

## Erlang C 计算公式

在呼叫中心摸爬滚打 4 年了，主要从事排班工作，近来发现很多同行跪求 Erlang C 公式，身边很多同事都不知 Erlang C 为何物。想想当年刚入行时自己搜遍 yahoo、google 也没搜索到多少国外有价值的资料，最多只是个 30 天试用版的软件，更别提国内的信息了，走了很多弯路，耗费了不少青春在资料收集上。

今天就把它作为我开博的第一篇文章，给大家简单地讲讲，如有不足还望海涵  
Erlang 全名：Agner Krarup Erlang（1878~1929），丹麦人，数学家，电气工程师



其发明的 Erlang B 和 Erlang C 公式 在金融、电信、运输、网络、呼叫中心等领域得到广泛运用

呼叫中心主要利用这个公式来计算满足服务水平目标所需要的人员数量以及中继线数量即通常所说的根据每小时电话量要求 20 秒达到 80% 的接通率需要多少人？

目前绝大部分排班软件都采用 Erlang C + Abandon Rate 模型，两大产品 Aspect 和 Blue Pumpkin 均以此为模型。

接下来进入正题：Erlang C 的公式

$$E_c(m, u) = \frac{\frac{u^m}{m!}}{\frac{u^m}{m!} + (1-\rho) \sum_{k=0}^{m-1} \frac{u^k}{k!}} :$$

第一反应是不是和我当初一样 “晕，虾米东西！” 呵呵！

这也算是当初网上能搜索到的最详尽资料了😂

别害怕，我可不会写到此就结束的 先从各个参数说起：

假设呼叫中心每半个小时进线量 360 通、平均处理时长 4 分钟、一共有 55 个客服人员、服务水平目标为 15 秒

### 第 1 个参数

$$\lambda = \text{average arrival rate} = \frac{360 \text{ calls / half hour}}{1800 \text{ seconds / half hour}} = 0.2 \text{ calls / second}$$

$\lambda$  = average arrival rate （来电频率/密度）= 360 通/半小时 ÷ 1800 秒 = 0.2 通/秒

### 第 2 个参数

$$T_s = \text{average call duration} = 240 \text{ seconds} = 4 \text{ minutes}$$

$T_s$  = average call duration （平均每通电话时长）= 240 秒/通



### 第 3 个参数

$$m = \text{number of agents} = 55 \text{ agents}$$

m=numbers of agents (坐席数) =55 人

### 第 4 个参数

$$u = \lambda \cdot T_s = \text{traffic intensity} = 48$$
$$= (0.2 \text{ calls / sec}) \cdot (240 \text{ secs / call})$$

traffic intensify(话务强度)=  $\lambda \times TS = 0.2 \text{ 通/秒} \times 240 \text{ 秒/通} = 48 \text{ 秒/秒}$

我理解为每秒需要处理 48 秒的工作量，即每秒需要 48 个人

换另一种工作量算法可能大家更容易懂，即 360 通电话，每通 240 秒，那么处理这些电话共需 86400 秒，而每个座席员每半小时有 1800 秒，在最理想化的状态下我们也需要  $360 \times 240 \div 1800 = 48$  人

### 第 5 个参数

$$\rho = \frac{u}{m} = \text{agent occupancy} = \frac{48}{55} = 0.873$$
$$\text{or } (0.873) \cdot 100\% = 87.3\%$$

P= agent occupancy (占用率) = 48 人  $\div$  55 人 =87.3%

好了，接下来就开始代入 Erlang C 公式

$$E_c(m, u) = \frac{\frac{u^m}{m!}}{\frac{u^m}{m!} + (1-\rho) \sum_{k=0}^{m-1} \frac{u^k}{k!}} = 0.239$$

$m!$ 即  $m$  的阶乘，这里即  $1*2*3*4*.....*54*55$  Excel 中可用 `=fact(55)` 计算

$$\sum_{k=0}^{m-1} \frac{u^k}{k!}$$

这个呐就是加总从  $K=0$  开始一直算到  $k=m-1$  为止，这里即算到  $k=54$  为止因此得出最终结果 0.239

当然死算也是可以算出来的，可是我们有这个世纪最强大便捷的工具 Excel，作为跨时代的接班人就要好好掌握利用（不是广告，呵呵）

因为 Erlang 方程是由泊松方程(poisson)推出来的, excel 中也有该函数什么？泊松方程(poisson)是什么？这个就要从概率讲起，在此略过这个强大的过程，各位勤奋好学的自己搜索，哈！😎

那么该公式  $E_c(m, u)$  在 Excel 中可表达

为 `=poisson(m, u, false)/(poisson(m, u, false)+(1-P)*poisson(m-1, u, true))`

$$= 0.239$$

$$\text{Prob(call has to wait)} = E_c(m, u) \text{ or } 0.239 \times 100\% = 23.9\%$$

prob(call has to wait)呼叫等待的概率=23.9%



$$\begin{aligned}
T_w &= \text{average waiting time} \\
&= \text{ASA} \\
&= \frac{E_c(m, u) \cdot T_s}{m \cdot (1 - \rho)} = \frac{0.239 \times 240}{55 \times (1 - 0.873)} \\
&= 8.2 \text{ seconds}
\end{aligned}$$

$T_w$ =average waiting time 平均等待时长=Average Speed of Answer 平均应答速度 (ASA) =8.2 秒

$$\begin{aligned}
t &= \text{target answer time} & t &= 15 \text{ seconds} \\
W(t) &= \text{Prob( waiting time } \leq t) & W(t) &= 1 - 0.239 \times e^{-(55-48) \times \frac{15}{240}} \\
&= 1 - E_c(m, u) \cdot e^{-\frac{(m-u)t}{T_s}} & &= 1 - 0.239 \times e^{-0.4375} \\
& & &= 0.846 \text{ or } 84.6\%
\end{aligned}$$

$t$ =target answer time (服务水平目标) =15 秒 (这里为 15 秒)

$e$  为自然数的底数 即 2.718281828

在 Excel 中可用 **=exp(1)** 计算 Excel 中次方用 **^** 计算，这里 excel 公式即 **=1-0.239\*exp(1)^(-0.4375)**

$W(t)$  即该服务水平目标下可能的服务水平=84.6%

讲到这里大家应该掌握的差不多了吧！自己也能写进 Excel 运用了吧！希望对大家能够有帮助！

最后感谢 Mike Tanner，是他的文章让我了解了 Erlang C 并充分运用到工作实践中