The Menace of Mains Away

This is an article about a common problem which occurs in a pub which serves food: sometimes guests are left waiting for their main courses to arrive. I attempt to explain the causes of this phenomenon.

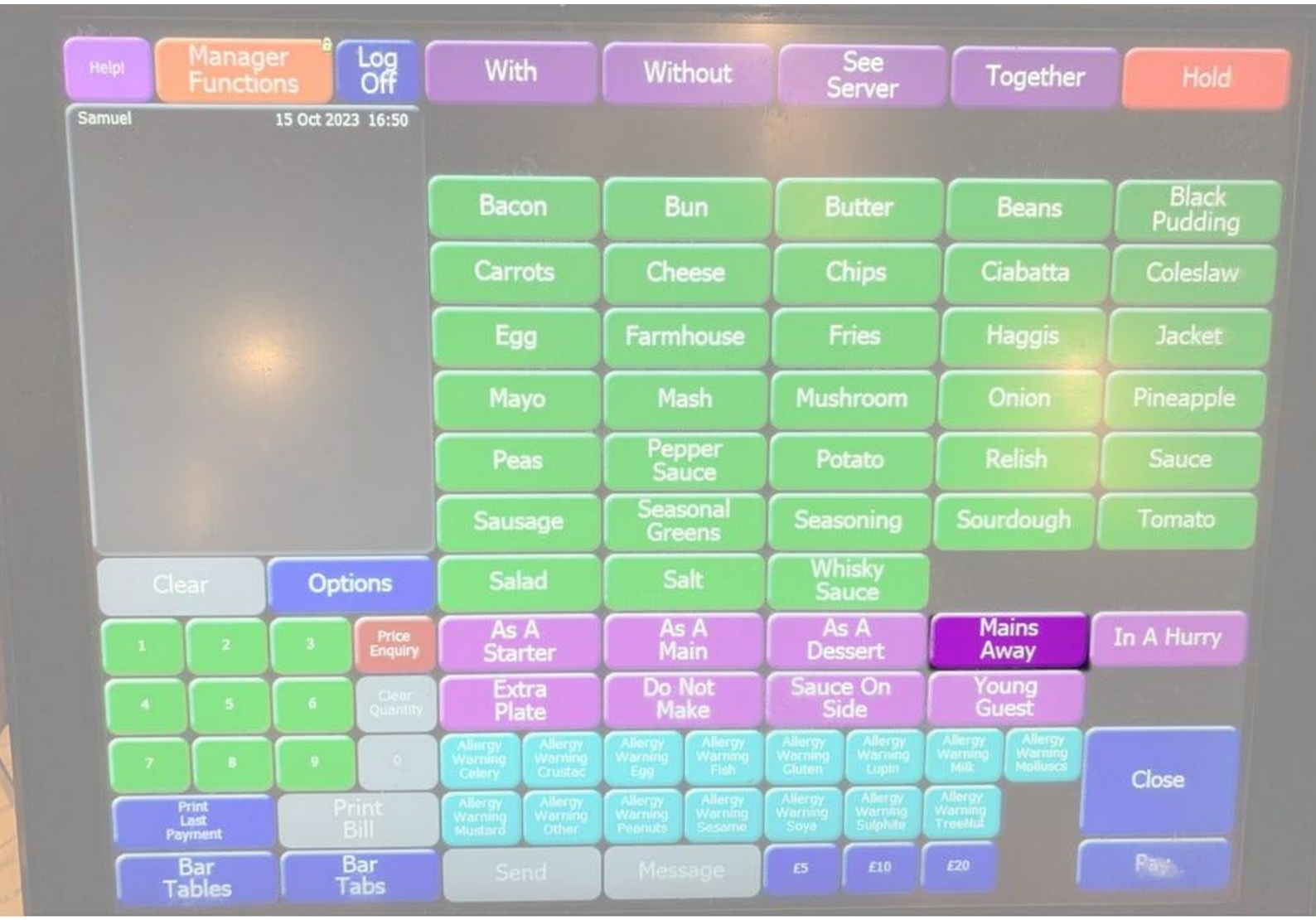
I conclude that the causes lie in the way teams in the pub are trained, and the configuration in the Point of Sale (POS) machines.

I arrive at the following conclusions. “Lift Picking” should be discouraged. This is the practice of taking meals straight from the food lift and to a table without using the meal to reconfigure the tickets on the rail. Tickets are a language amongst front-of-house staff and if they’re not updated, mis-communication occurs. Second, the team should make it clear, amongst one another, that the person who serves starters has sole responsibility for triggering the main courses.

