"Using the STAMP systems-based approach to identify hazards for the Transient Operating State: What is it and How can it help us?"

aeSolutions Technical Team

Prepared for Presentation at
American Institute of Chemical Engineers
2022 Spring Meeting and 18th Global Congress on Process Safety
San Antonio, TX
April 10 – April 14, 2022

“Using the STAMP systems-based approach to identify hazards for the Transient Operating State: What is it and How can it help us?”

aeSolutions Technical Team

Keywords: Accident Models, Systems thinking, Emergence, STAMP, Safety vs. Reliability

Abstract

STAMP (Systems Theoretic Accident Model and Processes) is a relatively new accident causality model based on systems theory. It draws its main tenets from systems thinking that (1) accidents can happen even when there has been no failure, (2) that interactions between components of the system create emergent properties that can lead to accidents, and (3) it treats safety as a control problem rather than a failure problem. STPA (Systems Theoretic Process Analysis) or colloquially “Stuff That Prevents Accidents” is a powerful hazard analysis technique based on STAMP. The STPA technique is based on a control structure rather than a traditional hardware-based structure as typically shown on a P&ID (Piping & Instrumentation Diagram). STPA is not so concerned with identifying component failures, but rather how those components interact and what controls or constraints are placed on the interactions and how they can be violated, which can lead to accidents.

The STPA technique is a good fit for identifying the ways hazards can arise during transient operating states such as maintenance, start-up, or response to abnormal situation. It identifies unsafe or missing controls related to the transient mode needed to prevent an accident. It works off of a control structure of the transient mode versus procedures or P&IDs. A typical control structure can include components, humans, software, requirements, expectations (written and unwritten). Traditional PHA (Process Hazards Analysis) methods such as HAZOP or What-if will not provide the same perspective.

This paper will provide two examples of transient mode control structures, one for maintenance and one for response to abnormal situation, and show how to perform the STPA hazard analysis on those control structures to ensure the proper controls and constraints are identified to prevent an unwanted event.

1 Introduction

Professor Nancy Leveson of MIT created the accident model STAMP (Systems Theoretic Accident Model and Processes) in the 2000's and published a book on it in 2012 [1]. Since that time the adoption of the model and its primary tool of implementation STPA (Systems Theoretic Process Analysis) has been rapid and widespread. Multiple industries ranging from Aviation to Health Care all over the world are now using the STAMP framework as a basis for their hazard analysis. The Process Industries have been slower in the uptake of STAMP (but we're curious). Is it because we already have adequate models and tools? Certainly we have plenty of tools and activities like PHA, LOPA, IPL verification, audits, assessments, Fault-Tree, Bow-tie, etc. But what accident models do we claim to guide our safety work (before the fact, not after an accident has already happened)? That is how we will begin, with a brief discussion of the accident model that influenced the creation of STAMP, before we consider its implementation via STPA.

2 Accident Models

Safety Science has recently produced new ways of thinking about how accidents happen. For example, STAMP which is the subject of this paper, and also FRAM (Functional Resonance Accident Model) [2]. Both models utilize the principles of systems thinking (to be discussed). These newer models can be traced back even further, to whom many consider as the grandfather of modern safety science, Jens Rasmussen, the Danish safety pioneer who worked across all fields in safety including systems safety, human factors, human reliability, risk management, control of major accident hazards, etc. It is his model shown in **Figure 1** that has been so influential in thinking about how accidents happen.

