



TOCICO is pleased to recognize this paper as part of the TOC Body of Knowledge.

Magic Druids

A Thinking Process tool for clarifying and solving recurring problems or conflicts

Theory of Constraints Body of Knowledge White Paper
By James Powell and Peter Cronin

(Note: Magic druids are also known as figure-8 loops, druid loops, or just druids.)

Sections:

Part 1 – Overview

- Background
- Current TOC Approach
- Application Testing and Adoption

Part 2 – Magic Druid Diagram

- Description
- Example

Part 3 – Case Study

- Abstract
- Introduction
- Method
- Results
- Discussion

Appendices

- Appendix A – Druids Which Made Up the Original List of Conflicts
- Appendix B – Grouped Causal Chains, Separated into Similar Themes

Part 1 – Overview

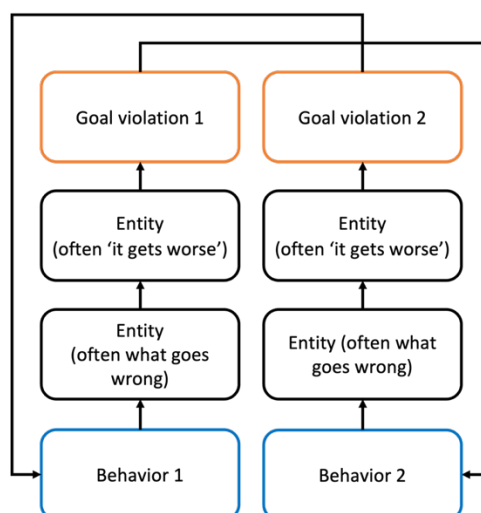
Background

The magic druid tool was originally created by extracting the mental process utilized by an expert in the Thinking Processes when working with a client. This mental process was developed into a framework that could be replicated by anyone, using logic structures that are part of the Thinking Processes (TP).

Druids do not address a deficiency in the toolset for TOC experts. They came about from observations in tool-use competency and adoption of learners new to the TP, as well as speed increases (with no quality loss) for experienced TP users.

The observation is that, while the evaporating cloud makes immediate sense to the uninitiated, it requires some focused practice to find sufficient fluency and confidence for people to use the tool themselves. Especially with other people, this leads to the new user deferring the attempt to use the tool until they have time to try to get it ‘right.’

As experts, we find it very easy to get participants in a conflict to articulate why the opposing behavior drives a need / goal violation. We tested teaching a person new to the TP to follow the process of describing why the other person’s behavior caused increasingly negative outcomes that culminated in a goal violation (equivalent to a need violation of a B or C in a cloud). We found that both sides could quickly and collaboratively do this. The resultant figure-8 loop consists of two negative causal branches (similar to NBRs) which are linked in that the goal being violated on one side compels the other side to do the behavior at the base of the opposing branch – known as the ‘magic druid.’ This is demonstrated in the diagram below.



It is then very easy to get a good quality cloud from the druid if necessary (see Appendix A). We found that the parties were also more collaborative and understanding of each other, as well as exhibiting an increased tendency to then use the tool immediately in a business context, with no further training or encouragement. This would spread around the company, resulting in it ‘going viral.’

Current TOC Approach

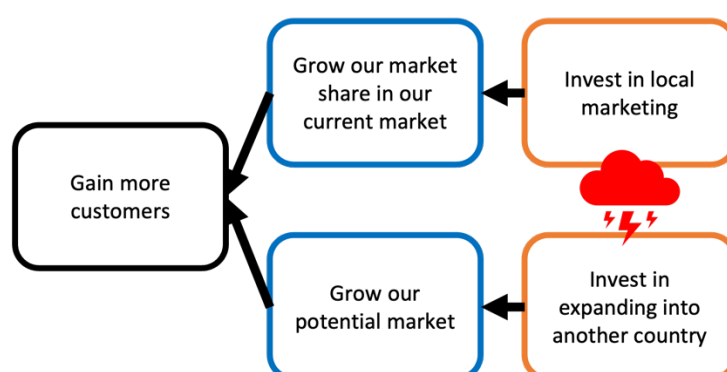
Clouds

The cloud is an effective tool for capturing and resolving conflicts. The cloud focuses on achieving win-win solutions, where the needs of both parties or both alternatives are met, in addition to the goal or common objective. However, the cloud does not provide the detail of current problems and how they go wrong, nor does it document the looping effect of recurring issues. Our experience shows that the druid is easier to use for those with no TP experience. We speculate that people find it easier, and quite natural, to use the druid to explain how a cause-and-effect branch of other's behavior causes things to go wrong. We also note that despite this accusatory approach to the initial use of the tool, once the druid is diagrammed, the same 'both sides together against the problem' mindset comes through, like with a cloud.

While a cloud or druid can be used for a conflict or dilemma, as shown in Appendix A, we have found that they are better suited for different situations once people are experienced in both. The druid is often the best choice when the scenario is understood quite well, is played out multiple times, and has a recurring loop. When one side is implemented, the negative effects cause people to switch to the actions and logic of the other side, resulting in a vicious loop.

Thus, a druid is best used where the scenario has played out multiple times and you really know the details. You can fill out the details of the steps along the way; you can add assumptions; you can apply the categories of legitimate reservations and get a clearer picture of exactly what's going on.

