the main marketing strength towards a more aggressive growth strategy. Several new products were introduced beginning in mid-November for retail mutual fund clients. This has resulted in a sudden and major increase in total call volume. The minimum investment for these products is only $1000 and is attracting a significant number of interested buyers. Total call volume is about 200% the volume predicted by the application.

The application indicated that up to 15 agents are needed to cover the shifts required to service forecasted call volume. Since there are only 7 agents currently, all agents had had to log-in for the entire workday. No time remained to accomplish many of the other tasks that agents are assigned to do.

However, the tool has been a considerable advantage for the Contact Centre in proposing for more resources. In the past, even with the 7 agents, it was difficult to ensure that each agent had enough time for "other" work or tasks unrelated to incoming calls. The tool has been powerful in showing the relationship between the staff to workload ratio and service levels. Recently, the Contact Centre was able to hire two more agents.

The forecasting model will need to be analyzed and reparamaterized yearly while the tool is in use. The model reparamaterized on two years of data should provide better predictive accuracy. At the onset of the project, it was agreed that the COE would take on the responsibility.

Only about 20% of calls received by retail mutual fund clients are sent to the Fund Advisor queue. As the number of calls to the Fund Advisor increase, using this tool was considered as a possibility. However a call centre for the Fund Advisors would be a revenue centre and staffing to optimize revenue generation requires different techniques.

As the Internet becomes more and more prevalent and the Contact Centre becomes capable of receiving enquiries online, inbound calls may decrease significantly in the long run and the queueing system will undergo a fundamental change.

BIBLIOGRAPHY

[1] Phillips, Hager & North Investment Management Ltd. <http://www.phn.com>

[2] Onyx Software Corp. November 1999. <http://www.onyx.com/FrontOffice/default.asp>

[3] Hiom, Paul. *Simulation Modeling as a Decision Analysis Support Tool: A Case Study at the PH&N Telephone Centre.* Unpublished MScB thesis, Faculty of Commerce, The University of British Columbia, March 2000.

[4] Sanegre, Rafael. *Scheduling Customer Service Representatives for the Workers' Compensation Board of British Columbia.* Unpublished MScB thesis, Faculty of Commerce, The University of British Columbia, July 1998.

[5] TCS Management Group, Inc. *Staffing for the Call Centre.* 1996.

[6] Reynolds, Penny. "Staffing for the Customer Service Center." *ICSA Journal*, Spring 1992.

[7] Harris, C.M., Hoffman, K.L., and Saunders, P.B. "Modeling the IRS Telephone Taxpayer Information System." *Operations Research* 35, 4 (1987): 504-523.

[8] Gross, D., Harris, C. M. *Fundamentals of Queueing Theory.* John Wiley & Sons, Inc., 1974.

[9] TCS Management Group, Inc. September 1999 <http://www.tcsmgmt.com>

[10] Taske Technology, Inc. September 1999 <http://www.taske.com>

[11] Portage Communications, Inc. September 1999 <http://www.portagecommunications.com>

[12] Pipkins, Inc. September 1999 <http://www.pipkins.com>

[13] Reynolds, Penny. "Automating Workforce Management in the Customer Service Center." *ICSA Journal*, Spring 1993.

[14] Reynolds, Penny. "Improving the Bottom Line in Your Call Centers." *ICSA Journal*, Fall 1992.

[15] Nortel Networks Corp. 3 December 1999 <http://www.nortelnetworks.com/products/01/callcenter/acd.html>

[16] Klenke, Maggie. "ACDs Get Skills-Based Routing." *Business Communication Review*, July 1995, 48-51.

[17] Morris, J.G. and Showalter, M.J. "Simple Approaches to Shift, Days-Off, and Tour Scheduling Problems." *Management Science* 29, 8 (1983): 942-950.

[18] Bechtold, S.E. and Jacobs, L.W. "Implicit Modeling of Flexible Break Assignments in Optimal Shift Scheduling." *Management Science* 36, 11 (1990): 1339-1351.

[19] Baker, K.R. "Scheduling a Full-Time Workforce to Meet Cyclic Staffing Requirements." *Management Science* 20, 12 (1974): 1561-1568.

[20] Bechtold, S.E., Brusco, M.J., and Showalter, M.J. "A Comparative Evaluation of Labor Tour Scheduling." *Decision Sciences* 22, 4 (1991): 683-699.

[21] Gaballa, A. and W. Pearce, "Telephone Sales Manpower Planning at Quantas," *Interfaces*, 9,3,(1979), 1-9.

[22] Barfield, Jon. *The User Interface: Concepts & Designs*. Addison-Wesley Publishing Company, 1993.

APPENDIX I. SKILLS-BASED ROUTING

Traditionally, ACDs have focused on spreading incoming calls among call centre agents so that each agent handles an equitable share of the load as well as ensuring that the caller has the best chance of being served quickly.

Skills-based routing is a shift in thinking that is more widely entering the area of call centre management. It is a queueing system that focuses on routing an incoming call to the available agent whose skills are based matched to the caller's needs. The system is designed to queue a call to more than one agent group simultaneously. An agent group or skill group consists of agents with the skills that a specific call type requires such as product knowledge. An agent can also be logged into more than one skill group. However, an agent may be logged into three skill groups but only get calls from the first skill group if the first one consumes a high percent of the total call volume. Skills can be prioritized to reduce the number of high volume calls going to an agent possessing a different low volume consumption but unique skill.

Formulas based on traditional queueing systems cannot be applied. Thus, skills-based routing technology typically comes with a simulation module. A routing simulator allows the user to test the effects of different call routing rules and skill assignments on service level, given the forecasted workload and staff schedule information.

Skills-based routing is very attractive to call centres where agents do not have the same set of skills. Queueing to multiple groups in this situation is more effective and can provide better overall customer service. Although the concept of skills based routing has been around for many years, it has not gained the momentum it seems to deserve. Queuing calls to more than one agent group and allowing agents to log-in to more than one skill group creates a complex situation in which a large number of variables interact. Skills-based routing software products vary considerably in features and still have many limitations. The newest release of PBX and ACD technology as well as an IVR unit to perform the front-end sorting of incoming calls is also generally required.

All of the unique combinations of caller needs must be defined. If a call centre handles sales and repair inquiries for refrigerators, washers, dryers, and microwave ovens. A call regarding a refrigerator purchase would be one need and refrigerator repair another. Continuing in the same manner would result in a total of eight combinations. If the call centre needed to support both English speaking customers and a large Chinese speaking customer base, there would be another eight combinations. The script used by the IVR must not only lead the customer to the right option but also prevent a premature selection.

Agent skills must also be defined. Although skills-based routing software tends not to limit the number of skill groups, most ACDs restrict agents from actually logging-in to more than three or four. It would then be impossible to use some of the skills possessed by experienced agents who have more than four skills.

Another issue is that the agents who are logged-in can change every period for breaks and lunches. An agent may call in sick, go on holidays, get trained in a new skill, etc. Routing patterns and the agent skills matrix must be manually updated with every change to achieve the full value of skills-based routing. Managing a call centre using skill-based routing can be very challenging, and increasingly so with every additional call type, agent, and skill type.

APPENDIX II. ACD OUTPUT

Queue Report

Wed Sep 29 19:00:24 1999 Page 1

ACD Queue Activity Periodic Totals

Date.....: 09/29/1999
Day of Week...: Wednesday
Queue Group...: PH&N-Vancouver/Cust. Service

_Period Ending	Avg Agents Available Transfers	Required Agents	Calls Accepted	Calls Answrd	ASA	Longest Ans Delay	Calls Abandoned No.	%	Avg Wait	TSF %	Total
ACD-DN.....: 6200											
Queue Name....: Client Services											
7:30	2	1	4	4	0:04	0:08	0	0.00%	****	100	0
8:00	2	2	5	4	0:02	0:04	0	0.00%	****	100	0
8:30	3	2	4	5	0:14	0:30	0	0.00%	****	80	0
9:00	3	3	12	11	0:09	0:22	1	8.33%	0:26	83	2
9:30	4	3	11	11	0:19	0:50	0	0.00%	****	63	3
10:00	5	4	10	10	0:12	0:38	0	0.00%	****	80	0
10:30	5	3	10	11	0:05	0:08	0	0.00%	****	100	2
11:00	5	4	7	7	0:08	0:24	0	0.00%	****	85	0
11:30	4	3	3	3	0:48	1:54	0	0.00%	****	33	0
12:00	4	2	9	8	0:09	0:24	1	11.11%	0:08	88	1
12:30	4	2	8	8	0:08	0:22	0	0.00%	****	87	2
13:00	5	3	7	7	0:08	0:24	0	0.00%	****	85	0
13:30	4	3	9	8	0:07	0:12	1	11.11%	0:02	100	2
14:00	3	2	9	9	0:09	0:20	0	0.00%	****	100	2
14:30	3	3	5	5	1:24	4:04	0	0.00%	****	60	1
15:00	4	3	11	10	0:10	0:46	1	9.09%	0:16	90	3
15:30	3	2	4	4	0:10	0:24	0	0.00%	****	75	0
16:00	3	2	8	8	0:17	0:56	0	0.00%	****	75	2
16:30	2	2	2	2	0:12	0:20	0	0.00%	****	100	0
17:00	2	2	2	2	0:06	0:10	0	0.00%	****	100	0
TOTAL (for 6200):			140	137	0:13		4	2.86%	0:13	85	20

Detail Report for location: Vancouver
Number of records involved in report : 20

----- end of report ---

Agent Report

Thu Oct 28 18:59:49 1999 Page 1

Daily Statistics Summary by Individual Agent

Queue Group: PH&N-Vancouver/Cust. Service
Date.....: 10/28/1999
Day of Week: Thursday

Agent Name	Agent ID	Calls Answrd	Talk Time Total	Talk Time Avg	Hold Time Total	Hold Time Avg	Not Ready Total	Not Ready Avg	Wait Time Total	Wait Time Avg	DN Calls In	DN Calls Out	NonACD Time	Logged-In Time
Bissillion	1021	14	26:28	1:51	0:56	0:04	2:34	0:11	1:24:54	6:03	1	9	12:32	2:06:54
Catania	1002	11	52:02	4:44	6:22	0:35	3:08:58	6:28	1:25:48	6:33	3	21	32:50	6:07:18
Dawood	1014	32	1:27:52	2:44	1:58	0:04	1:21:28	2:02	2:11:28	3:23	3	33	52:16	5:53:50
Peterson	1018	14	22:52	1:38	3:04	0:13	12:42	0:43	2:11:04	7:35	7	26	29:48	3:18:58
Seegers	1030	28	55:34	1:47	1:24	0:03	1:08:22	1:35	2:26:28	4:59	5	42	53:16	5:24:08
Somerville	1031	42	1:14:10	1:46	6:14	0:09	2:32:50	1:49	2:35:10	3:38	7	20	37:06	7:03:30
TOTAL (for 10/28/1999):		141	5:18:58	2:13	19:58	0:08	8:26:54	1:55	12:14:52	4:42	26	151	3:37:48	29:54:38

Summary Report for location: Vancouver
Number of records involved in report : 70

----- end of report ---

APPENDIX III. SCHEDULING HEURISTIC

Shift Selection

The purpose of this heuristic is to maximize total shift preferences while ensuring that there are enough agents to satisfy staffing requirements, and that shift constraints are met. For example, if 3 agents are required from 7am to 7:30am, then at least 3 agents must be scheduled to work the 7am to 3pm shift. If more agents are selected to work the 7am to 3pm shift, then only 3 agents will be scheduled to log-in to the ACD.