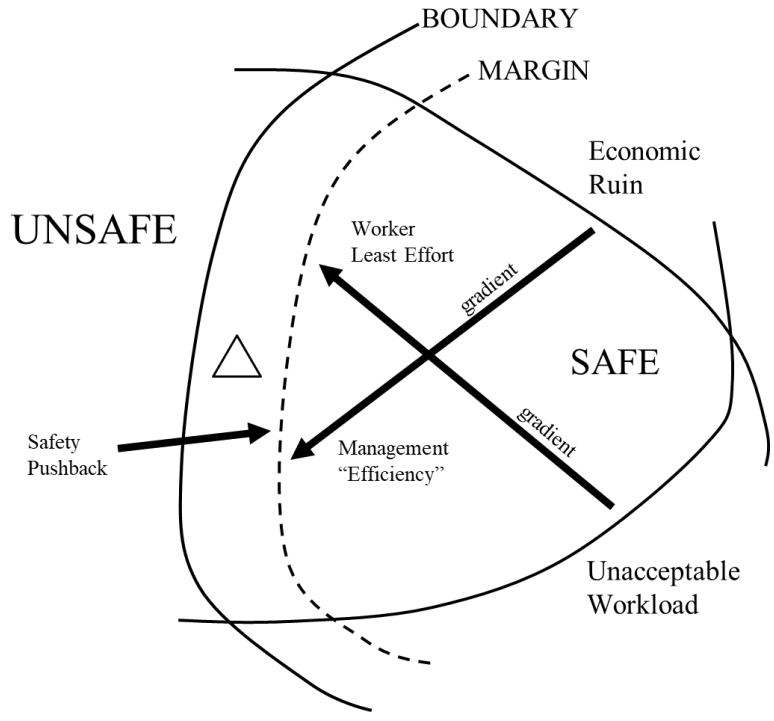

Figure 1. Rasmussen's "Migration toward the Boundary" Model [3].

2.1 Brief discussion of Rasmussen's "Migration toward the Boundary" Model

Rasmussen's model incorporates at least three distinguishing features that makes it unique from more traditional and linear "chain-of-events," "falling dominoes," or "swiss-cheese barrier" type accident models.

1. Drift – pressure gradients produce drift toward the boundary
2. Time – the drift occurs over time
3. Humans and the Organization – these are the gradients
4. Boundary – between safe and unsafe

STAMP incorporates several features of Rasmussen's model including extending the system boundary (i.e., go broad), include humans and organization, and provide constraints and feedback (i.e., control) to the drift gradients to prevent reaching the boundary. Safety resides in the constraints (see **Fig. 9** for some examples). The system boundary is expanded beyond potential proximal causes (i.e., front line workers), to include conditions (e.g., design errors or maintenance deficiencies, etc.) as well as systemic factors (e.g., management decisions, employee turnover, engagement, etc.).

2.2 STAMP and Systems Theory

The creation of STAMP was also influenced by Systems Theory (or systems thinking). Systems thinking originated in the 1950's as a way to holistically analyze problems [4]. This is opposite to what is called Reductionism, which is when a system is broken down, and the parts analyzed individually. Reductionism is how most engineering activities occur. Systems thinking holds, that when the system is broken down for analysis (ie, structural decomposition), some properties that belong to the system will be lost. The properties that are lost, are called emergent properties, i.e., they are properties of the system, not the parts of the system. The way to analyze the emergent properties of a system is by functional abstraction. Emergence has been called the single most fundamental systems idea [4]. Mechanical systems provide the easiest way to visualize emergence. The parts of a car have different properties than the car as a whole (system), for example, the component properties of the brakes vs. the emergent properties of the car such as mobility, or comfort, or style, or even something like environmental footprint. Three elements of systems thinking are described in more detail here.

1. Accidents can happen even when there has been no failure of individual components. High component reliability does not mean a system is safe. Reliability and safety are different properties of a system, where safety is more of an emergent property compared to component reliability. Levenson gives many examples of this in her book [1]. For example, a chemical plant run-away reaction occurred when the high integrity computer software created an unsafe state. The field sensors were not failed, and the computer did

exactly what it was specified to do. It's true enough that there was no component failure, yet the accident still occurred. Still, we recognize that while software doesn't "fail" in the traditional sense, however, as Process Safety professionals we would see this as a deficiency in the software requirements specification to anticipate potential unsafe states. This is a big part of STPA (the tool used to implement STAMP) to identify these unsafe software states and modes.

Another interpretation of how accidents can happen when there has been no failure, comes from Perrow [15] "normal accident theory" and then developed further by Hollnagel [2] as "functional resonance." These are referred to as "system accidents." Here, there are no failures (per se), instead it is the variation (resonance) in normal work processes that when superimposed together in a non-linear (complex and tightly coupled) system produces accidents. An initial reading of these concepts one may conclude this is just semantics. A deeper reading shows there to be some value in distinguishing between the meanings of "failure" and "variation in normal work." The potential for leading indicators based on the variance from 'normal' is one example.

2. Interactions between components of the system create emergent properties including failure. The above example of the interaction between the field hardware and the computer software (neither of which failed) is an example of an emergent failure. Other examples abound. The author knows of a scenario between a process engineer and an MOC coordinator. Both individually did their jobs well by generating and reviewing MOCs, respectively. But over time the process engineer perceived the MOC coordinator as "nitpicking" (confirmed a non-offensive term) his MOC's, and stopped generating them when changes were to be made. Failure emerged only between the interaction of the two. The heart of STPA is to evaluate the interactions between components (as opposed to the components themselves).