The cloud, however, is more suited to when you've got a forward-looking decision or a decision that doesn't have to be made all the time. When using a druid for such a situation, we must be careful we are not making up the detail/causality; we are essentially looking at NBRs. An example of this is a business considering its first major expansion efforts. The situation is forward-looking and focused on the benefits:



Comparing and Contrasting a Druid and Evaporating Cloud

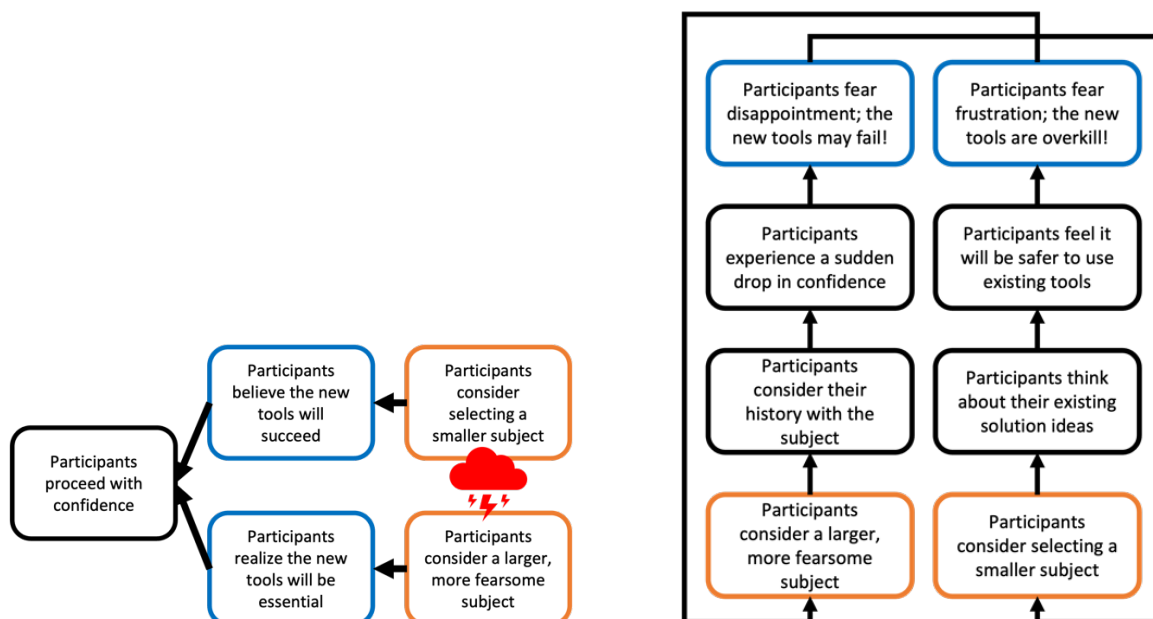
While the above states where clouds and druids tend to be best used; it may help the readers' understanding to see the similarities and differences of the cloud and druid compared directly.

An example of a dilemma faced by new students of the Thinking Processes on our Black Belt in Thinking course is used to illustrate this. Students consider the size and importance of a

subject to be analyzed using a Current Reality Tree (and solved with the rest of the Thinking Processes). Given this their first use of the tools, many would fear the tools would not be enough to solve their subject. On the other hand, they worry that a small subject would be a waste of a whole week of analysis. The situation can be presented as a cloud or a druid.

The conflicting behaviors of a druid are equivalent to the D and D' of a cloud. This is indicated in orange in the example below.

The goal violations of a druid can be represented as the B and C of a cloud. As they are written as a positive in a cloud, they must be worded as such and are delivered by the opposing behavior / D & D'. This is indicated in blue, below.



CRTs

The Current Reality Tree (CRT) documents the thorough detail of a negative situation in cause-and-effect. The tool provides an effective analysis of what is going wrong and why, and unless we identify a set of injections to solve it (and implement those injections), we are stuck in this situation. There is no substitute for this analysis on a system level.

However, the tool is too heavy for many to employ on a day-to-day basis. It takes substantial time to construct a CRT and it also requires considerable training, which many will not invest in for all staff. It would typically only be for strategic staff or management. The complexity of the tool also makes it difficult for many to use with an uninitiated audience or it leads to those formally trained in CRTs to never construct one again after the training.

Druids can be used to rapidly build a CRT. Multiple druids (usually 3 to 5) in a subject area are essentially a set of cause-and-effect branches for a CRT. The druids can be split in half, and then individual branches can be grouped by goal violations. Using the CLR, the branches can be expanded as well as interbranch links being created (these are called crosslinks). At the top of the CRT, two common and high-level goal violations (these are called system goal violations) are identified.

At the base of the CRT, the pressures that drive the behaviors can be summarized in a similar approach to the process in the three-cloud merge method. The system goal violations and pressures are then used to create the core problem cloud, leaving only the 'A' to be added.