The minimum number of agents that are required to work each shift is computed based on the staffing level requirements. Below is a possible set of staffing level requirements for logging-in to the ACD.

Period Ending

7:30	8:00	8:30	9:00	9:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30	16:00	16:30	17:00
1	2	3	3	3	4	4	4	4	4	4	4	3	3	3	3	3	2	2	2

Set of Staffing Level Requirements, Example 1

Based on the table, it is clear that at least 2 agents will have to work the 7-3 shift to cover the first two periods and at least 2 agents will have to work the 9-5 shift to cover the last two periods. In other words, if only one agent was scheduled to start at 7, there will only be one agent working from 7 to 8 at which time the next shift starts. As a result, the contact centre will not have the 2 agents required to meet predicted demand from 730 to 8.

At this point, only the 2 agents working 7-3 are available in the periods ending 8:30 and 9 since the other 2 agents are not available until 9am. Assigning one agent to work 8-4 is one option. The other alternative is to assign another agent to work 7-3 and another agent to work 9-5. Although the second alternative may improve total shift preferences, the first option is used in shift selection. Minimizing staffing requirements is more important.

The other possibility is that the additional agents required for the periods ending at 8:30 and 9 is not equal to the additional agents required in the periods ending at 3:30 and 4 (see Figure 15).

Period Ending

7:30	8:00	8:30	9:00	9:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30	16:00	16:30	17:00
1	2	3	4	4	4	4	4	4	4	4	4	3	3	3	3	3	2	2	2

Set of Staffing Level Requirements, Example 2

In this case, 2 agents can be assigned the 8-4 shift. Alternatively, 1 agent can be assigned the 8-4 shift and another agent assigned the 7-3 shift. Since neither option impacts the number of shifts required for the schedule, preferences can be used to dictate the

solution. Initially this was the option used. However assigning agents to the 8-4 shift when there was a choice, improved the lunch selection heuristic. Agents working 8-4 can be assigned one of three lunch breaks whereas only two lunch breaks are open to the other two shifts. Thus the 8-4 shift allows more flexibility in minimizing the number of agents away for lunch during any period.

So far the heuristic considers only the first four and last four periods when determining shift requirements. If the maximum number of agents required in any period between 9am to 3pm is greater than total number of shifts required thus far, this difference adds to the number of total shifts required. Although all shifts cover this interval, the 8-4 shift is used, again because of the flexibility it affords.

Also, in determining the minimum number of CSRs required for the schedule, lunch breaks must be considered. The two lunch breaks for the 7-3 shift overlap in the period ending at noon. Thus, if two agents are scheduled for the 7-3 shift, two agents will not be available to log-in in the period ending at noon. The same concept is applied to all shifts and a minimum of total shifts can be estimated.

Although having less than this estimated minimum might result in understaffing during the lunch interval, this minimum is just a recommendation. The user is given the flexibility of generating a schedule then manually changing lunch breaks out of the official time intervals if necessary to meet staffing requirements rather than adding additional agents.

The CSRs are assigned to this distribution of shift requirements according to preferences. The agents rank the shifts in order of preference, one being the most and three being the least preferred. It is possible to indicate indifference towards two or all start times. For example, an agent may be indifferent between the 7-3 and 8-4 shifts, but would prefer either of these to the 9-5 shift. In this case, the 7-3 and 8-4 shifts would both be ranked one and the 9-5 shift would be ranked two.

The method used to assign is as follows. The number of each type of shift has been predetermined. For each shift, starting with 7-3, CSRs are assigned to the shift if the number of CSRs required to work this shift equal the number of CSRs who ranked this shift first. However a shift or shifts may have fewer or more than the required CSRs who ranked the particular shift first. The shift where there are fewer than the required agents ranking it first is assigned to CSRs next before any shift where more than the required are available to select from.

This assignment heuristic method is used to prevent the following situation. 3 agents may be required to work 7-3 but only 2 agents ranked this shift first. Then an agent who ranked the 7-3 shift second must be selected. There may be two agents who ranked the 7-3 shift second, one who ranked 8-4 first and one who ranked 8-4 second. If fewer than the required agents ranked 8-4 first then, the agent who ranked 7-3 second but 8-4 first should not be selected to work the 7-3 shift. He or she would improve overall preferences by being assigned to the 8-4 shift.

One issue concerning the shift selection heuristic is that it does not consider agents with shift restrictions. Shift restrictions vary considerably and do not occur frequently. Considering all possible shift restrictions would have greatly increased the complexity of the heuristic. Shift restrictions may be for an agent who will not be available all morning in order to attend a meeting, for two periods so that he or she can cover an agent on lunch who was assigned to handle walk-in for the entire day, etc. Agents with shift restrictions will tend to be available a limited number of periods to log-in; thus, the impact in minimizing the total number of shifts is minor.

However, an agent may have asked to start work at 7 and work straight through lunch on the day being scheduled in order to leave one hour earlier. This agent is basically working a 7-3 shift. The heuristic will not consider this agent for a 7-3 shift; thus there will be a redundant agent. That is, the heuristic may compute that a total of 8 agents are required. If the user selects 8 including this agent working 7-2, the tool will recommend that one more agent be selected.

Users are not required to select the recommending number of agents and knowing this can help them decide in selecting available agents. Nevertheless, having more agents as long as they are available reduces the number of blocks each agent will have to log-in on average which increases the flexibility in managing everyone's needs.

In addition, although the 8-4 shift provides more flexibility for the lunch selection heuristic, favouring the 8-4 shift results in having only the minimum number of agents working 7-3 and 9-5. Then in the first and last two periods of the workday, all available agents will likely be scheduled to log-in. To ensure that these agents will be able to devote their time to answering incoming calls without interruption, there must be at least one other agent not logged-in to handle urgent situations such as walk-ins. Since the agents currently working 7-3 and 9-5 always prefers 8-4 second, this functionality would have no impact right now. As well the affected periods are slow enough that it is easy to stay logged-in for the entire hour. It is a matter of whether minimizing the number of agents away at the same time for lunch is more important than having the extra agent or agents available in the first and last two periods.

Lunch Selection

The purpose of this heuristic is to minimize the number of agents away for lunch. However, agents' preferences for lunch are considered where possible.

Agents must rank lunch breaks for each shift in the same manner they rank shifts. The heuristic begins by selecting each of the agent's first lunch preference for the shift he or she has been assigned to. If more than one lunch break has been ranked as one or first for the shift due to indifference, then one is randomly assigned to the agent. A possible set of first lunch preferences is shown below (area left of dashed line in Figure 16).

Next, the heuristic aims to minimize the number of agents away each period. Starting from the top of the list, each agent is moved to the lunch that minimizes the sum of the

total number of agents away in each of the impacted periods. If changing the lunch makes no difference to this sum, then the agent's lunch assignment is left as is.

	Period Ending									
	11:30	12:00	12:30	13:00	13:30	14:00	Shift	11:30	12:00	12:30
Agent 1	*L*	*L*					7-3		*L*	*L*
Agent 2	L	L					7-3	L	L	
Agent 3	L	L					7-3	L	L	
Agent 4				L	L		8-4			
Agent 5			L	L			8-4			L
Agent 6				L	L		9-5			
Agent 7				L	L		9-5			
Total No. of Agents Away	3	3							3	2

First Preferences for Lunch

Agent 1 is currently assigned lunch from 11am to noon. The sum of the total number of agents away for the periods he or she is assigned to take lunch is 6. If Agent 1's lunch were moved to the other possible option this sum would decrease to 5 agents. Figure 17 shows the results once this is applied to each agent.

	Period Ending						
	11:00	11:30	12:00	12:30	13:00	13:30	Shift
Agent 1		L	L				7-3
Agent 2	L	L					7-3
Agent 3	L	L					7-3
Agent 4				L	L		8-4
Agent 5			L	L			8-4
Agent 6					L	L	9-5
Agent 7					L	L	9-5
Total No. of Agents Away							

Modified Lunch Selections

Changing any one lunch will not improve the overall lunch schedule. There are only two periods where 3 agents are away for lunch at the same time. This is not the only possible combination but this combination does maximize overall lunch preferences. However, it is biased towards the top of the list. For instance, any of the top three agents could have been moved to their second option. The shift selection heuristic takes care of this since agents are selected for each shift randomly rather than alphabetically or in order from a list.

In some cases, it is still possible to shift lunch breaks to reduce the number of agents away at the same time even after this method has been applied to each agent. The decision to move an agent assumes that the lunch breaks for agents below will remain unchanged. Applying this method twice solves this.

Generating Work Schedules

Once all agents have been assigned a shift and lunch break, each agent's 14 work periods have to be scheduled either to log-in to the ACD or for other work. The schedule will look something like a sample below before this last heuristic is applied.

																					BLOCKS:		
	7:30	8:00	8:30	9:00	9:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30	16:00	16:30	17:00	AVAILABLE	USED	LEFT
Albert										L	L						X	X	X	X	8	0	8
Betty									L	L							X	X	X	X	9	0	9
Charles									L	L							X	X	X	X	4	0	4
Darlene	X	X									L	L							X	X	8	0	8
Elliot	X	X										L	L						X	X	9	0	9
Frank	X	X	X	X									L	L							7	0	7
George	X	X	X	X									L	L							7	0	7
	2	2	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2	2	2	2	52	0	52
	2	2	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2	2	2	2			

Sample Work Schedule Template

The blocks available column is data inputted by the user to satisfy the one of the requirements of the scheduling policy, which is to allow users the autonomy to indicate the number of periods or blocks they will be available to log-in to the ACD as long as minimum total requirements are met. The above schedule will require a total of 52 blocks to meet incoming calls within the target service level of TSF 80% within TAT 20 seconds.

Overall a desirable schedule is one that minimizes the number of single periods assigned to either work type, the number of times each agent is switched between work types, the number of periods that are under or over staffed.

APPENDIX IV. SCHEDULING HEURISTIC CODE

The lines of code required for the heuristics amounted to approximately 2250 out of an estimated total of 7000. The programming language used by EXCEL 97 is Visual Basic for Applications 5.0.