The Mains Away Button

There is a button on the till called “Mains Away”. What does this mean? Well, it is for situations in which customers have a starter

course as well as a mains course. They might have potato soup as a starter and a steak pie for their main course.



If two unrelated customers were each eating a meal, we'd want to cook their meals at the same time, if we could. If one had soup and another a pie, it would be okay if both the pie and soup were placed on their separate tables, at the same time.

But if both pie and soup are being sent to the same individual, this would be problematic. For the person wants to eat the

soup *first*. Only *afterwards* do they want their pie. The customer does not even want the pie on the table while they are eating their soup. So, we need a delay between the courses.

The steak pies at Nicholson’s pubs are very popular.

How do we manage this? Well, we hold back on the main course until the customer has finished their soup. We plan to send some signal to the chefs when the soup is finished. This signal says "initiate the pie", or "mains away" — as in "chocks away", shouted as an aeroplane takes off.

In practice, there is a button which says "mains away". It is within a container on the till called Food Modifications (Food Mods).

A person needs to go up to the till (or POS machine). They authenticate with their Dallas Key. They click on "Table Numbers" and then select the appropriate table number from the floorplan. Then, they select "Food Mods" and "Mains Away". This prints a ticket in both the kitchen (for the chefs) and on the ground floor (for the waiters). It says "Mains Away". Now the chefs know to begin the main course and all the waiters can see that the mains away signal has been given. This latter fact helps to prevent multiple waiting staff giving Mains Away signals for the same customer.

What is the problem with this arrangement?

The explanation for why the arrangement is deficient hinges upon the fact that in a reasonably busy pub (one likely to generate agreeable profits), we cannot rely upon the empty dishes of the customer as a visual trigger for sending the Mains Away signal.

The *Serving Staff* help one another in a flexible way. There is **dynamic cooperation**. What I mean by this is that it is very hard to predict who will do a specific task. I might serve the main

course for somebody, even though my colleague served their starter; I might put the new bin liner in a bin, even though it was my colleague that removed the full bag before; my colleague might sweep up the shattered fragments from the plate I dropped just two seconds ago. We truly work as a team. One person picks up seamlessly from where the other person left off. To some extent, this is unavoidable in any retail outlet. It can also also be very efficient to *maximise* this sort of dynamic cooperation (this is the rarer thing which we have). It allows team members to follows a **general principle** which we might call Deliver At Pace.

In effective teams, ego is left at home and there is dynamic cooperation. In a starling murmuration, each bird picks up on signals that are outputted by their neighbours.

This **dynamic cooperation** means that one person might go around the pub and collect any dirty plates and bowls which they find. This will be *any* dirty plates. To establish some rule such as

Clear only those tables for which they placed the food order at the till

is impractical. This rule is a sort of ownership rule — as if each employee is responsible for the entire service cycle relating to a specific customer. This is a nice romantic idea. The same person that rings their meals through the till takes food to their table and asks them if their meal is okay and so on.

But it's impractical for two reasons. First, the person clearing dirty dishes from tables may not have served *anyone* on the till. Perhaps they don't know how to. Perhaps the team has simply decided to divide up their duties. For all of today, John will clear the tables and Alice will be on the bar.

Second, it is massively cognitively demanding. An individual might — while at the bar where the POS machines are — serve food to over ten groups in a ten minute period. They might serve some drinks to a group. Another colleague takes payment for their meals a few minutes later after they have looked at the menus. Keeping track of all these details in memory would be massively cognitively demanding. For each face, the employee would need to think *Did I put their food through the POS? Or was it John who took payment for their meals? Did I simply do drinks for this individual?* In short, the rule above is impractical. It cannot always be the person who put the food order through the till that clear the dirty dishes from the table.

But the rule looks tempting. After all, one of the dirty bowls collected in will be the soup bowl which was somebody's starter. And this dirty bowl is the perfect trigger for the Mains Away signal to be sent.

This event *ought* to be the trigger for the “Mains Away” signal to be sent.

So, why won’t the dirty dish be the trigger? Because the server collecting in the empty soup bowl has no idea that the soup is simply the first course. It’s perfectly possible that the customer is having *only* soup. Or perhaps they are having *only* calamari. Perhaps they are having *only* the Garden Platter (or any of the other starter items). Once they’re done, they’re walking out.

Now it might be objected that it is *obvious* that a soup bowl was a starter because soup is rarely anything but a starter. And you can tell from the bowl that the meal was soup. The problem with this general thought, however, is that some starters are served on plates. Examples include the tomato tart and the salmon on toast. With these starters, there is nothing about the dirty dish *in itself* which tells the server that it was a starter. In other words, starters have no intrinsic properties. There is nothing that can be observed that tells you it is a starter.