3. Systems Theory is ground in classic control theory (ie, control signals with feedback loops). STPA uses this same philosophy of control with constraints on and feedback from the interactions between parts. STPA is a control-theory based hazard analysis technique [1]. Safety resides in the constraints (i.e., control actions). That is why there is only one guideword used in STPA – "unsafe control action" (UCA), which looks at how and why the safety constraints may be violated (or, alternatively can be used to identify what safety constraints are needed).

2.3 The "Nuts and Bolts" of STPA

STPA (Systems Theoretic Process Analysis) is the "tool" used to implement the new way of thinking embodied by STAMP. It is a hazard analysis tool. It has a look and feel similar to HAZOP or FMEA, but there are a few key high-level differences, listed here.

1. STPA has the ability to "go broad" with respect to both the system studied, as well as causes. While HAZOP or FMEA may list (for example) 'human error' as a proximal

cause, STPA has the ability to study the 'human error' as a system including mental models, conditions, and systemic factors related to the potential error.

2. It works top down. You begin by identifying the top event "loss." The loss could be related to safety, environmental, reputation, etc. Or, the loss could be related to the "mission" (for example, to respond, to isolate, to maintain). The loss should be related to the emergent property (caused by interactions between the parts) which you need to control or constrain. In practice, the top event loss will be identified in a PHA (Process Hazard Analysis) type study and then STPA will be invoked to further evaluate the scenario as shown in **Figure 3**. The STPA study is independent of PHA.

3. You need a control structure (drawing). Often this will be a custom drawing made for the system you are analyzing. P&IDs and procedures aren't the primary document that you'll STPA against.

4. You'll analyze the interactions between the elements on the control structure, not the elements themselves (the exception is **Fig. 5** the "Engineering for Humans" model requires evaluating the internal mental models of the operator). You're looking for unsafe interactions that lead to the Loss.

There are some similarities between STPA and traditional methods such as HAZOP and FMEA. These are listed here.

1. It works by Deviation (parameter + guideword). The guideword used is "control action." You're looking for ways the control action is unsafe or otherwise violates the safety constraints.

2. It creates scenarios that lead to the undesired outcome (ie. Loss). But it works backwards (ie, from top event Loss to Cause).

3. It generates recommendations related to preventing the Loss (at the last step).

4. It documents all this in table format similar to a HAZOP or FMEA spreadsheet.

5. It's performed as a facilitated team meeting.

6. In many ways it is HAZOP or FMEA done backwards.

The order of STPA proceeds per below, and as mentioned above uses a top-down approach. All of this information is captured in a table format (see **Fig. 2**).

1. Identify the Loss (top event) to be studied.

2. Draw the control structure (that is used to 'control' or provide 'constraints' to prevent the Loss) and show the interactions between parts (as arrows) as well as call out the specific control actions and feedback. Once the system is defined it is then studied in minutiae.

3. Identify unsafe control actions(UCA), using the following guidewords:

a. Control not provided

b. Control provided in the wrong way (when not needed, too early, too late, out of order, stopped too soon, stopped too late, etc.)

4. Write the "loss scenario" (can be a narrative) by looking for 'causes' of the UCA that lead to the Loss. You first identify the context (mode or state in which the UCA occurs). Then, STPA looks at the "process model" (and/ or "automation model") of the human or hardware or software, (ie, their 'beliefs' as in why would they - or what would make them – take this unsafe action). The 'causes' can be broad, working from proximal, to conditions, to systemic issues.

5. Make recommendations related to UCA, causes, controls, constraints, feedback, and other requirements, etc. to prevent the loss scenario.

LOSS EVENT: Autonomous car crashes

SYSTEM BOUNDARY: Control computer and the controlled vehicle

Control action	**Source**	**Context (Mode or State)**	**UCA1 (Not provided)**	**UCA2 (Provided wrong way)**	**Cause (UCA1)**	**Cause (UCA2)**	**Recommendation**
Brake Cmd given	Computer	When stopping needed	Unsafe – describe consequence	Unsafe (too short) – describe consequenc	Obstructed View, etc. (list all)	Faulty sensor, etc. (list all)	1. Require longer stopping distance 2. Provide sensor redundancy
Next Control action	….	…..	……				

Figure 2. STPA worksheet to capture the team discussion. The worksheet is structured very similar to a HAZOP or FMEA.
UCA = Unsafe Control Action (guideword)

The claim is made that STPA costs much less than traditional methods such as HAZOP and fault-tree, etc. to implement. The MIT group has machine coded structured text related to the deviations, which then (apparently) with some minimal set-up, is able to autogenerate the STPA study. As we all know there is no such thing as a free lunch when attempting to identify hazard scenarios, so we won't waste any more time here on this claim. And from what I can glean watching many hours of STPA tutorials (all of this material is available for free on-line), STPA can be every bit as detailed and time consuming as a traditional HAZOP and FMEA. There are no short-cuts.