Shift Selection Heuristic

```
'*****************************************************************************
'Procedure: SelectShifts
'Arguments: myDay, the day currently being scheduled
'Purpose: Assigns shifts to agents.
'*****************************************************************************
Sub SelectShifts(myDay)
Dim myWorkingNames As Range

  Set myWorkingNames = Range(Range("Schedule" & myDay).Cells(1).Address, Range("Schedule" &
    myDay).End(xlDown).Address)

  Sheets("Schedules").Range(Range("Schedule" & myDay).Cells(myWorkingNames.Cells.Count + 1).
    Address, Range("Schedule" & myDay).Cells(25).Address).EntireRow.Hidden = True

  ShiftSevenToThree (myDay)

  ShiftEightToFour (myDay)

  ShiftNineToFive (myDay)

  SelectFinalShifts (myDay)

  Range("AA1").EntireColumn.ClearContents
End Sub


'*****************************************************************************
'Procedure: SelectUnderMinShifts
'Arguments: myDay, the day currently being scheduled
'Purpose: Assigns shifts to agents if the number of agents is below the required minimum.
'*****************************************************************************
Sub SelectUnderMinShifts(myDay)
  Dim myWorkingNames As Range
  Dim C As Integer

  Set myWorkingNames = Range(Range("Schedule" & myDay).Cells(1).Address, Range("Schedule" &
    myDay).End(xlDown).Address)

  For myCell = 1 To myWorkingNames.Cells.Count
    Sheets("Schedules").Select
    If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" Then
      myName = Range(myWorkingNames.Cells(myCell).Address).Value
      Sheets("ShiftPreferences").Select
      Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
      Selection.Find(What:=myName, After:=ActiveCell, _
        LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
```

```
    SearchDirection:=xlNext, MatchCase:=False).Activate
If ActiveCell.Offset(, 1).Value = "1" And ActiveCell.Offset(, 2).Value <> "1" And
    ActiveCell.Offset(, 3).Value <> "1" Then
    Sheets("Schedules").Select
    For C = 17 To 20
        Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
    Next C

ElseIf ActiveCell.Offset(, 1).Value <> "1" And ActiveCell.Offset(, 2).Value = "1" And
    ActiveCell.Offset(, 3).Value <> "1" Then
    Sheets("Schedules").Select
    For C = 1 To 2
        Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
    Next C
    For C = 19 To 20
        Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
    Next C

ElseIf ActiveCell.Offset(, 1).Value <> "1" And ActiveCell.Offset(, 2).Value <> "1" And
    ActiveCell.Offset(, 3).Value = "1" Then
    Sheets("Schedules").Select
    For C = 1 To 4
        Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
    Next C

ElseIf ActiveCell.Offset(, 1).Value = "1" And ActiveCell.Offset(, 2).Value = "1" And
    ActiveCell.Offset(, 3).Value <> "1" Then
    Sheets("Schedules").Select
    myShift = Int((2 * Rnd) + 0)
    If myShift = 1 Then
        For C = 17 To 20
            Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
        Next C
    Else
        For C = 1 To 2
            Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
        Next C
        For C = 19 To 20
            Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
        Next C
    End If

ElseIf ActiveCell.Offset(, 1).Value <> "1" And ActiveCell.Offset(, 2).Value = "1" And
    ActiveCell.Offset(, 3).Value = "1" Then
    Sheets("Schedules").Select
    myShift = Int((2 * Rnd) + 0)
    If myShift = 1 Then
        For C = 1 To 2
            Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
        Next C
        For C = 19 To 20
            Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
        Next C
    Else
        For C = 1 To 4
            Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
```

```
            Next C
          End If

        ElseIf ActiveCell.Offset(, 1).Value = "1" And ActiveCell.Offset(, 2).Value <> "1" And
          ActiveCell.Offset(, 3).Value = "1" Then
          Sheets("Schedules").Select
          myShift = Int((2 * Rnd) + 0)
          If myShift = 1 Then
            For C = 17 To 20
              Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
            Next C
          Else
            For C = 1 To 4
              Range(myWorkingNames.Cells(myCell).Offset(, C + 1).Address).Value = "X"
            Next C
          End If
        End If
      End If
    Next myCell
End Sub

'*****************************************************************************************
'Procedure: ShiftSevenToThree
'Arguments: myDay, the day currently being scheduled
'Purpose: Gets the number of required 7-3 shifts and assigns them to agents according to preferences.
'*****************************************************************************************
Sub ShiftSevenToThree(myDay)
  Dim myWorkingNames As Range
  Dim I As Integer    'The number of agents required for the shift.

  'Store the range with the names of the agents selected to work.
  Set myWorkingNames = Range(Range("Schedule" & myDay).Cells(1).Address, _
              Range("Schedule" & myDay).End(xlDown).Address)

  'Get the number of agents required to work the 7-3 shift.
  I = Range("Shift" & myDay).Cells(1).Value

  'Find those whose first preference is for the 7-3 shift
  If I > 0 Then
    For myCell = 1 To myWorkingNames.Cells.Count
      Sheets("Schedules").Select
      If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
        And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
        myName = Range(myWorkingNames.Cells(myCell).Address).Value
        Sheets("ShiftPreferences").Select
        Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
        Selection.Find(What:=myName, After:=ActiveCell, _
                LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
                SearchDirection:=xlNext, MatchCase:=False).Activate
        If ActiveCell.Offset(, 1).Value = "1" Then
          Range("ShiftSelection").Cells(myCell).Value = myName
        End If
      End If
    Next myCell
  End If
```

```
'Randomly select as many as are needed from this list.
Sheets("Schedules").Select
Static myOldValue
If Range("ShiftSelection").Cells(25).Value = "" Then
   Range("ShiftSelection").Cells(25).Value = "1"
   Range("ShiftSelection").Cells(25).ClearContents
End If
If 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count >= I Then
   Do Until I = 0
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> ""
         myName = Int((25 * Rnd) + 1)
      Loop
      For C = 17 To 20
         Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      Range("ShiftSelection").Cells(myName).ClearContents
      I = I - 1
   Loop
ElseIf 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count > 0 _
   And 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count < I Then
   Do Until I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> "" _
            And Range("ShiftSelection").Cells(myName).Value <> "1"
         myName = Int((25 * Rnd) + 1)
      Loop
      Range("ShiftSelection").Cells(myName).Value = "1"
      For C = 17 To 20
         Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      I = I - 1
      If I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count Then
         Range("ShiftSelection").Cells(myName).ClearContents
         Exit Do
      End If
   Loop
End If
Range("ShiftSelection").ClearContents

'If additional agents are still required to work the 7-3 shift,
'find those second preference is for the 7-3 shift.
If I > 0 Then
   For myCell = 1 To myWorkingNames.Cells.Count
      Sheets("Schedules").Select
      If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
       And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
         myName = Range(myWorkingNames.Cells(myCell).Address).Value
         Sheets("ShiftPreferences").Select
         Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
         Selection.Find(What:=myName, After:=ActiveCell, _
                  LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
                  SearchDirection:=xlNext, MatchCase:=False).Activate
         If ActiveCell.Offset(, 1).Value = "2" Then
```

```
            Range("ShiftSelection").Cells(myCell).Value = myName
          End If
        End If
      Next myCell
    End If

    'Randomly select as many as are needed from this list.
    Sheets("Schedules").Select
    If 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count >= I Then
      Do Until I = 0
        myName = 0
        Do Until Range("ShiftSelection").Cells(myName).Value <> ""
          myName = Int((25 * Rnd) + 1)
        Loop
        For C = 17 To 20
          Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
        Next C
        Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
        Range("ShiftSelection").Cells(myName).ClearContents
        I = I - 1
      Loop
    ElseIf 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count > 0 _
     And 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count < I Then
      Do Until I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count
        myName = 0
        Do Until Range("ShiftSelection").Cells(myName).Value <> "" _
              And Range("ShiftSelection").Cells(myName).Value <> "1"
          myName = Int((25 * Rnd) + 1)
        Loop
        Range("ShiftSelection").Cells(myName).Value = 1
        For C = 17 To 20
          Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
        Next C
        Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
        I = I - 1
        If I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count Then
          Range("ShiftSelection").Cells(myName).ClearContents
          Exit Do
        End If
      Loop
    End If
    Range("ShiftSelection").ClearContents

    'If additional agents are still required to work the 7-3 shift,
    'find those whose third preference is for the 7-3 shift.
    If I > 0 Then
      For myCell = 1 To myWorkingNames.Cells.Count
        Sheets("Schedules").Select
        If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
         And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
          myName = Range(myWorkingNames.Cells(myCell).Address).Value
          Sheets("ShiftPreferences").Select
          Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
          Selection.Find(What:=myName, After:=ActiveCell, _
                    LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
                    SearchDirection:=xlNext, MatchCase:=False).Activate
```

```
        If ActiveCell.Offset(, 1).Value = "3" Then
          Range("ShiftSelection").Cells(myCell).Value = myName
        End If
      End If
    Next myCell
  End If

  'Randomly select as many as are needed from this list.
  Sheets("Schedules").Select
  If 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count >= I Then
    Do Until I = 0
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> ""
        myName = Int((25 * Rnd) + 1)
      Loop
      For C = 17 To 20
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      Range("ShiftSelection").Cells(myName).ClearContents
      I = I - 1
    Loop
  ElseIf 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count > 0 _
   And 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count < I Then
    Do Until I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> "" _
            And Range("ShiftSelection").Cells(myName).Value <> "1"
        myName = Int((25 * Rnd) + 1)
      Loop
      Range("ShiftSelection").Cells(myName).Value = "1"
      For C = 17 To 20
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      I = I - 1
      If I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count Then
        Range("ShiftSelection").Cells(myName).ClearContents
        Exit Do
      End If
    Loop
  End If
  Range("ShiftSelection").ClearContents
End Sub

'*****************************************************************************
'Procedure: ShiftEightToFour
'Arguments: myDay, the day currently being scheduled
'Purpose: Gets the number of required 8-4 shifts and assigns them to agents according to preferences.
'*****************************************************************************
Sub ShiftEightToFour(myDay)
  Dim myWorkingNames As Range
  Dim I As Integer
  Set myWorkingNames = Range(Range("Schedule" & myDay).Cells(1).Address, Range("Schedule" &
    myDay).End(xlDown).Address)
```

```
I = Range("Shift" & myDay).Cells(2).Value

If I > 0 Then
  For myCell = 1 To myWorkingNames.Cells.Count
    Sheets("Schedules").Select
    If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
     And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
      myName = Range(myWorkingNames.Cells(myCell).Address).Value
      Sheets("ShiftPreferences").Select
      Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
      Selection.Find(What:=myName, After:=ActiveCell, _
        LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
        SearchDirection:=xlNext, MatchCase:=False).Activate
      If ActiveCell.Offset(, 2).Value = "1" Then
        Range("ShiftSelection").Cells(myCell).Value = myName
      End If
    End If
  Next myCell
End If

Sheets("Schedules").Select
If 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count >= I Then
  Do Until I = 0
    myName = 0
    Do Until Range("ShiftSelection").Cells(myName).Value <> ""
      myName = Int((25 * Rnd) + 1)
    Loop
    For C = 1 To 2
      Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
    Next C
    For C = 19 To 20
      Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
    Next C
    Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
    Range("ShiftSelection").Cells(myName).ClearContents
    I = I - 1
  Loop
ElseIf 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count > 0 _
 And 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count < I Then
  Do Until I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count
    myName = 0
    Do Until Range("ShiftSelection").Cells(myName).Value <> "" _
        And Range("ShiftSelection").Cells(myName).Value <> "1"
      myName = Int((25 * Rnd) + 1)
    Loop
    Range("ShiftSelection").Cells(myName).Value = "1"
    For C = 1 To 2
      Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
    Next C
    For C = 19 To 20
      Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
    Next C
    Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
    I = I - 1
    If I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count Then
      Range("ShiftSelection").Cells(myName).ClearContents
```

```
        Exit Do
      End If
    Loop
  End If
  Range("ShiftSelection").ClearContents

  If I > 0 Then
    For myCell = 1 To myWorkingNames.Cells.Count
      Sheets("Schedules").Select
      If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
        And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
        myName = Range(myWorkingNames.Cells(myCell).Address).Value
        Sheets("ShiftPreferences").Select
        Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
        Selection.Find(What:=myName, After:=ActiveCell, _
                LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
                SearchDirection:=xlNext, MatchCase:=False).Activate
        If ActiveCell.Offset(, 2).Value = "2" Then
          Range("ShiftSelection").Cells(myCell).Value = myName
        End If
      End If
    Next myCell
  End If

  Sheets("Schedules").Select
  If 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count >= I Then
    Do Until I = 0
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> ""
        myName = Int((25 * Rnd) + 1)
      Loop
      For C = 1 To 2
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      For C = 19 To 20
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      Range("ShiftSelection").Cells(myName).ClearContents
      I = I - 1
    Loop
  ElseIf 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count > 0 _
   And 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count < I Then
    Do Until I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> "" _
            And Range("ShiftSelection").Cells(myName).Value <> "1"
        myName = Int((25 * Rnd) + 1)
      Loop
      Range("ShiftSelection").Cells(myName).Value = "1"
      For C = 1 To 2
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      For C = 19 To 20
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
```

```
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      I = I - 1
      If I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count Then
        Range("ShiftSelection").Cells(myName).ClearContents
        Exit Do
      End If
    Loop
End If
Range("ShiftSelection").ClearContents

If I > 0 Then
  For myCell = 1 To myWorkingNames.Cells.Count
    Sheets("Schedules").Select
    If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
     And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
      myName = Range(myWorkingNames.Cells(myCell).Address).Value
      Sheets("ShiftPreferences").Select
      Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
      Selection.Find(What:=myName, After:=ActiveCell, _
              LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
              SearchDirection:=xlNext, MatchCase:=False).Activate
      If ActiveCell.Offset(, 2).Value = "3" Then
        Range("ShiftSelection").Cells(myCell).Value = myName
      End If
    End If
  Next myCell
End If

Sheets("Schedules").Select
If 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count >= I Then
  Do Until I = 0
    myName = 0
    Do Until Range("ShiftSelection").Cells(myName).Value <> ""
      myName = Int((25 * Rnd) + 1)
    Loop
    For C = 1 To 2
      Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
    Next C
    For C = 19 To 20
      Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
    Next C
    Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
    Range("ShiftSelection").Cells(myName).ClearContents
    I = I - 1
  Loop
ElseIf 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count > 0 _
 And 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count < I Then
  Do Until I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count
    myName = 0
    Do Until Range("ShiftSelection").Cells(myName).Value <> "" _
         And Range("ShiftSelection").Cells(myName).Value <> "1"
      myName = Int((25 * Rnd) + 1)
    Loop
    Range("ShiftSelection").Cells(myName).Value = "1"
    For C = 1 To 2
      Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
```

```
        Next C
        For C = 19 To 20
          Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
        Next C
        Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
        I = I - 1
        If I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count Then
          Range("ShiftSelection").Cells(myName).ClearContents
          Exit Do
        End If
      Loop
    End If
    Range("ShiftSelection").ClearContents
End Sub

'*****************************************************************************
'Procedure: ShiftNineToFive
'Arguments: myDay, the day currently being scheduled
'Purpose: Gets the number of required 9-5 shifts and assigns them to agents according to preferences.
'*****************************************************************************
Sub ShiftNineToFive(myDay)
    Dim myWorkingNames As Range
    Dim I As Integer
    Set myWorkingNames = Range(Range("Schedule" & myDay).Cells(1).Address, Range("Schedule" &
    myDay).End(xlDown).Address)

    I = Range("Shift" & myDay).Cells(3).Value

    If I > 0 Then
      For myCell = 1 To myWorkingNames.Cells.Count
        Sheets("Schedules").Select
        If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
         And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
          myName = Range(myWorkingNames.Cells(myCell).Address).Value
          Sheets("ShiftPreferences").Select
          Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
          Selection.Find(What:=myName, After:=ActiveCell, _
                    LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
                    SearchDirection:=xlNext, MatchCase:=False).Activate
          If ActiveCell.Offset(, 3).Value = "1" Then
            Range("ShiftSelection").Cells(myCell).Value = myName
          End If
        End If
      Next myCell
    End If

    Sheets("Schedules").Select
    If 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count >= I Then
      Do Until I = 0
        myName = 0
        Do Until Range("ShiftSelection").Cells(myName).Value <> ""
          myName = Int((25 * Rnd) + 1)
        Loop
        For C = 1 To 4
          Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
        Next C
```

```
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      Range("ShiftSelection").Cells(myName).ClearContents
      I = I - 1
    Loop
ElseIf 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count > 0 _
  And 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count < I Then
    Do Until I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> "" _
            And Range("ShiftSelection").Cells(myName).Value <> "1"
        myName = Int((25 * Rnd) + 1)
      Loop
      Range("ShiftSelection").Cells(myName).Value = "1"
      For C = 1 To 4
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      I = I - 1
      If I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count Then
        Range("ShiftSelection").Cells(myName).ClearContents
        Exit Do
      End If
    Loop
End If
Range("ShiftSelection").ClearContents

If I > 0 Then
    For myCell = 1 To myWorkingNames.Cells.Count
      Sheets("Schedules").Select
      If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
        And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
        myName = Range(myWorkingNames.Cells(myCell).Address).Value
        Sheets("ShiftPreferences").Select
        Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
        Selection.Find(What:=myName, After:=ActiveCell, _
                LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
                SearchDirection:=xlNext, MatchCase:=False).Activate
        If ActiveCell.Offset(, 3).Value = "2" Then
          Range("ShiftSelection").Cells(myCell).Value = myName
        End If
      End If
    Next myCell
End If

Sheets("Schedules").Select
If 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count >= I Then
    Do Until I = 0
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> ""
        myName = Int((25 * Rnd) + 1)
      Loop
      For C = 1 To 4
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      Range("ShiftSelection").Cells(myName).ClearContents
```

```
      I = I - 1
    Loop
ElseIf 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count > 0 _
  And 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count < I Then
    Do Until I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> "" _
            And Range("ShiftSelection").Cells(myName).Value <> "1"
        myName = Int((25 * Rnd) + 1)
      Loop
      Range("ShiftSelection").Cells(myName).Value = "1"
      For C = 1 To 4
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      I = I - 1
      If I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count Then
        Range("ShiftSelection").Cells(myName).ClearContents
        Exit Do
      End If
    Loop
End If
Range("ShiftSelection").ClearContents

If I > 0 Then
    For myCell = 1 To myWorkingNames.Cells.Count
      Sheets("Schedules").Select
      If Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
       And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
        myName = Range(myWorkingNames.Cells(myCell).Address).Value
        Sheets("ShiftPreferences").Select
        Sheets("ShiftPreferences").Range("C7", Range("C7").End(xlDown).Address).Select
        Selection.Find(What:=myName, After:=ActiveCell, _
                LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
                SearchDirection:=xlNext, MatchCase:=False).Activate
        If ActiveCell.Offset(, 3).Value = "3" Then
          Range("ShiftSelection").Cells(myCell).Value = myName
        End If
      End If
    Next myCell
End If

Sheets("Schedules").Select
If 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count >= I Then
    Do Until I = 0
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> ""
        myName = Int((25 * Rnd) + 1)
      Loop
      For C = 1 To 4
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      Range("ShiftSelection").Cells(myName).ClearContents
      I = I - 1
    Loop
```

```
  ElseIf 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count > 0 _
   And 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count < I Then
    Do Until I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count
      myName = 0
      Do Until Range("ShiftSelection").Cells(myName).Value <> "" _
            And Range("ShiftSelection").Cells(myName).Value <> "1"
        myName = Int((25 * Rnd) + 1)
      Loop
      Range("ShiftSelection").Cells(myName).Value = "1"
      For C = 1 To 4
        Range(myWorkingNames.Cells(myName).Offset(, C + 1).Address).Value = "X"
      Next C
      Range(myWorkingNames.Cells(myName).Offset(, 26).Address).Value = "1"
      I = I - 1
      If I = 25 - Range("ShiftSelection").SpecialCells(xlCellTypeBlanks).Cells.Count Then
        Range("ShiftSelection").Cells(myName).ClearContents
        Exit Do
      End If
    Loop
  End If
  Range("ShiftSelection").ClearContents
End Sub
```