A dish is a starter in virtue of how it *relates* to the other courses. The clue’s in the name: it starts off the main course. So, some dishes certainly have the property of being a starter. No one disputes that. But this is an *ex*trinsic property — rooted in how the dish relates to other dishes. It is not an intrinsic property. It is certainly nothing that can be observed.

For all those reasons, a person collecting in a dirty dish cannot use it as a trigger for sending the Mains Away signal. Recall that the person collecting in the dish will be likely a distinct individual from the person that placed the order.

Yet managers in pubs *today* wander around with this faulty belief sat in their mind. They possess the belief that the dirty starter dish should be a visual cue for Mains Away. (It follows that when they blame those who collect in dirty dishes for not sending the Mains Away signal, it is not just.)

If we are relying on the empty dishes as a visual cue — and we do not pick up on the cue — then the kitchen do not initiate the mains course. After all, the person collecting in the dirty dish is oblivious. I have already described why: there is dynamic cooperation and to expect them to have this sort of knowledge is cognitively impractical.

So, the customer on the table waits and waits. The serving staff are pre-occupied with other matters *(in a completely genuine sense)*.

The ticket just sits there in the prep kitchen, which is staffed by Front of House staff. The ticket waits for someone to give the Mains Away signal.

The empty soup bowl is **not** the *ideal visual cue for sending a Mains Away signal, however counterintuitive this may sound.*

It is irrational to simply stand and examine tickets to inspect whether they demand a Mains Away signal, at arbitrary points in time. This might work some of the time. But really, we need a **procedure** for dealing with Mains Away. Without it, the signal does not reliably get sent to the kitchen. (I hope to describe that procedure in this piece.)

And yes, this really happens. Recently, there was a large party of Spanish tourists in the pub. All on one table, they paid for their food separately. So, there are several tickets. Some tickets will have starters *and* mains on them. Other tickets will only have mains on. The consequences of this are that some individuals on the table receive their mains, but others wait and wait for their mains. *The signal for Mains Away on their ticket was not sent to the kitchen because the team lacked a* ***clear procedure*** *for sending Mains Away signals.*

The incongruity amongst the guests — some sitting waiting for their mains while others tucked into their mains — made the social occasion painful. The occasion essentially became a *cause* of a negative review online. So, pubs suffer because of managers wandering around with a faulty belief in their mind.

A party of people at a table might all have separate tickets. Some will have starters and mains.

What is the solution to this? We need an established procedure for sending the Mains Away signal. And it must preserve the usefulness of the practice. You see, simply sending the signal immediately (when the person pays for their soup and pie) will certainly mean that nobody forgets the signal. But you've killed the usefulness of the practice — now the starters and mains will come at the same time.

We want to *preserve* the usefulness of delaying the mains course. We want to optimise the customer experience. It is useful to have the main course be delayed. It means that the mains course is hot and fresh when you eat it, instead of sitting on the table, or under the hot lamp in the prep kitchen (the side wing of a pub's main stage).

If using dirty dishes as a visual cue won't work, here is a **second proposed solution**. We make all the servers who collect in plates

be responsible for *checking* whether the plate was a starter. Now this looks like a good solution because the empty dish of a starter is evidence that the customer has finished their starter. If there is nothing about the dish itself which establishes whether it was a starter, then the ticket will make this clear.

But this is impractical. As they moved away from the table — dirty dish in hand — servers would have to remember the table number. Otherwise, they would not know which ticket to check.

Furthermore, this would result in a massive number of trips. Sometimes, servers go from table to table taking plates. With this second solution, servers would always have to visit one table and then "check in" with the prep kitchen. *Every time*. Table then prep kitchen, table then prep kitchen. Sometimes servers would forget the table number by the time they returned to the prep kitchen.

A third problem with this solution is that most of the time people do *not* have multiple courses which require a Mains Away signal. So servers would feel like a fool doing this. These are the reasons it is irrational to ask servers who collect in dirty dishes to be responsible for sending a Mains Away signal on a ticket.

Reflective Equilibrium

Stepping back for a minute, we are involved in a negotiation between theory and practice. Suppose we construct a theory. Suppose it demands we do things which people will not do. This will render the theory pointless, since its purpose was to influence action.

So, the theory has to fully accommodate reality. Even though it will advise we act differently in some limited cases — the way that Germ Theory advised people to *override* their default actions and *not* ignore the handwashing sink — it must still fit people's intentions

We want it to be the case that what the theory demands we do is likely to be done. The philosopher John Rawls used the term *reflective equilibrium* for this back-and-forth between **general principles** and **particular judgements**. The problem with using dirty dishes as the trigger to check for tickets requiring a Mains Away signal is that it doesn't account for reality. It is incompatible with the general principle *Deliver at Pace*.