Example

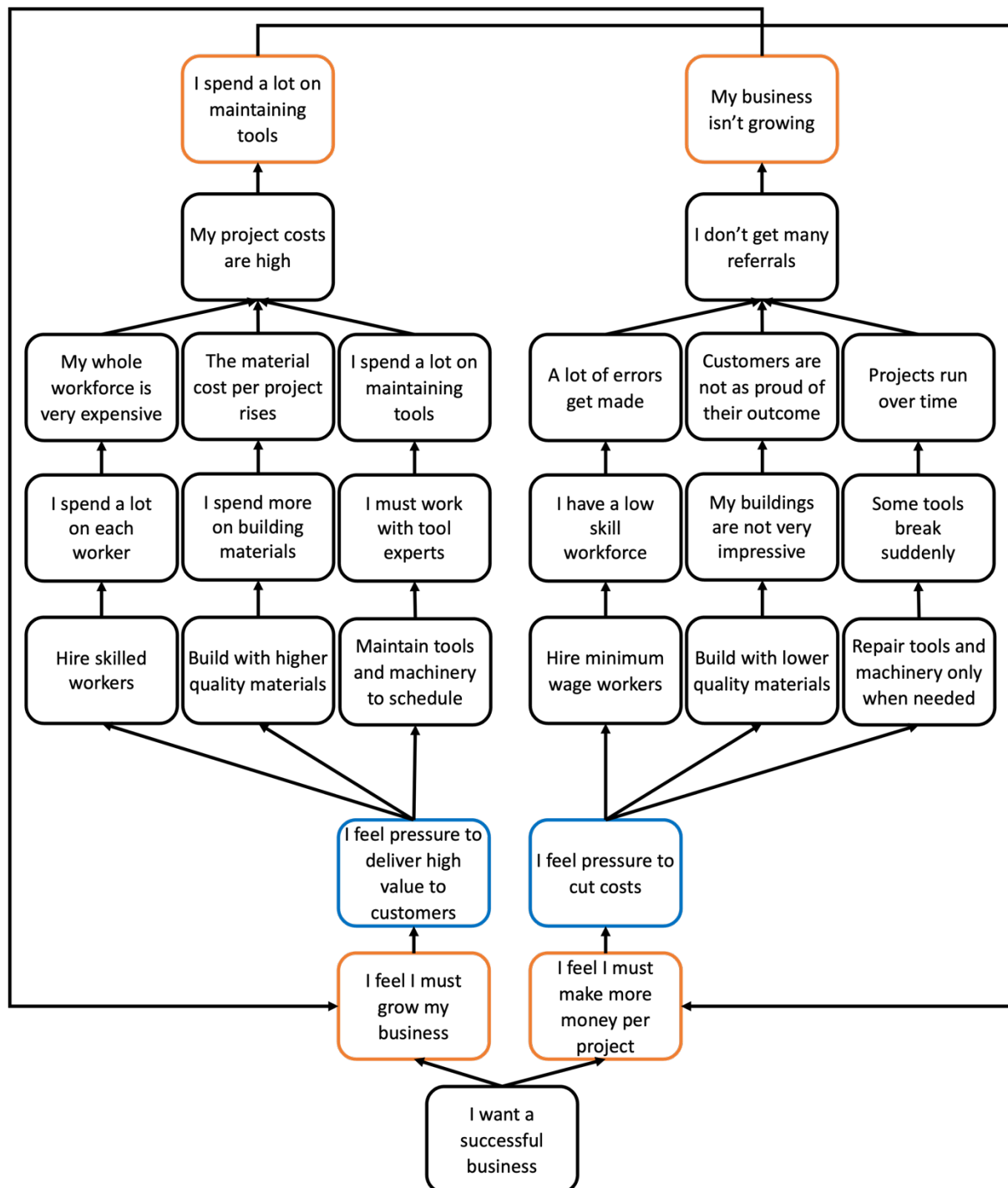
This is a basic CRT constructed from three druids. The CRT is kept very basic for this example to focus on the druids and how they come together to form a CRT. In practice, there is more detail to be added, and scrutiny applied via the categories of legitimate reservations. This ensures the logic of the analysis is robust.



Each side contains three branches, which are each half of a druid. These branches have been grouped based on commonalities in theme or goal violation. The branches are labelled to show which opposite branch they were originally paired with (as a druid). We can see in this example that there is a common goal violation on each side, making less money from projects on the left and the business not growing on the right. We can assume these to be the core needs the business is seeking to protect so we use these to determine the B and C of the core conflict.

The behaviors at the base of each druid branch can be used to determine the D and D' of the core conflict. The behaviors are summarized into core pressures which bridge the gap between the core needs (B and C) and the behaviors that drive each druid branch. In this case, the behaviors at the base of the left group of druid branches can be summarized as a pressure to deliver high value to customers. On the right-hand group, there is a pressure to cut costs. These pressures are the D and D' of the core conflict, caused by the core needs (B and C) and causing many individual behaviors, some of which are represented in the druid branches.

Once put together we get a CRT with a core conflict at the base, like so:



A drawback to this method is that some struggle with the crosslink stage, whereas the older UDE linking jigsaw shows the interconnectedness of the problems as a more obvious 'UDE soup.' In the example above, the six branches are independent and look like pillars, it is an additional step to add the detail of how entities within the tree cause each other. Experienced users find they can use druids to quickly form a CRT and still find crosslinks readily.

Our experience is that people who have been taught the druid CRT method use CRTs more frequently post-course, due to two reasons. Firstly, they can recall the process easier, as well as remember it as a more simple and approachable process. Secondly, the ability to easily scale down the CRT, say using only three druids, means people can quickly understand and solve the core problem in a smaller area that they would not use a larger process for.

In the last step of our training, participants form a three-druid CRT, identify injections, and test the logic as an FRT in a two-hour window.

Application Testing and Adoption

At the time of writing this paper, we have trained around five hundred people in the druid tool as part of either the Black Belt in Thinking Foundations (a day-to-day Thinking Process training course) or the Jonah-style Black Belt in Thinking Boot Camp.

After completing a live CRT build using druids, James Powell and Peter Cronin realized how effective the druid tool is and decided to try it on a BBIT course. To their surprise, it took a day off the CRT build stage, and increased quality! They have continued to use the druid-made CRT extensively and is far more dependable for new users – that is, less instructor oversight is required, and more graduates are using the tool after the training.

Those who had learned the previous way were shown the druid CRT method and switched to this method entirely. The previous way seemed less process-driven, and more time-consuming, although it does tend to show better interaction between the UDEs within the CRT.

WiseTech Global pivoted from clouds as the primary TOC tool for solving and documentation, to druids due to bringing in the CRT aspects to give more detail of the looping causality.