Lunch Selection Heuristic

```
'*****************************************************************************
'Procedure: SelectLunches
'Arguments: myDay, the day currently being scheduled
'Purpose: Assigns lunch times to agents.
'*****************************************************************************
Sub SelectLunches(myDay)
  SelectFirstPreferenceLunch (myDay)
  MinimizingPeopleAwayForLunch (myDay)
  MinimizingPeopleAwayForLunch (myDay)
  Range("AA1").EntireColumn.ClearContents
End Sub

'*****************************************************************************
'Procedure: SelectFirstPreferenceLunch
'Arguments: myDay, the day currently being scheduled
'Purpose: Assigns each agent's first preference for lunch.
'*****************************************************************************
Sub SelectFirstPreferenceLunch(myDay)
  Dim myWorkingNames As Range

  Set myWorkingNames = Range(Range("Schedule" & myDay).Cells(1).Address, Range("Schedule" &
    myDay).End(xlDown).Address)

  For myCell = 1 To myWorkingNames.Cells.Count
    Sheets("Schedules").Select
    If Range("LunchRestriction" & myDay).Cells(myCell).Value <> "L" _
      And Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" _
      And Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value <> "1" Then
      If Range(myWorkingNames.Cells(myCell).Offset(, 2).Address).Value = "" Then
```

```
  S = 1  'S = 1 means the agent is working the7-3 Shift
ElseIf Range(myWorkingNames.Cells(myCell).Offset(, 4).Address).Value = "" Then
  S = 2  'S = 2 means the agent is working the 8-4 Shift
Else
  S = 3  'S = 3 means the agent is working the 9-5 Shift
End If
myName = Range(myWorkingNames.Cells(myCell).Address).Value
Sheets("LunchPreferences").Select
Sheets("LunchPreferences").Range("C8", Range("C8").End(xlDown).Address).Select
Selection.Find(What:=myName, After:=ActiveCell, _
        LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows, _
        SearchDirection:=xlNext, MatchCase:=False).Activate
'Determining lunch preference for agents working the 7-3 shift
If S = 1 Then
  If ActiveCell.Offset(, 1).Value = "1" And ActiveCell.Offset(, 2) = "1" Then
    Sheets("Schedules").Select
    myLunch = Int((2 * Rnd) + 1)
    If myLunch = 1 Then
      For C = 10 To 11
        Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
      Next C
        Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
    Else
      For C = 11 To 12
        Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
      Next C
        Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
    End If
  ElseIf ActiveCell.Offset(, 1).Value = "1" And ActiveCell.Offset(, 2) <> "1" Then
    Sheets("Schedules").Select
    For C = 10 To 11
      Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
    Next C
      Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
  Else
    Sheets("Schedules").Select
    For C = 11 To 12
      Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
    Next C
      Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
  End If
'Determining lunch preference for agents working the 9-5 shift
ElseIf S = 3 Then
  If ActiveCell.Offset(, 6).Value = "1" And ActiveCell.Offset(, 7) = "1" Then
    Sheets("Schedules").Select
    myLunch = Int((2 * Rnd) + 1)
    If myLunch = 1 Then
      For C = 13 To 14
        Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
      Next C
        Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
    Else
      For C = 14 To 15
        Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
      Next C
        Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
```

```
      End If
    ElseIf ActiveCell.Offset(, 6).Value = "1" And ActiveCell.Offset(, 7) <> "1" Then
      Sheets("Schedules").Select
      For C = 13 To 14
        Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
      Next C
        Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
    Else
      Sheets("Schedules").Select
      For C = 14 To 15
        Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
      Next C
        Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
    End If
'Determining lunch preference for agents working the 8-4 shift
Else
    If ActiveCell.Offset(, 3).Value = "1" And ActiveCell.Offset(, 4) = "1" _
      And ActiveCell.Offset(, 5) = "1" Then
      Sheets("Schedules").Select
      myLunch = Int((3 * Rnd) + 1)
      If myLunch = 1 Then
        For C = 11 To 12
          Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
        Next C
          Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
      ElseIf myLunch = 2 Then
        For C = 12 To 13
          Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
        Next C
          Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
      Else
        For C = 13 To 14
          Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
        Next C
          Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
      End If
    ElseIf ActiveCell.Offset(, 3).Value = "1" And ActiveCell.Offset(, 4) = "1" _
      And ActiveCell.Offset(, 5) <> "1" Then
      Sheets("Schedules").Select
      myLunch = Int((2 * Rnd) + 1)
      If myLunch = 1 Then
        For C = 11 To 12
          Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
        Next C
          Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
      Else
        For C = 12 To 13
          Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
        Next C
          Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
      End If
    ElseIf ActiveCell.Offset(, 3).Value <> "1" And ActiveCell.Offset(, 4) = "1" _
      And ActiveCell.Offset(, 5) = "1" Then
      Sheets("Schedules").Select
      myLunch = Int((2 * Rnd) + 1)
      If myLunch = 1 Then
```

```
            For C = 12 To 13
              Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
            Next C
              Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
          Else
            For C = 13 To 14
              Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
            Next C
              Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
          End If
        ElseIf ActiveCell.Offset(, 3).Value = "1" And ActiveCell.Offset(, 4) <> "1" _
          And ActiveCell.Offset(, 5) = "1" Then
          Sheets("Schedules").Select
          myLunch = Int((2 * Rnd) + 1)
          If myLunch = 1 Then
            For C = 11 To 12
              Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
            Next C
              Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
          Else
            For C = 13 To 14
              Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
            Next C
              Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
          End If
        ElseIf ActiveCell.Offset(, 3).Value = "1" And ActiveCell.Offset(, 4) <> "1" _
          And ActiveCell.Offset(, 5) <> "1" Then
          Sheets("Schedules").Select
          For C = 11 To 12
            Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
          Next C
          Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
        ElseIf ActiveCell.Offset(, 3).Value <> "1" And ActiveCell.Offset(, 4) = "1" _
          And ActiveCell.Offset(, 5) <> "1" Then
          Sheets("Schedules").Select
          For C = 12 To 13
            Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
          Next C
            Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
        ElseIf ActiveCell.Offset(, 3).Value <> "1" And ActiveCell.Offset(, 4) <> "1" _
          And ActiveCell.Offset(, 5) = "1" Then
          Sheets("Schedules").Select
          For C = 13 To 14
            Range(myWorkingNames.Cells(myCell).Offset(, C).Address).Value = "L"
          Next C
            Range(myWorkingNames.Cells(myCell).Offset(, 26).Address).Value = "1"
        End If
      End If
    End If
  Next myCell
End Sub
```

```
'*****************************************************************************
'Procedure: MinimizingPeopleAwayForLunch
'Arguments: myDay, the day currently being scheduled
'Purpose: Moves assigned lunches if it will reduce the total number of agents away each period.
'*****************************************************************************
Sub MinimizingPeopleAwayForLunch(myDay)
  Dim myWorkingNames As Range

  Set myWorkingNames = Range(Range("Schedule" & myDay).Cells(1).Address, Range("Schedule" &
    myDay).End(xlDown).Address)
  Sheets("Schedules").Select

  For myCell = 1 To myWorkingNames.Cells.Count
    Sheets("Schedules").Select
    If Range("LunchRestriction" & myDay).Cells(myCell).Value <> "L" _
      And Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" Then
      If Range(myWorkingNames.Cells(myCell).Offset(, 2).Address).Value = "" Then
        S = 1  'S = 1 means the agent is working the 7-3 Shift
      ElseIf Range(myWorkingNames.Cells(myCell).Offset(, 4).Address).Value = "" Then
        S = 2  'S = 2 means the agent is working the 8-4 Shift
      Else
        S = 3  'S = 3 means the agent is working the 9-5 Shift
      End If
      If S = 1 Then
        A = 0
        B = 0
        C = 0
        For myCount = 1 To myWorkingNames.Cells.Count
          If myWorkingNames.Cells(myCount).Offset(, 10).Value = "L" Then
            A = A + 1
          End If
          If myWorkingNames.Cells(myCount).Offset(, 11).Value = "L" Then
            B = B + 1
          End If
          If myWorkingNames.Cells(myCount).Offset(, 12).Value = "L" Then
            C = C + 1
          End If
        Next myCount
        If myWorkingNames.Cells(myCell).Offset(, 10).Value = "L" Then
          If A + B > B + C + 1 Then
            Range(myWorkingNames.Cells(myCell).Offset(, 10).Address).Value = ""
            Range(myWorkingNames.Cells(myCell).Offset(, 12).Address).Value = "L"
          End If
        Else
          If B + C > A + B + 1 Then
            Range(myWorkingNames.Cells(myCell).Offset(, 10).Address).Value = "L"
            Range(myWorkingNames.Cells(myCell).Offset(, 12).Address).Value = ""
          End If
        End If
      End If
    End If
  Next myCell

  For myCell = 1 To myWorkingNames.Cells.Count
    Sheets("Schedules").Select
    If Range("LunchRestriction" & myDay).Cells(myCell).Value <> "L" _
```

```
And Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" Then
 If Range(myWorkingNames.Cells(myCell).Offset(, 2).Address).Value = "" Then
   S = 1   'S = 1 means the agent is working the7-3 Shift
 ElseIf Range(myWorkingNames.Cells(myCell).Offset(, 4).Address).Value = "" Then
   S = 2   'S = 2 means the agent is working the 8-4 Shift
 Else
   S = 3   'S = 3 means the agent is working the 9-5 Shift
 End If
 If S = 2 Then
   A = 0
   B = 0
   C = 0
   D = 0
   For myCount = 1 To myWorkingNames.Cells.Count
     If myWorkingNames.Cells(myCount).Offset(, 11).Value = "L" Then
       A = A + 1
     End If
     If myWorkingNames.Cells(myCount).Offset(, 12).Value = "L" Then
       B = B + 1
     End If
     If myWorkingNames.Cells(myCount).Offset(, 13).Value = "L" Then
       C = C + 1
     End If
     If myWorkingNames.Cells(myCount).Offset(, 14).Value = "L" Then
       D = D + 1
     End If
   Next myCount
   If myWorkingNames.Cells(myCell).Offset(, 11).Value = "L" Then
     If A + B > B + C + 1 And B + C + 1 <= C + D + 1 + 1 Then
       Range(myWorkingNames.Cells(myCell).Offset(, 11).Address).Value = ""
       Range(myWorkingNames.Cells(myCell).Offset(, 13).Address).Value = "L"
     ElseIf A + B > B + C + 1 And B + C + 1 >= C + D + 1 + 1 Then
       Range(myWorkingNames.Cells(myCell).Offset(, 11).Address).Value = ""
       Range(myWorkingNames.Cells(myCell).Offset(, 12).Address).Value = ""
       Range(myWorkingNames.Cells(myCell).Offset(, 13).Address).Value = "L"
       Range(myWorkingNames.Cells(myCell).Offset(, 14).Address).Value = "L"
     End If
   ElseIf myWorkingNames.Cells(myCell).Offset(, 12).Value = "L" _
     And myWorkingNames.Cells(myCell).Offset(, 13).Value = "L" Then
     If B + C > A + B + 1 And A + B + 1 <= C + D + 1 Then
       Range(myWorkingNames.Cells(myCell).Offset(, 11).Address).Value = "L"
       Range(myWorkingNames.Cells(myCell).Offset(, 13).Address).Value = ""
     ElseIf B + C > A + B + 1 And A + B + 1 >= C + D + 1 Then
       Range(myWorkingNames.Cells(myCell).Offset(, 12).Address).Value = ""
       Range(myWorkingNames.Cells(myCell).Offset(, 14).Address).Value = "L"
     End If
   Else
     If C + D > B + 1 + C And B + 1 + C <= A + 1 + B + 1 Then
       Range(myWorkingNames.Cells(myCell).Offset(, 12).Address).Value = "L"
       Range(myWorkingNames.Cells(myCell).Offset(, 14).Address).Value = ""
     ElseIf C + D > B + 1 + C And B + 1 + C >= A + 1 + B + 1 Then
       Range(myWorkingNames.Cells(myCell).Offset(, 11).Address).Value = "L"
       Range(myWorkingNames.Cells(myCell).Offset(, 12).Address).Value = "L"
       Range(myWorkingNames.Cells(myCell).Offset(, 13).Address).Value = ""
       Range(myWorkingNames.Cells(myCell).Offset(, 14).Address).Value = ""
     End If
```

```
            End If
          End If
        End If
      Next myCell

      For myCell = 1 To myWorkingNames.Cells.Count
        Sheets("Schedules").Select
        If Range("LunchRestriction" & myDay).Cells(myCell).Value <> "L" _
          And Range("ShiftRestriction" & myDay).Cells(myCell).Value <> "S" Then
          If Range(myWorkingNames.Cells(myCell).Offset(, 2).Address).Value = "" Then
            S = 1   'S = 1 means the agent is working the7-3 Shift
          ElseIf Range(myWorkingNames.Cells(myCell).Offset(, 4).Address).Value = "" Then
            S = 2   'S = 2 means the agent is working the 8-4 Shift
          Else
            S = 3   'S = 3 means the agent is working the 9-5 Shift
          End If
          If S = 3 Then
            A = 0
            B = 0
            C = 0
            For myCount = 1 To myWorkingNames.Cells.Count
              If myWorkingNames.Cells(myCount).Offset(, 13).Value = "L" Then
                A = A + 1
              End If
              If myWorkingNames.Cells(myCount).Offset(, 14).Value = "L" Then
                B = B + 1
              End If
              If myWorkingNames.Cells(myCount).Offset(, 15).Value = "L" Then
                C = C + 1
              End If
            Next myCount
            If myWorkingNames.Cells(myCell).Offset(, 13).Value = "L" Then
              If A + B > B + C + 1 Then
                Range(myWorkingNames.Cells(myCell).Offset(, 13).Address).Value = ""
                Range(myWorkingNames.Cells(myCell).Offset(, 15).Address).Value = "L"
              End If
            Else
              If B + C > A + B + 1 Then
                Range(myWorkingNames.Cells(myCell).Offset(, 13).Address).Value = "L"
                Range(myWorkingNames.Cells(myCell).Offset(, 15).Address).Value = ""
              End If
            End If
          End If
        End If
      Next myCell
End Sub
```

