**Introduction to Systems Thinking - D372**

Task 3

**Shawn Watts**

Western Governors University

**Step 1**

**Iceberg Tool – Scenario 1**

1. **What happened?**

For Jaechap, the CEO mandated a new tool to keep up with the competition. Middle management passed this along to the team (Scenario 1).

* What did people do?
  + - The data analysts are resisting due to the time it takes to learn new software, based on experience, and the thinking that service quality will decline.
    - The CEO still wants to make this happen or fears that the company will fall behind.

1. **What changed or stayed the same?**

Currently, there is resistance to the new software. The data analysts will keep using it.

* + Anticipate
    - Service quality will suffer eventually due to limiting factors that prevent the use of new features from new software that customers will begin to demand. Thus, the competition will start to pull away.
    - Effort will continue to rise to try to compensate. However, quality will continue to decline.

1. **What patterns are present?**

There is a reinforcement pattern happening with the old software that is hard for the data analysts to break away from, because they manage customer requests very efficiently today with it. Because the CEO knows that the competition will soon pull away as the software being used today is dated. As time goes on, there will be a problem maintaining service quality with the current toolset.

* + Possible Solutions
    - Bring the data analysts into the process for choosing the new tool.
    - Use it yourself and highlight the possibilities.
    - Communicate the serious need for this. Show them it is an investment in themselves as well.
    - Mandate it.
    - Slow roll out, use both tools and transition as workload allows. Use the new software for new projects while maintaining legacy projects with the old software.

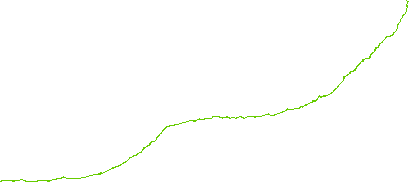
**Step 2**

**Behavior Over Time Graph**

**Time**

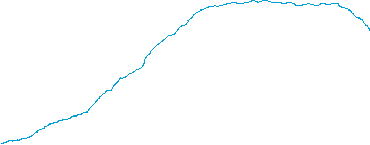
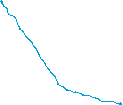
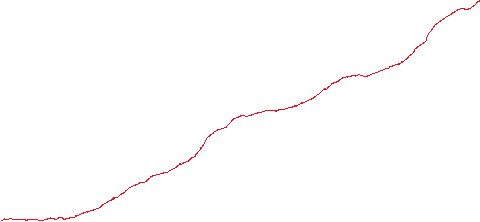
**Behavior**

