Service Engineering – a Subjective View

- Contrast with the traditional and prevalent

  Service Management (Business Schools; U.S.A.)

  Industrial Engineering (Engineering Schools; Europe)

- Goal: Develop scientifically-based design principles (rules-of-thumb) and tools (software) that support the balance of service quality, process efficiency and business profitability, from the (often conflicting) views of customers, servers and managers.

  - Theoretical Framework: Queueing Networks
  - Applications focus: Call (Contact) Centers

Example: Staffing - How many agents required for balancing service-quality with operational-efficiency.

Example: Skills-Based Routing (SBR) – Platinum and Gold and Silver customers, all seeking Support or Purchase, via the Telephone or IVR or e.mail or Chat.

Example: Service Process Design + Staffing + SBR

Multi-Disciplinary: Typical (IE/OR, Marketing, CS, HRM)

Workforce Management: Hierarchical Operational View

Forecasting Customers: Statistics, Time-Series
Agents : HRM (Hire, Train; Incentives, Careers)

Staffing: Queueing Theory

Service Level, Costs

FTE's (Seats) per unit of time

Shifts: IP, Combinatorial Optimization; LP

Union constraints, Costs

Shift structure

Rostering: Heuristics, AI (Complex)

Individual constraints

Agents Assignments

Skills-based Routing: Stochastic Control

4CallCenters™

Personal Optimization Tools for Call Centers

Downloads:

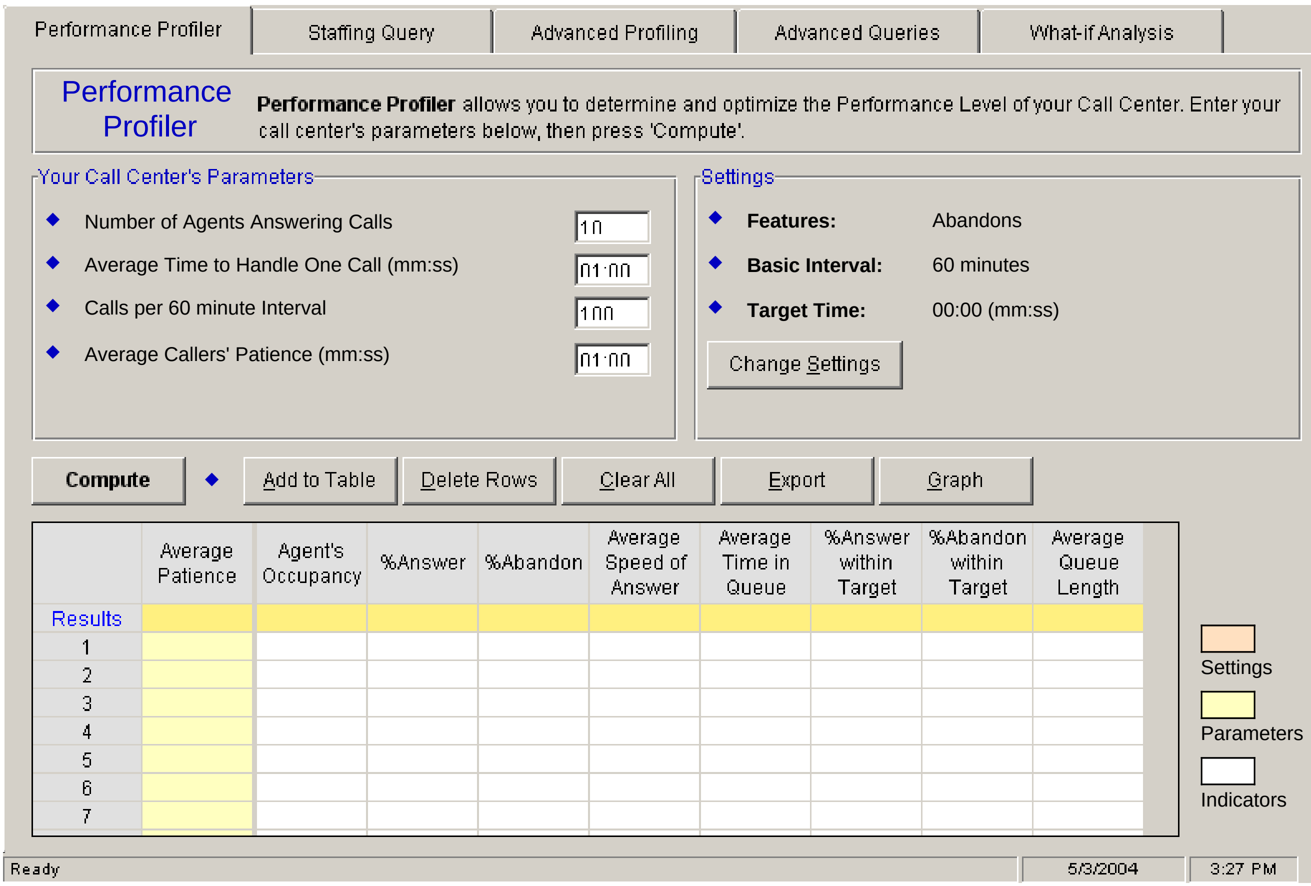
1. 4CallCenters v2.01(zip file- **5.4mb**)
   Desktop application offering personal profiling and optimization tools.

   - **For installation: Download the zip file, open it, activate setup.exe and follow the instructions.**

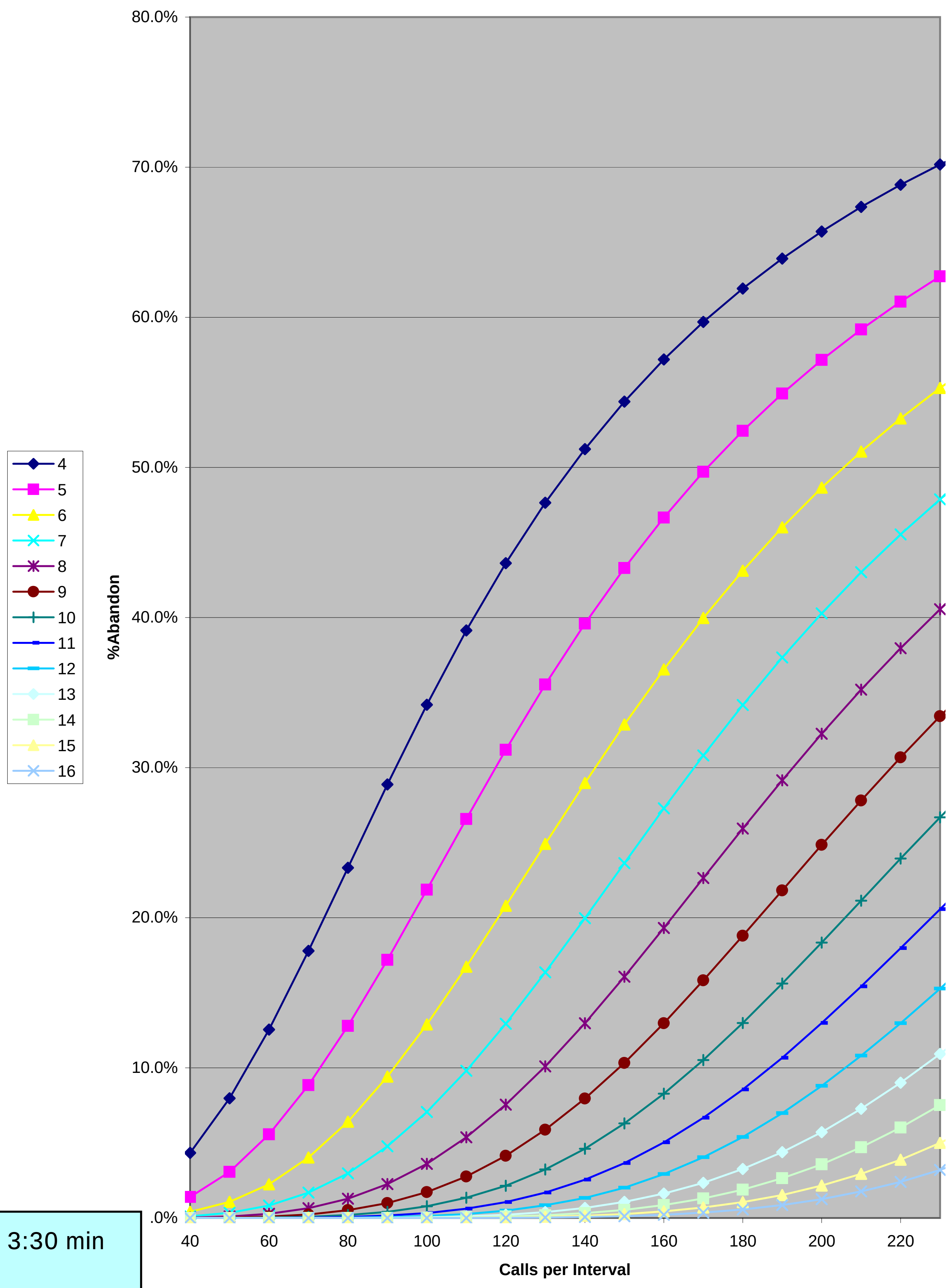
   - **To uninstall the installed software: Go to Start/Programs/4CallCenters v2.01/Uninstall 4CallCenters v2.01**

2. 4CallCenters v2.01 - Help Document (90**kb**)
   Word document containing the 4CallCenters application's help pages.

