

Introduction to Traffic Modeling and Resource Allocation in Call Centers

Cars on a road, a customer line at a bank, or telephone calls arriving at a call center, share similar traffic characteristics. Load may be high, leading to long travel and wait time, or may be light and move smoothly. Highways, toll booths, telephone lines and bank tellers may be overloaded, generating long delays and providing poor service, or could be idle due to low demand. Planner and analysts must be able to estimate the right number of resources—toll booths, bank tellers, support agents and telephone lines—to provide adequate service at reasonable costs.

Traffic modeling provides the theory and practice to analyze traffic patterns and determine the necessary resources to handle it optimally. Traffic modeling originated in the telephone industry, and many of the theories still in use today were developed between 1909 and 1917 by the Danish mathematician [Agner Krarup Erlang](#).

Traffic Models

Basic Definitions

Sources and Servers

Traffic modeling involves *sources* generating service requests and *servers* that fulfill these requests. In a telephone system, sources are the callers and the servers are the telephone company's resources that provide the dial tone and rout the calls to their destination. In a bank, the customers are sources, and the bank tellers are the servers.

Traffic modeling assumes that there is very large number of sources R requesting service, and a limited number of servers N . The number of sources is significantly larger than the available servers, so that virtually $R \rightarrow \infty$ (infinity). In addition, we assume that:

- Sources generate service requests at random and independently of each other.
- The average number of service requests per time unit from all sources is constant.
- Service requests arrive at intervals that follow a *Poisson distribution* (see below).
- The time required to service a request is distributed *exponentially* (see below), and is independent of the arrival rate.
- Service requests are honored on a first-in, first-out (FIFO) basis.

Traffic Volume and Intensity

The volume of traffic is determined by the number of service requests per time unit and the time that a server satisfies each request consumes. For instance, if the arrival rate of 100 calls per hour (CPH), and each call requiring 9 minutes (0.15 hour) of service, the traffic volume in an 8-hour day is: $100 * 0.15 * 8 = 120$ Call Hours (Ch)

Traffic intensity or load is measured as traffic volume per time unit and is measured in *Erlang units*. One Erlang equals one Ch/hour, or, stated differently, one Erlang equals one telephone line carrying traffic for one hour. The traffic load in the previous example is $120 / 8 = 15E$.

Calls Arrival Pattern

A naive approach to figuring out the number of agents needed in a call center is to divide the number of calls expected in an hour by the average length of the calls. For example, if servicing each call takes, on the average, 15 minutes, then each service agent can take 4 calls per hour. If 100 calls arrive in one hour, it appears that 25 agents and 25 telephone lines should be able to service the anticipated call load.

The flaw in this logic is that service requests do not arrive in an orderly fashion one right after the other. Like customers at a bank, telephone calls arrive at random times and independent of each other: some calls will arrive while others are still being served; some calls will arrive simultaneously, and during periods of the day no calls will arrive at all. The *Poisson distribution* expresses the probability of a number of events occurring in a fixed period of time:

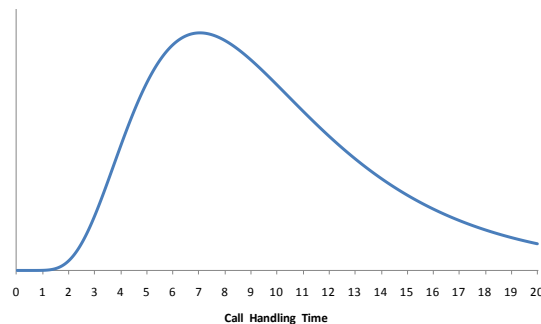
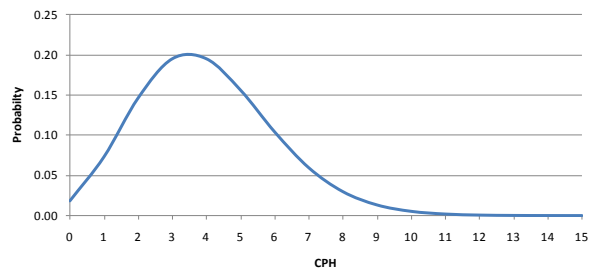
$$P_p(\lambda, x) = \sum_{i=0}^x \frac{\lambda^x e^{-\lambda}}{x!}$$

Where λ is the mean arrival rate and x is the arrival time.

The Poisson probability distribution on the right shows the probability of arrivals of calls with mean arrival time of 4 per hour, which is what the example above assumes each agent will experience.

The duration of service requests is not uniform either. Call lengths are distributed exponentially, and as Figure 2 shows, most calls are shorter than the average call, but some are much longer than the average.