Now I will describe Solution 3, which is better. There is always one individual in the prep kitchen that removes the order ticket from the ticket-printing machine and *pins* it on the pass (for any given ticket). It's not always the same individual pinning all tickets, but for each ticket, somebody passing through will pin it up on the rail.

Tickets are hung on a rail in the “prep kitchen”. They are waiting to match physical dishes that come from the kitchen, ready to be served.

The ticket is ready to *match* one of the meals that arrives in the lift. When a ticket matches a dish, servers can establish which table to carry the dish to. (Errors comes when tickets concurrently cover multiple dishes, something we might discuss elsewhere).

The third solution involves assigning responsibility to *this* person, the *Ticket Pinner*. Their responsibility has two parts. It is to

recognise and then communicate with a specific person. Allow me to flesh this out in more detail.

First, they ought to **recognise** the ticket as one that demands Mains Away. This is quite simple and comes with very little training. Follow this rule:

> **If** one dish is designated a starter and another a Mains
>
> **then** the ticket demands Mains Away.

The *Ticket Pinner* has a responsibility of recognition. They must be trained by managers to recognise tickets that demand Mains Away. Pinning without recognising is a definite failure point.

Zonal's EPOS system makes it the *default* that a ticket demands Mains Away if there is a starter and a mains course. This means that there is no special button which must be pressed to say "makes the mains wait". If a person orders a starter and a mains, then that ticket will generate starters first and then the chefs will *wait* for a trigger to cook the mains. This catches a lot of new serving staff off guard. They do not spot tickets which require a Mains Away signal and customers are left waiting. Negative reviews are reliably generated.

On a Zonal EPOS, the default is that a ticket with starters and mains demands mains away.

In fact, when placing an order at the Point of Sale (POS) machine, you have to go out of your way to press a button called “All Together”. This button overrides the default. It means that the starter and mains come all together .The ticket does not demand a Mains Away signal.

So really, our *Ticket Pinner* ought to follow a rule such as:

> **If** one dish is designated a starter and another a mains
>
> **and** “All Together” has not been specified,
>
> **then** the ticket demands Mains Away.

But that is still pretty simple. Ticket Pinner does not only have a duty as recognizer. They also must alter the ticket in some way. They

must mark up the ticket so as to *communicate*. They are communicating with the person that is going to serve the Starters.

It's really important that we get this correct. They are not communicating with the person that removes the dirty soup bowl. They are not communicating with the person that placed the order on the till. They are not communicating with the person that is going to serve the main course. They are communicating with the person that is going to *serve the starters*. In all likelihood we don't know who this person will be yet, because of the way the team dynamically cooperate — that is fine.

This asynchronous communication (similar to sending an email to a colleague) involves marking the ticket with an "MA" in **red marker pen**. *Red* so it is quickly visible in the rush of service. *Marker pen* (whiteboard pen) so that the ink will deposit on the thermal paper. A regular biro will make a subtle mark on ticket paper. A pencil is hopeless. Highlighters are no better. *MA* because it can be written quickly, the letters standing for Mains Away. This is the way Ticket Pinner communicates with Starter Server.

Put a red whiteboard pen next to the ticket printing machine.

Signal to Starter Server

Once a starter has been cooked by the chefs, it will appear in the lift. Somebody on the serving staff will become available to serve it. They match a ticket on the rail to the starter.

The ticket which was matched to the dish then informs them of which table to walk to. But also, our *Starter Server* receives a signal (reaching from back in time) from *Ticket Pinner*. And what does the

signal do? It means that Starter Server has a responsibility. Staff must be trained to understand that picking up such a marked ticket is an act of taking on a responsibility.

Starter Server (Star-Ver for short, since if they do not act responsibly, they will starve the customer by depriving them of their mains) must send the Mains Away signal to the chefs. No one else is responsible. This, I believe, is the most optimal arrangement. This means that this individual, arbitrarily selected, ought to remember to trigger Mains Away. If it is understood amongst the team that this is how it works, then at least there is a clear assignment of responsibilities.

For this requires some concentration on Starver's behalf. So, yes, they may make a mistake on occasions — as frequently as with any other task which can be done badly. But really, there are only two situations. **(1) The starter is small**. *Starver* can just send the signal to the chefs immediately. By the time the main has been cooked, the customers will be finished. The desired staggering of the meals is achieved. **(2) *Starver* really must wait a little bit.** There are *lots* of starters — perhaps two Garden Platters and a Landlord's Platter on a large table. So you really ought to wait a bit before initiating the cooking of the mains, since they would arrive too soon and begin deteriorating before the starters are finished.