QSetup



%Abandon vs. Calls per Interval for various Number of Agents



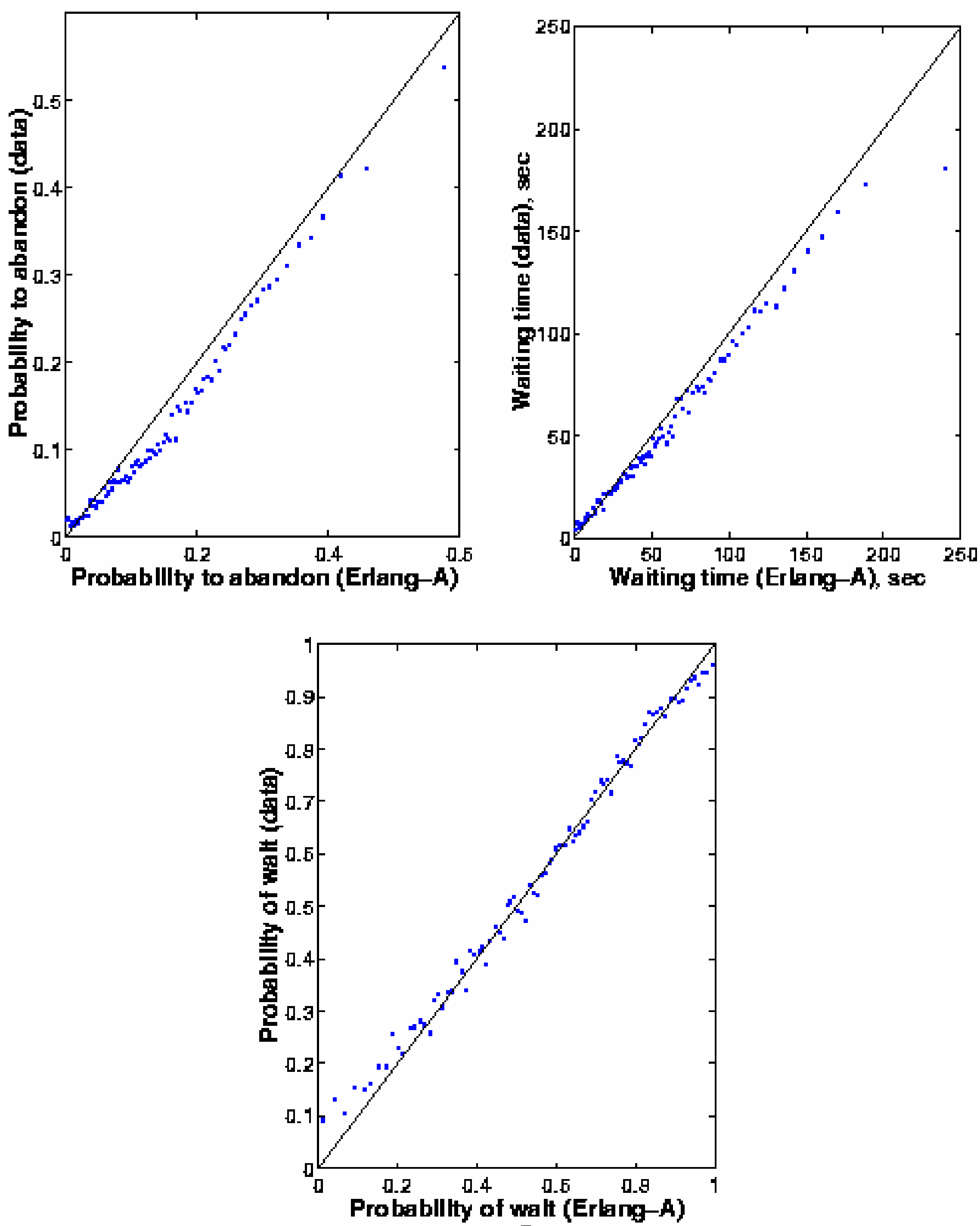
E(S) = 3:30 min

E(R) = 6:00 min

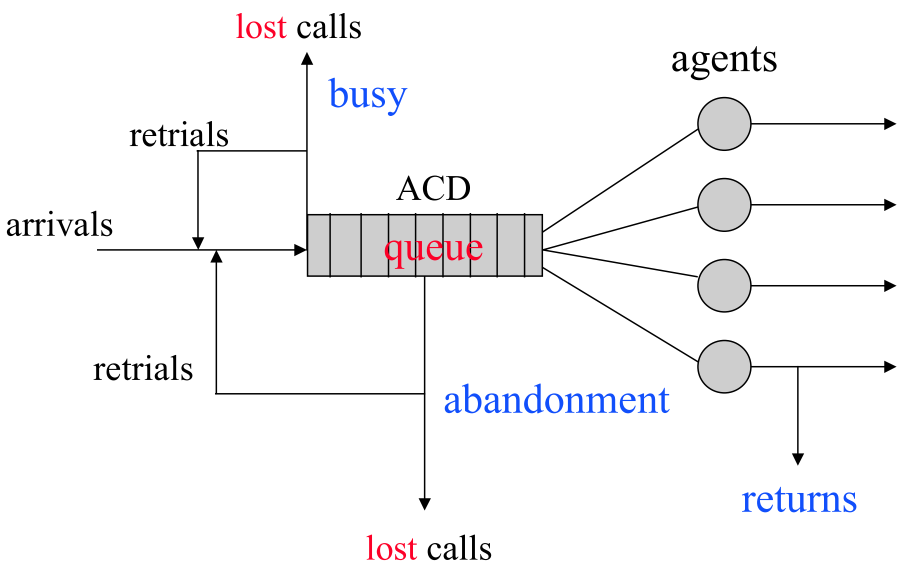
Interval = 1 **hour**

Fitting a Simple Model to a Complex Reality

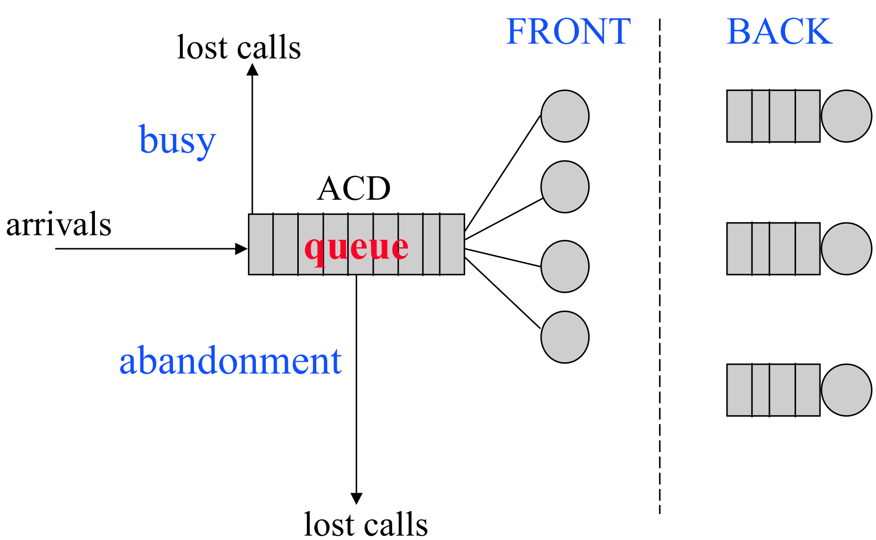
Erlang-A Formulae vs. Data Averages



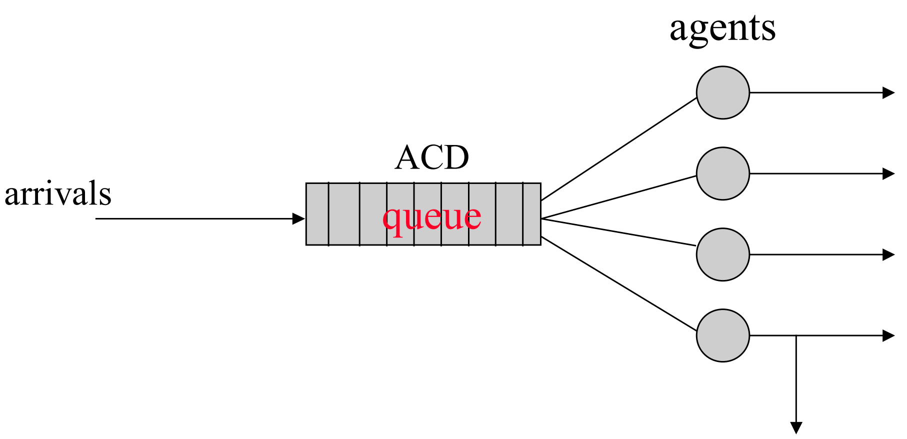
A Basic Call Center



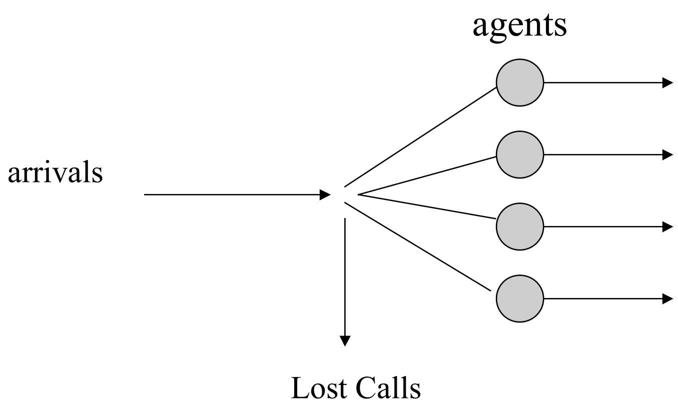
4CallCenters (Erlang-A)