Once the phone lines have filled to capacity and all servers are busy, there is an equal probability of a call ending and a new call starting, reaching a *stochastic equilibrium*.



Erlang B model

There are variants of the model originally developed by Erlang. Erlang B is a "blocked calls lost" model, in which, if servers are unavailable, the source (service request) is denied service. This is the situation in a telephone switch, where all resources (telephone trunks) are exhausted and the caller receives a busy signal or is diverted to a voice mail system. The only way to receive service is to hang up and redial repeatedly until a server becomes available.

Erlang B calculates the blocked call probability (*loss probability*) for a given traffic load and a number of servers. $P_B(N,A)$ is the probability that a caller will receive a busy signal with a traffic load of A Erlangs and N telephone trunks.

$$P_B(N, A) = \frac{\frac{A^N}{N!}}{\sum_{i=0}^N \frac{A^i}{i!}}$$

Where:

A is traffic load in Erlang units

N telephone trunks

λ is the mean arrival rate

x is the arrival time

There is an easy way to calculate blocking probability using Erlang B tables available online (e.g. [Bilkent University](#)). Table 1 shows traffic loads that 15 to 25 servers can support with loss probabilities of 1%, 2%, 5% and 10%. For instance, if the anticipated load is 15 Erlangs, and the desired blocking probability is 2% or better, the number of telephone lines to handle this traffic load is 23. Because Erlang B is a blocked calls lost model, the number of service agents (CSRs) is the same as phone lines, or 23. If resources are at a premium and degradation in service to a blocking probability of 10% is acceptable, the CSR headcount can be reduced to 18.

Erlang B tables are available in different configurations, for example, showing blocking probabilities for different traffic load and server combinations. For instance, using Table 2, we can determine that for the given load of 15 Erlangs, a staff of 20 will provide service with a blocking probability of 0.0456 — slightly better than 5%.

Table 1. Partial Erlang B Table

N	Loss Probability			
	1%	2%	5%	10%
15	8.108	9.010	10.633	12.484
16	8.875	9.828	11.544	13.500
17	9.652	10.656	12.461	14.522
18	10.437	11.491	13.385	15.548
19	11.230	12.333	14.315	16.579
20	12.031	13.182	15.249	17.613
21	12.838	14.036	16.189	18.651
22	13.651	14.896	17.132	19.692
23	14.470	15.761	18.080	20.737
24	15.295	16.631	19.031	21.784
25	16.125	17.505	19.985	22.833

Table 2. Partial Erlang B Table

A	Servers			
	17	18	19	20
14.0	.0861	.0628	.0442	.0300
14.5	.0994	.0741	.0536	.0374
15.0	.1132	.0862	.0637	.0456
15.5	.1273	.0988	.0746	.0546
16.0	.1417	.1118	.0860	.0644

Erlang C Model

Unlike the Erlang B model, in which blocked service requests are lost, the Erlang C model describes the behavior of a call center in which requests that cannot be satisfied immediately are delayed until a server is available. The model defines the probability $P_c(N,A)$ that a service request will have to wait for service if N agents are assigned to handle traffic of A Erlangs:

$$P_c(N, A) = \frac{\frac{A^N N}{N!(N-A)}}{\sum_{i=0}^N \frac{A^i}{i!} + \frac{A^N N}{N!(N-A)}}$$

Where:

- A is traffic load in Erlang units
- N telephone trunks
- λ is the mean arrival rate
- x is the arrival time

Unlike Erlang B probability, which can be calculated using static tables, Erlang C calculations are carried out iteratively. The call center planner has to calculate different staffing levels iteratively until the probability of delayed (queued) calls is less than the target service level as described in the following sections. Special call center calculator software such as [EasyErlang](#) is a practical approach to performing this task.

Call Center Metrics

In this section we discuss the key metrics used in the planning and analysis of call centers.

Call Load: Calls Per Hour (CPH) and Average Handling Time (AHT)

The volume and intensity of incoming service requests are key factors in call center planning, as discussed earlier. The metrics that define call load are Calls Per Hour (CPH) and Average Handling Time (AHT), and is measured in Erlang units.

AHT is the time it takes a CSR to service to a single customer call, and includes the time the agent provides service (talk time), as well as any additional activities to complete a call and prepare for the next one (wrap-up time).

Peak Hour Traffic (PHT), Busy Hour Traffic (BHT)

Peak hour is the busiest one-hour period of the day, when incoming service requests are most likely to be delayed or blocked and turned away. This is the call load for which resources are calculated.