3 Applying STPA to Transient Operating States

As mentioned in **Section 2**, STPA has the ability to "go broad" ie, extend the boundary of the study to include what is traditionally outside of a HAZOP or FMEA to evaluate in detail. This includes transient or non-routine operating states such as:

1. Operator Emergency Response to abnormal condition **(Section 3.2)**.
2. Maintenance activity that has process safety implications (for example, through some loss of containment scenario) **(Section 3.2)**.
3. Barrier integrity (ie, safeguard, Independent Protection Layer, etc.) control system **(Section 3.3)**.

This paper will look at all three below. It should be noted here and as described in **Section 2**, STPA starts with (assumes) the top event loss has been identified (beforehand). Losses related to transient operating modes can be identified by a PHA (Process Hazard Analysis). STPA will then be invoked for high hazard transient or non-routine scenarios especially those with a significant human element involved. The STPA will take a "deeper-dive" into the causal scenarios of the loss (ie, will look "behind" the human error). This work-flow is shown in **Figure 3**.

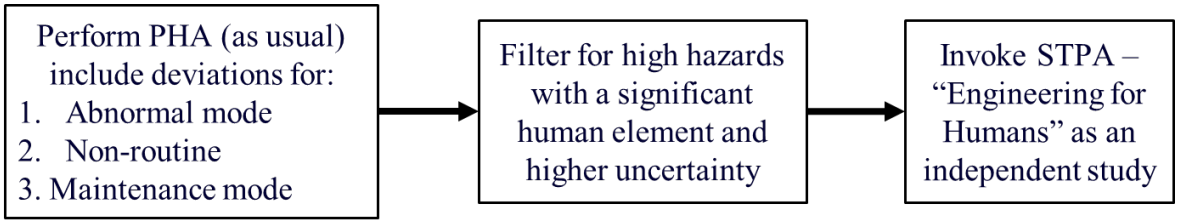

Figure 3. STPA work-flow. Because STPA works "backwards" (i.e., starts with the top event and works down to the causes) you must bring with you the loss or high hazard to evaluate. The traditional PHA can be used to identify high hazard scenarios related to maintenance and other non-routine modes of operation that are primarily human involved. Operator's performing the actual maintenance task is a good example (as opposed to only preparation for maintenance). Another is any maintenance task that has the potential to harm more than just the person (or persons) doing the task.

3.1 Drawing the Control Structure (generic)

The control structure should be created early in the STPA process (before the team meets). The control structure will be adjusted during the team meeting as new information is learned.

A generic control structure is shown in **Figure 4** [6]. Keep in mind that the "controller" can be a human controller interfacing with a control system. The lines with arrows represent generically

inputs and outputs, such as control actions, information, feedback, etc. You'll create deviations for the lines and arrows (missing, wrong, inadequate, etc.).

Figure 5 shows a model that was created specifically for a human controller [6]. It is known as the STPA extension "STPA - Engineering for Humans." It draws heavily from existing human factors type models including for decision making. It's designed to help practitioners develop a richer set of causal scenarios related to human operator behavior – the "why would they" potentially violate the safety constraints of the system. Remember that "causal" does not necessarily mean proximate. It could also include conditions and systemic factors that are correlated (ie, latent or contributory) with unsafe control actions. **Figure 5** is where the lone guideword used in STPA, "Control Action" begins to become inadequate. Other deviations and descriptive conditions will be needed to write the causal scenarios for the "Engineering for Humans" model. These are discussed below.