In this second situation, You **must** set a ten minute timer on your phone. Ask a colleague to remind you at 16:10 to send the Mains Away signal. Write a note and put it under your watch. The point is

is to put in place concrete techniques to remember. (This essay assumes, by the way, a severely crowded London pub. You are sweating like a pig. Two colleagues phoned in sick. You are juggling multiple customer requests which actively cause *Starver* to forget the MA signal). The phone alarm is a reliable technique. A a colleague will tolerate your fault if you forget, owing to Dynamic Cooperation. I will expand on this point shortly. But recall that in this essay I am concerned with outlining *one clear procedure*.

Let's summarise the parts of the procedure we've established so far. We have two people who bear responsibilities at distinct points in time. *Ticket Pinner* is responsible for recognizing and marking up. They make a mark using the red marker pen *which is next to the ticket machine*. Then, *Starver* (the server of the Starter) is responsible for looking for that marked up ticket and sending the signal to the kitchen — they are responsible for pressing Mains Away on the POS.

Now, when there is a failure with Mains Away, it will be the responsibility of one or both of these people. Perhaps (1) Ticker Pinner failed to mark up the ticket and then Starver starved the guests of their mains by not pressing the Mains Away button. Or perhaps (2) Ticker Pinner did her job and wrote MA in red on the ticket but nobody sent the signal for Mains Away. (3) Ticket Pinner might fail to mark up the ticket and Starver nevertheless recognises the ticket as demanding Mains Away. They act, and press the button on the till, covering up for Ticker Pinner's sloppiness. Those, logically, are all the possible scenarios.

Crucially, Situation (3) is the system tolerating a fault. From the customer's perspective, everything is fine and they got their mains on time. But we cannot rely on this sort of heightened attention to detail from the Starter Server. The argument of this essay relies on the fact that Situation (1), with its double failure, is the most common situation to occur in reality, when there is a Mains Away failure— by far. I argue that there are *genuinely* two failures of responsibility in this scenario. *Ticket Pinner* ought to have left that record (the red-stained ticket) which can be picked up and acted on by colleagues; Starver ought to have sent the signal to the chefs.

If there is an important insight to convey, however, it is that situation (2) is virtually a **fiction**. To have a ticket marked up, and have nobody send the MA signal — this very rarely occurs. This might be surprising since at first blush, it seems perfectly possible that a person serve starters to a table without checking the ticket, marked in red ink as it is.

Yet the fact is that:

> **if** a ticket has been marked with some demanding sign
>
> **then** *some* team member will respond to the sign eventually.

It might not be Starver, but *som*e team member will respond to the demanding sign. This gets to the heart of why Solution 3, the Mark-up Solution, is most optimal.

Children at a table are an example of a **Demanding Sign**. Namely, some observable state of affairs which calls for action in trained employees. We see children at a table — we ask if they would like some crayons. It is simple.

I note that many such signs are such that one would argue that they are *instinctively* responded to; only psychopaths must be *trained* to comfort crying children etc. This may be true, and the concept still works.

Another example of a Demanding Sign is the set of behaviours in a drunken individual which means that they should no longer be served (e.g. slurred speech, drooping eyes).