Erlang-C



Erlang-B



Review: Markov Jump-Processes (MJP)

MJP $X = \{X_t,\ t \geq 0\}$ on $\mathcal{S} = \{i, j, \ldots\}$ countable.
Markov property: $P_r\{X_t = j | X_r,\ r < s;\ X_s = i\} = P_{ij}(s,t), \quad \forall\, s < t,\ \forall\, i, j \in \mathcal{S}$.
Time homogeneity: $P_r\{X_{s+t} = j | X_s = i\} = P_{ij}(t), \quad \forall\, s, t,\ i, j,$ transition probabilities.

Characterization: π^0 = initial distribution and $P(t) = [P_{ij}(t)],\ t \geq 0$, stochastic.
Finite-dimensional distributions:
$P_r\{X_0 = i_0,\ X_{t_1} = i_1, \ldots, X_{t_n} = i_n\} = \pi^0(i_0) P_{i_0, i_1}(t_1) \ldots P_{i_{n-1}, i_n}(t_n - t_{n-1})$.

$P(t)$: stochastic ; $P(s+t) = P(s)P(t), \quad \forall\, s, t$ (Chapman Kolmogorov);
$\exists\, P(0) = I$; $\exists\, \dot{P}(0) = Q = [q_{ij}]$, infinitesimal generator $\left(\sum_{j \in \mathcal{S}} q_{ij} = 0\right)$.

Micro to Macro : $\dot{P}(t) = P(t)Q \quad (= QP(t))$ and $P(0) = I$
Forward (Backward) equations.

Solution : $P(t) = \exp[tQ] = \sum_{n=0}^{\infty} \frac{t^n}{n!}\, Q^n\ ,\ t \geq 0$.

Animation: $i \xrightarrow{q_{ij}} j; \quad \forall\, i, j \in \mathcal{S}\ \exists$ exponential clock at rate q_{ij}, call it (i, j).
Given i, consider clocks $(i, j),\ j \in \mathcal{S}$; move to the "winner" when rings.
Thus: stay at $i \sim \exp(q_i = \sum_{j \neq i} q_{ij})$ and switch to j with probability $P_{ij} = q_{ij}/q_i$
$(q_{ij} = q_i P_{ij}, i \neq j; q_{ii} = -q_i)$.

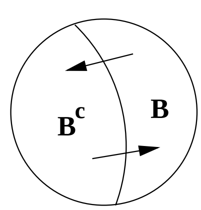
Transient analysis	vs. long-run/limit	stability/steady-state
	$\exists\, \lim_{t \uparrow \infty} P_{ij}(t) = \pi_j,\ \forall\, i;$	$\pi = \pi P(t),\ \forall\, t.$

Calculation via **steady-state equations:** $\dot{P}(\infty) = P(\infty)Q \Rightarrow \left\{ \begin{array}{l} 0 = \pi Q \\ \sum_i \pi_i = 1,\ \pi_i \geq 0 \end{array} \right\}$

or balance equations: $\sum_{i \neq j} \pi_i q_{ij} = -\pi_j q_{jj} = \sum_{i \neq j} \pi_j q_{ji},\ \forall\, j$.

Transition rates: $\pi_i q_{ij}$ = long-run average number of switches from i to j.

Cuts: $\sum_{i \in B} \sum_{j \in B^c} \pi_i q_{ij} = \sum_{i \in B^c} \sum_{j \in B} \pi_i q_{ij},\ \forall\, B \subset \mathcal{S}$.



Ergodic Theorem: Let X be *irreducible* ($i \leftrightarrow j$). Assume that there exists a solution π to its steady-state equations. Then, X must be "unexplosive" and π must be its stationary distribution, its limit distribution and

SLLN $\bullet \lim_{T\uparrow\infty} \frac{1}{T} \int_0^T f(X_t)dt = \sum_i \pi_i f(i)$ ("$=$" $Ef(X_\infty)$) ; eg. $f(x) = 1_B(x)$.

$\bullet \lim_{T\uparrow\infty} \frac{1}{T} \sum_{t\le T} g(X_{t-}, X_t) = \sum_i \pi_i \sum_j q_{ij} g(i,j)$, for $g(x,x) = 0$, $\forall x$; e.g. $g(x,y) = 1_C(x,y)$.

Birth-and-death process: MJP on $S = \{0, 1, 2, \ldots\}$, where all jumps are between adjacent states: $q_{ij} = 0$ if $|i - j| > 1$.

Cuts: $\pi_i q_{i,i+1} = \pi_{i+1} q_{i+1,i}$.
(Take $B = \{0, 1, \ldots, i\}$ and $B^c = \{i+1, i+2, \ldots\}$.)

Reversibility: A stochastic process $X = \{X_t, \ -\infty < t < \infty\}$ is called *reversible* if for any τ

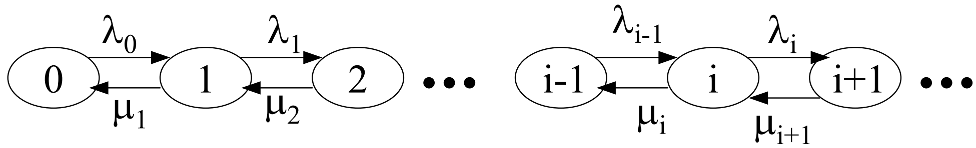
$$\{X_t, \ -\infty < t < \infty\} \stackrel{d}{=} \{X_{\tau - t}, \ -\infty < t < \infty\} .$$

Fact. Ergodic MJP in steady-state is reversible if and only if the *detailed balance equations* hold:

$$\pi_i q_{ij} \ = \ \pi_j q_{ji} \, , \qquad \forall\, i, j \in \mathcal{S} \, .$$

Corollary. Every ergodic birth-and-death process is reversible.
(Follows from the cut equations.)

Birth & Death Model of a Service Station



- i – number-in-system;
- λ_i – arrival rate given i customers in system;
- μ_i – service rate given i customers in system.

Cuts at $i \leftrightarrow i+1$ yield:

$\pi_i \lambda_i = \pi_{i+1}\mu_{i+1}, \ i \geq 0$, and

$$\pi_{i+1} = \frac{\lambda_i}{\mu_{i+1}} \pi_i = \frac{\lambda_i \lambda_{i-1}}{\mu_{i+1}\mu_i} \pi_{i-1} = \cdots = \frac{\lambda_0 \lambda_1 \ldots \lambda_i}{\mu_1 \mu_2 \ldots \mu_{i+1}} \pi_0 \ .$$

Steady-state distribution exists iff

$$\sum_{i=0}^{\infty} \frac{\lambda_0 \ldots \lambda_i}{\mu_1 \ldots \mu_{i+1}} < \infty \ .$$

Then

$$\begin{cases} \pi_i &= \frac{\lambda_0 \ldots \lambda_{i-1}}{\mu_1 \ldots \mu_i} \pi_0 \ , \ i \geq 0 \\ \pi_0 &= \left[\sum_{i \geq 0} \frac{\lambda_0 \ldots \lambda_i}{\mu_1 \ldots \mu_{i+1}} \right]^{-1} \end{cases}$$

$$\text{Arrival rate} \ = \ \sum_{i=0}^{\infty} \pi_i \lambda_i = \sum_{i=1}^{\infty} \pi_i \mu_i \ = \ \text{Departure rate.}$$

Additional assumptions (classical queues):

- n statistically identical servers;
- FCFS discipline – First Come First Served;
- Work conservation: a server does not go idle if there are customers in need of service;
- Customers do not abandon.

Measures of Performance

- L - number of customers at the service station (sometimes L_s);
- L_q - number of customers in the queue;
- W - sojourn time of a customer at the service station (W_s);
- W_q - waiting time of a customer in the queue.

