



10 Procedure Pitfalls

(and how to fix them)

Procedures only work if people can follow them. After 20+ years in utility operations, including time as a Navy Reactor Operator and 300+ field validations, I've seen the same pitfalls appear across industries, facilities, and procedure programs of every size. These are real patterns that real workers encounter every day. This guide covers the 10 most common procedure pitfalls I encountered, why they matter, and how to fix them.

1. The Guidance Gap

What is it?	How do I fix it?
When a procedure says "adjust as necessary" or "verify as applicable," it's not being flexible... it's creating a pitfall. Vague conditional language pushes workers from rule-based performance into knowledge-based performance, where error rates are significantly higher. In high-risk industries, "figure it out in the field" is never an acceptable procedure design strategy.	Replace vague language with specific criteria. Tell the worker exactly what to do, under what conditions, and what the expected result should be.

2. The Missing IF/THEN

What is it?	How do I fix it?
Decision points are inevitable in complex operations. But when a procedure presents a decision without a clear conditional structure, workers are left to interpret rather than execute. An IF/THEN or WHEN/THEN structure tells the worker exactly what condition triggers what action. Without it, two workers following the same procedure can make two completely different decisions... both convinced they're right.	Every decision point needs a conditional structure. IF this condition exists, THEN take this action. No interpretation required.

3. The Assumption Trap

What is it?	How do I fix it?
A procedure step should never assume the worker already knows something critical. When detail is omitted because "everyone knows that," you've created an assumption trap. Procedures should be written for a qualified but inexperienced user with no direct supervision. That's the standard. If a step requires knowledge that isn't captured in the document, it's an error waiting to happen, especially during shift changes, abnormal conditions, or when a less experienced worker is following the procedure for the first time.	Write for the least experienced qualified user. If it's critical to executing the step correctly, it belongs in the procedure.



4. Step Overload

What is it?	How do I fix it?
One step, one action. It sounds simple because it is. When a single procedure step contains multiple unrelated actions, workers are more likely to miss one. The eye moves to the next step, the checkmark goes in the box, and the second action never happens. In high-risk operations, that missed action can have serious consequences.	Break multi-action steps into individual steps. Two actions are only acceptable in a single step when they must be performed simultaneously to achieve a single result.

5. The Passive Voice Problem

What is it?	How do I fix it?
Procedure steps should be short, active, and imperative. "The valve should be opened" is not a procedure step, just a suggestion. Passive voice creates ambiguity about who is responsible for the action and whether it's mandatory. In high-stakes environments, ambiguity is a liability. Workers need clear, direct instructions that leave no room for interpretation.	Write action steps as short imperative sentences with a strong action verb. "Open valve V-101." That's clear, direct, and unambiguous.

6. Hidden Actions

What is it?	How do I fix it?
Precautions, cautions, warnings, and notes have one job: to inform. They are never the place for actions. When an action is buried in a caution or warning, workers following the procedure sequentially can easily miss it. Precautions are often skimmed once and never revisited. If it requires the worker to do something, it belongs in a procedure step.	Audit every caution, warning, note, and precaution in your procedures. If it contains an explicit or implied action, move it to a procedure step immediately.

7. The Rabbit Hole

What is it?	How do I fix it?
Branching and referencing are sometimes necessary. But when a worker has to jump between multiple documents, attachments, and references to complete a single task, you've created a procedural rabbit hole. Every branch is an opportunity for a worker to lose their place, skip a step, or return to the wrong location in the original document. The more complex the navigation, the higher the error rate.	Minimize branching wherever possible. When referencing is necessary, make the return path crystal clear. If a procedure requires excessive navigation, consider whether it needs to be restructured entirely.



8. Lost in Translation

What is it?	How do I fix it?
Terminology matters. Using two different words to mean the same thing, relying on facility slang, or introducing uncommon terms forces workers into knowledge-based performance trying to decode the language rather than execute the task. Consistency in terminology isn't just a style preference... it's a human factors requirement. Workers build mental models around the words they see repeatedly. Inconsistency breaks those models.	Establish a standard terminology list and use it consistently across all procedures. When in doubt, use the most commonly understood term (not the most technical one).

9. The Consistency Problem

What is it?	How do I fix it?
Inconsistent format, layout, and writing style isn't just aesthetically displeasing. Workers develop expectations about where information lives and how steps are structured. When those expectations are violated, cognitive load increases and errors follow. This is especially problematic in large procedure programs where multiple writers contribute over time without a strong writer's guide to anchor the work.	Develop and enforce a robust procedure writer's guide that standardizes format, layout, typography, numbering, and writing style across your entire program. Consistency is a safety issue, not a preference.

10. Procedural Bloat

What is it?	How do I fix it?
More is not better. When procedures are packed with boilerplate language, redundant administrative information, and non-value added content, workers start skimming... and skimming leads to missed steps. Every piece of information in a procedure should earn its place. If it doesn't directly support the worker in safely and correctly executing the task, it's bloat. Bloat erodes the credibility and usability of your entire procedure program over time.	Ruthlessly edit. If a piece of information doesn't add value for the worker executing the task, remove it. Leaner procedures are safer procedures.

Ready to Audit Your Procedure Program?

If any of these pitfalls sound familiar, you're not alone. At Praxis Procedures, we bring 20+ years of operational experience to procedure program assessments, human-factored rewrites, and program development. Whether you need a single procedure revised or an entire program overhauled, let's talk.

Want to understand the science behind these pitfalls? Download our companion resource: *Human Factors Reference Guide* at www.praxisprocedures.com.