



$$L = \lambda W$$

$$W = L \div \lambda$$

works for any arrival pattern, order of service, or number of servers, as long as the system is stable

THE FORMULA

L Items in the system, on average: people in the line, tickets in the queue, machines in the shop.

λ Arrival rate, on average: two people a minute, fifty tickets a week. In a stable system as much leaves as arrives, so this is also your throughput.

W Time each item spends inside, on average: from the moment it walks in to the moment it walks out.

Time in system = items in system $\div$ throughput. Know any two, and the third is fixed. Not estimated. Fixed.

THREE USES

1 Predict any wait

Divide what's inside by how fast it finishes. You are last in a line of 10, served at 2 a minute: you wait 5 minutes. Everyone ahead of you has to leave first. *The line is the wait.*

2 Find the two levers

A wait only gets shorter if one of two numbers moves: **shrink the queue** (less inside) or **raise throughput** (finish more per day).

Not on the list: working harder on the same capacity, daily escalation calls, motivational emails, new dashboards, expediting. Expediting moves one item to the front and pushes everyone else back. The average does not budge.

3 Know when it lies

Four checks before you trust the number: see "When it breaks down".

WHEN IT BREAKS DOWN

1 It needs a steady state

It describes averages over a stretch of time when about as much leaves as arrives. In a rush (the queue growing every day), today's queue $\div$ today's throughput **understates** the wait; during a drain, the opposite. Check that arrivals $\approx$ completions over the window. If not, the honest answer is not a number: *"the wait is getting longer."*

2 Keep units and the window consistent

200 open $\div$ 50 closed *a week* = 4 *weeks*, not 4 days. Measure all three numbers over the same period. Last quarter's throughput against today's queue is a guess. Mind the boundary: count only the items waiting and you get only the waiting time.

3 Averages, not individuals

It won't tell you when *your* ticket closes. A 4-week average can hide urgent tickets closed in a day and forgotten ones that sit for months.

4 It tells you what, not why

It gives the wait, not the reason throughput is low. For that, find the bottleneck. (Next episode: the Theory of Constraints.)

WORKED EXAMPLES FROM THE EPISODE

SITUATION	MATH	ANSWER
Coffee shop: 10 people ahead, about 2 served a minute	$10 \div 2$	5 min wait
Restaurant (in reverse): 60 seats, all full, guests stay 1.5 h	$60 \div 1.5 \text{ h}$	40 / hour capacity
Support queue: 200 open tickets, 50 closed a week	$200 \div 50$	4 weeks wait
Repair shop: 300 machines in, 100 fixed a day	$300 \div 100$	3 days out
Goal 1.5 days, lever 1: shrink the queue to 150	$150 \div 100$	1.5 days
Goal 1.5 days, lever 2: raise throughput to 200/day	$300 \div 200$	1.5 days

Restaurant: more than 40 arrivals an hour and the crowd at the door grows. The only fixes are more seats or finishing a little sooner. Shrinking a queue is usually a decision you can make this week. Raising throughput usually takes people, money, or a fix to the bottleneck.

TRY THIS WEEK

Pick one queue you own or depend on: a backlog, a review queue, a hiring pipeline, a repair line. Write down three numbers:

If the last two are close, divide the first by the second. That's your average wait: _____. Compare it with what people *believe* the wait is. The gap is often where the frustration lives. Then ask: **too much in the queue, or too little throughput?**

A line for your next meeting:

"Our wait is queue size divided by throughput. To cut it, we either finish more each day or let less in. Which one are we choosing?"