



4

State the Problem: The TOSCA Framework

The first S in the 4S method stands for “state the problem.” To introduce it, we’ll stray from the exciting but sometimes dry world of business problems and consider a critical situation familiar to opera lovers: the challenge Tosca faces in the second act of the eponymous Puccini masterpiece.

Here are the facts:

- Mario, Tosca’s lover, has been arrested and will be executed tomorrow.
- Tosca is understandably worried.
- She would like nothing more than to get Mario out of jail alive before tomorrow and escape with him.
- Tosca, however, is a virtuous woman, and there are things she won’t do, even to save her lover.
- Unfortunately, Scarpia, the chief of police, is the key player here, and what he wants from Tosca is what she isn’t willing to give.

The problem Tosca faces is thorny, but crystal clear: “How do I get Mario out of jail alive, without yielding to Scarpia?” The “resolution” of this problem illustrates what game theorists call a prisoner’s dilemma, in which two adversaries would benefit from cooperating, but are tempted to betray each other. In Puccini’s opera, Tosca promises Scarpia that, if he saves Mario by staging a mock execution, she’ll give herself to him; but when the time comes, she kills Scarpia instead. Alas, Scarpia also betrayed Tosca: Mario’s execution was real. A crisp problem, a disastrous “solution”—this makes for great tragedy.

Most business problems are not this neatly defined. Thankfully, they don’t end quite as badly, either. Formulating problems this sharply, however, is a

valuable discipline. Before stating the core question, a problem solver must ask five questions that Tosca's situation illustrates—and that spell the acronym TOSCA:

- *Trouble*: What makes this problem real and present? (Mario's arrest)
- *Owner*: Whose problem is this? (Tosca's)
- *Success criteria*: What will success look like, and when? (Escape)
- *Constraints*: What are the limits on the solution space (e.g., resources, timeline, and context)? (Virtue)
- *Actors*: Who has a say in the way we solve this problem, and what do they want? (Scarpia, who wants a night with Tosca.)

Once we answer the TOSCA questions, it becomes possible to state the core question that will guide the problem-solving effort.

To illustrate and explain this process, let's revert to the example we used in Chap. 2 to show the perils of a flawed problem definition: the music industry facing digital piracy.

Trouble: What Makes This Problem Real and Present?

The reason any problem solver embarks on a problem-solving journey (and perhaps the reason you picked this book) is a perceived problem or opportunity. We call this initial perception the “trouble,” to distinguish it from the real problem that will emerge from the problem statement phase.

The basic definition of “trouble” is a gap between an observation and an aspiration. If you aspire to grow revenues and revenues decline, that is trouble. But by the same definition, “trouble” can also be a perceived opportunity. For instance, if your revenues are growing 10 percent a year and you have reason to believe it is possible to double that growth rate. It's the discrepancy between aspiration and reality, the dissatisfaction, which defines trouble.

This definition implies that both terms—the aspiration and the reality—must be defined carefully. Consider the case of the music industry. The trouble seems obvious—people are illegally downloading millions of files. But what is the aspiration? Is it zero illegal downloads? Or fewer illegal downloads than legal ones? This question seems odd for an apparent reason: a music industry executive doesn't define the aspirations in terms of downloads—whether they're legal or not. The executive defines them in terms of revenues.

Around 1999, the revenues of the music industry are still growing at a healthy clip. Illegal downloads aren't, in and of themselves, the trouble. They are a symptom, or more accurately a precursor, of future trouble: future revenue decline. The music industry probably has a revenue growth aspiration, and it is rightly concerned about the impact that illegal downloads will have on future revenues.

As this example shows, the trouble isn't always as obvious as it seems. A few tips are helpful in formulating it:

- *Be specific.* Don't accept "fake problems"—vague gripes you can't possibly "solve." If you're running a customer service center, "we must create a results-oriented culture" isn't trouble, nor a problem that can be solved. "Twenty percent of customer calls remain unanswered" is trouble, and may (or may not) be the symptom of a culture problem.
- *Don't let interpretation (or solution ideas) creep into your definition of "trouble."* For instance, someone who says "Our product has lost consumer appeal" is providing an interpretation, not describing a symptom. A description that stays at the symptom level might be, for instance, "Our product has lost five points of market share over the past year." This matters because the problem may not be the product's appeal: the market share loss may be due to many issues, such as a competitor's moves or a decline in distribution.
- *Ask "Why now?"* If the gap between reality and aspirations is a generic, eternal one—for example, "We would like to increase revenues," it is unlikely to provide a good basis for a problem statement. When the "trouble" as formulated would have been the same five years before and would be the same five years hence, asking why it hasn't been acted upon before and why it has become pressing now will often reveal valuable insights.