In steady state (in the long run):

$$\mathrm{E}[L] \;=\; \sum_{k\geq 0} k\cdot\pi_k \;=\; \lim_{T\to\infty}\frac{1}{T}\cdot\int_0^T L(t)dt\,.$$

$$\mathrm{E}[L_q] = \sum_{k=n+1}^{\infty}(k-n)\cdot\pi_k\,.$$

If λ – arrival rate to the system, Little's formula implies:

$$\mathrm{E}[L] = \lambda\cdot\mathrm{E}[W]; \qquad \mathrm{E}[L_q] = \lambda\cdot\mathrm{E}[W_q]\,.$$

4CallCenters Software.

Mathematical engine based on the M.Sc. thesis of Ofer Garnett (in References)

Will be taught and used in our course.

Install at

`http://iew3.technion.ac.il/serveng/4CallCenters/Downloads.htm`

4CallCenters™

Personal Optimization Tools for Call Centers

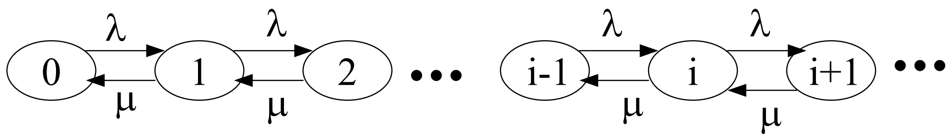
Downloads:

1. 4CallCenters v2.23(setup.exe file- **3 MB**)
   - **For installation: Open setup.exe and follow the instructions.**
   - **To uninstall the installed software: Go to Start/Programs/4CallCenters v2.23/Uninstall 4CallCenters v2.23**
2. 4CallCenters v2.01 - Help Document (**100 KB**)
   Word document containing the 4CallCenters application's help pages.

QSetup **We are grateful to QSetup for their support.**

M/M/1 queue

- Poisson arrivals, rate λ;
- Single exponential server, rate μ; $\mathrm{E}[S] = 1/\mu$.



$\lambda_i = \lambda,\ i \geq 0; \quad \mu_i = \mu \cdot 1_{i \geq 1}$.

Cut equations: $\lambda \pi_i = \mu \pi_{i+1}\,, \qquad i \geq 0\,.$
Traffic intensity $\rho = \frac{\lambda}{\mu} < 1$ (assumed for stability).

Steady-state distribution $L \overset{d}{=} Geom(\mathrm{p} = 1\text{-}\rho)$ (from 0):

$$\pi_i = (1-\rho)\rho^i,\ i \geq 0.$$

Properties:

- Sojourn time is exponentially distributed:

$$W \sim \exp\left(\text{mean} = \frac{1}{\mu(1-\rho)} = \frac{1}{\mu}\left[1 + \frac{\rho}{1-\rho}\right]\right).$$

Proof: Via moment generating functions.

According to PASTA, with $N = L + 1$,

$$W \overset{d}{=} \sum_{i=1}^{N} X_i\,, \quad X_i \sim \exp(\mu) \text{ i.i.d.}\,, \quad N \overset{d}{=} \mathrm{Geom}(1-\rho) \text{ (from 1)};$$

N and $\{X_i\}$ are all independent.

Moment generating function:

$$
\begin{aligned}
\phi_W(t) &\stackrel{\Delta}{=} \mathrm{E}\left[\exp\{tW\}\right] \;=\; \mathrm{E}\left[\exp\{t\cdot\sum_{i=1}^{N} X_i\}\right] \\
&= \mathrm{E}\left[\mathrm{E}\left[\exp\{t\cdot\sum_{i=1}^{N} X_i\}\middle|\, N\right]\right] \\
&= \text{(Moment generating function of Erlang r.v.)} \\
&= \mathrm{E}\left[\left(\frac{\mu}{\mu-t}\right)^{N}\right] \;=\; \sum_{k=1}^{\infty}(1-\rho)\rho^{k-1}\left(\frac{\mu}{\mu-t}\right)^{k} \\
&= \frac{\mu(1-\rho)}{\mu-t}\cdot\sum_{k=0}^{\infty}\left(\frac{\mu\rho}{\mu-t}\right)^{k} \;=\; \frac{\mu(1-\rho)}{\mu(1-\rho)-t} \\
&= \phi_{\exp(\mu(1-\rho))}(t)\,.
\end{aligned}
$$

- Delay probability (PASTA): $\mathrm{P}\{W_q > 0\} = \rho$.
- Waiting time in queue, given delay, is exp:

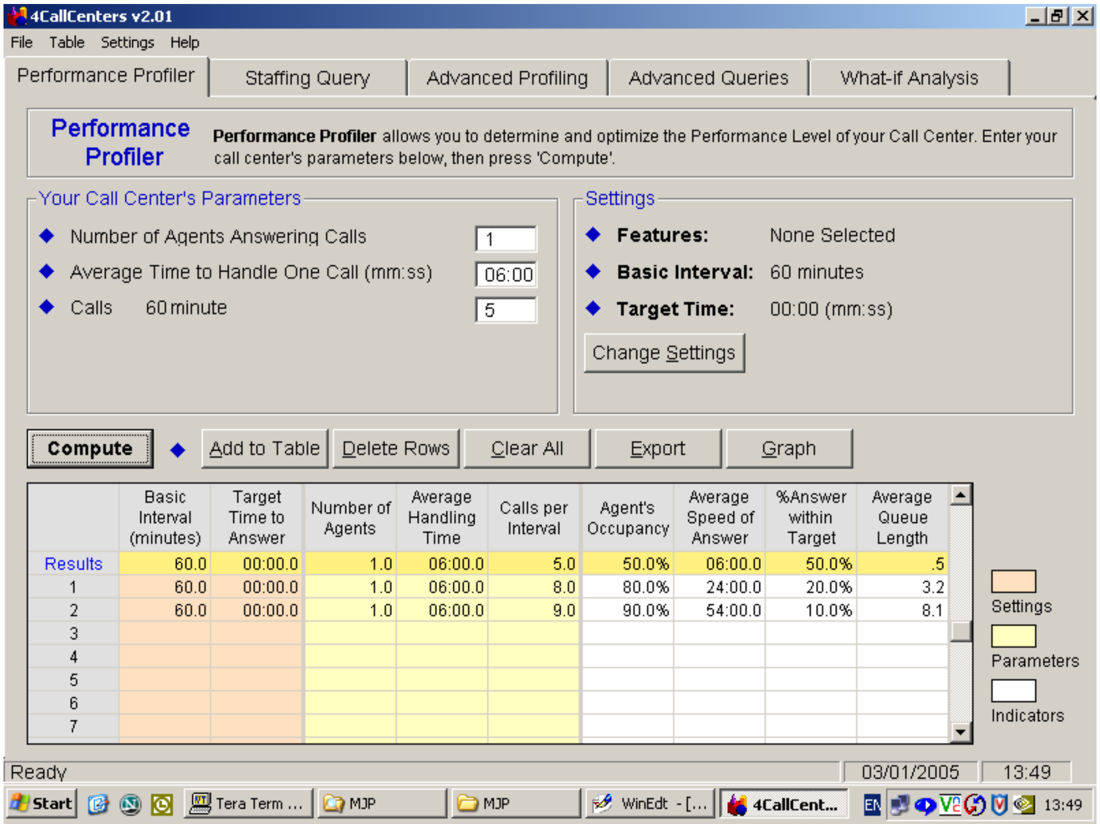
$$
\frac{W_q}{1/\mu} \stackrel{d}{=} \begin{cases} 0 & \text{wp } \; 1-\rho \\ \exp\left(\text{mean} = \frac{1}{1-\rho}\right) & \text{wp } \; \rho \end{cases}
$$

- Number-in-system: $\mathrm{E}[L] = \frac{\rho}{1-\rho}$; $\quad \mathrm{E}[L_q] = \frac{\rho^2}{1-\rho}$.
- Server's utilization (occupancy) is $\rho = \lambda/\mu$.
  (Little's formula, system = server.)
- Departure process in steady state is Poisson (λ)
  (Burke theorem) – important in queueing networks.

Partial support : average inter-departure =

$$
\frac{1}{\mu}\cdot\rho + \left(\frac{1}{\mu}+\frac{1}{\lambda}\right)\cdot(1-\rho) = \frac{1}{\lambda}.
$$

M/M/1. 4CallCenters output



Note large waiting times:

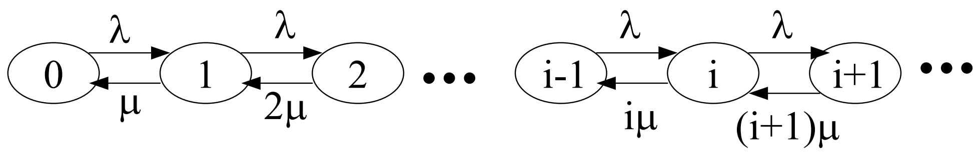
$\mathrm{E}[S]$ for $\rho = 50\%$, $\quad 9 \cdot \mathrm{E}[S]$ for $\rho = 90\%$, $\quad 19 \cdot \mathrm{E}[S]$ for $\rho = 95\%$.