Scheduling Heuristic

```
'*****************************************************************************
'Procedure: Schedule
'Arguments: myDay, the day currently being scheduled
'Purpose: This is an overall procedure that will select each of the columns in succession and run each of
'the scheduling subroutines relevant for each period.
'*****************************************************************************
Sub Schedule(myDay)

  Sheets("Schedules").Select
  FillShiftColumns (myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myColumnNo As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion
  For myColumnNo = 1 To 20

    Set myColumn = myRange.Columns(myColumnNo)
    myColumn.Select

    FillRows (myDay)

    FillColumn (myDay)

    Priority1 (myDay)
      FillFinishedColumn
    Priority2 (myDay)
      FillFinishedColumn
    Priority3 (myDay)
      FillFinishedColumn
    Priority4 (myDay)
      FillFinishedColumn
    Priority5
      FillFinishedColumn
    Priority6 (myDay)
      FillFinishedColumn
    Priority7 (myDay)
      FillFinishedColumn

    ColumnSatisfiedCheck1 (myDay)
    ColumnSatisfiedCheck2 (myDay)
    ColumnSatisfiedCheck3 (myDay)
    ColumnSatisfiedCheck4 (myDay)
    ColumnSatisfiedCheck5 (myDay)
    ColumnSatisfiedCheck6 (myDay)

  Next myColumnNo
End Sub
```

```
'*****************************************************************************
'Procedure: FillShiftColumns
'Arguments: myDay, the day currently being scheduled
'Purpose: Fills blank cells in columns with X's with 1's if the number of' blank cells equals the number of
'        agents required that period.  Ensures that agents covering the end of the afternoon shift are
'        available.  This procedure is applied only once before all other scheduling procedures are applied.
'*****************************************************************************
Sub FillShiftColumns(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myBlanks As Range
  Dim myColumnNo As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  For myColumnNo = 1 To 4
    Set myColumn = myRange.Columns(myColumnNo)
    myColumn.Select
    On Error Resume Next
    Set myBlanks = myColumn.SpecialCells(xlCellTypeBlanks)
    If Err.Number = 0 Then
      If myBlanks.Cells.Count <= myColumn.Resize(1).Offset(28).Value Then
        myBlanks.Value = "1"
      End If
    End If
  Next myColumnNo
  For myColumnNo = 17 To 20
    Set myColumn = myRange.Columns(myColumnNo)
    myColumn.Select
    On Error Resume Next
    Set myBlanks = myColumn.SpecialCells(xlCellTypeBlanks)
    If Err.Number = 0 Then
      If myBlanks.Cells.Count <= myColumn.Resize(1).Offset(28).Value Then
        myBlanks.Value = "1"
      End If
    End If
  Next myColumnNo
End Sub

'*****************************************************************************
'Procedure: FillRows
'Arguments: myDay, the day currently being scheduled
'Purpose: This fills each row with ones if the number of available blocks left for that row equals the
'        number of blanks in the row.  Ensures that an agent in this situation is scheduled as first priority.
'        As well, this agent's remaining schedule is predetermined and no longer considered to save run
'        time. This is the first scheduling procedure applied to a column.
'*****************************************************************************
Sub FillRows(myDay)
  Dim myRange As Range
  Dim myRow As Range
  Dim myBlanks As Range
  Dim myRowNo As Integer
  Dim myRowNoEnd As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion
  Set myColumn = Selection
```

```
  For myRowNo = 2 To myColumn.Cells.Count
      Set myRow = myRange.Rows(myRowNo)
      On Error Resume Next
      Set myBlanks = myRow.SpecialCells(xlCellTypeBlanks)
      If Err.Number = 0 Then
        If myBlanks.Cells.Count <= myRow _
                    .Cells(myRow.Cells.Count) _
                    .Offset(0, 4) _
                    .Value Then
          myBlanks.Value = "1"
          For myCell = 1 To myRow.Cells.Count
            If myRow.Cells(myCell).Offset(29 - myRowNo).Value < 0 Then
              If myRow.Cells(myCell).Value = "1" Then
                myRow.Cells(myCell).Value = "0"
              End If
            End If
          Next myCell
        ElseIf myRow.Cells(myRow.Cells.Count).Offset(0, 4).Value = "0" Then
          myBlanks.Value = "0"
        End If
      End If
  Next myRowNo
End Sub

'*****************************************************************************************
'Procedure: FillColumn
'Arguments: myDay, the day currently being scheduled
'Purpose: This procedure checks to see if the number of blank cells in the column equals the number of
'        required agents. If the two numbers are equal and the agents corresponding to the blank cells are
'        available, the blank cells are filled with ones. If the agents corresponding to the blank cells are not
'        available, nothing is changed and the next scheduling procedure for the column begins.  This
'        procedures prevents all remaining scheduling procedures from running unneccessarily.  This is the
'        second scheduling procedure applied to the column and last procedure prior to the priority
'        procedures.
'*****************************************************************************************
Sub FillColumn(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myColumnBlanks As Range
  Dim myRowBlanks As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  On Error Resume Next
  myColumn.SpecialCells(xlCellTypeBlanks).Select
  Set myColumnBlanks = Selection
  If Err.Number = 0 Then
    If myColumnBlanks.Cells.Count = myColumn.Resize(1).Offset(28).Value Then
      For myCell = 2 To myColumn.Cells.Count
        If myColumn.Cells(myCell).Value = "" _
        And myRange.Rows(myCell) _
              .Cells(myRange.Rows(myCell).Cells.Count) _
              .Offset(0, 4) _
              .Value = "0" Then
```

```
            Exit Sub
          End If
        Next myCell
        For myCell = 2 To myColumn.Cells.Count
          If myColumn.Cells(myCell).Value = "" _
          And myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
                .Value > "1" Then
            myColumn.Cells(myCell).Value = "1"
          End If
          If myColumn.Cells(myCell).Value = "" _
          And myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
                .Value = "1" Then
            myColumn.Cells(myCell).Value = "1"
            On Error Resume Next
            myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
            If Err.Number = 0 Then
              myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Value = "0"
            End If
            'Set myRowBlanks = myRange _
                        .Rows(myCell) _
                        .SpecialCells (xlCellTypeBlanks)
            'myRowBlanks.Value = "0"
          End If
        Next myCell
      End If
    End If
    myColumn.Select
End Sub

'The following priority routines are applied to the column in order
' and consists of the bulk of the scheduling.

'*****************************************************************************************
'Procedure: Priority1
'Arguments: myDay, the day currently being scheduled
'Purpose: This goes to blank cells in the column from the top and continues with  the procedure (i.e.
'        deciding whether the blank cell will be scheduled) only if the cell has priority 1.  Priority 1 means
'        the two cells prior are not a 1 and then a 1 and there is only one available block left for the agent.
'        Minimizes single scheduled blocks by scheduling available agents two blocks in a row and also
'        by scheduling available agents with only 1 block left in priority.
'*****************************************************************************************
Sub Priority1(myDay)
Dim myRange As Range
  Dim myColumn As Range
  Dim myRowBlanks As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  On Error Resume Next
  myColumn.SpecialCells(xlCellTypeBlanks).Select
```

```
  If Err.Number = 0 Then
    For myCell = 2 To myColumn.Cells.Count
      If myColumn.Cells(myCell).Value = "" _
      And myColumn.Resize(1).Offset(28).Value > "0" _
      And myColumn.Cells(myCell).Offset(0, -2).Value <> "1" _
      And myColumn.Cells(myCell).Offset(0, -1).Value = "1" _
      And myRange.Rows(myCell) _
            .Cells(myRange.Rows(myCell).Cells.Count) _
            .Offset(0, 4) _
            .Value = "1" Then
        myColumn.Cells(myCell).Value = "1"
        On Error Resume Next
        myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
        If Err.Number = 0 Then
          myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Value = "0"
        End If
        'Set myRowBlanks = myRange _
                    .Rows(myCell) _
                    .SpecialCells(xlCellTypeBlanks)
        'myRowBlanks.Value = "0"
        'myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
        'Set myRowBlanks = Selection
        'myRowBlanks.Value = "0"
      End If
    Next myCell
  End If
  myColumn.Select
End Sub

'*****************************************************************************************
'Procedure: FillFinishedColumn
'Arguments: myDay, the day currently being scheduled
'Purpose: This procedure is to be run after every priority procedure to fill blank cells with zeros if no more
'         agents are required that shift.
'*****************************************************************************************
Sub FillFinishedColumn()
  Dim myColumn As Range

  Set myColumn = Selection
  On Error Resume Next
  myColumn.SpecialCells(xlCellTypeBlanks).Select
  If Err.Number = 0 Then
    If myColumn.Resize(1).Offset(28).Value < "1" Then
      myColumn.SpecialCells(xlCellTypeBlanks).Value = "0"
    End If
  End If
  myColumn.Select
End Sub
```

```
'*****************************************************************************
'Procedure: Priority2
'Arguments: myDay, the day currently being scheduled
'Purpose: This goes to blank cells in the column from the top and continues with the procedure (i.e.
'        deciding whether the blank cell will be scheduled) only if the cell has priority 2.  Priority 2 means
'        the two cells prior are not a 1 and then a 1 and there is more than one available block left for the
'        agent.  Minimizes single scheduled blocks by scheduling agents two blocks in a row.
'*****************************************************************************
Sub Priority2(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  On Error Resume Next
  myColumn.SpecialCells(xlCellTypeBlanks).Select
  If Err.Number = 0 Then
    For myCell = 2 To myColumn.Cells.Count
      If myColumn.Cells(myCell).Value = "" _
      And myColumn.Resize(1).Offset(28).Value > "0" _
      And myColumn.Cells(myCell).Offset(0, -2).Value <> "1" _
      And myColumn.Cells(myCell).Offset(0, -1).Value = "1" _
      And myRange.Rows(myCell) _
            .Cells(myRange.Rows(myCell).Cells.Count) _
            .Offset(0, 4) _
            .Value > "1" Then
        myColumn.Cells(myCell).FormulaR1C1 = "1"
      End If
    Next myCell
  End If
  myColumn.Select
End Sub