Is "Trouble" a Diagnosis?

When we ask about the trouble, we focus on the symptoms. Some readers—both practitioners and experts—may be surprised by this approach: a common prescription in problem solving is to go *beyond* the symptoms to interpret the problem's cause. In many types of problem solving, this step—diagnosis—is a crucial one: doctors diagnose a disease before prescribing a treatment, and consultants often borrow medical vocabulary and start projects with a diagnostic phase.

This approach, however, isn't the only possible one. Presuming you can make a diagnosis is equivalent to adopting a hypothesis-driven problem-solving approach. A diagnosis is much more than a problem statement: the diagnostician may claim he is just defining the problem, but he's already offering a solution.

For a physician, this makes sense. Diseases belong to pre-existing categories, which are continuously updated in medical reference books. A physician's education consists, in part, of learning to recognize them—and then to verify, through clinical examination or tests, that her initial diagnosis is correct before recommending a treatment. Pattern recognition and hypothesis testing are appropriate problem-solving approaches in medicine.

But is it always the case in business problems? When we attempt to diagnose business problems, we implicitly assume they're like diseases, and just like physicians, we can classify them into preexisting, universal "pathologies." Often, this assumption is justified. For instance, a sudden and unexplained dip in your bank account may be symptomatic of a pathology called theft. Likewise, unwanted variability in a manufacturing process is a well-specified, measurable problem, and has a finite number of possible causes. But the medical analogy becomes misleading, and the hypothesis-driven approach detrimental, when problems are more complex. A defining feature of complex problems, in business and elsewhere, is that, unlike diseases, they don't always belong to well-defined, preexisting categories with recognizable symptoms and proven therapies.

This distinction is critical for problem solvers. The observation of trouble is the first step in stating the problem. It's an excellent time to ask yourself whether you're dealing with a well-known problem that belongs to a recognizable class of situations that calls for standardized remedies. If so, you'll continue to state the problem with a clear candidate solution already in mind. You'll take a hypothesis-driven approach, and proceed down the first column in the 4S flowchart (Fig. 3.1). If you don't have a strong hypothesis about a possible reason for the perceived trouble, you should keep an open mind.

The risk is in wrongly believing you recognize a problem you know, just like a physician misdiagnosing a disease. This is the "flawed problem definition" trap into which the music industry fell: it falsely diagnosed illegal file-sharing as one more instance of a well-known pathology called piracy. The industry couldn't see what aspect of the problem was new and how to tackle it. Many experienced problem solvers fall into this trap. Perversely, the more experienced we are, the more likely we are to recognize a new situation as a familiar one and make a misguided diagnosis.

Therefore, the bar for adopting a hypothesis-driven approach should be a high one. You should define trouble as a symptom, without attribution to a known diagnosis, unless you have reason to be confident that you can make an appropriate diagnosis (and propose a candidate solution). More simply: when in doubt, stick to trouble.

Owner: Whose Problem Is This?

Observing symptoms leads to the “owner” question: whose job is it to take care of the symptoms? This question is sometimes obvious: no one but Tosca will bother to save Mario. Conversely, some issues—including serious ones—are no one’s problem. Many people, for example, believe that global business today is dominated by a culture of short-termism that presents a real danger for capitalism and society, but that is a problem without a clear owner. Often, political and societal problems of this sort don’t have a clear owner or have multiple owners with conflicting, irreconcilable objectives. Such problems are sometimes called “wicked” problems and don’t lend themselves to the problem-solving approach described here.¹

Few business problems, however, are “wicked.” Most problems don’t have a single, obvious owner, but they can be stated *from the perspective* of a particular owner; and that choice will bear on the way the problem is defined. In our example, whose problem is piracy? Who, exactly, is “the music industry?” Are we talking about the RIAA, the industry body that represents it? The RIAA has a mission to further the industry’s interests, and as a neutral party, it represents all players equally in theory. But its mission is lobbying, and not much else. Its resources and skills most likely match its mission. If it is the problem owner, the solution space is limited to things it can do—lobbying, advertising, and so on. The RIAA may well define success as “doing something about piracy that gets my members off my back.”