4CallCenters: performance measures.

- Average Speed of Answer = $\mathrm{E}[W_q]$
(will be different in queues with abandonment);
- %Answer within Target = $\mathrm{P}\{W_q < T\}$;
- Average Queue Length = $\mathrm{E}[L_q]$.

M/M/∞ queue

- Poisson arrivals, rate λ;
- Infinite number of exponential servers, rate μ.



$\lambda_i = \lambda, \quad i \geq 0; \qquad \mu_i = i \cdot \mu, \quad i > 0.$

Cut equations:

$$\lambda \pi_i = (i+1) \cdot \mu \pi_{i+1}\,, \qquad i \geq 0\,.$$

Always stable.

Steady-state distribution is Poisson:

$$\pi_i = e^{-R} \cdot \frac{R^i}{i!}\,,\; i \geq 0\,,$$

where $R = \dfrac{\lambda}{\mu} = \lambda \cdot E(S)$ is the **offered load** (measured in Erlangs).

$$\mathrm{E}[L] \;=\; \mathrm{E}(\#\text{ busy servers}) \;=\; \lambda \cdot \frac{1}{\mu} \;=\; R\,.$$

(Little's formula, system = service.)

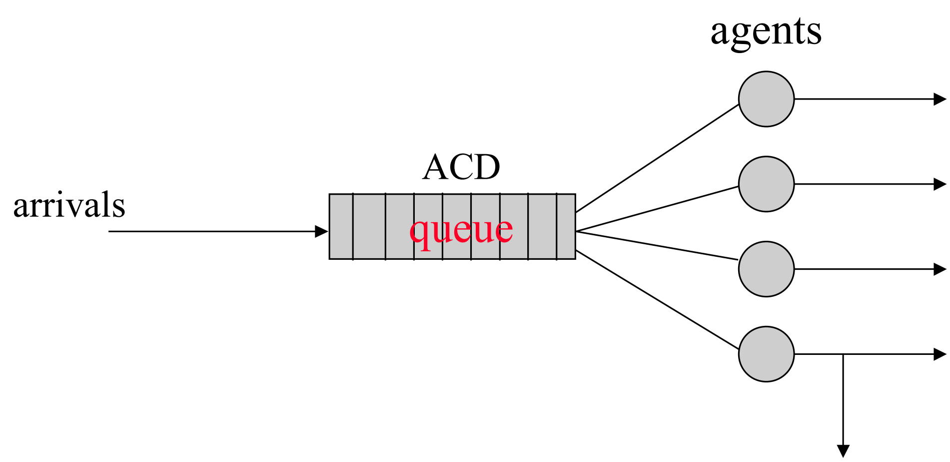
Very *useful*: ∞-server models provide bounds.

Results above valid for M/G/∞ – generally distributed service times. (Insensitivity to the service-time distribution.)

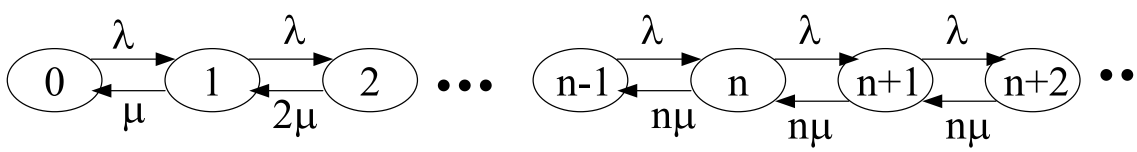
M/M/n (Erlang-C) queue

- Poisson arrivals, rate λ;
- n exponential servers, rate μ.

Widely used in call centers.



Transition-rate diagram



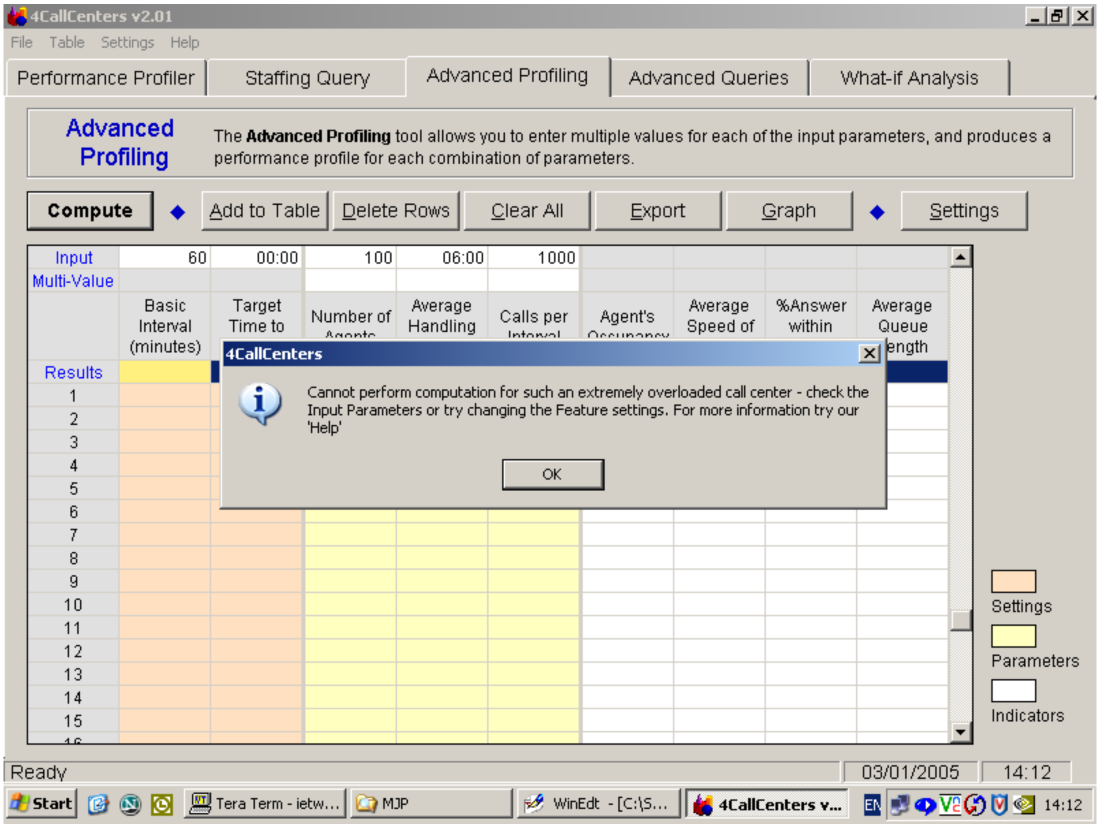
$$\lambda_j = \lambda, \qquad j \geq 0,$$
$$\mu_j = (j \wedge n)\mu, \quad j \geq 1.$$

Agents' utilization

$$\rho = \frac{\lambda}{n\mu}.$$

Assume $\rho < 1$ ($R < n$) to ensure stability (as in M/M/1).

4CallCenters output: Instability, $\rho \geq 1$



Steady-state distribution:

$$
\begin{aligned}
\pi_i &= \frac{R^i}{i!}\,\pi_0, & i \leq n,\\
&= \frac{n^n \rho^i}{n!}\,\pi_0, & i \geq n,\\
\pi_0 &= \left[\sum_{j=0}^{n-1} \frac{R^j}{j!} + \frac{R^n}{n!(1-\rho)}\right]^{-1},
\end{aligned}
$$

where $R = \dfrac{\lambda}{\mu}$ is the **offered load**.

Erlang-C Formula (1917):
Delay probability:

$$\mathrm{P}\{W_q > 0\} \;\stackrel{\Delta}{=}\; E_{2,n} = \sum_{i\geq n} \pi_i = \frac{R^n}{n!}\,\frac{1}{1-\rho}\,\cdot \pi_0 \,.$$

Erlang-C computation: recursion, see Erlang-B below.

Number-in-queue:

$$\mathrm{P}\{L_q = i\} \;=\; E_{2,n} \cdot (1-\rho)\rho^i \,, \qquad i > 0,$$

or

$$L_q \;=\; \begin{cases} 0 & \text{wp } \; 1 - E_{2,n} \\ \mathrm{Geom}(1-\rho) & \text{wp } \; E_{2,n} \end{cases}$$

Waiting time distribution:

$$\frac{W_q}{1/\mu} \;=\; \begin{cases} 0 & \text{wp } \; 1 - E_{2,n} \\ \exp\left(\text{mean} = \frac{1}{n} \cdot \frac{1}{1-\rho}\right) & \text{wp } \; E_{2,n} \end{cases}$$

Compare with M/M/1!

Departure process: Poisson(λ) in steady-state.
Proof via reversibility, as with M/M/1.

