Chapter 4
Model Creation

After realizing the existing systems, described in the previous chapter, the next step is to create the simulation model by using Arena 3.0 as a tool. Other tools that used for establishing the model are: SPSS 10.0, Microsoft Access 2000, and Microsoft Excel 2000.

4.1 Model Formulation

This step is to develop a preliminary model by defining the components, variables, and interactions that constitute the system. The components of this model consist of 3 resources: telephone lines, IVR ports, and number of agent, interarrival time, service time of IVR and agent, the probability of the customers who require contacting only IVR and that of customers who require contacting with the agent, and the approximate number of executive customers that contact agent directly.

However, the service times of IVR and agent are not classified by service type because of the limitation of data. In case of IVR, the database doesn't provide the service times of each transaction but it only provides the time of start service and the time of finish service. Moreover, most customers demand the various service combinations, thus it would have been inappropriate to treat the service times for each service type separately as independent variables because of the dependent between some service demands from individual customer (Jerry Banks, John S. Carson, II and Barry L. Nelson, 1996. pp. 357). Thus, the service times of IVR are grouped as one variable.

In case of agent part, the agents have to make one wrap up heading per one customer even the customer may be served more than one service so the service times of agent also are grouped as one variable.

The decisions of this model are based on three benchmarks: queue time before entering to the IVR, and queue time before entering to the agent, and the proportion of successive line and busy line. Thus, these data have to be hold for making the decisions about the appropriate resources.

Moreover, the data of simulation service times (IVR service time, transfer time, and agent service time also have to be hold for comparing with the actual data whether mean of service time generated by simulation model close to the actual data (not significantly differ by statistic).

The preliminary of this model is shown below:

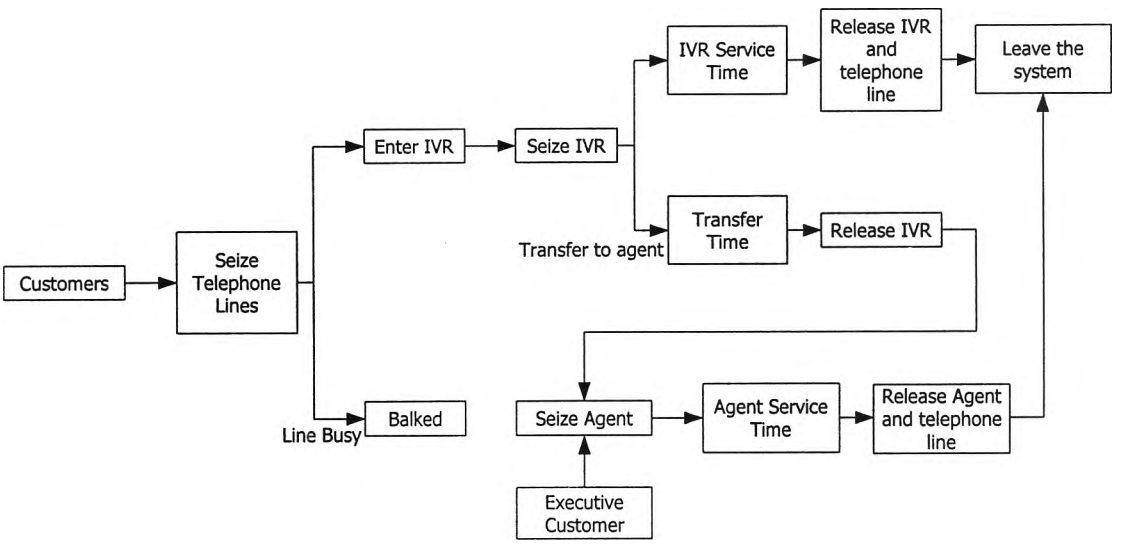


Figure 4-1: Preliminary Call Center Model

4.2 Experimental Design

To know the appropriate fixed resources, telephone lines and IVR ports, the maximum service time mean of IVR service, transfer time, and agent service are used as inputs for the model but these service times are chosen on peak hours workdays (9:00 Am to 18:00 AM) because even some service time mean of holiday may higher than that of workdays but the numbers of customer in holiday is significantly less than that in workdays.

However, the assumptions that have to applied in this model are:

- The distribution of interarrival time of customer is exponential because it is very hard to track the actual time that customer coming to the system (if the telephone line is full, the customers will be balked) and according to Fred F. Easton (2000), most of North America's 70,000 call centers use Erlang queueing models to determine appropriate staffing levels for each period their systems operate. These standard queueing models (M/M/C or M/M/C/N) assume a one-stage single-queue service, FCFS priority, multiple identical servers, and exponential inter-arrival times and service times.
- When starting the simulation, there is no any customer in the system.
- The distribution of interarrival time of executive customer that contact to agent directly is exponential because the database doesn't classified type of customer.
- The job of agent is only to serve the customer that call in system

The length of simulation is equal to 9 hours (9:00 AM to 5:00 APM) because it's the duration that has highest incoming calls. The number of calls classified by duration is conclude as follows:

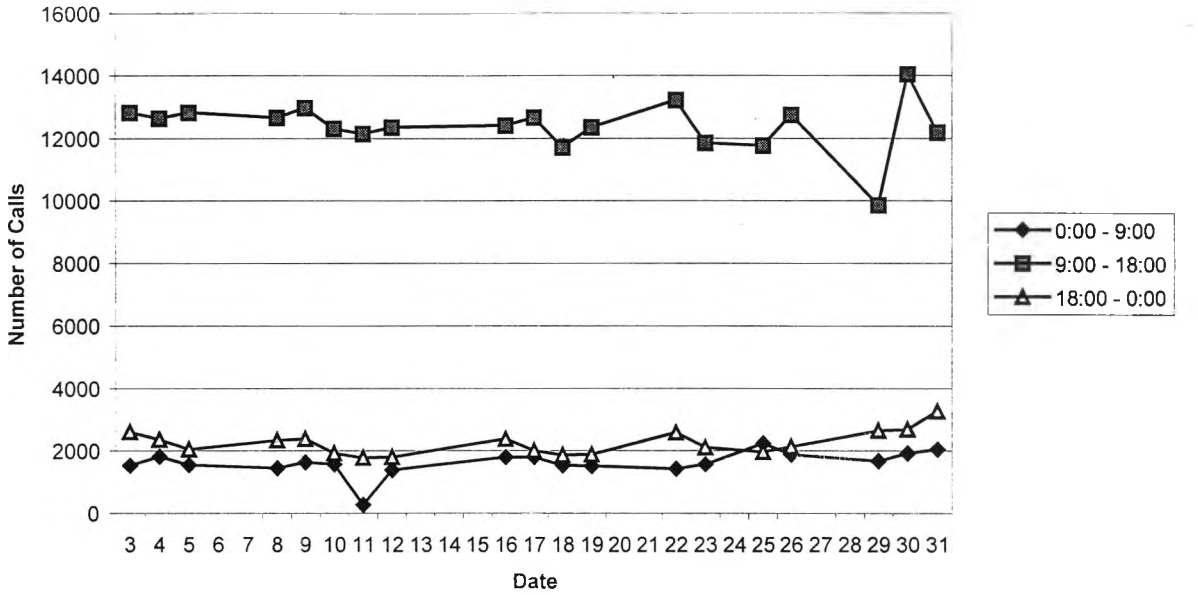


Figure 4-2: Numbers of call in each period.

To find the service time distribution easily and reduce the variation, the data of service time will be classified by hour. Thus, this model need 9 loops (3600 seconds for each loops) to complete the simulation.

4.3 Data Gathering

Most data of this call center system are recorded in the bank database, Oracle database. To take these data, we need to transform the data in the table of Oracle database to ASCII files, thus we can transform these ASCII file to the table of Microsoft Access. The chosen data are picked from January 1, 2001 to January 31, 2001. The ASCII files are contained in file "CALL_HISTORY.txt" and "CALL_HISTORY_LOG_0101.txt"

The data, required for this call center system, are: transaction date, transaction start time, transaction finish time, number of customer that connect to the agent, and agent service time. The required data are saved in file "Call Center.mdb" in Microsoft Access 2000 format.

Transaction date, transaction start time, transaction finish time, number of customer that connect to the agent are stored in table "IVR_Time" while agent service time is stored in table "AgentCALLHISTORY".

The detail of IVR_Time table is shown below:

Field Name	Data Type
CallStartDate	Date/Time
CallStartTime	Date/Time
CallFinishTime	Date/Time
TransferAgent	Text (only contain "y/n" string)

Figure 4-3: IVR_Time table detail

The detail of AgentCALLHISTORY table is illustrated below:

Field Name	Data Type
AgentTransDate	Date/Time
AgentTransTime	Date/Time
WrapUp	Text
AgentServiceTime	Number

Figure 4-4: AgentCALLHISTORY table detail

Approximately, the probability that the customer will be served IVR is 0.9 while the rest of customer require contacting with the agent. After interview with PABX officers, there are approximately 8,000 times of calls that try to enter the call center system in the workday's peak hour. The executive customers per hour (peak hour, workday) are approximately equal to 50. The executive customers have higher priority than the customers that enter the call system by phone. The time that agent use for making the wrap up heading is approximately equal to 45 seconds.

The expected queue time before enter to the IVR is equal to 10 seconds (average) while to the agent is also equal to 10 seconds (average). The proportion between the calls that can enter to the system and the number of balked calls is equal to 0.85.

4.4 Model Translation

After we have already been gotten the required data, the next step is to create the model by using simulation-programming package, Arena version 3.0 in this case.