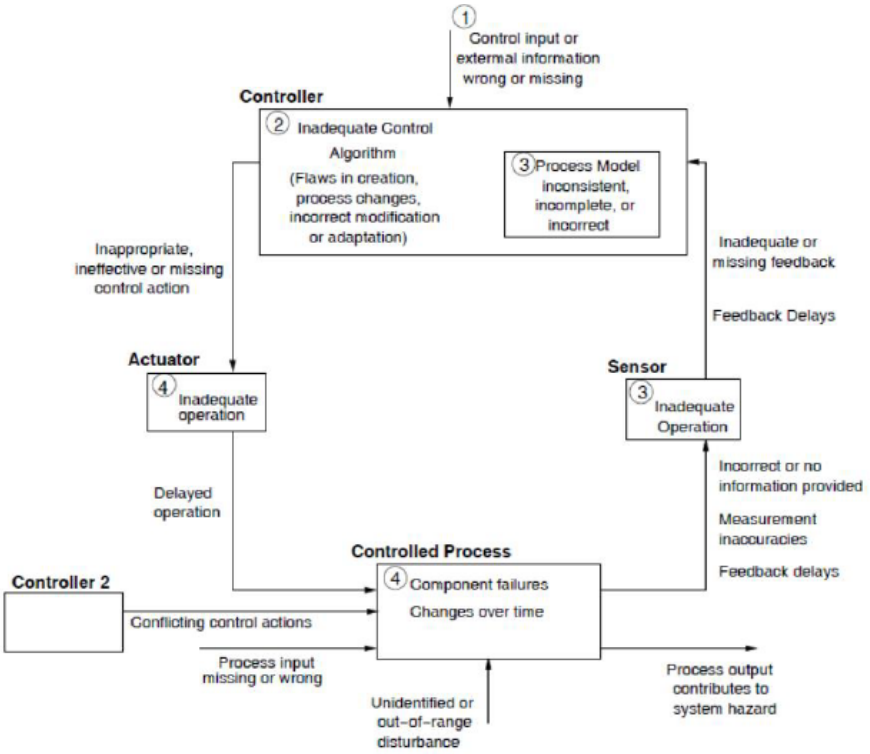

Figure 4. Generic Control Structure Drawing [6]. The "controller" could be a human. The arrows represent control actions, information, and feedback. Deviations are written for unsafe control actions (missing, wrong, etc.)

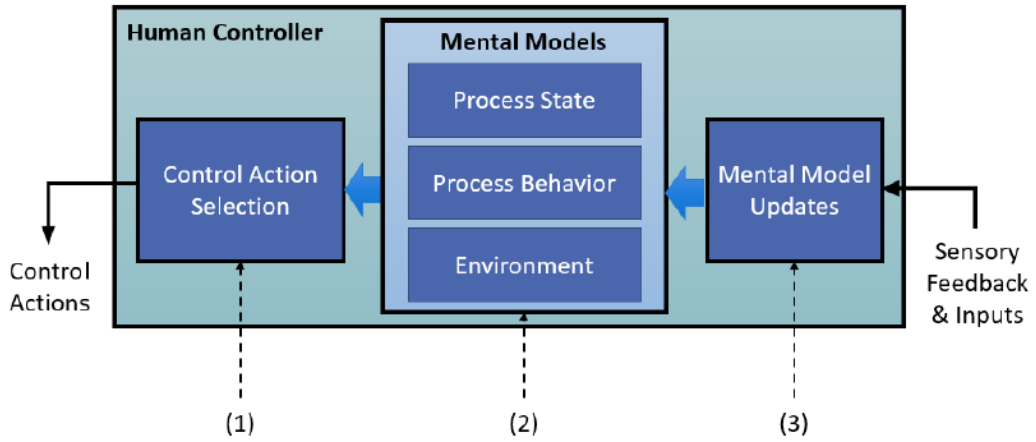

Figure 5. The Engineering for Humans Model [6]. Input (1) looks at how decisions are made, input (2) looks at our current beliefs, and input (3) looks at how we update those beliefs in light of new data. Except for the connections to the external environment, this model is not suitable to analysis by deviations (parameter + guideword). Instead, human factors aids will be used to evaluate the internal human model.

3.2 Drawing the Control Structure – two specific examples

Let's look at two specific instances of control structures that are derived from the generic **Figures 4 and 5**.

First will be a control room operator's response to abnormal situation **(Fig. 6)**. This control structure is intended to be a specific instance of **Figure 5**. In this model, operator behavior as to "why they might do or believe something" is a large part of this analysis in developing (and mitigating) the causal scenarios. You'll notice the need to incorporate human factors aids to help guide this STPA.