Part 2 – Magic Druid Diagram

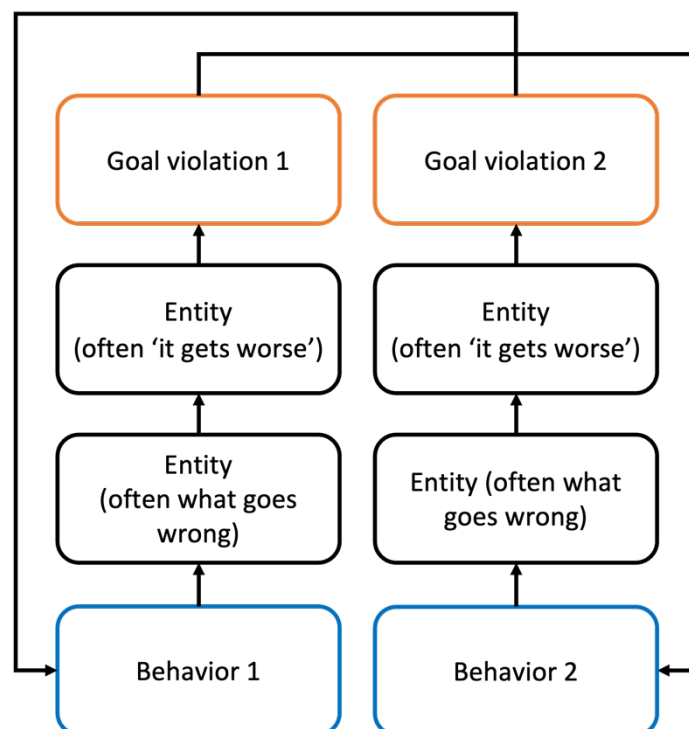
Description

The druid diagram is structured using sufficiency logic. The structure has two branches detailing the cause-and-effect. We use four entities on each side as a guide, as we found for most situations, only using three entities left out useful detail, and using five didn't add too much. The branches can have more or less than four and can be lopsided with more entities on one side than the other. Assumptions can also be added.

The base entities are behaviors, and the top entities are goal violations (often UDEs). The branches are linked by the loopbacks. The loop goes from goal violation on one side to the behavior on the other. For simplicity, the assumptions / additional entities are often not shown. It is usually taken as accepted that we have seen many times when 'X' goal is violated a person will do 'Y' behavior.

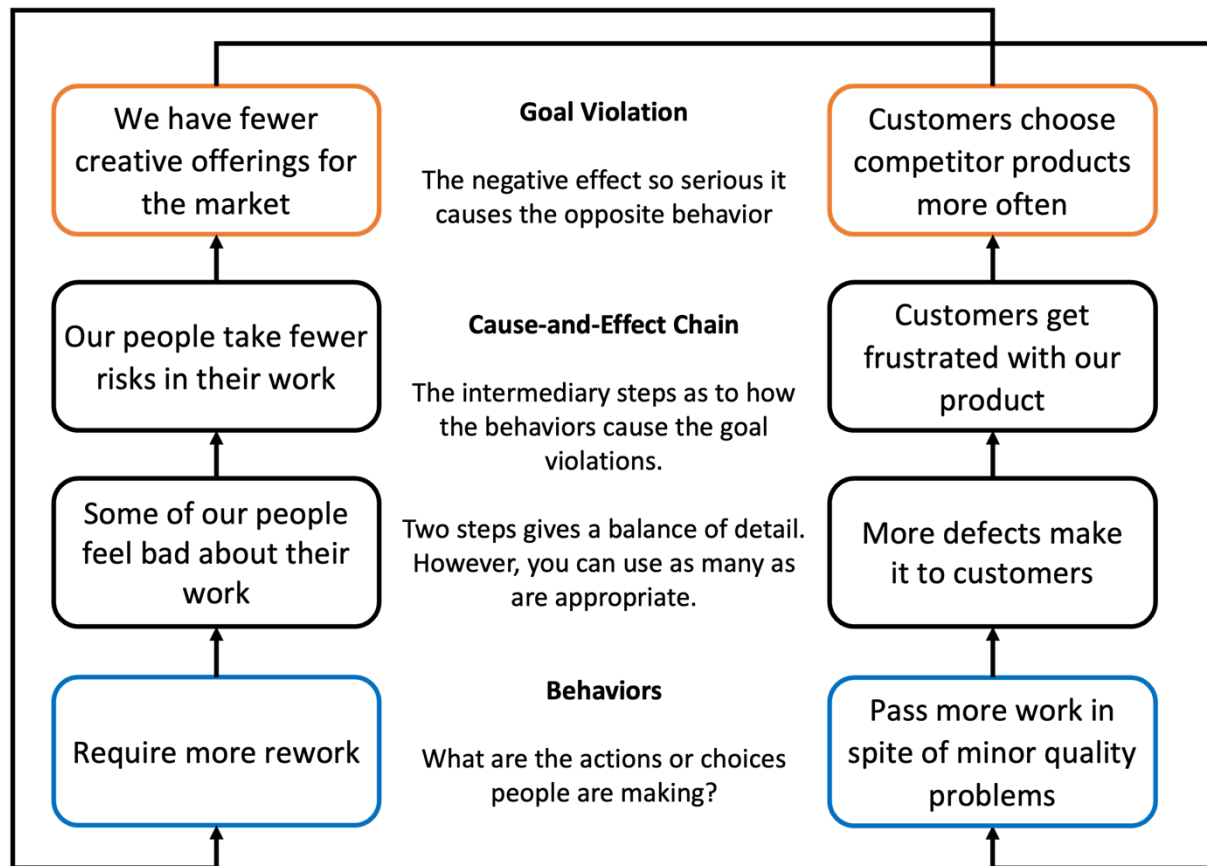
It can also be said, "If X goal is violated, and we have a policy or goal to protect X, then I will do Y, as Y is the best course of action I know to protect X."