Let’s assume the problem owner is the head of one of the large music labels. The range of possibilities now feels very different. Lobbying is still an option, but a label, alone or with other parties, can do many more things, such as changing its product and pricing strategy, or launching new business models. If you discuss problem definition with this owner, they will probably define success as “doing something that saves *my* business from the deadly threat posed by piracy—whether or not the rest of the industry follows suit.”

As this example shows, asking who owns the problem can sometimes lead to an illuminating discussion. The identity of the problem owner shapes the potential solution space, and hence the problem definition. In practice, you

rarely have a choice: either you own the problem or someone is posing the challenge to you. But in all cases, being explicit about problem ownership is essential.

Identifying the problem owner has another important consequence. As we'll see, problem definition is an iterative process, and only stops when you reach a definition that is "good enough." Likewise, later in the process, we'll consider the problem solved when the solution is "good enough." But who will be the judge of a "good enough" problem statement or solution? The problem owner. If you don't know who's on the hook to solve the problem, you can never define or solve it.

Success Criteria: What Will Success Look Like, and When?

One virtue of having identified a problem owner is that there is someone you can ask the most critical question in problem solving: *what do you want?*

This question is unlikely to elicit the answers you hope for—at least until you probe. Let's assume from here onward that the owner of the music industry problem is the top management team of one of the record labels. Had you asked them, back in 1999, "what they wanted," they would probably have answered something like "stop this piracy." However, this is merely a restatement of the "trouble," the situation that triggered the question. It doesn't get to the actual objectives the owners are pursuing.

The standard advice to get past this point is to ask "why" as often as needed (typically five times). This can be tricky. Trying it with a music industry executive in 1999 might trigger this conversation:

"We want to stop this piracy!"

"Why?"

"To protect our sales, of course."

"Why?"

"Because kids who download free music don't buy our CDs anymore. Are you dumb?"

"Why?"

"Get the hell out of here!"

This doesn't mean that probing for reasons is irrelevant: *why* kids prefer to download music rather than buy CDs is a crucial issue (and price isn't the only answer). But asking why isn't always specific enough to get to the right question.

A more productive way to ask “why” is to ask *what success will look like*. An effective way to ask this question is as follows: “*We are in the future and this problem-solving effort has been a great success. What is the date, and what do we see?*” This creates an open-ended discussion about *success criteria*. Let’s role-play this approach with the same music industry executive:

“We are in the future, having dinner together to reflect on this project and celebrate its success. What is the date?”

“Well, I guess it’s at least three years from now. We aren’t going to solve this problem overnight, are we?”

“I guess not. And how do we know we’ve succeeded?”

“Well, if we’re having dinner together, it means I still have a job, for starters!”

“Good to know dinner will be on you. What else?”

“We’ve solved the piracy problem.”

“Sure. But how do you know? How do you measure it?”

“Obviously, our revenues are growing again. If we’ve stopped this piracy, we’ve restored the growth trajectory we were on before it started.”

With this simple line of questioning, you got somewhere: the critical success metric is revenues. It’s not how many files are downloaded, or how many people are thrown into jail for sharing them. This approach leaves room for a very different—and more productive—discussion.

Another benefit of asking the question in this way is that it will call out a frequent (and shoddy) practice: defining the problem by its proposed solution. Here, an executive might say, “The problem is that we need to make it more difficult to download pirated CDs,” or even “We need to raise our game on theft prevention.” This frequent mistake leads straight into the “solution confirmation pitfall” we illustrated in Chap. 2 with the Grameen–Danone story: when we define a problem so it suggests the solution, we’re in great danger of blindly confirming that conclusion.

As with many mistakes, problem owners usually commit this one with the best intentions. Isn’t that what we call being “results-oriented”? Don’t good bosses instruct employees to “come to us with solutions, not with problems”? And yet, to properly define a problem, we must resist the urge to solve it too quickly. To focus on stating the problem, we need first to *ignore* the possible solutions. Asking the “success criteria” question is a tool to do just that.

An important question when considering success criteria is whether they should include a specific, quantifiable target. When such a number is chosen, there’s always a lot of discussion around it, and for good reason: it can be difficult at the beginning of the problem definition process to set an ambitious

yet realistic aspiration level. In our example, if we continued the conversation with the music industry executive, the executive might be compelled to specify an annual revenue growth of 5–7 percent as the target. This is an understandable, but probably unrealistic, perspective. In hindsight, no strategy could have maintained the music industry's previous trajectory of growth and profitability in the face of digital disruption.