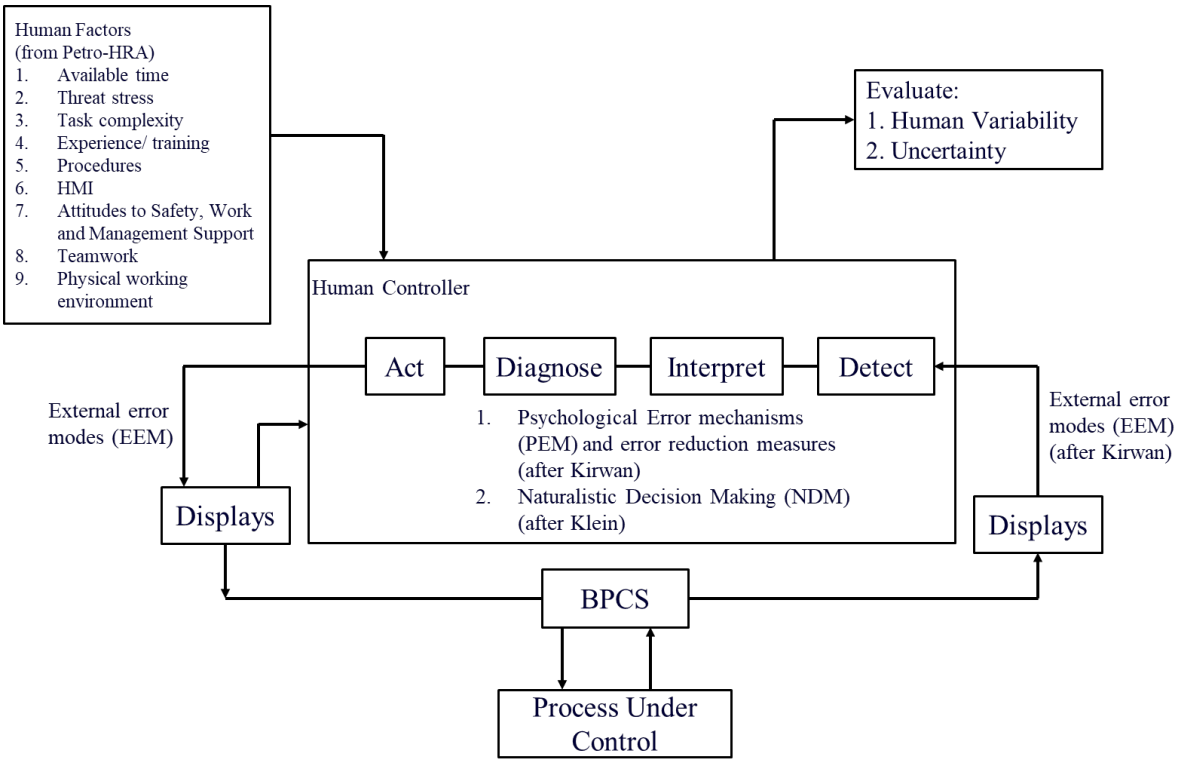

Figure 6. Specific example of the "STPA-Engineering for Humans" model. Console operator response to abnormal situation with potential for catastrophic outcome. Note that it draws on a variety of existing human factors tools and methods, such as Petro-HRA [12], Kirwan [13], and Klein [11]. Typical tools like HAZOP and FMEA do not provide the same look.

Recall from above that step 4 of the STPA method is writing the "loss scenario." This can be more of a narrative when performing "STPA-Engineering for Humans" versus documenting a hardware/ software centric control structure which is more amenable to the table format (see **Fig. 2**). When evaluating a human controller (i.e., operator) as shown in **Figure 6** especially when cognition (i.e., thinking) will be involved with responding to the abnormal event, understanding the operator's mental models can be an important aspect of the task analysis. An operator has two mental models – one of the process and one of the automation [6]. **Figure 7** presents a convenient way to categorize an operator's mental model to help guide the task analysis.

	Positive (+)	Negative (-)
System	How the system works: Parts, connections, causal relationships, process control logic	How the system fails: Breakdowns and limitations
Person	How to make the system work: Detecting anomalies, appreciating the system's responsiveness, performing workarounds and adaptations	How users get confused: The kinds of errors people are likely to make

Figure 7. The Mental Model Matrix after Gary Klein [10]. There will be one instance for the operator's mental model of the process, and one instance for the operator's mental model of the automation. This matrix can be used when discussing the abnormal event response with the operator to identify potential latent weaknesses in the system. Notice the mental model not only includes how something works, but also how it can fail.

The next example is intended to be a specific instance of **Figure 4**, the generic control structure. This is shown in **Figure 8**. In this example, the amount of cognitive related functions is much lower than in **Figure 6**. For example, a skill-based procedural task that is performed by a human. In this example as well as in **Figure 8**, human factors will play a significant role in evaluating and developing the causal scenarios related to the loss.