Effort



Quality

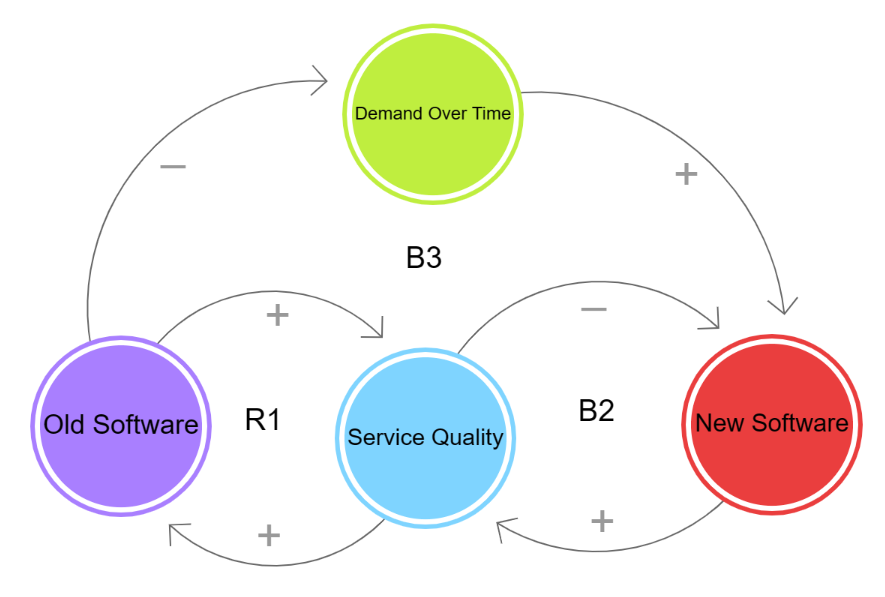
Demand



*Figure 1*

**Step 3**

**Causal Loop Diagram**



[*Figure 2*](https://ncase.me/loopy/v1.1/?data=%5b%5b%5b1,687,530,1,%22Service%2520Quality%22,4%5d,%5b2,484,522,1,%22Old%2520Software%22,5%5d,%5b3,918,521,1,%22New%2520Software%22,0%5d,%5b4,692,314,1,%22Demand%2520Over%2520Time%22,3%5d%5d,%5b%5b2,1,94,1,0%5d,%5b1,2,89,1,0%5d,%5b1,3,100,-1,0%5d,%5b3,1,80,1,0%5d,%5b4,3,101,1,0%5d,%5b2,4,116,-1,0%5d%5d,%5b%5b579,529,%22R1%22%5d,%5b801,523,%22B2%22%5d,%5b686,421,%22B3%22%5d%5d,4%5D)

1. **Archetype Chosen**

I chose the Limits to Success archetype. There is a constraint and a re-enforcing loop and balancing loop. It would be easy to use Shift the Burden / Addiction as well, as there is a fit for that too. In the end, I went with Limits to Success, because this is a scenario that has not occurred yet and we are exploring a future state of the business. Hopefully, this will flesh out problems and offer insights for solutions.

1. **Explain Choice Using Diagram**

As you can see in Figure 2, the software developers like using the old software (R1). To keep service quality up, they will continue to use the old software as for right now they are getting high service quality. At the same time, there is a constraint of customer demand (B3). Over time, if the analysts keep using the old software, they will not keep up with the customers and demand will start to faulter. When this happens, there will be less use of the new software as analysts will go with what they are most familiar with. Using the new software would help improve service quality over time (B2). Only the new software can keep up with the future customer demand. Due to the comfort of using the old software, the analysts will use it to deliver new product, thus the use of the new software will be negated. The main constraint here is that customers will demand new product that can only be delivered by the new software that other competitors are already using.

1. **Main Problem**

The main problem that needs to be addressed here is that data analysts need to start using the new software. Eventually, they will be forced to anyway as customers demand for new product will force their hand.

**Step 4**

1. **Solution**

I propose bringing in the team leaders and explaining the analysis and where the current trends will take us. Share with them this data and documentation and make them understand why they need to move over to the new software. For training, have an acceptable loss of productivity. A slow roll out here would help. There may be a time where both products are in play and teams split to specialize in one or the other. Ask for options from analysts and have the participate in the solutions.

1. **Strengths**

Being able to anticipate markets and understand how your customer might change is key to success. Bringing the leaders of the team in on this decision, will create buy-in and help alleviate friction and push-back. By not mandating the solution, it will feel more like a choice for the team.

1. **Challenges**

There is risk that you still may not convince the team. People are not math formulas and if you are not sincere in your delivery, people can sometimes tell. This failure might leave you worse off in the end. Training is also still a challenge and not addressed in the main solution. To even get there, we really need staff input and buy-in first.

1. **Alternatives Considerations**

I considered mandating the software and giving no choice to analysts, but this could make them jump ship and we would be worse off than before. My solution brings them into the discussion.

I thought about hiring some new people who only work on the new software. This could work, but it could also alienate your current staff. Again, no one likes being left behind.

Gamification is another solution I thought about. Give some large incentives for learning the new software. However, I feel like their motivation would still not have the basic understanding of why they need to do this.

### References

Karash, R. (2016, August 19). How to see “Structure.” The Systems Thinker. <https://thesystemsthinker.com/how-to-see-structure/>

Kim, D. (2018, March 23). Behavior over time diagrams: Seeing dynamic interrelationships. The Systems Thinker. <https://thesystemsthinker.com/behavior-over-time-diagrams-seeing-dynamic-interrelationships/>

H Kim, D. (1992). Systems Archetypes I: Diagnosing systemic issues and designing High-Leverage interventions. *The Toolbox Reprint Series*. <https://thesystemsthinker.com/applying-systems-archetypes/>

WGU. (2020). Introduction to Systems Thinking. <https://apps.cgp-oex.wgu.edu/wgulearning/course/course-v1:WGUx+OEX0102+v01/block-v1:WGUx+OEX0102+v01+type@sequential+block@1f3cc1fe0efb4e5cbb169acf0c0951b4/block-v1:WGUx+OEX0102+v01+type@vertical+block@79edae5127984499b969c05f1934a237>