Because of this difficulty, there are two schools of thought on whether targets should be quantified. One approach is to give up quantification and formulate the question with an open-ended target that recognizes uncertainty. In the music example, such a target might be “maximize revenues while maintaining ROS” (return on sales—profit as a percentage of sales).

Some argue, however, that in an organizational context, this may not be enough to push the problem-solving squad to think hard enough. Picking an aspirational number, even if it is arbitrary, has real benefits. It focuses the mind, stretches the thinking, justifies allocating resources to the problem-solving effort, and generally raises aspirations. This approach is often attractive, provided you're prepared to revisit your target as you discover new facts—the initial target may have been set too high or too low.

Constraints: What Are the Limitations and Trade-Offs?

Let's assume you've identified the Trouble, Owner, and Success criteria. A picture of the problem you're trying to solve is emerging. But in solving any problem, there are limits to what you can do. Be aware of these constraints.

You should define constraints from the perspective of the problem owner. But asking the question “Do you face any constraints?” will probably not get you far. A practical approach is to consider three types of constraints.

First, there are always *constraints on the success criteria*, arising from conflicts with other objectives and commitments. Although achieving success as you define it is your primary objective, it's rarely your only one. For instance, when the record label defines revenues as the critical success metric, it implicitly assumes it must maintain a minimum level of profitability. For this year, and perhaps the next one, the label may be committed to achieving specific revenue and profit targets. Such commitments are a constraint on the possible solution. These constraints qualify the success criteria—success is success, but it can't be achieved at all costs. You should identify these trade-offs as early as possible in the process.

Second, the owner's resources and capabilities may also impose *constraints on the solution*. We mentioned that if the owner were the RIAA, its limited capabilities would rule out certain types of solutions. With a record label as owner, capability constraints are different but important. One capability gap is the absence of any "digital" skills in the organization.

Third, there are often *constraints on the problem-solving process* itself: for instance, limitations to the time and budget that can be devoted to solving the problem, or confidentiality constraints that prevent you from accessing people and information you would like to involve. Setting up a visible problem-solving effort may sometimes worsen the problem by giving it visibility. "Panicky music industry sets up emergency task force to address rampant piracy" is a headline the problem owner doesn't want to generate.

Discussing constraints early on can save you time and effort. Exploring solutions only to discover later they're incompatible with constraints you hadn't identified earlier can be costly. But take your initial discussion about constraints with a grain of salt. If you identify too many constraints, you may end up defining the problem as "just making the trouble disappear without changing anything else." That's usually mission impossible. Few problems would get solved if their owners didn't, at some point, relax some of the constraints they face.

The music industry, for instance, should realize that restoring past levels of revenue growth *and* profitability is unrealistic. But this realization will probably not happen in the first discussion. The need to reevaluate constraints, and possibly to relax them, is one of the reasons to revisit the problem definition periodically throughout the problem-solving process.

Actors: Who Are the Stakeholders?

Finally, the "owner" is usually not the sole person dealing with a problem and its consequences. The owner must contend with other stakeholders, who rarely have the same objectives (i.e., success criteria). This might be just another constraint to deal with, but because it's a crucial element of the problem definition, it's worth treating it separately. While constraints are usually stable or at least predictable, actors are reactive: they can deliberately respond to our recommendations in supportive or detrimental ways. It's therefore indispensable to understand their objectives and the stakes they have in the problem.

Stakeholder analysis is a useful technique to identify, systematically, the stakeholders and their objectives. The music industry, for instance, invested considerable time and effort trying to get Congress to pass legislation to crack down on illegal file-sharing. A stakeholder analysis might have revealed that few members of Congress wanted to be seen as persecuting teenagers and stifling technological innovation, just so the music industry could continue to sell CDs at \$14 apiece. A contributing factor was that the music industry had done nothing to endear itself to lawmakers, having fiercely resisted their attempts to police explicit lyrics.

Write the Core Question

Now that you've completed the five TOSCA steps, you can write the *core question* you'll answer. It should be a question, not a statement—"We must stop piracy" isn't a question. But beyond this simple requirement, there are many degrees of freedom in formulating the core question, and there is usually no single "best" way to define a problem.

The essential choice you must make at this stage is *question scope*. Like any question, a problem definition can be an open question ("How can we stop piracy?") or a closed one ("Should we launch a music download service?"). A closed question entails a narrower scope than an open one. However, the critical choice isn't the grammatical form of the question but its scope.