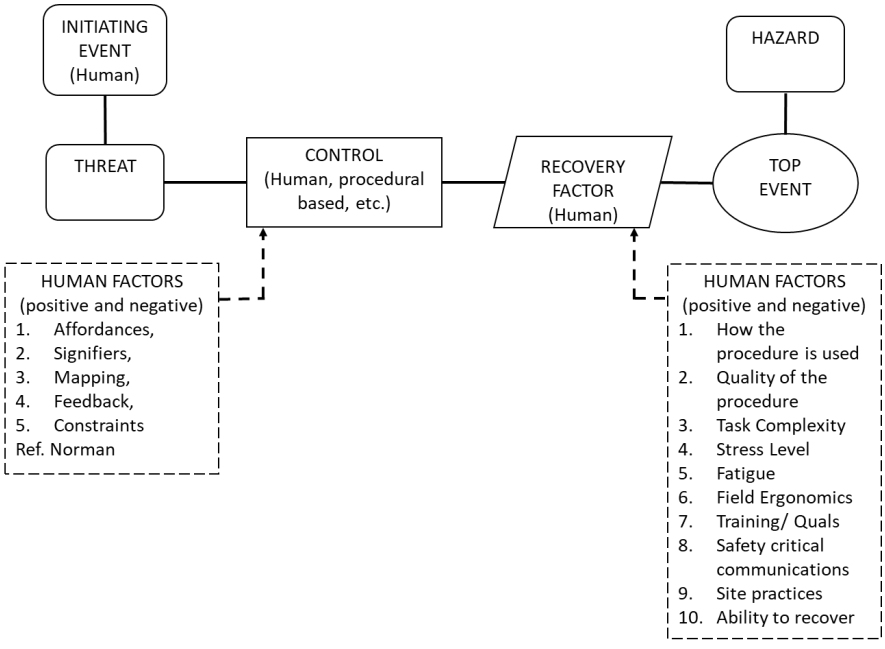

Figure 8. Specific example of the generic control structure drawn for an operation's Maintenance task that has process safety implications. Note that it draws on a variety of existing human factors tools and methods, such as Norman [5] and Swain [9]. Additional taxonomies can

be found in HFACS (Human Factors Analysis and Classification System) [7] and SHERPA (Systematic Human Error Reduction and Prediction Approach) [8]. Typical tools like HAZOP and FMEA do not provide the same look.

The clear value in evaluating the control structure is the qualitative Task Analysis component, to find and fix negative human factors associated with the task. To that end, the following human factors (after Norman [5]) should be investigated for maintenance and other non-routine tasks, for example, the one or more critical steps of the procedure (either as a cause of a catastrophic event, or response to it).

- **Affordances** are used to afford a location to perform work. A simple example is a valve handle. It provides a place for the operator to place their hands. Affordances can be used improperly, for example, an operator may have to stand on said valve handle (or the piping) to reach a manual valve above him. This may make the likelihood of a procedural error more likely.
- **Signifiers** tell the operator how to interact with the affordance. A round valve handle turns. A levered valve handle rotates. The best signifiers are intuitive (don't require conscious thought). Signs require effort and conscious thought (to see, read, and comprehend), and are therefore not the best use for signifiers (this is not to suggest getting rid of signs, but to understand the limitation). Populational stereotypes are learned signifiers, for example, "righty-tighty, lefty-loosey." Designs that violate populational stereotypes are considered error-likely.
- **Mapping** is used to spatially connect controls with affordances. For example, procedures and designs that use labels and drawings to map a procedural step to the actual equipment in the field.
- **Forcing functions** force desired behavior. For example, car-seals, chain locks, and captive key systems placed on valves are recognized as reducing errors of omission and commission related to valve manipulation.
- **Feedback** to allow an operator to recover from their error before something bad happens.

3.3 Drawing the Control Structure – A safeguard or IPL (Independent Protection Layer)

Lastly, a barrier control diagram is shown in **Figure 9** along with its safety constraints.