M/M/n derivation of waiting-time distribution

Via the "M/M/1 - analogy",

$$\frac{1}{E(S)} W_q \mid W_q > 0 \stackrel{d}{=} exp(n(1-\rho))$$

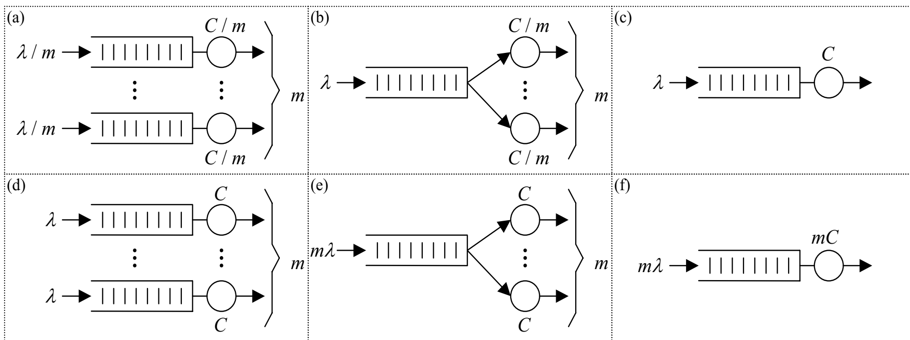
$$P\{\frac{1}{E(S)} W_q > t \mid W_q > 0\} = e^{-n(1-\rho)t}.$$

Formally:

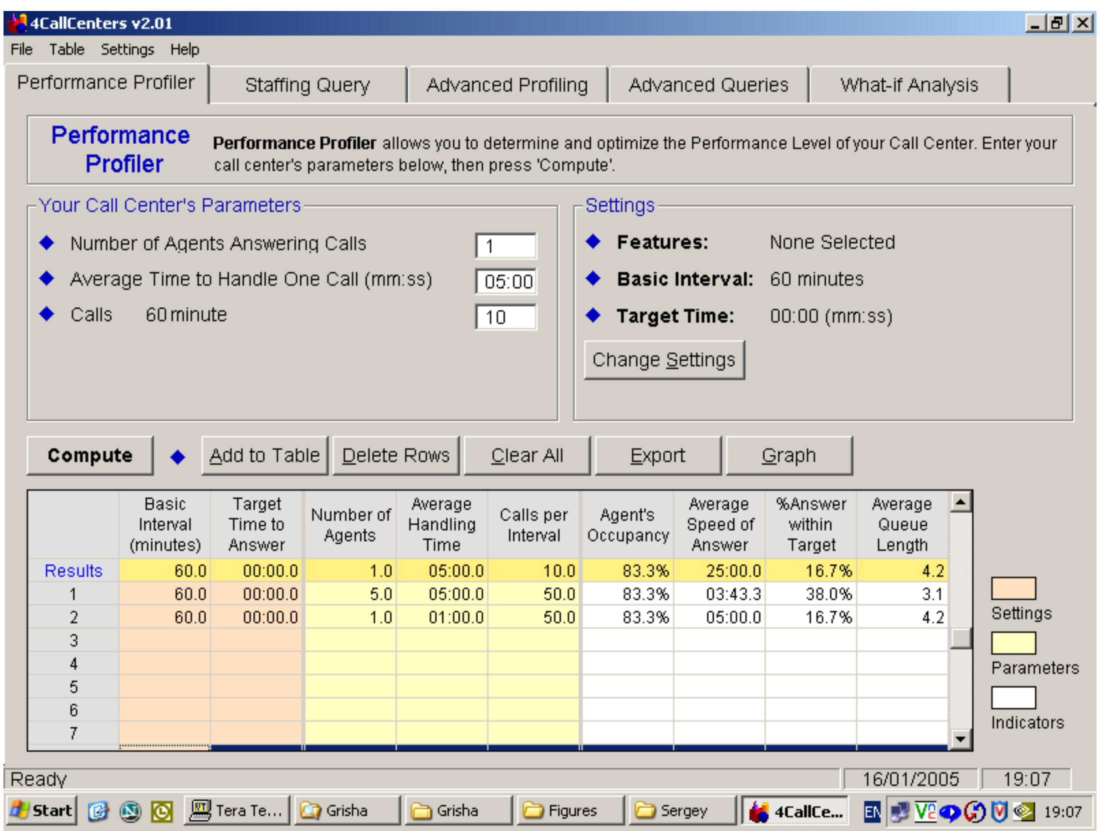
$$\begin{aligned}
P\{W_q > t\} &= \sum_{k=1}^{\infty} P\{L_q = k-1\} \cdot P\{E_k > t\} \\
&\quad \text{(where } E_k \sim \text{Erlang}(k, n\mu)) \\
&= E_{2,n} \cdot \sum_{k=1}^{\infty} \left[(1-\rho)\rho^{k-1} \cdot \int_t^{\infty} \frac{n\mu(n\mu x)^{k-1}}{(k-1)!} e^{-n\mu x} dx \right] \\
&= E_{2,n} \cdot n\mu(1-\rho) \cdot \int_t^{\infty} \left(e^{-n\mu x} \sum_{k=1}^{\infty} \frac{(n\mu\rho x)^{k-1}}{(k-1)!} \right) dx \\
&= E_{2,n} \cdot n\mu(1-\rho) \cdot \int_t^{\infty} e^{-n\mu(1-\rho)x} dx \\
&= E_{2,n} \cdot e^{-n\mu(1-\rho)t}.
\end{aligned}$$

Pooling; Economies of Scale

Example: Kleinrock, L. Vol.II, Chapter 5 (1976)



4CallCenters output



	1		**2**		**3**
	$n \times \text{M/M/1}$	$\xrightarrow{\text{pooling}}$	$\text{M/M/}n$	$\xrightarrow{\text{technology}}$	M/M/1
	λ, μ		$n\lambda, \mu$		$n\lambda, n\mu$
$\mathrm{P}\{W_q > 0\}$	ρ		$E_{2,n}$		ρ
$\mathrm{E}[W_q]$	$\frac{1}{\mu} \cdot \frac{\rho}{1-\rho}$		$\frac{1}{\mu} \cdot \frac{E_{2,n}}{n(1-\rho)}$		$\frac{1}{n\mu} \cdot \frac{\rho}{1-\rho}$
$\mathrm{E}[S]$	$\frac{1}{\mu}$		$\frac{1}{\mu}$		$\frac{1}{n\mu}$
$\mathrm{E}[W]$	$\frac{1}{\mu} \cdot \frac{1}{1-\rho}$		$\frac{1}{\mu} \cdot \left[\frac{E_{2,n}}{n(1-\rho)} + 1\right]$		$\frac{1}{n\mu} \cdot \frac{1}{1-\rho}$

(0)

Statement: $1-\rho \; < \; 1-E_{2,n} \; < \; n(1-\rho)\,.$

Proof: Consider M/M/n.

$$1-\rho \;=\; \mathrm{P}\{\text{server } i \text{ idle}\}, \quad \text{for} \;\; i = 1, \ldots, n\,;$$

$$1-E_{2,n} \;=\; \mathrm{P}\{\text{at least one server idle}\} \;=\; \mathrm{P}\left\{\bigcup_{i=1}^{n} \{i \text{ idle}\}\right\}$$

$$n(1-\rho) \;=\; \sum_{i=1}^{n} \mathrm{P}\{\text{server } i \text{ idle}\}$$

Conclusions

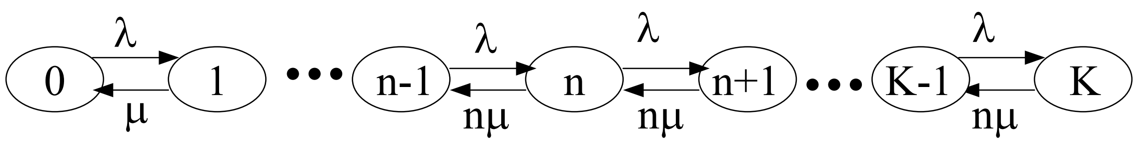
1 → 2 : Pooling yields $\mathrm{E}[W_q]$ decrease by more than factor n;

1 → 3 : Fast server yields $\mathrm{E}[W]$ and $\mathrm{E}[W_q]$ decrease by factor n;

2 → 3 : Fast server better for $\mathrm{E}[W]$;
Pooling better for $\mathrm{E}[W_q]$.

M/M/n/K queue

- Poisson arrivals, rate λ;
- n exponential servers, rate μ;
- K trunks ($K \geq n$);
- If all trunks busy, arriving customer blocked (busy signal).