'*****************************************************************************
'Procedure: Priority3
'Arguments: myDay, the day currently being scheduled
'Purpose: This goes to blank cells in the column from the top and continues with the procedure (i.e.
'        deciding whether the blank cell will be scheduled) only if the cell has priority 3.  Priority 3 means
'        the two cells prior to the blank cell are both 1's and there is only one available block left for the
'        agent.  Facilitates scheduling an agents available blocks in sequence and minimizes the number of
'        scheduled periods separated by breaks.  This procedure gives priority to agents with only 1
'        available block left.
'*****************************************************************************
Sub Priority3(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myRowBlanks As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  On Error Resume Next
  myColumn.SpecialCells(xlCellTypeBlanks).Select
```

```
  If Err.Number = 0 Then
    For myCell = 2 To myColumn.Cells.Count
      If myColumn.Cells(myCell).Value = "" _
      And myColumn.Resize(1).Offset(28).Value > "0" _
      And myColumn.Cells(myCell).Offset(0, -2).Value = "1" _
      And myColumn.Cells(myCell).Offset(0, -1).Value = "1" _
      And myRange.Rows(myCell) _
            .Cells(myRange.Rows(myCell).Cells.Count) _
            .Offset(0, 4) _
            .Value = "1" Then
        myColumn.Cells(myCell).Value = "1"
        On Error Resume Next
        myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
        If Err.Number = 0 Then
          myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Value = "0"
        End If
        'Set myRowBlanks = myRange _
                    .Rows(myCell) _
                    .SpecialCells(xlCellTypeBlanks)
        'myRowBlanks.Value = "0"
        'myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
        'Set myRowBlanks = Selection
        'myRowBlanks.Value = "0"
      End If
    Next myCell
  End If
  myColumn.Select
End Sub

'*****************************************************************************************
'Procedure: Priority4
'Arguments: myDay, the day currently being scheduled
'Purpose: This goes to blnk cells in the column from the top and continues with the procedure (i.e.
'        deciding whether the blank cell will be scheduled) only if the cell has priority 4.  Priority 4 means
'        the two cells prior to the blank cell are both 1's and there is more than one available block left for
'        the agent.  Facilitates scheduling an agents available blocks in sequence and minimizes the
'        number of scheduled periods separated by breaks.  Also saves the remaining available work
'        blocks for agents who started working at 7 am, have already worked an hour and only have 4 or
'        less half-hour blocks left until the lunch period (11am) to ensure that enough agents will be
'        available to cover the phones when the late shift agents go for lunch.
'*****************************************************************************************
Sub Priority4(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  On Error Resume Next
  myColumn.SpecialCells(xlCellTypeBlanks).Select
  If Err.Number = 0 Then
    For myCell = 2 To myColumn.Cells.Count
      If myColumn.Cells(myCell).Value = "" _
      And myColumn.Resize(1).Offset(28).Value > "0" _
      And myColumn.Cells(myCell).Offset(0, -2).Value = "1" _
```

```
        And myColumn.Cells(myCell).Offset(0, -1).Value = "1" _
        And myRange.Rows(myCell) _
              .Cells(myRange.Rows(myCell).Cells.Count) _
              .Offset(0, 4) _
              .Value > "1" Then
          If myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
                .Value = "3" _
          And myColumn.Cells(myCell).Offset(0, 1).Value = "" _
          And myColumn.Cells(myCell).Offset(0, 2).Value = "L" Then
            myColumn.Cells(myCell).Resize(1, 2).Value = "0"
          ElseIf myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
                .Value = "2" _
          And myColumn.Cells(myCell).Offset(0, 1).Value = "" _
          And myColumn.Cells(myCell).Offset(0, 2).Value = "" _
          And myColumn.Cells(myCell).Offset(0, 3).Value = "L" Then
            myColumn.Cells(myCell).Resize(1, 3).Value = "0"
          On Error Resume Next
          ElseIf myRange.Rows(myCell) _
                  .Cells(myRange.Rows(myCell).Cells.Count) _
                  .Offset(0, 4) _
                  .Value < "5" _
          And myRange.Rows(myCell).Cells(1).Value = "1" _
          Or myRange.Rows(myCell).Cells(myRange.Rows(myCell).Cells.Count).Offset(0, 4).Value <
            "5" _
          And myRange.Rows(myCell).Cells(3).Value = "1" Then
            myRange.Rows(myCell) _
                .Resize(1, 8) _
                .SpecialCells(xlCellTypeBlanks) _
                .Select
            If Err.Number = 0 Then
              If myRange.Rows(myCell) _
                    .Resize(1, 8) _
                    .SpecialCells(xlCellTypeBlanks) _
                    .Count > "0" Then
                myRange.Rows(myCell) _
                    .Resize(1, 8) _
                    .SpecialCells(xlCellTypeBlanks) _
                    .Value = "0"
                If myColumn.Cells(myCell).Value = "" Then
                  myColumn.Cells(myCell).Value = "1"
                End If
              End If
            End If
          Else
            myColumn.Cells(myCell).Value = "1"
          End If
        End If
      Next myCell
    End If
    myColumn.Select
End Sub
```

```
'*****************************************************************************
'Procedure: Priority5
'Arguments: myDay, the day currently being scheduled
'Purpose: This goes to blank cells in the column from the top and continues with the procedure only if the
'         cell has priority 5.  Priority 5 means the two cells prior are not a 0 and then a 0  Tries to ensure
'         that an agent's breaks or unscheduled periods are at least 2 blocks long (1 hour).
'*****************************************************************************
Sub Priority5()
Dim myColumn As Range
  Dim myCell As Integer

  Set myColumn = Selection
  On Error Resume Next
  myColumn.SpecialCells(xlCellTypeBlanks).Select
  If Err.Number = 0 Then
    For myCell = 2 To myColumn.Cells.Count
      If myColumn.Cells(myCell).Value = "" _
      And myColumn.Cells(myCell).Offset(0, -2).Value <> "0" _
      And myColumn.Cells(myCell).Offset(0, -1).Value = "0" Then
        myColumn.Cells(myCell).Value = "0"
      End If
    Next myCell
  End If
  myColumn.Select
End Sub

'*****************************************************************************
'Procedure: Priority6
'Arguments: myDay, the day currently being scheduled
'Purpose: This goes to blank cells in the column from the top and continues with the procedure only if the
'         cell has priority 6.  Priority 6 means the two cells prior to the blank cell are both 1's.  Facilitates
'         scheduling an agents available blocks in sequence and minimizes the number of scheduled periods
'         separated by breaks.
'*****************************************************************************
Sub Priority6(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myRowBlanks As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  On Error Resume Next
  myColumn.SpecialCells(xlCellTypeBlanks).Select
  If Err.Number = 0 Then
    For myCell = 2 To myColumn.Cells.Count
      If myColumn.Cells(myCell).Value = "" _
      And myColumn.Resize(1).Offset(28).Value > "0" _
      And myColumn.Cells(myCell).Offset(0, -2).Value = "0" _
      And myColumn.Cells(myCell).Offset(0, -1).Value = "0" _
      And myRange.Rows(myCell) _
            .Cells(myRange.Rows(myCell).Cells.Count) _
            .Offset(0, 4) _
            .Value > "1" Then
        If myRange.Rows(myCell) _
```

```
            .Cells(myRange.Rows(myCell).Cells.Count) _
            .Offset(0, 4) _
            .Value = "3" _
        And myColumn.Cells(myCell).Offset(0, 1).Value = "" _
        And myColumn.Cells(myCell).Offset(0, 2).Value = "L" Then
          myColumn.Cells(myCell).Resize(1, 2).Value = "0"
        ElseIf myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
                .Value = "2" _
        And myColumn.Cells(myCell).Offset(0, 1).Value = "" _
        And myColumn.Cells(myCell).Offset(0, 2).Value = "" _
        And myColumn.Cells(myCell).Offset(0, 3).Value = "L" Then
          myColumn.Cells(myCell).Resize(1, 3).Value = "0"
        Else
          myColumn.Cells(myCell).Value = "1"
        End If
        If myColumn.Cells(myCell).Value = "" _
        And myColumn.Resize(1).Offset(28).Value > "0" _
        And myColumn.Cells(myCell).Offset(0, -2).Value = "0" _
        And myColumn.Cells(myCell).Offset(0, -1).Value = "0" _
        And myRange.Rows(myCell) _
              .Cells(myRange.Rows(myCell).Cells.Count) _
              .Offset(0, 4) _
              .Value = "1" Then
          myColumn.Cells(myCell).Value = "1"
          On Error Resume Next
          myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
          If Err.Number = 0 Then
          myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Value = "0"
          End If
          'myRange.Rows(myCell).SpecialCells (xlCellTypeBlanks)
          'Set myRowBlanks = Selection
          'myRowBlanks.Value = "0"
        End If
      End If
    Next myCell
  End If
  myColumn.Select
End Sub

'*****************************************************************************
'Procedure: Priority7
'Arguments: myDay, the day currently being scheduled
'Purpose: This goes to blank cells in the column from the top and continues with the procedure only if the
'         cell has priority 7.  Priority 7 means that the period needs another agent and the agent
'          corresponding to the blank cell is avaiable.  Priority 7 goes to all blank cells indiscriminate of the
'         prior to cells after all previous priority procedures have been applied.
'*****************************************************************************
Sub Priority7(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myRowBlanks As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion
```

```
  Set myColumn = Selection
  On Error Resume Next
  myColumn.SpecialCells(xlCellTypeBlanks).Select
  If Err.Number = 0 Then
    For myCell = 2 To myColumn.Cells.Count
      If myColumn.Cells(myCell).Value = "" _
      And myColumn.Resize(1).Offset(28).Value > "0" Then
        If myRange.Rows(myCell) _
              .Cells(myRange.Rows(myCell).Cells.Count) _
              .Offset(0, 4) _
              .Value = "1" Then
          myColumn.Cells(myCell).Value = "1"
          On Error Resume Next
          myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
          If Err.Number = 0 Then
            myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Value = "0"
          End If
          'Set myRowBlanks = myRange _
                    .Rows(myCell) _
                    .SpecialCells (xlCellTypeBlanks)
          'myRowBlanks.Value = "0"
        ElseIf myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