For this reason, 'Policies, Measures, Consequence, Behaviors' can be used to explain the loop (with the behavior being the base entity, and the policy being the goal that is being violated). To solve this, the usual process of surfacing assumptions and identifying an injection is used, as with an NBR branch.



Example

In this example, a team is caught in a recurring dilemma between letting quality issues slide; and being stricter on quality. Over time, they have implemented stricter controls, only to find it leads to the team shutting down and harming creativity. Because of this, they have relaxed the controls, which then leads to customers turning away from their lower quality product, and around the druid they go.



Part 3 – Case Study

Abstract

This case study demonstrates not only the ability of the druid to break down problems and create a quick and effective Current Reality Tree, but also, the ability of a relatively new learner of the tool, to break down their problems and identify common themes on their own accord. This case study is based on solving real-life business problems and a new learner of the tool being able to create nine druid loops on various problems she was facing in her role.

Following a reality engineering process, those nine loops were then formatted in such a way that when presented to an expert in the Thinking Process tools, together (learner and expert) were able to identify the core problem, which in the learner's words became 'very obvious.'

Introduction

Cassie oversees marketing and product design at MyTreat and had been having issues to do with a lack of time in the business. There seemed to be a lot of work that needed to be done but not enough time to do it. So, she was trying to come to the business owner, James, with the problem and reasons why she was squeezed for time.

Cassie had started to learn the Thinking Process tools through the Black Belt in Thinking Foundations course, clouds, druids, etc. and it was suggested to her by a colleague that she put some of her new learnings to work and communicate the issues with James in his own language – druids. An important point to note here is that James did not ask for this way of communication. The idea to solve the situation using the tool and the resulting content was done by Cassie.

Method

Cassie created nine druids based on each of the nine issues that she was experiencing (Appendix A). The content of those druids is not important to this case study, although we have included the druids for your interest. What is important is constructing the druids, which compelled Cassie to give considered and balanced thought to what was bothering her prior to taking it up with her boss. She documented the two actions/behaviors of the recurring problems that she was facing. Following the druid construction process, Cassie proceeded to map out the logic of those actions; what went wrong, what got worse and finally what caused her to do an alternative action. She then proceeded to follow the same steps on the alternative action until she had her druid loop.

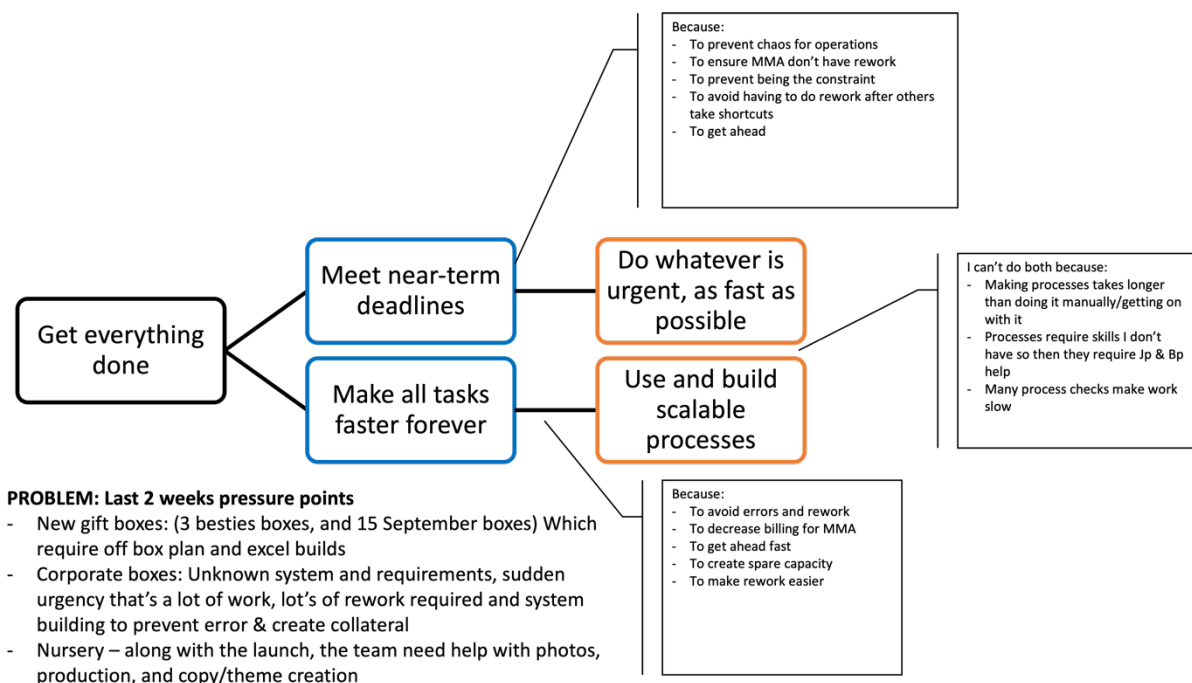
Working with her colleague, who had experience in the Thinking Process tools, Cassie then separated the nine druids in half (creating eighteen causal chains) and placed them on either the right or left side depending on what 'theme' she thought they fit in (Appendix B). This process enabled Cassie to group issues in the business that were similar, or that were causing similar negative side effects.

It was at this point Cassie shared her work with James. During this conversation, they worked together to review the separated druids in their 'left hand, right hand' groupings. They made minor changes, flipping the side a few causal chains were on, and color coding the themes.