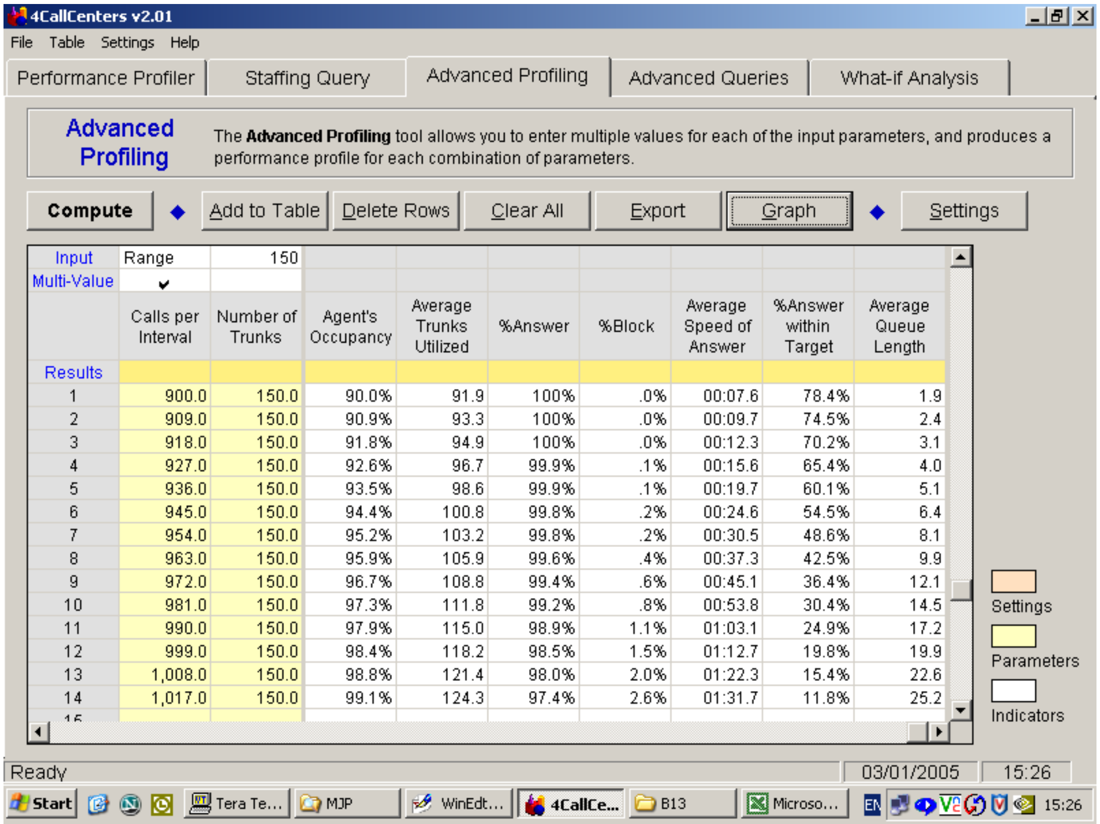


$$\lambda_j = \lambda, \qquad 0 \leq j \leq K-1,$$

$$\mu_j = (j \wedge n)\mu, \quad 1 \leq j \leq K.$$

Formulae straightforward but cumbersome (simply truncate M/M/n). Always reaches steady state.

4CallCenters output.



Results	Calls per Interval	Number of Trunks	Agent's Occupancy	Average Trunks Utilized	%Answer	%Block	Average Speed of Answer	%Answer within Target	Average Queue Length
1	900.0	150.0	90.0%	91.9	100%	.0%	00:07.6	78.4%	1.9
2	909.0	150.0	90.9%	93.3	100%	.0%	00:09.7	74.5%	2.4
3	918.0	150.0	91.8%	94.9	100%	.0%	00:12.3	70.2%	3.1
4	927.0	150.0	92.6%	96.7	99.9%	.1%	00:15.6	65.4%	4.0
5	936.0	150.0	93.5%	98.6	99.9%	.1%	00:19.7	60.1%	5.1
6	945.0	150.0	94.4%	100.8	99.8%	.2%	00:24.6	54.5%	6.4
7	954.0	150.0	95.2%	103.2	99.8%	.2%	00:30.5	48.6%	8.1
8	963.0	150.0	95.9%	105.9	99.6%	.4%	00:37.3	42.5%	9.9
9	972.0	150.0	96.7%	108.8	99.4%	.6%	00:45.1	36.4%	12.1
10	981.0	150.0	97.3%	111.8	99.2%	.8%	00:53.8	30.4%	14.5
11	990.0	150.0	97.9%	115.0	98.9%	1.1%	01:03.1	24.9%	17.2
12	999.0	150.0	98.4%	118.2	98.5%	1.5%	01:12.7	19.8%	19.9
13	1,008.0	150.0	98.8%	121.4	98.0%	2.0%	01:22.3	15.4%	22.6
14	1,017.0	150.0	99.1%	124.3	97.4%	2.6%	01:31.7	11.8%	25.2

Use `Change Settings` ⟹ `Features` ⟹ `Trunks`.

Note new indicators:
`Average Trunks Utilized` and `%Blocked`.

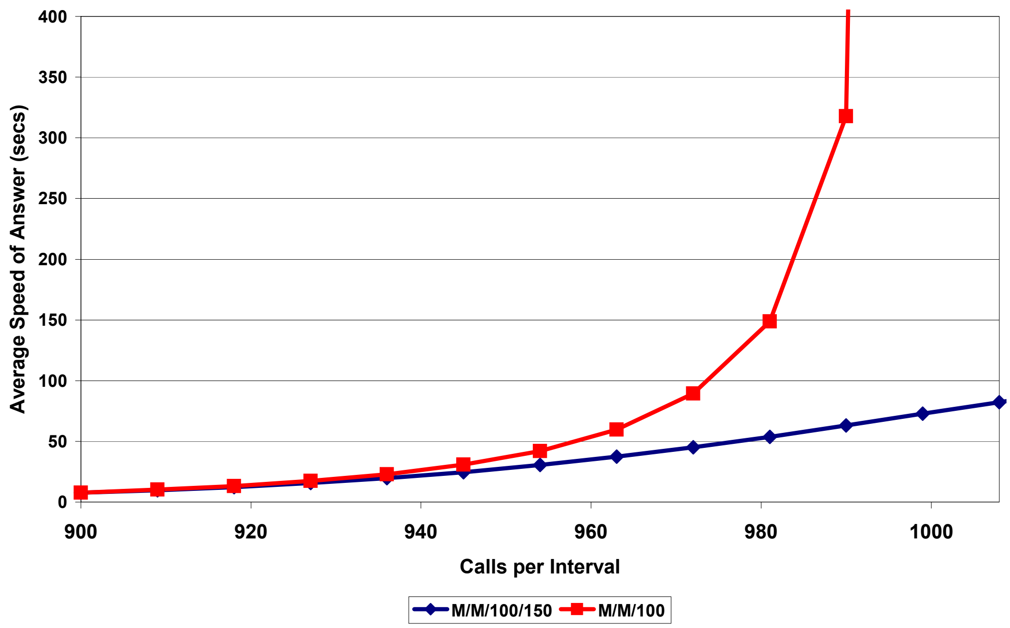
4CallCenters: Advanced Profiling

Arrival rate varied from 900 to 1017 per hour, in step 9.

Excel interface: graphs and spreadsheets.

M/M/n/K vs. Erlang-C

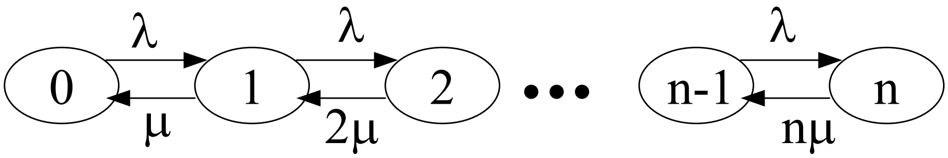
Average service time = 6 min, 100 agents, 150 trunks