The data modules we need to place for this model are shown below:

Figure 4-5: The data modules

We have placed 3 Resource modules. First modules is used to define the telephone lines, equal to 70 lines in the existing system. However, for further analysis, we can change the telephones line to realize the effect. The detail of this module is shown below:

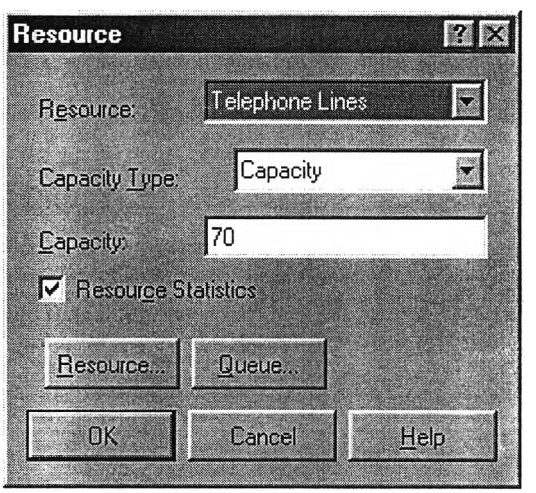


Figure 4-6: Resource module: Telephone Lines

The second resource module is used to define the port number of IVR, equal to 60 ports for the existing system. The number of port can be adjusted for analyzing the effect. The detail of this module is shown below:

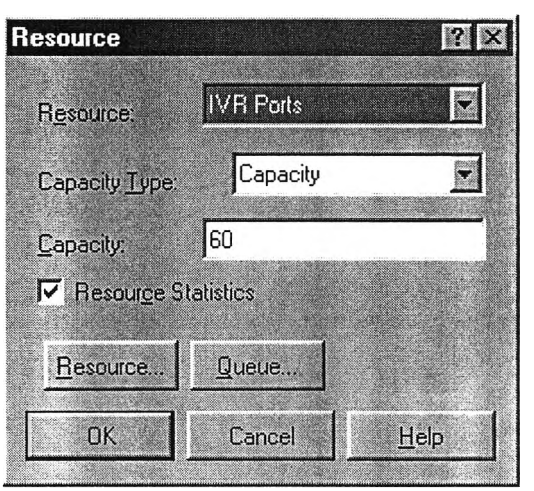


Figure 4-5: Resource Module: IVR Ports

The last resource module is used to define the number of the agent, varied by time. The number of agent can be changed for finding the appropriate condition. The detail of this resource module is shown below:

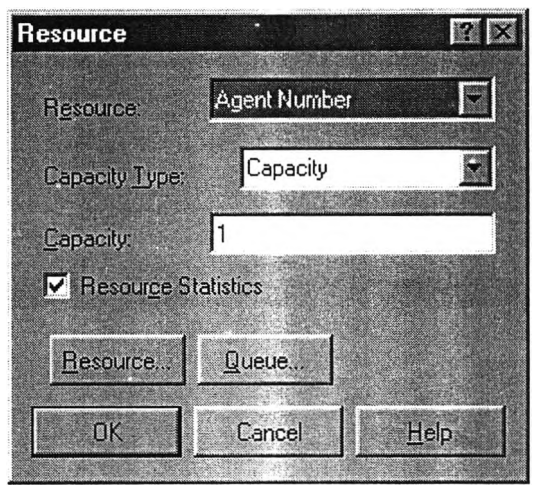


Figure 4-6: Resource Module: Agent Number

There are 5 variables: probability of the customer who want to contact only IVR, interarrival time, IVR service time, Transfer time, and Agent service time, in this model. The fist variable is kept in Variables Module in form of array while the rest is hold in Expressions Modules in form of array. The size of array is equal to 9 for representing the distribution and value in each simulation period.

Figure 4-7: Variables and Expressions Module

The detail of Variables Module is shown below:

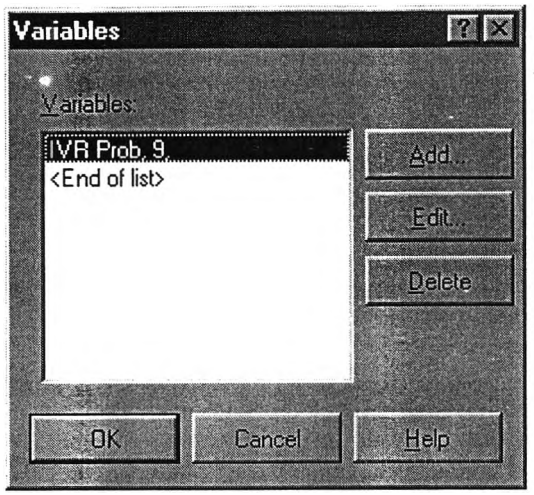


Figure 4-8: Detail of Variables Module

The detail of Expressions Module is shown below:

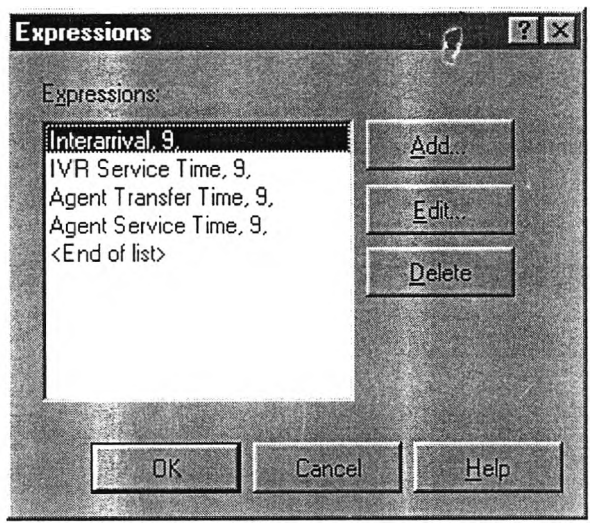


Figure 4-9: Detail of Expressions Module

Time Period Incrementing

The logic of time period incrementing is shown below:

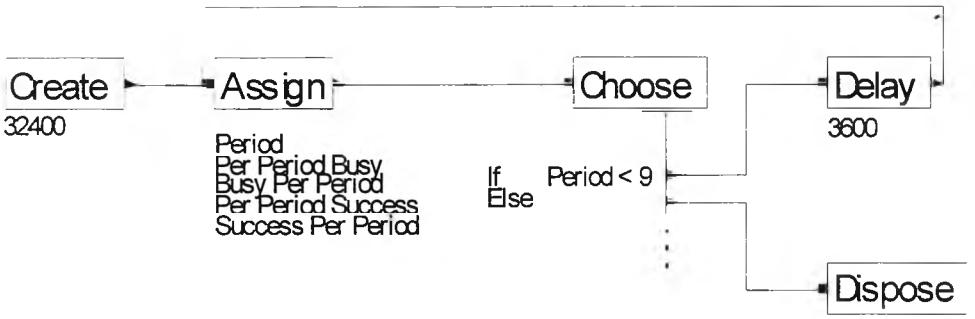


Figure 4-10: Time Period Incrementing Logic

These five modules are for setting and incrementing the time-period indicator and assigning variables that we'll use later. The general idea is to create the entity at the start of each day will be assigned the current time period, then delay that entity for 3600 seconds (one time period) and increment the time period by 1. When the time period equals to 9 (the end of the simulation), dispose of entity.

At the Create Module, we enter 32,400 for the time between arrivals – the length of simulation (9 hours). Thus, at the start of each new simulation, we create an arrival and set the user-defined variable Period equal to 0.

The entity is then sent to Assign Module where 5 assignments are made. The variable Period is incremented by 1 to reflect the start of the current 3600 seconds time period. The four additional user-defined variables, Per Period Busy, Busy Per Period, Per Period Success, and Success Per Period, are for keeping track of the number of balked and success phone calls during the 3600 second time period. In the next section, we'll increment the variable Busy Per Period by 1 each time a call balks from the system. Moreover, we'll also increment the variable Success Per Period by 1 each time a call enters the system. Thus, the first two assignments set the variable Per Period Busy equal to the number of calls balked during the last time period – 0 and set the variable Per Period Success equal to the number of success calls during the last time period – 0. The next assignments set the variable Busy Per Period equal to 0 and set the variable Success Per Period equal to 0, as we want to start over at the start of each time period.

The entity is than sent to the Choose Module. The Choose Module provides entity branching based on the If conditional rule in conjunction with the deterministic Else and Always rules. Branch destinations are defined by graphical connections or by specifying a label destination. When an entity arrives at the Choose module, it examines each of the defined branch options and sends the original arriving entity (the primary entity) to the destination of the first branch whose condition is satisfied. This process continues, creating duplicates (secondary entities) of the original entity for each true branch encountered until all defined branches are exhausted or until Maximum Branches Taken copies of the arriving entity have been generated. All secondary entities are exact duplicates (i.e., same attribute values and animation picture) of the original arriving entity. If no branches are taken, the arriving entity is

disposed. In this case, Take First True Condition is selected, and we also define our branches so there will always be a condition satisfied.

The detail of previously mentioned module is shown as follows:

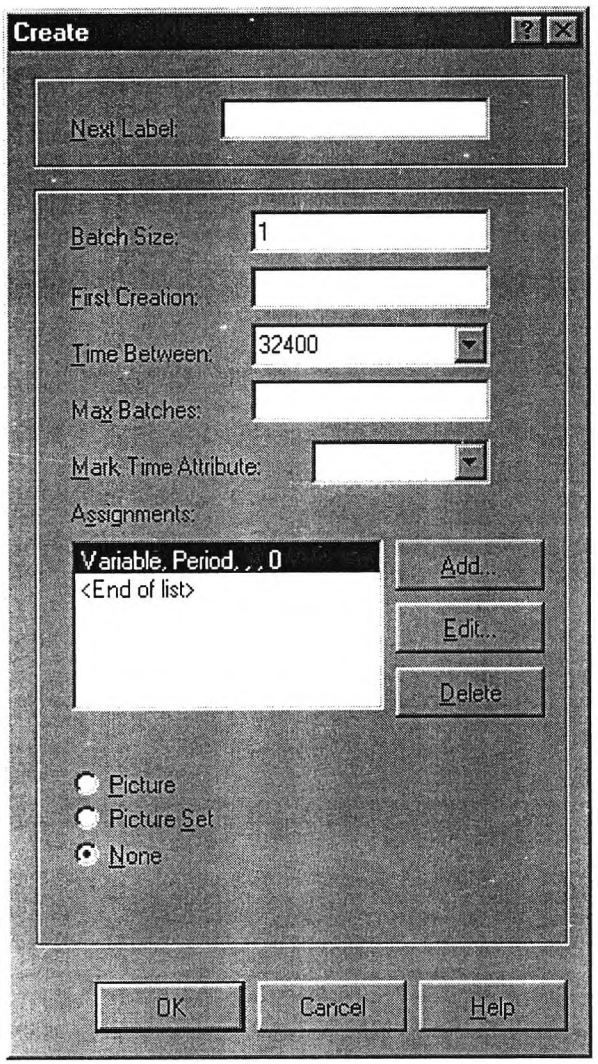


Figure 4-11: Create Module: Time Period Incrementing

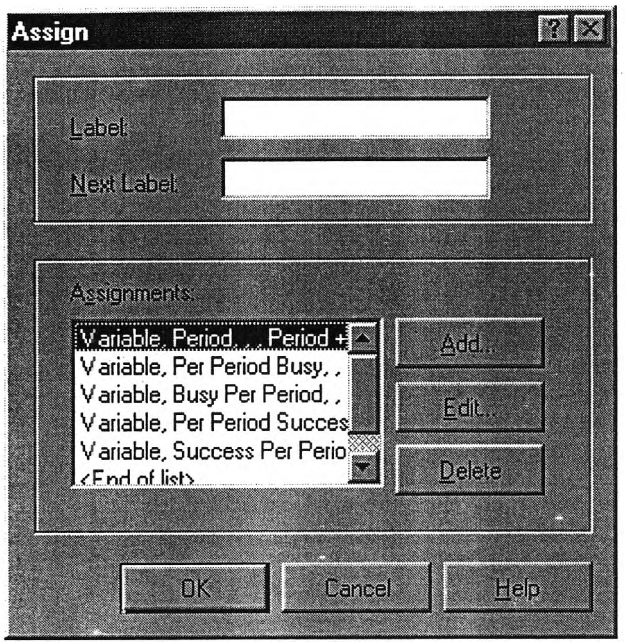


Figure 4-12: Assign Module: Time Period Incrementing

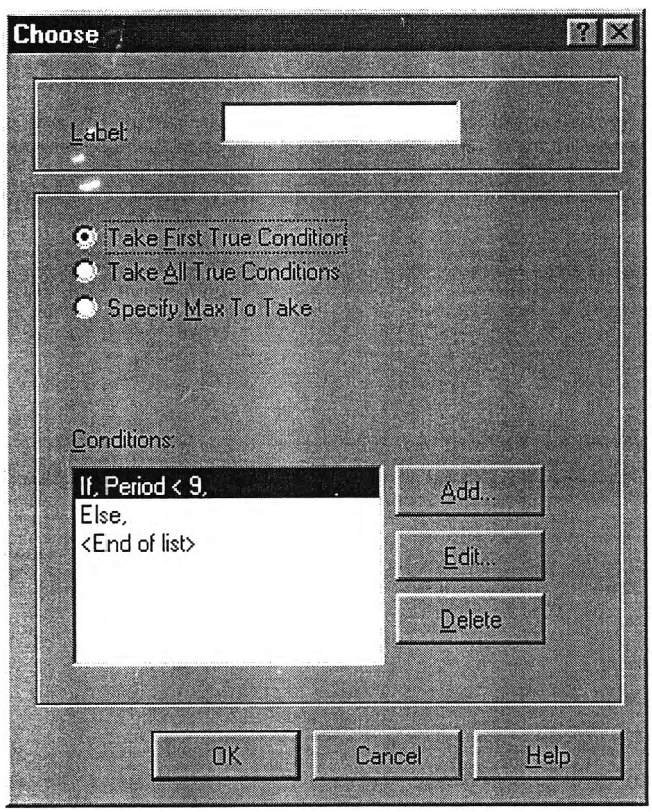


Figure 4-13: Choose Module: Time Period Incrementing

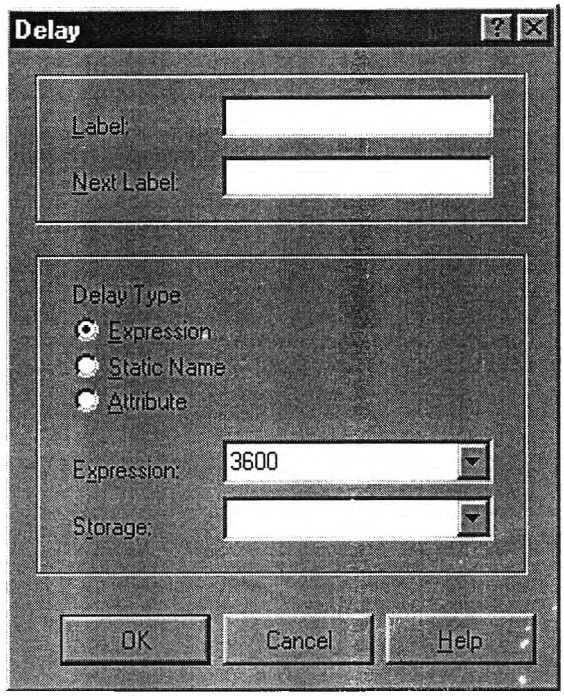


Figure 4-14: Delay Module: Time Period Incrementing

Crete Arrival For the System

After we have defined the data modules, the next step is to create the arrivals for the system. The detail of logic in this step is illustrated below:

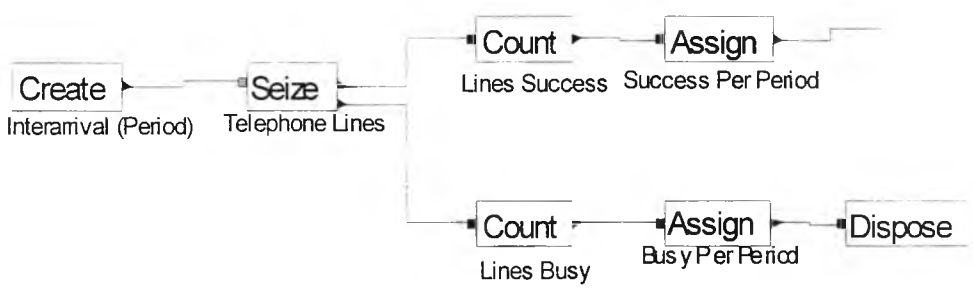


Figure 4-15: Arrival Process Logic

The Create module produces arriving entities to a model and may also specify initial attribute, variable, rate and level assignments to be made upon creation of each entity. In this module, the entities are assigned the mark time, named "Call In", to capture the time that the entities enter the model. The Time Between box is filled with expression "interarrival (period)", represent the interarrival time of the interested period. Moreover. we've offset after time 0 at the First Creation box by a very small, and somewhat arbitrary, value because we want to be ensured that the logic that define the expression in Time Between box has occur first. The figure of the module is illustrated below:

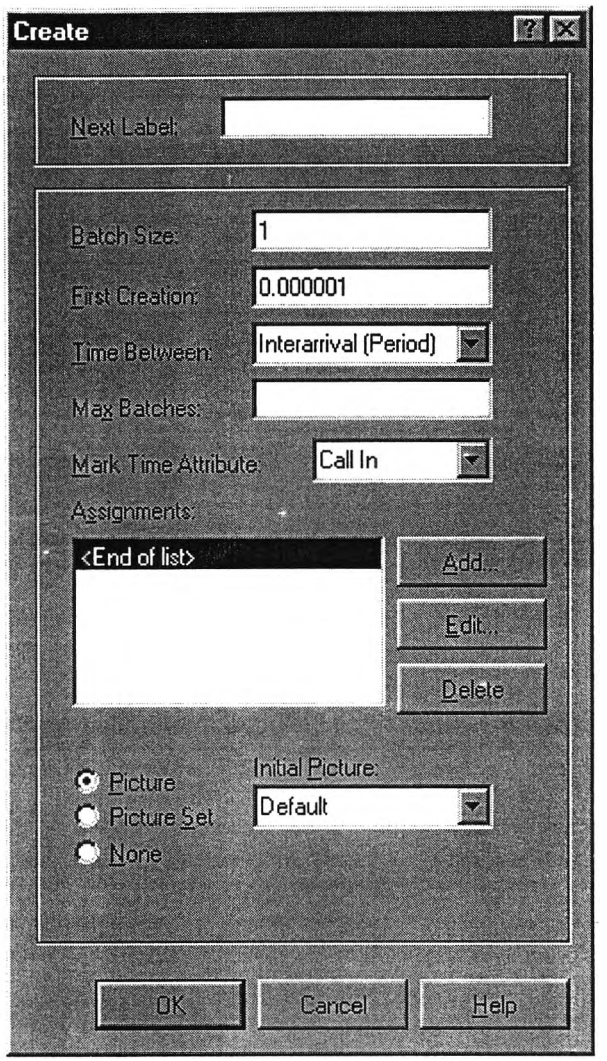
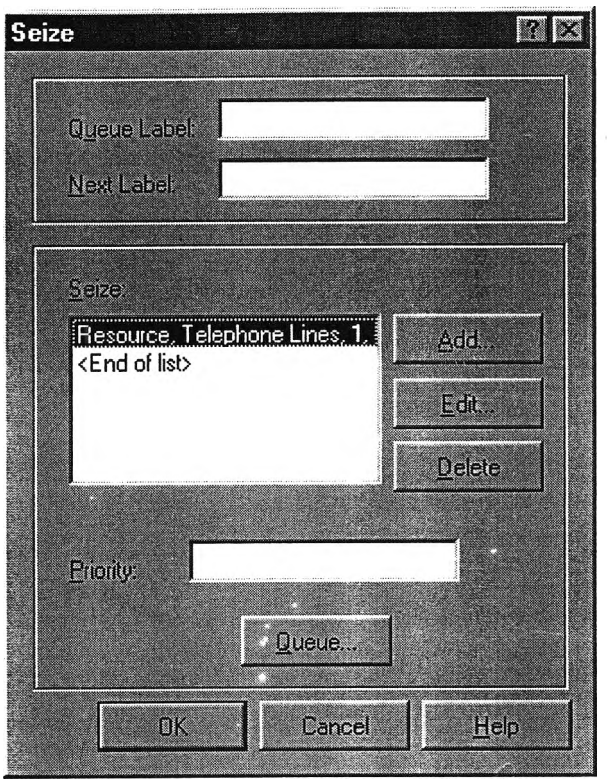


Figure 4-16: Create Module: Arrival Process

After an entity enter to the system, one of telephone line will be sized. This Seize module is shown below:



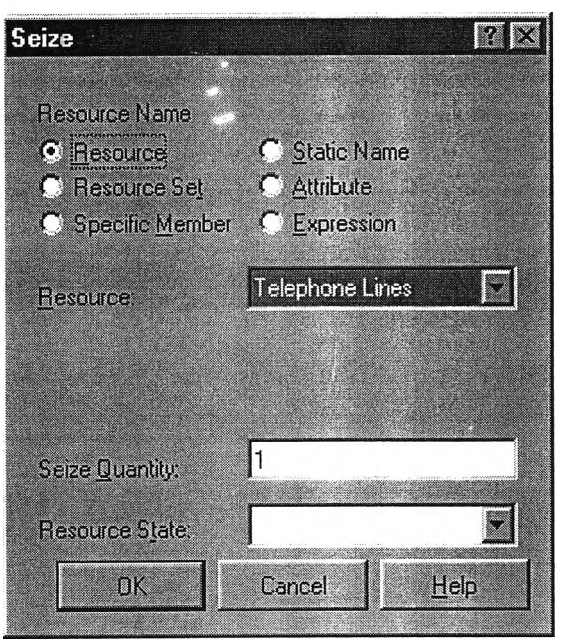


Figure 4-17 and 4-18: Seize Module: Telephone Lines

Because there is no queue in the telephone line, so the capacity of queue is set equal to 0. The detail of setting is illustrated below:

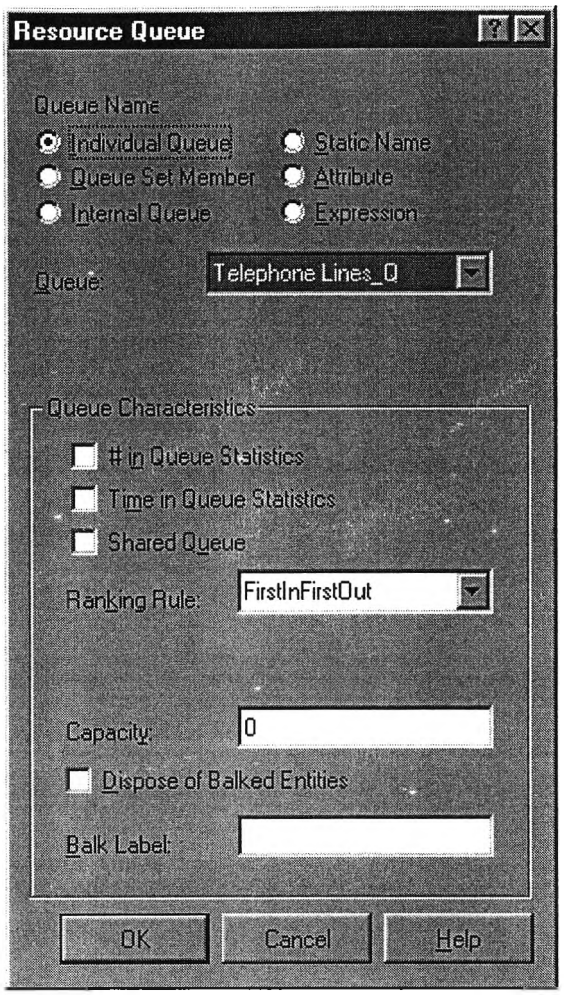


Figure 4-19: Queue Setting for Seize module (Telephone Lines)

If the telephone lines are available, the entity can enter the system. It's allocated to the arriving call, the entity is immediately sent out of the upper connection point in the right side of the Seize Module to the following Count Module, called Lines Success. If the telephones are unavailable, the entity is immediately sent to another Count Module, called Lines Busy. Both counters Lines Success and Lines Busy are incremented by 1. The detail of these Count modules is shown below:

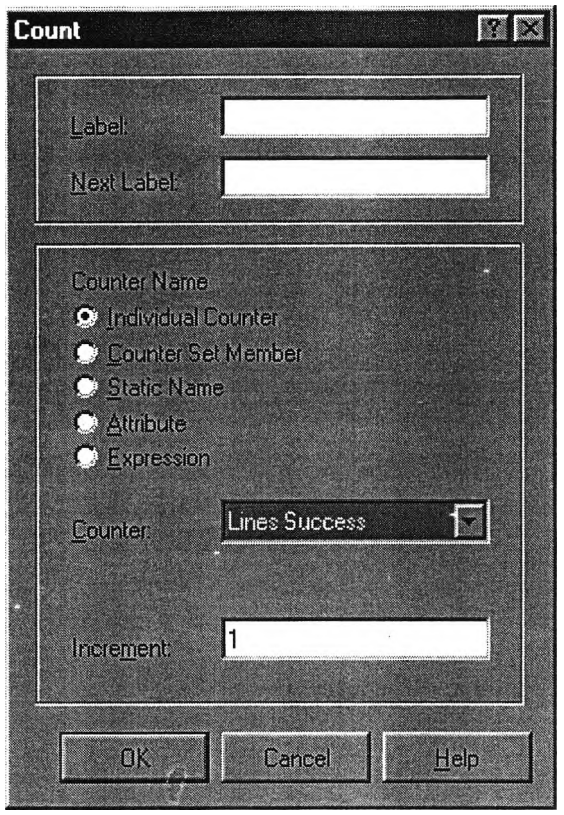


Figure 4-20: Count module: Lines Success

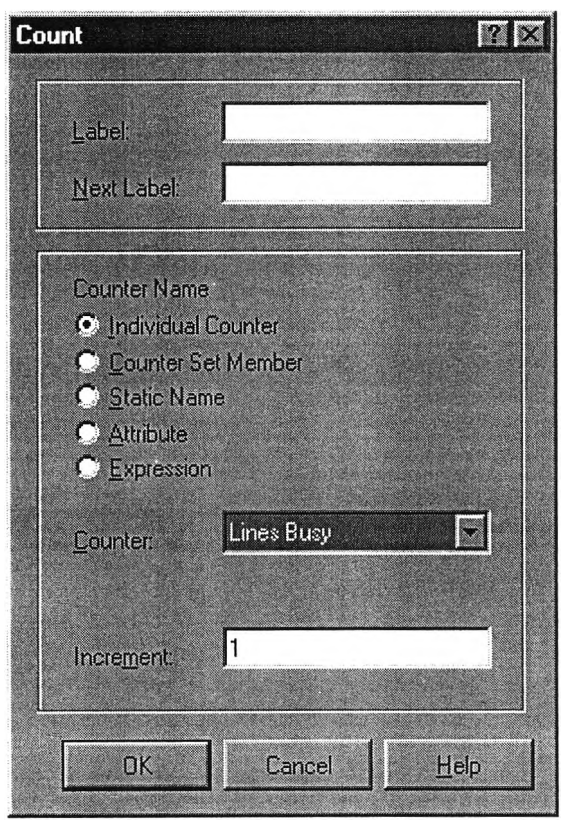


Figure 4-21: Count module: Lines Busy

In case of the entities can enter the system, they are directed to an Assign Module where the variable Success Per Period is incremented by 1 while, in case of the entities are balked to enter the system, they are sent to another Assign Module where the variable Busy Per Period is incremented by 1. Recall that in the previous section we reset this variable to 0 at the start of each 3600 second period, Thus, these variable will be equal to the total number of balked and success calls during that period. The detail of both Assignment Modules is shown as follows:

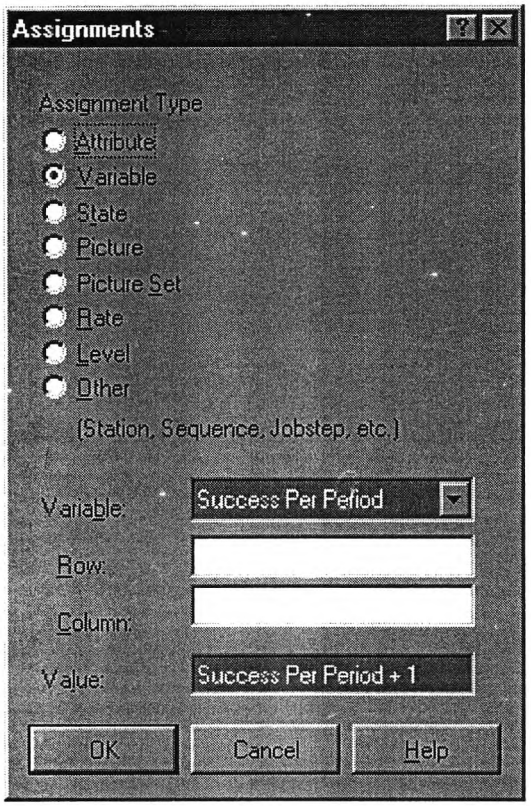


Figure 4-22: Assignment Module: Success Per Period

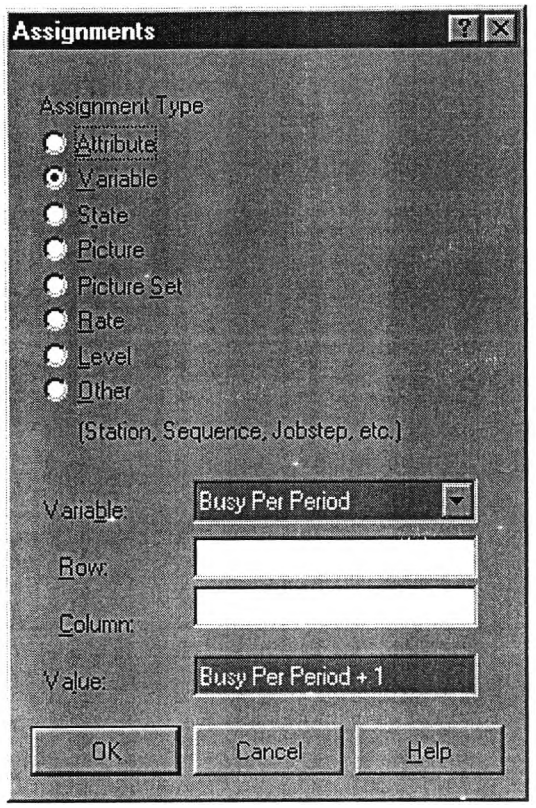


Figure 4-23: Assignment Module: Busy Per Period

IVR Service Process (No Agent Transfer)