Reading across the branches, two major themes/actions were then identified.
'Do whatever is urgent, as fast as possible' and *'Use and build scalable processes.'*

Taking these two major actions, they were then built into a cloud, with Cassie authoring the needs for those actions as well as the overall goal. Using the logic "In order to... we must... because..." Cassie was then able to create a list of assumptions on the connections that could then be used to break the cloud, knowing that she could now come up with a solution to the conflict whilst also protecting the overall goal and both needs.

Core Cloud – Expresses the Common Theme Across the Nine Druids



Results

Instead of providing a list of wants, or a list of obstacles, James got a considered picture of reality that showed why the answer was not obvious – because there are two sides to the story, which is demonstrated on nine fronts (through the nine druids).

From the theme grouping of the causal chains (left-hand, right-hand side), it was said to be glaringly obvious what the two themes of the problem were. Seeing the common themes separated in such a way, James and Cassie were able to dissect that the actual issue was that there was a sudden influx of tasks to Cassie that James was not aware of.

It was also identified that Cassie did not know that it was this influx that was causing all these problems to happen. Since neither had taken into consideration how much capacity those tasks would take, it made all the negatives (lack of time, errors, etc.) happen a lot quicker.

Using the druid tool, Cassie and James got to a core problem cloud very quickly, which provided Cassie with a way to produce a solution (through raising assumptions) that protects both needs. James, as a business owner, also benefits, as providing the structure for Cassie to solve the problem herself was a time-effective way of coming to a solution. The negative of not having this problem-solving ability is that Cassie would otherwise want James to adjudicate between the two actions. Now, both needs are protected, and James does not have to manually interfere for the right action to occur.

Discussion

From Cassie's point of view, the tools helped her simplify her situation and words. Instead of saying, "These are all my complaints," and giving a long list of reasons, she was able to put her problems into the framework, explaining the causality of the situation. The druid tool takes away a lot of pressure to carefully word the problem, as you are essentially filling in the framework and you share both sides of the issue with very little to no bias.

As someone who has newly learnt the tool, Cassie identified that communicating using druids was helpful, not only when communicating with someone else who knows the tools, but also, from an individual standpoint in terms of breaking down your problems.

"Instead of creating a storm about your problems and venting, the tools break down the problem into what is happening," said Cassie.

As a business owner, what James was always bothered by was, when in situations, people would ask him, "Do you want me to do this, or that?" and the answer is, "I want you to do both." When posed with that answer, most people respond by saying that they can't do both, which is then followed by a long-convoluted tale of woe that bears no resemblance to reality because it's not thought through. So, when comparing that with what Cassie did in this situation, it's a considered and careful analysis of the consequences of choosing one option over the other. James also has the ability to leave Cassie with the responsibility of protecting both sides instead of having to do that himself.

That's why it was powerful for James, he didn't have to make a ruling that he knew was going to have negative consequences.

James Powell's 2019 TOCICO presentation provides additional insights on the topic:

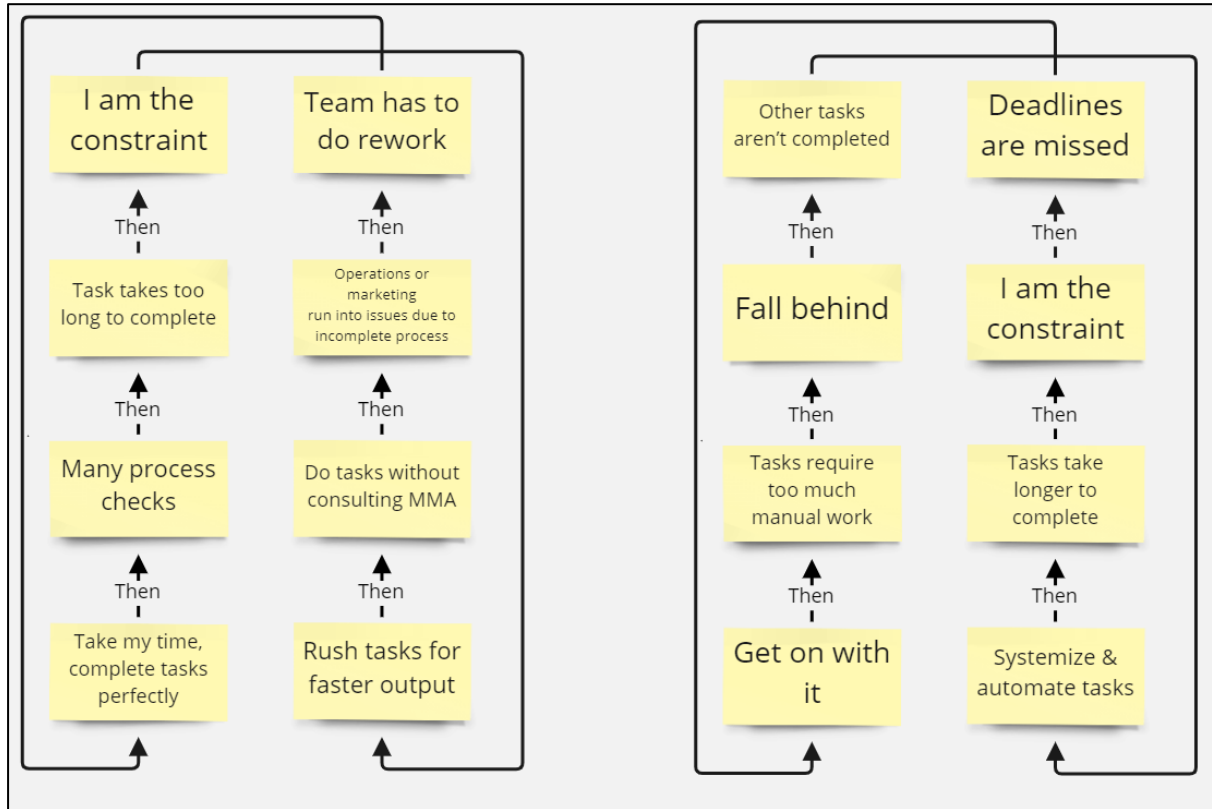
<https://www.tocico.org/page/2019VideoPowell>