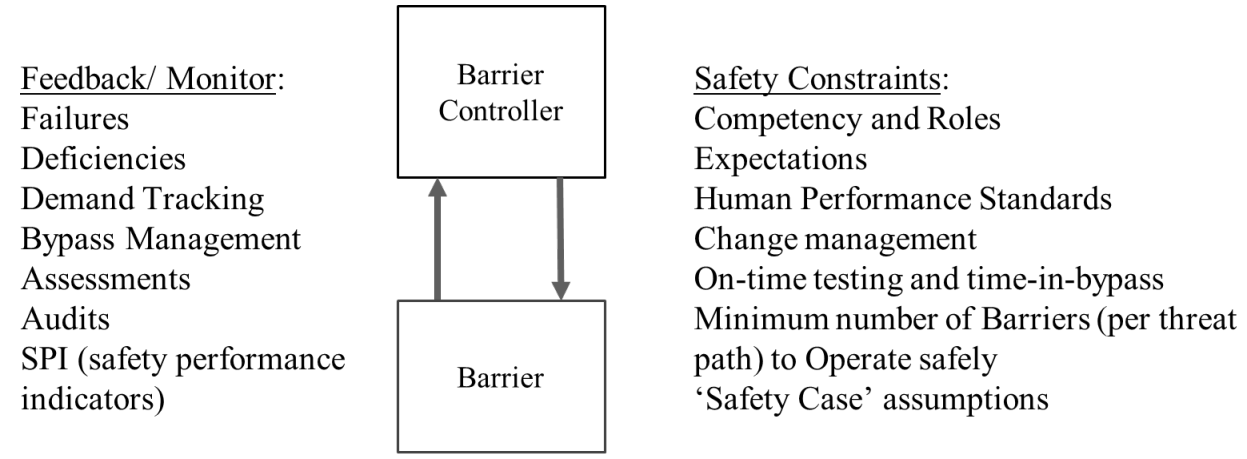

Figure 9. Barrier, safeguard, or IPL (Independent Protection Layer) safety constraints and feedback and monitoring control system [14]. Safety resides in the constraints (i.e., control actions). That is why there is only one guideword used in STPA – "unsafe control action" which looks at how and why the safety constraints may be violated.

4 Conclusion

This paper reviewed the new accident model known as STAMP and its primary tool for implementation known as STPA which is a hazards analysis tool. A more recently developed extension of STPA known as "Engineering for Humans" was also described. The value of STAMP/ STPA (for process safety) is not in having another tool to use to conduct a PHA (Process Hazard Analysis). Those methods already exist and have the capability to evaluate "unsafe control actions" as part of BPCS or human error, etc. initiating causes. However, STAMP/ STPA and its extension "Engineering for Humans" does provide an unrealized opportunity to evaluate human factors scenarios related to potential major accident hazards that are currently being ignored by the traditional methods. To be clear, methods such as PHA (Process Hazards Analysis) do consider the human element in terms of errors and mishaps that can occur (ideally for any mode of operation). However, the analysis stops there. What STPA and its extension "Engineering for Humans" do is look beyond stopping at "human error" (i.e., to see human error not as the cause of a scenario but as a consequence of the system, in order to identify latent conditions and systemic factors that make the potential for catastrophe more likely). Evaluating non-routine modes such as high-risk maintenance and abnormal situations is a good place to start. As discussed, STPA requires the potential loss to be identified up front. PHA will provide this. In addition, STPA will be executed as a separate study from PHA. Together, they will provide a more complete systems look at potential major accident hazards.

5 References

[1] Leveson, N.G., 2012. Engineering a Safer World: Systems Thinking Applied to Safety. MIT Press, Cambridge.

[2] Hollnagel, E., 2014. Safety-I and Safety-II The Past and Future of Safety Management, CRC Press, Boca Raton.

[3] Rasmussen, Jens, 1997, Risk Management in a Dynamic Society: A Modelling Problem, Safety Science Vol. 27.

[4] Chapman, Jake, 2004. System Failure, second ed., Demos, London.

[5] Norman, D., 2013. The Design of Everyday Things, Revised and Expanded ed, Basic Books, New York.

[6] France, Megan Elizabeth, 2017, Engineering for Humans: A New Extension to STPA, MIT.

[7] Wiegmann, D.A., Shappell, S.A., 2003. A Human Error Approach to Aviation Accident Analysis: The Human Factors Analysis and Classification System, Ashgate, England.

[8] Stanton, N.A., Salmon, P.M., Rafferty, L.A., Walker, G.H., Baber, C., Jenkins, D.P., 2013. Human Factors Methods – A Practical Guide for Engineering and Design, second ed. Ashgate, England.

[9] Swain, A.D., Guttmann, H.E., 1983. NUREG/ CR – 1278. Handbook of Human Reliability Analysis. U.S. NRC.

[10] Klein, Gary, 2021, The Mental Model Matrix: Important aspects of mental models often get ignored. Psychology Today.

[11] Klein, G. and Kahneman, D., Conditions for Intuitive Expertise A Failure to Disagree?, American Psychologist, October 2009.

[12] Institute for Energy Technology (IET), 2017. The Petro-HRA Guideline, Report IFE/HR/E-2017/001. Halden, Norway.

[13] Kirwan, B., 1994. A Guide to Practical Human Reliability Assessment. CRC Press, Boca Raton.

[14] Grattan, David, 2018. Improving Barrier Effectiveness Using Human Factors Methods, Journal of Loss Prevention in the Process Industries, 55 (2018) 400–410.

[15] Perrow, C., 1984. Normal Accidents - Living with High Risk Technologies. Princeton University Press, New Jersey.