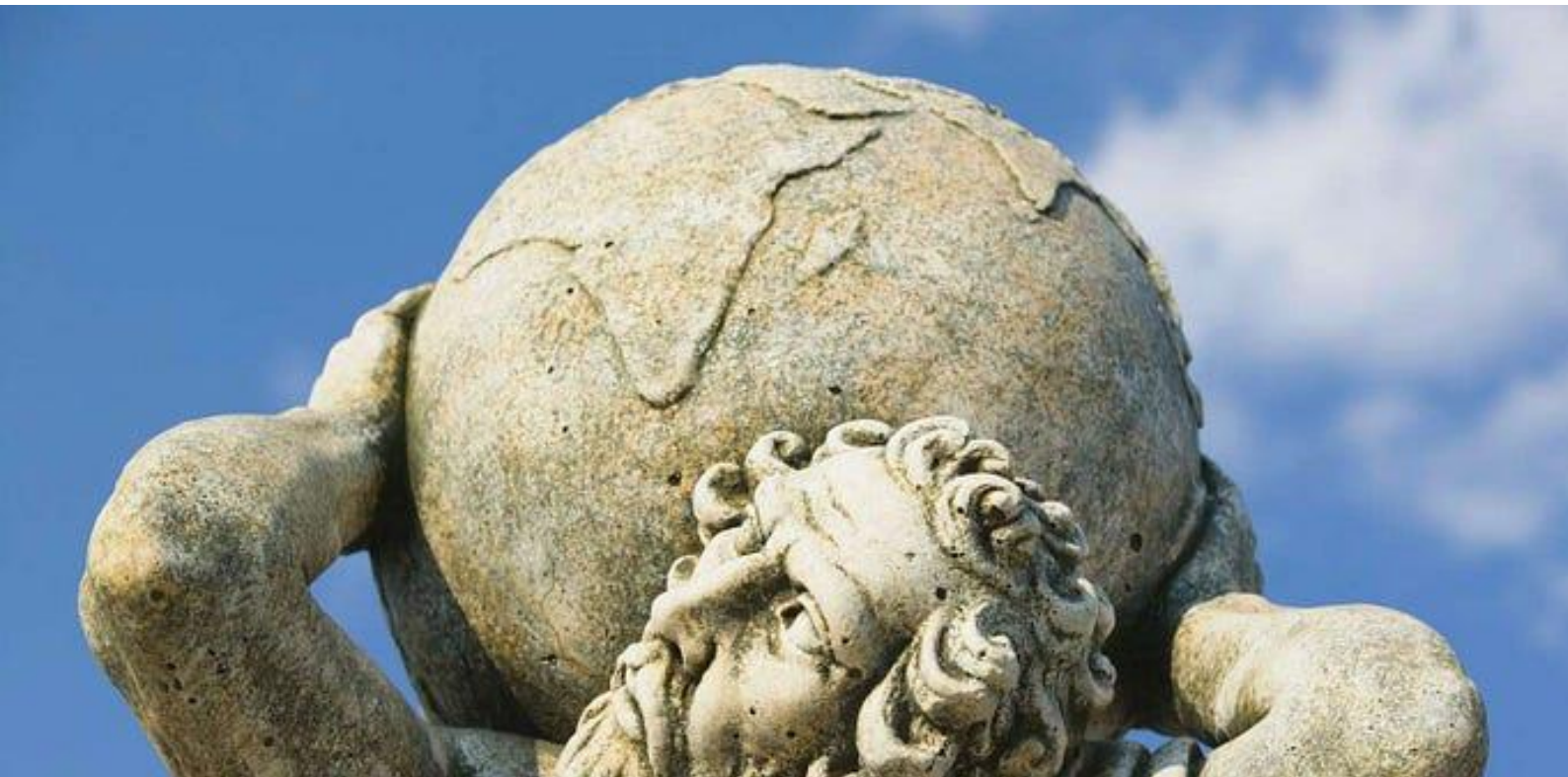
By *demanding sign* I mean some observable situation which calls for action. For those trained, a customer stood at the bar is a *demanding sign:* The customer requires serving. For those trained, children at a table are a *demanding sign:* They require crayons. For those trained, children at a gambling machine are a demanding sign: They require a reminder of the law on gambling. And so on. Well, for those trained, a ticket with "MA" written on is a *demanding sign:* It requires that you press Mains Away on the till within a reasonable amount of time.

The consequence of this is that it is no disadvantage of Solution 3 at all if servers occasionally take food from the lift, never checking Rail Tickets. That fact is somewhat remarkable. It is true because *somebody* will observe the demanding sign on the ticket rail eventually. It is not an *essential* feature of Solution 3 that Ticket Pinner communicate with Starver, even if it is the ideal and I would maintain that Ticker Pinner *does* have a responsibility to attempt communication with Starver.

`Objection — Starver has Sole Responsibility`

I now want to address an objection to my solution. The objection runs like this: the proposed solution is not true. Ticket Pinner does not have any responsibility at all.

Now, why might you think this? Well, the reason you'd think this is that the duty of recognition which I assign to Ticket Pinner might as well be assigned to *Starter Server*. The recognition responsibility I assign to Ticker Pinner is superfluous.



It looks very reasonable to assert that Starver has sole responsibility for sending the Mains Away signal.

If Starver is matching tickets to dishes when they serve dishes, then they are looking at tickets. If they are looking at tickets, then they should recognise tickets that demand Mains Away. In this way, *Starter Server* may as well be given all the responsibility. So *Ticket Pinner* is not responsible for recognising tickets demanding Mains Away, and therefore not responsible for communicating with Starter Server.