Appendices

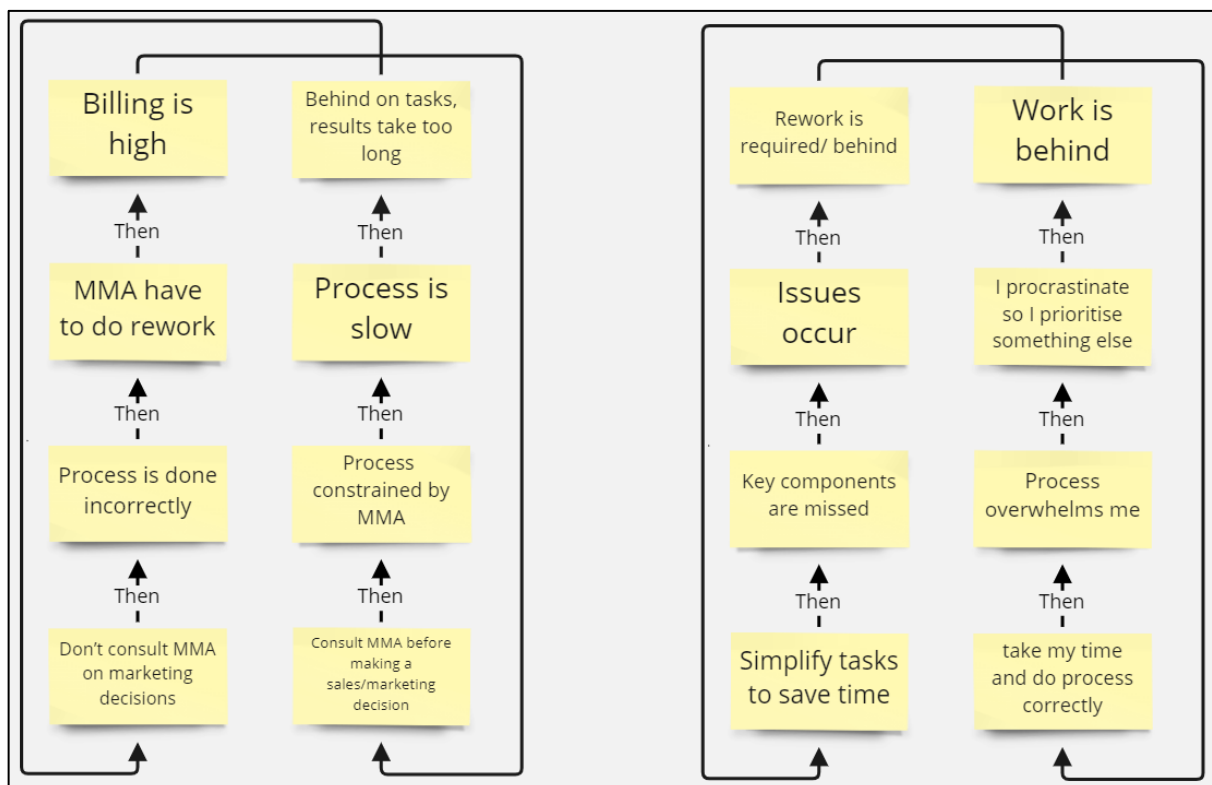
Appendix A

Druids which made up the original list of conflicts

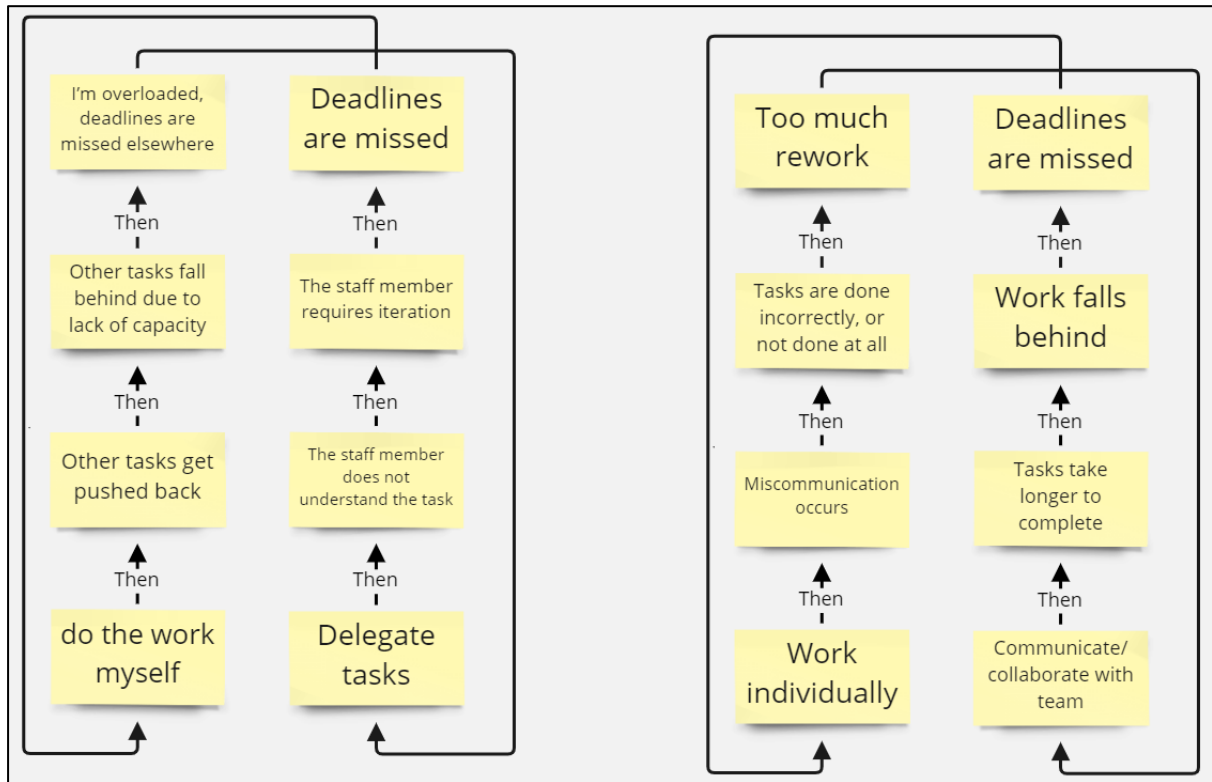
Druids 1 & 2