While sufficient resources need to be available to handle peak traffic, it is a good practice to establish traffic arrival and duration patterns during the course of the entire day and each day of the week. Daily traffic should be sampled at half-hour or even 15-minute intervals, because the peak time is unlikely to correspond with the sampling interval and therefore measured incorrectly. Analyzing resource requirements during different times of the day and all days of the week will allow better optimization of daily staff schedules.

Average Speed of Answer (ASA)

Average speed of answer (ASA) is the average time a caller waits to speak with an agent. In general, averages are acceptable for estimations and trending, but they have to be used with great caution because of the high variability in the natural distribution of call arrival and duration as explained in the previous section. and many callers will experience delays significantly longer than the average. For instance, a staff of 12 taking 80 calls per hour with AHT of 7 minutes can deliver an average speed of answer of 50 seconds. However, as we will see later, this average figure applies to only 78% of the calls; 22% of the callers will experience longer delays, and some are likely to abandon the queue before their turn arrives.

Grade of Service (GoS)

Instead of targeting ASA as a single figure of merit, a more appropriate and precise method is to set a desired *grade of service*, which is the percentage of calls that will be answered within a target threshold. For example, a target grade of service may be for 90% of the calls to be answered within 15 seconds, and for the remaining 10% that will end up waiting longer, the delay will be no longer than 60 seconds. A good call center design should establish the staffing level and telephone lines that are required to support that grade of service. Moreover, it should ascertain how many callers will miss the 10 second target and characterize their experience: how long they will have to wait to receive service and how many are likely to abandon the call prematurely.

Putting it Together: Calculating Resource Levels

In call centers, planners must establish the required number of agents and allocate the necessary telephone lines, balancing the desired level of service against the availability and operating costs of these resources.

Telephone Lines

The computation of the required number of telephone trunks is based on the Erlang B model. The target blocking probability depends on the service model employed in the call center.

If the call center is designed as "loss" model, that is, calls that cannot be serviced immediately are diverted to a voicemail service or simply receive a busy signal, you use the Erlang B tables discussed earlier to calculate the

number of telephone trunks that will provide an acceptable level of service. A blocking probability of 5% or better is usually considered adequate.

However, most call centers cannot afford the staff to operate in a pure "blocked calls lost" mode and still maintains high service levels; they must employ a queuing system and enough telephone trunks to allow callers to hold for as long as they wish. In practice, it is impossible to place an infinite number of calls on hold, so the number of lines is set so that only in rare cases will callers be denied the opportunity to wait for service and receive a busy signal. Use Erlang B tables to calculate the number of lines that will provide sufficiently low blocking probability. The examples in the following discussion were calculated to deliver a loss probability of 0.001 (0.1%), although 1% should suffice in most cases.

Staffing

In a well designed call center, some calls, especially during peak time, are queued. The first step in calculating staffing levels is to establish a target grade of service. Calculating staffing levels to support that target is an iterative process and is most easily carried out using an Erlang C software program or a spreadsheet.

The screen image below shows the output of the [EasyErlang Erlang C calculator](#). The target service level (GoS) is defined as 95% of calls should be answered within 30 seconds. The maximum allowed wait time is 20 seconds, after which we assume callers will abandon the queue. The expected call volume is 230 calls per hour (CPH), and the average handling time (AHT) is 510 seconds.

Agents	SL %	SL Time	ASA	Abandoned	Capacity	Tolerance	Queued	Q Time	Max. Q Time
43	90%	48	11	12%	217	-6%	16%	69	650
44	93%	35	7	8%	224	-3%	12%	61	622
45	95%	23	5	6%	230	0%	8%	54	607
46	97%	8	3	4%	236	3%	6%	49	593
47	98%	0	2	3%	243	6%	4%	45	580

Using the Erlang C formula, the software calculates the GoS for different staffing and phone line combinations. The highlighted line shows that 45 telephone agents will meet or exceed the target service level. In fact, this staff will be able to answer 95% of the calls within 23 seconds, with an average speed to answer of 5 seconds. As staff is likely the most expensive call center resource, a reduced staff of 44 CSRs can deliver a somewhat degraded service level, answering 93% of the calls in 30 seconds, or 95% of the calls in 23 seconds, and ASA of 7 seconds. Note that a modest reduction of headcount by 5% to 43 agents will have a profound effect on service levels, doubling the ASA.

In addition, the Erlang calculator shows the following parameters:

% Abandoned - the percentage of callers that are likely to abandon the call while waiting in the queue. This number is calculated based on Queue Time.

% Queued - the percentage of calls that will not be answered within the target ASA and will be queued.

Queue Time - the average time callers will spend in the queue while waiting to receive service.