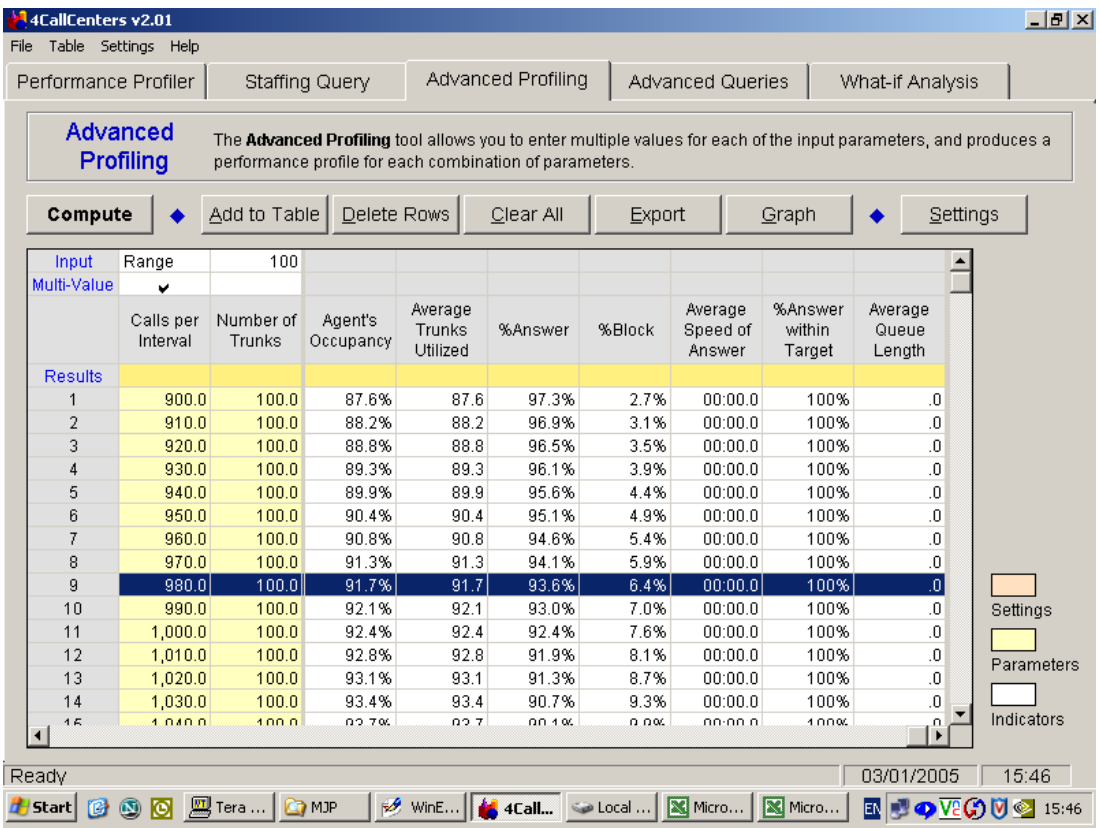
Similar performance for light loads.

Erlang-C "explodes" as $\rho = \dfrac{\lambda}{n\mu} \uparrow 1$.

M/M/n/n (Erlang-B) queue



$$\lambda_i \equiv \lambda, \qquad 0 \le i \le n-1\,,$$
$$\mu_i = i \cdot \mu, \qquad 1 \le i \le n\,.$$



Results	Calls per Interval	Number of Trunks	Agent's Occupancy	Average Trunks Utilized	%Answer	%Block	Average Speed of Answer	%Answer within Target	Average Queue Length
1	900.0	100.0	87.6%	87.6	97.3%	2.7%	00:00.0	100%	.0
2	910.0	100.0	88.2%	88.2	96.9%	3.1%	00:00.0	100%	.0
3	920.0	100.0	88.8%	88.8	96.5%	3.5%	00:00.0	100%	.0
4	930.0	100.0	89.3%	89.3	96.1%	3.9%	00:00.0	100%	.0
5	940.0	100.0	89.9%	89.9	95.6%	4.4%	00:00.0	100%	.0
6	950.0	100.0	90.4%	90.4	95.1%	4.9%	00:00.0	100%	.0
7	960.0	100.0	90.8%	90.8	94.6%	5.4%	00:00.0	100%	.0
8	970.0	100.0	91.3%	91.3	94.1%	5.9%	00:00.0	100%	.0
9	980.0	100.0	91.7%	91.7	93.6%	6.4%	00:00.0	100%	.0
10	990.0	100.0	92.1%	92.1	93.0%	7.0%	00:00.0	100%	.0
11	1,000.0	100.0	92.4%	92.4	92.4%	7.6%	00:00.0	100%	.0
12	1,010.0	100.0	92.8%	92.8	91.9%	8.1%	00:00.0	100%	.0
13	1,020.0	100.0	93.1%	93.1	91.3%	8.7%	00:00.0	100%	.0
14	1,030.0	100.0	93.4%	93.4	90.7%	9.3%	00:00.0	100%	.0
15	1,040.0	100.0	93.7%	93.7	90.1%	9.9%	00:00.0	100%	.0

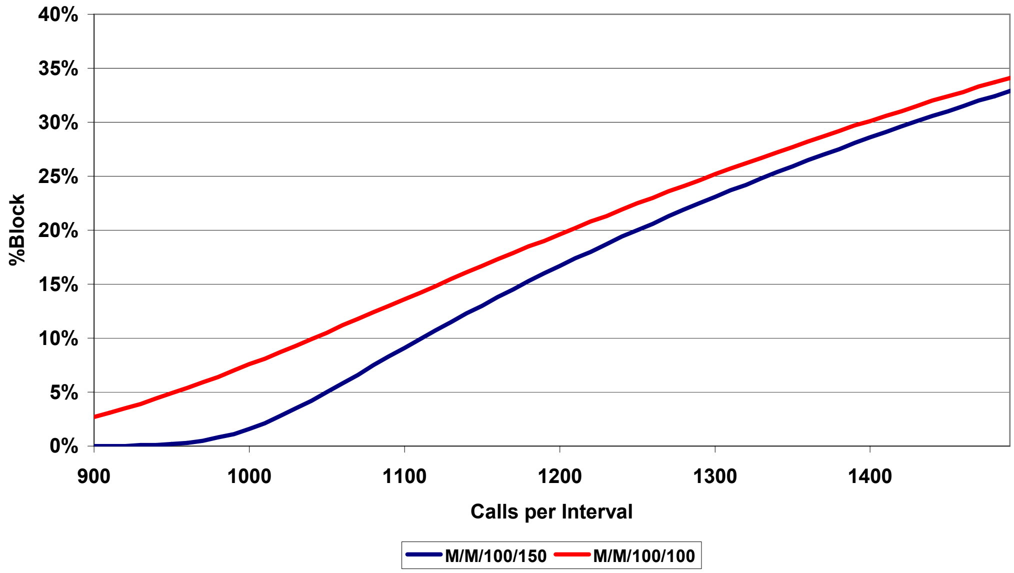
No queue $\Rightarrow$ no wait.

$$\pi_i = \frac{R^i}{i!} \Big/ \sum_{j=0}^{n} \frac{R^j}{j!}\,, \qquad 0 \le i \le n.$$

Note: interval = 1 hour

M/M/n/K vs. Erlang-B

Average service time = 6 min, 100 agents



Moderate load: additional trunks prevent blocking.
Heavy load: % blocking $\approx 1 - 1/\rho$ (*"fluid limit"*).

Erlang-B Formula (1917):
Loss probability

$$E_{1,n} = \pi_n = \frac{R^n}{n!} \Bigg/ \sum_{j=0}^{n} \frac{R^j}{j!} \qquad (1)$$

Follows from PASTA.
(1) valid for M/G/n/n! (Generally distributed service time.)

$\lambda \pi_n$ – rate of lost customers,
$\lambda(1 - \pi_n)$ – effective throughput.

Erlang-B computation: via recursion

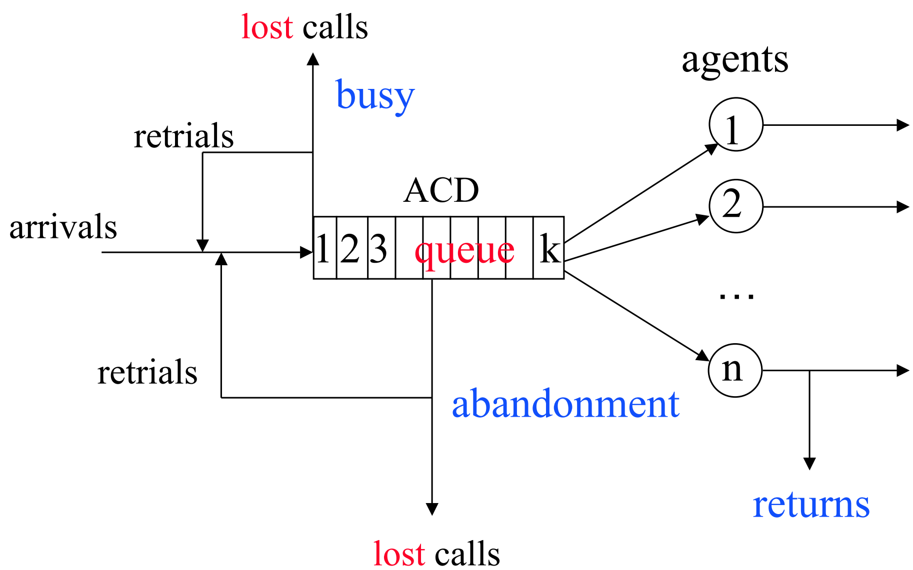
$$E_{1,n} = \frac{RE_{1,n-1}}{n + RE_{1,n-1}} = \frac{\rho E_{1,n-1}}{1 + \rho E_{1,n-1}} \qquad E_{1,0} = 1\,.$$

Note:

$$E_{1,n} = \frac{(n-R)E_{2,n}}{n - RE_{2,n}}\;; \qquad E_{2,n} = \frac{E_{1,n}}{(1-\rho) + \rho E_{1,n}}\;;$$

$E_{2,n} > E_{1,n}$, as expected: why?

Schematic representation of a telephone call center



Two customer - centric (subjective) operational measures of performance:

- Abandonment (impatient)
- Retrials (often negligible)

How to model Abandonment?