                .Value > "1" Then
          myColumn.Cells(myCell).Value = "1"
        End If
      End If
    Next myCell
  End If
  myColumn.Select
End Sub

'After all priority procedures have been applied and the column is replete
' of blank cells the ColumnSatisfiedCheck procedures are applied if agents
' are still required to cover the period and there are agents currently
' scheduled off that can be scheduled to work.

'*****************************************************************************************
'Procedure: ColumnSatisfiedCheck1
'Arguments: myDay, the day currently being scheduled
'Purpose: If agents are needed still and an agent currently scheduled off in the column is available, this
'         schedule procedes only if the agent was in the prior block and has only 1 available block left.
'*****************************************************************************************
Sub ColumnSatisfiedCheck1(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myRowBlanks As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  If myColumn.Resize(1).Offset(28).Value > "0" Then
  On Error Resume Next
```

```
  Selection.Find(What:="0", _
            After:=ActiveCell, _
            LookIn:=xlFormulas, _
            LookAt:=xlWhole, _
            SearchOrder:=xlByRows, _
            SearchDirection:=xlNext, _
            MatchCase:=False).Activate
    If Err.Number = 0 Then
      For myCell = 2 To myColumn.Cells.Count
        myColumn.Cells(myCell).Select
        If myColumn.Cells(myCell).Value = "0" _
        And myColumn.Resize(1).Offset(28).Value > "0" _
         And myColumn.Cells(myCell).Offset(0, -1).Value = "1" _
         And myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
                .Value = "1" Then
          myColumn.Cells(myCell).Value = "1"
          On Error Resume Next
          myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
          If Err.Number = 0 Then
            myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Value = "0"
          End If
          'myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Selection
          'Set myRowBlanks = Selection
          'myRowBlanks.Value = "0"
        End If
      Next myCell
    End If
  End If
  myColumn.Select
End Sub

'*****************************************************************************************
'Procedure: ColumnSatisfiedCheck2
'Arguments: myDay, the day currently being scheduled
'Purpose: This is the next priority column satisfied check.  If agents are needed still and an agent currently
'         scheduled off in the column is available, this schedule procedes only if the agent was' in the prior
'         block and has more than 1 available block left.
'*****************************************************************************************
Sub ColumnSatisfiedCheck2(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myRowBlanks As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  If myColumn.Resize(1).Offset(28).Value > "0" Then
  On Error Resume Next
  Selection.Find(What:="0", _
            After:=ActiveCell, _
            LookIn:=xlFormulas, _
            LookAt:=xlWhole, _
            SearchOrder:=xlByRows, _
```

```
            SearchDirection:=xlNext, _
            MatchCase:=False).Activate
      If Err.Number = 0 Then
        For myCell = 2 To myColumn.Cells.Count
          myColumn.Cells(myCell).Select
          If myColumn.Cells(myCell).Value = "0" _
          And myColumn.Resize(1).Offset(28).Value > "0" _
          And myColumn.Cells(myCell).Offset(0, -1).Value = "1" _
          And myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
                .Value > "1" Then
            myColumn.Cells(myCell).Value = "1"
          End If
        Next myCell
      End If
    End If
    myColumn.Select
End Sub

'*****************************************************************************************
'Procedure: ColumnSatisfiedCheck3
'Arguments: myDay, the day currently being scheduled
'Purpose: If agents are needed still and an agent currently scheduled off in the column is available, this
'         schedule procedes only if the agent was on lunch the prior period or will be on lunch the next
'         period and has only 1 available block left.
'*****************************************************************************************
Sub ColumnSatisfiedCheck3(myDay)
    Dim myRange As Range
    Dim myColumn As Range
    Dim myRowBlanks As Range
    Dim myCell As Integer

    Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

    Set myColumn = Selection
    If myColumn.Resize(1).Offset(28).Value > "0" Then
    On Error Resume Next
    Selection.Find(What:="0", _
            After:=ActiveCell, _
            LookIn:=xlFormulas, _
            LookAt:=xlWhole, _
            SearchOrder:=xlByRows, _
            SearchDirection:=xlNext, _
            MatchCase:=False).Activate
      If Err.Number = 0 Then
        For myCell = 2 To myColumn.Cells.Count
          If myColumn.Cells(myCell).Value = "0" _
          And myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
                .Value = "1" Then
            If myColumn.Cells(myCell).Offset(0, -1).Value = "L" Then
              myColumn.Cells(myCell).Value = "1"
            On Error Resume Next
            myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
```

```
            If Err.Number = 0 Then
              myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Value = "0"
            End If
              'myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
              'Set myRowBlanks = Selection
              'myRowBlanks.Value = "0"
            ElseIf myColumn.Cells(myCell).Offset(0, 1).Value = "L" Then
              myColumn.Cells(myCell).Value = "1"
            On Error Resume Next
            myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
            If Err.Number = 0 Then
              myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Value = "0"
            End If
              'Set myRowBlanks = myRange _
                        .Rows(myCell) _
                        .SpecialCells(xlCellTypeBlanks)
              'myRowBlanks.Value = "0"
            End If
          End If
        Next myCell
      End If
    End If
    myColumn.Select
End Sub

'*****************************************************************************************
'Procedure: ColumnSatisfiedCheck4
'Arguments: myDay, the day currently being scheduled
'Purpose: If agents are needed still and an agent currently scheduled off in the column is available, this
'         schedule procedes only if the agent was on lunch the prior period or will be on lunch the next
'         period and has more than one available block left.
'*****************************************************************************************
Sub ColumnSatisfiedCheck4(myDay)
    Dim myRange As Range
    Dim myColumn As Range
    Dim myRowBlanks As Range
    Dim myCell As Integer

    Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

    Set myColumn = Selection
    If myColumn.Resize(1).Offset(28).Value > "0" Then
    On Error Resume Next
    Selection.Find(What:="0", _
            After:=ActiveCell, _
            LookIn:=xlFormulas, _
            LookAt:=xlWhole, _
            SearchOrder:=xlByRows, _
            SearchDirection:=xlNext, _
            MatchCase:=False).Activate
      If Err.Number = 0 Then
        For myCell = 2 To myColumn.Cells.Count
          If myColumn.Cells(myCell).Value = "0" _
          And myRange.Rows(myCell) _
                .Cells(myRange.Rows(myCell).Cells.Count) _
                .Offset(0, 4) _
```

```
            .Value > "1" Then
          If myColumn.Cells(myCell).Offset(0, -1).Value = "L" Then
            myColumn.Cells(myCell).Value = "1"
          ElseIf myColumn.Cells(myCell).Offset(0, 1).Value = "L" Then
            myColumn.Cells(myCell).Value = "1"
          End If
        End If
      Next myCell
    End If
  End If
  myColumn.Select
End Sub

'*****************************************************************************************
'Procedure: ColumnSatisfiedCheck5
'Arguments: myDay, the day currently being scheduled
'Purpose: If agents are needed still and an agent currently scheduled off in the column is available, this
'         schedule procedes if the agent has only 1 available block left.
'*****************************************************************************************
Sub ColumnSatisfiedCheck5(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myRowBlanks As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  If myColumn.Resize(1).Offset(28).Value > "0" Then
  On Error Resume Next
  Selection.Find(What:="0", _
          After:=ActiveCell, _
          LookIn:=xlFormulas, _
          LookAt:=xlWhole, _
          SearchOrder:=xlByRows, _
          SearchDirection:=xlNext, _
          MatchCase:=False).Activate
    If Err.Number = 0 Then
      For myCell = 2 To myColumn.Cells.Count
        If myColumn.Cells(myCell).Value = "0" _
        And myColumn.Resize(1).Offset(28).Value > "0" _
        And myRange.Rows(myCell) _
              .Cells(myRange.Rows(myCell).Cells.Count) _
              .Offset(0, 4) _
              .Value = "1" Then
          myColumn.Cells(myCell).Value = "1"
          On Error Resume Next
          myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Select
          If Err.Number = 0 Then
            myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Value = "0"
          End If
          'myRange.Rows(myCell).SpecialCells(xlCellTypeBlanks).Selection
          'Set myRowBlanks = Selection
          'myRowBlanks.Value = "0"
        End If
      Next myCell
```

```
    End If
  End If
  myColumn.Select
End Sub

'*****************************************************************************************
'Procedure: ColumnSatisfiedCheck6
'Arguments: myDay, the day currently being scheduled
'Purpose: If agents are needed still and an agent currently scheduled off in the column is available, this
'         schedule procedes if the agent has more than 1 available block left.
'*****************************************************************************************
Sub ColumnSatisfiedCheck6(myDay)
  Dim myRange As Range
  Dim myColumn As Range
  Dim myRowBlanks As Range
  Dim myCell As Integer

  Set myRange = Range("Schedule" & myDay).Resize(1).Offset(, 2).CurrentRegion

  Set myColumn = Selection
  If myColumn.Resize(1).Offset(28).Value > "0" Then
  On Error Resume Next
  Selection.Find(What:="0", _
          After:=ActiveCell, _
          LookIn:=xlFormulas, _
          LookAt:=xlWhole, _
          SearchOrder:=xlByRows, _
          SearchDirection:=xlNext, _
          MatchCase:=False).Activate
    If Err.Number = 0 Then
      For myCell = 2 To myColumn.Cells.Count
        If myColumn.Cells(myCell).Value = "0" _
        And myColumn.Resize(1).Offset(28).Value > "0" _
        And myRange.Rows(myCell) _
              .Cells(myRange.Rows(myCell).Cells.Count) _
              .Offset(0, 4) _
              .Value > "1" Then
          myColumn.Cells(myCell).Value = "1"
        End If
      Next myCell
    End If
  End If
  myColumn.Select
End Sub
```