Let me now address why the objection does not hold. When you are serving food, you are not thinking about whether it is a starter or a main. A person sees food in the lift, works out which table it must be

taken to, and takes it to the table. It is very hard to judge, from the dish itself, whether it is a starter or a main. As we previously noted, starters have no intrinsic properties which are observable. The property "is a starter" is an extrinsic property. This means that usually, people do not *know* that they are serving starters even if they are serving a dish which is a starter. The dish is not a demanding sign.

You can argue that they *ought* to establish that what they serve is a starter. Well, fine. Let's suppose we *do* argue this. "It's your job to know that what you are serving is a starter." Why might this be preposterously inefficient? It's because in truth, one person takes dishes that appear in the lift and obtains the minimum knowledge required to get the dishes to the table. I'd even argue that this is an essential characteristic of the sort of fast pace of delivery that is expected in my pub. Such "minimum knowledge" is the table number, which is the dish's destination.

Such knowledge of table destination can be obtained from the **Tag Ticket** that appears alongside the meal in the lift. With the Tag Ticket, t*hey have all they need and they go*. The food server does not actually **check the ticket on the rail** (the Rail Ticket)**.** Nor do they need to. The Tag Ticket in the lift has the table number written on it.

The Rail Ticket, which would tell the server that the dish they are serving is a *starter* (because of its relation to future dishes) — well, this is never checked.

So the question is: Do we really want a world in which we enforce that the food runner must consult the rail *every single time?* Even when there are situations in which they *know* where the food must go? We are to enforce this tedious ritual of walking to the rail on the off-chance that the meal they are serving renders them a *Starver*, with responsibilities for sending Mains Away? Of course the answer is no.

It is not going to happen in practice — people take food from the lift and just run with it, coming back later to *retroactively* place the Rail Tickets onto the skewer. They come back after having delivered the dish to its destination and then examine the tickets on the rail. They recollect, saying “Ah, this ticket has been done” and “I also did this one earlier”, removing the clutter from the rail. Giving physical food to customers is, after all, a more time-sensitive task than skewering tickets. They are acting in accordance with the general principle *Deliver at Pace*. Any theory needs to respect this general principle if it is to work.

When Rail Tickets have matched dishes, they are skewered on a spike.

For these reasons, it is unreasonable to say that *Starver* has sole responsibility for sending the Mains Away signal. The problem with this line of thought is as follows. It is all well and good referring to them as Starver (or starter-server), but in many cases, they do not *know* that the dish they just served was a starter. They do not *know* that they are starter server. They do not *know* that they suddenly acquired the responsibilities of sending the Mains Away signal.

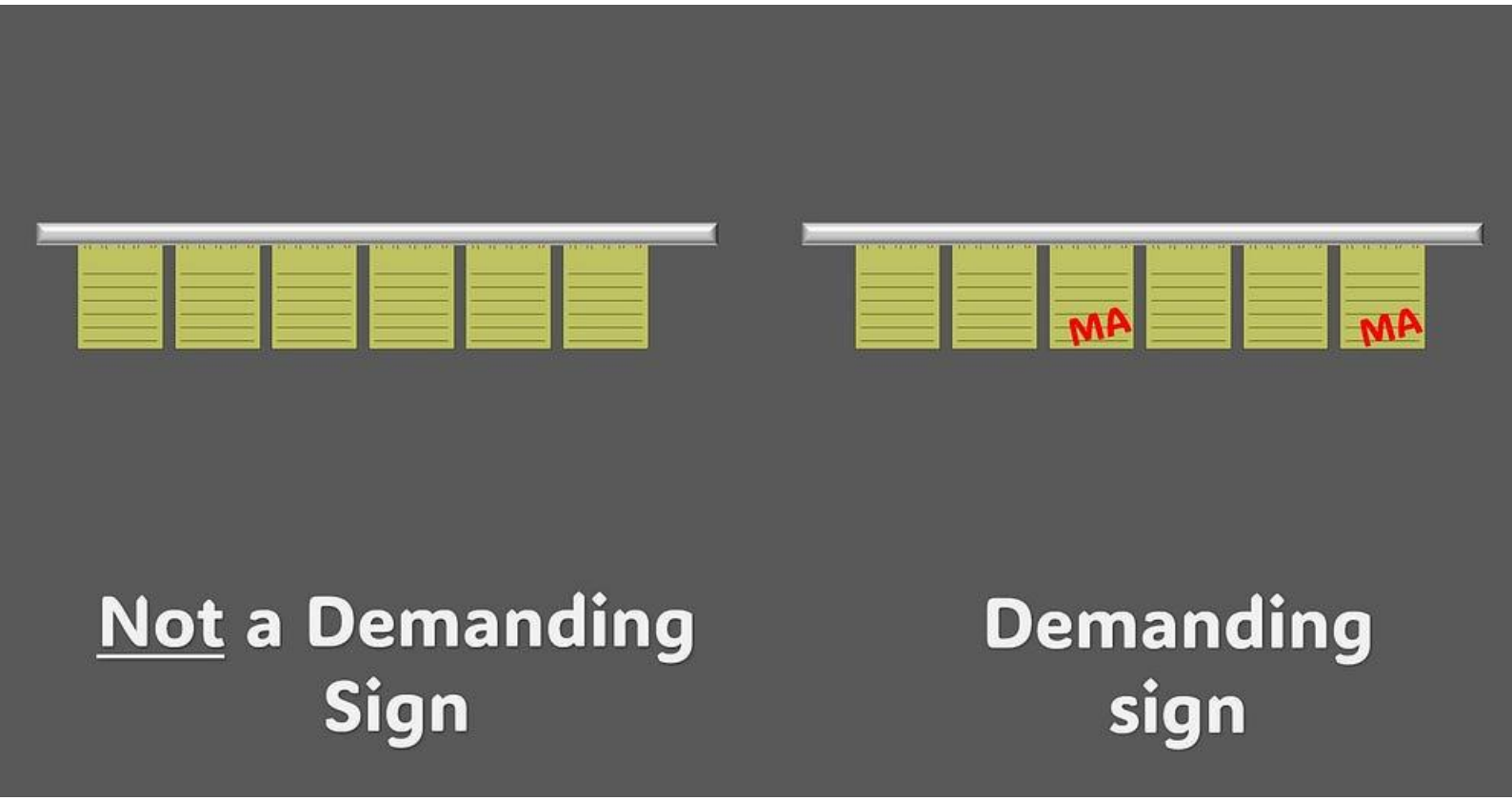
Recall that "being a starter" is not an intrinsic property. Furthermore, starters are no *demanding sign* in the way that children at gambling machines are, or customers stood waiting at the bar. Our procedure must ensure that those who serve starters are fully aware that what they just served was a starter, and thus that they now carry the responsibility for sending the Mains Away signal.