Take, for example, a company considering an acquisition to enter a new market. An obvious (closed) question would be: "Should we proceed with this deal, at the price and terms currently offered?" A slightly broader scope, still with a closed question, would be: "What is the maximum price we would be prepared to pay for this company?" But you could also open the aperture of your inquiry by asking, for instance, "What approaches are possible, including but not limited to acquiring Company X, to enter this market?" Whether you phrase it in this open way or as a closed question ("Is acquiring Company X the best way to enter this market?"), you will have broadened the question scope considerably.

The harder the problem is and the earlier you are in the problem-solving process, the more likely it is that a broad question scope will be preferable. But the only absolute requirement is that the question scope follows the TOSCA steps you've listed. These steps can serve as a checklist to verify that the core question meets the tests of a good problem definition:

- Does the question address the *trouble* that got you to consider the problem in the first place? In the music industry case, the symptoms include pirated downloads, but also the fast-growing availability of broadband Internet and the emergence of various providers of digital playback devices. A problem definition that doesn't mention these symptoms would be a generic question about growth and profitability, not a statement of the urgent problem you're facing.
- Is the question phrased from the perspective of the *owner*? For instance, asking "Why do teenagers download music illegally?" is a broad, interesting, important, and relatively difficult question. It will play a part in the problem-solving process, as we'll see when we discuss problem structuring. But it's not asked from the perspective of the music industry executive. It can't be our problem statement.
- Would answering this question meet *success criteria*? This is equivalent to asking whether the question reflects the specific metrics and time horizon in your success criteria. For instance, "How can we face the digital music threat?" doesn't explicitly address the success criteria. To do so, you might ask a question that starts with "How can we restore an x-percent revenue growth rate in three years' time?"
- Does the question recognize the *constraints*? As just phrased, for instance, it doesn't: sharply cutting CD prices would probably result in revenue growth, but would violate a key constraint. Adding a profitability constraint to the problem definition question addresses this.
- Does the question consider relevant *actors*? It is usually impractical to list all the stakeholders in a core question, but you must identify the key players. Here, for instance, one prominent "actor" whose behavior matters is the consumer illegally downloading music.

Going through this list might lead you to a core question like this: "In a context where young consumers are increasingly downloading pirated music files, and knowing that enablers of that behavior—broadband access and digital playback device—are bound to become more accessible, what actions can we take that would result in restoring an X-percent revenue growth rate, with a minimum return on sales of Y percent, in three years' time?"

Figure 4.1 offers a worksheet for applying the TOSCA checklist to your problem.

| | |
|--|---|
| TROUBLE | <p><i>What are the symptoms that make this problem real and present? (Be specific; avoid interpretation or solution ideas; ask "why now?")</i></p>  |
| OWNER | <p><i>Whose problem is this?</i></p>  |
| SUCCESS CRITERIA | <p><i>What will success look like, and by when? (Include quantified target if possible.)</i></p>  |
| CONSTRAINTS | <p><i>Are there prior commitments or conflicting objectives, resource constraints on the solution, or constraints on the problem-solving process?</i></p>  |
| ACTORS | <p><i>Which other stakeholders have a say, and what do they want?</i></p>  |
| ↓ | |
| CORE QUESTION TO BE ANSWERED | |
| <p><i>Reflects a clear choice of scope, and is consistent with TOSCA checklist, i.e., addresses Trouble, is phrased from Owner perspective, states Success criteria, recognizes Constraints, and identifies relevant Actors.</i></p>  | |

Fig. 4.1 TOSCA problem statement worksheet

Singing TOSCA as a Choir

In describing the steps in the problem definition process, we've made them appear sequential and reasonably straightforward: spot the trouble, verify the owner, define success, identify constraints, list actors—and then, write the question. In reality, the process is more complicated.

The first reason is easy to see: the steps aren't sequential, but overlap with one another. It's hard to define "trouble" accurately without an identification of the owner. It's difficult to determine success without acknowledging the constraints. Identifying actors you initially neglected may lead you to revisit success criteria, and so on. You'll probably need to write your core question not just once, but several times, and come back to the problem owner several times to test it. When the owner agrees that the problem will be solved if you bring the answer to that question, you'll know you have a solid problem statement.

But rewriting the question doesn't stop at the problem statement phase. Your problem owner, who decides if your problem definition is relevant, may change her mind—because of new facts you bring, or because she thinks about the same facts differently. Stating the problem is an iterative process of discovering or shaping a question, which doesn't stop when the solving phase begins.

This iterative process isn't performed alone by a problem solver holding a problem statement worksheet and a pen. For each component of TOSCA, many people will have different opinions, insights, and perspectives to contribute. These views may be complementary or entirely at odds with one another, but they are bound to differ. To state a problem well, you must integrate these various perspectives into your problem statement.