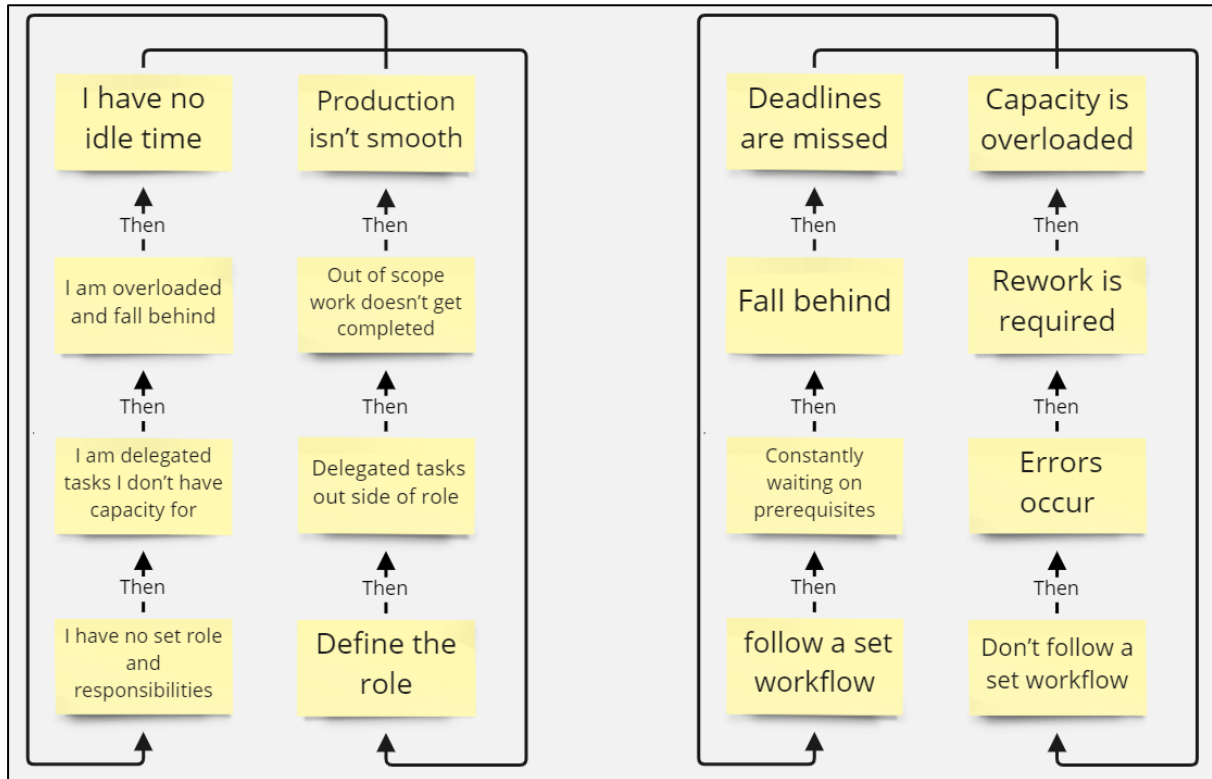
Druids 3 & 4



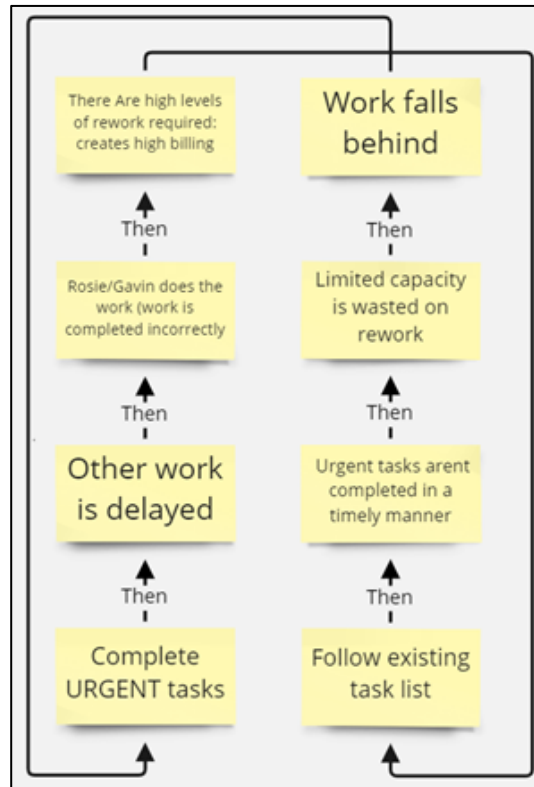
Druids 5 & 6



Druids 7 & 8



Druid 9



Appendix B

Grouped causal chains, separated into similar themes