Saying that Starter Server has the responsibility for recognising starters — for recognising tickets that demand Mains Away — is not a solution to the problem. For is is possible to *unknowingly* be Starter Server. This is because it is possible (indeed faster) to serve dishes from the lift without consulting the ticket rail. There is a physical distance between the lift in which dishes appear from the kitchen, and the ticket rail. It is the ticket rail which confers knowledge of which dishes are starters. Not the lift in which food appears. And *that* is the essence of the problem with this objection — of the problem with giving Starver all the responsibility. To take it seriously is to ask Starver to cease Delivering at Pace.

So, I hold that *Ticker Pinner* ought to communicate with Starver. Ticket Pinner must recognise and then mark up the ticket. This makes the system fault tolerant. The reason it does is that if somebody delivers a dish to a table, unknowingly being the Starver (something we might consider a fault), they will usually *then* go and skewer the Rail Ticket. It says “MA” in red ink that cannot be ignored. Thus, the fault is tolerated; the ticket will inform the person that they just acted as a *Starver* and establish that *they* are responsible for now sending the Mains Away signal. It has informed them of this in good time; there is usually about ten minutes available for a person to learn they were Starver.

Furthermore, though, anyone in the prep kitchen who is available can observe the marked up ticket and take it upon themselves to send the Mains Away signal. You don’t need to learn that *you* recently acted as Starver, only that *somebody recently* acted as Starver. This is strength of the Markup Method. It allows that the Starver come back and learn that they were Starver. It also takes into account situations in which they do *not* even do this. It tolerates this fault too. It fully accommodates a reality in which there is some degree of dynamic cooperation in teams and also team members Delivering at Pace.

Is there still a responsibility on Ticket Pinner? Yes, because we are grown up enough to recognise that the person serving dishes will not always skewer the Rail Ticket for the dish. They are delivering at pace. So they will not always learn that they are Starver. So they will not send an MA signal to the chefs.



The situation on the right demands action. The reason is that a filter has been performed on the tickets ("some tickets are red"). By marking some tickets with red, Ticket Pinner has distinguished a subset of tickets which require action. They have created the right conditions for **anybody** to come and dynamically cooperate.

So, there *is* a responsibility on Ticket Pinner to mark up tickets that demand MA. It is about leaving a record that *anyone* can pick up and act upon. It is about turning the whole *set* of tickets into a *demanding sign*. His failure to use this red marker pen by the ticket printer really *does* have a relation to bad reviews online. The good news is that it is eaily remediated. *Ticket Pinner* must output

signals that others can pick up on, the way starlings in a murmuration do.