To do so isn't an additional step in the problem statement process. It's a mindset—an attitude of openness, an ability to see the same problem simultaneously from several angles. Design thinking experts call it "empathy": to state a meaningful problem, you must put yourself in the shoes of various constituents, including the problem owner, and the other actors whose worldviews, choices, and behaviors shape the problem and may contribute to its solution.

To illustrate this, let's look at the music industry again. From the perspective of the recording label executive, the problem is well defined. But what does it look like to a college student storing illegally downloaded files on a computer hard drive in his dorm room? Asking that question would reveal that the price of CDs, although a component of the problem, isn't the whole issue. At their peak, Napster and other file-sharing sites provided a user experience—immediacy, ease of use, excitement—that was hard to match. These consumers also were frustrated at the time by the lack of user-friendly devices to play digital music away from their computers. Apple's iPod and iTunes later

succeeded primarily because they offered an attractive user experience (and the music industry's attempts to provide legal downloading alternatives failed because they provided a ridiculously unwieldy one). Without empathizing with all the actors, the problem statement can't be complete.

At the problem statement stage of the 4S process, a practical way to “empathize” (i.e., to see the problem from the points of view of several stakeholders) is to contact them and ask them how they view the problem. Usually, this means conducting problem definition interviews: early in the problem-solving process, you can meet multiple stakeholders, and ask them the TOSCA questions. What is the trouble, in *your* view? Whose problem do *you* think it is? What would success look like *to you*? And so on. The more diverse the interviewees and their viewpoints, the better.

However, interviews are sometimes not enough to get to real needs and wants, particularly if some stakeholders are unwilling or unable to express them. Recall, for instance, the story of Ron Johnson in Chap. 2, and his admission that he didn't understand the needs and preferences of J.C. Penney shoppers, particularly regarding promotions. In such cases, careful observation of the actors' behaviors may reveal their needs and expectations. Design thinking practitioners call this approach *immersion*. Whenever you don't know enough about actors to state the problem, consider putting yourself in their shoes—or immersing yourself in their situation. We will further discuss empathy and immersion techniques in Chap. 8.

Whether it is achieved through problem definition interviews or immersive techniques, empathizing with stakeholders will be beneficial in several ways. It will help you refine your problem statement, the primary objective. But it will also give you ideas for the next stage in the problem-solving process, as interviewees will almost always volunteer what they think the solution is. Finally, it may be a good way to build goodwill with stakeholders.

* * *

If you take only one thing away from this chapter, it should be that there is no “right” problem definition, although there are many wrong ones. Each coherent problem statement (i.e., each problem definition question consistent with a TOSCA set) is a frame on the problem, and multiple frames can coexist. Crafting a problem statement is an iterative process, and a collegial effort involving various stakeholders with different perspectives. As the problem-solving effort proceeds and you gather facts and generate options, you'll continue to refine your problem statement. The structuring stage of the problem-solving process, which we'll examine next, will be instrumental in helping you reflect on and refine your problem statement.

Chapter 4 in One Page

- To state the problem, use the *TOSCA checklist* to formulate the *core question*.
- *Trouble*: a gap between a situation and an aspiration (problem or opportunity):
 - Defined in specific terms, including the answer to the “Why now?” question
 - Usually a symptom, not an interpretation or a diagnosis
 - *Music industry: trouble = declining sales, not “piracy”*
- *Owner*: the person asking for this problem to be solved and who will judge a good problem definition and a good solution:
 - *Music: is the “owner” the industry or the CEO of one music label?*
- *Success criteria*: spell out “what success will look like”:
 - Should not predefine the solution
 - May include a quantified target (if you’re prepared to revise it)
 - *“We are in the future and the problem has been solved. What is the date, and what do we see?”*
- *Constraints* include:
 - Preexisting objectives and commitments that constrain the success criteria
 - Resource and capability limitations that constrain the solution scope
 - Time, budget, skill, or confidentiality issues that constrain the problem-solving process
- *Actors* are important stakeholders whose objectives you must understand.
- *The core question* can be open or closed, and its scope can be narrow or broad, but it must be compatible with all the elements of TOSCA.
- Problem statement is iterative and collaborative:
 - Empathize with stakeholders through problem-definition interviews and immersion
 - Integrate the perspectives of multiple actors
 - Revisit the problem statement periodically throughout the problem-solving process