After the entities can enter to the system, they will seize IVR resource first. The time of the entities that can enter to the system will be kept for analyzing the service time of IVR. The IVR ports will be seized by the enter entities. The detail of this logic is shown below:

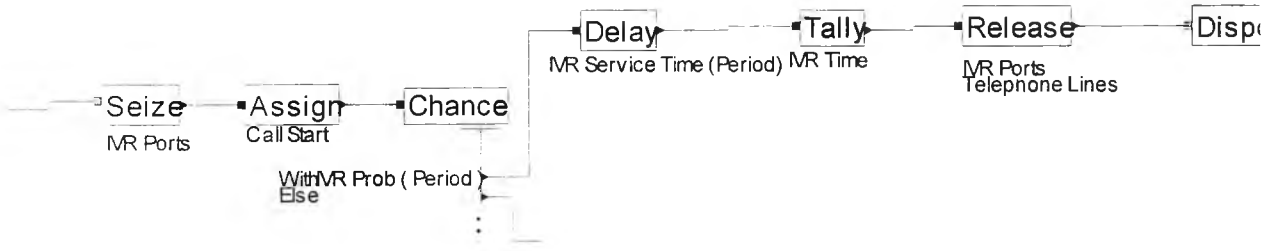


Figure 4-24: Seize the IVR ports logic

The detail of Assign module (Call Start) is illustrated below:

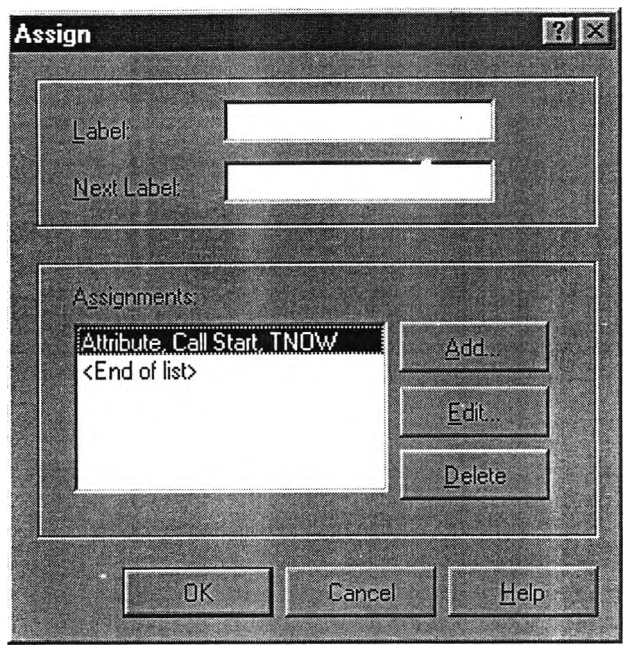


Figure 4-25: Assign Module: Call Start

TNOW is an Arena variable that gives the current simulation clock time.

The detail of Seize module (IVR Ports) is illustrated below:

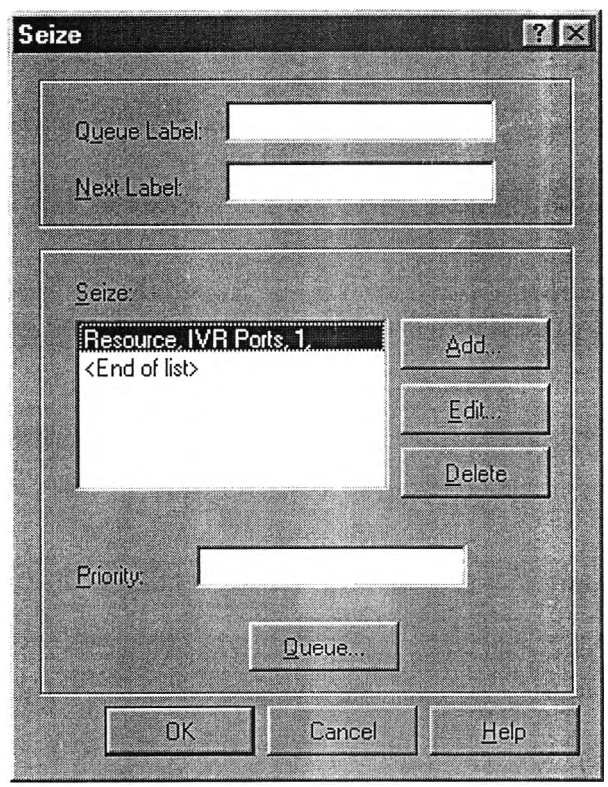


Figure 4-26: Seize module (IVR Ports)

However, because the queue may occur in case that all IVR ports are full, so it is necessary to keep queue statistic. The queue setting for Seize module (IVR Ports) is shown below:

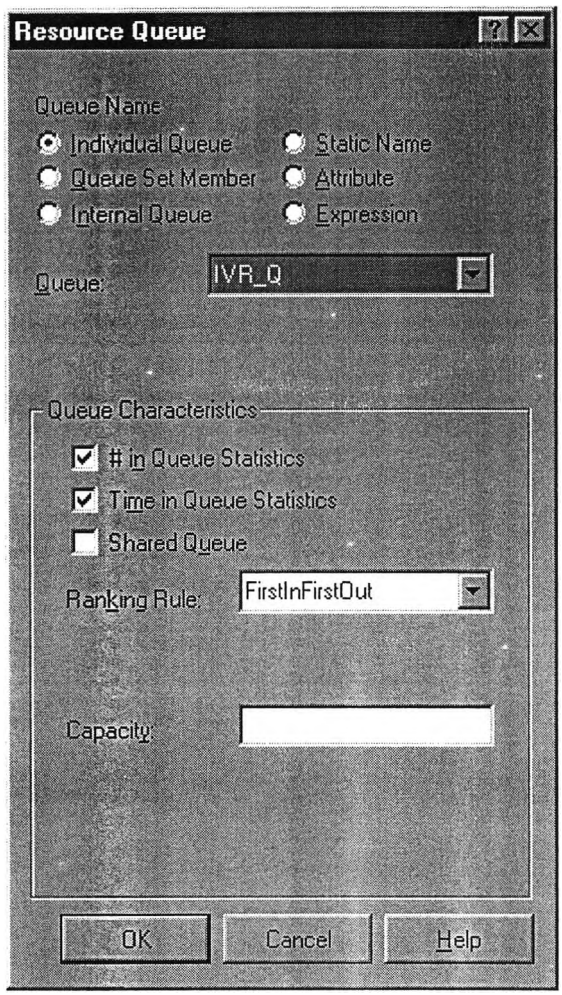


Figure 4-27: Queue setting for Seize module: IVR Ports

The entities will be classified into 2 groups: the entities that want to contact with agent and the entities that don't want to contact with the agent. The Chance module provides branching based on the With probability rule and the Else deterministic rule. When an entity enters at a Chance module, it determines which probability branch, if any, will be taken based on a random sample and the probabilities of the With statements specified. If the random sample falls outside of the cumulative probability range representing the With probability branches, the Else branch is taken. In this case, the Probability box is filled with the variable IVR Prob (Period), which represents the probability that customer would like to contact only IVR. The detail od Chance module is shown below:

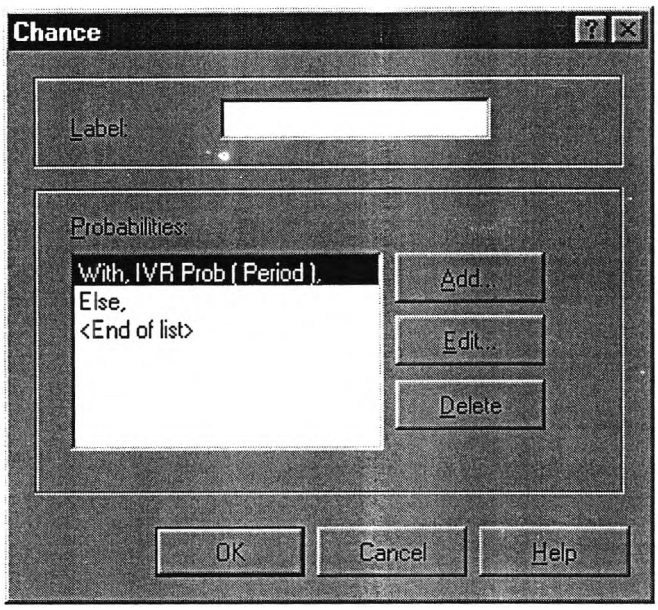


Figure 4-28: Chance Module

In case of the customer would like to contact only agent, the entities will be sent to the following Delay module, represents the IVR service time. The Expression box is filled with the expression, IVR Service Time (Period). The detail of this Delay Module is shown below:

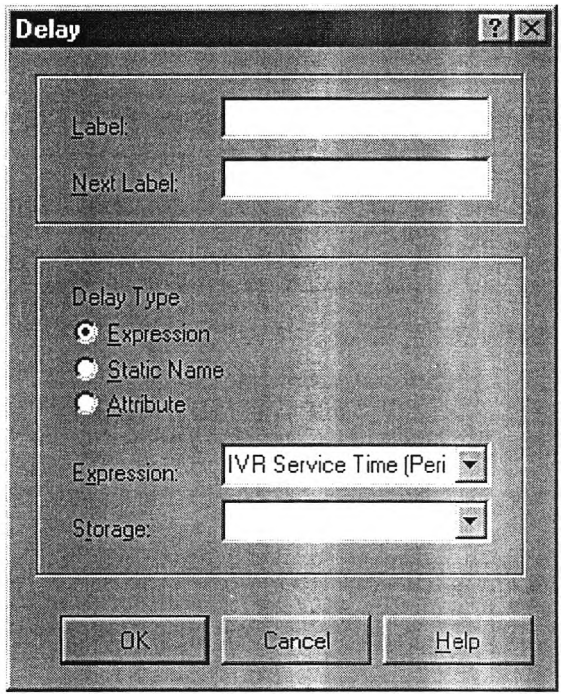


Figure 4-29: Delay Module: IVR service time

Then, the entities will be sent to the Tally module. The Tally module records observations to a specified tally. The observation is defined either as an expression or as one of two special functions, Interval or Between. The Interval records the time between a previous event and current simulated time for the entity arriving at the Tally module (often used for collecting time in system statistics). The Between type

of Statistics stores the time between arrivals of entities to this Tally module. Tallies maintain statistics on the mean, coefficient of variation, minimum and maximum values, and number of observations. Individual observations may be recorded in an external file. In this case, the time recorded is the interval of call start, assigned after seizing the IVR resources. This Tally Module is defined as IVR Time. The detail of this Tally module is shown as follows:

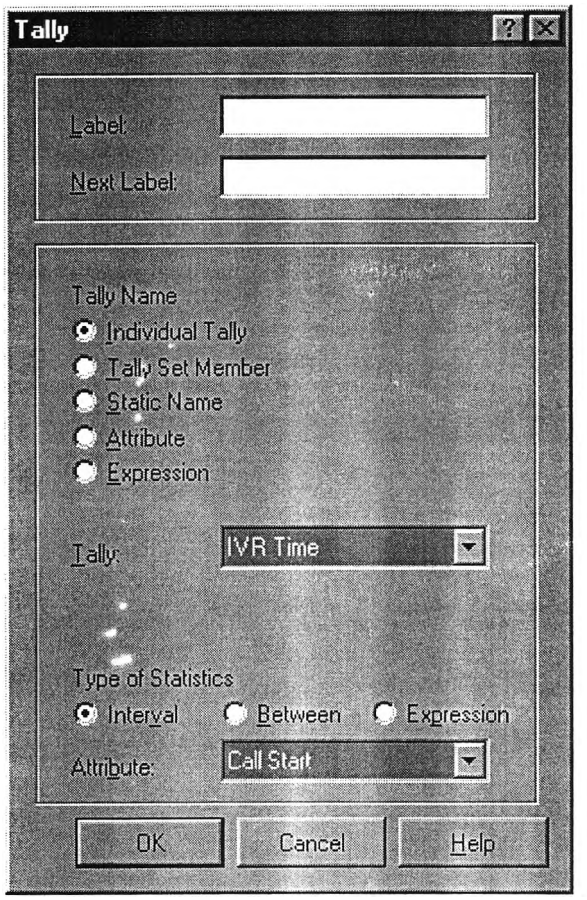


Figure 4-30: Tally Module: IVR Time

Then, the entities are sent to the Release Module. The Release module is used to release units of a resource that an entity previously has seized. The Release module may be used to release individual (simple) resources or may be used to release resources declared in a set. For each resource to be released, the name and quantity to release are specified. When the entity enters the Release module, it gives up control of the specified resource(s). Any entities waiting in queues for those resources will gain control of the resources immediately. In this case, the IVR Port Resource and Telephone Line Resource are released from these entities. The detail of Release module is shown below:

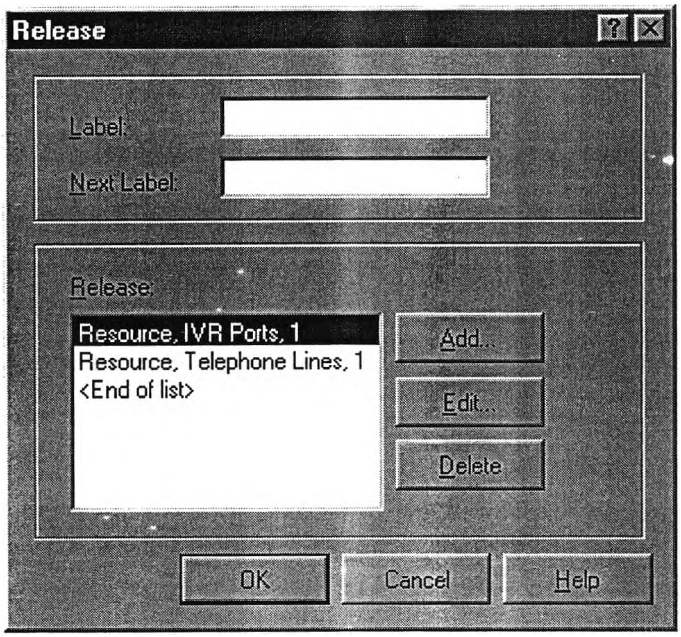


Figure 4-31: Release Module: IVR no agent transfer

Transfer to Agent

In case of the customer would like to contact with IVR and agent, or agent only, the entities will be sent to another Delay module, represents the service time that serve the customers who require contacting agent. However, when the customers go to agent part, they will release the IVR Port Resource. The detail of transferring to agent logic is shown below:

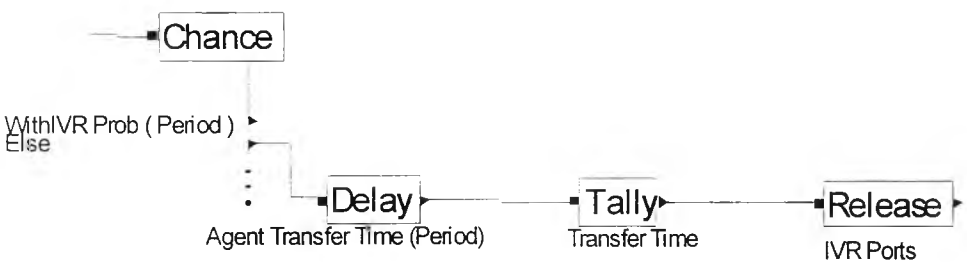


Figure 4-32: Transferring to Agent Logic

The Expression box of the Delay module in this case is filled with the expression Agent Transfer Time (Period). This module represents the transfer time (may include the service time that customers request the services of IVR). The detail of Delay module is shown below:

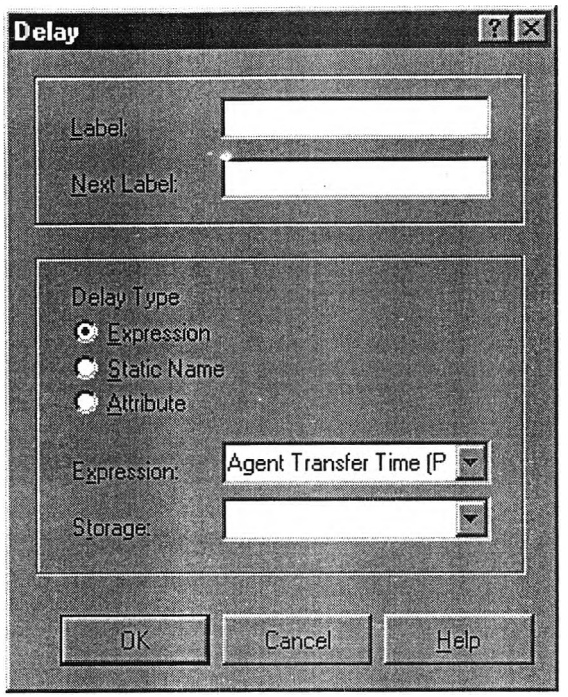


Figure 4-33: Delay Module: Transfer to agent

Then, the entities are sent to the Tally module to record the transfer time. This Tally Module is called Transfer Time. The detail of it is shown below:

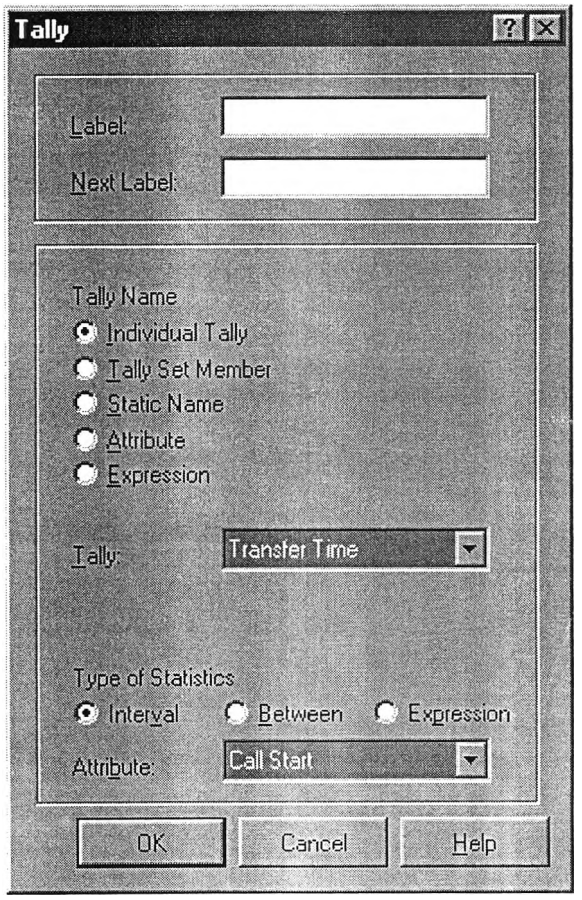


Figure 4-34: Tally Module: Transfer Time

Then, the entities will be sent the Release module to release the IVR port. The detail of this Release module is illustrated below:

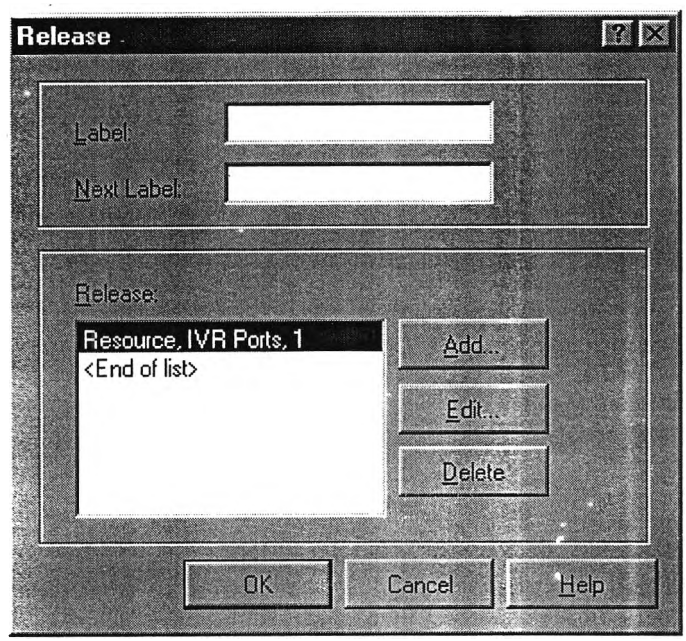


Figure 4-35: Release Module: Transfer to Agent

Contacting Agent

After the entities release IVR Port Resource, they will be served by the agent. Then, they will release the Telephones Lines Resource and Agent Number Resource. However, because the agents need approximately 45 seconds to make the wrap-up heading so the agents will available after they complete making the wrap-up heading. The detail of contacting agent logic is shown below:

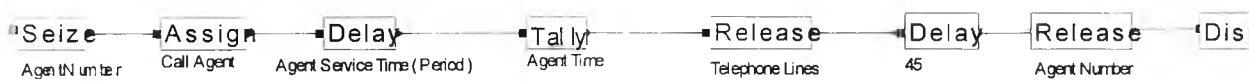


Figure 4-36: Contacting Agent Logic

The detail of this Seize module is illustrated below:

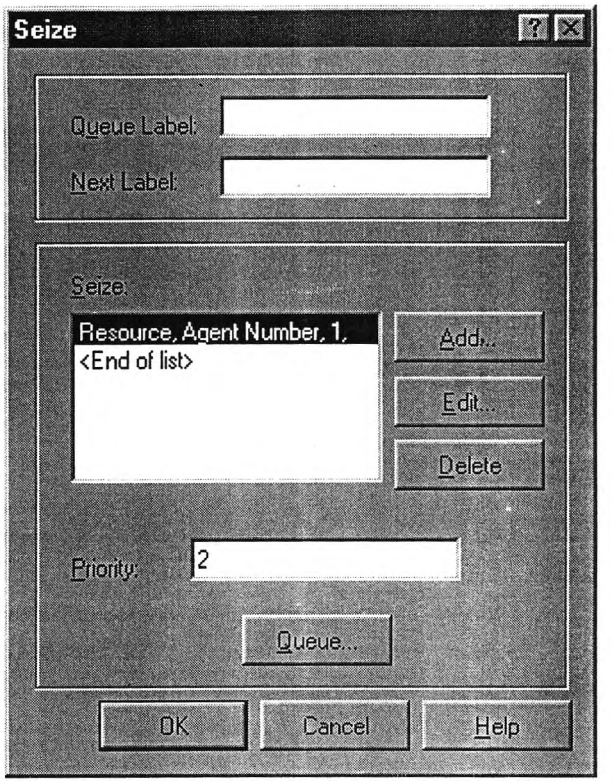


Figure 4-37: Seize module: Contacting Agent

However, in this module, the entities is set the priority as 2 because there will be the executive customers that also want to use the agent service, but these customers have the higher priority than normal customers so the executive customers will be set the priority as 1.

Moreover, for queue setting of this module, the queue name of this module is called Agent_Q, and this queue is shared from the executive customer. The queue setting of this module is shown below:

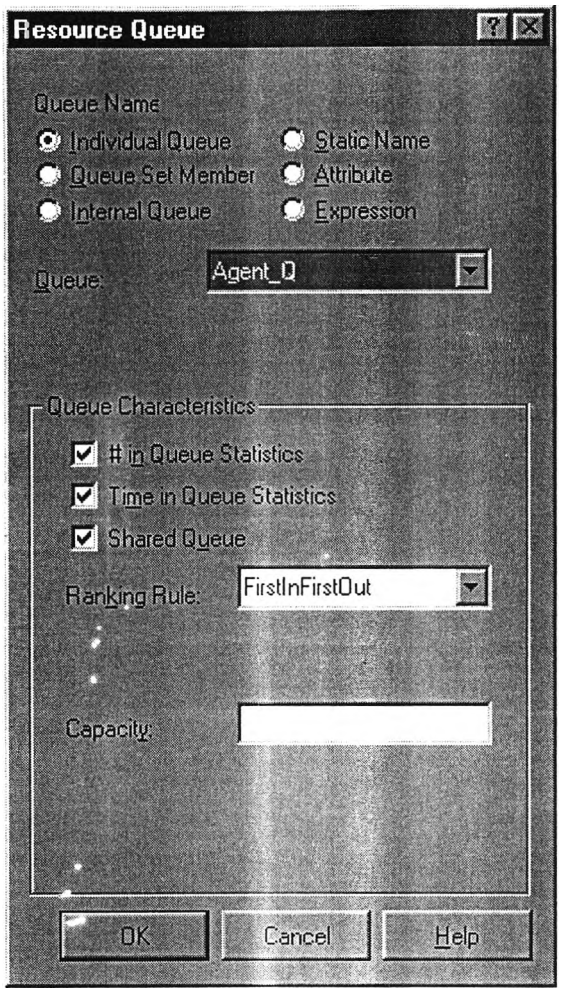


Figure 4-38: Resource Queue Setting: Agent Number Resource

After seizing the Agent Number Resource, they will sent to the Delay module, represents the service time of agent. The Expression box of this module is filled with the expression Agent Service Time (Period). The Delay module is shown below:

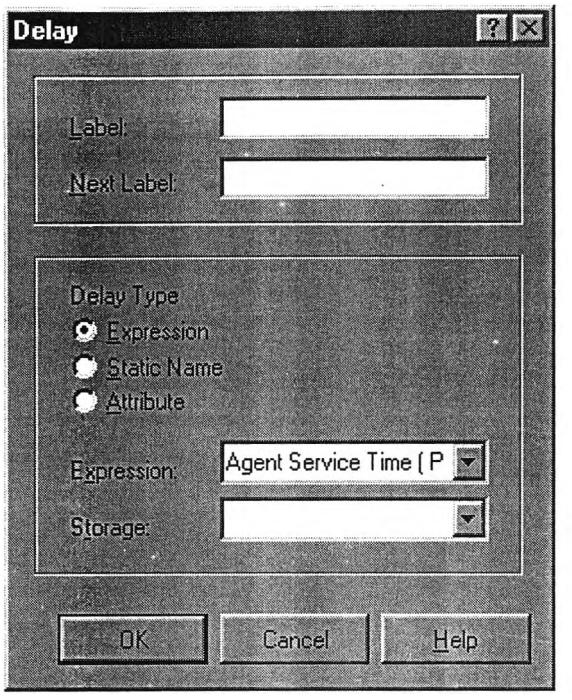


Figure 4-39: Delay Module: Agent Service Time

Then, the entities will be sent to the Tally module to record the agent service time. This Tally module is named as Agent Time. The detail of this Tally Module is shown below:

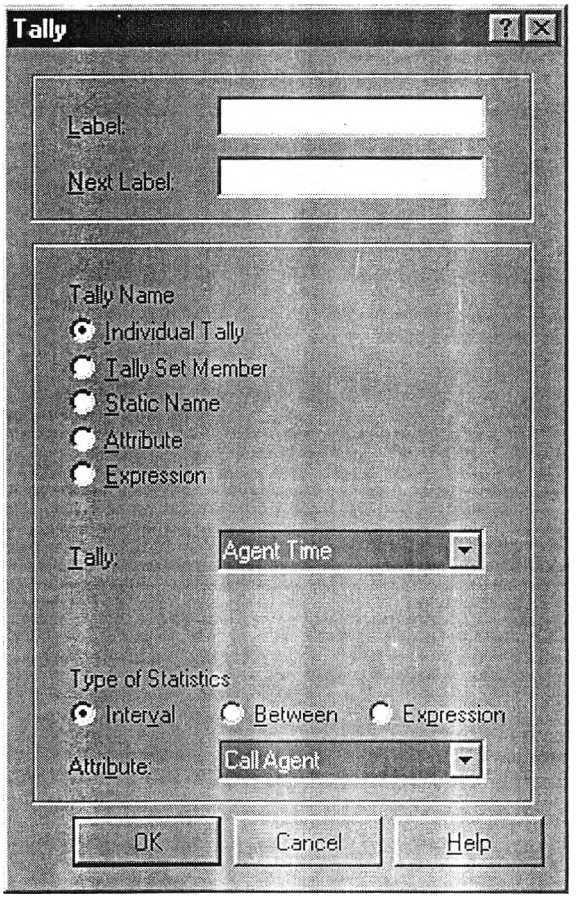


Figure 4-40: Tally Module: Agent Time

Then, the entities will release Telephone Line Resource. The detail of this Release module is shown below:

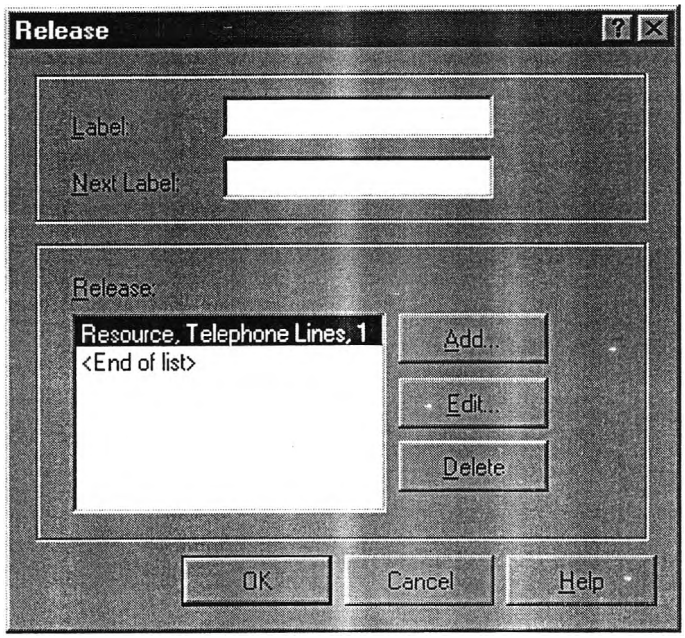


Figure 4-41: Release module 1: Contacting to Agent

The agents use approximately 45 seconds to make the wrap-up heading, then the Agent Number Resource will be released. The detail of these modules is shown below:

