0025 What Design Is

The Summers Principle Book · https://kb.jeffreysummers.com/docs/0025-what-design-is/

Design is engineered intent. That is the definition. But engineered intent is a phrase most operators have never had to sit with directly, and the definition needs unpacking to become operating vocabulary.

Design has three components. All three have to be present for design to be running. If any one is missing, design has collapsed into something else — usually intention without follow-through, which is a specific flavor of default.

The first component is the decision. The operator has decided what the outcome should be. Not vaguely — specifically. Not eventually — now. The operator can state, out loud, what the outcome is that they are engineering toward. If they cannot state it, they have not decided it, and design is not running yet.

Most operators skip this step. They have preferences (better food, happier Guests, more profit) but they have not translated preferences into decisions. A preference is a wish. A decision is a commitment to a specific outcome that can be built toward and measured against. The difference between operators who design and operators who default often starts here — at whether they have actually decided anything, or whether they are running on preferences.

The second component is the built condition. The operator has translated the decision into a condition that produces the outcome. Not a hope. Not an announcement. An actual structural change to the operation that makes the outcome more likely to occur.

If the decision is to hold a specific Guest Experience standard, the built condition is the hiring criteria that filter for cast who can hold it, the training that installs it, the accountability that enforces it, the measurement that confirms it. All of that is the condition. Announcing the standard is not the condition. Wanting the standard is not the condition. The condition is what is different in the operation because the decision was made.

Most operators skip this step too. They make decisions and stop there, believing that having decided is the same as having designed. It is not. A decision without a built condition is a memo — it may change the tone of the meeting but it does not change the outcome. The outcome traces to

the condition, not the announcement.

The third component is the maintenance. The operator holds the condition over time. Because man-made systems drift, and any condition that is not maintained decays. The hiring criteria drift as the operator gets desperate for warm bodies. The training decays as the trainer moves on and nobody rebuilds it. The accountability erodes as the operator gets tired of enforcing. The measurement disappears when the operator stops looking at the numbers.

Maintenance is not glamorous. Maintenance is the boring work of holding the same standard in week 47 that was held in week 3. Maintenance is the reason most restaurants that briefly ran well eventually stop running well — the operator built the condition but did not maintain it, and drift did what drift does.

Decision plus built condition plus maintenance equals design. Three components. If any one is missing, design is not running on that dimension. The outcome will still occur — outcomes always occur — but it will be a defaulted outcome, produced by whatever filled the space the design was supposed to occupy.

Design is not perfection. It is not the operator running every dimension of the restaurant at maximum intensity. That is impossible for any operator, at any scale, in any market. The operation has too many dimensions and the operator has finite capacity. What design requires is presence — the operator's engineered intent being present on the dimensions they have chosen to design, at the depth their current capacity supports.

An operator who has designed the front of house tightly and defaulted on the back may have a coherent GX and a leaking food cost. That is design on one dimension and default on another. Both are producing outcomes right now. The principle governs both. And the operator's work is to see which is running where, and to shift more of the operation into design over time as their capacity expands.

Design also is not micromanagement. Micromanagement is decisions and built conditions with no delegated authority — the operator making every choice themselves, in every moment, until they collapse. That produces short-term design results and long-term default, because the operator's capacity is a bottleneck and eventually the maintenance layer breaks down.

Real design distributes authority through built conditions that hold when the operator is not watching. The hiring criteria run when the operator is not hiring. The training runs when the operator is not training. The accountability runs when the operator is not correcting. The measurement runs when the operator is not looking. Because the conditions were built to produce

the outcome without the operator's continuous intervention, the design runs on the operation, not on the operator's presence.

That is the mature form of design. Decision. Condition. Maintenance. Distributed through the operation so that the outcome is produced by the system the operator built, not by the operator's continuous involvement. Fragile design requires the operator's constant presence. Mature design produces outcomes even when the operator is at home.

The operator moving from default to design is not moving from doing less work to doing more. They are moving from reactive work — putting out fires, absorbing shocks, cleaning up messes — to constructive work — building the conditions that produce the outcomes so the fires do not start, the shocks do not compound, and the messes do not accumulate.

Most operators think design is more work. It is not more work. It is different work. And in the long run, it is less work, because designed conditions produce outcomes without continuous intervention, while defaulted operations require continuous intervention just to hold the current level.

Design is engineered intent, made structural, held over time. That is what it is. That is what it takes. That is what the principle asks of the operator.

One last thing about design before default. Design is not a business strategy the operator learns. It is a human instinct the operator stops suppressing. Every person who has ever sat at a family table, been welcomed into someone's home, or felt genuinely seen by another human being already knows what design toward a Guest feels like. They have experienced it. They carry it. The capacity for connection, belonging, community, the need to be seen and to see others, is not a skill set. It is the human condition. The operator has already seen it. Not in a restaurant. Not in a training manual. Not in a leadership framework. At a table, where someone held the room together without trying. Where people felt welcome before they sat down. Where the person at the head of the table knew what everyone needed before anyone asked. That is not nostalgia. That is engineered intent working the way it was always supposed to work, in a domain the operator already understood before the industry taught them to stop trusting it.

The operator who designs the Guest Experience is not inventing something foreign. They are recovering something they already knew, and building a delivery system for it inside the operation they actually run. Different context. Different cast. Different Guests. Same destination. The felt knowledge is the target. The delivery system is new. And the operator who runs it is doing what most operators never attempt: designing the emotional outcome of the Guest Experience from the

inside out, starting from what they already know is possible.

That is what design is. Engineered intent. Three components. Distributed authority. And, underneath all of it, a human instinct the operator stops suppressing and starts building for.