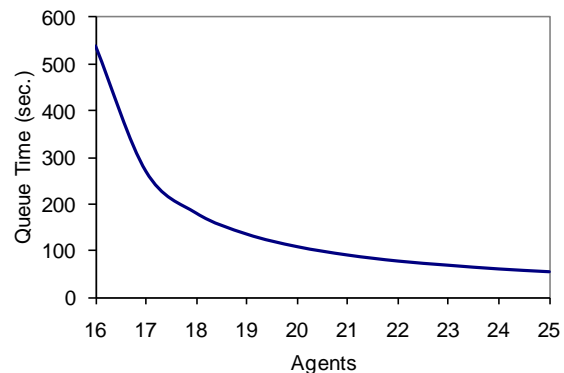
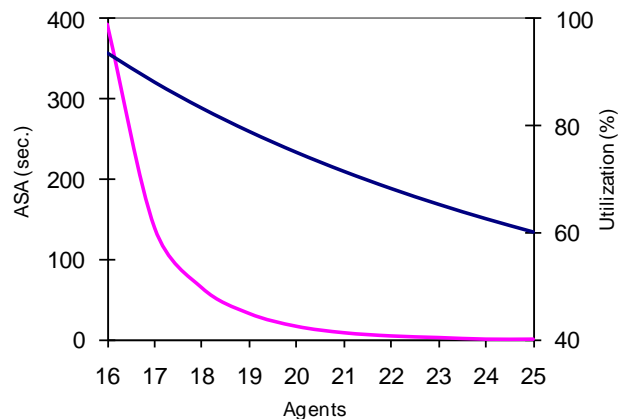
Non-Linearity

Earlier we saw that back-of-the-envelope methods do not work for call center planning because of the statistical distribution of call arrival patterns and call handling times.

The statistical behavior of these factors also indicates that changes in resource allocation will have a non-linear effect on staffing levels and GoS. For example, an increase of 10% in staffing will not result in 10% improvement in ASA or in caller wait time. We have already seen evidence of this effect in the example discussed earlier, in which a reduction of 5% in staff doubled the ASA.

The graph on the right shows the non-linear relationships between staffing level and ASA. The more agents the call center employs, the better average speed of answer it can provide, but the improvement is not linear and at some point adding agents has only a negligible impact on ASA. Moreover, as agents are added, their utilization, or the time they spend servicing customers, is decreasing.

We see the same non-linear relationships between staffing level and other key call center metrics. For example, the graph on the right depicts the impact on queue time, that is, the average time that callers who do not receive immediate service (within the ASA window.) Analysis of queue waiting times is important because it helps understand call abandonment.



Limitations of Erlang Models

Erlang queuing models originated in the telephony industry and were designed primarily to deal with physical switchboard resources, namely, telephone trunks. Therefore, the fundamental Erlang models make certain assumptions concerning the expectations and behavior of customers that are not always met in the real world. For example, the model assumes that callers will be willing to wait as long as it takes in order to speak with a service agent. In practice, of course, some callers will hang up the phone as soon as they are put on hold, and others will abandon the call after waiting in the queue for some time. Some callers will redial soon after they hang up, thinking it improves their odds. These human behavior patterns will change the actual call statistics and the performance of the call center overall.

In a similar fashion, the Erlang model treats servers as non-human resources. It assumes that they are always available and work at maximum capacity. While this is adequate for telephone lines, a reliable call center model must account for vacation and sick time as well as for training, meetings and other work, which may decrease utilization by 15% or more.

The standard Erlang model also assumes that the call center has unlimited queuing capacity. In practice, queuing resources are limited (by the number of available telephone lines) and when the system is overloaded, exceeding its queuing resources, callers will receive a busy signal or be connected to a voice-mail service. Automatic Call Distribution (ACD) systems can employ various strategies to lower the probability of this happening by overflowing calls to another agent group, or implementing a ring delay where the number of rings before the ACD picks up the line increases proportionally to the number of queued calls.

Various adaptations of the standard Erlang method exist to account for some of these issues, especially the infinite queuing problem. However, because of the complexity of the subject and the lack of a wide theoretical and practical base, these special versions should be used sensibly. In very large call centers, where approximations and rounding errors may result in significant numbers, simulation can offer a good substitute or a complementary analysis method.

Further Reading

- [The Origin of the 80/20 Rule](#)
- [Are Abandoned Calls Important?](#)
- [Service Level Calculations](#)
- [Advanced Topics in Call Center Staffing](#)
- [Introduction to Traffic Modeling and Resource Allocation in Call Centers](#)
- [Benchmarking in Call Centers](#)
- [Does Self-Help Really Help?](#)
- [Service Level Elasticity](#)
- [An Alternative to the Erlang Traffic Model](#)