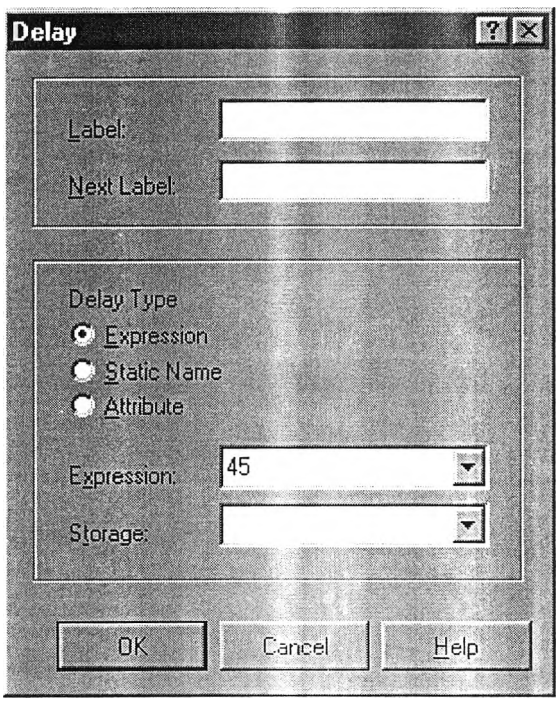


Figure 4-42: Delay: Making Wrap-UP heading

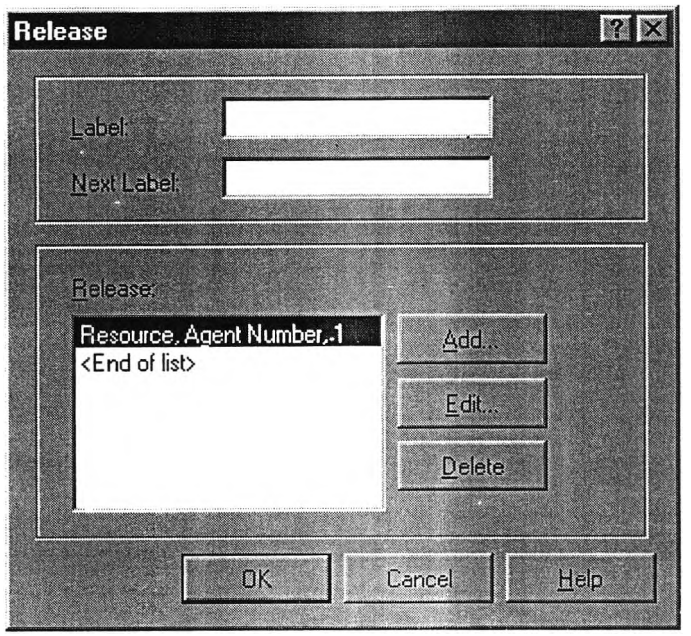


Figure 4-43: Release module 2: Contacting to Agent

Contacting to Agent by Executive Customer

From the interview with call center officer, there are about 50 executive customers, such as board of the bank or important person, per hour require contacting with the agent directly by using another telephone lines. The logic of this process is shown below:

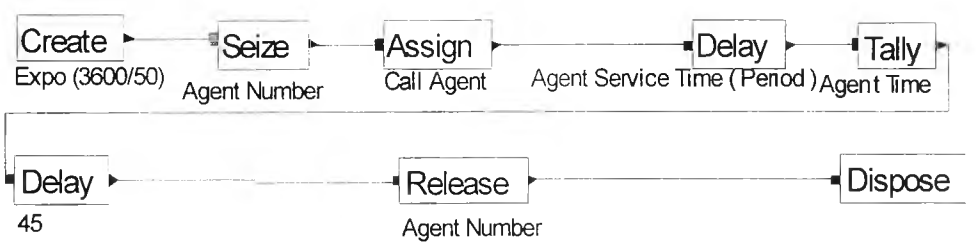


Figure 4-44: Contacting to Agent by Executive Customer Logic

Create module represent the number of executive customer per hour so the Time Between box is filled with expression Expo (3600/50). The detail of Create module is shown below:

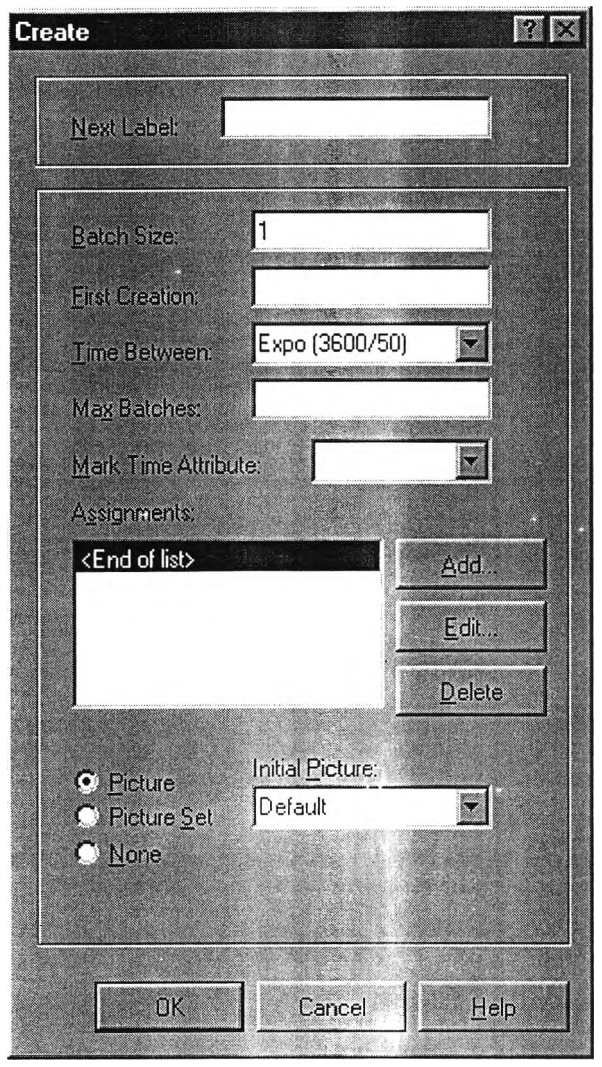


Figure 4-45: Create Module: Executive Customer

The detail of other modules in this logic is the same as that in Contacting to Agent logic except seize module. The Seize module in this logic is set the entity priority as 1. The detail of this module is shown below:

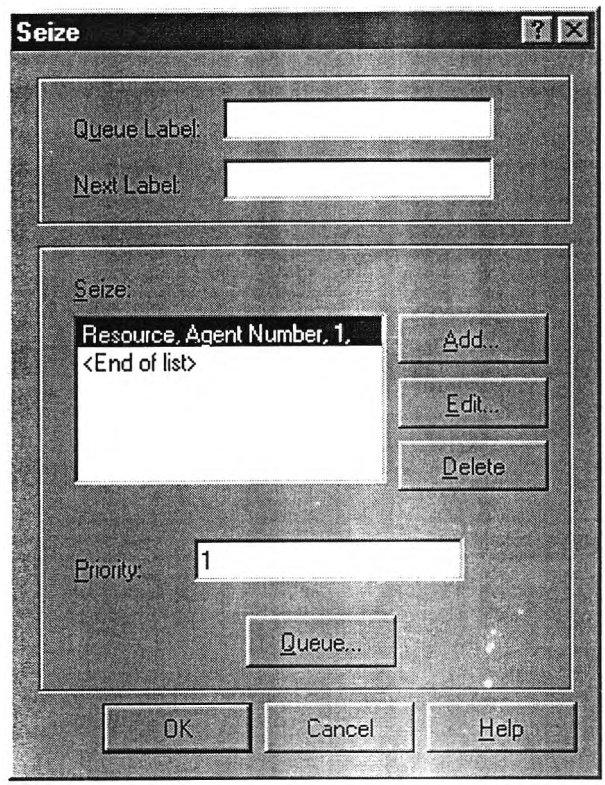


Figure 4-46: Seize Module: Contacting to Agent by Executive Customer

4.5 Model For Determining the Dynamic Resource (Agent)

For determining the appropriate number of agent, we assume that the customers can contact to the agent randomly, freely, and independently so the interarrival time for the agent is exponential distribution. And, the parts of model that concern with IVR port resources and telephone line resources is not used for this purpose so that the logic of contacting the agent is shown below:

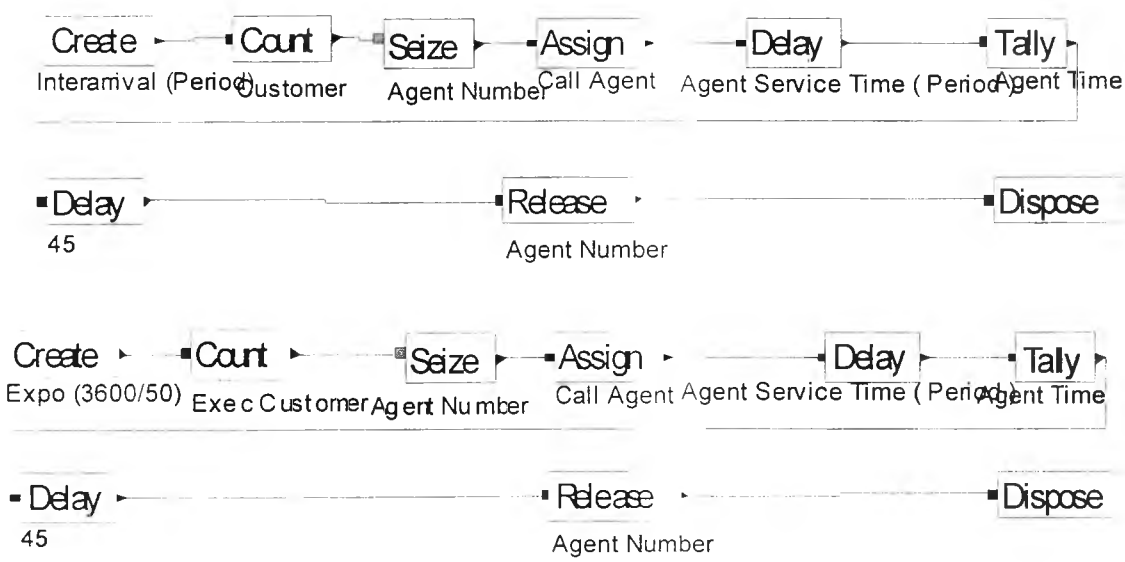


Figure 4-47: Model for finding out the appropriate number of agent

The detail of these modules is the same as them that mentioned above. In this model, we also assume that there are approximately 50 executive customers on the peak period of workdays and interarrival time distribution of this customer type is exponential.